FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Uncertainty interpretations for the robustness of object detection in self-driving vehicles

Filipa M. Ramos Ferreira

Doctoral Program in Informatics Engineering

Supervisor: Rosaldo J. F. Rossetti

January 22, 2025

Uncertainty interpretations for the robustness of object detection in self-driving vehicles

Filipa M. Ramos Ferreira

Doctoral Program in Informatics Engineering

January 22, 2025

Abstract

Ensuring the reliability and robustness of deep learning remains a pressing challenge, particularly as neural networks gain traction in safety-critical applications. While extensive research has focused on improving accuracy across datasets, generalisation, interpretability and robustness in the deployment domain remain poorly understood. In fact, in real-world scenarios, models often underperform without clear explanations. Addressing these concerns, uncertainty quantification has emerged as a key research direction, offering deeper insight into neural networks and enhancing confidence, interpretability, and robustness. Among critical applications, self-driving vehicles stand out, where uncertainty-aware object detection can significantly improve perception and decision-making.

This thesis explores interpretations of uncertainty tailored to object detection in the context of self-driving vehicles. In this sense, two novel methods to estimate the aleatoric component and one approach to modelling the epistemic uncertainty are proposed. Through the utilisation of anchor distributions readily available in any anchor-based object detector, uncertainty is estimated holistically while avoiding costly sampling procedures. Further, the concept of existence is introduced, a probability measure that indicates whether an object truly exists in the real-world, regardless of classification. Building upon these ideas, three applications of uncertainty and existence are explored, namely the *Existence Map*, the *Uncertainty Map* and the *Existence Probability*. Whilst the aforementioned maps encode the existence measure and the aleatoric uncertainty over the space of input samples, the Existence Probability merges the information provided by the Existence Map with the standard detections, supplementing model outputs. Evaluation showcases the coherence of uncertainty estimates and demonstrates the usefulness of the Existence and Uncertainty Map in supporting the standard model, providing open-set capabilities and giving a degree of confidence to true positives, false positives and false negatives. The merging strategy of the Existence Probability reports a considerable improvement in the performance of the object detector both in validation and perturbation, while detecting all classes of the dataset despite being trained only on cars, pedestrians and cyclists.

The second part of this thesis features a study of the underspecification distribution and its connection with the epistemic component of uncertainty. Underspecification, recently coined, greatly endangers deep learning deployment in safety-critical systems as it depicts the variability of predictors generated by a single architecture with increasingly diverging performance in application domains. The analysis performed showcases that, if the uncertainty estimates are correctly calibrated, a single predictor is sufficient to predict the spread of the underspecification distribution, avoiding running repeated costly training sessions. All proposed methods are designed to be *model-agnostic*, *real-time* compatible, and seamlessly applicable to *deployed models* without requiring retraining, underscoring their significance for robust and interpretable object detection in autonomous driving.

Resumo

Assegurar a fiabilidade e robustez de *deep learning* continua a ser um desafio, particularmente com a pressão para utilizar redes neuronais em contextos de segurança crítica. Apesar da investigação extensiva a melhorar a eficácia destes modelos em variados *datasets*, a generalisação, interpretabilidade e robustez no domínio de aplicação continuam pouco entendidas. Em cenários do mundo real, é frequente modelos falharem ou terem eficácia sub-par sem explicação aparente. Tendo em conta estas preocupações, quantificação de incerteza emerge recentemente como uma direção chave de investigação, oferecendo uma vista mais aprofundada de redes neuronais e melhorando a confiança, interpretabilidade e robustez destes sistemas. Dentro das aplicações de segurança crítica, condução autónoma destaca-se como uma área onde deteção de objetos consciente de incerteza pode significativamente melhorar a perceção e a tomada de decisão destes veículos.

Esta tese explora interpretações de incerteza especificamente desenhadas para deteção de objetos no contexto de condução autónoma. Neste sentido, dois novos métodos para estimar o componente aleatórico e uma abordagem para modelar a incerteza epistémica são propostos. Através do uso de distribuições de âncoras já disponíveis em qualquer detetor de objetos baseado neste mecanismo, a incerteza é estimada holisticamente evitando procedimentos custosos de geração de amostragem. Para além disto, o conceito de existência é introduzido, uma medida probabilística que indica se um objeto realmente existe no mundo real, independentemente da sua classificação. Construindo a partir destas ideias, três aplicações de incerteza e existência são exploradas, em concreto o *Existence Map*, o *Uncertainty Map* e a *Existence Probability*. Enquanto que os mapas de existência e incerteza codificam a medida de existência e a incerteza aleatórica no espaço dos exemplos de entrada, a probabilidade de existência combina a informação disponibilizada pelo *Existence Map* com o retorno standardizado do modelo, suplementando-o. A avaliação mostra a coerência das estimativas de incerteza e demonstra a utilidade dos mapas de existência e incerteza a suportarem o modelo standardizado, permitindo capacidades de *open-set* e atribuindo um grau de confiança a verdadeiros e falsos positivos e a falsos negativos. A estratégia de fusão da probabilidade de existência reporta uma melhoria considerável na performance do detetor de objetos tanto em validação como em perturbação, enquanto deteta todos os tipos de objetos apesar de o modelo ter sido apenas treinado em carros, peões e ciclistas.

A segunda parte desta tese apresenta um estudo sobre a distribuição de *underspecification* e a sua conexão com a incerteza epistémica. *Underspecification*, demonstrada recentemente, gravemente ameaça a utilização eficaz de *deep learning* em sistemas de segurança crítica visto que descreve a variabilidade de *predictors* gerados por uma arquitetura com desempenhos divergentes no domínio de aplicação. A análise mostra que, com estimativas de incerteza corretamente calibradas, um único *predictor* é suficiente para prever dispersão, evitando custosas e repetidas sessões de treino. Para além disto, todos os métodos propostos são desenhados para serem independentes de modelo, compatíveis com processamento em tempo-real e aplicáveis a modelos já em utilização sem requerir novo treino, reinforçando a sua relevância para deteção de objetos robusta e interpretável em veículos autónomos.

Acknowledgments

First of all, I would like to thank my supervisor, Professor Rosaldo Rossetti, for always trusting me and going the extra mile to support me in every way. You are not just my supervisor, but a great friend! Secondly, I am thankful for my parents, for always being there at the ready to help. My brother, thank you for the insights and discussions, especially in the mathematical notation.

Finally, I have to thank my husband and my beautiful daughter with all of my heart. This time of our lives was the most transformative and one I will never forget. Despite the hardness, we blossomed into life, became parents and now take on the ever growing challenges of the future. This PhD is merely the stepping stone that now, once again, leads us into the unknown.

Filipa M. Ramos Ferreira

"It's a dangerous business, Frodo,
going out your door.
You step onto the road, and if you don't keep your feet,
there's no knowing where you might be swept off to."

J.R.R. Tolkien

Contents

List of Figures

List of Tables

Abbreviations and Acronyms

AD	Autonomous Driving
AI	Artificial Intelligence
AU	Aleatoric Uncertainty
BbB	Bayes by Backprop
BNN	Bayesian Neural Network
BS	Brier Score
EU	Epistemic Uncertainty
ECE	Expected Calibration Error
ENCE	Expected Normalised Calibration Error
FN	False Negatives
FP	False Positives
IN	In-Distribution
IoU	Intersection over Union
MAE	Mean Absolute Error
MC-Dropout	Monte Carlo Dropout
MLE	Maximum Likelihood Estimation
NLL	Negative Log-Likelihood
OD	Object Detection
OOD	Out-of-Distribution
PCC	Pearson's Correlation Coefficient
RGB	Red Green Blue
RoI	Region of Interest
SBN	Stochastic Batch Normalisation
TN	True Negatives
TP	True Positives
TU	Total Uncertainty

Chapter 1

Introduction

1.1 Context

Artificial Intelligence has established itself as a significant trend of the 21st century. In particular, as the effectiveness of neural networks spreads to a wide panoply of fields, novel AI-based systems find large adoption in the public. Tools like *ChatGPT* [OpenAI (2022)] have opened the trust of end-users in these technological developments, with the urge for artificial systems increasing exponentially in every day life.

Within the current panorama of Artificial Intelligence surfaces a domain with extensive research history, that of Sustainable Cities. Including a wide range of challenges still to be fully addressed, one field within Sustainable Cities has sported high traction lent from industry, research and consumer demand - that of autonomous driving [McKinsey (2023); Dingyi et al. (2018)]. Despite the numerous pop culture depictions of self-driving vehicles, the development of the technology has proven increasingly challenging. Not only are these vehicles safety-critical systems, numerous problems have also to be tackled. In this sense, perception tasks stand as a staple in the autonomous driving pipeline [Willers et al. (2020)].

Perception in autonomous driving aims to construct a reliable and detailed representation of the dynamic environment surrounding the vehicle through the collection of sensory data. Despite recent advances in perception tasks supported by the increasingly effectiveness of deep learning [Minaee et al. (2022); Xie et al. (2022); Zaidi et al. (2022)], there are still several dangerous characteristics of neural networks that need to be addressed.

In fact, deep learning has often been criticised for its *black box* condition [Rudin and Radin (2019)]. The fact is that, even for practitioners and scientists, it is hard to predict the generalisation capability of a developed architecture. Even more daunting, the task of determining what the model *does not know* proves considerably difficult. Stemming from these characteristics, it is very hard to explain failure conditions to end-users. Even more, generic neural networks only produce point estimates, not offering much information about the overall knowledge that the model has attained. As such, despite the push for AI-based facilities and the remarkable accuracies sported

by neural networks, adoption in self-driving vehicles is a highly complex problem [Bonnefon et al. (2016); Shariff et al. (2017); Awad et al. (2020)].

These characteristics have been thoroughly observed in state-of-the-art architectures. Nguyen et al. (2015) demonstrate that neural networks can output high class confidence even on unrecognisable samples. Szegedy et al. (2014) showcase that, with slight alterations to the input, models can change their classification drastically. Consider, for example, a model that is trained to distinguish cats from dogs in images. In the deployment domain, an image of a monkey may be fed to the system. Since standard outputs are in the form of categorical distributions, the model is unable to communicate or recognise that it can not answer [Weideman (2016)].

On another level, most research developing state-of-the-art models focuses on dataset-dependant metrics [Noor and Ige (2024)]. Moreover, D'Amour et al. (2022) demonstrate an alarming phenomenon, underspecification, that plagues most machine and deep learning pipelines. Training a deep learning architecture, even with slight changes to its hyperparameters, leads to the generation of divergent predictors with significantly different performance in deployment domains. These aspects also greatly hinder the robustness and interpretability of neural networks as there is no insurance that an input will lead to a predictable output. For a safety-critical system like a self-driving vehicle, where human lives may be at risk, these characteristics are inadmissible.

Focusing in one of the most prominent perception tasks, this thesis develops around the problem of object detection. Further than the aforementioned characteristics expressed by the formulation of neural networks, even highly performing models are unable to completely avoid erroneous outputs. Situations such as *misclassifications*, *incorrect heading and location regressions* and even entirely *missed or incorrectly signalled objects* are impossible to fully eradicate. Especially considering the dynamic environment around self-driving vehicles, the randomness induced by sensor data collection and the ever-changing representations of objects in the real world, reaching a *perfect* model is likely to even be impossible. Shafaei et al. (2018) enumerate the central aspects of the applicability of deep learning to autonomous driving to be the incompleteness of training data, possible distributional shifts, the differences between training and operational environments and the uncertainty of prediction, all relevant in generic machine and deep learning applications. In this sense, object detection models based on deep learning architectures have to be increasingly more robust and interpretable, to ensure safety and trust from end-consumers [Dingyi et al. (2018)].

1.2 Uncertainty, Robustness and Object Detection

Addressing the aforementioned challenging aspects of deep learning, several auxiliary research directions have emerged in recent literature [Zhang et al. (2021); Salehi et al. (2021); Yang et al. (2024a)]. One prominent field is uncertainty quantification for deep learning, which aims to identify and measure sources of uncertainty in neural networks. Through the availability of uncertainty estimates, greater insight into the outputs and inner workings of models can be gained.

Consider once more the previously explored scenario of a classification model trained to distinguish between cats and dogs being fed the sample of a monkey. To classify the sample, the

model outputs a categorical distribution. Through uncertainty quantification, this distribution can be calibrated, for example, outputting equal probabilities for cat and dog, thus communicating that the network is unaware of which object features in the sample. Other options include supplementing the categorical distribution with confidence intervals or entropy-based information metrics, all adding to the outcome of the neural network that previously could only attribute an uncalibrated probability to each pre-defined, closed-set class. As such, the availability of uncertainty estimates could greatly impact and mitigate the aforementioned undesirable characteristics of deep learning, improving robustness and interpretability [Goan and Fookes (2020); Feng et al. (2021); He and Jiang (2023)].

Despite the recent developments in the field, there are still many challenges to address, particularly when integrating uncertainty quantification with object detection. In fact, current object detection models still face limitations in their ability to provide probabilistic, robust and interpretable outputs [Feng (2021); Hess et al. (2022)]. Some of these weaknesses stem from the cost of current uncertainty quantification methods. Since a predictive distribution needs to be sampled, costly sub-runs [Gal and Ghahramani (2016); Ayhan and Berens (2018)], sub-parametrisations [Guo et al. (2017); Teye et al. (2018)] or sub-models [Lakshminarayanan et al. (2017); Matsubara et al. (2020)] are employed, heavily increasing the time complexity of these models. Real-time compatibility, on the other hand, is essential to self-driving vehicle deployment.

Further than this, there are a large panoply of uncertainty interpretations available, with no common definition agreed upon in literature [Gal and Ghahramani (2016); Malinin (2019); Liu et al. (2020); Lahlou et al. (2023)]. Most authors sub-divide uncertainties in two parcels, *aleatoric*, namely *data-dependent* and irreducible, and *epistemic*, *model-dependent* and reducible. Even considering the same components, some works connect the aleatoric uncertainty with the conditional entropy [Ash (1965); Malinin (2019)], whilst others interpret it as the variance of the predictive distribution, possibly being approximated directly through the loss function regressed during training [Kendall and Gal (2017)]. This divergence in interpretations makes it difficult to assess the suitability of uncertainty quantification methods for specific applications.

Moreover, many methods to quantify uncertainty induce substantial changes in architectures, forcing adaptation and retraining sessions [Lakshminarayanan et al. (2017); Teye et al. (2018)], especially to ensure estimates' quality. For already deployed models, retraining is often impractical or infeasible, posing a barrier to real-world adoption. Even after several retraining sessions and changes, the quality of the estimated uncertainties is not easily assessed, often reliant on parametrisations, making it hard to justify its implementation.

Pertaining to the layed out limitations of uncertainty quantification and object detection, the problem statement for this thesis may be resumed as follows: to study and propose interpretations of uncertainty for deep learning-based object detection that are *model-independent*, *real-time* compatible and applicable to already *deployed models* without retraining, as well as to demonstrate applications of uncertainty that enhance robustness and interpretability. In this sense, various challenges of object detection and uncertainty quantification are to be addressed, namely the enabling of probabilistic, better calibrated outputs that can classify object existence beyond closed-

set classification and interpret uncertainty within predictions. These methods are to address the limitations of traditional detection systems by providing additional information on both the presence of objects in a global concept, ultimately making object detection outputs more robust and interpretable.

1.3 Contributions and Opportunities

From the previously discussed ideas, several opportunities arise. Firstly, the possibility to study object detection specific uncertainty interpretations with increased adequacy to the problem. Further, to address and thoroughly evaluate performance and uncertainty estimates in standard and perturbation data, reporting on robustness. Moreover, to supplement the output of object detectors with meaningful and useful information that deliberates on existence with a global conceptual representation of objects. Lastly, to reflect upon undesirable characteristics of neural networks such as underspecification, and to find methods to diffuse their impact during deployment.

Building upon these opportunities and taking into consideration the aforementioned problem definition, the contributions are *three-fold*. Upon the domain of uncertainty quantification, the proposal of two novel aleatoric uncertainty interpretations and one epistemic estimation method specific to region proposal-based object detectors. Exploring then applications of uncertainty to improve robustness and interpretability, the introduction of the *Existence Map*, the *Uncertainty Map* and the *Existence Probability*. The *Existence Map* expresses a probabilistic outcome on the existence of objects regardless of class, supplementing the model's standard output by giving a degree of confidence to true positives, false positives and true negatives. The *Uncertainty Map*, that encodes aleatoric uncertainty over the space of the input sample and communicates the certainty of the model in a human-readable manner, directly impacting interpretability. Lastly, the *Existence Probability* that merges Existence Map information with the standard output of the object detector, recalibrating softmax confidences and considerably improving robustness in validation and perturbation modes. Upon the domain of underspecification, presenting a study on the relationship between the epistemic uncertainty and the underspecification distribution, showcasing that one single predictor is sufficient to approximately regress the spread of accuracy of an architecture in in-and out-of-domain data. Finally, all proposals are conditioned to be *model-independent*, *real-time* compatible and applicable to *deployed models* without the need for retraining.

1.4 Thesis Structure

The remainder structure of this thesis is as follows:

Background In the Background 2 chapter, all the theoretical foundations to support the discussions in this thesis are layed out, from the exploration of neural networks and their probabilistic component to the statistical basis of many techniques that are employed to quantify uncertainties

in deep learning. The chapter is closed with a report on the domains of interest, that of object detection, robustness and interpretability.

Uncertainty and Deep Learning Chapter Uncertainty and Deep Learning 3, addresses uncertainty quantification for deep learning, starting with the demonstration of the existence of different types of uncertainties and how these naturally occur. Finally, uncertainty interpretations found in the literature are compared, demonstrating how these are used in many different contexts and applications.

Interpretations of Uncertainty and Applications The Chapter Interpretations of Uncertainty and Applications 4 contains the main proposals of this thesis. These include two novel interpretations of the aleatoric parcel and a method to estimate the epistemic component. Applications of uncertainty estimates are also proposed, enabling the improved robustness and interpretability of deep learning-based object detection.

Object Detection in Autonomous Driving The chapter Object Detection in Autonomous Driving 5 presents the experimental validation of the proposed uncertainty interpretations and derived applications in autonomous driving contexts. This evaluation concerns the assessment of the coherence of uncertainty estimates as well as the performance of its reported applications.

Underspecification and Interpretations of Uncertainty In the Underspecification and Interpretations of Uncertainty 6 chapter, a study is presented on the connection between the epistemic uncertainty and the underspecification distribution. The validity and calibration of the proposed epistemic interpretation is also assessed.

Conclusions The concluding chapter, Conclusions 7, summarises the contributions, discusses limitations and addresses possible directions of exploration in future work.

Chapter 2

Background

In this chapter, the foundational concepts to the understanding of the discussions in this thesis are layed out. Since a panoply of topics are covered, this chapter is broad in scope and discusses many different concepts. These concepts are an integral part of the proposals of following chapters, and, as such, the background theory that concerns them must be carefully demonstrated.

Section 2.1 provides useful remarks on the notation that is to be used throughout the chapter. Concerning the background concepts, starting with deep learning, a foundational approach to neural networks is given in Section 2.2, including standard architectures, loss functions and training procedures. Following this, a demonstration of a probabilistic interpretation of a neural network is provided in Subsection 2.2.2.

After having discussed the base of neural networks, some uncertainty quantification topics are explored in Section 2.3. In this section, the entropy as considered in information theory is explained as it is a central theme in this thesis. Following, an illustration of bayesian inference and variational inference is given, concepts essential to the evaluation of existent uncertainty approaches. Finally, the section is closed with the demonstration of how a model of uncertainty may be derived from the training procedure of a neural network as well as the addition of closing remarks.

Moving from uncertainty, another central subject in this thesis is that of object detection and autonomous driving. For this reason, Section 2.4 contains background information on object detection, ranging from standard architectures to anchor proposals and an overview of possible datasets to evaluate this task in autonomous driving contexts.

Finally, Section 2.5 discusses the interpretability of neural networks, demonstrating how this condition is reflected in recent state-of-the-art models. The central topics of object existence and underspecification are also explained, other two key concepts in following chapters.

2.1 Remarks on notation

In this chapter, some mathematical foundations of the discussed topics are presented. In order to standardise notation, there are some remarks on the use of common representations to be made.

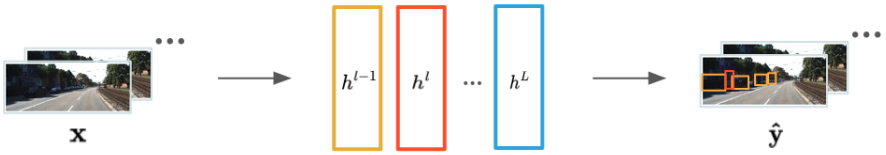


Figure 2.1: Neural networks are composed of several L hidden layers. These layers apply non-linear transformations to the input sample, composing the final predictor $\hat{\mathbf{y}}$.

The data source, commonly called a dataset in deep learning, is referred to as $\mathscr{D}$. The dataset is composed of a set of input samples, X, and respective labels, Y. This means that x_i represents a sample from $\mathscr{D}$ with a corresponding ground-truth label, y_i, with $i \in \{0, 1, ..., n\}$. In some equations, x and y are also used to represent a sample from X and its given label in Y.

For classification problems, y_i is a categorical variable from a range of accepted classes. In regression settings, labels are assumed to belong to a continuous distribution with restrictions on its representation being implemented as intervals.

The provenience of X is not discussed nor relevant for the information presented in both this and the following chapter. As such, the reported formulae can be applied to any input domain with little to no adjustment. Some diagrams showcase examples of application based upon images as the input. This only specifies the input manifold as $x_i \in \mathbb{R}^{l \times h}$, where l is the length and h the height of the image, and is the chosen representation to ease human understanding.

In all equations, the set of weights of the network are referred to as θ and are interchangeably called parameters of the model. Some techniques require the use of extra parameters, and these are represented as ω. This necessity arises, for example, when approximating a custom distribution (with parameters ω) to the distribution of weights of the network (parametrised by θ).

All predictors, either models or estimated values, are signalled with a hat in superscript.

2.2 Neural Networks

Neural networks typically represent a highly non-linear, complex function. This function is generally a mapping from the subspace of X to the subspace of Y in relation to a set of parameters. The output of a neural network may thus be visualised as in Equation 2.1.

$$\hat{\mathbf{y}} = f_\theta(x) \tag{2.1}$$

where f_θ is the output of the neural network parametrised by θ.

Deep learning models are structured as a sequence of hidden layers, as showcased in Figure 2.1. Layers apply non-linear transformations to the input sample, as described in Equation 2.2. A neural network architecture can thus be defined as a specific amount of layers alongside the

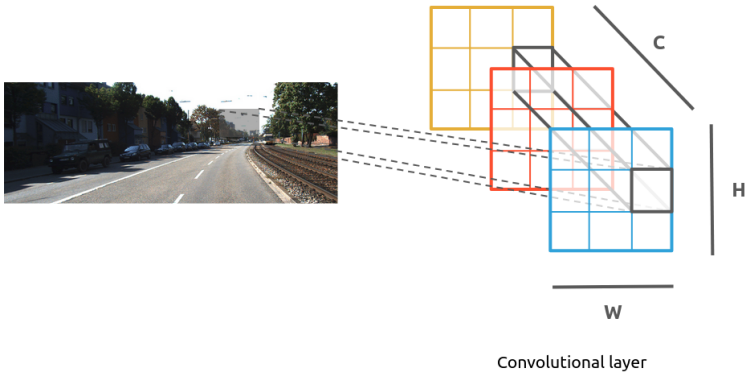


Figure 2.2: Convolutional layer filtering a 2-dimensional input sample.

operations performed within each.

$$\begin{aligned} \mathbf{h}^0 &= f(x;\theta^0) \\ \mathbf{h}^l &= f(\mathbf{h}^{l-1};\theta^l) \\ \hat{\mathbf{y}} &= f(\mathbf{h}^L;\theta^L) = f_\theta(x) \end{aligned} \tag{2.2}$$

The learning process intends to encounter the best set of parameters that fit the training data $\mathscr{D}$ through the minimisation of an objective function, namely the loss. Loss functions are adapted to each problem as these control the optimisation procedure.

2.2.1 Convolutional Layer

As discussed, the hidden layers are the building blocks of a neural network. There are many different types of layers in literature and the operations that are performed within each depend highly on the desired application. Most Computer Vision tasks are tackled with Convolutional Neural Networks. Since this field is the central focus of this thesis, the discussion of a convolutional layer is essential to enable a deep understanding of the concepts discussed further along.

Figure 2.2 demonstrates the operations performed in a convolution for a 2-dimensional sample input. Convolutions filter the input, thus transforming it and revealing structural and semantic information of interest. For this reason, convolution inherently fits feature extraction in visual manifolds Zhao et al. (2024). Typical deep learning architectures for visual tasks are composed of many convolutional hidden layers where each further filters the output of the previous layer.

The matrix, or, as typically called, the tensor, resulting from a convolutional layer $\mathbb{H}^l$ will be 3-dimensional with the dimensions corresponding to the height, width and channels, namely $H^l \times W^l \times C^l$. Each height times width slice of this tensor represents a feature map, and there are as many feature maps as channels admitted, C^l.

Typical convolutional blocks also feature other important operations, such as pooling, activations and normalisation. Figure 2.3 contains a brief demonstration of these concepts.

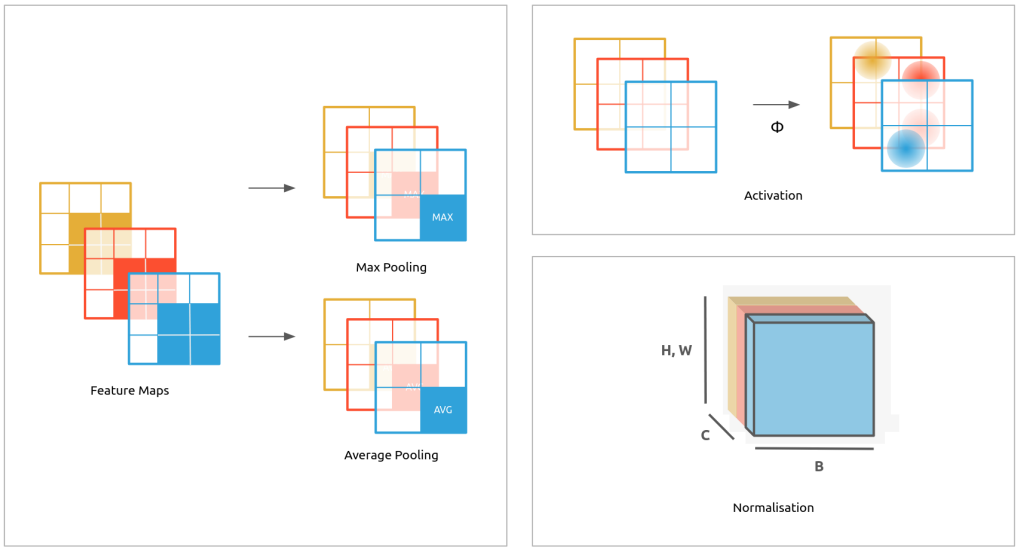


Figure 2.3: Operations typically performed in convolutional blocks of neural networks for vision applications.

Pooling intends to summarise the features obtained during the convolution, thus reducing the dimensions of the feature maps and consequently computation efforts. Common approaches apply maximum or average pooling although other methods do exist [Gholamalinezhad and Khosravi (2020)]. Another operation is the activation, a common mechanism in all types of hidden layers. Activations introduce non-linearities and combine inputs with weights [Dubey et al. (2022)]. Some common activations are shown in Equation 2.3.

$$\begin{aligned} \phi_{sigmoid}(x) &= \frac{1}{1+\exp(-x)} \\ \phi_{ReLU}(x) &= max(0,x) \\ \phi_{LReLU}(x) &= max(\beta - x, x); 0 < \beta < 1 \end{aligned} \tag{2.3}$$

Finally, the normalisation operation is discussed. Normalisation helps stabilise training and regularise outputs, enabling better generalisation. Figure 2.3 demonstrates Batch Normalisation on the bottom right corner. Batch Normalisation [Ioffe and Szegedy (2015)] updates the moving average used to regularise inputs with the mean and standard deviation of each batch, as demonstrated in Equation 2.4.

$$\begin{aligned} \mu_B &= \frac{1}{|B|}\sum_{b=1}^{|B|} B_b \\ \sigma_B^2 &= \frac{1}{|B|}\sum_{b=1}^{|B|} (B_b - \mu_B)^2 + \varepsilon \\ BNorm(B) &= \gamma \odot \frac{B - \mu_B}{\sigma_B^2} + \beta \end{aligned} \tag{2.4}$$

The output of a convolutional layer may thus be reduced to the following mathematical transformation:

$$\begin{aligned} \mathbb{H}^l &= \phi(f_{conv}(H^{l-1})) \\ \text{with } H^{l-1} &\in \mathbb{R}^{H^{l-1}\times W^{l-1}\times C^{l-1}} \end{aligned} \tag{2.5}$$

where ϕ is the chosen activation function and $f_{conv}(H^{l-1})$ is the function that maps the inputs of the convolutional layer to the resulting feature maps after convolution, pooling, normalisation, and any other operations involved.

2.2.2 A Probabilistic Interpretation

Previously, a formal definition of a neural network was given. In order to ascertain and reflect upon uncertainty in deep learning, it is also relevant to present a probabilistic interpretation of a neural network.

Consider a finite dataset $\mathscr{D} = \{x_i, y_i\}_{i=0}^n$ as the training data. This dataset is naturally sampled from an underlying distribution, referred to as $\mathscr{D}_i r$, as its pattern may be represented by a mathematical function.

Contrary to Equation 2.1, a probabilistic view of a neural network assumes that the predictor is approximating the conditional distribution of the targets in function of the input data and the set of parameters of the network, as depicted in Equation 2.6.

$$\hat{\mathbf{y}} \approx P(Y|X,\theta) \tag{2.6}$$

Both in classification and regression settings, the training objective is to maximise the likelihood of the observed data. This can be achieved through Maximum Likelihood Estimation (MLE), as depicted in Equation 2.7.

$$\begin{aligned} \hat{\theta} &= \arg\max_{\theta}(P(Y|X,\theta)) \\ &= \arg\max_{\theta}(\prod_{i=0}^{n} p(\hat{y}_i|x_i,\theta)) \end{aligned} \tag{2.7}$$

Depending on the setting, the maximum likelihood may be estimated and optimised during training in different forms.

Classification For a classification problem, the maximum likelihood estimation culminates in the product of an indicator function that analyses the correspondence between the predicted categorical variable and its ground-truth target, as described in Equation 2.8.

$$\hat{\theta} = \arg\max_{\theta}(\prod_{i=0}^{n}\prod_{c=1}^{k} \delta_{y_i,\text{Cat}_k(x_i)}) \tag{2.8}$$

where $\delta_{y_i,\text{Cat}_k(x_i)}$ corresponds to the following indicator function:

$$\delta_{y_i,\text{Cat}_k(x_i)} = \begin{cases} 1, & \text{if } y_i = \text{Cat}_k(x_i) \\ 0, & \text{otherwise.} \end{cases} \tag{2.9}$$

with $\text{Cat}_k(x_i)$ being the categorical classification of sample x_i.

Since the obtained expression is not ideal for optimisation, the training procedure usually follows the minimisation of a logarithmic function of the likelihood, namely the negative log-likelihood, as reported in Equation 2.10.

$$\hat{\theta} = \arg\min_{\theta}(\underbrace{-\sum_{i=0}^{n}\sum_{c=1}^{k}\delta_{y_i,\text{Cat}_k(x_i)}\ln p(\hat{y}_i = \text{Cat}_k(x_i)|x_i,\theta)}_{-\mathscr{L}(x,y,\theta)}) \tag{2.10}$$

Dividing this loss by the number of samples n does not impact the location of the minimum, allowing the loss to be interpreted as the expectation over the true underlying distribution of the data $\mathscr{D} \sim p_{true}(x,y)$, as shown in Equation 2.11.

$$\begin{aligned} \hat{\theta} &= \arg\min_{\theta}(-\frac{1}{n}\sum_{i=0}^{n}\sum_{c=1}^{k}\delta_{y_i,\text{Cat}_k(x_i)}\ln p(\hat{y}_i = \text{Cat}_k(x_i)|x_i,\theta)) \\ &= \arg\min_{\theta}(\underbrace{\mathbb{E}_{\hat{p}_{true}(x,y)}[\mathscr{L}(x,y,\theta)]}_{\mathscr{L}(\mathscr{D},\theta)}) \end{aligned} \tag{2.11}$$

After training, the classification model will be a representation of a distribution $p(Y|X,\theta)$, which is transformed through the received observations of X. To this end, the softmax likelihood is generally assumed for classification tasks, represented through Equation 2.12.

$$p(y|x,\theta) = \frac{\exp\{f_\theta(x)\}}{\sum_{i=0}^{n}\exp\{f_\theta(x_i)\}} \tag{2.12}$$

Regression On a regression setting, on the other hand, the maximum likelihood estimation may progress through the minimisation of the expectation of the negative log likelihood, as depicted in Equation 2.13.

$$\begin{aligned} \theta^* &= \arg\min_{\theta}(\frac{1}{n}\sum_{i=0}^{n}\underbrace{-\ln p(\hat{y}_i|x_i,\theta)}_{\mathscr{L}(x,y,\theta)}) \\ &= \arg\min_{\theta}(\underbrace{\mathbb{E}_{\hat{p}_{true}(x,y)}[\mathscr{L}(x,y,\theta)]}_{\mathscr{L}(\mathscr{D},\theta)})) \end{aligned} \tag{2.13}$$

The resulting regression model also represents the distribution $p(Y|X,\theta)$. Regression tasks can be represented by a Gaussian likelihood, as demonstrated in Equation 2.14.

$$p(y|x,\theta) = \mathscr{N}(y; f_\theta(x), \tau^{-1}I) \tag{2.14}$$

where τ represents model precision. Often times, the distributions being modelled are highly complex and not capturable by a Gaussian. As such, the common approach is to model a mixture of known distributions.

2.3 Preliminaries on Uncertainty

Recall Bayes rule in a deep learning context through Equation 2.15.

$$p(\theta|\mathscr{D}) = \frac{p(\mathscr{D}|\theta)p(\theta)}{p(\mathscr{D})} \tag{2.15}$$

$$\text{with } \mathscr{D} = \{x_i, y_i\}_{i=0}^{n},$$
$$p(\theta|X,Y) = \frac{p(Y|X,\theta)p(\theta)}{p(Y|X)}$$

where $\mathscr{D}$ is the dataset, X are the input samples, Y the corresponding labels and θ are the model parameters (weights).

Bayes rule is one of the building blocks of modern statistics. Also in uncertainty quantification, several methods rely in its interpretation of probability. In fact, calculating the posterior in Bayes theorem would provide a solution to the uncertainty estimation problem.

Building upon this central idea, an introduction to a probabilistic interpretation of a neural network is discussed, followed by an overview of Bayesian and Variational inference respectively.

2.3.1 Entropy

The entropy as discussed in this thesis refers to the concept of measuring the expected amount of information that is necessary to transmit the state of a variable, taken from Information Theory [Shannon (1948)]. As such, entropy is relative to a random variable and quantifies the information associated with the variable's possible outcomes. Entropy can be calculated as the sum of probabilities multiplied by the logarithmic probability over the space of the variable states, as described in Equation 2.16.

$$\begin{aligned} \mathrm{H}_\Psi &= \mathbb{E}[\mathrm{I}(\Psi)] = \mathbb{E}[-\log p(\Psi)] \\ &= -\sum_{\mathrm{z}\in\Psi} p(\mathrm{z}) \log_b p(\mathrm{z}) \end{aligned} \tag{2.16}$$

The base, b, of the logarithm controls the scaling of the resulting information. When using a base of two, for example, the entropy corresponds to the number of bits that are needed to convey the state of the random variable.

Consider a coin that is not fair and that is tossed with known probabilities. Imagine the base two logarithm is chosen to convey the result. When the coin is tossed and results in a probability of $\frac{1}{2}$, we are at the maximum amount of information necessary to convey the outcome, that is one

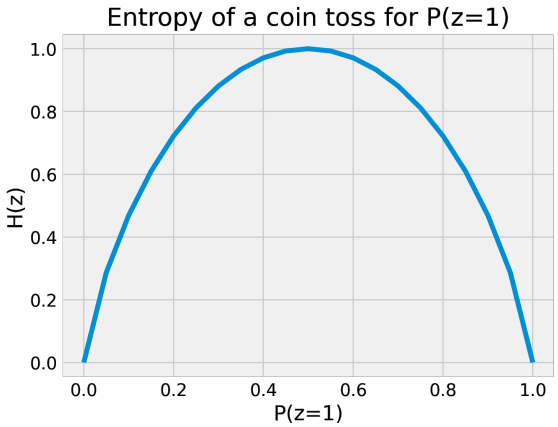


Figure 2.4: Entropy of a coin toss for $p(\mathrm{z}) = 1$.

full bit. Since both categorical outcomes are equally probable, the result, heads or tails, must be conveyed in full. When the probability reaches one or zero, the amount of information needed to transmit the result will always be zero bits, since the outcome is known at all times. This relation is better seen when plotting the entropy for this example, as shown in Figure 2.4.

The formulation of entropy has some useful properties, some of which are highlighted as follows:

$\mathrm{H}_\Psi \leq \log_b |\Psi|$ Entropy is maximal when all sampled outcomes are equally probable.

$\mathrm{H}(S,T) = \mathrm{H}(S|T) + \mathrm{H}(T)$ The entropy of joint events corresponds to the entropy of one random variable plus the entropy of the other in function of the first.

$\mathrm{H}(S,T) \leq \mathrm{H}(S) + \mathrm{H}(T)$ The entropy of joint events corresponds at most to the sum of the entropy of both variables, reaching equality when both random variables are independent.

2.3.2 Bayesian Inference

Following Bayesian theory, a *prior* - $p(\theta)$, *posterior* - $p(\theta|X,Y)$ and *likelihood* - $p(Y|X,\theta)$ can be defined in the context of neural networks. Likelihood has been previously discussed, with its form being dependent on the task. Weight distributions can be initialised incorporating previously obtained knowledge, namely a prior. This is the case in Bayesian Neural Networks that assume a distribution over each parameter instead of a point estimate [Buntine and Weigend (1991); Neal (1996); Phan et al. (2019); Goan and Fookes (2020)].

In a Bayesian framework, predictions of target variables can be obtained by applying Equation 2.17.

$$p(Y|X,D) = \mathbb{E}_{p(\theta|\mathscr{D})}[p(Y|X,\theta)] \tag{2.17}$$

Some uncertainty quantification techniques make use of the concept of having a prior belief upon the parameters of the model, using thus Bayes rule to obtain an estimation of the posterior.

2.3.3 Variational Inference

> "For most probabilistic models of practical interest, exact inference is intractable, and so we have to resort to some form of approximation." *Bishop* Bishop (2006)

We have mentioned that obtaining the posterior would give a solution to the uncertainty quantification problem. However, its analytical calculation is generally intractable in statistics. This is due to the denominator in Bayes Rule, the marginal likelihood, that is obtained through Equation 2.18.

$$P(Y|X) = \int p(Y|X,\theta)p(\theta)d\theta \tag{2.18}$$

This integral becomes intractable as the number of dimensions in X grows, deeming its analytical calculation impossible. In deep learning, this is a more prominent issue since generally X is highly dimensional and even if it is not, we typically have no knowledge over the underlying distributions.

As such, in order to infer the posterior, it becomes invaluable to study and develop approximation techniques. This subsection provides a brief description a very common approximation method, that of Variational Inference.

In Variational Inference, a distribution, $q(\omega)$, with a different set of parameters, is optimised in order to approximate the posterior $p(\theta|X,Y)$. For this purpose, a divergence metric that quantifies the *closeness* between distributions must be utilised.

In Equation 2.19 is presented an asymmetric measure of the divergence between distributions, the Kullback-Leibler Divergence. It should be noted that the KL-divergence is asymmetric since $D_{KL}(q(\omega)||p(\theta|X,Y))$ is different than $D_{KL}(p(\theta|X,Y)||q(\omega))$.

$$D_{KL}(q(\omega)||p(\theta|X,Y)) = \int q(\omega)\log\frac{q(\omega)}{p(\theta|X,Y)}d\omega \tag{2.19}$$

Using this divergence metric, the optimisation objective thus becomes to estimate the parameters of the approximated distribution $q(\omega)$:

$$\begin{aligned}
\omega^* &= \arg\min D_{KL}(q(\omega)||p(\theta|Y,X)) \\
&= \int q(\omega)\log\frac{q(\omega)}{\frac{p(Y|X,\theta)p(\theta)}{p(Y|X)}}d\omega \\
&= \int q(\omega)(\log q(\omega) - \log p(Y|X,\theta)p(\theta) + \log p(Y|X))d\omega \\
&= \int q(\omega)\log\frac{q(\omega)}{p(Y|X,\theta)p(\theta)}d\omega + \log p(Y|X)\int q(\omega)d\omega \\
&= \int q(\omega)\log\frac{q(\omega)}{p(Y|X,\theta)p(\theta)}d\omega + \log p(Y|X)
\end{aligned} \tag{2.20}$$

Even though Equation 2.19 is very important for the approximation process, we cannot compute it as the posterior is unknown. For this reason, we look for methods in which to eliminate the posterior from our measure function. In this sense, we use Equation 2.15 and substitute the posterior in D_{KL}, obtaining the relation in Equation 2.20.

The final form of the objective demonstrates that the $\log p(Y|X)$ quantity is a constant in respect to our optimisation goal. As such, we eliminate it from our cost function. Note that the removal of this component can be demonstrated through the use of the evidence lower bound (ELBO).

The optimisation function can then be defined as follows:

$$\begin{aligned} F(Y,X,\theta) &= \int q(\omega)\log\left(\frac{q(\omega)}{p(\theta)}\right) - q(\omega)\log p(Y|X,\theta)d\omega \\ &= D_{KL}(q(\omega)||p(\theta)) - \mathbb{E}_{q(\omega)}[p(Y|X,\theta)] \end{aligned} \tag{2.21}$$

For more fine-grained information on this technique including application to neural networks refer to [Jordan et al. (1999); Graves (2011); Hensman et al. (2012); Hoffman et al. (2013); Zhang et al. (2019)].

2.3.4 Preliminaries of a Model of Uncertainty

In Section 2.2.2, a probabilistic interpretation of a neural network was discussed. As seen in Equation 2.11 and Equation 2.13, both in classification and regression, the optimisation objective may be interpreted as finding the minimum expected value of the negative log likelihood over the true underlying distribution of the data.

In order to enable the formulation of the estimation of uncertainty that is discussed in depth in the next chapter, a brief demonstration of a derivation of the expectation of the loss in relation to the true underlying distribution of x is carried out for both classification and regression tasks. This derivation is of relevance as it proves the foundation of many of the concepts that are introduced in further chapters.

In the following equations, t_c stands for $\text{Cat}_k(x)$, that is, the categorical classification of x for class c.

Classification Taking the expectation of the classification loss depicted in Equation 2.11 with respect to the true underlying distribution of x gives the following relation:

$$\begin{aligned}
\mathscr{L}(\mathscr{D}_{true},\theta) &= \mathbb{E}_{p_{true}(x)}[\mathbb{E}_{p_{true}(y|x)}[-\sum_{c=1}^{k}\delta_{y,\mathrm{Cat}_k(x)}\ln p(\hat{y}=t_c|x,\theta)]] \\
&= \mathbb{E}_{p_{true}(x)}[-\sum_{c=1}^{k}\mathbb{E}_{p_{true}(y|x)}[\delta_{y,\mathrm{Cat}_k(x)}]\ln p(\hat{y}=t_c|x,\theta)] \\
&= \mathbb{E}_{p_{true}(x)}[-\sum_{c=1}^{k}p_{true}(t_c|x)\ln p(t_c|x,\theta) \\
&= \mathbb{E}_{p_{true}(x)}[\sum_{c=1}^{k}p_{true}(t_c|x)\ln\frac{p_{true}(t_c|x)}{p(t_c)|x,\theta} - p_{true}(t_c|x)\ln p_{true}(t_c|x)] \\
&= \mathbb{E}_{p_{true}(x)}[D_{KL}[p_{true}(y|x)||p(y|x,\theta)] + \mathrm{H}[p_{true}(y|x)]]
\end{aligned} \tag{2.22}$$

where the $D_{KL}[p_{true}(y|x)||p(y|x,\theta)]$ component is optimised during training and is thus reducible and $\mathrm{H}[p_{true}(y|x)]$ corresponds to the underlying existent variance in the true distribution of the dataset and is thus irreducible by the neural network.

Regression As for regression, taking the expectation of Equation 2.13 results in the following expression:

$$\begin{aligned}
\mathscr{L}(\mathscr{D}_{true},\theta) &= \mathbb{E}_{p_{true}(x)}[\mathbb{E}_{p_{true}(y|x)}[-\ln p(y|x,\theta)]] \\
&= \mathbb{E}_{p_{true}(x)}[-\int p_{true}(y|x)\ln p(y|x,\theta)dy] \\
&= \mathbb{E}_{p_{true}(x)}[\int p_{true}(y|x)\ln\frac{p_{true}(y|x)}{p(y|x,\theta)} - p_{true}(y|x)\ln p_{true}(y|x)dy] \\
&= \mathbb{E}_{p_{true}(x)}[D_{KL}[p_{true}(y|x)||p(y|x,\theta)] + \mathrm{H}[p_{true}(y|x)]]
\end{aligned} \tag{2.23}$$

where $D_{KL}[p_{true}(y|x)||p(y|x,\theta)]$ once more represents the reducible component of the loss optimised through the training process whilst $\mathrm{H}[p_{true}(y|x)]$ is irreducible as an inherent characteristic of the dataset.

2.3.5 Additional remarks

In subsection 2.3.2, Bayesian Neural Networks are briefly mentioned. Even though uncertainty can inherently be quantified in such a model, BNNs have drawbacks that hinder its use in applications such as safety critical systems - that of its memory and computational inefficiency [Goan and Fookes (2020); Bykov et al. (2021)]. In fact, keeping a distribution of weights exponentially increases the memory needed to store the network and makes the optimisation process more complex. As a result, these models are not generally applied in practical settings. Safety-critical

systems would benefit the most from the availability of uncertainties. However, these applications generally demand real-time compatibility and have memory restrictions in the deployment environment, making thus the use of BNNs impractical.

Several techniques are employed in Bayesian Neural Networks in order to allow probabilistic inference. Most of these methods are sampling-based and are extensively used in the practice of statistics. Sampling-based techniques allow the approximation of an unknown probability distribution through the extraction of samples from the model. Sampling algorithms guarantee a perfect approximation given there are infinite samples. As such, the number of samples that are extracted are a parameter that must be tuned to interest. Examples of such techniques are Langevin Dynamics [Teh et al. (2016); Knolle et al. (2021)] and Hamiltonian Monte Carlo [Li et al. (2019a,b); Hernández et al. (2020); Cobb and Jalaian (2021)], both Markov Chain Monte Carlo [Shannon (1948); Kass et al. (1998)] algorithms.

2.4 Object Detection

Object detection is a central Computer Vision task that aims to localise and classify objects in many data sources. The task has wide applications, ranging from the medical segment [Laves et al. (2020); Gillmann et al. (2021)] to autonomous driving [Feng et al. (2020, 2021)].

In the context of self-driving vehicles, both 2D and 3D object detection is considered. This stems from the fact that vehicles are typically equipped with many sensors, some of which generate two-dimensional and others three-dimensional representations. The most popularised object detection models focus mostly on RGB images and pointclouds, collected by two essential sensors to automated driving. Fusion methods aim to concentrate and utilise the information from several sensors simultaneously, and, as such, these models perform both two-dimensional and three-dimensional object detection [Lin et al. (2023); Wang et al. (2024)].

The focus of this thesis is in both two-dimensional and three-dimensional object detection. For generic object detection evaluations, image detection is the priority. For the autonomous driving application, however, only pointcloud-based models are considered. This is due to the fact that pointcloud detection is central to the achievement of self-driving vehicles and yet, it is one of the sub-fields of object detection that still has many open points of improvement.

2.4.1 Deep Learning Architectures

In the topic of deep learning, several different object detection architectures exist [Ren et al. (2017); Liu et al. (2016); Wu et al. (2023a); Yang et al. (2022); Liu et al. (2024)]. Apart from the nuanced differences between each model, a standard layout of a typical object detection pipeline is depicted in Figure 2.5.

An object detection pipeline starts with the consumption and processing of the input data. For image-based detection, the whole image is processed in its entirety as a matrix. For pointcloud-based data, there are more difficulties in the manipulation of the three-dimensional space, leading

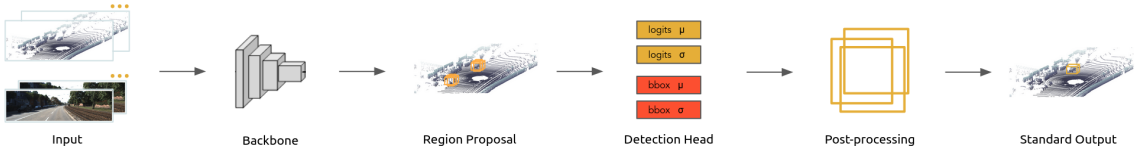


Figure 2.5: Standard pipeline of an object detection deep learning model.

to challenges in processing speeds and inaccuracies of the models. For this reason, several approaches exist in the literature [Qi et al. (2017); Lang et al. (2019); Meyer et al. (2019); Shi et al. (2020)], that extend from the transformation of the poincloud to bird's eye view or range image or voxelisation. Voxelisation, in particular, is central to most modern object detection models that consume three-dimensional data, and consists in the division of the data into voxels, akin to the concept of pixels in images, only with depth information.

Following the processing of the input data into a format fit to be digested by the network, the backbone module intends to extract features from this data so as to highlight patterns that aid classification and regression. For vision-based tasks, backbones are usually sets of convolutional blocks in sequence that filter the data and generate the feature maps [Zhao et al. (2024)].

Having extracted the features of interest from the data, it is then relevant to propose locations or regions of interest that could contain objects. This allows refined location regression and classification to occur later in the pipeline. A standardised method for the extraction of possible object locations is based on anchor proposals, a topic further discussed in Subsection 2.4.2.

Regions that are classified as having a high probability of containing an object are then forwarded to the detection head where fine-grained location regression takes place, as well as the classification of the object in pre-defined categories.

Since several boxes in surrounding areas that encapsulate the same object may be proposed, an additional post-processing step is necessary in order to ascertain that no duplicate boxes remain. The typical operation during this phase is non-maximum suppression [Rosenfeld and Thurston (1971)], a method that enables the extraction of the bounding box that best represents an area with several surviving proposals.

Throughout this thesis, these modules are referenced often as they represent the building blocks of object detection. These components are also of interest to the understanding of the concepts of existence and interpretations of uncertainty as different knowledge is encoded in each module.

2.4.2 Anchors

Since the proposal of regions of interest based on anchors is a central topic in further chapters, it is relevant to provide a detailed overview of this process.

The concept of anchors is to overlay a pre-defined set of bounding boxes in different aspect ratios and scales over the input sample, allowing fast detection of objects in several shapes and sizes.

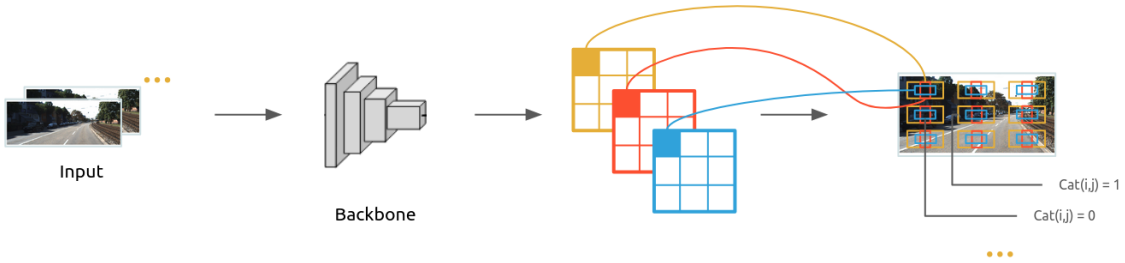


Figure 2.6: Anchors tiled over the input sample at the downsampled feature map level. Categorical classification as foreground/background is performed for each anchor proposal.

In the detection pipeline, the anchor boxes are tiled over the extracted feature maps and the model classifies the contents of each anchor either as foreground or background. Background boxes are discarded whilst the ones with a confidence level above a certain threshold are fed to further modules of the object detector, such as the detection head. In order to facilitate location regression, typically offsets are also calculated by the model for each anchor.

Following this procedure, the model does not in reality compute bounding boxes but rather refines pre-existent ones. All anchors can be tiled over the entirety of the input sample and over all classes simultaneously. This method enables the model to obtain much faster processing speeds and higher accuracy as anchors come in many shapes and sizes that can fit all objects of interest. In order to ensure the best fit, anchors are typically pre-computed for each dataset and object type. A popular approach is to use k-mean clustering [Zhong et al. (2020)].

Figure 2.6 demonstrates the overlaying of anchors in an object detection pipeline. Consider that anchors are represented by a tuple, as demonstrated in Equation 2.24. Anchors include a heading and often times, a single anchor may be overlayed several times in the input sample with different pre-defined headings α.

$$\mathrm{A}(c_x, c_y, c_z, d_x, d_y, d_z, \alpha) \tag{2.24}$$

Classification as foreground/background is given by the output of the last layer of the region proposal module, as depicted in Equation 2.25. Typically, this classification comes in the form of a categorical distribution with a logit that represents the confidence value for each of the object types contemplated in the model. Offsets for the improvement of the location regression may also be returned, so as to enable fine-grained centre and dimensions regression.

$$f_{rp}^{\mathbb{F}}(x) = Cat(\mathrm{A}(c_x, c_y, c_z, d_x, d_y, d_z, \alpha); \hat{\lambda}) \tag{2.25}$$

where $\mathbb{F}$ is the manifold of feature map dimensions obtained after downsampling in the convolutional blocks and $Cat(A(c_x, c_y, c_z, d_x, d_y, d_z, \alpha); \hat{\lambda})$ is the categorical distribution that, for each anchor, predicts a vector of probabilities $\hat{\lambda}$ for foreground and background classes.

For more information, the first proposals of anchors can be found in works like Faster-RCNN [Ren et al. (2017)] and SSD [Liu et al. (2016)], where this mechanism is proposed to move away

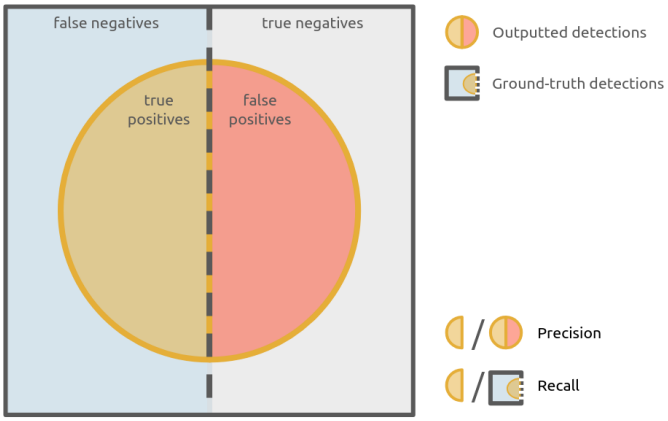


Figure 2.7: Relationship between precision and recall taking into consideration the outputted detections and the ground-truth targets.

from slow selective searches and sliding windows that were the staple in previous literature. A testament to the efficiency and long lasting impact of anchors is that most contemporary models still rely on these for region proposal due to their proven advantages [Meyer et al. (2019); Lang et al. (2019); Zhong et al. (2020); Shi et al. (2020); Wu et al. (2023b)].

2.4.3 Evaluation Methods

For the evaluation of object detection, a task that includes both classification and regression problems, dedicated metrics have been proposed over the years. Since these metrics are dedicated to vision and especially object detection settings, a brief overview is presented on the ones most utilised in this thesis.

The first concepts of evaluation that are central to the classification aspect of object detection are the true positive, true negative, false positive and false negative rate. True positives occur when the model detects an object that exists in the ground-truth. False positives, on the other hand, happen when the detector predicts a detection that does not actually exist. Following the same logic, true negatives are all the detections that are not made by the model that truly do not exist in the ground-truth. Finally, false negatives occur when the model does not detect an object that is in the ground-truth.

To summarise these concepts, the metrics of precision and recall are utilised. Precision corresponds to the rate of true positives over the total count of detections outputted by the model. Recall, on the other hand, measures the rate of true positives divided by the amount of ground-truth detections. The expressions that allow the calculation of both metrics are reported in Equation 2.26.

For clarification, Figure 2.7 demonstrates how these concepts intertwine. Both precision and recall are essential to the evaluation of an object detector. This stems from the fact that not only is it vital to have many correct detections, but also to have a significant representation in the set of ground-truth targets.

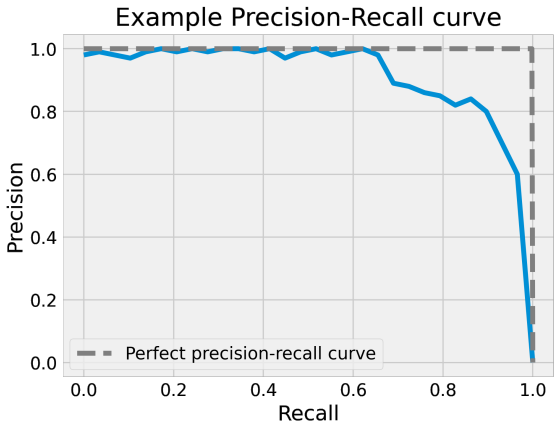


Figure 2.8: Relationship between precision and recall taking into consideration the outputted detections and the ground-truth targets.

$$\begin{aligned} \text{Precision} &= \frac{\text{true positives}}{\text{true positives + false positives}} \\ \text{Recall} &= \frac{\text{true positives}}{\text{true positives + false negatives}} \end{aligned} \tag{2.26}$$

Another metric of interest that provides a measure of predictive performance based on precision and recall is the F-score. In this thesis, only the F1-score is utilised, which corresponds to the harmonic mean of precision and recall, as seen in Equation 2.27.

$$F1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{2.27}$$

Further than the F-score, precision-recall curves also demonstrate the trade-off between the results of both metrics and are widely utilised in literature [Geiger et al. (2012)]. Figure 2.8 showcases an example of a precision-recall curve that may be obtained when evaluating an object detection model. The curve is obtained by measuring precision and recall when assuming different probability thresholds for the classification prediction scores. A good object detector achieves high precision throughout all levels of recall.

The intersection over union (IoU) is another metric of high importance to object detection that corresponds to the intersection area over the union area when evaluating two bounding boxes. This metric represents the regression target of object detection, namely the correspondence between predicted bounding boxes and the ground-truth. Typically, detections are only counted towards the aforementioned metrics when the box regression achieves a threshold IoU target. A target of $IoU = 0.7$ is commonly used for the evaluation of object detectors in recent literature [Wu et al. (2023b)].

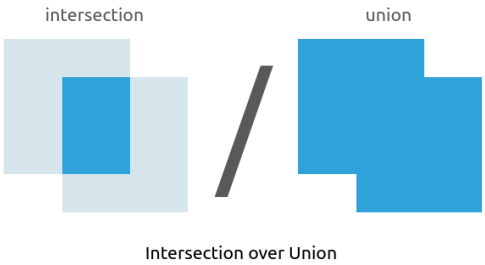


Figure 2.9: Intersection over Union corresponds to the area of intersection over the area of union.

2.4.4 Overview of Datasets

The training and testing of deep learning models entails the existence of a large collection of data. As such, this subsection presents an overview of relevant datasets for the task of object detection. Within the problem of recognising objects, there are several identifiable sub-fields that are covered by different data sources. In this thesis, there are two main approaches to object detection that are studied: generic and broad in scope collections of objects, and autonomous driving-specific.

Generic Objects Generic object detection intends to be able to identify objects from a panoply of settings and contexts, and these data sources reflect this intention. The main data banks considered usually for global evaluations are as follows:

CIFAR-10/CIFAR-100 [Krizhevsky (2009)] CIFAR datasets contain a collection of 32x32 images sporting different types of objects ranging from means of transportation to animals. CIFAR-10 considers 10 different classes whilst CIFAR-100 presents labels over 100 classes. This is a classification dataset.

ImageNet/Tiny-ImageNet [Le and Yang (2015)] ImageNet is a large scale research project that provides millions of generic images and labels for 21,841 classes. Tiny-ImageNet is a smaller subset of this dataset sporting 200 classes.

For some validation purposes in further chapters, perturbation datasets are needed. For these ends, ImageNet-C [Hendrycks and Dietterich (2019)] is considered as a source of algorithmically generated corruptions on the ImageNet test set.

Autonomous Driving Autonomous driving is the main area of application in the study carried out in this thesis. A brief description of the major datasets for object detection within this scope is presented as follows:

KITTI [Geiger et al. (2012)] Wide range of data available for research use in a panoply of tasks. *Available sensors:* RGB camera, grayscale camera, LiDAR.

Waymo [Sun et al. (2020)] Perception and motion datasets available for use in the scope of autonomous driving research. Includes motion and interaction challenges which allow validation for tasks such as prediction. *Available sensors:* camera, LiDAR.

nuScenes [Caesar et al. (2020)] Large-scale dataset with detection, tracking, segmentation and prediction benchmarking. Two topographies and a wide range of classes represented. *Available sensors:* camera, LiDAR, radar.

ApolloScape [Ma et al. (2019)] Dataset with a focus on 3D object detection. Includes data for validation on a variety of other supporting perception tasks.*Available sensors:* camera (unlabelled), LiDAR.

BDD100K [Yu et al. (2020)] Another large dataset with a wide variety of labelled data available. Includes data for imitation learning to be applied on driving trajectory tasks. *Available sensors:* camera.

Level 5 [Kesten et al. (2019)] Perception and motion predictions datasets available, with traffic lights considered for detection. Format is similar to nuScenes. *Available sensors:* camera, LiDAR, radar.

Cityscapes [Cordts et al. (2016)] Large perception dataset, including a benchmark suite. Focused on semantic segmentation tasks. *Available sensors:* camera.

Considering the delineated datasets, KITTI is selected as the major source of data due to its established prominence and relevance of the provided benchmark in the assessment of state-of-the-art models. More than this, KITTI showcases validation mechanisms for a wide variety of tasks, with the included data being labelled in all the desirable sensors.

In the context of autonomous driving, perturbation data is also useful in some experimental validations. For these ends, KITTI-C [Kong et al. (2023)] is considered as the replication of KITTI's training dataset with 10 types of added perturbations.

2.4.5 Additional remarks

Object detection has advanced considerably in recent years. In the scope of autonomous driving, great efforts have been made towards increasingly robust models with real-time processing capabilities, especially in the 3-dimensional realm.

Models like LaserNet [Meyer et al. (2019)], based on pointcloud data, focus on improving inference speed as well as provide uncertainty estimates, a factor that has become of high interest in current research to enable failure recognition and recovery in safety-critical systems.

Following the trend of improving processing speeds, PointPillars [Lang et al. (2019)] is proposed, allowing a simpler digestion of the pointcloud data by excluding complicated point based models and complex voxelisation pipelines. For these characteristics, alongside its remarkable inference speed, it became one of the most benchmarked models in a panoply of fields. PV-RCNN [Shi et al. (2020)], another notable model, builds upon the work of [Qi et al. (2017)] by leveraging point-based semantics with 3D sparse convolution. Its advances lie in its more complex 3D representation and abstraction of points in voxelised space, contrary to PointPillars. The voxelisation of the pointcloud is an increasingly popular sub-area of research within object detection. Several

models have since proposed improvements to this pre-processing step with positive results [Deng et al. (2021); Qian et al. (2022); Li et al. (2022)].

On the topic of sparse convolution, there have been many research efforts as well [Baoyuan Liu et al. (2015); Chen et al. (2022); Wu et al. (2022a); Liu et al. (2024)]. Focal Sparse Conv [Chen et al. (2022)] is an example that achieved state-of-the-art results on most benchmarks, from KITTI to nuScenes [Caesar et al. (2020)] and Waymo [Sun et al. (2020)]. Its proposal centres around showing that feature sparsity is learnable and indispensable to advanced object detection.

Other types of models have also been proposed that tackle different aspects of the recognition pipeline, including those based on graph networks [Shi and Rajkumar (2020); Yang et al. (2022)], spatial transformers [Wu et al. (2023a); Hoang et al. (2024)], and multi-sensor fusion [Li et al. (2023); Lin et al. (2023); Dong et al. (2024); Wang et al. (2024)], among many others.

Recently, VirConv [Wu et al. (2023b)] achieved state-of-the-art results on KITTI, especially on car detection, the most widely seen object on the real world. This network provides a lightweight version, and has reported high accuracy and a fast processing speed, characteristics that are fundamental to deployment in autonomous driving. For this reason, this model is further mentioned in the next chapters, especially Chapter 5.

2.5 Robustness and Interpretability

Deep learning predictors are expected to be deployed in a wide variety of highly complex, uncertain and dynamic environments. The levels of accuracy that have been achieved by neural networks, both in a panoply of different tasks and in a wide range of domains, have cemented the will to bring deep learning to the real-world [Pang et al. (2021a); Raut et al. (2021)]. The staggering popularity of this field has led to its explosion in both academic and industrial areas, with some models already reaching and affecting the lives of millions of people. ChatGPT [OpenAI (2022)] can be seen as an example of both the demand for artificial cognitive systems and the potential of deep learning to find its way to human daily life, for example.

Even so, the achievements of deep learning are still, for the large part, downplayed by an intrinsic lack of interpretability, which can become a paramount factor in their safe deployment to the unpredictable real-world [Willers et al. (2020)]. This is the point upon which the central ideas of this thesis are built and, as such, a detailed explanation of interpretability concepts and their motivation is necessary for the understanding of further chapters.

2.5.1 The Problem of Interpretability

Consider the simulated scenario presented in Figure 2.10. In this context, a network is tasked with a classification problem, being trained to distinguish cats and dogs from images. Naturally, a scientist or practitioner will construct a dataset composed of examples of both cats and dogs. In deployment, when an image of a dog, for example, is given as input to the neural network, the model will answer correctly with a given accuracy and confidence rate. However, if, say, an image of a monkey is fed to the model, a common occurrence in the real world, the classification network

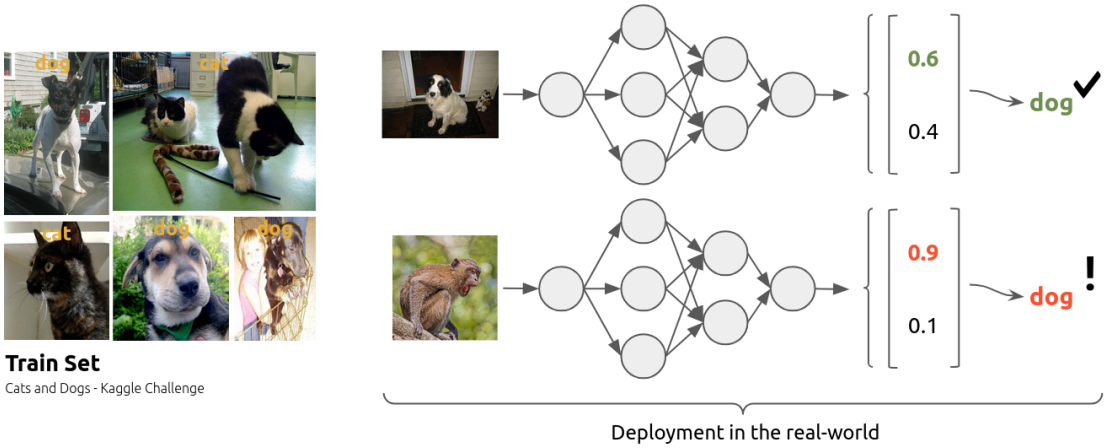


Figure 2.10: Neural networks lack expressiveness when deployed in the real-world. When out-of-domain samples are fed to the network, its structure is not flexible enough to encapsulate the desirable behaviour, which leads the model to predict incorrectly with high *confidence*.

has no choice but to attribute a *confidence* to both classes *dog* and *cat*. This lack of flexibility leads to several undesirable behaviours to occur. Firstly, the network is set to fail the classification assignment and, even worse, to not be aware that it should fail, as it has no knowledge of any other concepts apart from cats and dogs. Secondly, it is often possible, due to the similarity of highly-dimensional feature spaces, that the outputted classification is attributed a high *confidence*, even when resoundingly incorrect.

Studies have proven this behaviour is found in state-of-the-art neural networks. Szegedy et al. (2014) change small characteristics of images that are imperceptible to humans and show that neural networks can change their classification drastically, even with slight alterations. Moreover, in [Nguyen et al. (2015)], it is showcased that trusted state-of-the-art neural networks can output high softmax confidence even on unrecognisable samples. Another experiment that demonstrates the lack of awareness of neural networks is performed by Weideman (2016). The author trains a network on CIFAR-10 and evaluates its performance on CIFAR-100, with 90 additional classes. The findings demonstrate that the model outputs softmax probability of over 0.9 when classifying samples of the class *apple* from CIFAR-100 as *automobile* and *frog* from CIFAR-10.

These characteristics are inherent to the formulation of deep learning architectures. As established in literature, models in general are unable to indicate when they can not answer a query, are unaware of their own lack of knowledge and are commonly seen as black boxes by both end users and practitioners.

Interpretability is defined as the ability to correspond cause to effect in systems [Wu and Song (2019); Zhang et al. (2021); Wang et al. (2023)]. In the scope of deep learning, specifically, interpretability is high when inputs produce the expected outputs. In this sense, these characteristics being expressed by neural networks strongly endanger their interpretability and robustness. In particular, considering the aforementioned example, slight changes in the input samples would not be expected to lead to such divergent results, which demonstrates a high lack of interpretability by

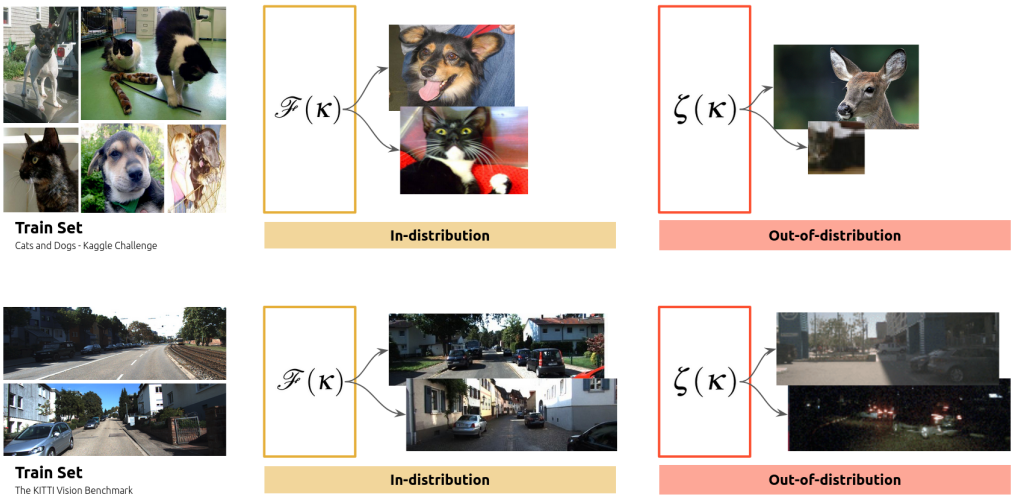


Figure 2.11: Datasets and corresponding in and out-of-distribution samples. Out-of-distribution samples may be from different classes, have a different resolution or may have been collected by another sensor.

the network. The fact that unanswerable queries are also attributed high confidences in incorrect classifications also showcases this characteristic of deep learning.

2.5.2 Dataset Distributions

In the context of producing predictable outcomes to given inputs, it is highly important to carefully consider the training distribution.

A model is trained on a dataset, $\mathscr{D}$, that approximates the true underlying distribution that describes some phenomenon, $p_{true}(\mathscr{D})$. Consider now that $\mathscr{F}$ is a function that enables the generation of samples from $\mathscr{D}$. A dataset can thus be generated with parameters κ as described in Equation 2.28.

$$\bigcup_{\kappa} \mathscr{F}(\kappa) \sim p_{true}(\mathscr{D}) \tag{2.28}$$

The resulting examples that compose $\mathscr{D}$ are sampled with $\mathscr{F}$, and thus all belong to the same distribution. Datasets are typically realisations of the same sampling distribution and its samples are referred to as *in-distribution*. This includes all x, y pairs used for both training, testing and evaluation standardly. This signifies that accuracies reported in *in-distribution* sets only ascertain the model's capability to capture $\mathscr{D}$ as a realisation of the true underlying distribution that represents some real phenomenon.

Consider now that a new sample is obtained with another function, ζ, that also enables the generation of a sampling distribution of $p_{true}(\mathscr{D})$. Since the sampling function differs, the obtained example does not belong to the same distribution as the previously generated ones. In this case, this sample is referred to as being *out-of-distribution*.

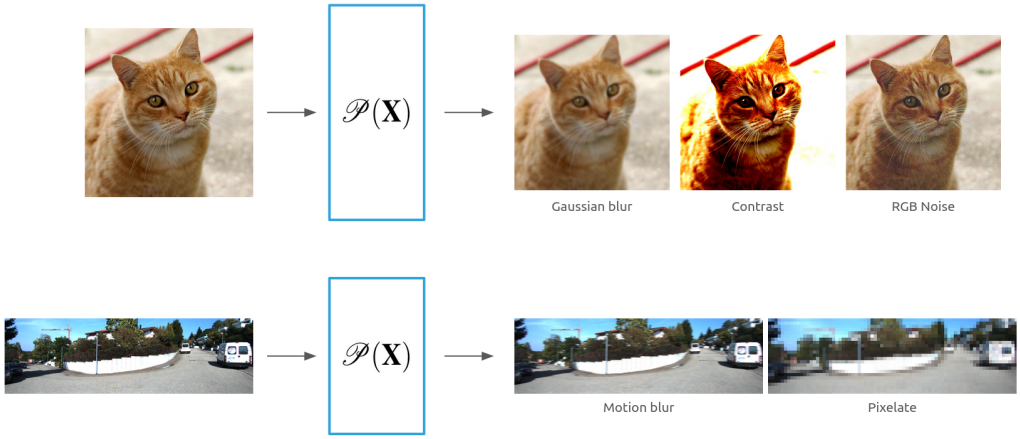


Figure 2.12: Datasets and corresponding in and out-of-distribution samples. Out-of-distribution samples may be from different classes, have a different resolution or may have been collected by another sensor.

Out-of-distribution samples are not necessarily entirely different, neither do they need to represent further classes from the in-distribution samples. Figure 2.11 showcases some examples of in and out-of-distribution examples in varying settings that demonstrate the different sources of sampling distributions. In the case of autonomous driving, for example, changing the laser scanner or the camera sensor is sufficient to produce out-of-distribution samples, even whilst collecting data in the same locations and conditions.

In the context of interpretability, in recent years, fields such as out-of-distribution, open-set and outlier detection have gained traction in literature [Vaze et al. (2021); Balasubramanian et al. (2021); Salehi et al. (2021); Yang et al. (2024a)], among many others with similar purposes [Salehi et al. (2021)]. These efforts demonstrate the need for expressive and interpretable models, not only for end-users but for practitioners and researchers alike.

Another dataset distribution generation process of interest to this thesis is perturbations. A perturbation dataset is generated by applying corruptions and perturbations, $\mathscr{P}$, to existent data sources, as demonstrated through Equation 2.29.

$$\bigcup_{\mathscr{D}} \mathscr{P}(\mathbf{X}) \sim p_{true}(\mathscr{D}) \tag{2.29}$$

The main objective of perturbation sets is to better evaluate and/or improve the robustness of neural networks to samples that differ from the training distribution. As mentioned in the previous subsection, interpretability is based on having expected outcomes. With both a standard and a perturbation set, interpretability may be evaluated by feeding both the unaltered sample and the perturbed one to the model. In an interpretable model, both of these samples should produce the same output. Examples of commonly applied perturbations are shown in Figure 2.12.

Corruptions and perturbations also enable the validation of the robustness of the model to distributional shifts, a characteristic that may be invaluable for safety-critical systems such as self-driving vehicles. This concept ties heavily with the phenomenon of underspecification, described

thoroughly in Subsection 2.5.4.

2.5.3 Object Existence

Another topic of interest in the scope of interpretability is that of open versus closed-set classification. In Subsection 2.5.1, a closed-set classification example was demonstrated, alongside its weaknesses. Since a pre-determined set of classes forces models to output a confidence for each, out-of-distribution samples or anomalies can not be correctly signalled, leading to incorrect classifications instead.

The problem of open-set recognition consists in the simulation of a more realistic setting where unknowns may arise during testing [Geng et al. (2021); Yang et al. (2024a)]. This field is based upon the difficulty of sourcing representative data of all object classes that exist in the real world. Open-set models are then expected to be able to deal with novel examples during both test and deployment, being more robust.

The level of unknowns that are used during the testing phase of these models may range from: positive labels used as negative samples, labelled negative samples, classes with no representation in the training set but with information available and classes completely unseen and unknown [Geng et al. (2021)].

In the context of object detection, region proposal modules also rely on closed sets to discern between anchors that are to be considered foreground or background. As such, the entire pipeline depends on the pre-determined set of classes that the model is trained to recognise. This notion does not enable networks to discern or interpret novel objects that span in the real world. For this reason, the term of object existence as mentioned in this thesis entails not only a global concept and definition of object beyond classification efforts but also the attribution of a probability of whether an object exists or not in the scene, as described in Equation 2.30.

$$\begin{aligned} p_{existence}(o) &= Bernoulli(o) \\ p_{existence}(\mathbf{O}) &\in p_{true}(\mathbf{O}|x) \end{aligned} \tag{2.30}$$

where o represents an object from the set of all objects $\mathbf{O}$ existent in sample x.

2.5.4 Underspecification

Pertaining to the interpretability of neural networks, there is one last concept of importance to discuss, that of underspecification. Underspecification is coined in [D'Amour et al. (2022)], where the authors demonstrate that the same deep learning architecture may generate several different predictors with contrasting biases, leading to high variability of results in the deployment environment.

Consider a neural network architecture defined over its layers, including all standard operations such as activations and normalisations. The stochastic training procedure of this model results in the generation of a predictor that is to be deployed in an environment.

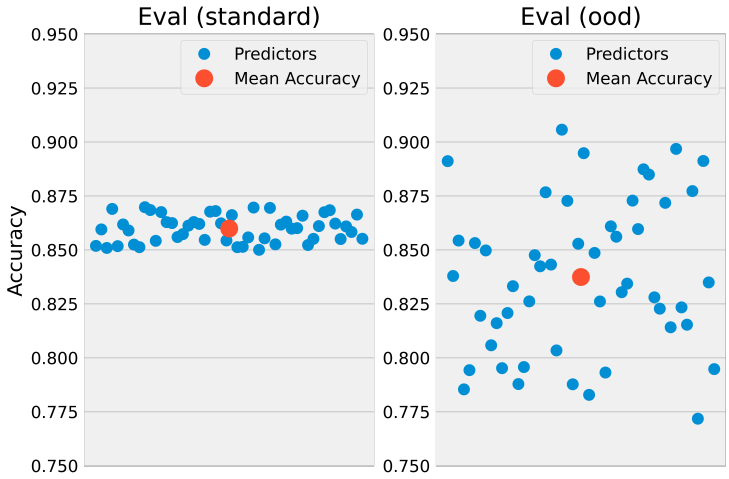


Figure 2.13: Predictors generated from the same architecture concentrate in a small area of accuracy during evaluation in a standard test set, however, when deployed in out-of-distribution and out-of-domain data, their accuracies diverge considerably.

Training this architecture with slight variations in its hyperparameters will result in the generation of different predictors. Variations may range from diverging seeds to batch sizes or weight initialisations, for example. It is expected that all generated predictors will have similar performance in the deployment domains or at least that their spread of accuracy is dismissible. D'Amour et al. (2022) demonstrate that this is not the case in practice. As shown in Figure 2.13, underspecification is observable as the accuracy spread among predictors is narrow during standard evaluation but widens significantly in an out-of-distribution setting.

Prior to the formalised experiments presented in D'Amour et al. (2022), models were thought to have somewhat predictable performance in deployment as a result from the evaluation process in test sets. From the leftmost plot in Figure 2.13, it is not obvious that the displayed predictors have, in fact, contrasting biases; and that these biases may lead to strikingly different results in tests outside the training distribution. This further cements the fact that there is still much to learn about deep learning models, especially their optimisation process which leads to such divergent outcomes even in the presence of almost dismissible changes.

Further than this, the authors in D'Amour et al. (2022) show underspecification to be an inherent characteristic of most machine learning pipelines. This means that this issue plagues not only deep learning but goes much further into the area of artificial intelligence research.

Finally, the fact that practitioners need to deploy a predictor in the real world without knowledge of its expected behaviour is concerning. In fact, underspecification is another characteristic that inhibits a neural network from being interpretable. If underspecification is present, it is hard to report expected outcomes with simple evaluation tests. As such, quantifying and safely dealing with underspecification is a staple topic in deep learning interpretability.

2.6 Chapter Summary

In this chapter, the foundational theoretical concepts that support the discussions in this thesis were presented. These concepts range from essential deep learning formulations to statistical notions. Knowledge from the fields of application that is indispensable for the understanding of further chapters is also presented, from object detection to the area of deep learning interpretability.

In this sense, Section 2.1 first gives an overview of some global notation that is used throughout the thesis. Following this, the next sections present a formal definition of neural networks, Section 2.2, and a probabilistic interpretation of a deep learning model, Section 2.2.2. The formalisations in these sections allow further definitions of uncertainty and probabilistic maps discussed in later chapters.

Completing the theoretical knowledge of interest for the derivations reported in this thesis, Section 2.3 addresses topics related to statistics that are of relevance for uncertainty models and quantification techniques. This includes the definition of Shannon's Entropy to Bayes' Rule and Variational Inference, finishing with a discussion of important formulations for the derivation of a model of uncertainty.

Concerning then the fields of application, Section 2.4 reports on object detection, including the description of a generalised sample architecture, evaluation methods and datasets of interest. A detailed discussion of anchors is also presented as this concept is relevant in further chapters.

The final Section 2.5 reflects upon the topic of deep learning interpretability and addresses the concepts that are more pertinent in the context of this thesis. This section demonstrates key aspects of neural networks that concern interpretability, from dataset distributions to the problems of open-set recognition and underspecification.

Chapter 3

Uncertainty and Deep Learning

Authors such as Gal and Ghahramani (2016) have extensively demonstrated the necessity of *knowing what our models do not know*, especially in safety-critical applications [Kendall and Gal (2017)]. Even more, the topic of accountability of AI systems is on the rise, and further efforts for operational transparency in neural networks are requested more and more [Novelli et al. (2022)]. From this, it becomes apparent that, for a systematic and safe integration of deep learning in real-world settings, its interpretability and expressiveness must be thoroughly studied.

In efforts to improve these aspects, some sub-fields of deep learning have arisen and seen increased popularity in recent years [Yang et al. (2024b); Pang et al. (2021b)]. A notorious subfield is uncertainty quantification [Feng et al. (2021)], a central topic in this thesis.

In fact, the main objective of uncertainty quantification for deep learning is to provide, alongside the common outputs of the network, a *truly* probabilistic measure of the certainty of these outputs. The field proposes to do so along two relevant axis - of the input data and of the knowledge contained within the model itself. A model equipped with uncertainty quantification, when presented with the example depicted in Figure 2.10, where a monkey is given as input, would attribute equal probability to both the *dog* and *cat* classes, thus demonstrating that the model is unable to answer the query. This result is incredibly relevant, not only to avoid disastrous incorrect answers, but also to demonstrate the lack of knowledge of the model.

There have been recent studies that utilise a panoply of methods for the integration of uncertainty estimation in deep learning-based predictors [Miller et al. (2019); Feng et al. (2019); de la Riva and Mettes (2019); Laves et al. (2020); Moshkov et al. (2020); Yelleni et al. (2024); Gong et al. (2022); Sirohi et al. (2023)]. Nevertheless, several challenges still remain to be tackled, which pinpoints the value of further research in the field. For example, most quantification methods rely on sampling, being incompatible with real-time use [Gal and Ghahramani (2016); Ghoshal et al. (2020)]. Another aspect that needs further assessment is the existence of different uncertainty interpretations [Kendall and Gal (2017); Malinin (2019)].

In this sense, this chapter presents an overview of the basic and underlying concepts for uncertainty quantification in deep learning. This starts with the formal definition of uncertainty

quantification, a topic presented in Section 3.1. These ideas are further deepened in the next Section 3.2, which aims to define uncertainty in a broad sense, including sources and properties of uncertainty as well as its existent interpretations.

Having established the fundamental topics of interest to uncertainty, the techniques utilised to estimate its quantities and parcels are discussed in Section 3.3. Section 3.4 then focuses on the evaluation of uncertainty estimates, including the topic of calibration.

3.1 Defining Uncertainty Quantification

Uncertainty quantification is a broad term that represents the field where the estimation, characterisation, management and evaluation of uncertainties in either computational or physical systems is carried out. This science aims to systematically determine the likelihood of outcomes in the presence of stochastic events.

Uncertainty in stochastic systems is a large concern, thus, there is broad literature studying its aspects, especially in computer simulation [Walker et al. (2003); Iman and Helton (1988); Santner et al. (2003); Ranftl and von der Linden (2021)]. Examples of applications that strongly depend on uncertainty awareness are decision-making, finance, risk analysis and most recently, in particular, deep learning.

In neural networks, it becomes of interest to have a quantitative measure of the certainty of the produced outputs. This may be achieved in different forms - one would be the estimation of an uncertainty value per each output (through the calculation of Entropy, for example); another may be the approximation of a predictive distribution that demonstrates the range of outputs that the network may produce and their respective probability. These, among many others, are some of the existing uncertainty interpretations, which are useful in different contexts and depending on their desired effect.

Particular to deep learning models, it is relevant to estimate and single out different sources of uncertainty [Kendall and Gal (2017)]. One of these relates to the intrinsic randomness in the data that generally can be traced back to the data collection process (typically due to incomplete information, low observability or noisy sensors). Another component is connected to the amount of knowledge that can be contained within an architecture through its parameters that, being defined over a stochastic optimisation method, have a degree of uncertainty over their values.

The randomness of the dataset is already somewhat captured in deep learning since, through the optimisation process, the model learns to recognise the pattern of variations. Common classification models also already output a distribution over classes through the use of an activation function (for example, the softmax). This demonstrates that neural networks naturally digest data-related uncertainties.

For consideration, in Figure 3.1 examples of applications where quantified uncertainty is demonstrated over the input sample of a deep learning model are included. The left diagram depicts medical imagery applications [Gillmann et al. (2021)] where areas of high pixel-level uncertainty are demarcated for computer-assisted diagnosis, for example. The right side, on the other

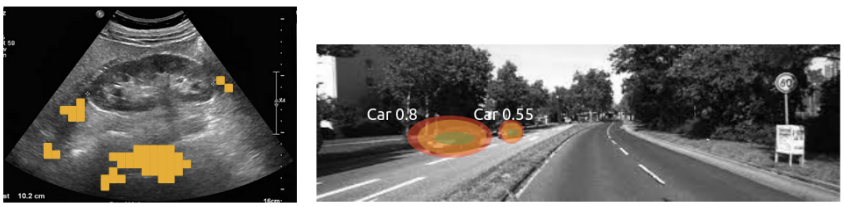


Figure 3.1: Examples of estimated uncertainty represented in input samples of deep learning models. On the left, pixels with high uncertainty are marked in yellow over an ultrasound image of a kidney. On the right, a sample from a perception model demonstrates location and classification uncertainty.

hand, showcases the usefulness of uncertainties in a safety-critical system - an autonomous vehicle [Michelmore et al. (2018); Feng et al. (2021)]. In the case of object detection, both regression and classification are of interest, with a distribution over each object representing the location uncertainty and a measure of confidence in the outputted label.

Quantifying uncertainty in deep learning may lead to helpful developments in several dimensions - from data-collection processes to inner workings of models and even for downstream layers that consume these outputs [Gal and Ghahramani (2016); Malinin (2019)]. In particular, several aspects upon which the availability of uncertainty estimates would be rather impactful are signalled:

- enhancement of data pipelines, from the collection process to the optimisation of model efficiency [Sharma and Bilgic (2017); Yang and Loog (2016); Beluch et al. (2018); Raj and Bach (2022); Nguyen et al. (2021)];
- model performance improvement [Kendall and Gal (2017)];
- risk management mechanisms in downstream tasks;
- increased overall interpretability for consumers;
- accountability and explainability enablement;
- scientific advancements in deep learning (generalisation capabilities, bias identification, etc...).

3.2 Defining Uncertainty

Uncertainty is a term that may have a broad range of meanings. In fact, the interpretation of uncertainty itself is an open topic with different notions available in literature [Gal and Ghahramani (2016); Malinin (2019); Liu et al. (2020); Lahlou et al. (2023)].

This aspect stems from the existence of uncertainty in several different components of both models and systems. Furthermore, the sources of uncertainty are usually identified by each practitioner in accordance to specific needs or designs, making a global, systematic definition of uncertainty to be difficult to assess.

For consideration, we discuss some examples of general sources that have been identified in prior literature:

Parameter Relates with model parameters whose exact values are unknown in a mathematical model. Commonly found in experimental physical setups [Shacham et al. (2012); Matsunami et al. (2018)].
Example: Several properties of materials in finite element analysis.

Parametric Relates with input variables of a system which may vary without control [Chen et al. (2018)].
Example: Transfer coefficients in chemical synthesis and production.

Algorithmic Relates to errors that arise with floating point approximations performed in digital systems [Shin (1998); Ganguly and Earp (2021)].
Example: Approximating the solution to a differential equation in a computer system.

Experimental Relates to observation errors in experimental measurements [Moffat (1988); Delgado et al. (2021)].
Example: Experimental setups with repeated trials.

The doubts in the definition of uncertainty extend even further to the sub-categorisation of uncertainty. In general, most authors accept the division of the uncertainty in two main parcels, the aleatoric and the epistemic [Malinin (2019)]. Even so, and especially in the field of deep learning, the formalisation of the distinction between both is still an open topic [Hüllermeier and Waegeman (2021)]. The representation and interpretation of uncertainty must thus be selected in accordance with the application and desired use.

In the particular case of deep learning, uncertainties are commonly seen through a statistical analysis perspective. Several authors have discussed a probabilistic view of neural networks [Ney (1995); Choe et al. (2017)]. In this context, a neural network may be seen as a non-linear, complex statistical model, with the uncertainty being a property of this model. These aspects are presented in detail in the following sub-sections, starting with the accepted properties and categories in literature pertaining to deep learning, followed by a description of the most widely used mathematical models for uncertainty.

3.2.1 Properties and Sources of Uncertainty

As previously discussed, there are several sources of uncertainty that can be identified in any stochastic system. For this reason, the decomposition of uncertainty into representative components is an important research topic.

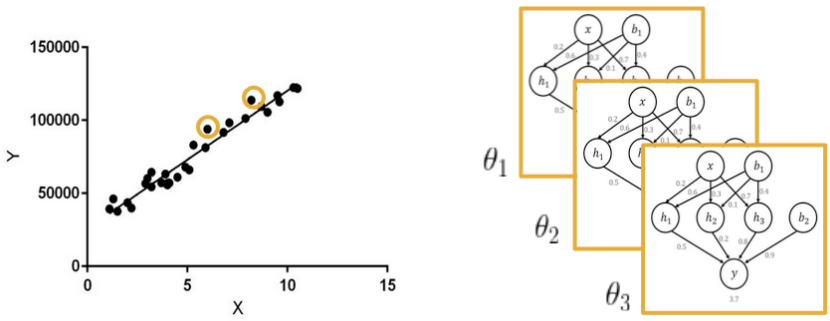


Figure 3.2: Representation of the aleatoric and epistemic components of uncertainty. The aleatoric parcel is a natural consequence of the inherent stochastic behaviour of the data collection process, whilst the epistemic component relates to the plurality of predictors that can be generated from the same deep learning architecture.

Most authors classify uncertainty in deep learning into two main categories: aleatoric and epistemic [Gal and Ghahramani (2016); Depeweg et al. (2018); Kendall and Gal (2017); Feng et al. (2021)]. The total uncertainty is thus derived upon the summation of these parcels.

Aleatoric uncertainty is commonly understood as the result of either intrinsic randomness or partial observability in the data generation process, corresponding to noise in the dataset [Feng et al. (2019)]. In this definition, the aleatoric parcel is irreducible as obtaining more data using the same process leads to the same noise patterns to be present in the resulting dataset. Even when employing different collection mechanisms, intrinsic randomness is inescapable due to our inability to observe the underlying true functions that describe the universe. Other interpretations for this parcel may also be considered and are more commonly found in statistical modelling [Walker et al. (2003)].

On the other hand, epistemic uncertainty represents the lack of knowledge in a model. In the field of deep learning, it can be described as the quantification of the disagreement between predictors that originate from using different sets of network parameters. This parcel is thus directly related to the knowledge captured within the network's architecture and is reducible as more information (in this specific case, training data) is obtained [Feng et al. (2021)].

Aleatoric uncertainty is further subdivided into homoscedastic and heteroscedastic [Lakshminarayanan et al. (2017); Gal and Ghahramani (2016); Guo et al. (2017)]. Whilst homoscedastic uncertainty assumes identical noise for all observations, thus representing an average, heteroscedastic uncertainty models the noise observed for each sample. This relationship may be observed in Figure 3.3.

Whilst both models are undoubtedly useful to many fields, some authors have studied their relevance in specific applications. For example, Kendall and Gal (2017) argue that, in Computer Vision tasks such as object detection, the most important are heteroscedastic models. On the other hand, homoscedastic uncertainty has been shown to be suitable to weigh loss functions in multi-task settings [Kendall et al. (2018)].

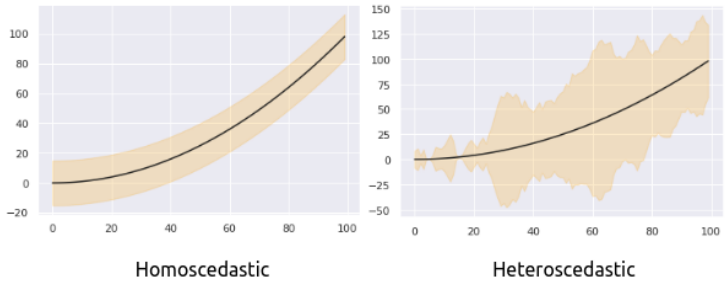


Figure 3.3: A representation of homoscedastic and heteroscedastic uncertainty, respectively. Whilst homoscedastic uncertainty assumes equal variance for all points, heteroscedastic noise depends on the input sample.

The predictive or total uncertainty (TU) is thus generally defined as the sum of the aleatoric (AU) and epistemic (EU) components, as seen in Equation 3.1. This formulae is widely supported in literature [Kiureghian and Ditlevsen (2009); Depeweg et al. (2018); Abdar et al. (2021); Valdenegro-Toro and Mori (2022)].

$$\text{Total Uncertainty (TU)} = \text{Aleatoric Uncertainty (AU)} + \text{Epistemic Uncertainty (EU)} \tag{3.1}$$

In the particular case of deep learning, some works comment on the necessity of further decomposition so as to delineate other sources of uncertainty. Malinin and Gales (2018), for example, assume the further division of epistemic uncertainty into the variability existent in the distribution of possible weights and a *distributional uncertainty* that is defined as the mismatch between in and out-of-distribution samples.

3.2.2 Interpretations and Models of Uncertainty

Taking into consideration the aforementioned sources of uncertainty, it is of interest to delineate existent approaches to the modelling of each component. A possible decomposition methodology [Malinin (2019)] for the estimation of uncertainty in parcels consists in defining the total uncertainty as the entropy of the predictor's output distribution. This relationship is represented in Equation 3.2.

$$\text{Total Uncertainty (TU)} = \mathrm{H}[p(Y|X,\mathscr{D})] = \mathrm{H}[\mathbb{E}_{p(\theta|\mathscr{D})}[p(Y|X,\theta)]] \tag{3.2}$$

Where H refers to the entropy of a distribution, a concept borrowed from the field of *Information Theory* [Shannon (1948)]. This signifies that the total uncertainty associated with a given input sample corresponds to the expected value of the output of the network over the distribution of parameters.

The aleatoric parcel, on the other hand, is calculated through the expected value of the entropy, the inverse relation, as described in the following equation:

$$\text{Aleatoric Uncertainty (AU)} = \mathbb{E}_{p(\theta|\mathscr{D})}[\mathrm{H}[p(Y|X,\theta)]] \tag{3.3}$$

Epistemic uncertainty may then be derived as the difference between both aforementioned components, represented through the subsequent equation:

$$\text{Epistemic Uncertainty (EU)} = \text{TU} - \text{AU} \tag{3.4}$$

Even though these formulae are highlighted for the decomposition of the uncertainty in parcels, and their use is further studied in forthcoming chapters of this thesis, several other interpretations can be found in the literature. For example, some authors interpret epistemic uncertainty as a representation of the distance between the dataset manifold and the training distribution domain [Liu et al. (2020)], a relation observable in Equation 3.5.

$$\begin{aligned} \text{Epistemic Uncertainty (EU)} &= \nu(d(x,\mathscr{X}_{IND})) \\ \text{with } d(x,\mathscr{X}_{IND}) &= \mathbb{E}_{x'\sim\mathscr{X}_{IND}}\|x-x'\|_X^2 \end{aligned} \tag{3.5}$$

Where ν is a monotonic function, x is a sample, $\mathscr{X}_{IND}$ is the training domain and $d(x,\mathscr{X}_{IND})$ is the expected distance between samples in the dataset X and the training set according to some distance metric.

Other authors define the total uncertainty as the expected loss [Lahlou et al. (2023)] or the variance of the output given the predictive distribution [Loquercio et al. (2020)]. This is illustrated in Equation 3.6 and 3.7 respectively.

$$\text{Total Uncertainty (TU)} = \mathbb{E}_{p(Y|X)}[l(f(x),y)] \tag{3.6}$$

$$\text{Total Uncertainty (TU)} = \sigma^2_{p(Y|X)}(y) \tag{3.7}$$

Where l is a loss function and $p(Y|X)$ is the predictive distribution.

Within the Bayesian framework, a Bayes-optimal predictor may be defined as the model that minimises the total uncertainty, demonstrated as follows:

$$f^*(x) \in \arg\min_{y'} \mathbb{E}_{p(Y|X)}[l(y',y)] \tag{3.8}$$

Where, as previously mentioned, l is the corresponding loss function. In this definition, the aleatoric uncertainty becomes the total uncertainty of the Bayes-optimal predictor.

Other summary statistics are utilised for the modelling of uncertainty. A commonly found approach is that of the predictive variance [Liu et al. (2020)]. Through this method, direct modelling or stochasticity in model components are used to obtain a predictive distribution and its variance is thus interpreted as the uncertainty of the underlying process [Gal and Ghahramani (2016)]. In

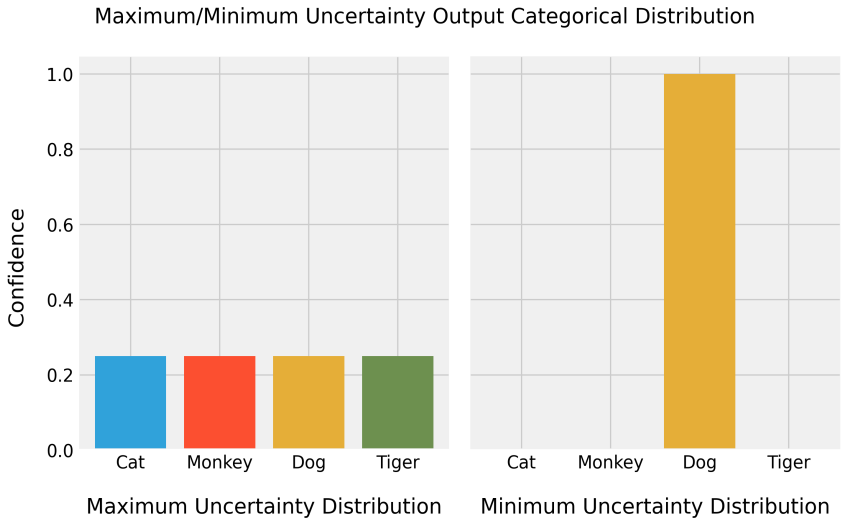


Figure 3.4: Examples of output categorical distributions in a multi-classification problem that express maximum and minimum uncertainty, respectively on the left and right.

this sense, several types of uncertainties can be modelled directly [Manivannan (2020)], from epistemic to homo-or heteroscedastic aleatoric uncertainty to observe some of the discussed models of uncertainty in practice.

Another topic of interest is that of proper scoring rules. Proper scoring rules are widely utilised in deep learning in order to finely represent uncertainty, and these usually combine several probabilistic metrics of interest such as entropy and information [Beluch et al. (2018)].

Further than these, there are even more interpretations of uncertainty to be found in the literature [Tagasovska and Lopez-Paz (2018); Goan and Fookes (2020); Valdenegro-Toro and Mori (2022)]. The wide range of definitions available for the modelling of uncertainty components can become staggering at first approach. Nevertheless, in the field of probability theory and statistics, interpretation plays a heavy role in the conceptualisation of useful abstractions. Probability itself may be digested as frequency-based in the frequentist approach; as belief-based in Bayesian Statistics; and several more. Some authors argue that these notions do not provide sufficient explanation to all use cases and examples encountered in the real world [Ülkümen et al. (2016); Tannenbaum et al. (2017)]. For this reason, literature still struggles to find a common construct that reconciles all the dimensions that arise in practice. In the same manner, the definition and modelling of uncertainty is an ever-changing, ever-growing field of its own. It is natural to expect that many more interpretations of uncertainty may be conjectured in future literature. As such, it is up to practitioners and scientists to select the most useful, most practical model by adapting the concept that more closely relates with the field of application and the desired outcome.

3.2.3 Visualising Uncertainty

Despite the plurality of existent sources and interpretations as discussed previously, the existence of uncertainty can be naturally observed in some aspects of neural networks. Through a visual rep-

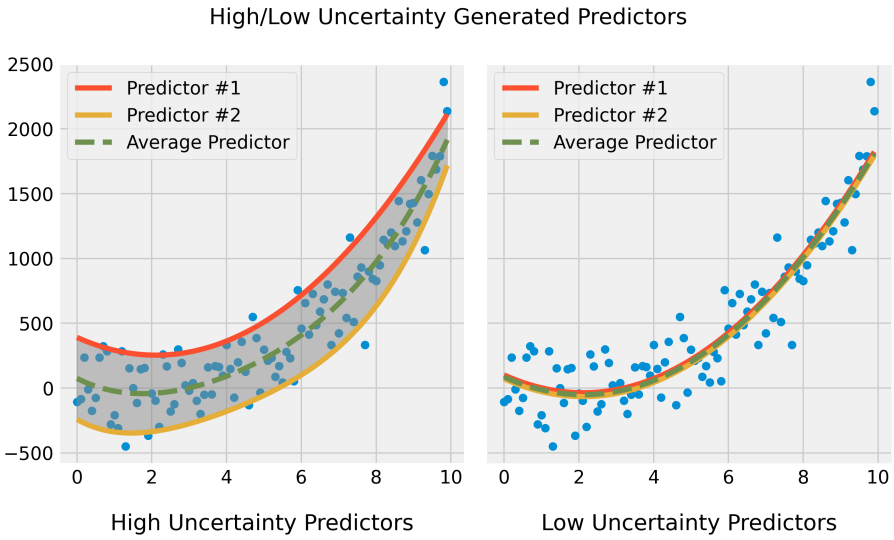


Figure 3.5: Examples of predictors that may be obtained in a regression setting that express high and low uncertainty.

resentation of such characteristics, it is easy to denote how the existence of uncertainty influences the final outcomes of neural networks.

In the setting of classification problems, uncertainty becomes apparent through the categorical distribution of confidence scores. Figure 3.4 features an example of distributions that demonstrate maximum and minimum uncertainty in an example of a multi-class problem with 4 classes. As it can be seen, a highly uncertain model features a predictive distribution with equal confidence for all available classes, $p = \frac{1}{C}$. On the contrary, a model that is entirely confident in its prediction attributes a very high probability to a class with respective low values to the others. This behaviour showcases how uncertain the model is on its chosen output, demonstrating different degrees of confidence.

The same behaviour can be observed in regression settings, albeit with the need for further processing. Since in classification problems a softmax is typically applied in the last layer, leading the output to express a categorical distribution directly, in regression problems the final processing happens in a linear layer in most architectures. Figure 3.5 showcases an example where, in a regression setting, several predictors that model the data demonstrate diverging results, exemplifying how uncertainty can be expressed in these models. In order to aid visualisation efforts, several approaches may be taken, some of which are discussed in forthcoming sections. Ensembles, for example, are a possible method to obtaining different predictors, or the retraining of the same architecture under slightly varied conditions, among many others.

3.3 Uncertainty Quantification Techniques

Concerning the literature of uncertainty quantification methodologies, a taxonomy of the main existing methods is presented in Figure 3.6. The two main groups of techniques are identified as

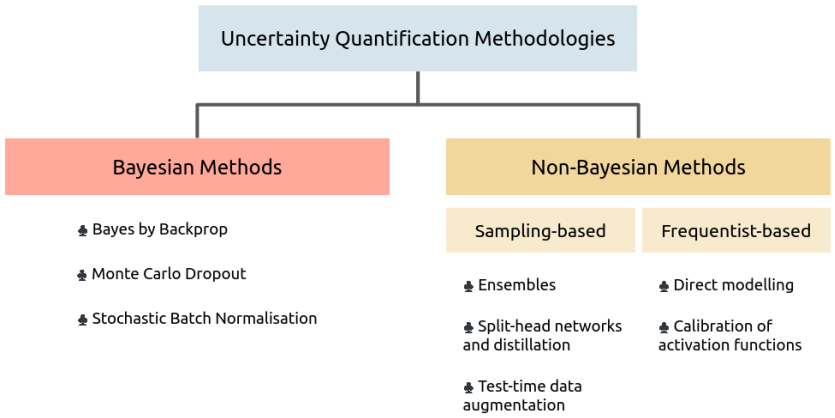


Figure 3.6: A taxonomy of the main methodologies for uncertainty quantification. Bayesian methods make use of stochastic routines carried out within neural networks. Non-Bayesian techniques may be further subdivided into sampling-based and frequentist-based.

Bayesian and non-Bayesian. In the first group, methods such as Bayes by Backprop, Monte Carlo Dropout and Stochastic Batch Normalisation may be found. In the non-Bayesian group, a further subdivision into Sampling-based and Frequentist-based techniques is considered. Frequentist-based methods utilise the frequentist view of probability and include techniques such as direct modelling and calibration of activation functions. Sampling techniques are based upon the principle that several different outputs are obtained for an input sample, thus allowing for the approximation of a distribution analogous to the principle of sampling. In this subgroup, methods such as ensembles, split-head networks and distillation and test-time data augmentation are identified. In the following sub-sections, these methods are explored in further detail including a condensed overview of works that have utilised them in different fields.

A general visualisation of the different basis for estimation techniques is presented in Figure 3.7. On one hand, a point estimate is either adapted or its stochastic components are utilised in order to, through the optimisation process, reach an approximated distribution. On another, a point estimate may be transformed into a set of point estimates that can be used to derive a distribution. These relations naturally reflect the groups identified in the taxonomy, as mentioned above.

Figure 3.7: A representation of the weights on distribution approximation versus sampling techniques. On the left is demonstrated the transformation of a point estimate in a continuous distribution of weights. On the right is showcased a sampling algorithm that generates several different discrete parameter sets.

3.3.1 Bayesian

Following the previously demonstrated taxonomy, in this subsection, Bayesian methods for uncertainty quantification are firstly explored. The three techniques introduced in detail are chosen given their prominence in literature and proven performance results, namely Bayes by Backprop (BbB) [Blundell et al. (2015)], Monte Carlo Dropout (MC-Dropout) [Gal and Ghahramani (2016)] and Stochastic Batch Normalisation (SBN) [Teye et al. (2018)].

These methods are characterised as Bayesian since they rely on the theory of Bayesian Statistics, featuring not only a Bayesian interpretation of probability but also relying on Bayesian inference techniques.

The featured structure is transversal to all techniques with the initial paragraphs being dedicated to a practical and/or theoretical explanation of the method followed by an asterisk under which a brief overview of relevant literature is presented.

3.3.1.1 Bayes by Backprop

Bayes by Backprop, proposed by Blundell et al. (2015), is a practical, backpropagation-compatible parametric variational methodology to estimate a posterior on the weights of a neural network. Further than allowing the estimation of an accurate posterior, the method also automatically applies regularisation to the network. These effects are achieved through the minimisation of a compression cost during training. The so-called compression cost is equivalent to the variational free energy or the expected lower bound (ELBO) on the marginal likelihood. This technique is another example of Variational Inference in neural networks that allows for a richer posterior approximation since it allows parametric distributions to be fit to $q(\omega)$.

*

Building upon the work by Blundell et al. (2015), Fortunato et al. (2017) extend the methodology to recurrent neural networks. The authors test an adaptation of truncated back-propagation and report accuracy improvements.

Bayes by Backprop, or BBB, finds especially wide adoption in medical imagery tasks. Ng et al. (2018) compare BBB with MC-Dropout for cardiac MRI segmentation. The authors also argue the importance of uncertainty quantification in an automated pipeline.

Analogous to image segmentation tasks, de la Riva and Mettes (2019) study the application of BBB to a 3D convolutional network designed to perform action recognition from video sequences. Furthermore, the authors explore the impact of scarce data on the Bayesian network. Their findings showcase that the addition of Bayesian inference increases the network's generalisation, proven by its increased performance as training examples are removed.

These findings suggest that Bayes by Backprop could be further studied and adapted to many networks with promising results. However, since the approximated distribution is multivariate and parametric, the heaviness of computation is parallel to the dimensionality of the parameters. In fact, the memory footprint may also be high since the mean and variance matrices must be kept

in memory through the backpropagation, even more so in the presence of full-rank covariances. Further than this, stochastic variances in SGD may lead to difficulty in convergence, raising the need for further sampling, which is reflected in time expensiveness [Folgoc et al. (2021)].

3.3.1.2 MC-Dropout

Dropout [Srivastava et al. (2014); McClure and Kriegeskorte (2016)] has been a prominent technique applied to neural networks to regularise outputs and avoid overfitting. Srivastava et al. (2014) apply dropout randomly throughout layers with a parametrised probability. Generally, initial layers are assigned a probability of 0.8, with later layers being typically assigned a lower level of 0.5. Even though dropout was initially proposed as a regularisation technique, Gal and Ghahramani (2016) interpret it as an approximate Gaussian process, enabling simple uncertainty quantification in standard neural networks.

In practical application, this technique boils down to applying dropout at test time to obtain different predictions on the same input. This enables the inference of a probabilistic distribution on the output, with Bernoulli-random variables at each neuron being the source of stochasticity. The predicted distribution's expected value becomes the mean of the T extracted outputs, a relationship represented in Equation 3.9. The number of samples, commonly referred to as T, may have an influence on the quality of the uncertainty estimates [Srivastava et al. (2014); Seoh (2020)].

$$P(Y|X) \approx \frac{1}{T}\sum_{t=1}^{T} p(Y|X,\theta_t) \tag{3.9}$$

MC-Dropout can be used in previously trained networks without re-training, making its adoption straightforward. For this reason, it facilitates the integration of uncertainty quantification to already constructed deep learning pipelines, including even already deployed models.

*

Monte Carlo Dropout has seen wide adoption in practice [McClure and Kriegeskorte (2016); Feng et al. (2018); Amini et al. (2018); Hernández et al. (2020); Ghoshal et al. (2020); Combalia et al. (2020)], and, as such, there are many examples of its application possible to be reported.

Liu et al. (2019) apply MC-Dropout when designing a Universal Adversarial Perturbation in order to explore hidden uncertainty in the model. The authors apply this method specifically to CNNs tasked with image classification. On the other hand, in Jin et al. (2019), the augmentation of a classification model is proposed by adding an unsupervised reconstruction pathway. With MC-Dropout utilised in the encoding pathways, the authors estimate predictive uncertainty, resulting in an MC-Dropout classifier and autoencoder. Improved performance is reported, especially in out-of-distribution samples.

Even though the technique has been widely adopted in literature, its variational estimation of the predictive posterior has been shown to be highly innacurate [Folgoc et al. (2021)], leading to

poor and unreliable uncertainty estimates. Nevertheless, due to its flexible and practical implementation, this technique has been widespread to a wide range of areas and fields. For example, in medical imagery analysis, several works can be found [Nair et al. (2018); Yu et al. (2019); Do et al. (2020); Ghoshal et al. (2020); Combalia et al. (2020); Laves et al. (2020)]. Likewise, in the field of AV there is also extensive literature to refer to [Kendall et al. (2017); Amini et al. (2018); Feng et al. (2018); Yelleni et al. (2024)].

3.3.1.3 Stochastic Batch Normalisation

In the same paradigm as MC-Dropout, Stochastic Batch Normalisation (SBN), Teye et al. (2018), takes advantage of stochastic processes already commonly employed in modern neural networks. In fact, whilst MC-Dropout relies on the randomised drop of neurons to encapsulate the uncertainty over the weights, SBN utilises the parameters of Batch Normalisation [Ioffe and Szegedy (2015)].

In order to obtain the uncertainty estimation, a similar process to MC-Dropout is followed, with the drawing of samples through T forward passes where the Batch Norm parameters are altered in each iteration. These parameters are obtained from passing T different batches of data through the network.

*

This method has been shown to perform on par with MC-Dropout unless the batch size is less than 16, where it proves to be highly ineffective due to the intricate workings of batch normalisation [Teye et al. (2018)]. In the fashion of sampling-based techniques, SBN also sports heavy computational costs.

Atanov et al. (2019) discuss the use of a distribution of batch norm parameters during training which would facilitate the sampling of parameters during the forward passes in the test set. The authors recommend using normal and log-normal distributions respectively for the approximation of the mean and standard deviation.

In Brach et al. (2020), the authors suggest using single-shot approximation to enable scalability and make test periods faster, thus contributing to the lessening of the computational expensiveness of this method. Another procedure that enables better time complexity is saving the T batch parameters directly during training. However, this leads to a trade-off with the memory toll of the training process.

3.3.2 Non-Bayesian

In this subsection, the other major group featured in the presented taxonomy, that of non-Bayesian methods for uncertainty quantification, is explored. The techniques that are introduced are once more chosen due to their prominence in literature and performance, being further sub-divided into the Sampling-based and Frequentist-based subgroups. In the first one, ensembles, split-head networks and distillation, and test-time data augmentation are discussed. Following this, the second group features direct modelling and calibration of activation functions.

The aforementioned methods are characterised as non-Bayesian since these either rely on producing multiple examples (point estimates) in the fashion of sampling procedures (sampling-based group) or on a frequentist view of probability (frequentist-based group).

Following the previously presented structure, all techniques contain initial paragraphs that are dedicated to a practical and/or theoretical explanation of the method followed by an asterisk that marks the start of a condensed overview of relevant literature.

3.3.2.1 Deep Ensembles

Ensembles are combinations of several models that can be seen as a larger single model. Due to the inherent characteristics of ensembles, their use can be extended to perform uncertainty quantification [Lakshminarayanan et al. (2017)].

In practice, ensembles already output an approximation of a predictive distribution. Since this methodology enables the coexistence of several different sub-models within a larger model, it is straightforward to obtain a group of T predictions from the same input sample. In this case, the number of samples T will correspond to the number of sub-models that are encapsulated in the ensemble and is thus a parameter to optimise in practical application.

*

Deep ensembles in specific are commonly referred to as an alternative to Bayesian Neural Networks, tackling some of its disadvantages. In fact, these are simple to implement, require less hyper-parameter tuning, can be used in parallel settings and have been reported to yield accurate results in uncertainty estimation. As an exemplary title, Lakshminarayanan et al. (2017) study the estimation of predictive uncertainty using deep ensembles and report results comparable to BNNs in both classification and regression tasks.

Nevertheless, ensembles can be memory intensive and may require high computational power, which in turn can make their application to down-scaled systems such as AVs unfeasible [Malinin (2019)]. This disadvantage has incited studies on the adaptation of deep ensembles to memory-intensive settings. Dorjsembe et al. (2021) address this memory issue with model pruning and quantization, for example. The authors report that an increase in the sparsity provoked by pruning is accompanied by a rise in uncertainty estimation, thus making the model more robust to perform in uncertainty populated tasks. Furthermore, the authors claim that a pruned, energy-efficient model can be appropriate even for memory-intensive tasks.

On the other hand, in Wenzel et al. (2020), deep and batch ensembles are studied, with the proposal of a method with notably lower costs of computation and memory use, namely *hyper-batch ensembles*. The authors claim that these are more parameter-efficient models. For memory-independent systems, the authors propose *hyper-deep ensembles*, which ensemble over the weights and hyperparameters, leading to a higher diversity in both characteristics and thus heightened performance.

Malinin (2019) propose performing distillation of ensembles into a single model in order to counter the memory intensiveness spawned from the use of several models. Moreover, the authors

tackle the general loss of differentiated forms of uncertainty that is associated with distillation by using a Prior Network capable of distilling both the predictions and the variety of information represented by the original ensemble. In Rahaman and Thiery (2021), the authors argue that deep ensembles do not necessarily produce a correctly calibrated probabilistic model and propose slight adjustments that can improve the process in relation to a standard deep ensemble.

S. Salem et al. (2020) perform aleatoric and epistemic uncertainty estimation using an ensemble of networks. In order to ensure interpretability from both machine and human observers, uncertainty is represented in prediction intervals, with a loss function controlling the quality of the relationship between point estimates and intervals. Concerning prediction intervals (PIs), in Pearce et al. (2018), the authors employ quality-driven ensembles, reducing PI width significantly through the use of a direct loss function with no distribution assumption.

Beluch et al. (2018) propose using an acquisition function to determine further uncertainty information. The authors describe the training of a deep-learning-based classification model on a set of previously defined samples from the training set whilst uncertainty is evaluated in the hold-out samples. The acquisition function encapsulates several measures of uncertainty such as entropy, mutual-information and variation ratio. Application is demonstrated through the implementation of an active learning pipeline. In this pipeline, the samples with the maximum value of the acquisition function are included in the training set, with the process being repeated through a chosen number of steps. This ensemble-based acquisition function yields better results than MC-Dropout, as the authors demonstrate.

BNNs have scalability limitations, are often hard to train and can have poor generalisation to out-of-distribution shifts, as commented in prior literature Sinha et al. (2021). In light of this, Sinha et al. (2021) propose using ensembles with an adversarial loss that foments diversity in prediction and showcases significant improvements on several benchmark datasets.

3.3.2.2 Split-head networks and distillation

Other possible approaches for uncertainty approximation relate to split-head networks and knowledge distillation [Matsubara et al. (2020); Matsubara and Levorato (2021)]. A multi-headed neural network allows the distillation of the knowledge contained within a backbone model that is branched into several heads that produce different point estimates. The set of point estimates can be seen as the results of a sampling procedure from a probability distribution, allowing for the approximation of the predictive distribution, making the retrieval of uncertainty trivial. One of the significant advantages of this technique is its compatibility with real-time applications. The distillation may occur only on the final layers of the network, thus reducing computations, or encapsulate further layers if desirable.

*

HydraNet [Tran et al. (2020)] is an example of a split-head network, and it allows for computations to be reduced by deriving a distribution on the final layers of the network. The same

behaviour can be replicated in other methods, such as Bayes by Backprop and MC-Dropout, with the same objective. In Bayes by Backprop [Blundell et al. (2015)], point estimates may be kept in initial layers, achieving the variational approximation of the parameter posterior using only the final layers. In the same fashion, dropout may be applied only on later layers, improving predictions by keeping some of the backbone knowledge of the network intact. This enables more efficient memory management in techniques that may otherwise have a larger footprint.

The quality of the estimates produced by a split-head method are unclear at the moment as further research efforts in the topic must be carried out.

3.3.2.3 Test-time Data Augmentation

Test-time data augmentation takes advantage of another common deep learning mechanism, that of data augmentation, in order to obtain a predictive distribution. The methodology consists in generating T augmentations of the same input that, when fed to the network, allow the derivation of a predictive distribution per sample. Since a distribution is calculated for each data point, the resulting uncertainty is heteroscedastic aleatoric [Ayhan and Berens (2018)].

*

Wang et al. (2019) consider the estimation of aleatoric and epistemic uncertainty in CNN-based 2D and 3D medical image segmentation tasks. Further, they propose a theoretical formulation of test-time augmentation, reporting the advantage of their method in relation to the respective baselines.

In Moshkov et al. (2020), test-time augmentation is utilised for cell segmentation in microscopy images. The authors report that TTA allows for the reduction of data dependency and even with simple transformations such as rotation or flipping, prediction accuracy can be substantially improved. This suggests that TTA may provide further benefits than just the availability of uncertainty estimates.

3.3.2.4 Direct Modelling

A method of interest for approximating aleatoric uncertainty is direct modelling [Feng et al. (2021); Kendall and Gal (2017)]. In direct modelling, a probability distribution is assumed over the outputs of the network, and the parameters for said distribution are directly modelled by the model's output layers. In order to achieve this, additional neurons may be added in the final layer of the model so that the outputs are transformed into the mean and variance.

A derivation of a loss function for heteroscedastic direct modelling may be achieved as follows:

$$p(y|f_\theta(x),\theta) \sim \mathcal{N}(y;\mu,\sigma) \propto \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}(\frac{y-f_\theta(x)}{\sigma})^2\right\} \tag{3.10}$$

$$\begin{aligned} \mathscr{L}_{NLL}(\theta) &= -\frac{1}{n}\sum_{i=0}^{n} \log p(y_i|x_i,\theta) \\ &= -\frac{1}{n}\sum_{i=0}^{n} \log \frac{1}{\sigma_i\sqrt{2\pi}} - \frac{1}{2\sigma_i^2}(y_i - f_\theta(x_i))^2 \\ &= \frac{1}{n}\sum_{i=0}^{n} \frac{1}{2\sigma_i^2}(y_i - f_\theta(x_i))^2 + \frac{1}{2}\log\sigma_i^2 \end{aligned} \tag{3.11}$$

*

Depending on the application, it is possible to model both homoscedastic and heteroscedastic uncertainties through this method. For regression probabilities, for example, Gaussian distributions [Kendall and Gal (2017); Gast and Roth (2018)] and Gaussian Mixture Models [Choi et al. (2018a); Meyer et al. (2019)] are typically used.

3.3.2.5 Calibration of activation functions

Typically, classification models already output a distribution over classes that represents the aleatoric parcel of uncertainty through the softmax activation function. Even though this approximation of aleatoric uncertainty is readily available, it has been found to be erroneous due to a lack of calibration [Guo et al. (2017); Neumann et al. (2018)]. As such, several methods have risen for the calibration of softmax, with this idea being extended to regression tasks as well. This category of methodologies is rather computationally light since almost no additional work is performed within the network.

*

Relevant calibration techniques range from Histogram Binning [Zadrozny and Elkan (2001)] to Isotonic Regression [Zadrozny and Elkan (2002)] and Platt Scaling [Platt (1999)] for binary models. For multiclass cases, adjusted binning methods Zadrozny and Elkan (2002) exist as well as Temperature Scaling [Guo et al. (2017)]. The same idea has been extended to regression models by Laves et al. (2020).

Ever since calibration of activation functions has been denoted as relevant to both reliable estimates and uncertainty quantification, more evolved calibration methods have been proposed in literature. Since the scope of these is quite large, we leave the following references for further study [Ma et al. (2021); Wang et al. (2021); Küppers et al. (2022); Gong et al. (2022)].

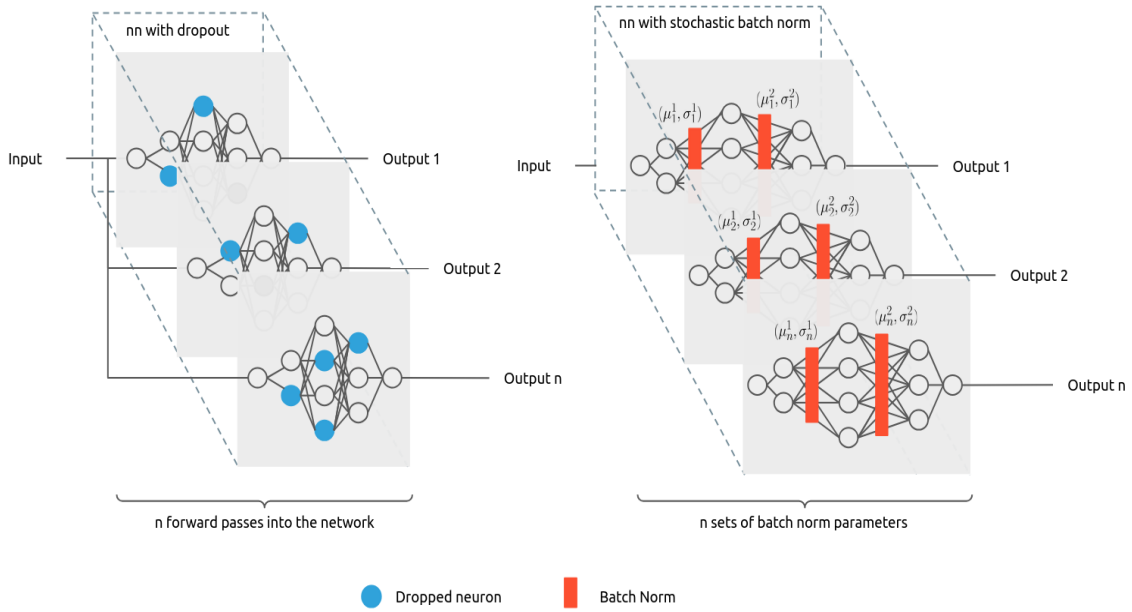


Figure 3.8: A representation of Monte Carlo Dropout, on the left, versus Stochastic Batch Normalisation, on the right.

3.3.3 Other Remarks

In order to clarify the distinctions between some of the aforementioned methodologies, diagrams that highlight the main differences between methods that may show similarities in their procedure are presented.

Figure 3.8 demonstrates the differentiating methodology between Monte Carlo Dropout and Stochastic Batch Normalisation. As it can be observed, the random variables that are considered in MC-Dropout, depicted on the left, originate in the dropout parameter of each neuron. On the other hand, on the right side of the diagram, the randomness originated in the consideration of different batch normalisation parameters (μ_n^l, σ_n^l). n sets of parameters may be found per each layer l of the network.

One of the key differences in the calculation of uncertainty is found in the consumption of the input. Using Monte Carlo Dropout, the same input sample needs to be introduced to the network n times so as to obtain n different outputs. On Stochastic Batch Normalisation, the sample may be thought of as being fed to the model once and the different batch normalisation parameters are leveraged in parallel in order to obtain different results. Both methods rely on the derivation of a predictive distribution through the consideration of the n outputs. However, these are obtained through diverging internal mechanisms of standard operations commonly found in neural networks.

It becomes apparent that each may have significant drawbacks in opposite domains. Monte Carlo Dropout requires the n-times multiplication of the testing time, turning computations heavier. Depending on the application domain, this can reveal rather costly. Stochastic Batch Normalisation heightens the memory footprint, with the n-times multiplication of the number of parameters

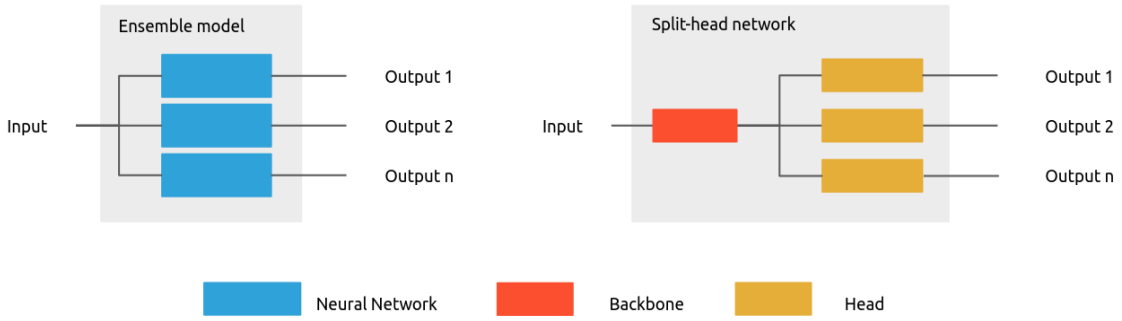


Figure 3.9: A representation of ensembles, on the left, versus split-head networks, on the right.

that have to be kept during operations. This characteristic, in turn, may also hinder implementation in certain domains.

Figure 3.9 delineates the differences between a standard ensemble and a split-head network. These methods rely more on architectural modifications rather than the source of randomness for the derivation of uncertainty. Ensembles may consider several different sub-models, leading to n outputs - one per each sub-network in the ensemble. Instead, split-head networks may keep a shared backbone network that performs feature extraction coupled with n heads that lead to n outputs. Once more, both methods utilise the n obtained outputs as samples to form a predictive distribution. Nevertheless, each output is obtained through what could be named a *physical* manifestation of either a network or a part of its architecture with the randomness lying in the optimisation procedure rather than the internal operations' parameters.

Even though the discussed taxonomy highlights some methods for uncertainty quantification, several other alternative techniques may be found. The methodologies presented were chosen since due to their common application in the literature, which suggests more widespread acceptance and application. This also highlights the effectiveness of the presented methods, which have been widely tested and evaluated by the scientific community prior to this thesis. For reference, refer to a small list of other possible approaches to uncertainty quantification as follows:

- Hyper Networks [Pawlowski et al. (2017); Krueger et al. (2017)]
- Laplace Approximations [Foong et al. (2019); Hobbhahn et al. (2022)]
- Mixture Density Networks [Bishop (1994); Choi et al. (2018b)]
- Prior Networks [Malinin and Gales (2018)]
- Epistemic Neural Networks [Osband et al. (2023)]
- Direct Epistemic Uncertainty Prediction [Lahlou et al. (2023)]
- Perturbation Theory [Wen et al. (2018); Csáji and Kis (2019); Liu et al. (2019); Boiarov et al. (2020)]

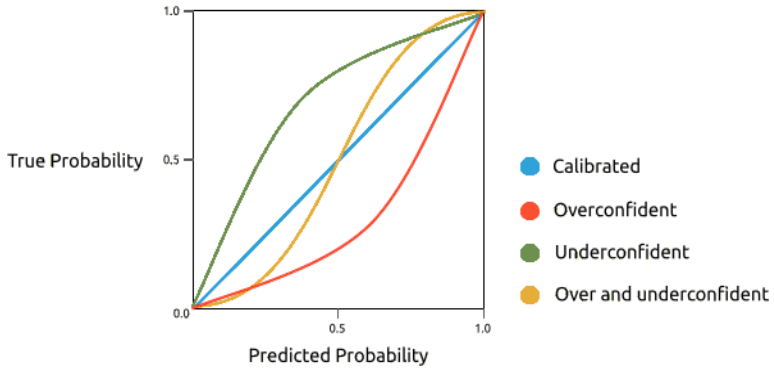


Figure 3.10: Calibration plots of a sample model. The predicted probability is the confidence measured in the model whilst the true probability is obtained through empirical evidence. Blue illustrates a calibrated model whilst red and green demonstrate respectively an over and underconfident model. Models may be over and underconfident simultaneously, represented in orange.

3.4 Calibration and Evaluation of Uncertainties

Having thoroughly discussed uncertainty and methods for its quantification in deep learning pipelines, it is pertinent to assess the fitness and coherence of these estimates. This evaluation is based upon some relevant characteristics that estimates must express in order to allow for their safe deployment in pipelines, especially in safety critical systems such as autonomous vehicles. These characteristics, supported in literature [Feng (2021)], are pivotal to the validation of forthcoming proposals and may be summarised as follows:

Explainability Higher uncertainty must correspond to more complex samples.

Calibration Confidence scores are coherent and have the same probabilistic interpretation regardless of model.

Usefulness Estimates enable the improvement of the overall system.

In line with this, an overview of the most relevant metrics for uncertainty estimation evaluation and calibration assessment is presented in the next subsections for both classification and regression problems.

Another aspect to denote that is indispensable to the motivation behind the metrics that are discussed is how calibration is expressed by a model in its output distribution. Figure 3.10 plots the predicted probability versus the true probability achieved in all samples of the dataset, thus demonstrating the distinction between *calibrated* and *uncalibrated* models. In a model with perfect calibration, the predicted confidence equals the empirically observed one. This is represented through a linear relationship, shown in blue on the plot. Uncalibrated models, on the other hand, may exhibit different behaviours. A model that constantly predicts a probability higher than the observed one is overconfident, a behaviour illustrated in the red curve. The opposite, underconfidence, shown in green, is present when the model predicts a lower probability than that empirically

achieved. There is also the possibility for an uncalibrated model to be overconfident in some areas and underconfident in others, demonstrated through the orange curve in Figure 3.10.

3.4.0.1 Classification

For classification tasks, the Expected Calibration Error (ECE) and the Brier Score are introduced.

Expected Calibration Error In a well-calibrated classification model, the confidence attributed by the model to a given prediction must be equal to the empirical probability of the sample belonging to that class. For example, in perception tasks, the categorisation of an object as a given class with a confidence of 80% is expected to be correct 80 out of 100 times.

In this sense, miscalibration may be interpreted as the difference between the expectation of accuracy and confidence. This relation, demonstrated in Equation 3.12, can be approximated with the Expected Calibration Error.

$$\mathbb{E}_{p^*}[|\mathbb{P}(y^* = y|p^* = p) - p|] \tag{3.12}$$

The Expected Calibration Error (ECE) [Guo et al. (2017)] can be calculated through the following methodology:

- Perform inference over the full dataset and register model confidences;
- Split the dataset into n buckets according to the obtained confidences;
- For each bucket B_i compute the average accuracy of the model $\text{acc}(B_i)$;
- For each bucket B_i compute the average confidence of the model $\text{conf}(B_i)$;
- Calculate the calibration error for the bucket as the absolute difference between model confidence and model accuracy;
- Obtain the ECE by averaging calibration error over all buckets.

As such, the ECE metric can be interpreted as the average ratio between model confidence and accuracy, as depicted in Equation 3.13.

$$\text{Expected Calibration Error (ECE)} = \sum_{i=1}^{n} \frac{|B_i|}{n} |\text{acc}(B_i) - \text{conf}(B_i)| \tag{3.13}$$

Brier Score The Brier Score (BS) measures the difference between predicted confidence and ground truth values, as demonstrated in Equation 3.14.

$$\text{Brier Score (BS)} = \frac{1}{n} \sum_{i=1}^{n} (c_i - g_i)^2 \tag{3.14}$$

where c_i is the predicted confidence and g_i is the ground truth binary class indicator.

3.4.0.2 Regression

For the task of regression, the modified Expected Calibration Error (ECE) and the Sharpness metrics are reported.

Expected Calibration Error In a well-calibrated regression model, it is expected that the predicted confidence intervals at a given level include the true probability at the respective level. This signifies that a confidence interval with a level of 80% should encapsulate the true probability 80 out of 100 times.

In the same fashion as the ECE for classification, the following steps must be followed for its calculation:

- Perform inference over the full dataset and register model predictions;
- For each threshold p_j compute the probability p_j^* of a sample having a predicted CDF value lower than p_j;
- For each threshold p_j compute the calibration error for the threshold;
- Obtain the ECE by averaging the calibration error over all thresholds.

The ECE in regression tasks may thus be interpreted as the average calibration error over a sample of thresholds. This relationship is illustrated in Equation 3.16 with Equation 3.15 demonstrating the calculation of p_j^* for a given threshold.

$$p_j^* = \frac{|y_i | F_i(y_i) \leq p_j, i = \{1, ..., k\}|}{k} \tag{3.15}$$

where y_i is the ground truth value and $F_i(y_i)$ is the predicted CDF value for sample y_i.

$$\text{Expected Calibration Error (ECE)} = \frac{1}{n} \sum_{i}^{n} |p_i - p_i^*| \tag{3.16}$$

Sharpness The Sharpness metric relates to the amount of information that the predictive distribution contains rather than only considering its calibration. As such, it corresponds to the mean of the variances of the predicted distributions, represented in Equation 3.17.

$$\text{Sharpness} = \frac{1}{n} \sum_{i=1}^{n} \sigma_i^2 \tag{3.17}$$

where σ_i^2 is the variance of prediction i.

3.4.0.3 Classification and Regression

For metrics with application in both classification and regression tasks, the Negative Log-Likelihood and Spearman's rank correlation coefficient are discussed. In particular, Spearman's coefficient has a direct application to the epistemic uncertainty.

Negative Log-Likelihood The Negative Log-Likelihood (NLL) is the negative logarithm of the likelihood of the ground truth values under the distribution predicted by the model. The NLL has seen wide adoption as a loss function for machine and deep learning applications. It relates to uncertainty since models that are better at estimating uncertainty have lower NLL [Guo et al. (2017); Feng et al. (2021)]. This relationship is obtained as follows:

$$\text{Negative Log-Likelihood (NLL)} = -\frac{1}{n}\sum_{i=1}^{n}\log f_\theta(y_i|x_i) \tag{3.18}$$

Spearman's rank correlation coefficient (Epistemic Uncertainty) Spearman's rank correlation coefficient or Spearman's ρ is a measure of the rank correlation between two variables. Epistemic uncertainty estimates should have a high positive correlation with sample error. As such, ρ is defined as follows:

$$\rho = \frac{\text{cov}(R(X), R(Y))}{\sigma_{R(X)}\sigma_{R(Y)}} \tag{3.19}$$

where $R(X)$ is the set of rank variables of X, $\text{cov}(R(X), R(Y))$ is the covariance between the rank variables and $\sigma_{R(X)}$ is the standard deviation of $R(X)$.

3.5 Chapter Summary

Building upon the theoretical background presented in Chapter 2, this chapter focuses on the topic of uncertainty, presenting an in-depth discussion. In order to provide a formal, structured overview of the wide range of topics involved in the field, the first aspect of discussion is a formal definition of uncertainty as expressed in a plurality of domains, followed by how this quantity can be digested and quantified.

Pertaining to this, Section 3.1 first reports on the field of uncertainty quantification, defining its problematic and central objective, and motivating its usefulness. Following this, the definition of uncertainty is approached in Section 3.2 with the exploration of aspects of interest. This ranges from providing a formal definition of uncertainty, to discussing its properties, sources and interpretations in prior literature, finalising with methods for the visualisation of uncertain behaviour in classification and regression problems.

Having established a complete notion of uncertainty, its quantification methods are then reported in Section 3.3 where a taxonomy of the main methodologies is presented alongside the definition and a condensed discussion of the literature on each technique. This section is divided into two main subgroups, each representing Bayesian and non-Bayesian methods.

Finally, Section 3.4 reflects existent notions on the calibration and evaluation of uncertainty estimates. Several metrics of interest are defined over the groups of classification, regression, and problems that tackle both. These metrics enable the validation of the coherence of estimates with a focus on the three main desirable characteristics of uncertainty - explainability, calibration and usefulness.

Chapter 4

Interpretations of Uncertainty and Applications

As established previously, uncertainty is an inherent characteristic of neural networks. Especially in safety-critical applications, estimating this uncertainty is paramount to the safe deployment of deep learning in a panoply of tasks. It has been demonstrated that existent methods either are not compatible with real-time inference or need the retraining and repurposing of existent pipelines into a probabilistic framework. These characteristics greatly hinder the application of uncertainty estimation methods, especially in already deployed neural networks.

Building upon these ideas, this chapter discusses the central proposals of this thesis, all surrounding the concepts of existence and uncertainty. In this sense, Section 4.1 first formally defines the problems that are targeted with the proposed methodologies. So as to support the definitions and concepts layed out in the chapter, Section 4.2 discusses preliminary notions indispensable to the full understanding of the proposals. Following this, the first application of existence is discussed in Section 4.3, the Existence Map, including the establishment of the necessary mathematical formulations.

Adding to these ideas, interpretations of uncertainty for both the aleatoric and epistemic parcels are explored in Section 4.4. Concerning the proposed models of uncertainty, an application is reported in Section 4.5, the Uncertainty Map. Finally, distilling both the existence and uncertainty concepts previously proposed, the Existence Probability is derived in Section 4.6, a merging strategy that integrates uncertainty and the Existence Map feedback into the standard outpu. To finalise the proposals, additional considerations on existence and uncertainty maps are detailed in Section 4.7.

Having established the proposals, some qualitative evaluations in the object detection domain are considered in Section 4.8. Finally, the discussion of the overall results and proposed methodologies is carried out in Section 4.9, including a comparison with existent literature. The Chapter Summary Section 4.10 presents another overview of the information contained within the chapter.

4.1 Problem Definition

Current object detection models face limitations in their ability to provide probabilistic and interpretable outputs that go beyond traditional deterministic bounding boxes and fixed closed-set classification. This problem includes multiple key challenges:

Probabilistic Output for Enhanced Information Traditional object detectors provide fixed bounding boxes with deterministic classifications, often lacking detailed information about the certainty of each detection. There is a need for models that offer probabilistic outputs, which could convey varying levels of confidence in both object presence and location. This probabilistic approach could lead to more informative predictions, which are valuable for tasks requiring nuanced decision-making.

Classification of Object Existence Object detection should not only localize objects but also classify them based on existence—determining if an object is indeed present or absent in a scene. Current models often assume objects within bounding boxes are always valid detections, but a model capable of differentiating between existing and non-existing objects could mitigate false positives and improve overall detection accuracy.

Recognition of Objects as Global Concepts Current detectors generally operate within a closed-set classification framework, meaning they are restricted to recognizing objects from predefined categories. There is an opportunity to study and enhance detectors to recognize objects based on a broader, more generalized concept of *objectness*. This would support open-set detection and enable the recognition of previously unseen object classes in a more flexible, concept-based manner.

Interpretation of Model Uncertainty Object detectors frequently lack mechanisms to convey uncertainty in their predictions. By leveraging insights from the region proposal module, object detectors can estimate and communicate uncertainty, allowing users to gauge the reliability of each detection and better interpret the results. Furthermore, these estimates should be holistic, dividing into two parcels of interest, the aleatoric and epistemic uncertainties.

Communication of Uncertainty Even with the availability of uncertainty estimates, the form of communicating and distilling their knowledge is still an open question. As such, visualisations of uncertainty in the form of maps of the input can be invaluable for scientists and end-users alike, giving a degree of confidence in the results showcased by *black box* models.

Calibration of Detection Outputs Many object detection models are not well-calibrated, meaning the confidence scores may not accurately reflect the true likelihood of an object's presence. Improving the calibration of model outputs would allow for more reliable detections, enhancing the model's usefulness in applications where precise confidence levels are critical, such as in safety or autonomous systems.

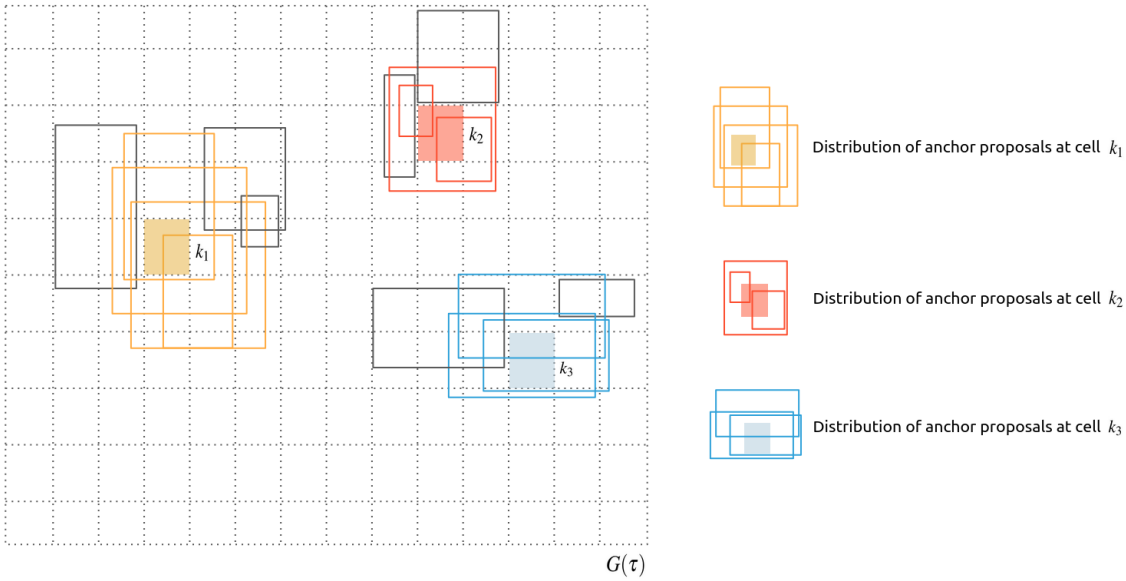


Figure 4.1: Representation of possible anchor distributions at each cell of a grid G with precision τ. Each cell k only considers the set of anchors that intersected it spatially, namely A_k.

As such, the problem statement for this chapter may be resumed as follows: to develop methodologies for object detection models that enable probabilistic, better calibrated outputs that can classify object existence, recognize objects beyond closed-set categories, and interpret uncertainty within predictions. These methods should address the limitations of traditional detection systems by providing richer, more reliable information on both the presence and concept of objects, ultimately making object detection outputs more interpretable and accurate.

4.2 Preliminaries

Before the main proposals in this chapter are discussed, it is important to clarify some key notions and assumptions made in order to develop the forthcoming concepts.

Consider anchor proposals, A, made by a region proposal module of a deep learning-based object detection model (following a standard architecture as described in Section 2.4.1). An anchor can be described as $\mathrm{A}(c_x, c_y, c_z, d_x, d_y, d_z, \alpha)$, including the centre coordinates, the dimensions of the box and its heading. Anchor characteristics contemplated in this thesis are standard to object detection models. Since the field of application of this thesis is autonomous driving, the formulations include anchor proposals over three dimensions. Nevertheless, the regressions shown in following chapters do not depend on this specific shape, with the presented notions being directly applicable to anchor proposals in any other dimensional spaces.

An input sample $\mathbb{S}$ to this aforementioned object detection model can be transformed into a grid, G with a given parametrisable precision τ, resulting in $G(\tau)$. Since anchor proposals are overlaid over the space of the entire sample $\mathbb{S}$ (for more information refer to Section 2.4.2), several cells on this grid are bound to be included in many anchor proposals.

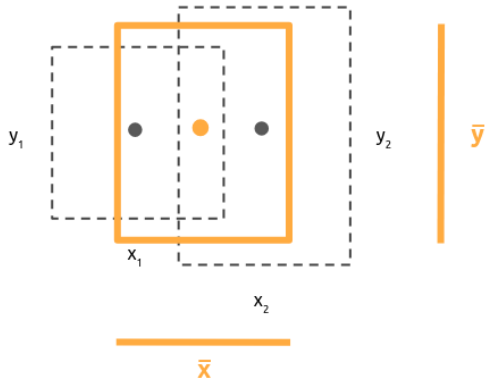


Figure 4.2: Representation of a possible patch, in yellow, that is defined by the intersection of a set of anchor proposals, in grey. The centre and dimensions of the patch are taken from the average of the characteristics of boxes that belong to the intersection.

In fact, a cell, k, on this grid can be intersected by a variable amount of anchor proposals, leading to the set A_k which contains all anchors whose area crossed this given cell. Intersections are defined over the space of the grid. This signifies that, upon transformation of the anchor's entire area into the grid's representation, all cells that enclose anchor's boundaries and contents are selected, regardless of the percentage of intersection. The precision, τ, of the grid thus controls the level of simplification performed to the data in question. These notions are described in Equation 4.1 where the notation for the set of anchors proposed for a sample and the set intersecting a given cell is defined.

$$\begin{aligned} \mathrm{A}_k &\subseteq \mathrm{A}_{\mathbb{S}} \\ \text{with } \mathrm{A}_k &= \{\mathrm{A}(\mathscr{T}(c_x,c_y,c_z,d_x,d_y,d_z,\alpha) \cap k) \mid k \in G(\tau)\} \end{aligned} \tag{4.1}$$

where $\mathrm{A}_{\mathbb{S}}$ is the set of all proposed anchors for sample $\mathbb{S}$, $G(\tau)$ is the generated grid and $\mathscr{T}(c_x,c_y,c_z,d_x,d_y,d_z,\alpha)$ is the transformation function that maps an anchor to a group of cells in the grid.

Considering these several proposals found in either a cell or an area of $G(\tau)$ as random variables, a probability distribution can be defined over a set of anchors. A probability from this distribution is referred to as p_a which is a short-hand notation for $p(x = a|\mathbb{S})$, namely the probability of proposal a in a given set of anchors A in a sample $\mathbb{S}$. A visual example of possible cell anchor distributions in a grid is showcased in Figure 4.1.

Each anchor proposal itself is classified in the form of a vector of confidences by the region proposal module. In this sense, each random variable (each anchor) expresses a distribution, namely $f(x = a|\hat{\pi})$, which is categorical, as defined in Equation 4.2.

$$\begin{aligned} f(x = a|\hat{\pi}) &= Cat(a;\hat{\pi}) \\ \text{with } \hat{\pi} &= \mathscr{F}(f_{rp}(c_x,c_y,c_z,d_x,d_y,d_z,\alpha)), \hat{\pi}_c \geq 0 \end{aligned} \tag{4.2}$$

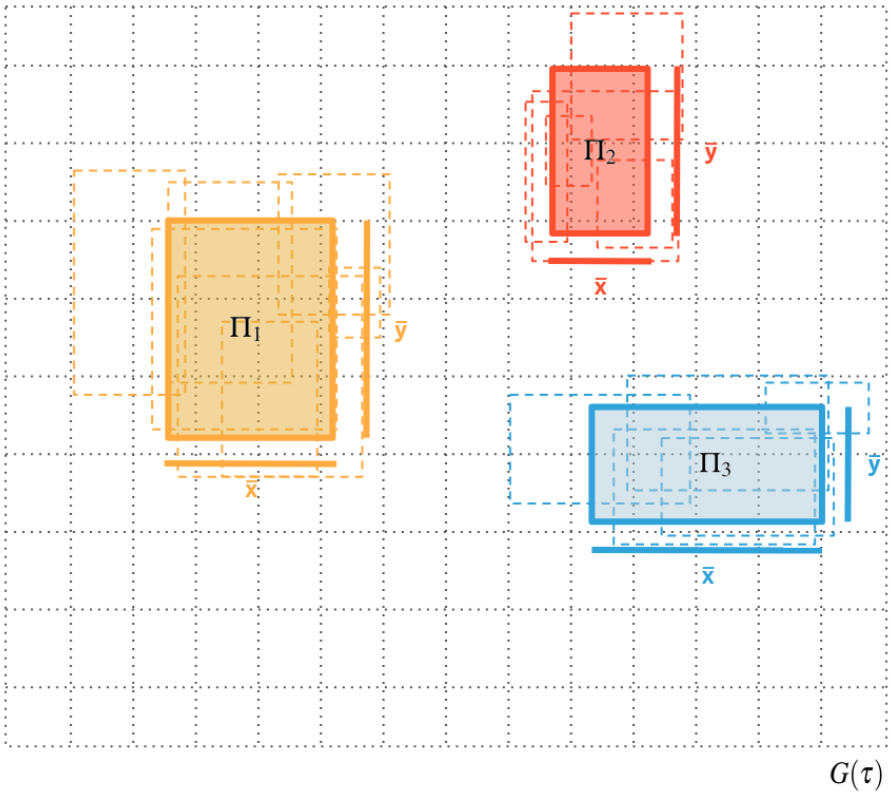


Figure 4.3: Each grid may have several patches distributed over its space. Since only the n highest ranking proposals are analysed, these patches have a high probability of containing objects.

The output of the region proposal layer f_{rp} is passed through a function $\mathscr{F}$ to ensure that values are between zero and one. These refer to a given anchor with characteristics $(c_x, c_y, c_z, d_x, d_y, d_z, \alpha)$.

In order to regress p_a, several approaches may be taken, depending on the chosen interpretation. Further sections explore different interpretations of this probability distribution with various aims.

Further than the concentration of anchors in cells, it is also of interest to analyse the distribution of proposals over areas of the grid. With this objective in mind, the concept of a patch is introduced.

A patch, Π, is defined as a set of cells in the space of grid $G(\tau)$ that results from an intersection of anchors. This means that each patch represents an area in space where anchors concentrate and overlap, as represented in Figure 4.3. The patch's borders are defined as the average dimensions and centre coordinates of all the anchor proposals in the intersection. Anchor headings are irrelevant to the patch's definition since its aim is to contain the majority of the space represented by a set of anchors, instead of defining a classical bounding box. Figure 4.2 illustrates the regression of a patch from a distribution of two anchors.

In order to define patches where there is a high probability of containing objects, only the n highest classified anchors are kept, with n being parametrisable and its effects in the resulting distributions and efficiency being studied in further sections. Due to this, an input sample

$\mathbb{S}$ can feature several patches in areas with a high chance of including objects, as represented in Figure 4.3.

In forthcoming definitions, the logarithm function is recurrently used due to the definitions relying on the expression of Shannon's entropy. In this sense, the base of the logarithm only controls the magnitude of the end result, and its interpretation - base 2 conveys bits, base e communicates natural units and so on. As such, the base is omitted from all expressions, being a choice of the practitioner or scientist that applies the proposals. For the scope of the evaluations carried out in this thesis, the base 2 is selected for all experiments.

4.3 Existence Maps

Existence maps aim to provide an estimation of areas where objects may exist in the input samples of an object detector. In order to encode these areas, a grid with arbitrary precision is overlaid in the input data and each cell in this grid is attributed a probability that reflects the confidence in the existence of an object in it. This grid, following the specifications layed out in the previous section, can be transformed to represent the input sample $\mathbb{S}$ with probabilistic suggestions of areas where a recognised object may exist.

In order to obtain a grid of probabilities, the distribution of anchor proposals is considered. This signifies that existence maps are a product of the region proposal layer of an object detector and thus directly represent the knowledge that is hidden within this module of the network.

Following this idea, the first consideration to be made is the object probability. The anchors that are contemplated for the calculation of the object probability of a given cell k are the ones that intersect the cell on the grid after performing the transformation to the space of the existence map, as previously described in Equation 4.1.

The object probability o_k is defined for a cell k as the maximum probability of all the anchors that intersect the cell, as demonstrated in Equation 4.3. Each cell is assigned only the maximum value found in the intersection in line with the ideas of non-maximum suppression [Rosenfeld and Thurston (1971)].

$$\mathrm{o}_k = \max_{\mathrm{A}_k} (p_a) \tag{4.3}$$

The region proposal layer overlays anchor proposals, a, in every considered location over the input data and attributes a value vector to each proposal, namely a_c with $c \in [1, C]$ where C is the number of classes contemplated in the classification. These output vectors must be transformed into a single value so that, after probabilities are merged in each cell, there is only a unified resulting value for each. In this case, the maximum probability found in the class confidence vector is kept. This signifies that the resulting value assumed as the probability of the anchor containing an object, mentioned as p_a in Equation 4.3, is then the maximum of this categorical

distribution, as shown in Equation 4.4. Probabilities are obtained with the sigmoid function.

$$\begin{aligned} p_a &= \max_C \{f(x=a|\hat{\pi})\} \\ \text{with } \hat{\pi} &= \sigma(f_{rp}(c_x, c_y, c_z, d_x, d_y, d_z, \alpha)) \end{aligned} \tag{4.4}$$

Since classification intends to categorise the contents of an anchor into a pre-defined set of classes, this output does not directly signify the probability of an object existing in the considered location but rather the likelihood of the anchor contents belonging to each class. As such, it is considered that an anchor's probability may not exceed the maximum likelihood found within the class vector leading to the assumption of the greater value as the confidence of the anchor.

The generation process then follows a simple procedure, as seen in Algorithm 1. The grid starts as a matrix with zeroed probabilities of shape $((r_{\max}(x) - r_{\min}(x))/\tau) + 1, (r_{\max}(y) - r_{\min}(y))/\tau) + 1, 1)$, where $r_{\max}(\gamma)$ and $r_{\min}(\gamma)$ are the maximum and minimum range of the input data (in the case of pointclouds, their range, for example) in axis γ. For each cell in the set of cells K belonging to all anchor proposals found in sample $\mathbb{S}$, the grid matrix is updated. These cells are assigned the object probability, i.e., the maximum found likelihood in the categorical distribution of the anchor's classification as predicted by the region proposal layer.

Algorithm 1 Algorithm for Existence Map generation.

$G \leftarrow \mathbf{0}_{(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, 1)}$
for $(f(x=a|\hat{\pi}), k) \in K$ **do**
  $G_k \leftarrow \max_{A_k} \{\max_C (f(x=a|\hat{\pi}))\}$
end for

The resulting grid contains object probabilities in areas where objects may be found and zero likelihood in the rest of the space. It is thus a discrete representation of the probability of an object existing in each cell of $G(\tau)$.

4.4 Uncertainty Modelling

As mentioned in the previous Section, the distribution of anchor proposals is taken into consideration to generate Existence Maps. In this section, the modelling of uncertainty in the context of this anchor distribution is defined.

Section 3.2.1 discusses the division of the uncertainty of a model in two parcels, the aleatoric, concerning how well X conveys Y considering the true underlying distribution of the data, and the epistemic, reflecting the variability of knowledge encoded in a deep learning architecture.

In this sense, Subsection 4.4.1 reflects upon the aleatoric parcel, demonstrating two possible interpretations of this type of uncertainty. Finally, Subsection 4.4.2 discusses a model of the epistemic uncertainty.

In all equations, the base of the logarithm is omitted as it is a parameter of the entropy calculation. The base simply reflects the units of the entropy's result. For example, when the base is

2, the unit of the entropy is bits, namely, the number of bits needed to convey the outcome of the event.

4.4.1 Interpreting Aleatoric Uncertainty

In Malinin (2019), the aleatoric uncertainty at a given data point x is measured by the entropy of the conditional true distribution of y given x, as reported in Equation 4.5.

$$\mathrm{H}[P_{true}(y|x)] = -\sum_{c=1}^{C} P_{true}(y = t_c|x) \log P_{true}(y = t_c|x) \tag{4.5}$$

The aleatoric uncertainty of the overall model thus corresponds to the expected value of this entropy over the true distribution of input samples x:

$$AU_{model} = \mathbb{E}_{p_{true}(X)}[\mathrm{H}[P_{true}(Y|X)]] \tag{4.6}$$

The expected value of the entropy reflects the overall amount of information each x conveys about Y. This measure thus corresponds to the aleatoric uncertainty as it demonstrates the connection between both variables. In a perfect modelling of the data, each x conveys the corresponding y entirely, leading to inexistent aleatoric uncertainty and zeroing the entropy.

These notions represent the aleatoric uncertainty at each point and over the entire model in classification tasks. For regression settings, the summation in Equation 4.5 is transformed into an integral.

Consider now the concept of object existence, as introduced in the previous section and formally defined in Section 2.5.3. In this context, the existence of an object may be communicated through a random variable, E. Since existence concerns objects, this variable must operate under portions of each input sample x. In forthcoming definitions, the area of interest to quantify existence in is the patch Π and at times a cell k on the grid $G(\tau)$.

A patch is a selection of an area of interest in x which signifies that the collection of all patches defined over dataset $\mathscr{D}$ is a subset of X, namely $\{\Pi_1, \Pi_2, ..., \Pi_s\} \subseteq X$.

Taking the aforementioned definitions of equations 4.5 and 4.6, the aleatoric uncertainty of random variable E given a patch Π over the set of all defined patches in the dataset can then be obtained through the following relation:

$$\hat{AU}_{model} = \mathbb{E}_{P(\Pi)}[\mathrm{H}[P(\mathrm{E}|\Pi)]] \tag{4.7}$$

The random variable E is regressed from the distribution of anchors described in Section 4.2. In the classification task of object detection, $P(Y|X)$ is a categorical distribution over the contemplated classes C and the random variable Y takes the value of the class with maximum confidence.

In order to model the distribution that describes random variable E, two different approaches are presented. The following subsections discuss these interpretations of uncertainty in detail.

4.4.1.1 Direct Entropy Estimation

The first approach reported to model the aleatoric uncertainty in a patch consists in directly utilising each anchor proposal as a sample of the existence random variable.

In this interpretation, the probability of a single anchor, p_a, is assumed to be the maximum likelihood found in its categorical distribution predicted when classifying the anchor's contents, as shown in Equation 4.4. The aleatoric uncertainty can thus be approximated as the entropy of each anchor's probability over the space of proposals in the patch, A_Π, as demonstrated through Equation 4.8.

$$\mathrm{H}[P(\mathrm{E}|\Pi)] = -\sum_{a=1}^{|A_\Pi|} p_a \log p_a \tag{4.8}$$

This calculation is direct and simple, being a natural modelling of the several proposals that are found in an area of the space of the input sample. Nevertheless, this approach is the one with the least statistical integrity compared to the interpretations that are discussed in the next subsections. This stems from the fact that the summation of anchor probabilities in a patch is not guaranteed to amount to one as each anchor is independent and their predicted likelihoods are entirely separated. In fact, the most probable scenario is that anchor proposals have similar confidences, leading to closely related probabilities.

The intuition behind this interpretation is to take advantage of both the density of proposals in a patch and their likelihoods. Generally, the more proposals are found within a patch, the smaller the entropy will be due to the negative summation over A_Π. This expresses higher consensus of the model on the existence of an object in the patch, so a smaller estimate of uncertainty is coherent with the model's belief.

Furthermore, the parcel $-p_a \log p_a$ reaches maximum value at $\frac{1}{e}(\approx 0.37)$ and decreases when moving away from this peak, as shown in Figure 4.4. This reduction is less steep as the probability of the anchor approximates one. This characteristic, combined with the summation, leads the entropy to reach maximum values when anchor probabilities surround $\frac{1}{C}$, which is the point of maximum uncertainty as the model is unable to easily classify the contents of anchors into a specific class.

These two properties of the direct entropy estimation enable it to have usefulness in practical settings. Some possible applications of this interpretation in the scope of this thesis are presented in further sections.

Finally, since the statistical guarantee of the distribution's measure can not be maintained, the entropy is not normalised through standard methods. A more detailed discussion on this topic is featured in forthcoming sections as well.

All subsequent sections and chapters of this thesis mention this interpretation of uncertainty as *direct entropy estimation*.

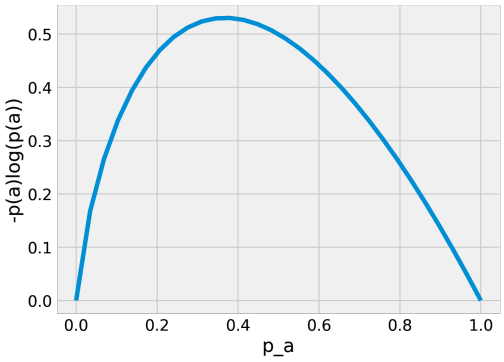


Figure 4.4: Variation of the parcel $-p_a \log p_a$ in function of p_a. The maximum surrounds $\frac{1}{e}$ with the value of the parcel monotonously decreasing both on the left and right of this point. The shown plot assumes a base 2 for the logarithm, which signifies the entropy's result is in shannons.

4.4.1.2 Joint Entropy Estimation

Another approach to the aleatoric uncertainty estimation using the distribution of anchors in a patch consists in utilising the summation property of the entropy described in Section 2.3.1. This property states that the entropy of simultaneous events corresponds to the sum of the entropy of each event when they are independent, namely:

$$\mathrm{H}(S,T) = \mathrm{H}(S) + \mathrm{H}(T) \tag{4.9}$$

The distribution of existence E is a realisation of the anchor distribution in a patch, as seen in Equation 4.10.

$$\begin{aligned} P(\mathrm{E}=1|\Pi) &= P(A \geq 0.5|\Pi) \\ P(\mathrm{E}=0|\Pi) &= P(A < 0.5|\Pi) \end{aligned} \tag{4.10}$$

As such, $P(\mathrm{E}|\Pi)$ is another notation for the joint distribution of the anchors found in a given patch, as follows:

$$P(\mathrm{E}|\Pi) = P(A|\Pi) = P(a_1, a_2, ..., a_{|A_\Pi|}|\Pi) \tag{4.11}$$

Recovering the aforementioned additivity property of the entropy, and denoting the fact that anchor proposals are both simultaneous and independent events, the entropy of the existence distribution given a patch Π can be estimated through the sum of the entropies of each anchor's

categorical distribution, as reported in Equation 4.12.

$$\begin{aligned} \mathrm{H}[P(\mathrm{E}|\Pi)] &= \mathrm{H}[P(a_1, a_2, ..., a_{|A_\Pi|}|\Pi)] \\ &= \sum_{i=1}^{|A_\Pi|} \mathrm{H}[p(a_i|\Pi)] \\ &= -\sum_{i=1}^{|A_\Pi|} \sum_{c=1}^{C} f(a_i = t_c|\hat{\pi}) \log\{f(a_i = t_c|\hat{\pi})\} \end{aligned} \tag{4.12}$$

This interpretation of the entropy $\mathrm{H}[P(\mathrm{E}|\Pi)]$ explores the independent nature of anchor proposals, which, despite being overlaid on the same input sample x, have no relation with each other. This method also considers the categorical distribution of anchor classification, an aspect not modelled in the *direct entropy estimation*. In further sections and chapters, this interpretation of uncertainty is named *joint entropy estimation*.

4.4.2 Interpreting Epistemic Uncertainty

The epistemic uncertainty of a model correlates with its lack of knowledge demonstrated by the disagreement between different predictors generated by the same architecture. In this sense, the anchor distribution, which differs from the model's standard output, can be a natural reflection of this lack of agreement. The degree of divergence can be measured by an appropriate distance function, thus leading to an estimate of the epistemic uncertainty.

Distilling these key ideas, the disparity between the anchors distributions' outcomes and the standard output is assumed to represent the variability encoded in a model's parameters, especially demonstrating the randomness generated through the feedforward and backpropagation optimisation mechanisms during training. This randomness reflects in the divergence in the generated weight distributions.

As such, in order to estimate the epistemic parcel, the distance between anchor distributions and detection outputs must be quantified. For this, the Kullback-Leibler Divergence is considered. The KL Divergence between P and Q quantifies the expected excess surprise of using Q as a model when P represents the true underlying distribution. In this sense, the standard output is assumed to be the true distribution modelled by the object detection architecture and the anchor distribution to represent the approximation model. The distance between these components can then be estimated through Equation 4.13, namely the KL Divergence between $P(Y|\Pi,\theta)$ and $P(A|\Pi,\theta)$.

$$D_{KL}(P(Y|\Pi,\theta)||P(A|\Pi,\theta)) = \sum_{\Pi} p(y|\Pi,\theta) \log \frac{p(y|\Pi,\theta)}{p(a|\Pi,\theta)} \tag{4.13}$$

A challenging situation arises in this methodology when an anchor distribution does not lead to a standard output bounding box or when there is no corresponding anchor distribution for an outputted detection. In this case, probabilities may amount to 0 leading the divergence metric to become $+\infty$. Contrary to this, in some instances, probabilities can be equal to 1, resulting in a $+\infty$ divergence as well.

Considering these factors, and in order to ensure a practical usable value of the epistemic uncertainty, probabilities need to be transformed to the range $]0,1[$. For this reason, a probability threshold is assumed for remaining detections, ρ_{out}, and proposals, ρ_{anchor}, with no match. Further, if a probability is equal to 1, its value is reduced to the closest float that is lesser than 1.

Moreover, another important factor to denote is that the KL Divergence metric is not symmetric which signifies that measuring the distance between Y and A is different from observing the divergence of A to Y. As such, this proposal considers the outcomes of the standard model, Y, to be the *original model*, namely P, that is approximated by another distribution Q, or, in this case, A. Nevertheless, in patches where no standard bounding box exists even though the Existence Map features an anchor distribution, the model that is assumed to be *original* is the one that made a proposal. This decision is taken in order to ensure that Existence Map suggestions do not decrease the overall epistemic uncertainty, as would happen with the inclusion of negative parcels. This is supported by the idea that having one sided bounding box proposals, either from the standard output or the region proposal layer, should express similar uncertainty. In this sense, the distance metric is updated accordingly, resulting in the expression communicated through Equation 4.14.

$$d(P(y|\Pi,\theta)||P(a|\Pi,\theta)) = \begin{cases} p(y|\Pi,\theta)\log\dfrac{p(y|\Pi,\theta)}{p(a|\Pi,\theta)}, \text{ if } bbox_y \cap bbox_a \text{ with } IoU > 0 \\ p(y|\Pi,\theta)\log\dfrac{p(y|\Pi,\theta)}{\rho_{anchor}}, \text{ if } a = 0 \\ p(a|\Pi,\theta)\log\dfrac{p(a|\Pi,\theta)}{\rho_{out}}, \text{ if } y = 0 \end{cases} \tag{4.14}$$

Similarly to the process followed by Existence Map generation, $p(a|\Pi,\theta)$ is assumed to be the maximum probability found in all anchors that intersected patch Π with the value of each proposal assumed as the maximum of the categorical distribution. Equation 4.15 reports this relation as a refresher.

$$p(a|\Pi,\theta) = \max_{A_\Pi}\{\max_{C}(f(x=a|\hat{\pi}))\} \tag{4.15}$$

The distance is estimated over all realisations of the random variables of interest. Since all samples of Y and A occur in given spaces of the input, namely patches, the summation is carried out over patches. Patches are the components of a dataset, $\mathscr{D}$. Considering all patches generated for a single dataset, and thus reflecting this in the aforementioned formulae, leads to the estimation of the model's epistemic uncertainty. This is expressed by the expected value of the distance metric over the distribution of patches, $P(\Pi)$, culminating in the following relation:

$$\hat{EU}_{model} = \mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(Y|\Pi,\theta)||P(A|\Pi,\theta))] \tag{4.16}$$

Considering that all patches feature an intersection of a standard bounding box with an anchor proposal, the epistemic model can be developed as follows, where N stands for the number of

patches with intersection:

$$\begin{aligned}
\hat{EU}_{model} &= \mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(Y|\Pi,\theta)||P(A|\Pi,\theta))] \\
&= \mathbb{E}_{P(\Pi)}[\sum_{\Pi} p(y|\Pi,\theta)\log p(y|\Pi,\theta) - p(y|\Pi,\theta)\log p(a|\Pi,\theta)] \\
&= -\mathbb{E}_{P(\Pi)}[-\sum_{\Pi} p(y|\Pi,\theta)\log p(y|\Pi,\theta)] + \mathbb{E}_{P(\Pi)}[-\sum_{\Pi} p(y|\Pi,\theta)\log p(a|\Pi,\theta)] \\
&= -\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(y|\Pi,\theta))] + \mathbb{E}_{P(\Pi)}[-\sum_{\Pi} p(y|\Pi,\theta)\log p(a|\Pi,\theta)\frac{p(a|\Pi,\theta)}{p(a|\Pi,\theta)}] \\
&= -\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(y|\Pi,\theta))] + \mathbb{E}_{P(\Pi)}[\sum_{\Pi} \frac{p(y|\Pi,\theta)}{p(a|\Pi,\theta)}]\mathbb{E}_{P(\Pi)}[-\sum_{\Pi} p(a|\Pi,\theta)\log p(a|\Pi,\theta)] \\
&= -\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(y|\Pi,\theta))] + \frac{1}{N}\sum_{\Pi} \frac{p(y|\Pi,\theta)}{p(a|\Pi,\theta)}\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(a|\Pi,\theta))] \\
&= \frac{1}{N}\sum_{\Pi} \frac{p(y|\Pi,\theta)}{p(a|\Pi,\theta)}\hat{AU}_{rp} - \hat{AU}_{out}
\end{aligned} \tag{4.17}$$

As such, it is determined that the formulated epistemic uncertainty quantifies the difference in the aleatoric uncertainty estimates of the standard output versus the region proposal layer as a factor of the average ratio between the predicted probabilities. This demonstrates that the proposed epistemic calculation represents the divergence produced by the different sets of parameters in the region proposal and output layer, thus expressing the disagreement within the network.

In the case of no support from either variable, the expression of the epistemic uncertainty reduces to Equation 4.18 when there is no Existence Map prediction, $a = 0$, and to Equation 4.19 if no standard output bounding box exists in the patch, $y = 0$.

$$\begin{aligned}
\hat{EU}_{model} &= \mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(Y|\Pi,\theta)||\rho_{anchor})] \\
&= \frac{1}{N\rho_{anchor}}\sum_{\Pi} p(y|\Pi,\theta)\underbrace{\mathbb{E}_{P(\Pi)}[\mathrm{H}(\rho_{anchor})]}_{\approx \hat{AU}_{rp} \text{ when } a=0} - \underbrace{\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(y|\Pi,\theta))]}_{\hat{AU}_{out}}
\end{aligned} \tag{4.18}$$

$$\begin{aligned}
\hat{EU}_{model} &= \mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(a|\Pi,\theta)||\rho_{det})] \\
&= \frac{1}{N\rho_{out}}\sum_{\Pi} p(a|\Pi,\theta)\underbrace{\mathbb{E}_{P(\Pi)}[\mathrm{H}(\rho_{out})]}_{\approx \hat{AU}_{out} \text{ when } y=0} - \underbrace{\mathbb{E}_{P(\Pi)}[\mathrm{H}(p(a|\Pi,\theta))]}_{\hat{AU}_{rp}}
\end{aligned} \tag{4.19}$$

As observed in Equation 4.18, the epistemic uncertainty also reduces to the difference in the aleatoric uncertainties as estimated by the output layer and as approximated for the region proposal layer in the case of a standard bounding box with no Existence Map support. Since ρ_{anchor} is constant over all patches in these conditions, the ratio becomes the average predicted confidence by the final layer normalised by the assumed anchor probability.

On the opposite situation, the anchor distribution assumes the role of the main model. As such, the quantified divergence is between the aleatoric uncertainties estimated by the region proposal layer and that approximated for the output layer. The ratio reflects the same behaviour as previously reported, only substituting the ρ_{anchor} by the constant ρ_{out} and the considered average probability is estimated by the anchor distribution. This procedure allows for a less conservative estimate of the epistemic uncertainty.

4.5 Uncertainty Maps

The generation of existence maps has been presented previously, introducing the concept of utilising the concentration of anchor proposals. On the other hand, the previous section explored possible models of uncertainty, considering both the aleatoric and epistemic parcels. In this sense, combining both of the aforementioned ideas, a possible practical application can be devised - the Uncertainty Map.

Recovering the concept of a grid $G(\tau)$ layed over an input sample x, the uncertainty expressed by the distribution of anchors can be quantified over each cell k. The distribution of anchors in each cell is defined in Section 4.2, particularly denoted in Figure 4.1. With the availability of an uncertainty estimate in each cell, a map of the input sample may be generated. Uncertainty Maps are similar in shape to Existence Maps, however, each cell contains the uncertainty of the anchor distribution instead of the probability of existence.

The following subsections explore the generation of Uncertainty Maps using the two methods discussed in Subsection 4.4.1, the *direct modelling estimation* and the *joint entropy estimation*. This signifies that Uncertainty Maps express the aleatoric parcel over each cell of the grid $G(\tau)$.

4.5.1 Using the Direct Modelling Estimation

In this subsection, the *direct entropy estimation* model for the quantification of aleatoric uncertainties is explored.

Deriving the aforementioned model for each cell k instead of a patch results in Equation 4.20. From the formulation of U_k, it can be seen that each cell in the Uncertainty Map contains the expected value of the negative logarithm of the distribution of anchor proposals, as defined in Equation 4.3. The assumption of p_a taking the value of the maximum class likelihood is maintained, as reported in the original estimation model.

$$\begin{aligned} U_k &= -\sum_{\mathrm{A}_k} p_a \log p_a \\ &= \mathbb{E}[-\log P(\mathrm{A}_k)] = \mathrm{H}_{P(\mathrm{A}_k)} \end{aligned} \tag{4.20}$$

Similarly to Existence Maps, Uncertainty Maps follow a likewise simple generation procedure, as described in Algorithm 2. The Uncertainty Map utilises a temporary existence grid which contains the maximum classification likelihood, $\max_C \{f(x=a|\hat{\pi})\}$, in each cell of the input space of all anchors that intersected it.

Algorithm 2 Algorithm for Uncertainty Map generation using the *direct entropy estimation*.

$G \leftarrow \mathbf{0}_{(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|)}$
for $k \in K$ **do**
  for $j \in [1, |\mathrm{A}_k|]$ **do**
    $G_{kj} \leftarrow \max \mathbf{p}_{\mathrm{A}_j}$
  end for
end for
$U \leftarrow \mathbf{0}_{(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, 1)}$
for $k \in K$ **do**
  $U_k \leftarrow -\sum_{\mathrm{A}_k} \mathbf{G_k} \log \mathbf{G_k}$
end for

This transitory existence grid includes all the likelihoods over all anchors that intersect cell k, thus resulting in the shape $(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|)$. The aleatoric uncertainty for each cell is then calculated through Equation 4.20, as previously explained. The summation is over all the proposals in the cell's anchor distribution.

Uncertainty Maps have a prominent visual component. Its information can be clearly communicated over the original input sample, giving another level of insight into the inner workings of a model. As such, it is of interest to be able to display colorised regions of uncertainty.

In this sense, to create a sequential colour map to enhance the interpretability and readability of this display format, it is necessary to regularise the estimated uncertainties. Since the modelling of the uncertainty is based on the *direct entropy estimation*, probabilities on the manifold of the random variable are not guaranteed to sum to one. For this reason, normalisation can not be carried out using the standard maximum entropy formulation. In this sense, and for detailed experimentation purposes, three regularisation methods are proposed, as described in Equation 4.21 (1), 4.22 (2) and 4.23 (3).

$$\begin{aligned} \tilde{U}_k &= \frac{\sum_{|\mathrm{A}_k|} p_a \log p_a}{\sum_{\max_K\{|\mathrm{A}_k|\}} \frac{1}{2} \log \frac{1}{2}} \\ &= -\frac{2}{\max_K\{|\mathrm{A}_k|\}} \log \frac{1}{2} \sum_{|\mathrm{A}_k|} p_a \log p_a \end{aligned} \tag{4.21}$$

Since each proposal is independent, the maximum uncertain outcome is when the probability of each equals 0.5, thus resulting in $\max\{\mathrm{H_A}\} = -\sum_{|\mathrm{A}|} \frac{1}{2} \log \frac{1}{2}$. If all anchors feature this probability, the model is entirely uncertain on the existence of an object in the cell.

$$\begin{aligned} \tilde{U}_k &= \frac{\sum_{|\mathrm{A}_k|} p_a \log p_a}{\sum_{|\mathrm{A}_k|} \frac{1}{2} \log \frac{1}{2}} \\ &= -\frac{2}{|\mathrm{A}_k|} \log \frac{1}{2} \sum_{|\mathrm{A}_k|} p_a \log p_a \end{aligned} \tag{4.22}$$

The first and second methods as seen in Equation 4.21 and 4.22 entail the normalisation by the maximum entropy possible for a cell's distribution. The differing aspect between both is that in

method (1), the entropy summation is considered over the maximum number of proposals found in all anchor distributions over the entire grid. On the other hand, on approach (2), the number of entropy parcels admitted for the maximum uncertainty calculation is the number of proposals in each cell's distribution. The other methodology, (3), simply regularises the entropy by the number of proposals intersecting the cell.

$$\tilde{U}_k = -\frac{1}{|\mathrm{A}_k|} \sum_{|\mathrm{A}_k|} p_a \log p_a \tag{4.23}$$

4.5.2 Using the Joint Entropy Estimation

Considering now the *joint entropy estimation* model, in order to calculate aleatoric uncertainties in each cell, a coherent categorical distribution that sums to one must be obtained. Contrary to the method utilised in the maximum likelihood approach, in order to ensure this property, the softmax function must be used instead of the sigmoid.

Logits in anchor classification are often times very large in one of the classes, which leads the softmax function to attribute a very high probability with very low values on the others. This expresses low uncertainty and many times stems from the overconfidence of these networks [Nguyen et al. (2015)]. As such, the softmax function must be adjusted with a temperature, often referred to as $\beta = \frac{1}{T}$, as shown in Equation 4.24. The choice of the β parameter influences the resulting Uncertainty Map as it adds or removes entropy in each anchor's categorical distribution.

$$\tilde{\sigma}(z_i) = \frac{e^{\beta z_i}}{\sum_{i=1}^{n} e^{\beta z_i}} \tag{4.24}$$

The anchor's categorical distribution is passed through the softmax, ensuring its probabilities sum to one, as depicted in Equation 4.25.

$$\begin{gathered} p_a = f(x = a|\hat{\pi}) = Cat(a;\hat{\pi}) \\ with\ \hat{\pi} = \tilde{\sigma}(f_{rp}(c_x, c_y, c_z, d_x, d_y, d_z, \alpha)), \sum_{c=1}^{C} \hat{\pi}_c = 1, \hat{\pi}_c \geq 0 \end{gathered} \tag{4.25}$$

With a full categorical distribution, the uncertainty in each cell k can then be calculated, following the *joint entropy estimation* method. Equation 4.26 demonstrates the quantification of the aleatoric uncertainty for a cell k using this interpretation.

$$\begin{aligned} U_k &= -\sum_{i=1}^{|\mathrm{A}_k|} \sum_{c=1}^{C} f(a_i = t_c|\hat{\pi}) \log f(a_i = t_c|\hat{\pi}) \\ &= \sum_{i=1}^{|\mathrm{A}_k|} \mathrm{H}[p(a_i|\mathrm{A}_k)] = \mathrm{H}[P(a_1, a_2, ..., a_{|\mathrm{A}_k|}|\mathrm{A}_k)] \end{aligned} \tag{4.26}$$

In this case, the generation of Uncertainty Maps requires the saving of more information on the temporary existence grid. Since each categorical distribution must be kept in the transitory grid,

its shape becomes $(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|, C)$. Otherwise, the generation process follows the same standard as the *direct entropy estimation*, as shown in Algorithm 3.

Algorithm 3 Algorithm for Uncertainty Map generation using the *joint entropy estimation*.

$G \leftarrow \mathbf{0}_{(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|, C)}$
for $k \in K$ **do**
  for $j \in [1, |\mathrm{A}_k|]$ **do**
    $G_{kj} \leftarrow \mathbf{p}_{\mathrm{A}_j}$ ▷ $\mathbf{p}_{\mathrm{A}_j}$ has shape (C)
  end for
end for
$U \leftarrow \mathbf{0}_{(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, 1)}$
for $k \in K$ **do**
  $U_k \leftarrow -\sum_{i=1}^{\mathrm{A}_k} \sum_{c=1}^{C} \mathbf{G_{k,i,c}} \log \mathbf{G_{k,i,c}}$
end for

Concerning the generation of a sequential colorised map for display purposes, the normalisation of the Uncertainty Map in the *joint entropy estimation* context utilises the maximum entropy of the categorical distribution, $\log C$. Equation 4.27 demonstrates the normalisation, which entails the division of U_k by the maximum entropy multiplied by the number of proposals as the final obtained entropy is a summation of entropies.

$$\tilde{U}_k = -\frac{1}{|\mathrm{A}_k| \log C} \sum_{i=1}^{|\mathrm{A}_k|} \sum_{c=1}^{C} f(a_i = t_c | \hat{\boldsymbol{\pi}}) \log f(a_i = t_c | \hat{\boldsymbol{\pi}}) \tag{4.27}$$

4.6 Existence Probability

The main goal of Existence Maps is to provide more information on a network's knowledge and improve its expressiveness beyond the considered closed set of classes. Since the world is an open-set, this characteristic may be of interest for deployment of deep learning-based object detectors in real-world conditions [Balasubramanian et al. (2021)]. However, Existence Maps can induce false positives when evaluating standard classification. This is not necessarily a flaw, but rather an intrinsic characteristic of their conceptualisation that stems from their suggestions of areas of interest being taken directly from the region proposal module without going through further fine-grained filtering contrary to the standard output.

Even though this characteristic is inherent to the method itself, its effects can be mitigated, so as to allow further distillation of the final output of the object detection model by merging standard detections with Existence Map suggested areas. In this sense, in this section, a method to extract more accurate bounding boxes from Existence Maps whilst also leveraging the standard output of the model is proposed. Merging candidates are found by calculating the Intersection over Union (IoU) between the standard output bounding box and the Existence Map patch. Since the information conveyed by Existence Maps is not fine-grained, the patch area may diverge considerably from the real location of the object. As such, all pairs with $IoU > 0$ are considered as merge candidates.

Further than this, since the final output considers the merging of both proposals, detection confidence is another aspect that can be further distilled by extracting more fine-grained bounding boxes. As such, in this section, a methodology to calibrate the outputted object probabilities is also proposed. This approach takes into account not only the object probability but also the density and entropy of proposals in each patch generated by anchor distributions. By merging these outcomes with the object probabilities, more fine-grained detection proposals can be obtained. These processed boxes may then be merged with the model's output, leading to the final detections.

This calibration method is referred to as the *calibrated existence probability* and its value can be deduced for a patch, as described in Section 4.2, showcasing the confidence of the model on the existence of an object in the defined area.

4.6.1 Modelling Patch Density

The first factor observed when merging a bounding box with a patch is the density of proposals in the anchor distribution, as shown in Equation 4.28. This density represents the number of anchor proposals found within the patch, $|\mathrm{A}_\Pi|$, divided by the maximum possible number of proposals, $|\Sigma_\Pi|$. The subset A_Π is defined by the anchors that intersected to form the patch and is contained in Σ_Π, namely $\mathrm{A}_\Pi \subseteq \Sigma_\Pi$. As such, A_Π is the set of all anchors that could be proposed within patch Π, taking into consideration the downsampling of the network, the number of classes and anchor rotations considered in the model, as seen in the second relation in Equation 4.28.

$$\begin{aligned} \rho_\Pi &= \frac{|\mathrm{A}_\Pi|}{|\Sigma_\Pi|} \\ |\Sigma_\Pi| &= 2C\frac{d_x^\Pi d_y^\Pi \dim(f_{rp}(x))\dim(f_{rp}(y))}{r_{\max}(x) r_{\max}(y)} \end{aligned} \tag{4.28}$$

where d_x^Π and d_y^Π are the dimensions of the patch in 2D, $\dim(f_{rp}(x))$ and $\dim(f_{rp}(y))$ are the output dimensions of the region proposal layer in each respective axis and r_x and r_y represent the maximum range of the pointcloud.

For ease of understanding, Figure 4.5 demonstrates the difference between the maximum number of proposals, $|\Sigma_\Pi|$, and the anchor distribution of a patch, A_Π, depending on the region proposal feature map downsampling. In reality, all possible proposals are made for each data point. However, only the top n anchors are considered for the generation of anchor distributions. This may lead the resulting anchor distributions to have less samples compared to the number of maximum proposals made over the span of patch's area. This variability is quantified by the density of the patch, ρ_Π, as described above.

4.6.2 Modelling Patch Entropy

Another characteristic taken into account when merging and calibrating the confidences of the standard output detections and the Existence Map suggestions is the entropy in each patch. This

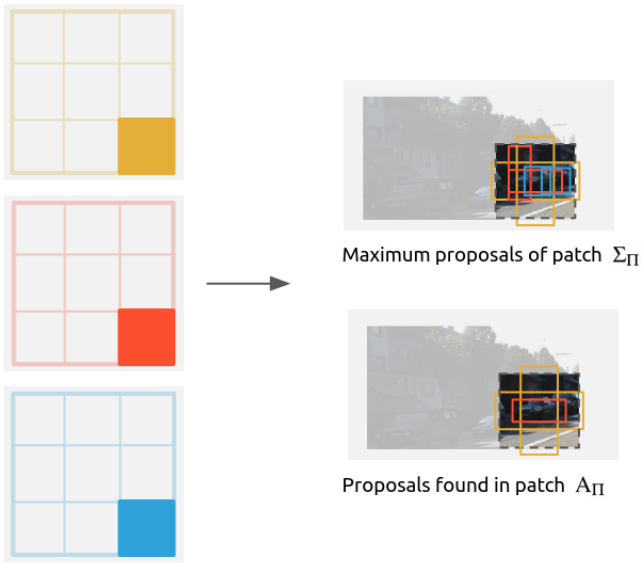


Figure 4.5: Representation of a patch, Π with the maximum number of proposals, $|\Sigma_\Pi|$, shown on the upper diagram and a possible selection of anchors that belonged to the top n proposals on the bottom. In this example, each feature map block overlays 2 anchors with different headings per class.

entropy aligns with the *direct modelling estimation* method, providing a coarse estimate of the aleatoric uncertainty in the patch.

This stems from the idea that each anchor proposal in a patch can be seen as a sample of the object distribution and its uncertainty may be interpreted as the entropy of the sampled outcomes. Equation 4.29 demonstrates how the entropy, H_p, is measured, which takes into consideration all the proposed boxes in the patch.

$$H_\Pi = -\sum_{|A_\Pi|} p_a \log p_a \tag{4.29}$$

The entropy is selected since it allows for the accounting of both the number of proposals in each patch and the respectively attributed confidence values. If there are many proposals within an area with either high or low confidences, the entropy will decrease as the model is more certain on the existence / non-existence of an object in the patch. The lowest entropy will be when the model is absolutely certain that there is an object (confidence close to one) or that nothing exists in the patch (confidence close to zero). In these cases, the entropy should not have a large effect on the merging process and, consequently, on the calibration of the existence probability. On the other hand, when a proposal has a confidence of 0.5, the model is rather uncertain on whether there is an object in the patch and uncertainty rises to maximum levels, thus having a higher weight in the final probability calibration. By introducing the entropy in the merging process, an intuition of the model's uncertainty on the outputted detections is also considered in the calculation, naturally leading to a better calibrated probability.

The entropy is normalised by dividing the value obtained in the patch by the maximum possible entropy. The maximum entropy is achieved when all the proposals in a patch have a 0.5 probability, as shown in Equation 4.30. This notion is mentioned previously in the context of the *direct entropy estimation* model, where the maximum entropy is also needed for normalisation purposes 4.5.1.

$$\begin{aligned} \max_{\Pi}(\mathrm{H}) &= -\sum_{|\mathrm{A}_\Pi|}\frac{1}{2}\log\frac{1}{2} \\ &= -\frac{|\mathrm{A}_\Pi|}{2}\log\frac{1}{2} \end{aligned} \tag{4.30}$$

As previously denoted in Equation 4.4, for merging purposes, p_a is approximated as the maximum probability found in the categorical vector of each anchor's classification output.

4.6.3 Calibrated Existence Probability

Leveraging the previously exposed notions, standard output bounding boxes can be merged with patches from existence maps. The final calibrated Existence Probability that is assigned to the merging target can be obtained by the relation demonstrated in Equation 4.31. This calculation linearly accounts for the maximum object probability, o_Π, as described in Section 4.3, and the density in the patch, ρ_Π, as well as penalising for high patch entropies, H_Π by a parametrisable factor, γ.

$$\mathrm{e}_\Pi = \begin{cases} \mathrm{o}_\Pi\rho_\Pi - 2\gamma(1-\frac{\mathrm{H}_\Pi}{\max_\Pi(\mathrm{H})})\mathrm{o}_\Pi\rho_\Pi & \text{if } \frac{\mathrm{H}_\Pi}{\max_\Pi(\mathrm{H})} \geq 0.5 \\ \mathrm{o}_\Pi\rho_\Pi & \text{else.} \end{cases} \tag{4.31}$$

The penalisation to the final probability is only applied if the metric entropy is over 0.5. This is due to the aforementioned characteristic of the entropy that features lower values when there is high confidence in either the existence or non-existence of objects and high quantities when the probability surrounds 0.5 that represents maximum uncertainty in the object's existence. Since the entropy is higher in these cases, thus representing high uncertainty, the object probability outputted by the model tends to be overconfident, needing a penalisation correspondent to the degree of uncertainty of the model.

Through the Existence Probability, a confidence can be attributed to each merged detection. Concerning the regression of the final distilled bounding box, a similar process to the patch generation methodology previously described in Section 4.2 is followed. Figure 4.6 describes the process that allows the distillation of a final bounding box including both the standard model's and the Existence Map's information.

A temporary bounding box is generated by taking the maximum corners of both boxes. The centre coordinates are then defined as the mid point on the x and y axis for two-dimensional applications. In the case of three-dimensional detections, the same procedure is followed, including the z coordinate as well. The corners of the final bounding box are then defined as the average width and height of both boxes. Once more, for three-dimensional boxes, the average over the z coordinate is also considered.

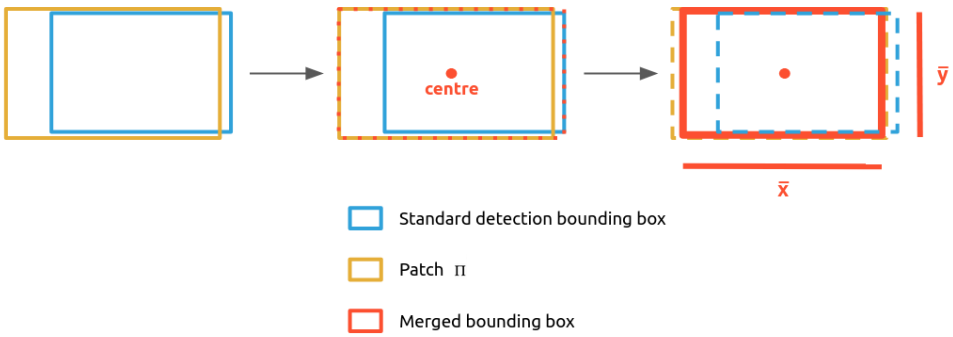


Figure 4.6: Process followed to obtain a distilled bounding box by merging a standard detection with the Existence Map's suggested patch.

4.7 A Global View of Existence and Uncertainty

The previous sections explored interpretations of uncertainty and possible applications where its estimation may provide more information into the inner knowledge of a deep learning model. Further than this, Existence Maps are proposed as a tool to complement the use of the distribution of anchors as an expression of the existence of objects. Concerning these concepts, some global considerations are of interest to be communicated.

Existence and uncertainty maps are generated in parallel to the standard pipeline of an object detector as seen in Figure 4.7. The framework can be added to any stage of a model's development, either during training, testing or deployment. This means that it is also possible to add the Existence Map generation module to a pre-trained network without the need to retrain it. The discussed method utilises knowledge already encoded in the model so it does not have any dependency on the training process which can progress in a standard manner.

The time complexity of the generation of existence and uncertainty maps is linear to the number of anchors considered in the distribution of each sample $\mathbb{S}$, namely n, which translates into a $\theta(n)$ complexity. This complexity stems from the filling in of the grid with the distributions of anchors.

As for spatial complexity considerations, in order to generate an Existence Map, a grid of shape $(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, 1)$ is needed for two and three-dimensional input data. For the generation of an Uncertainty Map, the memory needs differ depending on the interpretation of uncertainty that is selected. The spatial complexity arises from the creation of the transitory grid which can be discarded after the process is finished. For the *direct entropy estimation*-based map, the needed transitory grid has a shape of $(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|)$ whilst for the *joint entropy estimation* maps, this grid must have an additional dimension, culminating in the shape $(\frac{r_{\max}(x)-r_{\min}(x)}{\tau}+1, \frac{r_{\max}(y)-r_{\min}(y)}{\tau}+1, |\mathrm{A}_k|, C)$. Nevertheless, the final Uncertainty Map has the same shape as existence maps.

In the case of three-dimensional datasets, both existence and uncertainty maps may be either generated in two dimensions, as in bird's eye view for example, or in three dimensions, maintaining the original information intact.

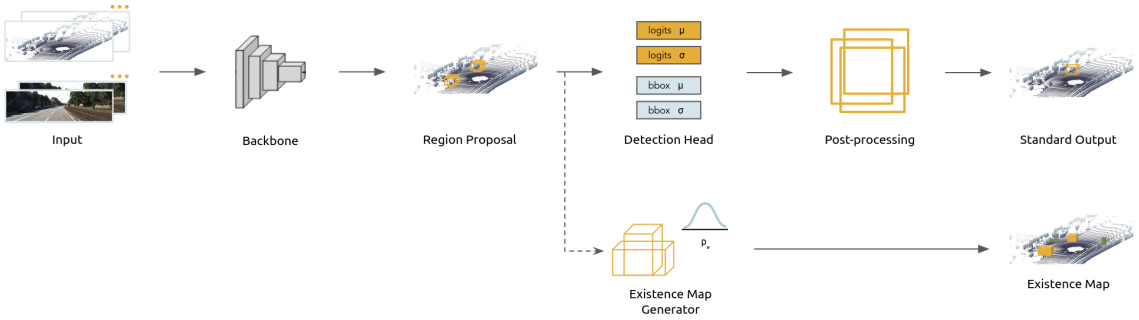


Figure 4.7: A standard anchor-based object detection model. A module for existence map extraction can be added to models without change to the standard pipeline. Extraction can be performed both during training, testing and/or in deployment settings. The framework may also be leveraged with pre-trained models without the need for retraining.

As for the estimation of uncertainties, considering first the aleatoric parcel, the standout operation is matrix multiplication. In the case of the *direct entropy estimation* method, this matrix is in reality a vector with size n since a single confidence value is kept for each anchor. On the contrary, for the *joint entropy estimation*, this matrix is two-dimensional with shape (n, C), leading to more costly computation in comparison with the other approach.

Concerning then the epistemic parcel, its estimation requires more effort since there is the need to match standard output bounding boxes with patches that reflect distributions of anchors. This operation entails the calculation of the IoU between both boxes. With clever implementation in two-dimensional spaces, and since the target for a match is $IoU > 0$, this process can be amortised to constant time. However, for three-dimensional settings, this process becomes inevitably more costly. As such, obtaining the epistemic uncertainty, in comparison with the aleatoric parcel, is much more time consuming. Despite this, this operation is akin to non-maximum suppression, a post-processing step present in most state-of-the-art, real-time compatible, object detectors.

In terms of space, no additional requirements are needed for the calculation of both the aleatoric and epistemic uncertainties when the Existence and Uncertainty Map module is leveraged since anchor distributions have already been collected.

The Existence Probability can be obtained in parallel to the Existence and Uncertainty Map generation procedure, as only the confidences for each anchor have to be obtained in order to enable its estimation. Furthermore, the merging of bounding boxes and patches for each detection can be achieved in constant time considering the match has already been ascertained.

4.8 Existence and Uncertainty in Object Detection

Taking into consideration the proposed concepts, it is of interest to analyse some properties of the interpretations of uncertainty and their discussed applications, including existence. In this sense, this section explores some qualitative results produced by the aforementioned proposals, with a focus on the visual information conveyed in each application of the uncertainty.

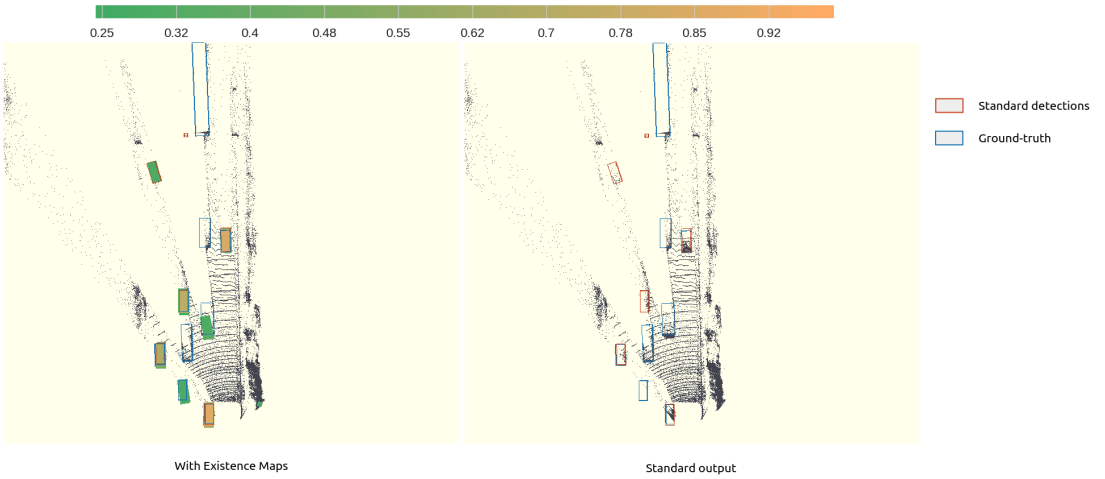


Figure 4.8: Visualisation of the output of an object detection model with Existence Maps (on the left) versus the standard output (on the right) with the ground-truth signalled in blue for both.

All of the reported figures are outputs of the VirConv-L [Wu et al. (2023b)] object detection model on the KITTI dataset [Geiger et al. (2012)] with the proposed existence module employed during testing. Since this section focuses more on the analysis of some properties and visual outcomes of the aforementioned concepts, the setup used to generate these results is not thoroughly outlined as the commented aspects do not depend on a specific model or environment. For an in-depth report of the testing setup, refer to Section 5.2 and 5.3 where the architecture of the object detection model, the splitting of the training and test sets and the relevant implementation details are discussed.

4.8.1 Concerning Existence Maps

Existence Maps allow for a more informative view of the model's knowledge by directly showcasing the state of the region proposal module in the form of a grid of probabilities. Figure 4.8 demonstrates an example of the output of an object detection model with and without existence maps.

As it can be seen, with the Existence Map module, bounding boxes are accompanied by probabilistic region delineations, directly demonstrating areas where the model is more or less confident on its output. These regions are defined by the patches upon which anchor distributions were processed.

Moreover, since Existence Maps are generated with the knowledge contained within the region proposal module, its output differs from the standard detections that are a consequence of the detection head. In Figure 4.8, on the left, it is observable that the Existence Map detected two objects that were missed by the model. The Existence Map also produced a false positive and rejected a false positive of the model. Since Existence Maps do not classify the contents of their signalled areas into a closed set, all objects are in the scope of its detection. This means that Existence Maps are not only able to provide more information on the network's knowledge, but

also enable the suggestion of the existence of objects regardless of the classes the model was trained to detect.

More insight into existence maps is given in forthcoming subsections, including a direct comparison between the information conveyed by both existence and uncertainty maps.

4.8.2 Concerning Uncertainty Maps

Uncertainty Maps, as proposed in Section 4.5, are a direct application of the aleatoric uncertainty interpretations in the context of improving the interpretability of neural networks. These entail the filling of a grid of the input with uncertainty estimated using the anchor distributions. It is relevant to analyse some aspects and key parameters in their generation, namely the normalisation methods discussed in the *direct entropy estimation*-based maps and the β parameter of the softmax function utilised in the *joint entropy estimation* model.

Following this, a comparative analysis between both methods is presented, highlighting the different information conveyed by each uncertainty interpretation.

4.8.2.1 Normalisations

The *direct entropy estimation* interpretation of uncertainty considers several different normalisation and regularisation techniques in order to transform the aleatoric uncertainty to the $[0,1]$ range. As proposed in Subsection 4.5.1, these methodologies are named (1) for the division by the maximum entropy times the maximum size of all cell's anchor distributions, (2) for the normalisation by the maximum entropy multiplied by the number of proposals in each cell and (3) for the regularisation by the number of cell proposals.

The scaling of the uncertainties obtained in each cell substantially impacts the colorisation of the Uncertainty Map, resulting in entirely diverging results as shown in Figure 4.9. The estimates obtained using normalisation technique (1) have support in most of the span $[0,1]$. On the contrary, with methods (2) and (3), the obtained entropies fall frequently on a narrower range, as observed by the colours shown in each. Even so, both techniques lead to significantly diverging quantities as (2) strongly supports $[0.7,1]$ and (3) covers $[0.25,0.55]$ mostly.

Further than the visual impact to the Uncertainty Maps, these regularised estimates communicate diverging aspects of the aleatoric uncertainty. In (3), the calculated entropies concentrate closely around 0.5 which transmits less variability and medium uncertainty over all areas of interest. On the other hand, regularisation (2) sports slightly larger variance than (3) but with results condensing in very high uncertainty levels. The first method, (1), contrary to the other two, normalises by the maximum number of proposals in the sample, leading uncertainties to spread much more and depicting significantly diverging results in between patches.

Regardless of these differences, it is observable that lower Existence Map probabilities (on the top) coincide with higher aleatoric uncertainties, whilst areas where the probability is close to one

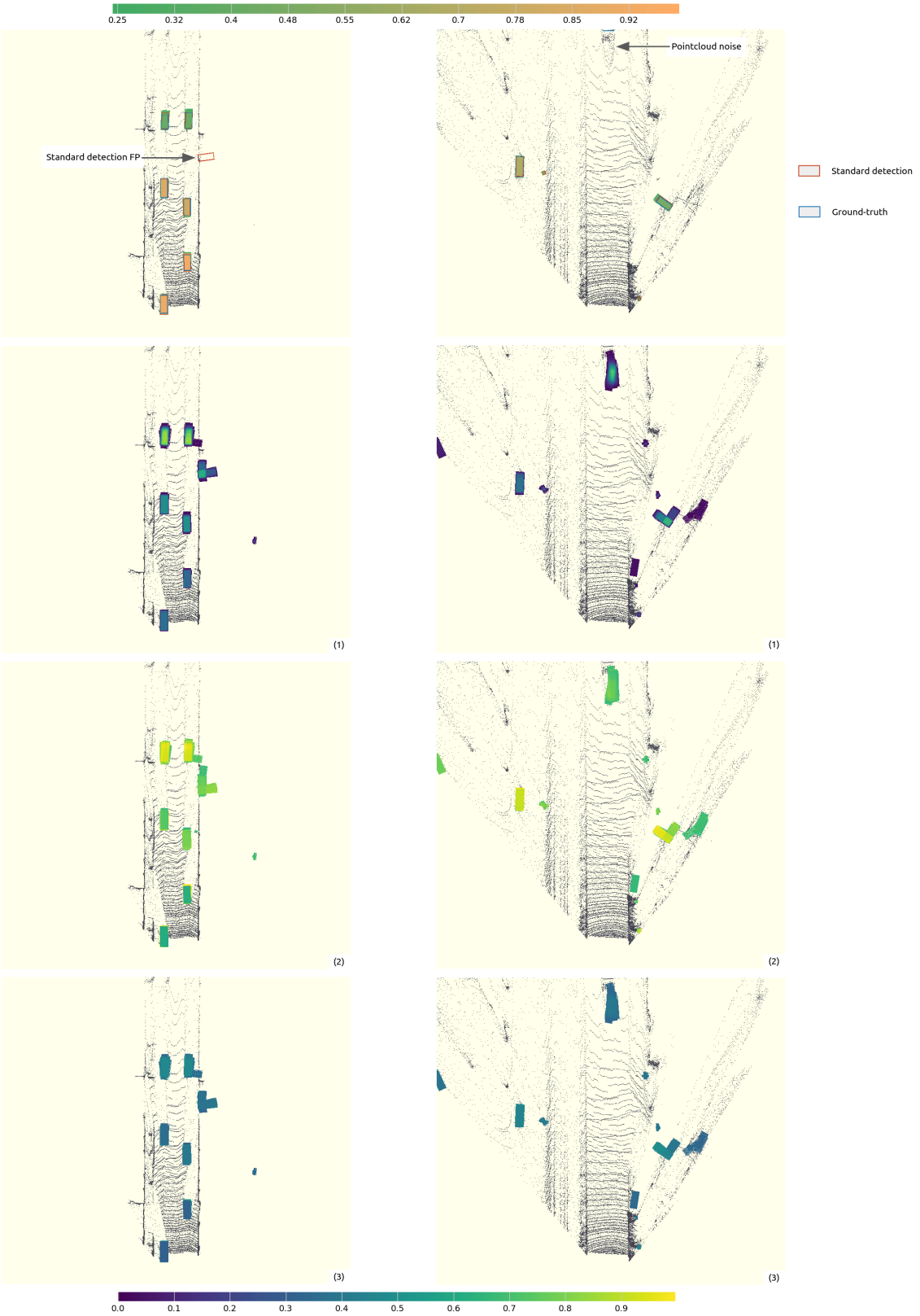


Figure 4.9: Existence maps generated on two test samples followed by the corresponding uncertainty maps using all the regularisation methods described in Subsection 4.5.1, namely (1), (2) and (3).

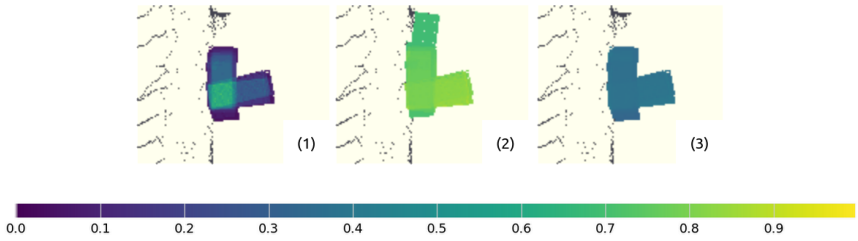


Figure 4.10: Detail of the standard output false positive through different Uncertainty Map regularisation techniques.

display lesser quantities of uncertainty. This pattern emerges throughout all regularisation techniques as expected since all depend on the entropy of the probability distributions, only differing on their scale in comparison to the divisor.

In this sense, and again considering the existence maps on the top of Figure 4.9, the Uncertainty Maps generated through (1) showcase the model's confidence on its detections more coherently. This is due to the fact that high aleatoric uncertainties more distinctly coincide with lower confidences whilst the areas with probabilities close to one, in orange, sport much lower aleatoric uncertainties. The span of this Uncertainty Map more closely resembles the spread of probability which also ranges from close to 0.4 to 1. This result fits expectations as the more an area diverges from the true underlying distribution of the data, the less proficient the model should be at detecting objects in its vicinity. This, in turn, results from the data diverging from the distribution that is being approximated through training, and, in the case of pointclouds, represents noise which prevents a clear definition of the objects in the area.

Focusing now on the left column of examples portrayed in Figure 4.9, the top diagram shows that the standard output of the model makes a false positive with no support from the Existence Map. In that specific area of the input, an anchor distribution exists since there are uncertainty estimates even though this result was filtered through the Existence Map generation process. Nevertheless, the entropy of this distribution is conveyed in significantly different formats in between regularisation techniques, as seen in Figure 4.10.

Aleatoric uncertainty estimates are progressively more smoothed out from technique (1) to (3), with little perceivable shift in the entropy over cells of interest on (3). Moreover, method (1) leads to peak uncertainty in the square that intersects proposals in different headings, with estimates dimming towards zero as distance from this area increases. On the contrary, using regularisation (2), there are still perceivable areas with higher and lower uncertainty even though the information on the intersection of headings is lost.

Another aspect of interest can be observed in the area with high pointcloud noise in the right column of Figure 4.9. As seen in Figure 4.11, methods (1) and (2) have a similar behaviour, with the divergence being mostly on the scale of uncertainty. Once again, the peak of uncertainty depicted through normalisation (1) is found in the middle of the patch, due to the expected higher

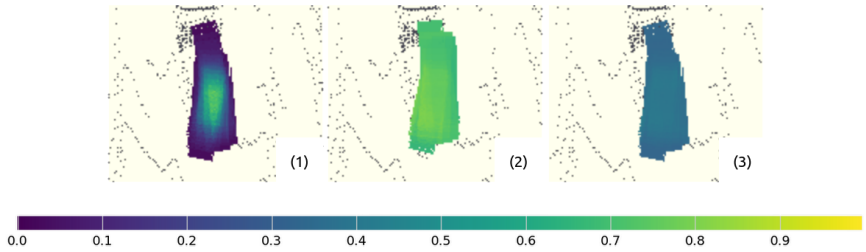


Figure 4.11: Detail of the area with high pointcloud noise before a ground-truth signalled object (not in the classification targets) through different Uncertainty Map regularisation techniques.

density of proposals in these cells. The regularisation method (3) leads to low variability in estimates over the considered area, as seen through the constant colouring in the corresponding map.

4.8.2.2 Beta parametrisation

Employing the *joint entropy estimation* model to generate Uncertainty Maps, there is the need to utilise a softmax function to transform logits in the classification vector into a categorical distribution. This function is parametrised by β, as shown in Equation 4.24, a factor that controls the entropy in the resulting distribution.

This β parameter introduces changes in the visual information conveyed by Uncertainty Maps. In Figure 4.12, some examples with $\beta \in \{0.15, 0.25, 0.4\}$ are shown for a comparative analysis.

It is observable that increasing the temperature, namely decreasing β (recall that $\beta = \frac{1}{T}$), leads to higher aleatoric uncertainty over the entire grid. This is due to the fact that the β parameter controls the weight given to the distance in the logits of the classification vector. With higher temperatures, the spread of the resulting distribution is higher, thus increasing the entropy. Since the *joint entropy estimation* model entails the summation of the entropies of each categorical distribution, the higher the spread, the higher the overall values of the aleatoric uncertainty.

Despite the difference denoted in the range of colours between β values, the information conveyed aside from the scale of the uncertainty is rather similar in all the generated maps. From Figure 4.12, it can be denoted that the progression between uncertainty levels through the space of the input sample stays constant despite the diverging parametrisation. This is the expected result as the β parameter should only control the magnitude of the uncertainty but not its tendency.

The observable differences in the predictions made by the uncertainty maps originate in the intrinsic randomness of the neural network which may lead to slightly diverging results using the same hyperparameters and model. Each column of Figure 4.12 was obtained from a test run, with all having the same setup, hyperparameters and model.

4.8.2.3 Comparing Uncertainty Interpretations

Having analysed some aspects of each aleatoric uncertainty interpretation when employed to generate an Uncertainty Map, it is then of interest to perform a comparative study between the results

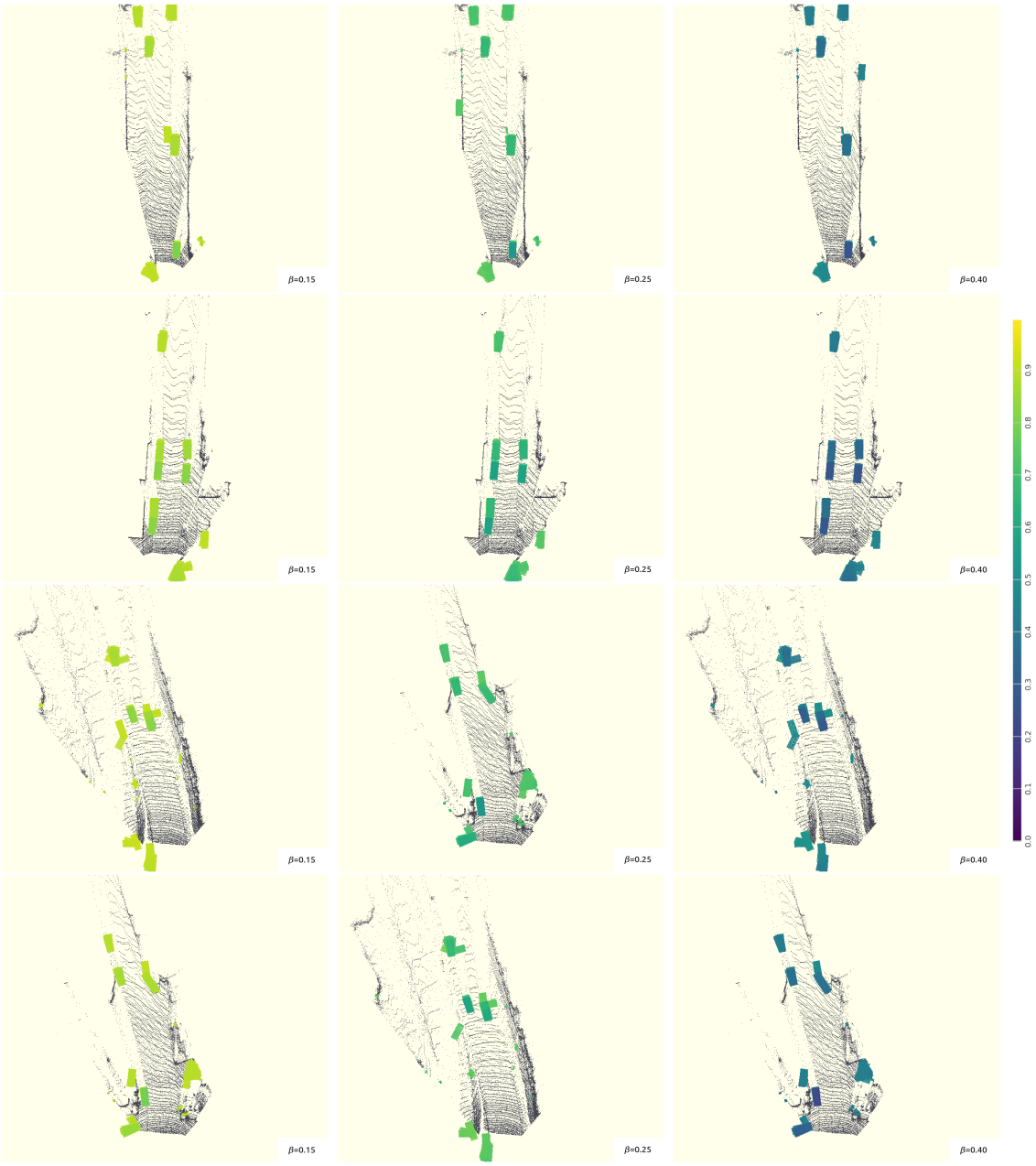


Figure 4.12: Uncertainty Maps generated with different β parameters in the softmax function. The β parameter controls the entropy of each anchor's categorical distribution.

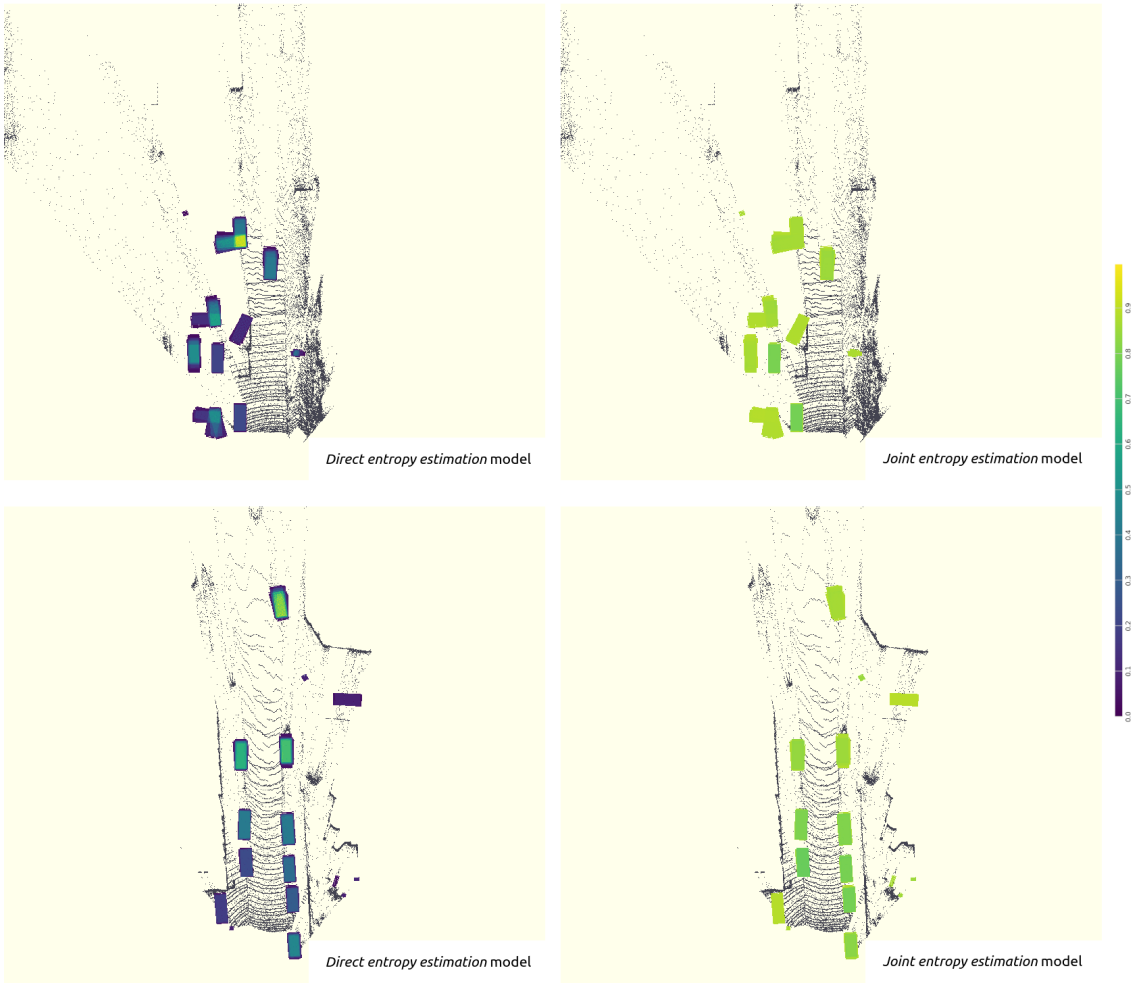


Figure 4.13: Uncertainty Maps obtained using the *direct entropy estimation* versus the *joint entropy estimation*. The colorisation changes abruptly due to the difference in the scale of the estimated uncertainties.

of each methodology. Figure 4.13 demonstrates two examples with the corresponding maps obtained by using the *direct entropy estimation* model, on the left, and by leveraging the *joint entropy estimation*, on the right. The maps based on the *direct entropy estimation* methodology have been normalised with technique (1) whilst the ones for the *joint entropy estimation* model were generated with a β parameter of 0.25.

The most apparent difference between both the generated maps is seen in the resulting colouring. Whilst the examples on the left sport a large range of colours from zero to one, the samples obtained on the right have estimates closely surrounding a narrow interval (in this case $[0.8, 1]$). This indicates the tendency of the *joint entropy estimation* model to generally lead to higher uncertainties which can be attributed in part to the summation of entropies. As mentioned previously, the particular maps on the right were generated with $\beta = 0.25$, which, as previously discussed, creates some closeness in the attributed probabilities in the categorical distribution, thus leading to higher estimates of uncertainty. Apart from the obtained values, which are highly dependant on the β parameter, the most important factor to denote is the overall smoothness of colour between all signalled areas. This showcases low variability in between estimates regardless of the anchor's distributions.

In fact, there is a high contrast when generating uncertainty maps using the *direct entropy estimation* and the *joint entropy estimation*. Whilst the maps of the *direct entropy estimation* method more closely relate to the inverse confidence of the model on each detection, the *joint entropy estimation* model expresses a more uniform uncertainty between patches without a close relationship to the confidence on each detection. This evidence is coherent with the formulation of each uncertainty interpretation as the *direct entropy estimation* expresses the summation of the confidences attributed to each anchor multiplied by its logarithm.

The separation between confidence and uncertainty is important and useful as a close correlation may demonstrate bias in the uncertainty estimation method. Even though it can be argued that uncertainty should be absolute, using a neural network in the estimation procedure will always lead to biased calculations and different predictors output diverging uncertainties.

This contrast in between the interpretations seems to suggest that the *direct entropy estimation* method has a more denotable bias and that its calculation is more attached to the object detection model's performance. On another note, the uncertainty maps generated through the *joint entropy estimation* methodology are not as informative as the conveyed uncertainty varies only slightly, demonstrating a dim contrast in between areas of the input data. This behaviour most likely originates in the summation of entropies which leads the final estimates to concentrate in a closer range.

4.8.3 Concerning the Existence Probability

Figure 4.14 showcases some false positives made by the standard model with significantly high confidences. As observed, the Existence Probability enables the calibration of this confidence to a more coherent value that expresses the end result.

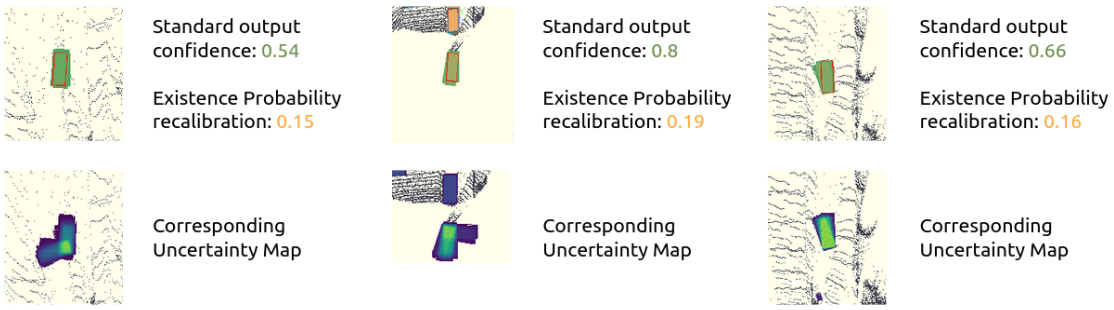


Figure 4.14: Examples of false positives made by the standard model sporting considerably high detection confidence alongside the Existence Probability recalibration. The lack of a blue bounding box showcases that the ground-truth does not signal these examples as objects.

In all three examples, the Existence Map features a low probability, indicating the object may not exist. This information, alongside the uncertainty seen in each corresponding Uncertainty Map, is integrated by the Existence Probability, dropping the confidences considerably.

Furthermore, it is observable that the aleatoric uncertainty surrounding these detections is close to the maximum, especially in the centre area. The detection in the middle, in particular, showcases a substantially high confidence of 0.8 despite being a false positive. This demonstrates severe overconfidence from the model, even more severely since the Existence Map shows little support on the existence of the object. From this qualitative analysis, it can be seen that the Existence Probability has the potential to recalibrate output probabilities, expressing more grounded values. Further evaluation is performed in the next chapter with a quantitative analysis of the results enabled by the use of the Existence Probability.

4.9 Discussion

The Existence Map and Existence Probability proposals in this chapter align with the fields of open-set classification and recognition [Vaze et al. (2021); Balasubramanian et al. (2021); Yang et al. (2024a)], out-of-distribution detection [Salehi et al. (2021)], and novelty and anomaly detection [Salehi et al. (2021)]. While these areas share similar objectives with the discussed concepts, their approach to the problem differs significantly. These proposals are positioned within the framework of standard object detection, without relying on out-of-distribution data, and focus solely on the knowledge embedded in an existing, unchanged model. In contrast, works on novelty and anomaly detection aim to identify specific outliers, often disregarding the traditionally detected objects. On the contrary, this thesis encompass objects in their original, conceptual form.

Conversely, the discussed applications of uncertainty aim to enhance the interpretability of neural networks [Zhang et al. (2021); Wang et al. (2023); Wan et al. (2024)]. On the topic of object existence, the works most closely related fall under the category of hidden semantics. For instance, Wu and Song (2019) propose a method to reveal the latent structures of models in regions

of interest for object detection. Another relevant study, by Sun and Jacobs (2017), introduces a convolutional network designed to detect missing objects in images.

Many existing research targets the estimation of epistemic uncertainty in object detection [Yun and Liu (2023); Riedlinger et al. (2023)]. Other works focus on providing probabilistic bounding boxes with semantic and spatial information encoded [He et al. (2019); Feng et al. (2020); Hall et al. (2020); Feng et al. (2021)]. As further experimentation in forthcoming chapters shows, by supplementing the standard output of an object detector, a broader understanding of the model's knowledge may be obtained, akin to the benefits provided by interpretability focused methods.

Concerning the Existence Map and the Existence Probability, the following works by Uchinoura et al. (2023) and Zohar et al. (2023) are worthy of note. Uchinoura et al. (2023) generate probabilistic maps through an instance segmentation pre-task in order to differentiate between background and foreground, leading to improved accuracy of the object detection model. Even though the generation of probabilistic maps over the input sample is discussed in this chapter, the proposed method does not rely on extra tasks, using only the knowledge already contained within the model. Further than this, the objectives of this thesis are not solely to mask background classes to improve detection accuracy but to provide insight on predictors generated by deep learning architectures. Zohar et al. (2023) address the novel task of open world object detection through the calculation of an objectness score for every proposal. The conceptualised Existence Probability has a similar scope, however, this methodology also enables the merging of the information from the standard output and the Existence Map as well as provide improvement of the calibration of confidence outputs of object detection models. Moreover, the proposed method is simple and there is no need to modify the network itself, unlike Zohar et al. (2023).

Pertaining to uncertainty maps, several recent applications have leveraged and studied parallel concepts to aid visual communication of uncertainties [Nazarovs et al. (2022); Maruccio et al. (2024)]. Whilst diverging in visual form, the effect and objective of these mechanisms is similar to that of the Uncertainty Map. Despite this, the information contained within an Uncertainty Map conveys further meaning as these are an outcome from anchor distributions, deliberating on existence as well as on the uncertainty of the input.

Overall, contrary to most existing approaches, existence and uncertainty maps can be employed in parallel to any object detection pipeline, as long as anchor distributions are generated, without the necessity to change the model's architecture. In fact, these proposals can be leveraged without the need for retraining, although their information may also be utilised during a network's development to shed light onto the optimisation process. This signifies that existence and uncertainty maps can be leveraged even in already deployed models. Further than this, the proposed module can be optimised in the target environment to be compatible with real-time processing.

Further than the applications of uncertainty, two interpretations for the quantification of the aleatoric parcel are proposed as well as an epistemic estimation methodology. Harakeh et al. (2019) discuss an uncertainty quantification method grounded in Bayesian statistics that leverages anchors. The authors specifically reformulate the standard inference and non-maximum suppression components of object detection models from a Bayesian perspective, leading to the availabil-

ity of uncertainty estimates. Nevertheless, the discussed method entails the modification of the standard architecture with training under the framework needed for the availability of uncertainty estimates. Further than these aspects, the complexity of the neural network's final processing layers is greatly increased as Bayesian inference is applied instead of non-maximum suppression. Similarly to the existence and uncertainty maps and the existence probability, the availability of uncertainty estimates using the proposals of this thesis is dependant only on the model generating anchor proposals. These proposals can be collected in parallel to the standard propagation, whether during training, testing or deployment. As such, with the interpretations of uncertainty discussed in this chapter, estimates are available at any stage of a deep learning development and deployment pipeline in a real-time compatible environment.

Even though there is not much literature concerning the use of the anchor distributions for the quantification of uncertainty, the formulated interpretations are based in pre-existent mathematical formulations. In Information Theory, the aleatoric uncertainty is deduced by the entropy of the true conditional distribution [Ash (1965); Malinin (2019)]. Further, several models of uncertainty isolate the epistemic component by taking the mutual information between Y and $Y|\theta$ [Ash (1965)]. This formulae boils down to taking the expected value of the KL divergence of $Y|\theta$ and Y.

Despite the existence of these concepts, the proposals in this chapter adapt these ideas to object detection, challenging the interpretations given to the predictive distribution as well as its conditional isolation and mutual information. In the *direct entropy estimation* model, the conditional distribution is interpreted as the distribution of existence given a set of generated patches, boiling down to the entropy of the maximum values in the classification categorical distributions. On the other hand, the *joint entropy estimation* method utilises the summation property of the entropy, expressing the conditional entropy as the summation of the entropies over each anchor distribution. In the case of the epistemic parcel, the outcome differs from the mutual information since the distance function diverges from the standard KL divergence. The mutual information aims to measure the expected information gained about one variable through the observation of another. In standard interpretations of uncertainty, these variables are Y and $Y|\theta$. In the proposals of this chapter, the distance is measured in order to obtain the information gained by using the anchor distribution in place of the predicted output which once more diverges from the existent interpretation [Ash (1965)].

Further than reinterpreting these existent formulae, the sampling of $P(Y|X,\theta)$ is done in a novel approach by re-utilising the variability already present in the region proposal layer through the anchor distributions. The use of internal mechanisms pre-existent in the neural network enable simple and immediate sampling of an approximated predictive distribution that is addressed by multiplied processing or memory efforts in current literature [Gal and Ghahramani (2016); Lakshminarayanan et al. (2017); Teye et al. (2018)].

4.10 Chapter Summary

In this chapter, the concept of the existence of an object is presented. This includes the definition of the existence random variable, which conveys whether an object exists or not in a bounded area, regardless of its class. Further than this, different interpretations of uncertainty are discussed, followed by some applications of these models in practical settings.

In this sense, Section 4.1 first formally details the problem formulation, reporting on the identified limitations of standard object detection models. Following this, Section 4.2 then addresses the preliminary definitions needed to support the discussions and derivations presented in further sections, ranging from the anchor distribution, the creation of a grid of the input sample and the idea of a patch.

Building upon the presented concepts, Section 4.3 proposes Existence Maps, a direct application of existence where areas are signalled with a probability of containing an object. After establishing the formulations and ideas of Existence Maps, Section 4.4 discusses the modelling of uncertainty with different interpretations being explored for both the aleatoric and epistemic parcels. Building upon this, an application of these uncertainty models in the context of existence is debated in Section 4.5. The proposed Uncertainty Maps offer the same format for visualisation of Existence Maps with each cell on the grid representing uncertainty estimates instead of reflecting the possible detection of an object. Another methodology that applies both the ideas of existence and uncertainty is reported in Section 4.6, namely the Existence Probability, which allows for the calibration of the confidence of object detection outputs.

After these ideas are thoroughly exposed, Section 4.7 reflects upon Existence and Uncertainty Maps in the sense of their time and spatial complexity. This is followed by Section 4.8 on which the qualitative evaluation of all the concepts discussed previously is carried out. These analysis focus on the properties of uncertainty, including the amount of information conveyed through its proposed applications.

Finally, Section 4.9 comments on prior works of interest and differentiates between the proposals made in this chapter and those existent in the literature.

Chapter 5

Object Detection in Autonomous Driving

Several interpretations of uncertainty and possible applications were discussed in the previous chapter. The provided descriptions include the needed formulations and methodologies to implement aforementioned interpretations, including object detection mechanisms where uncertainty information may be leveraged. The chapter also includes a panoply of qualitative validations that express some characteristics and behaviours expressed by the proposed methods.

In line with the previously exposed ideas, this chapter moves to a more formal, quantitative approach to the evaluation of these concepts. In this sense, firstly, the evaluation problematic addressed in this chapter is formally defined in Section 5.1 with the goals for the analysis of each interpretation and application of uncertainty described in detail. Section 5.2 follows, addressing the architecture of the model selected for evaluation purposes. Continuing the preliminaries is Section 5.3 that discusses relevant implementation details that allow the exact replication of the reported experiments. Finally, finalising the notable conditions and aspects of the validation environment is Section 5.4 with an explanation of some metrics of interest to the evaluation procedure.

The experimental procedures are then reported alongside their results in Section 5.5, for the uncertainty interpretations, Section 5.6, for the existence and uncertainty maps and Section 5.7 for the Existence Probability. Discussion of the results finalises the chapter, featured in Section 5.8.

5.1 Problem Definition

The proposals discussed in Chapter 4 include two aleatoric uncertainty estimation methods, the *direct entropy estimation* and the *joint entropy estimation*, one epistemic uncertainty quantification approach and several applications where the information provided by uncertainty estimates may be leveraged. These applications range from the Existence and Uncertainty Map to the Existence Probability.

In order to formally and quantitatively evaluate all of these concepts, the following aspects must be taken into consideration:

Uncertainty Interpretations The proposed methods target the estimation of both the aleatoric and epistemic parcels of uncertainty, as mentioned previously. The problematic of the evaluation of these interpretations is centred around the ascertainment of fitness, coherence and adaptability of the obtained estimates. In this sense, a valid approach is to perform a correlation analysis between the obtained quantities and factors of interest such as: distance to the ego vehicle, occlusion, class and detection confidence. The behaviour of uncertainty estimates concerning these aspects illustrates how each quantification method encapsulates desirable qualities of estimates such as explainability, calibration and usefulness [Feng (2021)]. The parameters needed for the calibration of the methods also need careful contemplation, with an analysis of the effects of interchanging values being indispensable.

Existence Map Existence maps can provide additional information to the standard output of a model as a stand-alone tool to improve the communication of object detection outcomes. Even though this information *as it is* is valuable, the effectiveness of the suggestion of areas with the possibility of object existence can be effectively evaluated and quantified. Following this idea, a formal method of evaluation of the accuracy of the maps is the measuring of the rates of true positives, false positives and false negatives. The outcomes of such a study can be directly compared to the standard output as precision and recall can be evaluated in test and out-of-distribution/perturbation sets in both scenarios - standard output and stand-alone Existence Map. This configuration reveals the stand-alone performance of existence maps in providing detection outputs regardless of the model's learning process after further fine-graining. Once more, the parametrisation of the Existence Map generation process is also a focal point of concern, with the reporting of adequate values needing to be based off of real experimental data.

Uncertainty Map Uncertainty maps, on the other hand, provide a visual representation of the aleatoric uncertainty over the space of input samples. This means that the application directly targets the usefulness aspect of uncertainty estimates, as mentioned previously. Since the evaluation of an Uncertainty Map heavily relies on the coherence of the uncertainty estimate itself, the aforementioned validation of the uncertainty interpretations matters the most to the indication of the fitness of generated maps. A specific aspect of this application is found in the regularisation techniques employed which greatly alter the generated maps, as discussed in the previous chapter under the qualitative analysis. On a formal paradigm, a study of the correspondence between these parameters and the intended behaviour of these maps in sampled detections can provide a qualitative sense of calibration, usefulness and explainability of these maps.

Existence Probability The Existence Probability, as well as the proposed method for merging standard output detections with patches containing anchor distributions, enables the distillation of bounding boxes utilising the knowledge of both the final model as well as its region proposal layer in the context of the existence concept. Qualitative assessment of the merging once again can be carried out by measuring the rates of true positives, false positives

and false negatives, leading to the estimation of the precision and recall metrics that are directly comparable to other approaches. Calibration of the model, on the other hand, can be observed through reliability diagrams, demonstrating the fitness of the model and the possibility of further uncertainty usefulness. At last, fine-tuning of the parameters needed for the calculation of the Existence Probability is also of interest, with its results impacting the calibration outcome.

Taking into consideration these aspects and characteristics, the problematic of this chapter may be defined as follows: to formally and quantitatively evaluate the proposed uncertainty interpretations as well as its suggested applications, namely the Existence and Uncertainty Map, and the Existence Probability in the context of the desirable characteristics of uncertainty estimates - to be explainable, calibrated and useful [Feng (2021)]. This procedure intends to demonstrate the fitness of the proposed concepts by reporting their suitability in real object detection for autonomous driving contexts.

5.2 The VirConv-L Model

The model selected for the evaluation of the proposals in this thesis is the VirConv [Wu et al. (2023b)]. This study falls under the modality of virtual points, a method popularised in recent object detection research. Leveraging this idea, three different architectures for object detection that utilise the VirConv backbone are discussed by Wu et al. (2023b): VirConv-L, a lightweight, efficient model that is based on the early fusion for a multimodal input; VirConv-T, a larger and more complex architecture that focuses on high precision with a transformed refinement scheme; and VirConv-S, a pseudo-label semi-supervised framework.

Since the desired target application is autonomous driving, the efficiency is a concern since deployment depends on real-time capabilities. For this reason and since the lightweight architecture also achieves competitive state-of-the-art results, the VirConv-L model is selected for the evaluation of the proposed methodologies. VirConv-L also relies on both RGB images and pointcloud data, thus covering the major and most relevant sensors available in self-driving vehicles.

The architecture of this model is composed of the common object detection modules, from the backbone that carries out the feature extraction, to the region proposal that leverages anchors, to the detection head that is followed by post-processing, thus generating final bounding boxes. The diverging aspects of this pipeline rely mostly on the transformation of the input data and the backbone network. Figure 5.1 showcases the VirConv-L architecture in a modular fashion, similarly to the global model discussed in Subsection 2.4.1. Each of the modules is detailed with the specific procedure followed by VirConv-L.

As mentioned previously, Wu et al. (2023b) propose the use of virtual points to enrich the input pointcloud data. The lightweight model, chosen for evaluation, first generates these virtual points through a depth completion pre-task, thus merging the RGB features from camera images with three-dimensional data. Before feature extraction, in the VirConv-L module, first the virtual

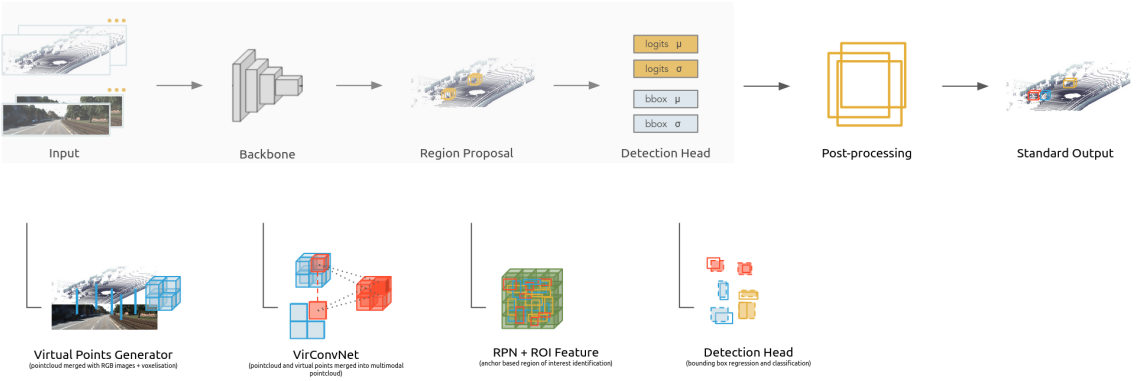


Figure 5.1: Architecture of the VirConv-L model.

points and the LiDAR points are fused, resulting in a multimodal pointcloud that encodes features from both domains. The module then carries out the extraction of features from this multimodal data, leading to a rich transformation of the input. This is followed by the region proposal which utilises anchor proposals in order to signal possible objects. The results from the anchor head are further refined in the Region of Interest Feature module, leveraging several transformations of backbone features. The obtained RoIs are fused together through box voting, an approach popularised by Wu et al. (2022b).

5.3 Implementation Details

After having defined the architecture of the model selected for evaluation purposes, to enable a complete replication of the forthcoming experiments, some implementation details must be discussed. These are divided in two subsections, 5.3.1 and 5.3.2. The first subsection reports all relevant hyperparameters and training settings of the VirConv-L model. The other follows with a description of the pertinent aspects of the generation of Existence and Uncertainty Maps.

5.3.1 VirConv-L

The codebase for the replication of the VirConv models is publicly available in a code hosting platform [1]

The construction of the VirConv module follows a similar code pattern as OpenPCDet [Team (2020)], a popular modular toolbox for three-dimensional object detection using pointcloud data. The basis of this open-source toolbox is to provide easy access to a panoply of state-of-the-art object detection models all in the same codebase, simplifying the creation of baselines and the comparison of proposals with existent literature. The code follows a highly modular approach and reflects the construction of object detection models as depicted globally in Subsection 2.4.1.

[1] Codebase can be found at Github. Development is carried out from version *3c00586*, last pulled in the 20th of September of 2023.

Table 5.1: Training parameters.

Batch Size	Epochs	Optimizer	Learning Rate	Weight Decay	Momentum
4, 16 (standard)	50	Adam	0.01	0.01	0.9

Since VirConv-L depends on the creation of a multimodal pointcloud before feature extraction, the generation of virtual points is the first aspect of interest to discuss. As mentioned in the previous section, virtual points are obtained by fusing RGB and pointcloud data through a depth completion pre-task. As such, for the preparation of the model for training and testing, the input data must be passed through a depth completion model. The model selected for the transformation of the datasets is PENet by Hu et al. (2021). This architecture achieves state-of-the-art results on the KITTI depth completion task, thus enabling a good conversion of the input data.

Continuing on the transformation of the input data into a suitable representation for feature extraction, the voxelisation of the multimodal pointcloud is an essential step in the VirConv-L pipeline. The maximum voxel size assumed is $0.05x0.05x0.05$ meters with the number of points per voxel capped at 5.

Concerning the model training, Table 5.1 reports the main hyper parameters selected for all experiments. Aside from the batch size which is varied at times (16 most commonly, at times 4), all other parameters remain equal throughout all reported evaluations. Following recommendations from the original work by Wu et al. (2023b), all predictors are trained during 50 epochs.

As for the region proposal module, Table 5.2 reports the parameters selected for the pre-defined anchor proposals per class. For each, two headings are considered, spawning a bounding box either parallel to the x or y-axis. The matched threshold corresponds to the minimum confidence for considering an anchor as foreground and respectively, the unmatched threshold indicates whether a proposal contains only background information. These thresholds are utilised in the model's pipeline, leading to the standard output detections.

5.3.2 Existence and Uncertainty Maps

In Section 4.7, the time complexity of the generation process is discussed in function of n that represents the number of anchor proposals in a given sample $\mathbb{S}$. In order to speed up experiments,

Table 5.2: Anchor configurations used with VirConv-L over the three classes of interest, *Car*, *Pedestrian* and *Cyclist*.

Class	Dimensions (dx,dy,dz) (m)	Rotations (radians)	Anchor Bottom Heights	Matched Threshold (standard output)	Unmatched Threshold (standard output)
Car	3.9 x 1.6 x 1.56	[0, 1.57]	-1.78	0.6	0.45
Pedestrian	0.8 x 0.6 x 1.73	[0, 1.57]	-0.6	0.5	0.35
Cyclist	1.76 x 0.6 x 1.73	[0, 1.57]	-0.6	0.5	0.35

n is set to 120. This is an approximation of considering an average of 5.5 objects per sample times the number of classes and headings multiplied by the mean amount of cells per feature map.

In order to keep false positives low and to enable higher accuracy, patches with an existence probability lower than an established lower bound are discarded. Low probability areas often stem from noise in the process, coming from either the uncalibrated confidences or data-intrinsic randomness. Especially in the case of pointclouds, there are frequently areas with a high concentration of noise. This parameter is referred to as the probability threshold and, when omitted, its standard value is 0.25.

Further than this, where approximations of expected values are needed, these are obtained by averaging entropies. This is utilised in the estimation of the aleatoric uncertainty of the model, for example, as showcased by Equation 4.6.

Finally, the codebase for the generation of Existence and Uncertainty Maps is independent of model, being transferable to any architecture. The framework can be leveraged in parallel to any object detector, as long as the region proposal module relies on anchor proposals for the identification of foreground and background areas. Concerning the evaluation model, VirConv-L, that follows the OpenPCDet [Team (2020)] structure, the existence class is constructed in the initiation function of the anchor head.

5.3.3 Datasets and splits

For the main experiments discussed in this chapter, the KITTI dataset [Geiger et al. (2012)] is selected due to its prominence in the field and availability of data for a large panoply of driving tasks.

The data provided by the KITTI benchmark features training, validation and test sets already split into proper sets. However, since benchmarking is made through a submission to their website, the labels for the test set are not publicly available. For this reason, in order to carry out the experiments with labelled samples, the training dataset is split into two parts, thus creating new train and test sets. Following prior literature [Wu et al. (2022b, 2023b)], the original training set, composed of 7481 frames, is split into training samples (3712 frames) and a test set with 3769 examples. As such, all predictors shown in forthcoming experiments are only trained on 3712 frames and results reflect performance on the testing set of 3769 frames.

For out-of-distribution and perturbation data, the KITTI-C [Kong et al. (2023)] dataset is leveraged. Kong et al. (2023) present a suite of popular autonomous driving datasets in several perturbation modes in order to allow better robustness evaluations of models for self-driving vehicles. The KITTI dataset is included in this suite with the following modes contemplated:

beam_missing Represents external disturbances that result in an incomplete pointcloud with some beams being missed by the LiDAR sensor;

cross_sensor Represents the variety of parameters found in a LiDAR sensor, thus resulting in entirely diverging pointclouds;

crosstalk Represents a phenomenon experienced when several sensors are close to each other, causing impulses with similar frequencies to interfere with each other;

fog Represents the weather condition of fog, which can cause back-scattering and attenuation of the impulses sent by the LiDAR sensor;

incomplete_echo Represents the incompleteness of the pointcloud when pulses are emitted towards objects with dark colours, causing the fading of the impulses;

motion_blur Represents the blur resulting from the movement of the ego vehicle, especially in bumpy surfaces;

snow Represents the weather condition of snow, which may cause confusion in the sensor with reflections of impulses being caused by the particles in the air, ultimately leading to occlusions;

wet_ground Represents the attenuated beams when reflecting off of wet surfaces.

Each of these modes is contemplated over three degrees of severity, namely light, moderate and heavy. For a visual representation of the differences between these modes and degrees of severity, refer to Kong et al. (2023).

5.4 Evaluation Methods

Pearson's correlation coefficient quantifies the linear relationship between two random variables. In forthcoming experiments, this coefficient is utilised to define the connection between uncertainty estimates and other variables such as occlusion and distance to the ego vehicle. The PCC score can be calculated by the formula depicted in Equation 5.1.

$$\begin{aligned} \text{PCC Score}(X,Y) &= \frac{\sum_{n=1}^{N}(x_n-\bar{x})(y_n-\bar{y})}{\sqrt{\sum_{n=1}^{N}(x_n-\bar{x})^2}\sqrt{\sum_{n=1}^{N}(y_n-\bar{y})^2}} \\ \text{with } \bar{x} &= \frac{1}{N}\sum_{n=1}^{N}x_n, \bar{y} = \frac{1}{N}\sum_{n=1}^{N}y_n \end{aligned} \tag{5.1}$$

The score can range from -1 to 1, where a negative score depicts a negative correlation between the variables and a positive one a positive relation. A zeroed coefficient demonstrates that there is no linear relationship of the variables. Scores in between these values may be interpreted differently depending on the context and observer. In this sense, previous literature is followed [Feng (2021)] when discussing PCC scores with the utilised interpretation of strength being displayed in Table 5.3.

For the analysis of uncertainty calibration, a metric is devised, as shown in Equation 5.2. Akin to the Expected Normalised Calibration Error (ENCE) [Levi et al. (2022)], the uncertainties and model outputs are divided into $|B|$ bins, with j representing one bin within B. For the study of

correlation with box regression error, Er corresponds to the average $1 - IoU$ found in j, whilst for classification, the cross entropy between targets and predicted probabilities is averaged in each bin. The calibration error is then obtained as a summation over all bins, being normalised by the avereage uncertainty in each bin, $\bar{U}(j)$.

$$\frac{1}{|B|}\sum_{j=1}^{N}\frac{|\bar{U}(j)-\bar{E}r(j)|}{\bar{U}(j)} \tag{5.2}$$

For the analysis of the efficiency of object detection, the standard metrics of precision and recall are utilised. As such, result comparison focuses on the rates of true positives, false positives and false negatives as previously defined in Section 2.4.3. A perfect object detector achieves a rate of 100% true positives and 0% false positives and negatives, resulting in 100% precision and recall.

In this sense, and since it is of interest to experiment with probability thresholds, precision-recall curves are also leveraged for parameter analysis. The target precision-recall curve is represented in Figure 2.8, as discussed in Subsection 2.4.3.

Furthermore, for the study of the calibration of a model's confidence scores, reliability diagrams are employed. These diagrams are typically presented in histogram form and demonstrate the levels of outputted confidence versus the achieved accuracy. Using the top middle points of each bin, a calibration plot can be obtained, which directly demonstrates the calibration of models as depicted in Figure 3.10.

In order to quantify the difference between two histograms, a measure of interest to evaluate the distance between predictor outputs, the mean absolute error (MAE) as reported in Equation 5.3 is utilised. This metric is calculated for each predictor as the divergence in positive detections and the amount of total detections made.

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|x_n - y_n| \tag{5.3}$$

Finally, a correlation matrix is also employed as an auxiliary tool for the study of the relationships between variables of interest. This is specifically used for the evaluation of the parcels that compose the existence probability. Values in a correlation matrix also range from -1 to 1 with zero values representing an inexistent correlation and the other scores demonstrating a negative or positive relationship depending on the sign of the correlation.

5.5 Experimental Evaluation of the Uncertainty Interpretations

The evaluation of the proposed uncertainty interpretations relies on the observation of the coherence of estimates pertaining to their explainability, calibration and usefulness. In line with this, firstly, it is of interest to analyse the parameters employed in each methodology. Secondly, in order to assess that estimates fulfil the aforementioned aspects, correlation analysis are carried out between the obtained uncertainties and a panoply of factors of relevance to the object detection

Table 5.3: Absolute Pearson's correlation coefficient and the interpretation of the strength of the relationship.

PCC Score (absolute)	Strength of relationship
0.00 - 0.19	Very weak
0.20 - 0.39	Weak
0.40 - 0.59	Moderate
0.60 - 0.79	Strong
0.80 - 1.00	Very strong

and self-driving vehicle domains. For the aleatoric parcel, a study of its relationship with distance to the ego vehicle, occlusion and difficulty of detections is presented. For the epistemic quantity, the distance to the ego vehicle and Existence Map probability are analysed.

5.5.1 Parameter Fine-tuning

Some of the methods reported in the proposals of the uncertainty interpretations feature selectable parameters. In the case of the aleatoric uncertainty, the *joint entropy estimation* model entails the definition of a β parameter that controls the entropy of the anchor's classification distribution. The epistemic uncertainty estimation, on the other hand, needs the choice of probability thresholds, ρ_{anchor} and ρ_{out}, for non matching boxes.

The β parameter is also analysed in Subsection 4.8.2 in a qualitative manner in the context of Uncertainty Maps. Contrary to this, in this subsection, the focus is to provide a quantitative validation and correlation analysis of these parameters in the face of some relevant aspects of uncertainty previously denoted.

5.5.1.1 The β Parameter

In order to analyse the influence of the β parameter in the *joint entropy estimation* model of aleatoric uncertainty, a set of three values are considered for β, namely $\{0.15, 0.25, 0.4\}$. Table 5.4 showcases the correlation scores calculated with the PCC score over the distance on the x and y axis, the occlusion and difficulty.

The obtained results suggest that the β parameter does not have a large impact on the behaviour of the uncertainty estimation procedure. It has been determined previously, in Subsection 4.8.2,

Table 5.4: Pearson's Correlation Coefficient scores obtained for diverging β parameters in the *joint entropy estimation* model over the distance, occlusion and difficulty.

β	Distance (x, y)	Occlusion	Difficulty
0.15	0.25, 0.18	0.0	-0.08
0.25	0.39, 0.29	0.03	-0.1
0.4	0.38, 0.29	0.01	-0.12

Table 5.5: Pearson's Correlation Coefficient scores obtained for different ρ_{anchor} and ρ_{out} parameters in the epistemic uncertainty interpretation over the distance and Existence Map confidence.

ρ_{anchor}	ρ_{out}	Distance (x, y)	Existence Map Confidence
0.1	0.1	0.017, 0.07	-0.375
0.1	0.25	0.06, 0.083	-0.382
0.25	0.1	0.001, 0.025	-0.29
0.25	0.25	-0.034, 0.001	-0.162

that this parameter controls the entropy of the categorical distribution. As such, its value significantly alters the generation of the uncertainty maps. Despite this, the observed maps showcased similar behaviour, even though uncertainties are in different ranges, also suggesting that the β parameter is inconsequential in the significance of the estimated aleatoric uncertainties.

In Table 5.4, an outlier to this reasoning can be found in the Pearson's coefficients obtained for a β of 0.15 over the distance. These values are the only parameters that showcase any variance in the obtained results, and as such, their origin could be explained in the intrinsic randomness to neural networks, that, despite being run with the same hyperparameters, lead to different outcomes spontaneously.

Considering the obtained uncertainty maps in Subsection 4.8.2, the chosen β value for forthcoming experiments is 0.25 so as to enable a better balance in the range of the represented uncertainties. The significance of these PCC scores in function of the distance, occlusion and difficulty are studied in depth in further sections.

5.5.1.2 The Epistemic Thresholds

Moving forward to the epistemic thresholds, the same procedure is followed, considering a set of values for the parameters of the interpretation of the epistemic uncertainty. The set of values assumed for the experiments are $\{0.1, 0.25\}$, for both ρ_{anchor} and ρ_{out}.

Table 5.5 reports the obtained results for diverging ρ_{anchor} and ρ_{out} factors in the epistemic uncertainty calculation. The impact of these factors is evaluated over the distance to each axis origin and the Existence Map confidence.

As observed, the variants considered for the controlling parameters of the epistemic uncertainty model have no perceivable impact on the relationship with distance and Existence Map confidence. The low variability seen is grounded in the intrinsic randomness of the neural network's processing that generates slightly different results each run.

Despite the correlation scores showing low divergence in terms of the controlling factors ρ_{anchor} and ρ_{out} and the analysed variables, through experimentation, it is observed that the increase of these parameters leads to overall lower epistemic uncertainties. This is expected as, with the increase of these values, the resulting distances are generally lower.

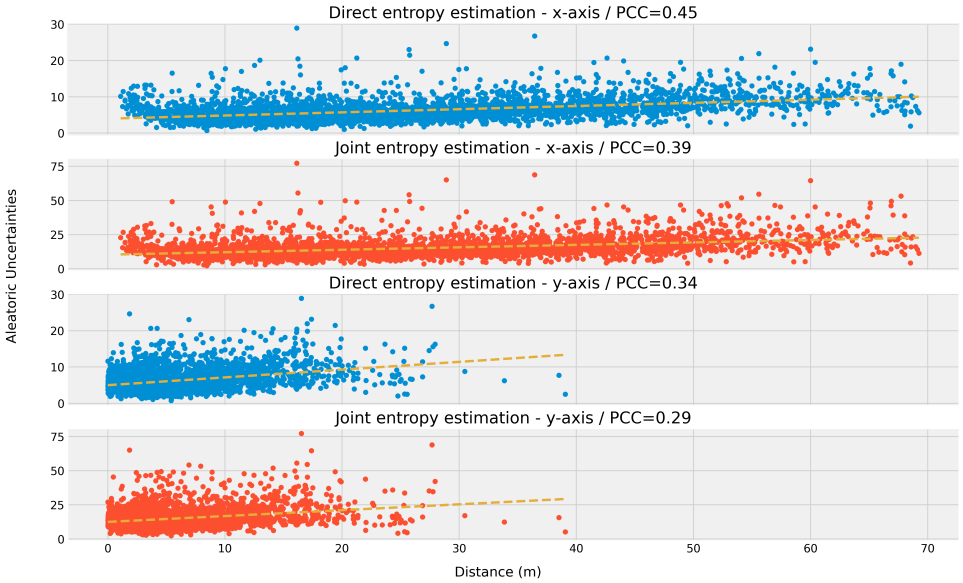


Figure 5.2: Aleatoric uncertainties estimated using both the *direct entropy estimation* and the *joint entropy estimation* models in function of the distance to the x and y axis' origins. Sample of 943 detections taken randomly from the test set.

5.5.2 Observations on Uncertainty Interpretations

In order to evaluate and observe the coherence of the proposed interpretations of uncertainty, aspects such as distance to the ego vehicle, occlusion, difficulty and Existence Map probability are considered. In these experiments, a population of 943 examples is sampled from the test set, representing 25% of the original evaluation data. Detections are obtained by feeding the VirConv-L model these samples.

5.5.2.1 Distance

The distance as interpreted for this correlation analysis corresponds to the distance of the centre coordinates of a detection to each axis origin, namely the distance of the bounding box to the ego vehicle. Since the z coordinate is more sparse in information than the $x-y$ plane, this axis is ignored in forthcoming reports.

Aleatoric uncertainties estimated by each quantification model are shown in Figure 5.2 over axis x and y in a sample of 943 detections. As it can be observed, for the *direct entropy estimation* method, as the distance to the ego vehicle increases, detections are significantly more sparse. This is coherent with reality as, the further away is the input information from the sensor, the definition in the data is diminished, increasing the difficulty of feature extraction, and thus leading to more scarce detections. It can also be denoted that the estimated aleatoric uncertainty has a tendency to increase as detections get farther from the axis origin. This linear relationship is more prominently seen in the x axis, as confirmed by the Pearson's correlation coefficient which equals 0.45, being classified as a relationship of *moderate* strength. Since uncertainty estimates must be explainable, namely, harder detections must correspond to higher values, this is coherent with the expected

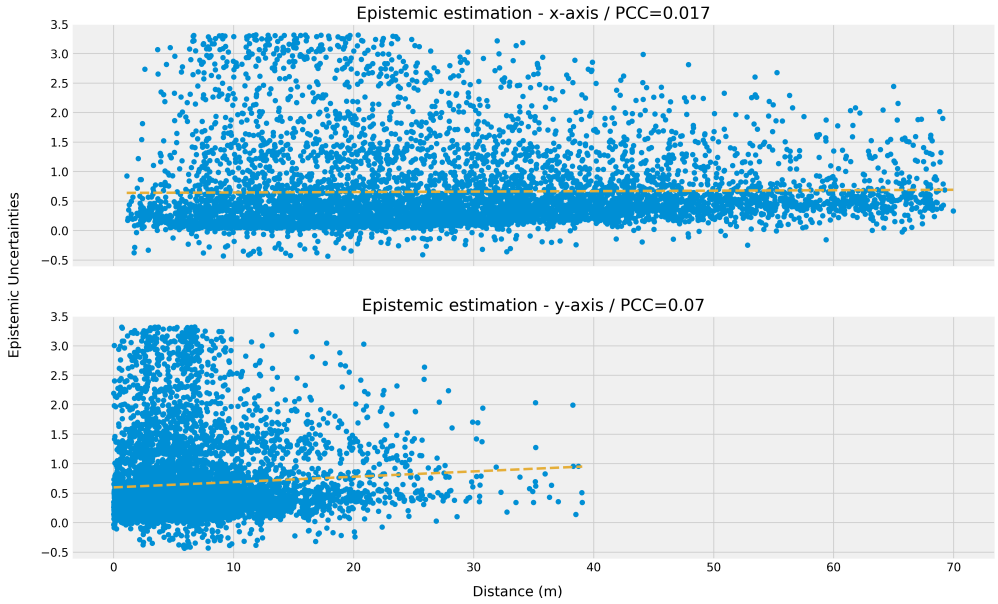


Figure 5.3: Epistemic uncertainties estimated in function of the distance to the x and y axis' origins. Sample of 943 detections taken randomly from the test set.

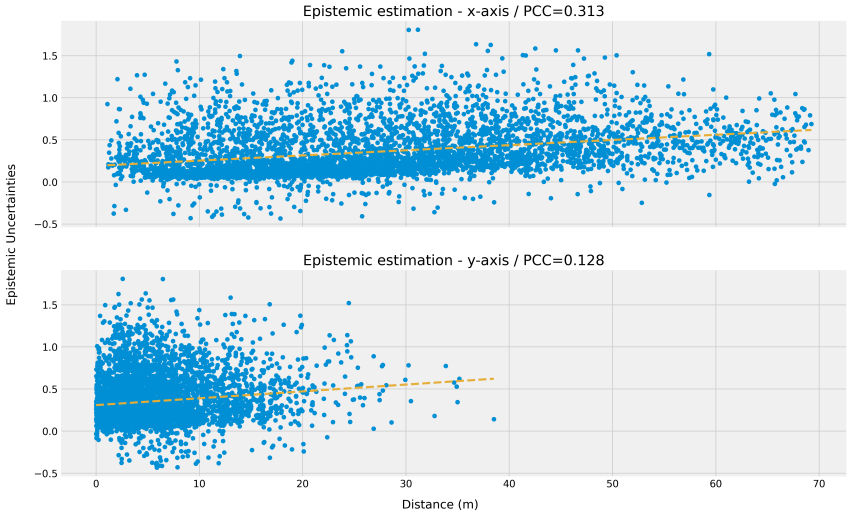
behaviour. As mentioned previously, the farther from the sensor, the less definition in the data, leading detections to become more complex. In the case of pointclouds, this usually corresponds with a low amount of points in each bounding box.

Concerning the y axis, the *direct entropy estimation* model showcases a less strong correlation with distance, achieving a PCC of 0.34 that is already classified as a *weak* relationship. This behaviour, once more, is expected, as the x axis represents the depth in relation to the ego vehicle and most objects in the utilised dataset concentrate alongside this axis, leading to a higher dependency to this coordinate.

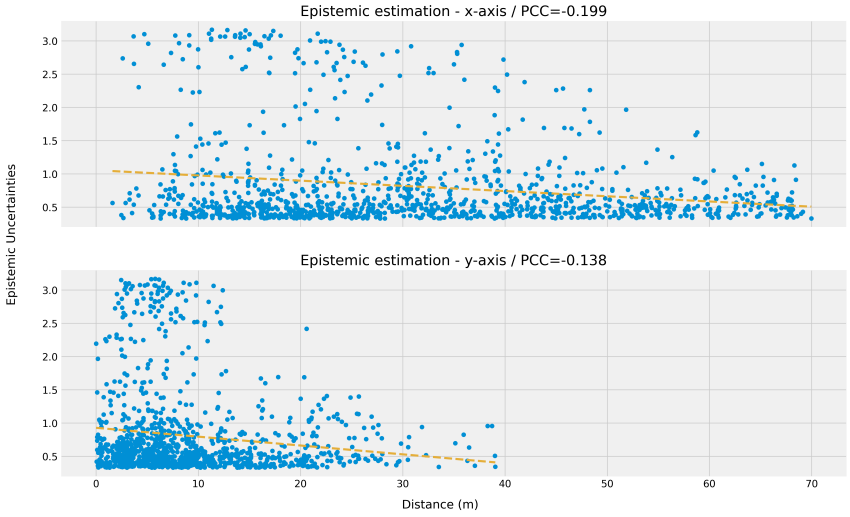
In the case of the *joint entropy estimation* method, this tendency is replicated, albeit with larger overall uncertainties over all detections. Nevertheless, the resulting PCC's, namely 0.39 on the x axis and 0.29 on the y axis, are lower than those obtained with the *direct entropy estimation*, with both expressing a *weak* correlation. This demonstrates that this approach to the quantification of the aleatoric uncertainty has less dependence on the distance to the ego vehicle which may signify that the summation of entropies dilutes the processed divergence to the underlying data distribution.

On the other hand, the epistemic uncertainty is not expected to feature a strong relationship with the distance to the axis' origins as its value depends on the variability of predictors generated from the same model, being independent from the input data.

The epistemic uncertainties obtained in function of the distance to the origin in each x and y axis are showcased in Figure 5.3. As observed, the estimates are randomly spread across the values along each axis, demonstrating no perceivable relationship. This is confirmed by Pearson's coefficient, which expresses a *very weak* correlation, assuming values 0.017 on the x axis and 0.07 on the y axis.



(a) Support from both the standard output and the Existence Map.



(b) Existence Map support.

(c) Standard output support.

Figure 5.4: Types of epistemic uncertainty parcels over the distance to the x and y axis' origins. Matched detections represent overlaps between the standard output and the Existence Map as shown in 5.4a, whilst detections with support solely from the Existence Map or the standard output respectively are seen in 5.4b and 5.4c. Sample of 943 detections taken randomly from the test set.

The methodology for the estimation of the epistemic uncertainty considers three diverging situations as mentioned in Subsection 4.4.2, namely when there is bounding box correspondence between the standard output and the Existence Map or when there is single support from either one ($a = 0$ or $y = 0$). In order to evaluate each component obtained in these divergent situations, Figure 5.4 shows the breakdown of the epistemic uncertainty in its components by the distance to each axis' origin.

It can be observed that each component has a significantly different behaviour. For the parcel that expresses correspondence between the standard output and the Existence Map, the correlation with the distance has some similarity to that featured by the aleatoric uncertainties, achieving a PCC score of 0.313 and 0.128 on the x and y axis respectively. Whilst the relationship with the x axis is significant, being classified as *weak*, in the case of the y axis, the strength is classified as *very weak*. This behaviour is possibly grounded in the fact that this component takes into consideration both the confidence of the standard output and the suggestions of existence maps. It has been established that there is generally higher aleatoric uncertainty as the distance grows, especially on the x axis. This data uncertainty can lead to higher predictor disagreement as the difficulty of performing correct detections increases as well.

As for the other components, their behaviour is similar, albeit with different strengths. With single Existence Map support ($y = 0$), there is a slight tendency to decreased uncertainty as the distance grows, whilst with the single standard detections as output ($a = 0$), there is more perceivable strength in this descent. This is confirmed by Pearson's coefficient assuming values -0.199 and -0.138 for Existence Map support ($y = 0$), depicting a *very weak* correlation on both axis'. It is also observable in the respective plot that the highest density of points concentrate on lower uncertainties with some sparse outliers in higher values.

On the other hand, on the standard output support plot ($a = 0$), it is observable that the points spread much more evenly over both the distance and the aleatoric uncertainties. The stronger correlation with distance is proven by the PCC score equalling -0.302 and -0.201, both corresponding to a *weak* relationship. This may demonstrate that the model is overconfident in its predictions as, in order to reach higher uncertainties, the epistemic interpretation shows that the output confidence is increased over the single Existence Map support boxes.

Decreasing values of uncertainty estimated by the model in singly supported bounding boxes as the distance increases may demonstrate low model confidence in these detections. Despite the fact that the considered detections have no backing from both the region proposal layer and the final fine-grained output, having low probabilities leads to small distances as expressed by the distance function utilised in the proposed model of epistemic uncertainty. This behaviour aligns with the expectation as less defined and detailed areas of the input data translate into more difficulty in the detections. This is also the case for most Existence Map supported boxes ($y = 0$) that, with no final output correspondence, typically sport low confidences, leading the distance to the controlling factor ρ_{out} to be small. As uncertainties are much higher in single standard support ($a = 0$), the confidences of the outputted boxes must be larger than those predicted by the Existence Map when $y = 0$.

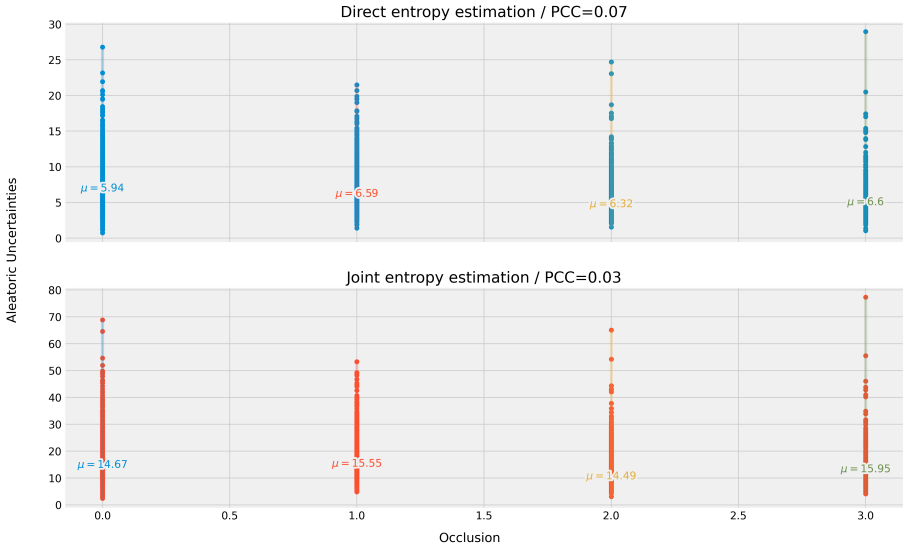


Figure 5.5: Aleatoric uncertainties estimated using both the *direct entropy estimation* and the *joint entropy estimation* models in function of the occlusion as classified in the KITTI dataset. Sample of 200 detections taken randomly from the test set.

5.5.2.2 Occlusion

Occlusion represents the visibility of an object as seen by the ego vehicle. This factor is already available in the ground-truth labels of the KITTI dataset. As determined in the dataset, four levels of occlusion are considered, with 0 corresponding to a fully visible object and 3 to a difficult to see target. Once more, this experiment features a subsample of 943 detections taken randomly from the test set.

Figure 5.5 demonstrates the obtained aleatoric uncertainties over each occlusion level for the proposed estimation methods. Both the *direct entropy estimation* and the *joint entropy estimation* models showcase similar behaviour over all levels, diverging slightly on the average magnitude of the estimated uncertainties. As observable, both methods show no correlation with the occlusion categorisation of the object, with the Pearson's coefficient equalling 0.07 and 0.03 respectively. Despite the low variability in the results over occlusion levels, there is still a perceivable tendency of the maximum occlusion level to sport higher uncertainties on average on both models, albeit with a small margin.

The expected outcome of this experiment, at first thought, would be at least a *weak* correlation between the estimated uncertainties and the occlusion level. Since occluded objects are covered by others, the resulting pointcloud would be expected to have less points in the area. Furthermore, since occlusion signifies the proximity of other objects, this can increase the probability of the occurrence of phenomenons that decrease the quality of the collected points in the area, leading to more noise in the resulting pointcloud.

Even so, occlusion itself is insufficient to judge whether a certain area should have high aleatoric uncertainty or not. As mentioned previously, these phenomenons may decrease point-cloud quality, however, by having a probability of occurring, there is also a high possibility of

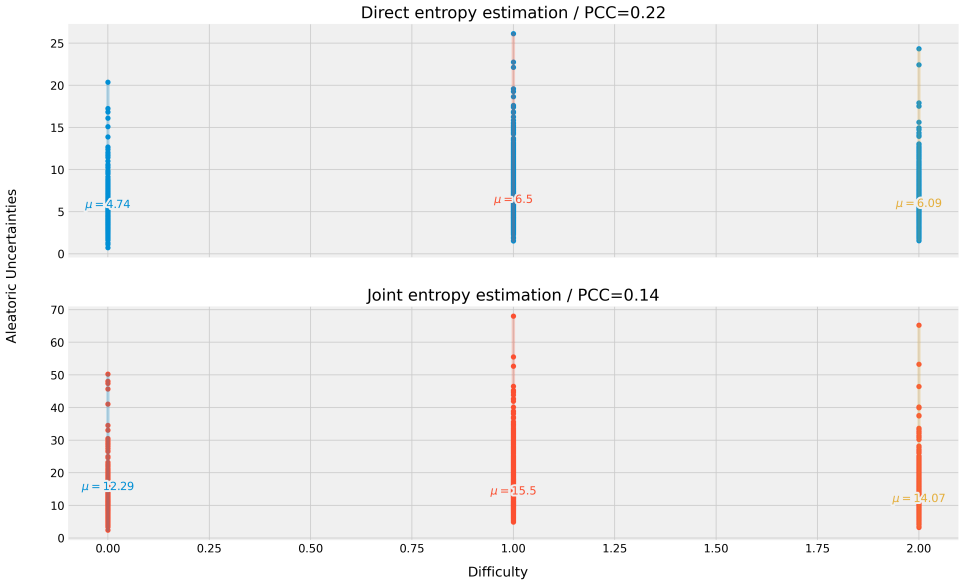


Figure 5.6: Aleatoric uncertainties estimated using both the *direct entropy estimation* and the *joint entropy estimation* models in function of the difficulty attributed in the ground-truth. Sample of 943 detections taken randomly from the test set.

the points remaining as defined as non-occluded objects. Since there are several factors that lead to aleatoric uncertainty in the data, the relationship with occlusion alone is inconclusive on the quality of these aleatoric estimates.

5.5.2.3 Difficulty

Difficulty is a factor present in the ground-truth labels of the KITTI dataset that aims to categorise the hardness of the detection. This categorisation includes results from several other metrics such as the minimum bounding box height, maximum occlusion level and maximum truncation. As such, difficulty levels summarise the conditions surrounding bounding boxes, taking a global look at the factors that lead a detection to be harder for models. Occlusion is also taken into consideration in this metric. The levels assumed for the difficulty are *Easy* (0), *Moderate* (1) and *Hard* (2).

Figure 5.6 showcases the aleatoric uncertainties obtained for each difficulty level. Despite not demonstrating a smooth linear ascent between the considered levels, the PCC scores achieve 0.22 and 0.14 for the *direct entropy estimation* and *joint entropy estimation* models respectively. Observing the averages represented in the plots, there is a clear divergence between easy detections and other levels of difficulty as these sport significantly lower average aleatoric uncertainties. On the contrary, comparing the *Moderate* and *Hard* levels, the first achieves the highest average uncertainty with values 6.5 and 15.5. Whilst on the *direct entropy estimation* the difference between both levels is negligible, from 6.5 to 6.09, in the other method, there is a significant divergence, from 15.5 to 14.07.

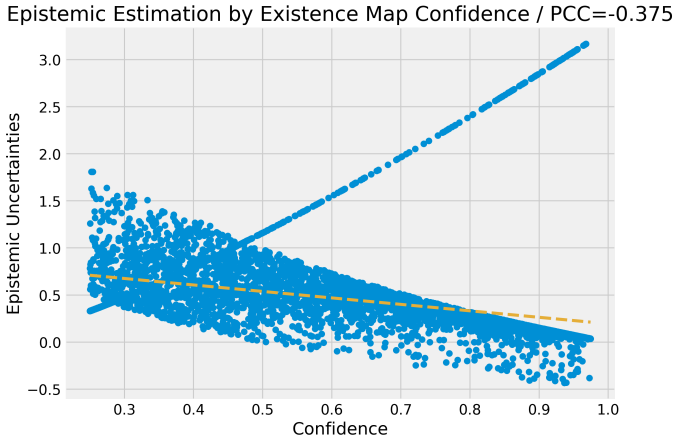


Figure 5.7: Epistemic uncertainties estimated in function of the Existence Map probability. Sample of 943 detections taken randomly from the test set.

Even though this variability leads Pearson's coefficient to indicate *weak* and *very weak* relationships between the variables of interest, the results suggest that at least in easy detections, the aleatoric uncertainty is indisputably lower. Once again, the observations suggest that the *joint entropy estimation* is less attuned to the evaluated factors that may induce variations in the input data. Once more, this may stem from the summation of entropies diluting the aleatoric uncertainty obtained for each anchor.

Despite the analysis of the relationship of the aleatoric uncertainties in function of the occlusion demonstrating a *very weak*, almost no existent correlation, the analysis towards the difficulty showcases diverging outcomes. Since the assignment of a difficulty level to a certain bounding box includes digestion of the occlusion severity, further than considering other aspects, these results expose that the proposed models do in fact tackle randomness arising from low density of points, occlusion and truncation in the least. This showcases that the variations present in the generated input data in the face of the true underlying distribution are rather complex and dependent on a panoply of different factors.

5.5.2.4 Existence Map Probability

The values contained on each cell of the Existence Map represent the probability of an object existing on it. Since this confidence results from the maximum value found in the anchor's classification distribution, this probability expresses the model at some point believed the object could exist. Taking this into consideration, Figure 5.7 showcases the epistemic uncertainties obtained by Existence Map confidence. This result contains all epistemic components and represents a sample of 943 detections taken randomly from the test set.

On this case, a descending tendency can be observed, demonstrating that the estimated epistemic uncertainties generally decrease as the Existence Map increases its confidence. Since the information contained in an Existence Map stems from the region proposal layer directly, high probabilities express the model's certainty on the anchor contents even without further fine-processing.

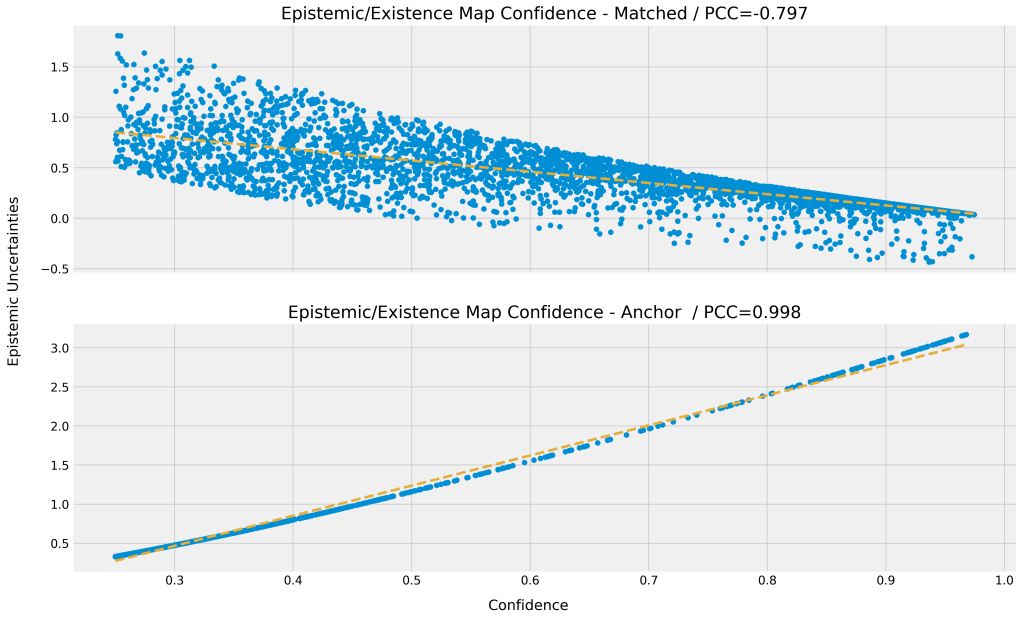


Figure 5.8: Epistemic uncertainties estimated in function of the Existence Map probability over each of the epistemic parcels. Sample of 200 detections taken randomly from the test set.

As such, it would be expected that such detections are less likely to be rejected in later layers, thus diminishing the overall model's uncertainty. Overall, this tendency is confirmed by the Pearson's coefficient that scores the relationship as negative (the uncertainties decrease as the confidence increases) with value -0.375, expressing a borderline *weak* strength correlation inching close to *moderate*.

Once again, it is relevant to observe the different behaviours of each of the components utilised to estimate the epistemic uncertainty. The epistemic uncertainties breakdown by Existence Map probability is shown in Figure 5.8.

Observing the breakdown, it becomes clear that the epistemic uncertainty in reality has a very strong relationship with Existence Map confidence with PCC scores of -0.797 and 0.998 for matching standard detections with Existence Map patches and single anchor distribution support, respectively.

Parcels of the epistemic uncertainty supported only by an anchor distribution tend to increase as Existence Map confidence rises. This is coherent with expectation as these detections are rejected by the standard model despite having an anchor distribution indicating existence, which leads to high uncertainty scenarios. The higher the Existence Map confidence, the higher divergence between the region proposal layer and the output, indicating maximal uncertainty levels. On the other hand, for boxes supported by both the region proposal layer and the output, when confidence in existence rises, it is natural that the epistemic uncertainty decreases as both layers confirm the detection.

Figure 5.9 showcases the component of the epistemic uncertainty when there is no Existence Map support ($a = 0$) over the standard output confidence. As it can be seen, the predicted epistemic

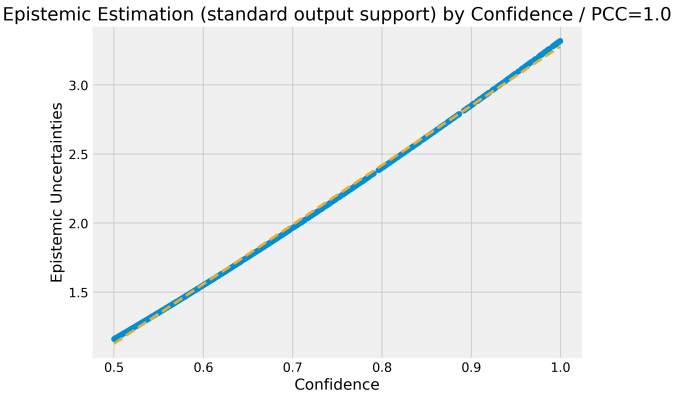


Figure 5.9: Component of the epistemic uncertainty (case $a = 0$) estimated in function of the standard output confidence. Sample of 943 detections taken randomly from the test set.

uncertainty aligns linearly with the confidence, resulting in a PCC of 1.0. This depicts the same scenario verified with the single anchor support plot. As the definition of the distance function weights both scenarios of single support equally, this result is coherent with the expectation.

5.5.2.5 F1-score

In order to guarantee that epistemic uncertainty estimates are *explainable*, it is expected to observe a connection with the accuracy of the model. Namely, when the performance drops, epistemic uncertainty should increase. Figure 5.10 showcases the behaviour of the epistemic estimates in function of the F1-score obtained by considering the standard output, the Existence Map patches and the merged boxes with the Existence Probability.

Upon observation, there is a clear tendency of the epistemic uncertainty to increase as the F1-score decreases, as seen when evaluating for the standard model and Existence Probability outputs. This is confirmed by the Pearson's coefficients of -0.583 and -0.5, respectively, that depict a *moderate* strength relationship, with the standard model inching very close to a *strong*

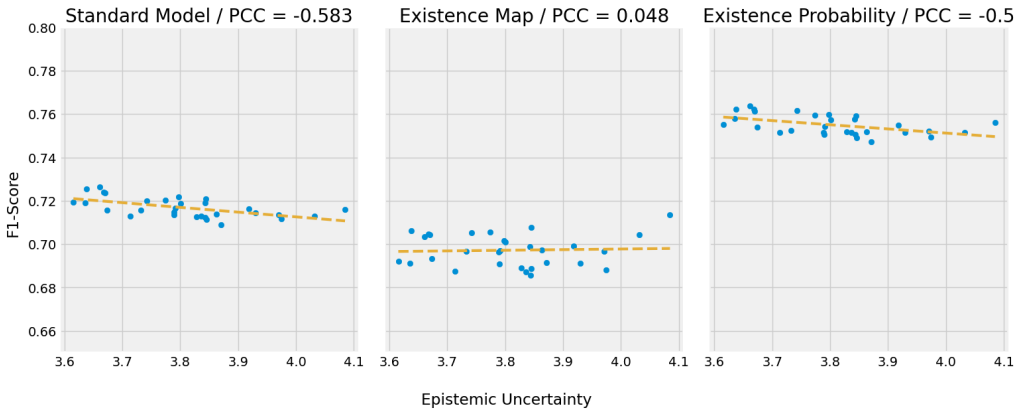


Figure 5.10: Epistemic uncertainties versus the F1-score obtained by the standard model, the Existence Map and the Existence Probability.

correlation. The proposed interpretation of the epistemic uncertainty thus behaves as expected, showcasing explainable and calibrated results.

In the case of the Existence Map, since these are outcomes of the region proposal layer that suggest areas of interest, there is a high rate of false positives. As such, the F1-score features high variability in conjunction with the epistemic uncertainty estimates as disagreement at this level is at a maximum. This is expressed by the inexistent correlation, culminating in a PCC score of 0.048. Since the epistemic estimates depend on the distance to the standard output, this result is expected and coherent since, at this level, there is a disconnect between the region proposal and final layers. Otherwise, according to the proposed model, the distance between both layers would be very small, leading to almost inexistent uncertainty.

5.5.2.6 Out-of-distribution / Perturbation

Since out-of-distribution data typically features unknown patterns and distributions, higher uncertainties may occur in the presence of these types of samples. Further, perturbations induce transformations in the original data, leading to an increased distance to the true underlying distribution that generated the dataset. As such, in perturbed samples, higher aleatoric uncertainty is expected compared to the original validation set. Since the data is also less defined and with higher intrinsic randomness, the divergence within the network increases, affecting the epistemic parcel as well. In this sense, for the evaluation of the estimates provided by the proposed uncertainty interpretations, it is relevant to study the obtained values for each parcel in the considered perturbation test set.

Table 5.6 displays the obtained uncertainties in all perturbations modes over all severity degrees of the KITTI-C dataset [Kong et al. (2023)]. As observed, the estimated aleatoric uncertainties are rather similar despite the severity of perturbation. In some modes, the *heavy* severity leads to the highest aleatoric parcel, however, this tendency is not linear nor replicated through all modes. In particular, with the modes *cross_sensor* and *snow*, the highest aleatoric components are obtained, with a visible increase over severity degrees.

The *crosstalk* perturbation seems to induce the most variability within the network. This is evident when observing the divergence between the estimated aleatoric and epistemic parcels. Despite the *heavy* severity sporting the lowest calculated aleatoric uncertainties, the epistemic component peaks over all severities, notably levelling at nearly the double of estimates obtained in other perturbations. Forthcoming experiments showcase that the region proposal layer is heavily affected by this perturbation, which demonstrates the coherence of the modelled uncertainty that manifests this behaviour.

The lack of a linear correlation is verified also in the epistemic uncertainties as there is no obvious behaviour to the obtained estimates. These results seem to suggest that different perturbations illicit considerably divergent behaviours from the same architecture, a condition connected to underspecification, a phenomenon also verified in Chapter 6. Furthermore, since the VirConv-L model relies on the generation of virtual points, completing the pointcloud with RGB and depth completion information, the architecture proves more resillient to perturbations. Since the virtual

Table 5.6: Uncertainties estimated in each interpretation over all perturbation modes and severity levels. The cells in orange achieve the highest uncertainties whilst the ones in green feature the lowest values estimated.

Modality	Degree	Aleatoric Uncertainty		Epistemic Uncertainty
		direct entropy estimation	*joint entropy estimation*	
beam_missing	light	3.829	13.212	4.878
	moderate	4.089	13.661	4.908
	heavy	3.946	13.401	4.896
cross_sensor	light	4.017	14.035	4.180
	moderate	4.157	14.289	4.132
	heavy	4.366	14.313	4.491
crosstalk	light	3.400	10.812	6.886
	moderate	3.371	10.547	7.460
	heavy	3.348	10.344	8.097
fog	light	3.997	13.321	5.600
	moderate	3.937	13.183	5.193
	heavy	3.902	12.656	5.664
incomplete_echo	light	3.872	13.280	4.886
	moderate	3.937	13.429	4.858
	heavy	4.017	13.587	4.798
motion_blur	light	3.671	12.776	5.022
	moderate	3.662	12.597	5.135
	heavy	3.635	12.427	5.226
snow	light	4.299	13.897	5.875
	moderate	4.531	14.160	6.533
	heavy	4.558	13.823	7.372
wet_ground	light	3.707	12.968	4.948
	moderate	3.692	12.918	4.880
	heavy	3.701	12.942	4.965

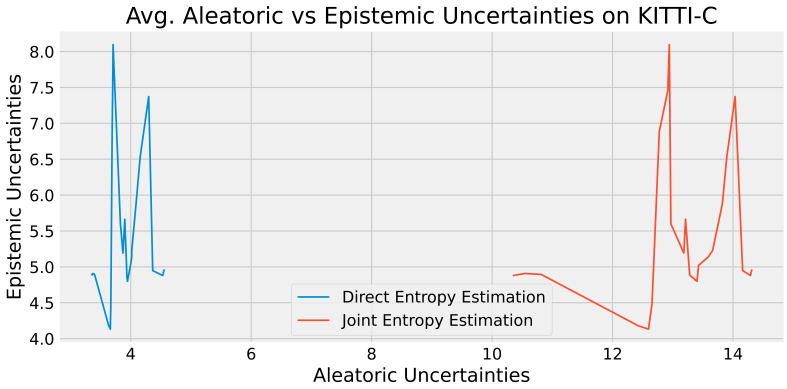


Figure 5.11: Aleatoric versus epistemic uncertainties estimated over all perturbation modes and severities. The blue plot represents the aleatoric uncertainties obtained with the *direct entropy estimation* model and the red one depicts the estimates of the *joint entropy estimation* method.

points gather data from several sensors and sources, the variations in the pointcloud are mitigated considerably. This may lead to the lack of correlation between severity levels and the calculated uncertainties.

Figure 5.11 depicts the relative behaviour of the uncertainty parcels obtained in the aforementioned perturbation tests. It can be observed that, for both models of the aleatoric uncertainty, there are certain levels of intrinsic randomness that lead to spikes in epistemic uncertainty. Despite this, no consistent linear connection is depicted in the presence of high aleatoric estimates. The divergent behaviour showcased by the model depending on the perturbation suggests that predictor disagreement stems from more than modelled aleatoric randomness.

These results seem to suggest that, when the randomness in the data is at low levels, the model is still able to maintain lower epistemic uncertainty. This may originate in the fact that neural networks naturally digest aleatoric uncertainties and generate functions that model around this variability, enabling confident results to still be outputted. However, at higher levels, the digestion of this randomness proves less sufficient, enabling higher divergence in predictor knowledge.

5.5.3 Uncertainty Interpretations Calibration Analysis

In order to assess the performance of the proposed uncertainty interpretations, a comparison of calibration with literature methods is carried out. For this analysis, the selected methodologies are Monte Carlo Dropout [Gal and Ghahramani (2016)], Deep Ensembles [Lakshminarayanan et al. (2017)] and Test-time Data Augmentation [Ayhan and Berens (2018)]. These are chosen amongst prior literature due to their notoriety and broad application. Since these methods are strongly proven and widely adopted, their results are of great importance when studying the effectiveness of the developed interpretations of uncertainty.

For the implementation of MC-Dropout, T is set to 12, amounting to the number of samples collected for the predictive distribution at each data point. Dropout layers are enabled during testing and are mostly employed in the detection heads. For Deep Ensemble evaluation, 6 sub-models

Table 5.7: Calibration errors obtained for the aleatoric uncertainties over the bounding box regression and classification.

	Bounding Box Regression	Classification
MC-Dropout	2.41	0.94
Deep Ensembles	2.33	0.87
Test-time Data Augmentation	2.40	0.94
AU_1	0.33	0.74
AU_2	0.19	0.20

are considered, all resulting from divergent initialisations, namely with changes in *seed*, *batch size* and *retraining checkpoint*. Concerning Test-time Data Augmentation, the tested methods are the insertion of RGB noise, random points dropped/inserted and ground perturbations. For each input sample, 12 variations of these augmentations are performed, varying the controlling parameters such as dropout rate, insertion rate and sigma. As previously denoted in the Evaluation Methods 5.4, a variant of the ECE/ENCE metrics is derived to assess the calibration error, with this value being observed for the bounding box regression, through the inverse of the IoU, and for the classification, with the cross-entropy.

Having obtained a predictive distribution, for all the reference methods, an entropy-based interpretation of uncertainty from the literature is leveraged, proposed by Malinin (2019), enabling the decomposition in both aleatoric and epistemic parcels. In this sense, both uncertainty models share base similarities, allowing a more direct comparison between the methodologies of interest that enable the approximation of a predictive distribution.

Table 5.7 features the errors calculated over the regression and classification for the aleatoric parcel. Both the *direct entropy estimation* (AU_1) and *joint entropy estimation* (AU_2) are featured, also enabling direct comparison between both proposals. As observed, the methods from literature achieve similar calibrations, with Deep Ensembles leading to slightly lower errors on both modalities. For the aleatoric parcel, a strong connection of the obtained uncertainties with Monte Carlo Dropout and Test-time Data Augmentation is reported.

In stark contrast to this, the *direct entropy estimation* model of uncertainty leads to an average -86% error over the bounding box regression. Whilst the difference is less demarcated for the classification assignment, there is still a decrease of -19% over the other methods average. Despite this, the *joint entropy estimation* method achieves significantly lower calibration errors over both regression and classification, surpassing all literature methods by a wide margin and also the *direct entropy estimation*, albeit with a less demarcated difference. These results demonstrate -92% and -79% of calibration error over the average obtained with the literature methods. Through these results, the statistical coherence of the *joint entropy estimation* approach demonstrates its superior correlation with detection performance. Nevertheless, both of the developed proposals feature low calibration errors, over-performing the analysed techniques.

Concerning the epistemic uncertainty, Table 5.8 reports the calibration errors calculated over both regression and classification in the same conditions. Contrary to the previous analysis, the

Table 5.8: Calibration errors obtained for the epistemic uncertainties over the bounding box regression and classification.

	Bounding Box Regression	Classification
MC-Dropout	1.84	0.97
Deep Ensembles	1.63	0.86
Test-time Data Augmentation	1.48	1.17
EU	1.27	0.69

literature techniques showcase more divergent results. With Test-time Data Augmentation leading to the lowest error in a regression target, for classification, this proves to be the most miscalibrated method, with Deep Ensembles once again achieving the best result overall. Despite the more variable results, the proposal of the epistemic uncertainty presented in Chapter 4 achieves significantly lower calibration errors over both assignments with -23% for regression and -31% for classification over the literature average.

Overall, these outcomes suggest that the proposed interpretations of uncertainty achieve strong calibration, with errors significantly lower than the established methods from the literature that were tested. Since the selected approaches are some of the most widely accepted and employed techniques in the area of uncertainty quantification, these results demonstrate the coherence and importance of the developed models.

5.6 Experimental Evaluation of Existence and Uncertainty Maps

Having studied the behaviour of the estimated uncertainties in each proposed interpretation, it is then of interest to evaluate the applications discussed in Chapter 4. In this sense, this section reports the performed experiments concerning the evaluation of Existence and Uncertainty Maps.

As a point of reference, results for the standard VirConv-L model as a classification baseline are presented in Table 5.9. Over several experiments, the best obtained model achieves 0.65 precision and 0.91 recall for the detection of cars, pedestrians, and cyclists in the test set described in Section 5.3.

Since the aim of the presented concept of existence is the detection of objects regardless of class, all available labels in the dataset are utilised for the following experiments, including the classes: *Car*, *Pedestrian*, *Cyclist*, *Tram*, *Truck*, *Van*, *People sitting*, and *Misc*. This applies to both existence and uncertainty maps.

Table 5.9: VirConv-L baseline evaluation on the classification of classes *Car*, *Pedestrian* and *Cyclist* on the test set.

Model	TP	FP	FN	Precision	Recall
VirConv-L (closed-set baseline)	5979	8793	1579	0.65	0.91

Table 5.10: Number of total ground-truth detections considered for each class.

Class	# Ground-truth detections	% of set
Car	14385	70.6%
Pedestrian	2280	11.2%
Cyclist	893	4.3%
Van	1617	7.9%
Truck	606	3.0%
Tram	287	1.4%
Person sitting	166	0.8%
Misc	131	0.6%

For reference, Table 5.10 demonstrates the number of ground-truth objects considered for each class. It is of note that there is high imbalance in the classes present in the test set as expected, with the vast majority of objects belonging to class *Car*, 70.6%.

5.6.1 Parameter Fine-tuning

To determine the optimal conditions for Existence Map generation, analytical experiments using different grid precision values, τ, are conducted.

For the parameter calibration experiments, the following thresholds for Existence Map confidence are used when plotting precision-recall curves: $[0, 0.1, 0.25, 0.35, 0.5, 0.7, 0.8, 0.9, 1]$. Cells with probabilities below the set threshold are zeroed out.

Figure 5.12 showcases the precision-recall curves obtained by varying the grid precision, τ. As it can be seen, the grid precision has a perceivable impact on precision and recall. For the lower recall values, finer grids lead to improved detection precision. As recall gets higher, more proposals are discarded due to the threshold probability being very high which leads to considerably lower precision. Following the decreasing trend of the recall, the grid precision has less and

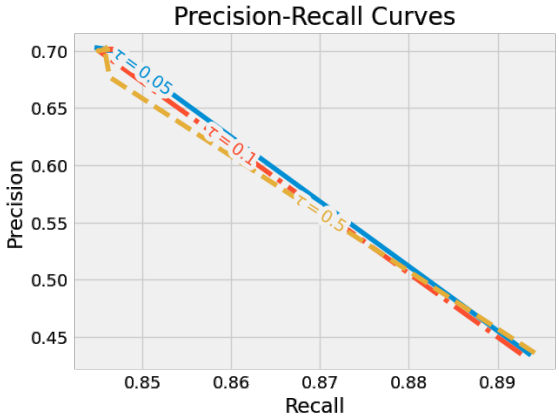


Figure 5.12: Precision-recall curves obtained with different grid precisions ($\tau \in \{0.05, 0.1, 0.5\}$).

Table 5.11: Results obtained with the standard output versus using existence maps when evaluating for all classes in the KITTI dataset.

Output	TP	FP	FN	Precision	Recall
Standard	15831	7535	5039	0.68	0.76
Existence Map	15813	8934	5057	0.64	0.76

less impact in end results which stems from precision not providing a substantial advantage when most proposals are rejected.

While slight variations in results can be observed across different τ values, this parameter appears to have a less significant impact on the final outcomes compared to its effect on computation speed. From these conclusions, the apparent suggestion is to use a grid precision of 10 cm, $\tau = 0.1$, to speed-up large scale evaluations and a precision of 5 cm, $\tau = 0.05$, when the generation of more fine-grained maps is necessary.

On most forthcoming experiments, τ equal to 0.1 is selected since the evaluation is performed over a large collection of data. For deployment, especially when evaluating a single sample, a precision of 5 cm, $\tau = 0.05$, provides the best results at a moderate computational cost.

5.6.2 Existence Map Performance

The final results obtained by the best calibrated Existence Map are showcased in Table 5.11 in comparison with the standard output of the model. This evaluation covers all classes available in the KITTI dataset. As the standard model only classifies cars, pedestrians, and cyclists, its recall drops significantly compared to the baseline when evaluated against all classes of the dataset. Leveraging only the Existence Map, the same recall as the standard model is achieved. However, this is not replicated in the precision that drops from 0.68 in the standard model to 0.64 using the Existence Map. Despite the lower precision, considering that the standard output uses further fine-grained information in order to produce its detections, the result obtained by existence maps as a standalone tool is impressive. This demonstrates the effectiveness of the conceptualisation of existence in indicating areas with the possibility of containing objects.

As demonstrated in forthcoming experiments, existence maps many times reject false positives of the model and suggest the existence of false negatives, which leads to improved metrics on some samples. This demonstrates the ability of the region proposal layer of the network to pinpoint areas of interest, despite a specialised training for a closed classification set.

Despite this, the lower precision is justified by the overall high amount of false positives, meaning that the maps also suggest areas that do not contain labelled objects. This, once again, is natural considering the incomplete information used by existence maps.

The results obtained by the standard model also suggest its ability to discern objects from classes further than those it was trained for. Even with the training targets including only cars, pedestrians and cyclists, the model naturally seems to digest and learn the representations of other types of objects.

Table 5.12: Number of detections made by the model leveraging existence maps versus the number of ground-truth objects by class. The rows in orange and green achieve the maximum and minimum amount of correct detections per ground-truth samples respectively.

Class	# detections	# ground-truth	%
Car	12919	14385	89.8
Pedestrian	1291	2280	56.6
Cyclist	596	893	66.7
Truck	63	606	10.4
Van	698	1617	43.2
Tram	61	287	21.3
Person sitting	33	166	19.9
Misc	152	636	23.9

In this sense, a more in-depth analysis reveals information of interest. Table 5.12 presents the number of correct suggestions made using the existence maps compared to the ground-truth labels in the test set, broken down by class. The model shows the highest detection accuracy for the *Car* class, while *Truck* is the least recognised. As anticipated, besides the classes the model is trained to detect (*Car*, *Pedestrian*, and *Cyclist*), there is a significant recognition rate for the *Van* class. This is may be based on the fact that vans share many features with cars, making the model more adept at detecting them. Even though distinguishing between the two may pose challenges for a classifier, in the broader notion of object, the similarity of features between the objects often found on the road is advantageous.

From these findings, it becomes clear that a model trained on just three specific classes is capable of generalising to other object categories. This process mirrors human reasoning, where new objects are often recognised by identifying similarities to known concepts, enabling their classification as objects of interest.

Further than this, these results demonstrate that the backbone and region proposal modules of the network have learned a suitable representation of the objects, having a good grasp of their defining features. Using the knowledge contained within this representation, areas of interest may be suggested for further refinement or for downstreaming into other tasks. The availability of this information may prove advantageous in other modules of the autonomous pipeline, leading to a possibility of recovering from errors made in prior steps.

From these results alone, it may be suggested that existence maps only provide a less accurate indication of the areas already signalled by the standard model. As further studies on the Existence Map are conducted in forthcoming sections, it becomes apparent that the areas suggested by the maps are not always coincident with the standard output bounding boxes. This relation manifests in two diverging situations: when the model misses an object that is considered in the Existence Map and when the model detects an object not present in the ground-truth and the Existence Map rejects its existence. As such, the information provided by the Existence Map adds to the outcomes produced by the standard model.

5.6.3 Out-of-distribution / Perturbation Performance

The previous subsection analysed the performance of the Existence Map in a test set taken from the training distribution. This demonstrates the effectiveness of the maps in suggesting areas of interest for further analysis on object existence. The ability of the standard model to digest a wide variety of object classes further from the training closed set is also established.

The reported performance in in-domain data depicts only the effectiveness of the models and tools to recognise patterns that these are trained to directly discern. As such, in order to provide a complete evaluation of the performance of both the model and the Existence Map as a stand-alone tool, it is indispensable to consider out-of-distribution data. As mentioned in Subsection 5.3.3, perturbation data is sourced from KITTI-C [Kong et al. (2023)].

Table 5.13 demonstrates the precision, recall and F1-scores obtained by the VirConv-L model either in its standard form or considering only the information given by existence maps over all the perturbation modes and severity degrees found in the KITTI-C dataset.

Replicating the tendency observed in the previous subsection, the standard model achieves higher precision and F1 scores over all perturbations and severities. Despite this, the Existence Map achieves higher recall on most modes, namely *beam_missing*, *cross_sensor*, *crosstalk*, *fog* and *incomplete_echo*. This demonstrates that, in these perturbation categories, existence maps make significantly less false negatives than the standard model.

The lower precision of existence maps stems from the increased difficulty in the digestion of the pointcloud with perturbations. This configuration of the data leads the region proposal layer to suggest significantly more patches than in the standard test set, several of which are false positives. Despite this, observing the results obtained in the recall metric demonstrates that the Existence Map also identifies many patches with ground-truth objects that are missed by the standard model. In this sense, the objective of the Existence Map is fulfilled as it provides useful information that is lacked by the model.

Another observation of interest is the performance of the Existence Map in the *crosstalk*, *fog* and *snow* modes. Even though the maps' overall precision is similar to the one achieved by the standard model, in these modes, the obtained values drop considerably. The lowest precision and F1-scores stand at 0.320 and 0.451 respectively on the *crosstalk* mode with *heavy* severity. This seems to suggest that, even though the model still achieves reasonable results, the disagreement between the region proposal layer and the final output is severe, indicating higher epistemic uncertainty. This is confirmed by the epistemic and aleatoric uncertainties estimated as seen in Table 5.6, especially in the case of *crosstalk*.

From this, it is reasonable to suggest that the VirConv-L predictor has acquired biases through its learning process that are not advantageous in severe weather conditions such as fog or snow and in highly aleatoric uncertainty scenarios generated by the presence of crosstalk. From these results, it becomes evident that in the presence of the phenomenon of crosstalk, a common occurrence in LiDAR sensors, additional safety measures must be employed to ensure the accuracy of the object detection. In this case, the Existence Map proves to be an important tool, especially since its recall

Table 5.13: Precision, recall and F1-scores obtained for each perturbation mode over all severity degrees with the standard output versus the Existence Map. Orange background represents the best metric performance in each row. Green cells highlight some of the lowest values obtained in some metrics. Despite the entirety of F1 results showcased by existence maps being lower than the standard output, in most perturbation modes and degrees of severity, the Existence Map achieves higher recall.

		Standard			Existence Map		
Modality	Degree	Precision	Recall	F1	Precision	Recall	F1
beam_missing	light	0.696	0.745	0.720	0.660	0.746	0.700
	moderate	0.699	0.728	0.713	0.651	0.738	0.691
	heavy	0.698	0.698	0.698	0.640	0.724	0.679
cross_sensor	light	0.741	0.724	0.732	0.711	0.726	0.718
	moderate	0.747	0.706	0.726	0.706	0.717	0.712
	heavy	0.705	0.660	0.682	0.641	0.702	0.670
crosstalk	light	0.686	0.754	0.718	0.383	0.760	0.509
	moderate	0.682	0.751	0.715	0.347	0.763	0.477
	heavy	0.674	0.750	0.710	0.320	0.764	0.451
fog	light	0.681	0.744	0.711	0.603	0.750	0.669
	moderate	0.696	0.730	0.713	0.624	0.742	0.678
	heavy	0.673	0.711	0.692	0.534	0.735	0.618
incomplete_echo	light	0.685	0.724	0.704	0.661	0.734	0.695
	moderate	0.680	0.710	0.695	0.656	0.725	0.689
	heavy	0.672	0.676	0.674	0.650	0.709	0.678
motion_blur	light	0.689	0.759	0.723	0.660	0.749	0.701
	moderate	0.683	0.755	0.717	0.641	0.751	0.692
	heavy	0.675	0.754	0.712	0.630	0.748	0.684
snow	light	0.676	0.721	0.697	0.563	0.716	0.631
	moderate	0.669	0.716	0.692	0.525	0.709	0.603
	heavy	0.655	0.708	0.680	0.460	0.699	0.555
wet_ground	light	0.694	0.760	0.726	0.667	0.752	0.707
	moderate	0.695	0.760	0.726	0.666	0.750	0.705
	heavy	0.693	0.761	0.725	0.663	0.752	0.705

Figure 5.13: Input samples with examples of objects identified by the existence maps where the model failed to detect an object, despite the ground-truth confirming its presence.

overcomes that of the standard model in these conditions, suggesting areas to be further verified for object existence.

5.6.4 Interpretability and Existence Maps

After analysing the quantitative results, it is also valuable to examine examples where existence maps offer additional information about the presence of objects that may not be captured by the model's standard output. The showcased samples demonstrate the increased interpretability of the object detection model when supported by the Existence Map.

All subsequent figures follow a consistent colour scheme, with the colour map of the existence maps ranging from green to orange, where green represents the lowest probability and orange the highest, akin to temperature. The colour map depicts probabilities starting from 0.25, as all cells with lower confidences are discarded. The model's standard detections are highlighted with dark orange bounding boxes, while the ground-truth annotations from the KITTI dataset are shown in blue.

As discussed in the forthcoming subsections, existence maps may enable the identification of false negatives, false positives and incorrect heading regressions made by the standard model.

5.6.4.1 False Negatives

Figure 5.13 presents a collection of samples where the object detection model failed to detect a real object, which is accurately identified in the Existence Map, as confirmed by the ground-truth. A strong tendency for existence maps to identify objects missed or discarded by the model during post-processing is observable, with many examples of this behaviour evident in the test set. This is coherent with the results obtained in prior experiments, where the recall achieved by existence maps is consistently high.

The provided samples show that the existence probability for these missed objects is relatively low (around 0.3 in the two leftmost samples), which suggests that the model may have discarded them due to low classification confidence. However, with the leverage of existence maps, these

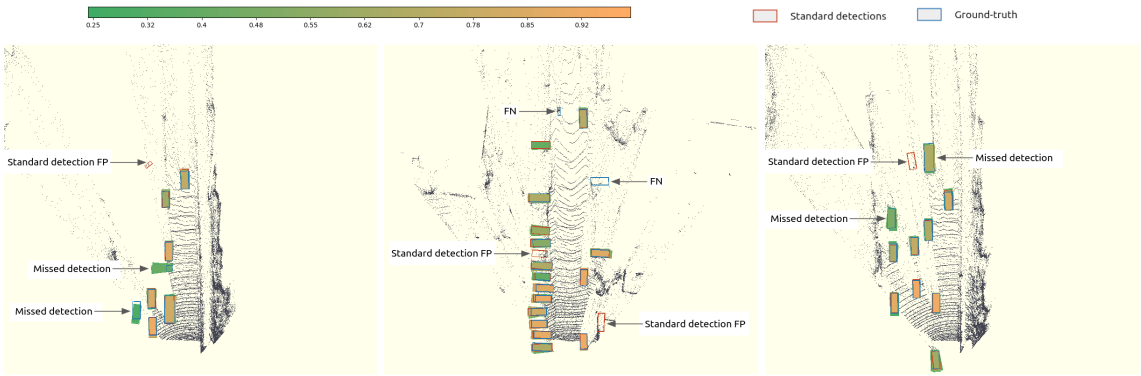


Figure 5.14: Input samples where the existence map correctly rejects false positives produced by the standard model.

areas containing objects can be recovered and further analysed by additional models or safety mechanisms. As shown, missed objects are found not only at considerable distances from the ego vehicle but also in close proximity, which could pose a safety risk to passengers.

In the final example on the right, the model misses a tram, as expected, since it is not a classification target. The existence map, however, highlights the possible presence of an object with moderate confidence, around 0.5. This illustrates the versatility of existence maps, as their broad object concept allows them to signal areas of interest in the input, independent of the specific classes the model is trained to recognise.

5.6.4.2 False Positives

Existence maps may also enable the detection of false positives made by the model. Figure 5.14 showcases examples where in several areas the existence map outputs that no object exists even though the standard output of the model identifies objects.

The samples on the left and right demonstrate false positives produced by the model despite the rejection of the presence of an object in the existence map. The middle example, further than showcasing false positives, also demonstrates objects that were missed by both the model and the Existence Map.

The presence of false positives on the standard output of the model without support from the existence map is harder to interpret. Since sometimes noise in the pointcloud or similar spatial structure leads to features that do not exist being recognised by the model, it is natural that all models will show some degree of falsely made detections. Yet, since the existence maps rely on anchor proposals, which lead to the final detections after further fine processing, it would be also natural to see support from the existence map in all detections. As such, it is believed that these cases, which occur a large number of times in the test data set, stem from high epistemic uncertainty present in the model [He and Jiang (2023)].

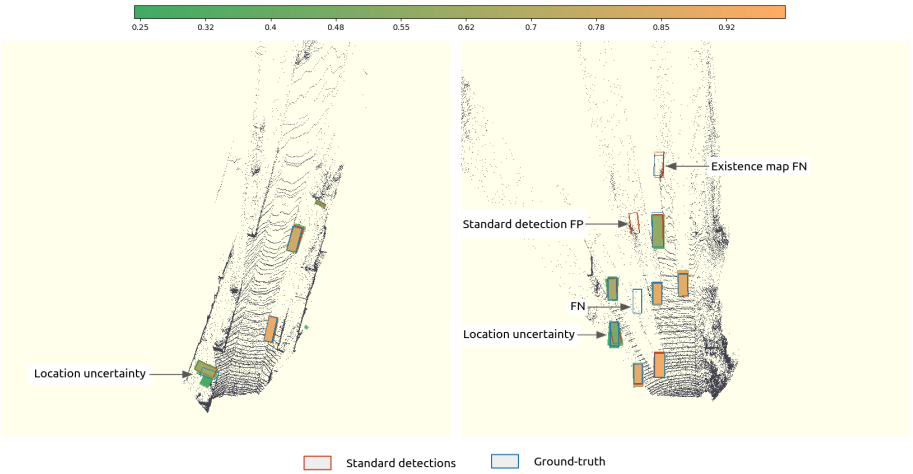


Figure 5.15: Examples of samples where existence maps indicate location uncertainty, as seen by the different heading regressed by the model and the ground truth.

5.6.4.3 Location Uncertainty

Additionally, existence maps can offer insights into location uncertainty. Figure 5.15 shows examples where the existence maps reveal uncertainty when regressing the bounding box, resulting in a significant error in the predicted heading compared to the ground-truth.

As observable in the leftmost sample, there is a detection with low confidence that exhibits a significantly different heading from the ground-truth. The model's uncertainty regarding the heading of this prediction is reflected in the Existence Map, which covers a large area around the regressed bounding box with varying probabilities and headings.

The right sample illustrates a similar situation, with the Existence Map also extending over a wide area around the bounding box of the detection with incorrect heading. In this example, a false positive of the standard model is observable, along with a false negative of the existence map and a false negative from both.

5.6.5 Interpretability and Uncertainty Maps

Having analysed the interpretability improvements produced by existence maps, a similar qualitative report on the Uncertainty Map ensues. Similarly, the focus is to demonstrate the additional information introduced by the Uncertainty Map on the model's performance and knowledge.

Pertaining to the aforementioned figure format, the provided examples also showcase a consistent colour scheme with the colour map of the existence maps varying from green (less probable) to orange (more probable), akin to a temperature scale. Uncertainty maps follow a standard thermic colour scheme, ranging from cold purple to warm yellow. Both representations diverge in colorisation so as to enable an easier distinction between the two maps. Existence Map probability ranges from $[0.25, 1]$ as explained previously with the standard model detections highlighted in dark orange bounding boxes and the ground-truth targets delimited in blue.

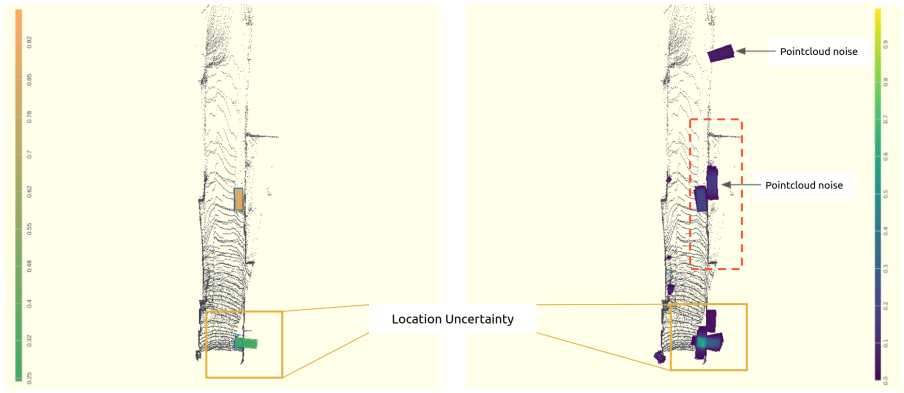


Figure 5.16: Uncertainty in the location regression, as seen by the incorrectly proposed heading, is demonstrated by the concentration of aleatoric uncertainty in the area.

Figure 5.16 depicts the Existence Map, on the left, coupled with its corresponding Uncertainty Map, on the right for a given input sample. In this particular example, uncertainty in the location regression is communicated by both the existence and uncertainty maps. Whilst the standard model misses the object entirely, the Existence Map suggests the presence of an object with a low probability. Despite this, the heading is considerably incorrect. The Uncertainty Map demonstrates the decision process carried by the model. Many anchor distributions span in the area, with diverging orientations and placings, leading to depiction of variable aleatoric uncertainties in the surrounding space. The heading uncertainty is clearly communicated by the heavy estimates for both considered anchor headings.

In this sample, pointcloud noise and randomness can also be extensively observed in two particular locations. The Uncertainty Map, picking up on these conditions, outputs aleatoric estimates surrounding these areas. In the case of the correct detection, outputted with high confidence and support from both the region proposal and final layers, low values of uncertainty are depicted as seen by the purplish, blueish colourisation (within the red dashed box shown in the Figure). The noisy points in the area of this detection, possibly leading up to a sidewalk are successfully represented in the aleatoric estimation despite the closeness of a correct detection.

Figure 5.17 represents another example with the relevant existence and uncertainty maps. As observable, noise and randomness produced by the shadow of an object are also detected by the Uncertainty Map, shown by the high aleatoric estimates as well as the wide uncertainty area, dragged further than the bounding box.

Further than this, the sample depicts three missed detections by both the standard output and the Existence Map, even though the ground-truth includes the respective targets. Despite the lack of information provided by the Existence Map, the Uncertainty Map suggests the possible existence of two objects with lower aleatoric uncertainty values. These particular detections probably feature a low probability, under 0.25, thus being excluded from the Existence Map. This threshold of the confidence allows the reduction of false positives made by these maps being necessary to further fine grain areas of interest. Nevertheless, combining this information with the Uncertainty Map can enable the recovering of further suggestions, possibly leading to improved performance.

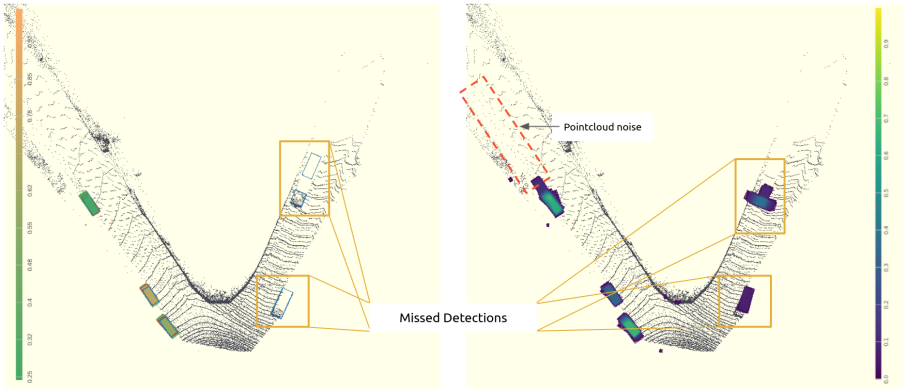


Figure 5.17: Ground-truth objects are missed despite the lower aleatoric uncertainty as seen on the Uncertainty Map.

Even with no merging with the output, the results showcase by the Uncertainty Map may be of interest to practitioners and scientists alike, as well as another safety measure to ensure trust from end-users.

5.7 Experimental Evaluation of the Existence Probability

The final focus of the evaluation procedure is the Existence Probability and its merging strategy that allows the distillation of better calibrated bounding boxes that integrate Existence Map information. In this application of the existence concept and of the uncertainty interpretations, it is firstly of interest to study the parameters that can control its outcomes. The experiments carried out in this domain are thoroughly reported in order to facilitate the choice of the best values.

Further than this, for the validation of the performance of the Existence Probability, the same process as previously reported is followed. The precision, recall and F1-scores achieved when distilling the bounding boxes are reported for both the standard test set and in perturbation mode so as to demonstrate the differences in the outcomes produced by the merging strategy.

Since the final confidence of each detection is updated through this method, the calibration before and after its employment are also studied with the assistance of reliability diagrams. Finally, the interpretability provided by the Existence Probability is qualitatively assessed so that practical examples of its use are communicated.

5.7.1 Parameter Fine-tuning

In order to evaluate the Existence Probability, a collection of experiments with discount factors γ varying from 0.15 to 0.7 and probability thresholds from 0.15 to 0.5 are performed. Table 5.14 demonstrates a selection of the obtained results. It is found that a γ value of 0.5 with a probability threshold of 0.25 provides the best balance of true positives, false positives and false negatives, thus enabling the best relation between precision and recall.

Table 5.14: Study of the true positives, false positives and false negatives obtained with varying γ and probability threshold values. The row in orange represents the best balance between true positives, false positives and false negatives.

γ	TP	FP	FN	Probability Threshold
0.15	17433	7437	5645	0.25
0.35	17402	7412	5676	0.25
0.5	17418	7383	5660	0.25
0.7	17398	7466	5680	0.25
0.35	17476	7659	5602	0.15
0.5	17462	7707	5616	0.15
0.35	17367	7470	5711	0.35
0.35	17398	7417	5680	0.5

From the showcased values, it is also observable that, for a low probability threshold of 0.15, a high rate of true positives is enabled. Nevertheless, in this scenario, the amount of false positives is also the highest, independently of the discount factor. This behaviour is coherent with expectation as the assumed acceptance threshold is rather low. Further, in this case, it is suggested from the results that a larger discount factor worsens results. On the other hand, for higher probability thresholds, such as 0.35 and 0.5, the amount of true positives decreases slightly whilst having a more prominent impact on false positives. For a fixed γ factor, the higher probability threshold enables the best results over all metrics.

Even though these findings showcase some relationship between both parameters, it is, however, difficult to deduce a linear correlation between the discount factor and the probability threshold. It is suggested by the results that the main factor that contributes to diverging outcomes is the threshold of probability. Even though that is the case, for the same probability, different γ factors may enable better results. As such, it is of importance to deduce the best combination of values for these parameters after experimentation.

Taking into consideration this discussion, for forthcoming experiments, the selected parameters are a probability threshold of 0.25 coupled with a γ factor of 0.5.

5.7.2 Existence Probability Performance

The assessment of the performance of the Existence Probability and the proposed merging strategy is carried out in similar conditions as the Existence Map evaluations. As such, the precision, recall and F1-scores for the standard test set are reported and analysed.

In this sense, Table 5.15 features the obtained values for the metrics of interest of all the proposed methods. Merging the standard output bounding boxes with the patches suggested by existence maps leads to improved performance on all metrics. In fact, the Existence Probability achieves a precision of 0.71 with a recall of 0.80, which culminates in an F1-score of 0.75 on the detection of all classes represented in the KITTI dataset.

Table 5.15: Results obtained with the standard output versus the Existence Map and the Existence Probability when evaluating for all classes in the KITTI test set.

Output	TP	FP	FN	Precision	Recall	F1
Standard	15831	7535	5039	0.68	0.76	0.72
Existence Map	15813	8934	5057	0.64	0.76	0.69
Existence Probability	16688	6715	4182	0.71	0.80	0.75

From these results, it becomes apparent that the information provided by the Existence Map contains the suggestion of diverging areas than those of the standard output. When leveraged as a stand-alone tool, the maps achieve lower precision due to the high number of false positives. Employing the proposed merging strategy enables an increase in the true positives whilst significantly decreasing the false positives and negatives.

After intent observation in many samples, it is evident that in high aleatoric uncertainty areas, many patches concentrate closely in space. In the case of uncertainty in the location regression, often times many patches are defined over the same area with different orientations, leading the false positives rate of the Existence Map to rise considerably. In these cases, the merging strategy removes these duplicates, thus leading to reduced false positives. Moreover, leveraging the suggestions of missed detections enables the model equipped with the Existence Probability to increase significantly the rate of true positives, directly decreasing the false negatives rate.

From these results, the potential of existence maps can be denoted. Even more, these outcomes also suggest the high performance of the proposed merging strategy. Both the Existence Map and the Existence Probability can be generated in parallel to any anchor-based object detection model and without the need to retrain, demonstrating the flexibility of these methods.

5.7.3 Out-of-distribution / Perturbation Performance

Following the analysis of the improvements in model performance on the standard test set achieved through the proposed merging strategy, the evaluation of robustness under perturbations is conducted. For these experiments, the procedure is consistent with what was previously reported, with the evaluation using the same predictor on the KITTI-C dataset [Kong et al. (2023)].

Table 5.16 demonstrates the results obtained by merging bounding boxes over all perturbation modes and severity levels versus the standard output. Observing each metric, the merging strategy showcases consistently and substantially improved values. The achieved precision is on average 3.5 points higher than the standard output. The recall, similarly, shows an increase of 3.9 points on average. The summarisation metric, namely the F1-score, is also 3.7 points higher on average.

Taking these metrics into consideration, the performance of the merging strategy in perturbation mode surpasses that obtained in the standard test set where an increase of 3 points is achieved. This once more showcases the potential of integrating Existence Map feedback into the standard output of the model.

Table 5.16: Precision, recall and F1-scores obtained for each perturbation mode over all severity degrees with the standard output versus using the merging strategy. Orange background represents the best metric performance in each row. The merging leads to staggering improved results over the standard output, obtaining the highest score in each row over all metrics.

		Standard			Merged Standard with Existence Map		
Modality	Degree	Precision	Recall	F1	Precision	Recall	F1
beam_missing	light	0.696	0.745	0.720	0.734	0.786	0.759
	moderate	0.699	0.728	0.713	0.738	0.770	0.754
	heavy	0.698	0.698	0.698	0.737	0.737	0.737
cross_sensor	light	0.741	0.724	0.732	0.778	0.762	0.770
	moderate	0.747	0.706	0.726	0.787	0.745	0.765
	heavy	0.705	0.660	0.682	0.745	0.699	0.721
crosstalk	light	0.686	0.754	0.718	0.712	0.791	0.750
	moderate	0.682	0.751	0.715	0.706	0.788	0.745
	heavy	0.674	0.750	0.710	0.698	0.785	0.739
fog	light	0.681	0.744	0.711	0.718	0.786	0.750
	moderate	0.696	0.730	0.713	0.735	0.772	0.753
	heavy	0.673	0.711	0.692	0.710	0.751	0.730
incomplete_echo	light	0.685	0.724	0.704	0.722	0.765	0.743
	moderate	0.680	0.710	0.695	0.719	0.752	0.735
	heavy	0.672	0.676	0.674	0.711	0.717	0.714
motion_blur	light	0.689	0.759	0.723	0.725	0.800	0.760
	moderate	0.683	0.755	0.717	0.717	0.794	0.753
	heavy	0.675	0.754	0.712	0.709	0.793	0.748
snow	light	0.676	0.721	0.697	0.707	0.757	0.731
	moderate	0.669	0.716	0.692	0.699	0.751	0.724
	heavy	0.655	0.708	0.680	0.683	0.743	0.711
wet_ground	light	0.694	0.760	0.726	0.730	0.801	0.764
	moderate	0.695	0.760	0.726	0.731	0.800	0.764
	heavy	0.693	0.761	0.725	0.730	0.802	0.764

Table 5.17: Correlation matrix of the variables used in the calculation of the Existence Probability.

	e_Π	o_Π	ρ_Π	$H_\Pi / \max_\Pi(H)$
e_Π	1.0			
o_Π	0.6239	1.0		
ρ_Π	0.5769	-0.0630	1.0	
$H_\Pi / \max_\Pi(H)$	0.1656	-0.1963	0.1418	1.0

Further than this, observing the standard output metrics in Table 5.16, and considering the aforementioned drop in performance of the Existence Map in the perturbation modes of *crosstalk*, *fog* and *snow* previously reported, the merging enables the improvement of the metrics considerably, achieving values in a similar range as the other perturbation modes. This showcases that the biases expressed by the standard model in out-of-distribution samples can be rectified by leveraging knowledge encoded within the network itself, namely the region proposal layer.

5.7.4 Existence Probability Calibration Assessment

Since the aim of the Existence Probability is to provide a more statistically calibrated measure of the existence of objects, including the classification outputs, the relationship between the main parameters used in Equation 4.31 is of interest to be analysed. The first step in this evaluation is the generation of a correlation matrix, shown in Table 5.17.

As expected, the highest correlation is found between the Existence Probability and the object probability, followed by the density of the patch. The entropy, on the other hand, also has a positive impact in the Existence Probability, however, its significance is reduced purposefully, with the correlation factor equalling only 0.1656. The aim is to only discount confidences that correspond with high uncertainties, regardless of the object probability value. This leads to the metric entropy being less impactful in the range of existence probabilities.

More interestingly, a negative correlation between the object probability and both the density and entropy in the patch is observed. This relationship with the uncertainty demonstrates a lack of calibration of the standard model, a characteristic further verified in forthcoming experiments. Moreover, the fact that the object probability is so disconnected from the density in the patch reveals that the classification outcomes do not fully utilise the knowledge contained within the network. If more proposals with probabilities over a threshold are attributed to an area, this space of the input sample should have a higher likelihood of containing objects, as expressed by the density. This would be expected to be linearly correlated with the attributed confidence, or in this case, the object probability.

Diving deeper into the disconnect of the object probability with the entropy in the patch, data is collected to evaluate calibration. Figure 5.18 features a reliability diagram as well as the calibration plot obtained with the standard VirConv-L model. It can be seen that the model is largely overconfident, with its predictions all scoring high confidences, even though its accuracy is considerably lower. Perfect calibration is achieved when outputs with a confidence of 0.5 are correct in half of the evaluated samples, namely obtaining 0.5 accuracy [Guo et al. (2017)].

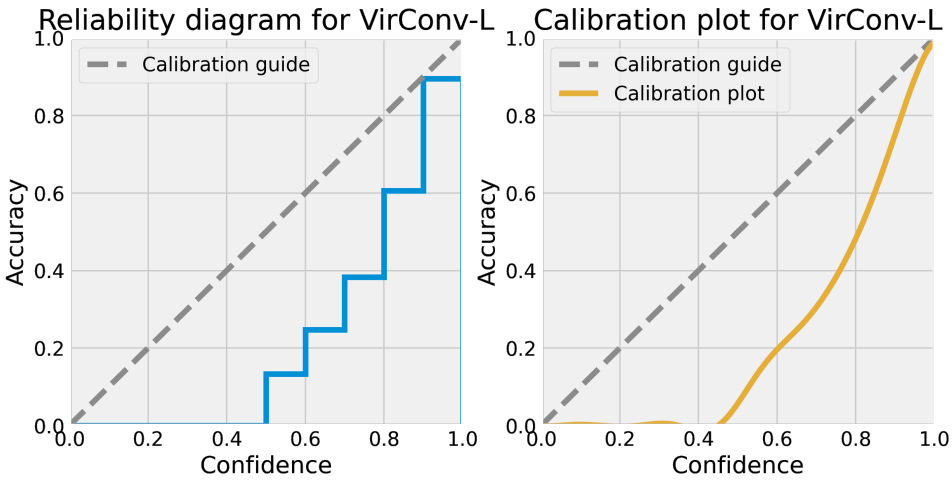


Figure 5.18: Reliability diagram of the VirConv-L model, shown on the left. The corresponding calibration plot is on the right, demonstrating the overconfidence of the model. The perfect calibrated model is shown in grey as a guide.

From these plots, it is denoted that the model makes no detections with confidences below 0.5. Even more, its overconfidence is more demarcated in the 0.5 to 0.8 range, where distance to the calibration guide is maximal. As the output probability grows larger and close to 1, so does the accuracy, diminishing the overconfidence featured by the model in this range.

Having established the lack of calibration of the standard model, experiments with the confidence of detections calculated through the Existence Probability method are carried out. Figure 5.19 shows the reliability diagrams for both the standard model output and the model incorporating the merged confidences. In grey is represented the histogram that would be obtained with a perfectly calibrated model, alongside its trend dashed line. The mean absolute error of the standard histogram equals 729.6 whilst with the Existence Probability, the MAE drops to 640.2, showcasing an improvement of 89.4 points.

Since detections with a probability lower than 0.25 are discarded, the calibration of the Existence Probability cannot align with the guide below this threshold. However, the diagrams illustrate that, overall, the proposed method for distilling and calibrating detections leads to improved calibration. Notably, in the final bin, the model coincides with the calibrated histogram. Despite this, the model still tends to be overconfident, a concerning issue in safety-critical applications like autonomous driving.

In the 0.2 to the 0.4 range, the Existence Probability aligns the histogram close to the perfect calibration. The $0.3 - 0.4$ bin even showcases a slight tendency to underconfidence. The bins in the $0.5 - 0.8$ span are the ones where calibration is further from the target. From 0.8 to 0.9, even though there is still a considerable distance to the goal histogram, this difference is not as demarcated. As mentioned previously, in the $0.9 - 1$ bin, the Existence Probability leads the model to perfect calibration.

This may suggest that the implemented discount using the entropy is not sufficient to cover

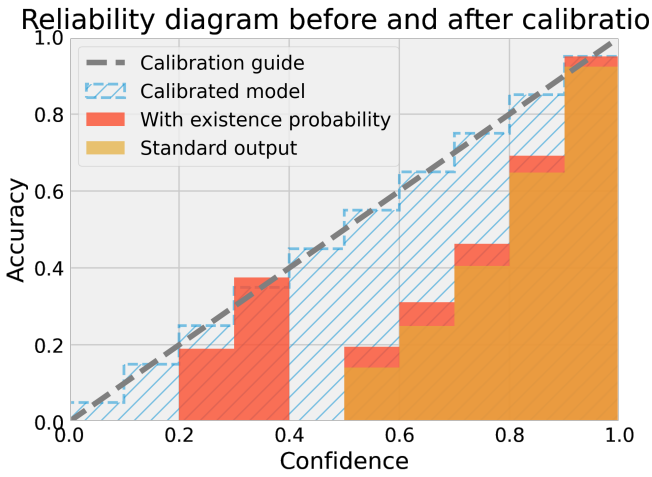


Figure 5.19: Reliability diagram showing a perfectly calibrated model versus the standard output and the Existence Probability. The proposed merging strategy demonstrates an approximation of the output of the model to the perfect calibration, leading to a better balance between confidence and accuracy.

detections with mid confidences, leaving these to maintain high values despite lower accuracies. In this sense, these outcomes demonstrate that the Existence Probability has a possibility to be further improved as a calibration tool.

5.8 Discussion

The main objective of this thesis is to study novel interpretations of uncertainty and to utilise their availability to improve the robustness and interpretability of object detection models for autonomous driving. Specifically, one of the main objectives is to enrich the output of these types of models, enabling the detection of areas where the standard architecture is unsure of the decision boundary. In this sense, the concepts of existence and uncertainty are thoroughly studied and evaluated through the many procedures depicted in this chapter.

Firstly, concerning the uncertainty interpretations, it is found that the proposed aleatoric estimation models feature a correlation with the distance to the ego vehicle. Contrary to this, the epistemic uncertainty demonstrates no evident connection, as desirable. Between the two proposed interpretations of the aleatoric parcel, the *direct entropy estimation* features a stronger relationship with this variable, showing higher coherence with the expected result.

Analysis of the correlation of the aleatoric component with occlusion and difficulty follows. Whilst the results showcase no relationship between the proposed interpretations and occlusion, a *weak* connection with the difficulty is observed. Since the measuring of the difficulty of a detection includes the occlusion, truncation and number of points within the bounding box, these outcomes suggest that the origin of randomness and noise in pointcloud data is a complex phenomenon controlled by a panoply of variables. Regardless, the behaviour of both the *direct entropy estimation* and the *joint entropy estimation* models seems inconsistent towards occlusion, especially since

areas behind objects are know to be highly noisy. Some examples of these situations are also analysed through the Uncertainty Map, as depicted in Figure 5.16 and 5.17. A possible explanation for this lies in the incoherencies of the measures of the conditional entropy and mutual information, as explored by Wimmer et al. (2023). As shown, these measures can display some inconsistent behaviour, possibly leading to the observed disconnect. Furthermore, since anchor proposals are many-to-one, much variability is expressed by the region proposal layer, possibly resulting in less stable estimates of the aleatoric uncertainty.

Another factor of relevance to the correlation analysis of the aleatoric uncertainties with the depicted aspects is the inner processing of the input data performed by the VirConv-L model. Since virtual points completed with RGB and depth completion information are leveraged before feature extraction, the resulting pointcloud can be much more resistant to variations and to intrinsic randomness, enabling the lower estimates and correlations seen in the experimental validation.

Focusing then on the proposed epistemic uncertainty, a strong correlation is observed with Existence Map confidence. Furthermore, single support patches sport higher uncertainties as the confidence increases. This is further observed in Figure 5.10 that depicts a strong tendency of the epistemic estimates to decrease as the F1-score increases in the standard model. These results showcase the coherence of the proposed interpretation that leads to explainable outcomes. Namely, higher disagreement within the network translates into increased uncertainty.

Concerning the values of uncertainty obtained in perturbation tests, the results demonstrate a more complex relationship between types of perturbations and estimates. It is seen that certain changes in the data provoke higher variance in the networks agreement, namely severe weather conditions and the phenomenon of crosstalk. In particular, the moderate crosstalk outcomes are of interest to evaluate in more detail. Despite achieving the highest epistemic uncertainty by a large margin, this modality also features the lowest aleatoric predictions. This demonstrates another weakness found in many uncertainty quantification methodologies [Gal and Ghahramani (2016); Kendall and Gal (2017); Teye et al. (2018)] that rely on neural networks estimates - uncertainty is modelled by the network, and as such, is an approximation controlled by the obtained predictive distributions and thus, by the optimisation process. As confirmed by Wimmer et al. (2023), the particular estimation of the aleatoric parcel based on the conditional entropy can not be entirely disconnected from the epistemic uncertainty, with it affecting the aleatoric calculation to some degree. Nevertheless, the availability of these estimates still proves *useful*. Returning to the crosstalk case, higher distance between aleatoric and epistemic parcels further proves that the model has been deeply affected by this perturbation, ensuing further study. These outcomes are confirmed in Table 5.13 where the performance of the Existence Map heavily drops in the presence of crosstalk affected data.

On another topic, the overall analysis of the calibration of the proposed interpretations of uncertainty revealed low errors for both parcels in comparison with the selected literature methods, especially for the aleatoric component. With significant lowered miscalibrations, reaching a maximum of -86% over the established literature, the coherence and strength of the developed models is evident. This analysis seems to suggest that both aleatoric interpretations are high performing

with the *joint entropy estimation* leading to the strongest mitigation of miscalibration. As for the epistemic parcel, despite the results not being as divergent from the literature as the other parcel, there is still a considerable decrease in calibration errors for this interpretation over the tested methodologies from the literature, reaching a maximum of -31%. This demonstrates that the proposed estimators of uncertainty enable better calibration over several established techniques in the area of uncertainty quantification for deep learning, whilst keeping a simple approach with low running time that does not enforce architectural changes.

Concerning the Existence Map, the evaluation suggests the efficiency of the proposed method. Despite leveraging knowledge only from the region proposal layer, its precision and recall results are not too distant to the standard output, with more noticeable variation in the analysis of perturbations. Even with inconsistent outputs in perturbation tests, higher recall than the standard model is achieved in most modalities, suggesting that the Existence Map does suggest a high range of areas of interest for fine-grained analysis on the existence of objects. As such, the effectiveness of the Existence Map is proven as a tool that enables failure detection and mitigation in object detection models, leading to increased robustness and interpretability. Further demonstrating the usefulness of the proposed aleatoric uncertainty interpretations, the uncertainty maps also showcase relevant information that, merged with Existence Map outcomes, has the potential to enable the recovery of further missed areas of interest as well as the rejection of false positives. Examples of these are shown in Figure 5.13, 5.14 and 5.15.

Another relevant aspect is the fact that existence and uncertainty maps operate over all classes in the dataset despite the model being trained in a closed set. In this sense, similar literature is found in the field of open-set classification. Open-set classifiers typically require training models under different frameworks [Balasubramanian et al. (2021); Yang et al. (2024a)] and aim to classify objects besides the considered classes as unknowns. On the contrary, Vaze et al. (2021) use a standard classifier and show its ability to perform well under open-set conditions, despite being trained for closed-set classification. This thesis also demonstrates that closed-set models have underlying knowledge that enables good open-set performance. However, the discussed proposals do not require retraining or repurposing of the model, being applicable at test time or during deployment of any anchor-based standard object detection model. Further than this, the generation of existence and uncertainty maps can ensure compatibility with real-time necessities.

Concerning the Existence Probability, results obtained with the proposed merging strategy overcome those of the standard model both in the validation and perturbation sets, showing considerably improved robustness. This once more showcases the ability of the proposed concepts of existence and uncertainty to enable the recovery of missed detections and to curb false positives. Focusing on the potential to recalibrate the standard output, despite the positive results, in comparison with literature, the Existence Probability proves insufficient. Guo et al. (2017) discuss several methods with higher accuracy, especially temperature scaling. The obtained results demonstrate the potential of the approach, however, further tuning and development may lead to even better outcomes. Despite this, the Existence Probability is designed as a tool to merge the information of the Existence Map with the standard output, with calibration being a collateral effect.

5.9 Chapter Summary

The evaluation of the main proposals of this thesis is carried out in this chapter. This entails the validation of the behaviour and estimates of uncertainty obtained with each interpretation as well as the explored uncertainty applications.

Pertaining to the same structure, the problem addressed in the chapter is first reported in Section 5.1. This is followed by the discussion of the architecture of the model selected for the validation, VirConv-L [Wu et al. (2023b)], in Section 5.2. Finalising the preliminaries are the implementation details in Section 5.3 and the evaluation methods in Section 5.4. Respectively, these focus on the reporting of all the parameters of interest to enable result reproducibility and on the metrics utilised to validate results.

The first experimental evaluation focuses on the uncertainty interpretations, in Section 5.5, where their behaviours and estimates are thoroughly analysed. The following Section 5.6 reports the results obtained with the existence and uncertainty maps, including analysis in perturbation. With a similar procedure, Section 5.7 explores the Existence Probability, dissecting its performance in standard and out-of-distribution data. Finally, the outcomes of the experimentation are outlined and discussed in Section 5.8, including a comparison with literature of relevance.

Chapter 6

Underspecification and Interpretations of Uncertainty

The phenomenon of underspecification, as thoroughly discussed in Subsection 2.5.4, depicts the variability of results obtained in deployment of diverging predictors that are naturally generated from the same model architecture. D'Amour et al. (2022) demonstrate through extensive experimentation that this condition is found in many machine learning pipelines, plaguing also state-of-the-art neural networks in a wide variety of domains.

One of the most concerning aspects resultant from the discussions in D'Amour et al. (2022) is the fact that the intrinsic randomness of deep learning models leads to the generation of different predictors even with little to no hyper-parameter changes. This demonstrates that not only is underspecification unavoidable in neural networks, but there is also no existent method in the literature to ascertain whether a retrained predictor conserves the former assumptions and biases. These aspects of the underspecification phenomenon greatly hinder a neural network's interpretability as behaviour in in- and out-of-domain is hard to grasp and explain. Since no given input guarantees an end result, the presence of underspecification is a huge threat to the deployment of deep learning models in safety-critical conditions such as autonomous driving. Even more daunting, each time a model is updated through retraining, there is no safety guarantee or predictability towards its behaviour, despite the availability of documented results from previous predictors.

In this sense, in this chapter, underspecification is studied in depth, especially relating to the applications of interest already discussed in previous chapters, culminating in two different analysis in the context of both classification and regression assignments. For the establishment of the goals of the chapter, Section 6.1 first formally defines the problems to be tackled. The underspecification distribution is defined in Section 6.2 followed by the methodologies applied to obtain the epistemic uncertainty in each problem in Section 6.3. Section 6.4 follows with a description of the utilised deep learning architectures as well as the datasets and the performed splitting operations. Relevant parameters for experimentation replication are also communicated. Before the experimental procedure is delved into, an exploration of the metrics utilised in the evaluation of the results is featured in Section 6.5.

As for the experimental procedures, the first study concerns a classification problem, discussed in Section 6.6. The other analysis features the problematic of object detection and is reported in Section 6.7. The results obtained from these procedures are then discussed in 6.8, thus finalising the chapter.

6.1 Problem Definition

Interpretability in neural networks is defined as the ability of a human to predict the result of a model with a given consistency. As such, the presence of underspecification in deep learning pipelines greatly hinders the deterministic forecast of the outcomes of networks, especially in deployment domains.

In order to study the interpretability of deep learning in the context of underspecification for object detection and autonomous driving applications, the following factors are of major relevance:

Underspecification The phenomenon of underspecification is severely understudied since its discovery has been coined recently. Further than acknowledging its presence, the severity of the spread generated by the diverging predictors in the deployment domain must be quantified so as to more concretely communicate the accuracy of deep learning models. A prediction of the divergence in results can largely improve the interpretability of these models, giving a more informed notion of the predictability of outputs given certain inputs and domains.

Epistemic Uncertainty The epistemic uncertainty, as previously covered, quantifies the disagreement of the different predictors generated by a deep learning architecture. This definition shares many concepts in parallel with the phenomenon of underspecification. As such, it is relevant to study a connection between these factors. The presence of epistemic estimates may enable the prediction of the spread of predictor accuracy in the deployment domain, contributing to the increased interpretability of neural networks.

From these notions, the problem of this chapter can then be defined as follows: to verify the presence of underspecification in both classification and object detection problems and to evaluate the estimation of the spread of predictor accuracy in perturbation domains through the availability of epistemic uncertainty estimates. Through experimentation with several models, including prominent computer vision backbones, these factors are studied and analysed, concluding whether the epistemic uncertainty can be a good approximation of predictor spread, enabling the prediction and mitigation of the phenomenon without multiple expensive training sessions.

6.2 The Underspecification Distribution

The underspecification distribution may be defined upon the random variable that expresses the accuracy of a predictor φ, generated by training the same architecture under different conditions. These diverging factors, which range from hyperparameter changes to stochastic initialisation and

optimisation procedures, enable the generation of a panoply of predictors that are expressions of the distribution $p(\theta|\mathscr{D})$.

The considered random variable, A, represents the model's accuracy and can thus be calculated for each predictor by averaging a metric of interest, as demonstrated in Equation 6.1.

$$\begin{aligned} A_\varphi &\in [0,1] \\ A_\varphi &= \mathscr{M}(Y_\varphi, X, \theta_\varphi) \end{aligned} \tag{6.1}$$

Pertaining to the definition of A, the population mean of the underspecification distribution can then be approximated by averaging the results obtained in the sampled predictors as follows:

$$\mu_A \approx \frac{1}{|\varphi|} \sum_{\varphi} \mathscr{M}(Y_\varphi, X, \theta_\varphi) \tag{6.2}$$

The metrics, $\mathscr{M}(Y_\varphi, X, \theta_\varphi)$, utilised in the context of this chapter's experimental evaluation are the classification accuracy, $\delta_{\arg\max}$, and the F1-score, as previously introduced in Subsection 2.4.3.

The classification accuracy is thus obtained through the $\delta_{\arg\max}$ function, defined as follows:

$$\delta_{\arg\max(p(y_\varphi|x,\theta_\varphi))l} = \begin{cases} 1, & \text{if } \arg\max(p(y_\varphi|x,\theta_\varphi)) = l \\ 0, & \text{otherwise.} \end{cases} \tag{6.3}$$

where $\arg\max(p(y_\varphi|x,\theta_\varphi))$ returns the class identifier for sample x and l is its corresponding ground-truth label. This definition of accuracy pertains to classification problems, which is the focus of the experimental evaluation.

Pertaining to these considerations, and in the context of performed experiments, the value of A_φ can be obtained through the following equations respectively for a classification assignment and an object detection problem:

$$A_\varphi = \frac{1}{N} \sum_{N} \delta_{\arg\max(p(y_\varphi|x,\theta_\varphi))l} \tag{6.4}$$

$$A_\varphi = 2\frac{precision \cdot recall}{precision + recall} \tag{6.5}$$

where N is the number of samples in the classification dataset and both precision and recall are obtained as discussed in prior chapters.

6.3 Modelling the Average Metric Epistemic Uncertainty

Having established the central concepts of the underspecification distribution, the transformation of the epistemic uncertainty is then tackled. Since the goal of this chapter is to provide an approximation of the spread of distribution A through the epistemic uncertainty, estimates must be transformed to the space of random variable A.

In this sense, this section describes two different approaches taken to achieve this transformation: one focused on a general classification problem, leveraging pre-existent literature methods, and another targeting an object detection assignment, adapting the uncertainty models proposed in this thesis.

6.3.1 A Classification Problem

For the study of a classification problem, a general concept of uncertainty is adopted, in the fashion of Malinin (2019). Following this idea, the parcels of uncertainty are decomposed as represented in Equation 6.6, where TU corresponds to the total uncertainty, AU to the aleatoric parcel and EU to the epistemic component. This interpretation of uncertainty is thoroughly discussed in previous Section 3.2.

$$\begin{aligned} \text{TU} &= \text{H}[\mathbb{E}_{p(\theta|D)}(p(y|x,\theta))] \\ \text{AU} &= \mathbb{E}_{p(\theta|D)}[\text{H}(p(y|x,\theta))] \\ \text{EU} &= TU - AU \end{aligned} \tag{6.6}$$

Total uncertainty is rather useful, however, for improved interpretability, it is indispensable to decompose predictive uncertainty in parcels with distinct origins. Through this interpretation, both data and parameter-originated uncertainties can be represented, assuring that estimates are holistic. For the underspecification experiments, only the epistemic parcel is analysed. Nevertheless, its calculation relies on the other parcels, so all components of the uncertainty have to be estimated in the context of the designed experimental procedures.

In order to apply these concepts to a classification problem, $p(y|x,\theta)$ is modelled as the sum of a predictor's softmax outputs after being fed all dataset examples over the number of parameter vectors $\theta_t \leftarrow q(\theta)$ with $t \in [1,T]$, as represented in Equation 6.7.

$$p(y|x,\theta) \approx \frac{1}{T}\sum_{t=1}^{T} Softmax(f_{\theta_t}(x)) \tag{6.7}$$

The previously reported equations omit the subscript φ to facilitate comprehension of the described relations. In the paradigm of the underspecification distribution, each parameter vector is specific to each predictor since, as explained previously, stochasticity and initialisation procedures may lead to different optimisation paths. Divergent weights also culminate in network output variability, leading to both y and θ being dependant on φ.

To enable the modelling of a distribution over the weights, both Monte Carlo Dropout [Gal and Ghahramani (2016)] and Deep Ensembles [Lakshminarayanan et al. (2017)] are leveraged. Dropout is a stochastic procedure that drops neurons with a pre-defined probability. As such, the use of dropout can capture a distribution over the weights, which in turn can be directly plugged in the aforementioned definition of uncertainty, previously demonstrated in Equation 6.6. On the

other hand, ensemble-like models are naturally attuned to uncertainty calculation as each sub-model's output can be treated as a sample of the predictive distribution, also allowing for its modelling through Equation 6.6.

Epistemic uncertainty captures the variability of outcomes that can be obtained with the same network architecture sporting different parameters. It is thus an expression of the information encoded in the distribution $p(\theta|\mathscr{D})$. Underspecification, on the other hand, arises as the same architecture can generate different predictors due to stochastic processes in the network's training. The resulting predictors are also realisations of the distribution of parameters $p(\theta|\mathscr{D})$.

If the distribution of parameters $p(\theta|\mathscr{D})$ is truly captured, the averaging of epistemic uncertainty should describe the variation between predictors in the underspecification distribution. With this in mind, and as previously established, a method to interpret the epistemic uncertainty calculated through Equation 6.6 is needed so as to translate it into the space of variable A.

Since the presented formula for calculating the epistemic uncertainty is a difference of entropies, the result also belongs to the manifold of the entropy. As such, and as the maximum entropy is $logC$, the resulting epistemic uncertainty can be transformed to the unitary space of the random variable A through the following formula:

$$\frac{1}{N|\varphi|\log C}\sum_{\varphi}\sum_{N}(\mathrm{H}[\mathbb{E}_{p(\theta|\mathscr{D})}(p_{\varphi}(y|x,\theta))]-\mathbb{E}_{p(\theta|\mathscr{D})}[\mathrm{H}(p_{\varphi}(y|x,\theta))]) \tag{6.8}$$

where C is the number of target classes. The presented measure represents the average-metric epistemic uncertainty obtained over all samples and predictors scaled by the maximum amount of units that are needed to convey any outcome of the random variable.

6.3.2 An Object Detection Problem

Concerning the object detection setting, uncertainties are modelled through the interpretations proposed in Section 4.4. For the quantification of the aleatoric uncertainty, the *direct entropy estimation* model is leveraged, as defined in Equation 4.7 where the entropy of the existence distribution is obtained through Equation 4.8. The epistemic component, on the other hand, utilises the expected value over the patches distribution of the defined distance metric, as reported in Equation 4.16.

The proposed epistemic uncertainty interpretation measures the distance between estimated aleatoric uncertainties in the standard output over the region proposal layer factored by the probability ratio. The assumed aleatoric interpretation leverages the entropy of each anchor distribution. As such, the resulting epistemic component is also a subtraction of entropies, as concluded in the classification problem. Despite this, the maximum uncertainty varies from the aforementioned standard value, being defined as the largest distance between an output probability and an anchor proposal, namely $1\log\frac{1}{0.1}$. Taking the expected value over the dataset in the maximum uncertainty condition leads to $|\mathscr{S}|\log\frac{1}{0.1}$ as the highest value, observable when all samples sport the aforementioned distance over the number of patches generated for each sample.

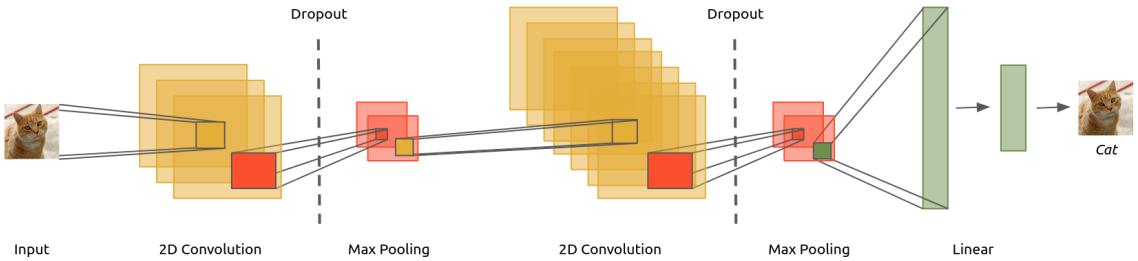


Figure 6.1: Architecture of the implemented LeNet-5 model.

Based on these ideas, the average-metric epistemic uncertainty results in the following expression:

$$\frac{1}{|\varphi||\mathscr{S}|\log 10}\sum_{\varphi}\hat{EU}_{model} = \frac{1}{|\varphi||\mathscr{S}|\log 10}\sum_{\varphi}\mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(Y|\Pi,\theta)||P(A|\Pi,\theta))] \tag{6.9}$$

where the proposed epistemic uncertainty, $\mathbb{E}_{P(\Pi)}[\sum_{\Pi} d(P(Y|\Pi,\theta)||P(A|\Pi,\theta))]$, is averaged over all predictors φ and normalised by the maximum uncertainty. In order to homogenise the calculation of the normalised estimates, and to ensure maximal value, $|\mathscr{S}|$ is taken as the maximum number of patches generated on an input sample over the entire dataset.

All the factors needed for the calculation of each parcel are computed within the object detection model, with the top n anchors spanning the anchor distributions in the region proposal layer. The output is obtained in the form of bounding boxes with an associated confidence. In order to calculate the F1 score, for the standard model, intersection with the ground-truth is analysed, with boxes being considered a correct detection for $IoU > 0$.

A study of the underspecification at Existence Map level is also performed. In this case, and following the same procedure, patches are considered a correct detection for $IoU > 0$ with the ground-truth.

6.4 Implementation Details

Before delving deep into the experimental procedures followed in this chapter, some considerations on the implementation side must be discussed and thoroughly reported so as to allow complete replication of the experimental procedures. This report includes the utilised models as well as parameters of interest, followed by the selected datasets and respective splits.

6.4.1 Models

For the assessment of the underspecification phenomenon in a generic Computer Vision context, two widely popularised [Guo et al. (2017)] backbones are selected, LeNet [Lecun et al. (1998)] and ResNet [He et al. (2016)]. LeNet is utilised due to its demonstrated almost perfect calibration

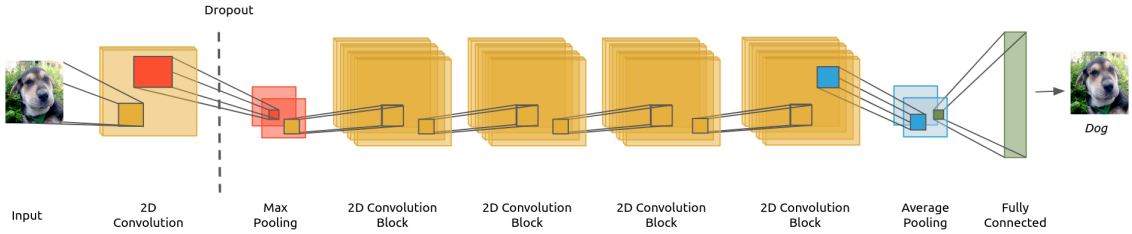


Figure 6.2: Architecture of the implemented ResNet-18 model.

versus ResNet which is proven to be overconfident [Guo et al. (2017)]. In both, the final layers are adapted to return a categorical distribution for classification tasks.

The LeNet architecture is a groundbreaking model designed to perform document recognition [Lecun et al. (1998)]. Due to its remarkable results at the time, its applications were numerous, being impactful in many fields. Since this network is prominent, several versions of it exist. Figure 6.1 features the LeNet-5 architecture that is employed in the experimental procedure following a modular fashion. This architecture sports two two-dimensional convolutional blocks separated by dropout and max pooling layers. The features are then forwarded to a linear block, with the categorical distribution computed in the final linear layer. The dropout layers are relevant for the quantification of uncertainty utilising Monte Carlo Dropout.

The ResNet model is another significantly impactful backbone in Computer Vision applications [He et al. (2016)]. Since its widespread reach enabled performance improvements in a panoply of domains, several versions of this network also exist. The selected architecture for the experimental procedures is the ResNet-18, as described in Figure 6.2.

This architecture features an initial convolutional layer that is forwarded to batch normalisation and maximum pooling operations. Dropout is added before the maximum pooling in order to enable once more the quantification of uncertainty through Monte Carlo Dropout. The results of the maximum pooling are then fed to four consecutive convolutional blocks, each with four layers, with its outputs led into an average pooling layer. The pooling is the final operation before the fully connected layer transforms the features into a categorical distribution.

Concerning the application of object detection to the autonomous driving context, the utilised model is once more VirConv-L. This model is thoroughly discussed in Section 5.3, including its architecture and any relevant implementation details. All the hyperparameters described in the aforementioned section are maintained in the forthcoming experiments.

6.4.2 Datasets and splits

The first part of the experimental procedure focuses on a generic Computer Vision setting, with the evaluation restricted to classification problems, as previously mentioned. For this purpose, the CIFAR-10 dataset [Krizhevsky (2009)] is selected, which, having fewer classes, enables a more in-depth observation of the behaviours in each. Since there are cases where out-of-distribution (OOD) data is needed, a web-crawled set of 18 images corresponding to several CIFAR-10 classes

Figure 6.3: Some example samples from the web-crawled out-of-distribution set with respective labels on the bottom. The considered classes match the existent CIFAR-10 targets.

is fetched. Figure 6.3 showcases some samples from this set. A variable number of examples per class are gathered with no representation of class *truck* to introduce further randomness in the out-of-distribution set. The *truck* class is selected since, over the course of several experiments, accuracy in this class varied the lesser, thus having a reduced impact in the underspecification analysis. The same methodology for the training and testing sets is followed for the out-of-distribution data, with all images being resized to 32×32 and normalised. Experiments performed on CIFAR-10 and the web-crawled set are based on the LeNet model.

ImageNet, on the other hand, is chosen due to its notoriety and wide range of images. The ResNet backbone is employed for classification experiments in this dataset. The tiny version [Le and Yang (2015)], Tiny-ImageNet, with 200 classes, is prioritised in detriment of the wider set in order to speed up computations in large-scale underspecification experiments. For out-of-distribution data, a version of ImageNet with several perturbations is leveraged, ImageNet-C [Hendrycks and Dietterich (2019)]. This dataset contains 15 types of perturbations admitting corruptions based on noise, blur, weather and digital categories over 5 severity levels. For the reported experiments, only the *blur/motion_blur*, *digital/contrast*, *digital/pixelate* and *weather/brightness* perturbations are considered.

Concerning the object detection for autonomous driving application, the KITTI [Geiger et al. (2012)] dataset is utilised for standard tests and for perturbation data, its modified version KITTI-C [Kong et al. (2023)] is leveraged.

As previously explained, the data provided by the KITTI benchmark features training, validation and test sets. Due to the public unavailability of the labels for the test set, the training dataset is once more split in two, thus creating reduced train and test sets. As described in previous works [Wu et al. (2022b, 2023b)], the original training set, composed of 7481 frames, is split into training samples (3712 frames) and a test set with 3769 examples. As such, all predictors generated for the forthcoming experiments are trained on 3712 frames. When a validation set is mentioned, it corresponds to the created testing split of 3769 frames.

In Chapter 5, the perturbation modes and severity degrees present in KITTI-C are thoroughly described. Since the amount of available data is staggering, in order to allow the study of the underspecification phenomenon in perturbation settings, the number of samples is distilled into a smaller, more manageable set.

In this sense, predictors in perturbation mode are evaluated in a perturbation test set with the

same number of samples as the original validation set (3769 examples). Samples are selected based on the topology of the perturbation. Three topologies exist in KITTI-C: severe weather conditions, external disturbances and internal sensor failure. The number of total samples are divided for each group, leaving 1257 examples for the severe weather conditions and the internal sensor failure modes. The external disturbances only contemplate two different phenomenons, missing beams and motion blur, so, in order to complete the 3769 samples, 628 and 627 examples are taken from these topologies. Predictor score can be divided over the degree of the perturbation. For a global analysis, however, the average result over all degrees is taken for the 3769 samples as described previously. This guarantees that the plots for the validation versus perturbation spread are based on the same amount of samples whilst maintaining a wide variety of perturbations with broad representation.

6.4.3 Parameters of the Experimental Procedures

In order to plot and assess underspecification, diverging predictors must be generated from the same architecture. To allow this spanning procedure, two methods are employed, one for the studied classification problem and another for the object detection assignment.

Taking into consideration the first approach to generate the underspecification distribution, two hyperparameters are manipulated - *seed* and *weight initialisation*. These are prominent neural network parameters that scientists and practitioners tweak heavily in order to obtain the best performing combination. Further than this, these are also selected as polar opposites since the seed is a code related environment parameter whilst the weight initialisation strategy directly impacts the network's optimisation procedure.

On the other hand, for the object detection assignment, the diverging predictors are obtained by varying three hyperparameters - *seed*, *batch size* and *retraining from checkpoint*. Once again, the selected factors are proven to be impactful in the end results of the network and these are usually extensively experimented with in the literature. Besides the variation of seed which is a core related environment variable, the change in batch size and performing retraining from a previous checkpoint directly influences the internal optimisation mechanisms of the network.

With these variations, several aspects of the learning process of neural networks are tackled, showcasing the variability produced by each in the end results. Predictors obtained from these variations suggest how impactful each change can be in the outcomes produced by the models.

For the classification problem study, the number of samples taken for Monte Carlo Dropout, namely T, is set to 20, whilst the ensemble method considers 3 sub-models. Dropout probability is set to 0.5 in all experiments.

Concerning the parameters assumed for the epistemic uncertainty calculation in the object detection problem, the number of anchors kept to generate the anchor distributions is 120 and the controlling factors ρ_{anchor} and ρ_{out} are both set to 0.1. The model and Existence Map parameters are maintained as described in Section 5.3.

In order to showcase the results of the underspecification analysis, several plots are presented either in training, validation and/or out-of-distribution sets. These plots are standardised and pertain to the following structure:

blue dot represents a single predictor φ;

red dot population mean calculated from the obtained samples;

bluish area area spanned by the average-metric epistemic uncertainty;

y-axis metric $\mathscr{M}$ / value of the random variable A_φ for sampled predictor φ;

x-axis meaningless - to allow better visualisation of the spread of predictor accuracy/metric.

Following the same process as reported in prior chapters, the base of the logarithm in any entropy formulae is set to 2, with the results being interpretable as bits. The evaluation is separated into the experiments related to a classification problem and into those concerning an object detection assignment. Despite this, experimental procedures are numbered continuously through the sections since these all pertain to the same objective of studying the relationship between the underspecification and the average-metric epistemic uncertainty. A procedure is defined as a set of experiments in a similar environment that intend to study one single aspect of the previously defined problem.

6.5 Evaluation Methods

The *linear spread* is defined as the average distance between each predictor and the maximum accuracy reached by the model in the experimental procedure. Equation 6.10 shows the calculation of this relationship.

$$\textit{linear spread} = \frac{\max_\varphi\{X\} - X_\varphi}{|\mathscr{S}|} \tag{6.10}$$

where X_φ represents the value obtained by predictor φ in the metric of interest (assuming that higher is better), $\max_\varphi\{X\}$ returns the maximum value of this metric found in all sampled predictors and $|\mathscr{S}|$ stands for the number of predictors generated in sampling procedure $\mathscr{S}$.

The higher the *linear spread* the more entropy is found in the distribution of predictor accuracy in the considered metric. As such, this value communicates the amount of divergence between the generated predictors. For the facilitation of the comparison of distances between different models, the normalised *linear spread* is utilised, being obtained by multiplying $|\mathscr{S}|$ by $\max_\varphi\{X\} - \min_\varphi\{X\}$, resulting in Equation 6.11.

$$\textit{normalised linear spread} = \frac{\max_\varphi\{X\} - X_\varphi}{|\mathscr{S}|\max_\varphi\{X\} - \min_\varphi\{X\}} \tag{6.11}$$

The considered metrics for X in the forthcoming experiments are the accuracy and the F1-score. The accuracy depicts the number of correct predictions made by the classification model

regularised by the number of samples. This metric is utilised for the generic Computer Vision experiments using the LeNet and ResNet architectures tested on CIFAR-10/web-crawled set and Tiny-ImageNet/ImageNet-C respectively.

For the object detection tests with VirConv-L, the metric utilised to summarise the model's performance is the F1-score, previously explored in Subsection 2.4.3. This score considers both the precision and recall achieved during evaluation.

Furthermore, in order to assess the relationship between the *linear spread* and the epistemic uncertainty, Pearson's, Spearman's and Kendall's correlation coefficients (PCC/SCC/K-τ) are leveraged. Similarly to the F1 metric, the PCC score is thoroughly explained in Section 5.4 of the previous chapter where it is frequently utilised to analyse correlation between variables of interest. For consistency reasons, this is the main correlation analysis tool explored in this chapter as well. In the same fashion, Table 5.3 is also the reference for the appreciation of the obtained PCC score that ranges from $[-1,1]$. In order to enrich the study of the connection between the epistemic uncertainty and the linear spread of the underspecification distribution under different assumptions, the SCC and K-τ measures are also reported. This allows a more complete characterisation of the relationship between both variables, as the linear assumption does not hold on all the considered measures.

Spearman's rank correlation coefficient depicts the statistical dependency of the ranks of two variables and can be expressed in function of the PCC as shown in Equation 6.12. The SCC score thus analyses the variables through a similar method as the PCC whilst focusing on the ranks instead.

$$\text{SCC Score}(X,Y) = \text{PCC Score}(\mathbb{R}[X],\mathbb{R}[Y]) = \frac{cov(\mathbb{R}[X],\mathbb{R}[Y])}{\sigma(\mathbb{R}[X])\sigma(\mathbb{R}[Y])} \tag{6.12}$$

Finally, Kendall's coefficient measures the statistical dependence through the τ coefficient. This score also expresses rank correlation, similarly to SCC, relying on correspondence between the two variables. Its calculation is made through the formulae depicted in Equation 6.13.

$$\text{K-}\tau\text{ Score} = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{(\text{number of pairs})} \tag{6.13}$$

Both of these measures vary in interval $[-1,1]$ with zero describing no relationship, similarly to Pearson's correlation coefficient. As such, Table 5.3 also serves as the medium for a qualitative description of the obtained SCC and K-τ scores.

6.6 Experimental Evaluation in a Classification Problem

The first set of experimental procedures entail the analysis of the underspecification distribution as well as the study of its connection with the epistemic uncertainty in a generic Computer Vision

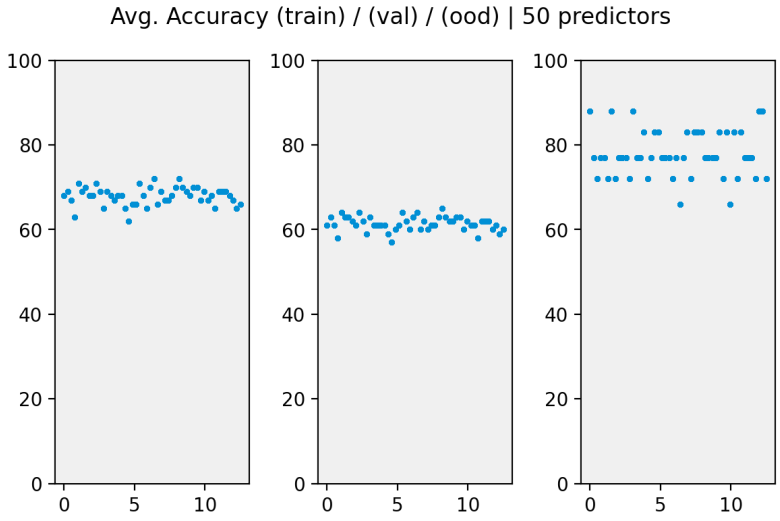


Figure 6.4: Predictors generated from training the implemented LeNet network [Lecun et al. (1998)] with equal hyperparameters 50 times on CIFAR-10. From left to right are the predictors' accuracies in training, validation and out-of-distribution data.

classification problem. As previously denoted, classification is performed on images of CIFAR-10/web-crawled set with LeNet and on Tiny-ImageNet/ImageNet-C with ResNet-18, where for each input sample, a single target class exists.

The main objective of the procedures described in this section is to first observe and confirm the existence of an underspecification distribution and then to study a possible connection between the average-metric epistemic uncertainty and the spread of the distribution of A_φ in a classification assignment using staple Computer Vision backbones from the literature.

6.6.1 Monte Carlo Dropout Epistemic Uncertainty

Leveraging the Monte Carlo Dropout method for quantifying uncertainties, two different experiments are performed, one focused on CIFAR and another targeting ImageNet. The methodological approach taken for each procedure is first described, alongside the objectives of the experiment followed by the report of relevant results obtained in each.

Procedure 1 - CIFAR Experiments Starting with the CIFAR experiments, a simple procedure is followed. During consecutive sessions, the LeNet model is trained from scratch with the same hyperparameters and dataset. Its performance is evaluated in training, validation and out-of-distribution data. Training and testing sets are from CIFAR-10, with most evaluations carried out in a random holdout set.

For OOD data, the web-crawled dataset is used. The described out-of-distribution set is rather small, which, for accuracy analysis, proves insufficient to arrive at structured conclusions. However, the objective of this procedure is simply to demonstrate the spread of predictors regardless of the accuracy range where these are located. For this objective, a smaller set is sufficient, speeding

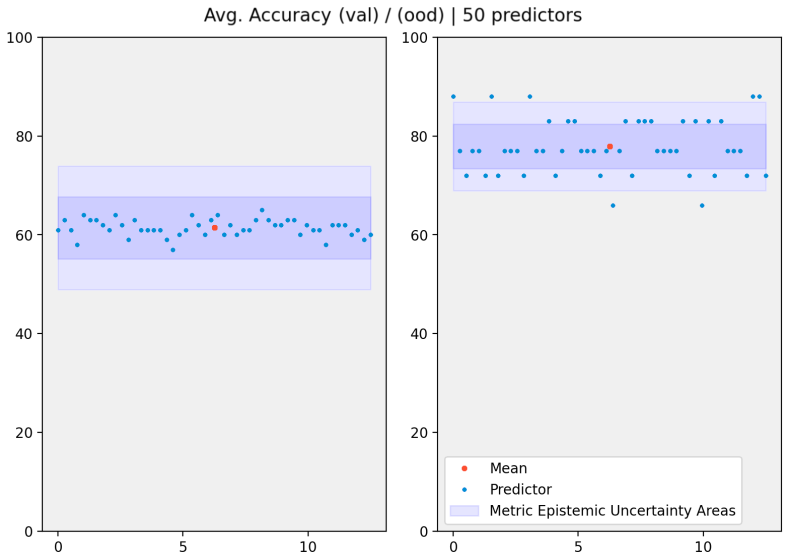


Figure 6.5: Predictors obtained with the LeNet-based network [Lecun et al. (1998)] trained with equal hyperparameters 50 times on CIFAR-10 with the average-metric epistemic uncertainty centred in the mean predictor.

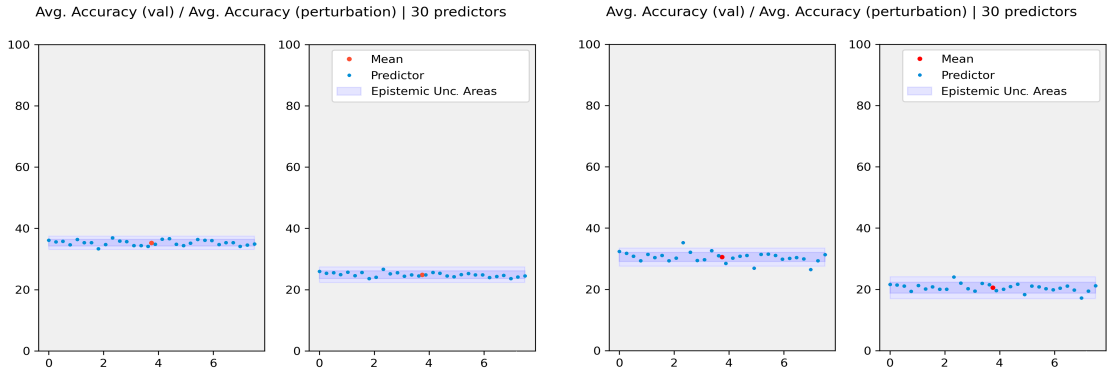
up results, and since the images are all web-crawled, the variability is expected to be maximal as the distribution in of itself is remarkably varied and has a striking distance between training and testing environments.

The objective of this procedure is to confirm the existence of an underspecification distribution generated by training the same architecture in the same conditions several times. Figure 6.4 demonstrates the underspecification distribution generated during training, validation in a holdout set, and in out-of-distribution data.

As it can be observed, model underspecification is an inherent characteristic of the LeNet model, with its prominence increasing in out-of-distribution data. Even though the average accuracy is higher in the OOD set due to its reduced size of only 18 samples, the variability between predictors is evident with the lowest result surrounding 65% and the highest 88%. This observed spread in predictor accuracy is strikingly larger than that observed in training and validation evaluations.

This experiment seems to suggest that commonly reported accuracy levels in deep learning models are an indicator of the performance of a single predictor generated in the network's optimisation path and not an exact nor measurable quantification of the model's knowledge or general performance. It is expected that these models, when deployed in out-of-distribution data as commonly found in the real world, would most likely perform poorly with even more substantial contrasting accuracy levels.

In Figure 6.5, the average-metric epistemic uncertainty is plotted centred in the sample mean for the validation and out-of-distribution sets. It can be seen that the metric epistemic uncertainty is able to encapsulate all the sampled predictors within its scope in the validation set. However, during OOD tests, the estimated value seems to be underestimated as it does not include even



(a) Predictors spanned on the random *seed* experiment.

(b) Predictors spanned on the random uniform *xavier initialisation* experiment.

Figure 6.6: Predictors resulting from training sessions of ResNet-18. In red is represented the mean predictor and the blue areas correspond to the average-metric epistemic uncertainty centred in the mean.

the majority of the obtained predictors. The other shade of blue demonstrates the average-metric epistemic uncertainty by a factor of 2. Even though this area is doubled, there are still predictors beyond the estimated variation.

This experiment suggests that the epistemic uncertainty may have the potential to predict the spread of the generated predictors correctly. Nevertheless, in higher variation conditions, in the presence of an OOD set for example, the estimation is not able to significantly predict the spread in predictor accuracy. This behaviour may originate from two main factors: either there is no possibility to predict underspecification through epistemic estimates; or the epistemic uncertainty obtained through Monte Carlo Dropout is underestimated rather than disconnected from A_φ. The underestimation of uncertainty is coherent with recent reports on Monte Carlo Dropout [Wilson and Izmailov (2020); Folgoc et al. (2021)]. Further experimental procedures reported in the next sections shed more light into these aspects.

Procedure 2 - Tiny-ImageNet Experiments Following the aforementioned procedure, similar experiments are run with ResNet18 on Tiny-ImageNet. Out-of-distribution data is sourced from ImageNet-C as described in Section 6.4. In order to force a more thorough sampling of the distribution of $p(\theta|\mathscr{D})$, the model is trained several times with the same hyperparameters except for the two variations previously explained: *seed* and *weight initialisation*. In the seed variation, at each training session, all seeds (numpy, torch, random, environment) are multiplied by a factor. In the weight variation, the weights are randomly initialised using the xavier uniform method so that each predictor is generated from a different set of initial parameters.

Within these conditions, the previously laid out procedure is followed, leading to an estimation of the underspecification distribution with the average-metric epistemic uncertainty plotted from the sample mean, as depicted in Figure 6.6. The first aspect to denote is that the *seed* experiment creates significantly lesser variability in predictor accuracy than the *weight initialisation*.

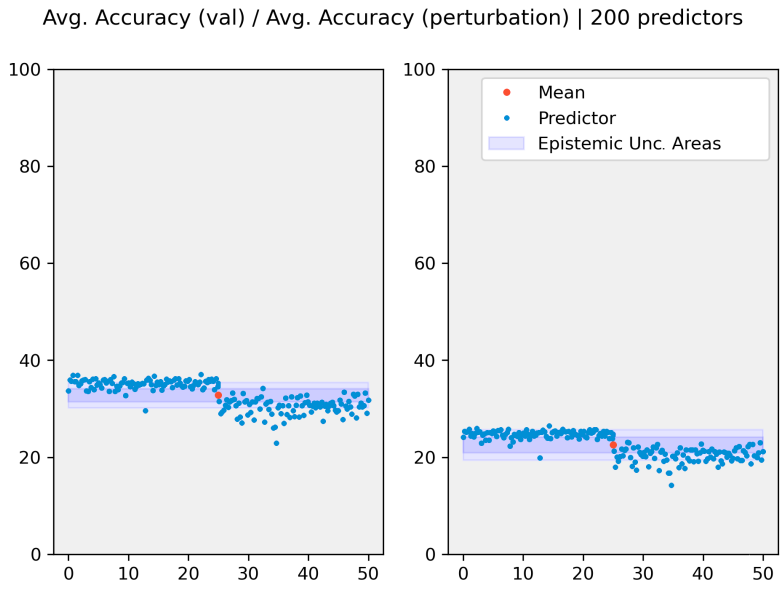


Figure 6.7: Predictors resulting from training ResNet18 where half originate from varying the *seed* and the others are a product of *weight initialisation* variations. Uncertainty estimates are obtained through Monte Carlo Dropout.

This seems to suggest that direct modifications to the network's parameters create much more variability during predictor generation than changes in the training environment. The complete underspecification distribution contains all of these possible variations and is thus much richer than what is expressed in Figure 6.6, which represents an approximation. However, for practitioners and researchers looking to gain more insight into the predictor that they will obtain after training a given architecture under different conditions, the observed aspects may be of relevance.

From these experiments, it can be seen that, with a larger dataset and network, epistemic uncertainty fails to fully capture the variation in predictor accuracy consistently, unlike in CIFAR-10 experiments where the only major discrepancy is found in out-of-distribution.

The first observation of interest is that, using a perturbation set instead of full out-of-distribution data reduces variability in A_φ, allowing the average-metric epistemic uncertainty to better characterize the distribution's spread in the overall. This effect is more pronounced in the seed experiment 6.6a, where predictors show lesser spread in accuracy. In contrast, in Figure 6.6b, as variability increases, uncertainty estimates cover fewer predictors, especially in the validation set. In the perturbation set, the predictors are spread out in a smaller area compared to validation. This result may originate in the intrinsic randomness of neural networks that can lead to the generation of outliers in validation.

Even though individual analysis of these variations is of interest, the true underspecification distribution captures a much larger panoply of predictors. To better approximate the underspecification distribution, another procedure is designed. In this sense, 200 predictors are spanned in total, with 100 being generated by varying the seed during training and the other resulting from random weight initialisations (all uniform Xavier). Figure 6.7 depicts the samples obtained from

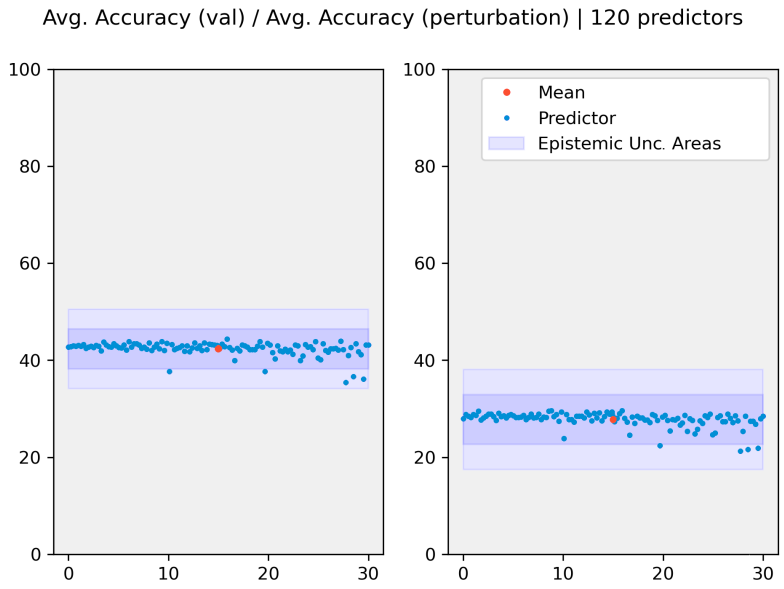


Figure 6.8: Predictors resulting from training ResNet18 where half originate from varying the *seed* and the others are a product of *weight initialisation* variations. Uncertainty estimates are obtained by ensembling 3 sub-models.

the underspecification distribution with the average-metric epistemic uncertainty plotted from the mean predictor. The first observation to be made is that the predictors concentrate in two major areas, one over the mean and another under. The area above the mean accuracy has a smaller range in values whilst the other area presents much more prominent variation. The previous experiment, with results shown in Figure 6.6, suggests that the *weight initialisation* predictors may lead to lesser overall accuracies and more variability in between predictors, which indicates that the area under the mean point could be spanned by these predictors. Once more, this implies that varying environment hyperparameters has a reduced impact when compared to changing internal components of the network, as compatible with expectation.

Focusing then on the average-metric epistemic uncertainty, opposed to what is depicted in Figure 6.6, but reflecting the suspected tendency, the uncertainty estimates are insufficient at covering the span of A_φ. In fact, when the sampled population is more approximated to the true underspecification distribution, the uncertainty estimated by Monte Carlo Dropout proves even more insufficient at capturing the variability of results. Once more, this suggests either severe underestimation or a disconnect of the epistemic uncertainty with A_φ.

6.6.2 Ensemble Epistemic Uncertainty

In the previous subsection, results obtained with Monte Carlo Dropout question the source of the lack of connection between the uncertainty estimates and the underspecification distribution. To exclude the possibility of underestimated uncertainties, which can result in an insufficient prediction of the spread of predictors generated by a single model, one last procedure is designed for a

classification problem, using ensembles [Lakshminarayanan et al. (2017)] for the generation of a predictive distribution.

Procedure 3 - Tiny-ImageNet Experiments In this procedure, a simple ensemble of three models is designed and the negative log-likelihood is incorporated during training, which enables the sampling of a predictive distribution by leveraging the diverse outputs from each sub-model [Lakshminarayanan et al. (2017)]. The experimental procedure remains consistent: produce variability both with the *seed* and *weight initialization* variants, with an equal number of predictors for each. This strategy allows the approximation of the underlying underspecification distribution more accurately. To optimize computational efficiency and resource usage, the number of predictors in this analysis is limited to 120.

Figure 6.8 showcases the underspecification distribution that was obtained with the average-metric epistemic uncertainty plotted from the mean accuracy. In this experiment, there is no clear division in areas, however, the predictors above the mean still demonstrate much less variability than those under it. This may suggest that there is a higher probability of outliers appearing in the area under the mean. More experimentation in this sense may provide clearer conclusions on this topic. Regardless, since the accuracy of each predictor is in this case obtained as an average of three sub-models, less variability is expected.

When it comes to the average-metric epistemic uncertainty, the difference is significant when using ensembles. As observed in Figure 6.8, the first area of the metric epistemic uncertainty covers on average 95% of the predictors sampled both in validation and perturbation. The divergence in results from the Monte Carlo Dropout experiment demonstrates that the most plausible explanation is the underestimation of uncertainty, an aspect of the method denoted in other works as well [Wilson and Izmailov (2020); Folgoc et al. (2021)]. This suggests that the epistemic uncertainty can be a reliable estimation of the range of the underspecification when it is not underestimated.

In order to more accurately characterise the relationship between the epistemic uncertainty and the underspecification distribution, the correlation between the linear spread and the metric epistemic uncertainty is quantitatively measured at each predictor through the Pearson's (PCC) and Spearman's (SCC) correlation coefficient and Kendall's tau (KT). In validation, the PCC, SCC and KT scores equal 0.8705, 0.8280 and 0.6162 respectively, corresponding to an average spread of 2.05, whilst in perturbation, its values are 0.7019, 0.6995 and 0.4802 with an average spread of 1.86. Both evaluation modes feature considerable strength through all considered metrics, exhibiting a strong correlation between the predictor-level accuracy variation and its estimated epistemic

Table 6.1: Average linear spread obtained using the standard model outputs, over all data modalities.

	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
average linear spread	0.0097	0.0080	0.0081	0.0063

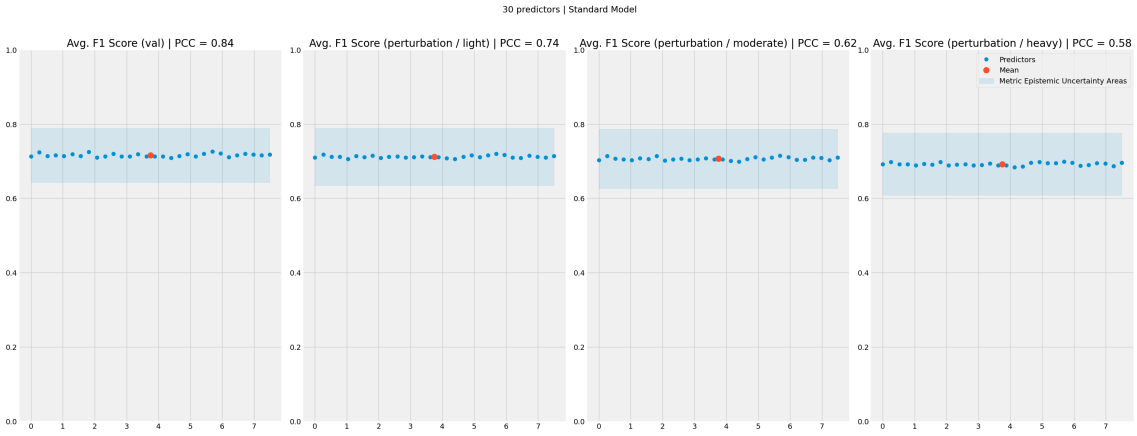


Figure 6.9: Predictors generated from training VirConv-L over the *seed*, *batch size* and *retraining from checkpoint* variants. F1 scores collected from standard model outputs. The average metric epistemic uncertainty is plotted from the median predictor.

uncertainty. The spread demonstrates that, in this case, the sampled underspecification distribution sports similar variability in accuracy both in validation and perturbation with an average of 2 points difference.

6.7 Experimental Evaluation in an Object Detection Problem

The final experimental procedures focus on the evaluation of the underspecification phenomenon in an object detection problem. For this reason, the VirConv-L model is once more leveraged with the KITTI/KITTI-C datasets for validation and perturbation data respectively.

The main objective of the forthcoming experiments is to analyse the behaviour of the proposed epistemic uncertainty interpretation as well as to further study the relationship with the underspecification distribution.

6.7.1 Anchor Distribution Epistemic Uncertainty

In the context of the proposed epistemic uncertainty interpretation, three different experimental procedures are defined, concerning the standard model output, the Existence Map suggestions and the boxes merged with the Existence Probability strategy.

Table 6.2: Measured values for each coefficient over all data modalities for the standard model outputs.

Coefficient	Measured Value			
	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
PCC	0.84	0.74	0.62	0.58
SCC	0.82	0.75	0.70	0.59
K-τ	0.61	0.57	0.51	0.38

Table 6.3: Average linear spread obtained using the Existence Map outputs, over all data modalities.

	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
average linear spread	0.0158	0.0103	0.0126	0.0122

For each experiment, the results are collected and evaluated simultaneously for the standard model outputs, the Existence Map and the merged bounding boxes, as well as the epistemic uncertainty. As such, the 30 predictors represented in each forthcoming procedure are the result of the same training sessions with the evaluation of the metric $\mathscr{M}(Y,X,\theta)$ performed over different outputs.

As reported in Section 6.4, the variations utilised to generate the predictors are the *seed*, *batch size* and *retraining from checkpoint*. For these variants, 10 different seeds are chosen at random, batch size varies between 4 and 16 and retraining is performed from epoch 30. Combining all of these factors leads to a total of 30 predictors sampled.

Procedure 4 - Standard Model Experiments From the aforementioned environment, the standard model outputs are collected and respectively evaluated. Figure 6.9 reports the results obtained for the standard output. The first observation of interest is the spread in A_φ. Table 6.1 depicts the obtained average linear spreads in each data modality. As observed, the variation is very low in predictor accuracy, suggesting very high consistency in the model's optimisation procedure. Despite the degree in the severity of perturbation increasing substantially, the sampled predictors still concentrate in a narrow F1-score area. This result may originate in the internal mechanisms of the VirConv-L model that leverage virtual points generated from a combination of RGB data and depth completion information being fused into the processed pointcloud. This step ensures high data similarity in between environments, even in the presence of perturbation data as the information is completed from other sensors. This suggests the relevance of sensor fusion in the face of data perturbations and anomalies.

Considering the area spanned by the metric epistemic uncertainty, it is observable that it includes 100% of all the spanned predictors. Despite the low variability, the epistemic uncertainty seems to suggest that, with further sampling of the underspecification distribution, a wider spread in accuracies could be manifested. As established in previous experiments, even though the VirConv-L model ensures high data consistency, its full underspecification distribution is impractical to be sampled due to the high cost. With 30 predictors, the obtained approximation may be less varied and expressive than expected. Further experimentation with substantially higher number of predictors could complete these assumptions, leading to a more informed conclusion on the characterisation of the underspecification distribution for the VirConv-L architecture.

Concerning the relationship between the uncertainty and the underspecification distribution, once more, the PCC, SCC and K-τ scores are computed. Table 6.2 reports the values measured for

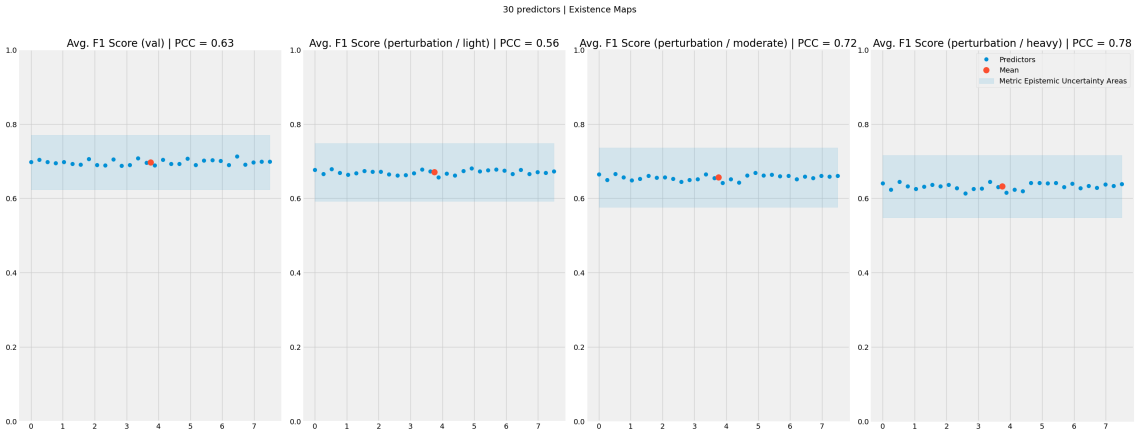


Figure 6.10: Predictors generated from training VirConv-L over the *seed*, *batch size* and *retraining from checkpoint* variants. F1 scores collected from Existence Map outputs. The average metric epistemic uncertainty is plotted from the median predictor.

each coefficient when comparing the observations for the linear spread and epistemic uncertainty at each predictor.

Pearson's coefficient in validation equals 0.84, depicting a *very strong* relationship in this modality. Perturbations *light* and *moderate* follow closely with *strong* at 0.74 and 0.62 respectively. Despite the *heavy* perturbation achieving a slightly lower score, 0.58, the characterisation of the measure still inches close to a *strong* relationship. This clearly showcases the ability of the estimated epistemic uncertainty to linearly depict the spread in the predictor distribution A_φ. A higher disconnect between uncertainties and heavier perturbations tests is expected since the further from the training distribution is the data, the more uncertainty is induced in the model, leading to higher variability and underestimation possibilities. Despite this, the reported results highlight a stronger connection than expected even in the face of strong perturbed data, reinforcing the relationship between the epistemic uncertainty and the underspecification phenomenon.

Spearman's correlation coefficient demonstrates similar results to the PCC scores, as seen in Table 6.2. Over all modalities, measured values concentrate around a *strong* relationship, with validation showing a *very strong* connection. On the other hand, Kendall's τ features lower values overall, which may indicate some inconsistencies in rank over all data modalities, albeit minor. Another possibility is the existence of duplicate values that lower Kendall's τ, as suggested through the concentration of predictor accuracies seen in Figure 6.9.

In summary, the values obtained over all data modalities for all considered measures suggest that there is a strong linear relationship between the epistemic uncertainty and the spread with a tendency to monotonicity. As such, this procedure demonstrates a clear connection between epistemic estimates and the underspecification phenomenon.

Procedure 5 - Existence Map Experiments Considering then the Existence Map output, predictor spread increases significantly as observed in Figure 6.10. Variability in predictor f1 scores is

Table 6.4: Measured values for each coefficient over all data modalities for the Existence Map.

Coefficient	Measured Value			
	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
PCC	0.63	0.56	0.72	0.78
SCC	0.65	0.55	0.65	0.79
K-τ	0.48	0.42	0.48	0.62

even more evident when compared with the previous experiment (*Procedure 4*). This is confirmed by the average linear spread values reported in Table 6.3 which show a substantial increase.

Over the range of perturbation severities, the variability in predictor f1 scores steadily improves. Despite this, the variability is still lower than the obtained results in *Procedure 2* and *Procedure 3*. This may be explained, once more, by the undersampling of the underspecification distribution. Increases in spread over more severe perturbation degrees aligns more closely with expectation since the region proposal layer has less fine-grained information, being thus more sensitive to variations in the data conditions. The divergence to the results showcased in Figure 6.9 seems to suggest that the VirConv-L model sports a high capability to recover from region proposal uncertainty and variability in its later layers.

Even in the face of much higher spreads, the average-metric epistemic uncertainty is still able to fully cover the spanned realisations of A_φ as observed through Figure 6.10. The measured values for the considered PCC, SCC and K-τ coefficients are reported in Table 6.5. Analysing Pearson's and Spearman's correlation scores reveals once more a consistency between both metrics as values concentrate closely over all data modalities. Similarly to the previous procedure, Kendall's τ results in lower overall values, a behaviour possibly explained once more by the duplicates in predictor f1 scores or minor rank inconsistencies. From these measures, the relationship between the proposed epistemic model and the outputs from the region proposal layer is characterised as being strongly linear and monotonic.

Whilst both PCC and SCC metrics align mostly on a *strong* connection, K-τ concentrates more on a *moderate* strength. The divergent factor from the previous procedure is the increasing coefficients as the perturbations become more severe. This may be explained by the higher discordance between epistemic estimates and f1 scores over increased severities of perturbation. It is observable from Table 6.3 and Figure 6.10 that, despite the considerably larger average linear spreads, these are constant over all evaluated data modalities whilst the uncertainty remains conformant to that measured in the previous procedure.

Table 6.5: Average linear spread obtained using the merging strategy of the Existence Probability, over all data modalities.

	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
average linear spread	0.0087	0.0080	0.0073	0.0097

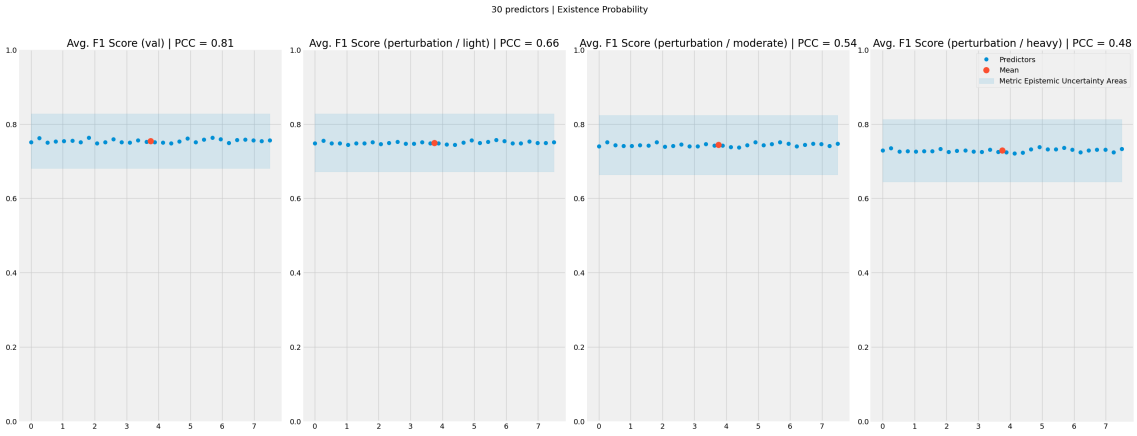


Figure 6.11: Predictors generated from training VirConv-L over the *seed*, *batch size* and *retraining from checkpoint* variants. F1 scores collected for Existence Probability outputs. The average metric epistemic uncertainty is plotted from the median predictor.

Procedure 6 - Existence Probability Experiments Lastly, the underspecification distribution is observed from the results of bounding boxes merged with the Existence Probability method, as showcased in Figure 6.11. Table 6.5 reports the average linear spread values obtained in these experiments. Similarly to *Procedure 4*, results demonstrate low values in average linear spread and, akin to *Procedure 5*, less variability of spread over perturbation severity is also observed. Since the Existence Probability merges bounding boxes as produced by the standard model with the Existence Map suggestions, the final outcomes include fine-grained features as well as information supported from two different mechanisms within the network. This leads the spread to stay very low despite the data uncertainty produced in perturbation and leads to similar accuracy values over all predictors even though these sport variable epistemic estimates. This is also a result of the calibration improvement produced by the Existence Probability merging strategy.

In this procedure, the metric epistemic uncertainty estimated using the proposed interpretation also covers the full span of predictor accuracy in the underspecification distribution. Despite the claims of being undersampled, the estimates cover a wider area, suggesting that, even under more prominent network parameter changes, the calculated epistemic uncertainty spans a sufficient area to cover A_φ fully.

Concerning the quantitative analysis of correlation, Table 6.6 reports the PCC, SCC and K-τ

Table 6.6: Measured values for each coefficient over all data modalities for the bounding boxes merged through the Existence Probability method.

Coefficient	Measured Value			
	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
PCC	0.81	0.66	0.54	0.48
SCC	0.75	0.57	0.58	0.49
K-τ	0.57	0.42	0.41	0.32

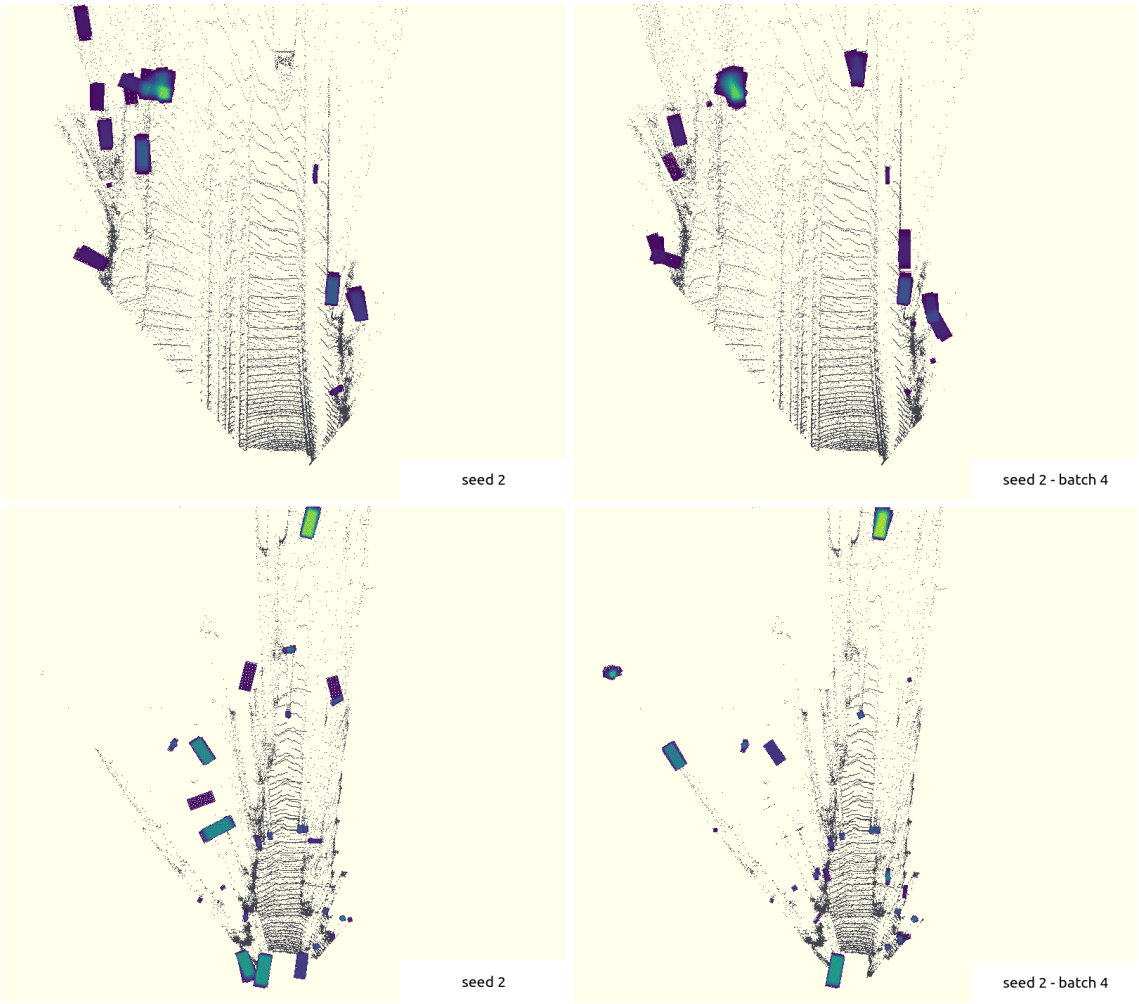


Figure 6.12: Different predictors generate diverging anchor distributions as seen in their Uncertainty Maps. The estimated uncertainties are equally impacted.

values obtained for the Existence Probability. Similarly to *Procedure 4*, a decrease in scores is seen over the severity of perturbation with even lower coefficients obtained in the *heavy* mode. Nevertheless, despite the outlier K-τ of 0.32, all the other correlations still depict relationships from *moderate* to *strong* strength with Kendall's measure sporting lower values for the same reasons as stated in prior procedures. In fact, since the merging strategy leads to higher uniformity over predictors (due to the inclusion of uncertainty estimates within the predictions), many repeated f1 scores are observed throughout Figure 6.11. Despite this, the PCC and SCC scores still both demonstrate a predominantly linear monotonic relationship, as concluded in previous experiments. The fact that the coefficients are slightly decreased over the other procedures may originate in less model sensitivity produced by the incorporation of uncertainty estimates, leading to more uniform results and thus reducing underspecification and its connection to the epistemic parcel. This demonstrates another advantage of the availability of uncertainty estimates, shown to result in the increased robustness of the model.

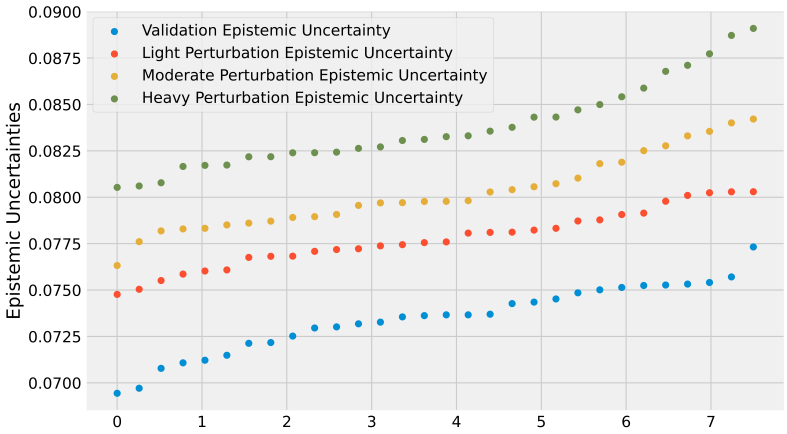


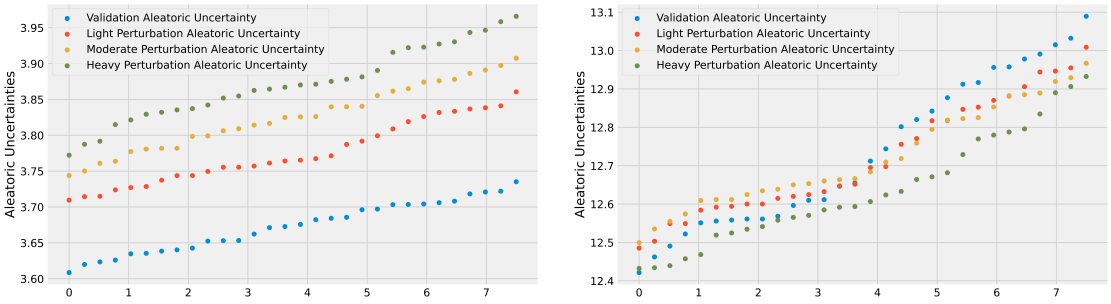
Figure 6.13: Epistemic uncertainties obtained for each predictor over the data modalities considered in the experimental procedures.

6.7.2 Uncertainty in-between Predictors

The aforementioned experiments focused on the underspecification distribution generated by using the outputs of both the standard model, the Existence Map and the Existence Probability. Further than the previously analysed aspects, the proposed interpretation of the epistemic uncertainty may also be observed in more detail.

From the generated predictors, some nuances stand out. Figure 6.12 demonstrates the Uncertainty Maps obtained by two different predictors on two input samples. As observed, the obtained anchor distributions diverge considerably. Whilst some areas share high aleatoric uncertainty prediction, others are singled out solely in some predictors. Typically, areas with low Existence Map probability correspond to anchor distributions with few samples, enabling an estimation of uncertainty to occur in these areas. Since the predicted probability of existence is quite low, these patches might be entirely left out in other predictors, leading the resulting Uncertainty Maps to diverge considerably. These divergences, as observed, directly tie to underspecification as either the suppression or maintenance of these detections in particular depends on the optimisation path of the predictor, enabling the observed variation in results. This is more prominently seen in out-of-distribution scenarios since, as uncertainty in detection outcomes rises, so does the sensibility of the optimisation path to the suppression or maintenance of these boxes.

Further than uncertainty maps, Figure 6.13 depicts the epistemic uncertainties obtained in each predictor over all data modalities. The estimates behave similarly between validation topology, a result coherent with the PCC, SCC and K-τ scores discussed in the aforementioned procedures. The first observation to be made is the estimation coherence between predictors. Despite the existence of some variability, results sport a range of 0.01 at its maximum, seen in the *moderate* and *heavy* perturbations. This stems from the use of both region proposal outcomes and the final layer outputs, enabling the summarisation of the knowledge encoded throughout the entire model's architecture. This stability in prediction seems to suggest that the proposed epistemic uncertainty model expresses the set of parameters possible for a given model architecture significantly.



(a) *direct entropy estimation* (b) *joint entropy estimation*

Figure 6.14: Aleatoric uncertainties obtained for each predictor over the data modalities considered in the experimental procedures. Both of the proposed aleatoric uncertainty models are depicted.

Another aspect to denote in Figure 6.13 is the increase in uncertainty levels as perturbations become more severe. This is coherent with expectation as more strongly perturbed data presents a more demarcated distribution shift compared to the training environment. Epistemic components of uncertainty are expected to in overall linearly follow distribution shifts as the change in underlying data presents challenges to the model's acquired knowledge during training.

Despite the aforementioned discussions being dedicated to the epistemic component of uncertainty, throughout the described experimental procedures, aleatoric uncertainties are also collected for both the *direct entropy estimation* and the *joint entropy estimation* models. In the fashion of the epistemic uncertainties, the aleatoric component is also plotted over the span of generated predictors, as shown in Figure 6.14.

Through the observation of Figure 6.14a, the range of aleatoric parcels estimated by the predictors using the *direct entropy estimation* model is found to be significantly larger than that found in the epistemic analysis. With a maximum range surrounding 0.2, more notable in higher degrees of perturbation, the *direct entropy estimation* showcases less consistency in the uncertainty calculation process. This once more, demonstrates the lack of statistical coherence of this interpretation, as denoted previously. Despite its lack of formal properties, as discussed in prior chapters, this model has several uses in application. On the other hand, the *joint entropy estimation* methodology reveals a considerably larger range between predictor estimates, surrounding 0.7, as showcased in Figure 6.14b. Even so, values of uncertainty with this model are much higher, which in reality suggests lower variability in comparison with the *direct entropy estimation* methodology. Further than the specific range, a surprising factor is the lack of a linear relationship between estimated aleatoric uncertainties and the degree of perturbation, with all data modality predictors calculating similar values. This, in turn, demonstrates the reduced sensibility of the *joint entropy estimation* model to data perturbations, impacting its performance negatively. Nevertheless, the statistical guarantees of this methodology lead to more consistent estimates between predictors, a desirable property as the aleatoric parcel should remain constant regardless of model and variate only with the data.

6.8 Discussion

In this chapter, six experimental procedures are reported, including the evaluation of a classification assignment and an object detection target. For the first problem, LeNet and ResNet-18 are utilised whilst, for the object detection modelling, the VirConv-L architecture is once more leveraged.

The underspecification distribution is observed, confirming its existence in the analysed contexts. Upon examination, it is concluded that tweaking different hyperparameters in deep learning models leads to substantially divergent results in the generated predictor distribution. From the tested variations, namely *seed*, *weight initialisation*, *batch size* and *retraining from checkpoint*, altering the *weight initialisation* induces far more spread in A_φ. The other variants result in similar distributions.

Another aspect of interest lies in the selected models for testing. A wide variety of networks are experimented with - LeNET, a proven well calibrated architecture, ResNet-18, a demonstrated miscalibrated model [Guo et al. (2017)] and VirConv-L, overconfident as observed in previous experiments. Pertaining to this analysis, results demonstrate that the calibration of the model does not impact its resistance to perturbation, out-of-distribution or underspecification phenomenons.

Further than these results, it is suggested that Monte Carlo Dropout underestimates uncertainty as confirmed in the literature [Folgoc et al. (2021); Verdoja and Kyrki (2021)]. Deep ensembles, on the other hand, prove much more efficient at predicting the spread of the underspecification distribution, even in more severe, variable environments.

Despite the spread being too low on the object detection experiments, the tested variants are commonly and frequently updated hyperparameters of neural networks. It is possible that, in many applications, the underspecification distribution spanned by the optimisation process would not express much further spread. Even in higher spread conditions, the results suggest that the proposed epistemic uncertainty interpretation does not suffer from underestimation. Compared to Monte Carlo Dropout, that struggles to even cover underspecification spread in validation with only one variant, namely *seed*, the proposed epistemic interpretation leads to much safer values, as confirmed by Figure 6.6 and Figure 6.9. This is a strongly positive factor of the proposed interpretation of the epistemic uncertainty as underestimation is a threat to the application of uncertainty quantification in deep learning for safety-critical conditions.

As observed in the procedures, the standard output and the Existence Probability merging strategy both lead to considerable low spread even through increasingly more severe perturbations. This stable performance might be connected with the strategy employed by the VirConv-L model to process the input data, relying on virtual points generated from merging the pointcloud with RGB and depth completion information. The addition of extra-sensor fused data to the perturbed pointcloud can lead to much less variability than that induced by the perturbation, enabling stable performance from different predictors. Existence maps, on the other hand, showcase much more variability and are demonstrated to be more liable towards perturbation. Since the region proposal layer features such heightened spread, the completeness of the virtual points solely is unable to

explain these results. In fact, these seem to suggest that it is within the detection head and post-processing that the network homogenises outcomes between predictors. This may demonstrate more deterministic behaviour from these layers. Furthermore, it showcases that the VirConv-L model, despite being severely overconfident, has significant interpretability in terms of underspecification.

Finally, overall, the outcomes of the procedures suggest that providing the range of accuracies estimated by the average-metric epistemic uncertainty as an interval of confidence to the accuracy of models could be a feasible approach to better indicate generic performance rather than communicating dataset-dependant metrics.

As for underspecification, even though its existence in machine learning pipelines has been reported, not much further literature exists. Teney et al. (2022) formalise the concept and propose a method to guarantee a better alignment between the data manifold and the model so as to enable higher out-of-distribution detection accuracy. The proposed methodology entails the training of multiple models with independence constraints so as to increase the chances of extracting features otherwise ignored. Even though the authors take a first step to address underspecification, their proposal only partially defines it over a small range of inputs expected during testing. Furthermore, there is still the need to train several models which proves more expensive than desirable.

Further than addressing improved performance in out-of-distribution, the experiments performed in this chapter aim to gain further understanding on the underspecification distribution and how it is expressed in practical application. In fact, the metric epistemic uncertainty is shown to have the potential to give a confidence interval to the spread of an architecture based on one predictor alone. This could enable effective mitigation of the underspecification phenomenon without the need to generate several predictors. This is confirmed by the evaluated correlation metrics, namely the Pearson's, Spearman's and Kendall's coefficients which concentrate between *very strong* and *strong* values, depicting a strongly linear monotonic relationship between the metric epistemic uncertainty and the spread of underspecification.

Another possible outcome of interest that has no prior literature is the assessment of the calibration of uncertainty estimates through underspecification. From the observed behaviours, if the underspecification distribution is effectively sampled, comparing the obtained epistemic uncertainty to the spread in predictor accuracy, A_φ, should indicate how well-calibrated the uncertainty estimation is and whether it is under or overestimated. Whilst calibration has been extensively addressed [Guo et al. (2017); Feng (2021)], as far as the literature review covers, no method utilises the underspecification distribution to this end. Nevertheless, as a calibration methodology, underspecification proves too resource consuming.

6.9 Chapter Summary

This chapter discusses the phenomenon of underspecification by characterising its distribution and studying its relationship with the average-metric epistemic uncertainty. Firstly, the problem definition is featured in Section 5.1 where the objective of the chapter is discussed alongside

aspects relevant to the proposals featured in the next sections. This is followed by the definition of the underspecification distribution alongside variables of interest in Section 6.2. Section 6.3 then discusses the models of uncertainty employed to obtain the average-metric epistemic uncertainty.

Focusing then on the experimental validation, Section 6.4 firstly discusses any relevant implementation details, starting with the definition of the utilised architectures, alongside the parameters selected for all employed methodologies, finalising with a report of the leveraged datasets and splits. To conclude the setup of the experimental procedures, Section 6.5 explores the metrics that are utilised for evaluation purposes.

The experimental procedures followed are reported in Section 6.6 and Section 6.7. The first part concerns evaluation of standard uncertainty quantification methods in a global classification problem whilst the following procedures experiment with the selected object detection model, VirConv-L, in an autonomous driving environment. The outcomes of both sections are thoroughly discussed in Section 6.8, including a comparison with existent literature.

Chapter 7

Conclusions

In this thesis, interpretations of uncertainty are studied in the context of deep learning models for object detection. The main targets of this analysis are to enable the improved robustness and interpretability of neural networks in the perception module of self-driving vehicles, a safety critical system. Further than aiming at developing novel uncertainty models and deriving applications where these estimates can be leveraged, the proposals target three different conditions of the utmost importance in safety-critical, autonomous systems: to be *model-independent*, to be *real-time compatible* and to be *applicable to deployed models* without the need for re-training.

Pertaining to this, the concept of existence is introduced, leveraging anchor distributions generated in the region proposal layer of an object detector. From this idea, two interpretations of aleatoric uncertainty are proposed, the *direct entropy estimation* and the *joint entropy estimation*, alongside an epistemic quantification method. The discussed models are based on the field of Information Theory and are built upon the concepts of the conditional entropy and mutual information. Taking the ideas of existence and of the suggested uncertainty interpretations, three applications are devised - the *Existence Map*, the *Uncertainty Map* and the *Existence Probability*.

As such, contributions of this thesis align over an intersection of fields, namely *uncertainty quantification*, *object detection* and *deep learning robustness and interpretability*. In this chapter, these domains are confronted, unveiling how the aforementioned proposals support these fields and fulfil the designed targets. In this sense, firstly, these contributions are reviewed, summarised and explored in a chapter breakdown, following with the future work that may enhance these even further.

7.1 Review of Contributions

Chapter 4 derives the *direct entropy estimation* and the *joint entropy estimation* interpretations of the aleatoric uncertainty. The building block behind both of these models is the conditional entropy that expresses the uncertainty left about a random variable Y given that the realisation of another variable, θ, is known [Ash (1965)]. The re-interpretation is performed to the random variables and, consequently, to the conditional distribution. Existence is interpreted as the unknown variable

whilst the given distribution is generated by the patches where existence is suspected. Through this concept, the *direct entropy estimation* interpretation directly obtains the probability density through the anchor distribution, measuring the entropy of the obtained probabilities. On the other hand, the *joint entropy estimation* model utilises the additive property of the entropy, considering that an anchor distribution is in fact a joint distribution of each anchor's categorical density.

On the topic of the epistemic uncertainty, a proposal based on the mutual information is discussed, where, once more, the anchor distributions are considered a source of disagreement to the final predictive distribution. Measuring the distance between these can quantify the level of agreement within the network itself, disregarding costly sampling procedures that either need a physical expression of sub-models [Lakshminarayanan et al. (2017); Matsubara et al. (2020)], sub-runs [Gal and Ghahramani (2016); Ayhan and Berens (2018)] or sub-parameters [Guo et al. (2017); Teye et al. (2018)]. The derivation of the proposed interpretation demonstrates that the given formulae culminates in expressing the difference in the estimators of aleatoric uncertainties generated by the region proposal and final layers of the network, amplified by the ratio between the predicted confidences.

These interpretations of uncertainty are thoroughly studied in Chapter 5, where extensive evaluation takes place. Correlations between the aleatoric models and factors such as distance to the ego vehicle, occlusion and difficulty are analysed, demonstrating the validity and coherence of the proposed interpretations. The *joint entropy estimation* is shown to have a higher disconnect from the observed factors, probably due to the summation of entropies that dilutes the correlation between each anchor proposal and the patch contents. Sources of randomness and noise in the pointcloud are also demonstrated to be highly complex, depending on a panoply of diverging aspects.

Further, in Chapter 5, the epistemic uncertainty interpretation is also observed to reflect high coherence with the accuracy of the model and with Existence Map confidence. These correlations demonstrate that estimates are *explainable* and *calibrated* since higher accuracies correspond with overall lower epistemic uncertainties. The observation of these relationships reflects on the interpretability of the network since, with the availability of uncertainty estimates, more information on these aspects can be deduced.

Finally, evaluation of calibration in comparison with staple literature methods, MC-Dropout, Deep Ensembles and Test-time Data Augmentation, shows that the conceptualised estimators enable uncertainties with significantly lower calibration errors, suggesting coherence and strength in the developed models.

In the specific case of the proposed aleatoric uncertainty interpretations, some limitations are observed. The correlation with occlusion and the lack of homogeneity between perturbation severities, especially, indicate some instability and sensitivity to the network's internal learning process. Whilst no uncertainty model estimated by a neural network can be entirely decoupled from the learned inner representations, higher consistency is desirable. Further, the use of the conditional entropy to estimate aleatoric uncertainties, as shown by Wimmer et al. (2023), is unable to entirely exclude the epistemic component.

Concerning then the proposed applications of uncertainty to enhance robustness and interpretability, in Chapter 4, the *Existence Map* is introduced. The Existence Map is a tool that encodes probabilistic information over an input sample, suggesting the existence of objects. Rather than operating within a closed-set classification assignment, thus being restricted to recognizing objects from predefined categories, the conceptualised existence operates over all objects regardless of their categorisation. As such, the presence of existence maps can enable the standard output of the model to be enriched with further information, even providing open-set classification possibilities [Yang et al. (2024a)]. In out-of-distribution data, in particular, the presence of extra-class knowledge is essential, as object representations may differ substantially from those learned in-distribution.

The evaluation of the Existence Map carried out in Chapter 5 reports the efficiency of the tool. In fact, in the test set and for all classes, quantifying precision and recall only with the patches suggested by these maps results in equal recall with only a decrease of 0.04 points in precision over the standard output. This demonstrates the effectiveness of the Existence Map in suggesting areas of interest where objects truly exist. Analysing samples demonstrates that the tool not only captures missed detections but also effectively represents heading and location regression uncertainty while rejecting standard false positives. Despite the lower perturbation accuracy, the Existence Map still achieves increased or approximately equal recall to that of the standard output. Once more, this suggests that this tool offers relevant insight to complete the common outcomes of object detection models. Even more, the information contained within these maps brings further insight into the model's learning process by probing the region proposal layer directly and representing this knowledge in a condensed, human-readable manner. The robustness and interpretability of the object detection model is thus greatly improved, as proven by the obtained results. A thorough search of the relevant literature yields no comparative method to that of the Existence Map.

The other application discussed in Chapter 4 is the *Uncertainty Map*. Conversely to the aforementioned maps, the Uncertainty Map is a tool that encodes aleatoric uncertainty over an input sample, revealing the estimates made by the model locally. In Chapter 5, these maps are shown to increase the interpretability of object detection, especially when combined with the Existence Map. Insights gained from the Uncertainty Map include but are not limited to the identification of heading and location regression uncertainty, reporting of noise and randomness in the input and providing suggestions of detections missed by the Existence Map. Especially for end-users of deep learning, the proposed representation may enable trustworthiness in autonomous systems such as self-driving vehicles.

The final proposal featured in Chapter 4 is the *Existence Probability*, a merging strategy that combines the outcomes of the standard model with the Existence Map, recalibrating detection confidence. The final classification probability is deduced over all classes of objects and integrates anchor categorical outcomes, aleatoric uncertainty and patch density. Once more, thorough analysis in Chapter 5 demonstrates that, employing the discussed merging strategy leads to improved precision and recall in 0.03 and 0.04 points respectively in testing. During out-of-distribution evaluations, the increased robustness lent by employing the Existence Probability leads to consid-

erably higher F1-scores over the standard output in all perturbation modes. From the reliability diagram assessment, the Existence Probability is observed to improve calibration over the standard model, albeit not at the desirable magnitude. Especially in comparison with the methods proposed by Guo et al. (2017), the explored recalibration does not prove as effective as seen in the literature, being a limitation of the merging strategy.

Chapter 6 features a study of the relationship between the underspecification distribution and the epistemic uncertainty. The underspecification phenomenon is present in most machine learning pipelines, showing that the interpretability of deep learning models is severely understudied when developing and deploying models [D'Amour et al. (2022); Teney et al. (2022)]. From the designed experimental procedures, it is revealed that Monte Carlo Dropout severely underestimates uncertainty in standard conditions, whilst ensembles provide much more calibrated and explainable estimates. In the face of this analysis, the proposed epistemic interpretation is proven to lead to accurate estimates, once more showing significantly better calibration than Monte Carlo Dropout in particular. These results also support the calibration error analysis carried out in Chapter 5.

Moreover, the performed evaluation showcases that there is a clear relationship between the epistemic uncertainty and the phenomenon of underspecification, mostly linear and monotonic. The proposed concept of utilising the metric epistemic estimates to quantify predictor spread from one single training session enables many paths to mitigating underspecification whilst avoiding the costs of running numerous sessions. Finally, a method to communicate the accuracy of state-of-the-art models is suggested, providing confidence intervals that better express the performance of architectures in in-and out-of-domain environments.

In overview of these contributions, the assessments performed showcase that the target conditions are met by all proposals, namely to be *model-independent*, to be *real-time compatible*, and to be *applicable to deployed models* without the need for re-training. The discussed uncertainty interpretations rely on anchor distributions already generated by most object detection models to enable sampling of the predictive distribution. Furthermore, the only unknowns needed for quantification are the categorical classifications of the top n anchors. This signifies that they are model-independent, being applicable to any proposal-based object detection model. Even more, their availability is only dependant on anchor generation, which allows for application to already deployed models. Finally, the processing needed to obtain uncertainty estimates relies only on simple mathematical operations over the confidences attributed to anchors, on filtering of the top n proposals and IoU operations. With careful implementation, these procedures are fully compatible with real-time processing in the deployment environment of self-driving vehicles. As such, and from the gathered existent uncertainty interpretations in the literature, no other methodology is as simple whilst leveraging the least amount of resources. Further, the adequacy of the obtained uncertainties is shown in the face of staple literature methods, with the aleatoric models sporting significantly low calibration errors. Especially in comparison to Monte Carlo Dropout, one of the most commonly applied techniques due to its simplicity [Gal and Ghahramani (2016); Feng et al. (2018); Ghoshal et al. (2020); Combalia et al. (2020)], the proposed epistemic interpretation enables considerably better calibrated estimates, as suggested by the experimentation in Chapter 5

and Chapter 6.

Finally, concerning the proposed applications of the uncertainty models, the same conditions are met. Existence and uncertainty maps are more memory intensive, needing the writing of matrices over three plus dimensions. Nevertheless, since these matrices are highly sparse, careful implementation can reduce this footprint considerably. Once again, the only unknowns are sourced directly from anchor distributions. The Existence Probability includes the collection of the standard output of the model, information already freely available upon deployment of any deep learning architecture. As such, these operations prove to be model-independent, real-time compatible and applicable to deployed models as well.

7.2 Future Work

From the overview of contributions, the identified directions of future work are as follows:

Uncertainty inconsistencies As previously discussed, the proposed interpretations of the aleatoric uncertainty showcase inconsistent results at times. Addressing this is a promising direction of future work.

Compatibility with further models Since the proposals utilise anchor proposals to approximate distributions, application to promising models based on Transformers Chen et al. (2023); Wu et al. (2024) is not direct. Conceptualising the anchor distribution approximation through positional queries in visual Transformers would be a relevant extension of this work, enabling these interpretations of uncertainty to be leveraged in a wider range of architectures.

Fine-grained regression The discussed uncertainty applications focus on the classification assignment of object detection, disregarding the regression component. As such, one of the major directions for future work is extending these tools to incorporate regression targets, especially in the Existence Probability. The merging strategy and the evaluation of the Existence Map assumes interception of bounding boxes from $IoU > 0$. For fine-grained, sophisticated object detection, however, high IoU in detections is essential. In this sense, it would be of interest to study the incorporation of this component of object detection into the proposed methodologies.

Existence Map rejection The discussed version of the Existence Map communicates existence when there is a probability distribution over an area and assumes non-existence in areas with no output. Regardless, from the conceptualisation of the Existence Map, no feedback does not entirely entail that no object exists. Further studying the rejection band of the Existence Map is another possible relevant study that could further improve the rate of false negatives.

Existence Probability calibration The evaluation showcased the potential of the Existence Probability to improve model calibration. Nevertheless, in comparison with the literature, as a calibration tool, the Existence Probability can be enhanced. Further fine-graining its confidence regression could enable better results and is a promising idea.

Underspecification From performed experiments, it is observed that the underspecification distribution varies strongly depending on model and environment. In this sense, further experiments in object detection contexts with variants that increase predictor spread could enable better assessment of the measure of calibration of the proposed epistemic interpretation.

Replication with other models Evaluation is carried out with a single state-of-the-art model. Extending the performed experimental procedures to other architectures is a pertinent direction of work.

Error recovery Another intended extension of this thesis is the incorporation of the obtained uncertainty estimates and Existence Map information into a planning layer to study error recovery capabilities.

References

M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, V. Makarenkov, and S. Nahavandi, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, vol. 76, pp. 243–297, dec 2021.

A. Amini, A. Soleimany, S. Karaman, and D. Rus, "Spatial uncertainty sampling for end-to-end control," *arXiv*, may 2018.

R. B. Ash, *Information Theory*. Dover Publications, 1965.

A. Atanov, A. Ashukha, D. Molchanov, K. Neklyudov, and D. Vetrov, "Uncertainty estimation via stochastic batch normalization," in *Advances in neural networks – ISNN 2019: 16th international symposium on neural networks, ISNN 2019, moscow, russia, july 10–12, 2019, proceedings, part I*, ser. Lecture notes in computer science, H. Lu, H. Tang, and Z. Wang, Eds. Springer International Publishing, 2019, vol. 11554, pp. 261–269.

E. Awad, S. Dsouza, A. Shariff, I. Rahwan, and J.-F. Bonnefon, "Universals and variations in moral decisions made in 42 countries by 70,000 participants." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 117, no. 5, pp. 2332–2337, feb 2020.

M. Ayhan and P. Berens, "Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks," *Medical Imaging with Deep Learning*, 2018.

L. Balasubramanian, F. Kruber, M. Botsch, and K. Deng, "Open-set recognition based on the combination of deep learning and ensemble method for detecting unknown traffic scenarios," *arXiv*, may 2021.

M. W. Baoyuan Liu, H. Foroosh, M. Tappen, and M. Penksy, "Sparse convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015, pp. 806–814.

W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler, "The power of ensembles for active learning in image classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2018, pp. 9368–9377.

C. M. Bishop, "Mixture density networks," *Technical Report*, 1994.

——, *Pattern recognition and machine learning*. Springer New York, 2006.

C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," *Proceedings of Machine Learning Research*, jun 2015.

A. Boiarov, O. Granichin, and O. Granichina, "Simultaneous perturbation stochastic approximation for few-shot learning," in *2020 European Control Conference (ECC)*. IEEE, may 2020, pp. 350–355.

J.-F. Bonnefon, A. Shariff, and I. Rahwan, "The social dilemma of autonomous vehicles." *Science*, vol. 352, no. 6293, pp. 1573–1576, jun 2016.

K. Brach, B. Sick, and O. Dürr, "Single shot MC dropout approximation," *arXiv*, jul 2020.

W. L. Buntine and A. S. Weigend, "Bayesian back-propagation," *Complex Systems*, 1991.

K. Bykov, M. M. C. Höhne, A. Creosteanu, K.-R. Müller, F. Klauschen, S. Nakajima, and M. Kloft, "Explaining bayesian neural networks," *arXiv*, 2021.

H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020, pp. 11 618–11 628.

X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "FUTR3D: A unified sensor fusion framework for 3D detection," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, jun 2023, pp. 172–181. [Online]. Available: https://ieeexplore.ieee.org/document/10208455/

Y. Chen, Z. Yuan, and B. Chen, "A novel method of integrating flexibility and stability for chemical processes under parametric uncertainties," in *13th international symposium on process systems engineering (PSE 2018)*, ser. Computer aided chemical engineering. Elsevier, 2018, vol. 44, pp. 211–216.

Y. Chen, Y. Li, X. Zhang, J. Sun, and J. Jia, "Focal sparse convolutional networks for 3D object detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2022, pp. 5418–5427.

Y. Choe, J. Shin, and N. Spencer, "Probabilistic interpretations of recurrent neural networks," Tech. Rep., 2017.

S. Choi, K. Lee, S. Lim, and S. Oh, "Uncertainty-aware learning from demonstration using mixture density networks with sampling-free variance modeling," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2018, pp. 6915–6922.

——, "Uncertainty-aware learning from demonstration using mixture density networks with sampling-free variance modeling," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2018, pp. 6915–6922.

A. D. Cobb and B. Jalaian, "Scaling hamiltonian monte carlo inference for bayesian neural networks with symmetric splitting," in *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, ser. Proceedings of Machine Learning Research, C. de Campos and M. H. Maathuis, Eds., vol. 161. PMLR, 27–30 Jul 2021, pp. 675–685.

M. Combalia, F. Hueto, S. Puig, J. Malvehy, and V. Vilaplana, "Uncertainty estimation in deep neural networks for dermoscopic image classification," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, jun 2020, pp. 3211–3220.

M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016, pp. 3213–3223.

B. C. Csáji and K. B. Kis, "Distribution-free uncertainty quantification for kernel methods by gradient perturbations," *Machine learning*, vol. 108, no. 8-9, pp. 1677–1699, sep 2019.

A. D'Amour, K. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, F. Hormozdiari, N. Houlsby, S. Hou, G. Jerfel, A. Karthikesalingam, M. Lucic, Y. Ma, C. McLean, D. Mincu, A. Mitani, A. Montanari, Z. Nado, V. Natarajan, C. Nielson, T. F. Osborne, R. Raman, K. Ramasamy, R. Sayres, J. Schrouff, M. Seneviratne, S. Sequeira, H. Suresh, V. Veitch, M. Vladymyrov, X. Wang, K. Webster, S. Yadlowsky, T. Yun, X. Zhai, and D. Sculley, "Underspecification presents challenges for credibility in modern machine learning," *Journal of Machine Learning Research*, 2022.

M. de la Riva and P. Mettes, "Bayesian 3D ConvNets for action recognition from few examples," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, oct 2019, pp. 1337–1343.

J. Delgado, F. A. Lucay, and F. D. Sepúlveda, "Experimental uncertainty analysis for the particle size distribution for better understanding of batch grinding process," *Minerals*, vol. 11, no. 8, p. 862, aug 2021.

J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel r-CNN: Towards high performance voxel-based 3D object detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, pp. 1201–1209, may 2021.

S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft, "Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 1184–1193.

Y. Dingyi, W. Haiyan, and Y. Kaiming, "State-of-the-art and trends of autonomous driving technology," in *2018 IEEE International Symposium on Innovation and Entrepreneurship*. IEEE, mar 2018, pp. 1–8.

H. P. Do, Y. Guo, A. J. Yoon, and K. S. Nayak, "Accuracy, uncertainty, and adaptability of automatic myocardial ASL segmentation using deep CNN." *Magnetic Resonance in Medicine*, vol. 83, no. 5, pp. 1863–1874, may 2020.

Z. Dong, H. Ji, X. Huang, W. Zhang, X. Zhan, and J. Chen, "PeP: a point enhanced painting method for unified point cloud tasks," *arXiv*, feb 2024.

U. Dorjsembe, J. H. Lee, B. Choi, and J. W. Song, "Sparsity increases uncertainty estimation in deep ensemble," *Computers*, vol. 10, no. 4, p. 54, apr 2021.

S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, sep 2022.

D. Feng, "Uncertainty estimation for object detection using deep learning approaches," *Universität Ulm*, 2021.

D. Feng, L. Rosenbaum, and K. Dietmayer, "Towards safe autonomous driving: capture uncertainty in the deep neural network for lidar 3D vehicle detection," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, nov 2018, pp. 3266–3273.

D. Feng, L. Rosenbaum, F. Timm, and K. Dietmayer, "Leveraging heteroscedastic aleatoric uncertainties for robust real-time LiDAR 3D object detection," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2019, pp. 1280–1287.

D. Feng, Y. Cao, L. Rosenbaum, F. Timm, and K. Dietmayer, "Leveraging uncertainties for deep multi-modal object detection in autonomous driving," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, oct 2020, pp. 877–884.

D. Feng, A. Harakeh, S. L. Waslander, and K. Dietmayer, "A review and comparative study on probabilistic object detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–20, 2021.

L. L. Folgoc, V. Baltatzis, S. Desai, A. Devaraj, S. Ellis, O. E. M. Manzanera, A. Nair, H. Qiu, J. Schnabel, and B. Glocker, "Is MC dropout bayesian?" *arXiv*, 2021.

A. Y. K. Foong, Y. Li, J. M. Hernández-Lobato, and R. E. Turner, "'in-between' uncertainty in bayesian neural networks," *arXiv*, 2019.

M. Fortunato, C. Blundell, and O. Vinyals, "Bayesian recurrent neural networks," *arXiv*, apr 2017.

Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," *Proceedings of The 33rd International Conference on Machine Learning, PMLR*, jun 2016.

A. Ganguly and S. W. F. Earp, "An introduction to variational inference," *arXiv*, 2021.

J. Gast and S. Roth, "Lightweight probabilistic deep networks." IEEE, jun 2018, pp. 3369–3378.

A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2012, pp. 3354–3361.

C. Geng, S.-J. Huang, and S. Chen, "Recent advances in open set recognition: A survey." *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3614–3631, oct 2021.

H. Gholamalinezhad and H. Khosravi, "Pooling methods in deep neural networks, a review," *arXiv*, 2020.

B. Ghoshal, A. Tucker, B. Sanghera, and W. Lup Wong, "Estimating uncertainty in deep learning for reporting confidence to clinicians in medical image segmentation and diseases detection," *Computational Intelligence*, oct 2020.

C. Gillmann, D. Saur, T. Wischgoll, and G. Scheuermann, "Uncertainty-aware visualization in medical imaging - a survey," *Computer Graphics Forum*, vol. 40, no. 3, pp. 665–689, jun 2021.

E. Goan and C. Fookes, "Bayesian neural networks: an introduction and survey," in *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018*, ser. Lecture notes in mathematics, K. L. Mengersen, P. Pudlo, and C. P. Robert, Eds. Springer International Publishing, 2020, vol. 2259, pp. 45–87.

Y. Gong, C. Liu, F. Yang, X. Cai, G. Wan, J. Chen, W. Zhang, and H. Wang, "Confidence calibration for intent detection via hyperspherical space and rebalanced accuracy-uncertainty loss," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, pp. 10 690–10 698, jun 2022.

A. Graves, "Practical variational inference for neural networks," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Eds., vol. 24. Curran Associates, Inc., 2011.

C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 1321–1330.

D. Hall, F. Dayoub, J. Skinner, H. Zhang, D. Miller, P. Corke, G. Carneiro, A. Angelova, and N. Sunderhauf, "Probabilistic object detection: definition and evaluation," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, mar 2020, pp. 1020–1029.

A. Harakeh, M. Smart, and S. L. Waslander, "BayesOD: A bayesian approach for uncertainty estimation in deep object detectors," *IEEE International Conference on Robotics and Automation*, mar 2019.

K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016, pp. 770–778.

W. He and Z. Jiang, "A survey on uncertainty quantification methods for deep neural networks: An uncertainty source perspective," *arXiv*, 2023.

Y. He, C. Zhu, J. Wang, M. Savvides, and X. Zhang, "Bounding box regression with uncertainty for accurate object detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2019, pp. 2883–2892.

D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," *Proceedings of the International Conference on Learning Representations*, 2019.

J. Hensman, M. Rattray, and N. Lawrence, "Fast variational inference in the conjugate exponential family," *Advances in Neural Information Processing Systems*, 2012.

S. Hernández, D. Vergara, M. Valdenegro-Toro, and F. Jorquera, "Improving predictive uncertainty estimation using dropout–hamiltonian monte carlo," *Soft computing*, vol. 24, no. 6, pp. 4307–4322, mar 2020.

G. Hess, C. Petersson, and L. Svensson, "Object detection as probabilistic set prediction," in *Computer vision – ECCV 2022: 17th european conference*, ser. Lecture notes in computer science. Springer Nature Switzerland, 2022, vol. 13670, pp. 550–566.

H. A. Hoang, D. C. Bui, and M. Yoo, "TSSTDet: Transformation-based 3-d object detection via a spatial shape transformer," *IEEE sensors journal*, pp. 1–1, 2024.

M. Hobbhahn, A. Kristiadi, and P. Hennig, "Fast predictive uncertainty for classification with bayesian deep networks," in *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, ser. Proceedings of Machine Learning Research, J. Cussens and K. Zhang, Eds., vol. 180. PMLR, 01–05 Aug 2022, pp. 822–832.

M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, 2013.

M. Hu, S. Wang, B. Li, S. Ning, L. Fan, and X. Gong, "Penet: towards precise and efficient image guided depth completion," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2021, pp. 13 656–13 662.

E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods," *Machine learning*, vol. 110, no. 3, pp. 457–506, mar 2021.

R. L. Iman and J. C. Helton, "An investigation of uncertainty and sensitivity analysis techniques for computer models," *Risk Analysis*, vol. 8, no. 1, pp. 71–90, mar 1988.

S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 448–456.

B. Jin, Y. Tan, Y. Chen, and A. Sangiovanni-Vincentelli, "Augmenting monte carlo dropout classification models with unsupervised learning tasks for detecting and diagnosing out-of-distribution faults," *ArXiv*, sep 2019.

M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," *Machine learning*, 1999.

R. E. Kass, B. P. Carlin, A. Gelman, and R. M. Neal, "Markov chain monte carlo in practice: A roundtable discussion," *The American Statistician*, vol. 52, no. 2, pp. 93–100, may 1998.

A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" *Advances in Neural Information Processing Systems*, 2017.

A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," in *Procedings of the British Machine Vision Conference 2017*. British Machine Vision Association, 2017, p. 57.

A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2018, pp. 7482–7491.

R. Kesten, M. Usman, J. Houston, T. Pandya, K. Nadhamuni, A. Ferreira, M. Yuan, B. Low, A. Jain, P. Ondruska, S. Omari, S. Shah, A. Kulkarni, A. Kazakova, C. Tao, L. Platinsky, W. Jiang, and V. Shet, "Level 5 perception dataset 2020," https://level-5.global/level5/data/, 2019.

A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic? does it matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, mar 2009.

M. Knolle, A. Ziller, D. Usynin, R. Braren, M. R. Makowski, D. Rueckert, and G. Kaissis, "Differentially private training of neural networks with langevin dynamics for calibrated predictive uncertainty," *arXiv*, 2021.

L. Kong, Y. Liu, X. Li, R. Chen, W. Zhang, J. Ren, L. Pan, K. Chen, and Z. Liu, "Robo3D: Towards robust and reliable 3D perception against corruptions," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, oct 2023, pp. 19 937–19 949.

A. Krizhevsky, "Learning multiple layers of features from tiny images," Tech. Rep., 2009.

D. Krueger, C.-W. Huang, R. Islam, R. Turner, A. Lacoste, and A. Courville, "Bayesian hypernetworks," *Second workshop on Bayesian Deep Learning (NIPS 2017)*, 2017.

F. Küppers, A. Haselhoff, J. Kronenberger, and J. Schneider, "Confidence calibration for object detection and segmentation," in *Deep neural networks and data for automated driving: robustness, uncertainty quantification, and insights towards safety*. Springer International Publishing, 2022, pp. 225–250.

S. Lahlou, M. Jain, H. Nekoei, V. I. Butoi, P. Bertin, J. Rector-Brooks, M. Korablyov, and Y. Bengio, "DEUP: Direct epistemic uncertainty prediction," *Transactions on Machine Learning Research*, 2023.

B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Proceedings of the 31st International Conference on Neural Information Processing Systems*, p. 6405–6416, dec 2017.

A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: fast encoders for object detection from point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2019, pp. 12 689–12 697.

M.-H. Laves, S. Ihler, J. F. Fast, L. A. Kahrs, and T. Ortmaier, "Well-calibrated regression uncertainty in medical imaging with deep learning," *Proceedings of Machine Learning Research*, sep 2020.

Y. Le and X. S. Yang, "Tiny ImageNet visual recognition challenge," Tech. Rep., 2015.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

D. Levi, L. Gispan, N. Giladi, and E. Fetaya, "Evaluating and calibrating uncertainty prediction in regression tasks." *Sensors (Basel, Switzerland)*, vol. 22, no. 15, jul 2022.

L. Li, A. Holbrook, B. Shahbaba, and P. Baldi, "Neural network gradient hamiltonian monte carlo." *Computational statistics*, vol. 34, no. 1, pp. 281–299, mar 2019.

X. Li, T. Ma, Y. Hou, B. Shi, Y. Yang, Y. Liu, X. Wu, Q. Chen, Y. Li, Y. Qiao, and L. He, "LoGoNet: Towards accurate 3D object detection with local-to-global cross- modal fusion," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2023, pp. 17 524–17 534.

Y. Li, X. Qi, Y. Chen, L. Wang, Z. Li, J. Sun, and J. Jia, "Voxel field fusion for 3D object detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2022, pp. 1110–1119.

Z. Li, T. Zhang, S. Cheng, J. Zhu, and J. Li, "Stochastic gradient hamiltonian monte carlo with variance reduction for bayesian inference," *Machine learning*, vol. 108, no. 8-9, pp. 1701–1727, sep 2019.

Z. Lin, Y. Shen, S. Zhou, S. Chen, and N. Zheng, "MLF-DET: Multi-level fusion for cross-modal 3D object detection," in *Artificial neural networks and machine learning – ICANN*, ser. Lecture notes in computer science. Springer Nature Switzerland, 2023, vol. 14260, pp. 136–149.

H. Liu, R. Ji, J. Li, B. Zhang, Y. Gao, Y. Wu, and F. Huang, "Universal adversarial perturbation via prior driven uncertainty approximation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, oct 2019, pp. 2941–2949.

H. Liu, J. Du, Y. Zhang, H. Zhang, and J. Zeng, "PVConvNet: Pixel-voxel sparse convolution for multimodal 3D object detection," *Pattern recognition*, vol. 149, p. 110284, may 2024.

J. Liu, Z. Lin, S. Padhy, D. Tran, T. Bedrax Weiss, and B. Lakshminarayanan, "Simple and principled uncertainty estimation with deterministic deep learning via distance awareness," *Advances in Neural Information Processing Systems*, 2020.

W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *European Conference on Computer VIsion (ECCV)*, ser. Lecture notes in computer science, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Springer International Publishing, 2016, vol. 9905, pp. 21–37.

A. Loquercio, M. Segu, and D. Scaramuzza, "A general framework for uncertainty estimation in deep learning," *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 3153–3160, apr 2020.

C. Ma, Z. Huang, J. Xian, M. Gao, and J. Xu, "Improving uncertainty calibration of deep neural networks via truth discovery and geometric optimization," *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, dec 2021.

Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "TrafficPredict: Trajectory prediction for heterogeneous traffic-agents," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6120–6127, jul 2019.

A. Malinin, "Uncertainty estimation in deep learning with application to spoken language assessment," *Apollo - University of Cambridge Repository*, 2019.

A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," *Advances in Neural Information Processing Systems*, 2018.

I. Manivannan, "A comparative study of uncertainty estimation methods in deep learning based classification models," *Hochschule Bonn-Rhein-Sieg*, 2020.

F. C. Maruccio, W. Eppinga, M.-H. Laves, R. F. Navarro, M. Salvi, F. Molinari, and P. Papaconstadopoulos, "Clinical assessment of deep learning-based uncertainty maps in lung cancer segmentation." *Physics in Medicine and Biology*, vol. 69, no. 3, jan 2024.

Y. Matsubara and M. Levorato, "Neural compression and filtering for edge-assisted real-time object detection in challenged networks," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, jan 2021, pp. 2272–2279.

Y. Matsubara, D. Callegaro, S. Baidya, M. Levorato, and S. Singh, "Head network distillation: Splitting distilled deep neural networks for resource-constrained edge computing systems," *IEEE access : practical innovations, open solutions*, vol. 8, pp. 212 177–212 193, 2020.

K. Matsunami, S. Tanabe, H. Nakagawa, M. Hirao, and H. Sugiyama, "Economic evaluation of batch and continuous manufacturing technologies for solid drug products during clinical development," in *13th international symposium on process systems engineering (PSE 2018)*, ser. Computer aided chemical engineering. Elsevier, 2018, vol. 44, pp. 2131–2136.

P. McClure and N. Kriegeskorte, "Representing inferential uncertainty in deep neural networks through sampling," *International Conference on Learning Representations*, vol. 2017-Conference Track Proceedings, nov 2016.

McKinsey, "The autonomous vehicle industry moving forward," 2023, last accessed 19 January 2025. [Online]. Available: https://www.mckinsey.com/features/mckinsey-center-for-future-mobility/our-insights/autonomous-vehicles-moving-forward-perspectives-from-industry-leaders

G. P. Meyer, A. Laddha, E. Kee, C. Vallespi-Gonzalez, and C. K. Wellington, "Lasernet: an efficient probabilistic 3D object detector for autonomous driving," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2019, pp. 12 669–12 678.

R. Michelmore, M. Kwiatkowska, and Y. Gal, "Evaluating uncertainty quantification in end-to-end autonomous driving control," *arXiv*, nov 2018.

D. Miller, F. Dayoub, M. Milford, and N. Sunderhauf, "Evaluating merging strategies for sampling-based uncertainty techniques in object detection," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, may 2019, pp. 2348–2354.

S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey." *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 7, pp. 3523–3542, jul 2022.

R. J. Moffat, "Describing the uncertainties in experimental results," *Experimental Thermal and Fluid Science*, vol. 1, no. 1, pp. 3–17, jan 1988.

N. Moshkov, B. Mathe, A. Kertesz-Farkas, R. Hollandi, and P. Horvath, "Test-time augmentation for deep learning-based cell segmentation on microscopy images." *Scientific Reports*, vol. 10, no. 1, p. 5068, mar 2020.

T. Nair, D. Precup, D. L. Arnold, and T. Arbel, "Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018: 21st International Conference*, ser. Lecture notes in computer science. Springer International Publishing, 2018, vol. 11070, pp. 655–663.

J. Nazarovs, Z. Huang, S. Tasneeyapant, R. Chakraborty, and V. Singh, "Understanding uncertainty maps in vision with statistical testing." *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2022, pp. 406–416, jun 2022.

R. M. Neal, *Bayesian learning for neural networks*, ser. Lecture notes in statistics. New York, NY: Springer New York, 1996, vol. 118.

L. Neumann, A. Zisserman, and A. Vedaldi, "Relaxed softmax: Efficient confidence auto-calibration for safe pedestrian detection," *NeurIPS Workshop*, 2018.

H. Ney, "On the probabilistic interpretation of neural network classifiers and discriminative training criteria," *IEEE transactions on pattern analysis and machine intelligence*, vol. 17, no. 2, pp. 107–119, 1995.

M. Ng, F. Guo, L. Biswas, and G. A. Wright, "Estimating uncertainty in neural networks for segmentation quality control," *Technical Report*, 2018.

A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2015, pp. 427–436.

V.-L. Nguyen, M. H. Shaker, and E. Hüllermeier, "How to measure uncertainty in uncertainty sampling for active learning," *Machine learning*, jun 2021.

M. H. M. Noor and A. O. Ige, "A survey on deep learning and state-of-the-arts applications," *arXiv*, 2024.

C. Novelli, M. Taddeo, and L. Floridi, "Accountability in artificial intelligence: what it is and how it works," *SSRN Electronic Journal*, 2022.

OpenAI, "ChatGPT: Optimizing language models for dialogue," https://openai.com/blog/chatgpt/, 2022, accessed: Fev 10, 2023.

I. Osband, Z. Wen, S. M. Asghari, V. Dwaracherla, M. Ibrahimi, X. Lu, and B. Van Roy, "Epistemic neural networks," *Advances in Neural Information Processing Systems*, dec 2023.

G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection," *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, apr 2021.

X. Pang, Z. Zhao, and Y. Weng, "The role and impact of deep learning methods in computer-aided diagnosis using gastrointestinal endoscopy," *Diagnostics (Basel)*, vol. 11, no. 4, apr 2021.

N. Pawlowski, A. Brock, M. C. H. Lee, M. Rajchl, and B. Glocker, "Implicit weight uncertainty in neural networks," *Second workshop on Bayesian Deep Learning (NIPS 2017)*, 2017.

T. Pearce, M. Zaki, A. Brintrup, and A. Neely, "High-quality prediction intervals for deep learning: A distribution-free, ensembled approach," *Proceedings of Machine Learning Research*, feb 2018.

B. Phan, S. Khan, R. Salay, and K. Czarnecki, "Bayesian uncertainty quantification with synthetic data," in *Computer safety, reliability, and security: SAFECOMP workshops*, ser. Lecture notes in computer science. Springer International Publishing, 2019, vol. 11699, pp. 378–390.

J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in large margin classifiers*, pp. 61–74, 1999.

C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in Neural Information Processing Systems*, 2017.

R. Qian, X. Lai, and X. Li, "BADet: Boundary-aware 3D object detection from point clouds," *Pattern recognition*, vol. 125, p. 108524, may 2022.

R. Rahaman and A. Thiery, "Uncertainty quantification and deep ensembles," *Advances in Neural Information Processing Systems*, dec 2021.

A. Raj and F. Bach, "Convergence of uncertainty sampling for active learning," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 18 310–18 331.

S. Ranftl and W. von der Linden, "Bayesian surrogate analysis and uncertainty propagation," in *The 40th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*. Basel Switzerland: MDPI, nov 2021, p. 6.

R. Raut, A. D. Mihovska, and D. Rine, Eds., *Examining the Impact of Deep Learning and IoT on Multi-Industry Applications*, ser. Advances in web technologies and engineering. IGI Global, 2021.

S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-CNN: Towards real-time object detection with region proposal networks." *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, jun 2017.

T. Riedlinger, M. Rottmann, M. Schubert, and H. Gottschalk, "Gradient-based quantification of epistemic uncertainty for deep object detectors," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, jan 2023, pp. 3910–3920.

A. Rosenfeld and M. Thurston, "Edge and curve detection for visual scene analysis," *IEEE Transactions on Computers*, vol. C-20, no. 5, pp. 562–569, may 1971.

C. Rudin and J. Radin, "Why are we using black box models in AI when we don't need to? a lesson from an explainable AI competition," *Harvard Data Science Review*, vol. 1, no. 2, nov 2019.

T. S. Salem, H. Langseth, and H. Ramampiaro, "Prediction intervals: Split normal mixture from quality-driven deep ensembles," in *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, ser. Proceedings of Machine Learning Research, vol. 124. PMLR, 03–06 Aug 2020, pp. 1179–1187.

M. Salehi, H. Mirzaei, D. Hendrycks, Y. Li, M. Rohban, and M. Sabokrou, "A unified survey on anomaly, novelty, open-set, and out-of-distribution detection: Solutions and future challenges," *Trans. Mach. Learn. Res.*, 2021.

T. J. Santner, B. J. Williams, and W. I. Notz, *The design and analysis of computer experiments*, ser. Springer series in statistics. New York, NY: Springer New York, 2003.

R. Seoh, "Qualitative analysis of monte carlo dropout," *arXiv*, 2020.

M. Shacham, N. Brauner, and M. B. Cutlip, "Considering physical property uncertainties in process design," in *22nd european symposium on computer aided process engineering*, ser. Computer aided chemical engineering. Elsevier, 2012, vol. 30, pp. 607–611.

S. Shafaei, S. Kugele, M. H. Osman, and A. Knoll, "Uncertainty in machine learning: A safety perspective on autonomous driving," in *Developments in Language Theory: 22nd International Conference, DLT 2018, Tokyo, Japan, September 10-14, 2018, Proceedings*, ser. Lecture notes in computer science, M. Hoshi and S. Seki, Eds. Springer International Publishing, 2018, vol. 11088, pp. 458–464.

C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, jul 1948.

A. Shariff, J.-F. Bonnefon, and I. Rahwan, "Psychological roadblocks to the adoption of self-driving vehicles," *Nature Human Behaviour*, vol. 1, no. 10, pp. 694–696, oct 2017.

M. Sharma and M. Bilgic, "Evidence-based uncertainty sampling for active learning," *Data mining and knowledge discovery*, vol. 31, no. 1, pp. 164–202, jan 2017.

S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "PV-RCNN: Point-voxel feature set abstraction for 3D object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020, pp. 10 526–10 535.

W. Shi and R. Rajkumar, "Point-GNN: Graph neural network for 3D object detection in a point cloud," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020, pp. 1708–1716.

J. Shin, "Finite-element approximation of a fourth-order differential equation," *Computers & Mathematics with Applications*, vol. 35, no. 8, pp. 95–100, apr 1998.

S. Sinha, H. Bharadhwaj, A. Goyal, H. Larochelle, A. Garg, and F. Shkurti, "DIBS: diversity inducing information bottleneck in model ensembles," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, pp. 9666–9674, may 2021.

K. Sirohi, S. Marvi, D. Büscher, and W. Burgard, "Uncertainty-aware panoptic segmentation," *IEEE robotics and automation letters*, vol. 8, no. 5, pp. 2629–2636, may 2023.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, 2014.

J. Sun and D. W. Jacobs, "Seeing what is not there: learning context to determine where objects are missing," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jul 2017, pp. 1234–1242.

P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: waymo open dataset," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020, pp. 2443–2451.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *International Conference on Learning Representations*, 2014.

N. Tagasovska and D. Lopez-Paz, "Frequentist uncertainty estimates for deep learning," *Third workshop on Bayesian Deep Learning (NIPS 2018)*, 2018. [Online]. Available: http://bayesiandeeplearning.org/2018/index.html

D. Tannenbaum, C. R. Fox, and G. Ülkümen, "Judgment extremity and accuracy under epistemic vs. aleatory uncertainty," *Management Science*, vol. 63, no. 2, pp. 497–518, feb 2017.

O. D. Team, "Openpcdet: An open-source toolbox for 3d object detection from point clouds," https://github.com/open-mmlab/OpenPCDet, 2020.

Y. W. Teh, A. H. Thiery, and S. J. Vollmer, "Consistency and fluctuations for stochastic gradient langevin dynamics," *Journal of Machine Learning Research*, vol. 17, no. 7, pp. 1–33, 2016.

D. Teney, M. Peyrard, and E. Abbasnejad, "Predicting is not understanding: Recognizing and addressing underspecification in machine learning," in *The European Conference on Computer Vision ECCV 2022*, ser. Lecture notes in computer science, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds. Springer Nature Switzerland, 2022, vol. 13683, pp. 458–476.

M. Teye, H. Azizpour, and K. Smith, "Bayesian uncertainty estimation for batch normalized deep networks," *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, pp. 4907–4916, jul 2018.

L. Tran, B. S. Veeling, K. Roth, J. Swiatkowski, J. V. Dillon, J. Snoek, S. Mandt, T. Salimans, S. Nowozin, and R. Jenatton, "Hydra: Preserving ensemble diversity for model distillation," *ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning*, jul 2020.

S. Uchinoura, J. Miyao, and T. Kurita, "An object detection method using probability maps for instance segmentation to mask background," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 27, no. 5, pp. 886–895, sep 2023.

G. Ülkümen, C. R. Fox, and B. F. Malle, "Two dimensions of subjective uncertainty: Clues from natural language." *Journal of Experimental Psychology: General*, vol. 145, no. 10, pp. 1280–1297, oct 2016.

M. Valdenegro-Toro and D. S. Mori, "A deeper look into aleatoric and epistemic uncertainty disentanglement," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, jun 2022, pp. 1508–1516.

S. Vaze, K. Han, A. Vedaldi, and A. Zisserman, "Open-set recognition: A good closed-set classifier is all you need," *International Conference on Learning Representations*, 2021.

F. Verdoja and V. Kyrki, "Notes on the behavior of MC dropout," *ICML 2021 Workshop on "Uncertainty and Robustness in Deep Learning"*, 2021.

W. Walker, P. Harremoës, J. Rotmans, J. van der Sluijs, M. van Asselt, P. Janssen, and M. Krayer von Krauss, "Defining uncertainty: A conceptual basis for uncertainty management in model-based decision support," *Integrated Assessment*, vol. 4, no. 1, pp. 5–17, mar 2003.

Q. Wan, R. Wang, and X. Chen, "Interpretable object recognition by semantic prototype analysis," in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, jan 2024, pp. 789–798.

G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, and T. Vercauteren, "Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks." *Neurocomputing*, vol. 335, pp. 34–45, sep 2019.

T. Wang, X. Zheng, L. Zhang, Z. Cui, and C. Xu, "A graph-based interpretability method for deep neural networks," *Neurocomputing*, vol. 555, p. 126651, oct 2023.

X. Wang, K. Li, and A. Chehri, "Multi-sensor fusion technology for 3D object detection in autonomous driving: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 2, pp. 1148–1165, feb 2024.

Y. Wang, B. Li, T. Che, K. Zhou, Z. Liu, and D. Li, "Energy-based open-world uncertainty modeling for confidence calibration," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, oct 2021, pp. 9282–9291.

H. J. Weideman, "Quantifying uncertainty in neural networks," https://hjweide.github.io/quantifying-uncertainty-in-neural-networks, 2016, accessed: Dec 9, 2022.

Y. Wen, P. Vicol, J. Ba, D. Tran, and R. Grosse, "Flipout: Efficient pseudo-independent weight perturbations on mini-batches," *International Conference on Learning Representations*, 2018.

F. Wenzel, J. Snoek, D. Tran, and R. Jenatton, "Hyperparameter ensembles for robustness and uncertainty quantification," *Advances in Neural Information Processing Systems*, 2020.

O. Willers, S. Sudholt, S. Raafatnia, and S. Abrecht, "Safety concerns and mitigation approaches regarding the use of deep learning in safety-critical perception tasks," in *Computer safety, reliability, and security. SAFECOMP 2020 workshops*, ser. Lecture notes in computer science. Springer International Publishing, 2020, vol. 12235, pp. 336–350.

A. G. Wilson and P. Izmailov, "Bayesian deep learning and a probabilistic perspective of generalization," *NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, p. 4697–4708, 2020.

L. Wimmer, Y. Sale, P. Hofman, B. Bischl, and E. Hüllermeier, "Quantifying aleatoric and epistemic uncertainty in machine learning: Are conditional entropy and mutual information appropriate measures?" *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, jul 2023.

H. Wu, J. Deng, C. Wen, X. Li, C. Wang, and J. Li, "CasA: A cascade attention network for 3-d object detection from LiDAR point clouds," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2022.

H. Wu, C. Wen, W. Li, X. Li, R. Yang, and C. Wang, "Transformation-equivariant 3D object detection for autonomous driving," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 3, pp. 2795–2802, jun 2023.

H. Wu, C. Wen, S. Shi, X. Li, and C. Wang, "Virtual sparse convolution for multimodal 3D object detection," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2023, pp. 21 653–21 662.

T. Wu and X. Song, "Towards interpretable object detection by unfolding latent structures," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, oct 2019, pp. 6032–6042.

X. Wu, L. Peng, H. Yang, L. Xie, C. Huang, C. Deng, H. Liu, and D. Cai, "Sparse fuse dense: Towards high quality 3D detection with depth completion," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2022, pp. 5408–5417.

X. Wu, L. Jiang, P.-S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, and H. Zhao, "Point transformer v3: simpler, faster, stronger," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2024, pp. 4840–4851. [Online]. Available: https://ieeexplore.ieee.org/document/10658198/

Z. Xie, X. Yu, X. Gao, K. Li, and S. Shen, "Recent advances in conventional and deep learning-based depth completion: A survey." *IEEE transactions on neural networks and learning systems*, vol. PP, sep 2022.

H. Yang, Z. Liu, X. Wu, W. Wang, W. Qian, X. He, and D. Cai, "Graph r-CNN: Towards accurate 3D object detection with semantic-decorated local graph," in *Computer vision – ECCV 2022: 17th european conference*, ser. Lecture notes in computer science. Cham: Springer Nature Switzerland, 2022, vol. 13668, pp. 662–679.

J. Yang, K. Zhou, Y. Li, and Z. Liu, "Generalized out-of-distribution detection: A survey," *International journal of computer vision*, jun 2024.

Y. Yang and M. Loog, "Active learning using uncertainty information," in *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, dec 2016, pp. 2646–2651.

Z. Yang, J. Yue, P. Ghamisi, S. Zhang, J. Ma, and L. Fang, "Open set recognition in real world," *International journal of computer vision*, vol. 132, no. 8, pp. 3208–3231, aug 2024.

S. H. Yelleni, D. Kumari, S. P.K., and K. M. C., "Monte carlo DropBlock for modeling uncertainty in object detection," *Pattern recognition*, vol. 146, p. 110003, feb 2024.

F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "BDD100K: A diverse driving dataset for heterogeneous multitask learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020, pp. 2633–2642.

L. Yu, S. Wang, X. Li, C.-W. Fu, and P.-A. Heng, "Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation," in *Medical image computing and computer assisted intervention – MICCAI 2019: 22nd international conference*, ser. Lecture notes in computer science. Springer International Publishing, 2019, vol. 11765, pp. 605–613.

P. Yun and M. Liu, "Laplace approximation based epistemic uncertainty estimation in 3D object detection," *Proceedings of Machine Learning Research*, mar 2023.

B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '02*. New York, New York, USA: ACM Press, jul 2002, p. 694.

B. Zadrozny and C. P. Elkan, "Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers," *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001.

S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital signal processing*, vol. 126, p. 103514, jun 2022.

C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt, "Advances in variational inference." *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 2008–2026, aug 2019.

Y. Zhang, P. Tino, A. Leonardis, and K. Tang, "A survey on neural network interpretability," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 5, pp. 726–742, oct 2021.

X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, p. 99, mar 2024.

Y. Zhong, J. Wang, J. Peng, and L. Zhang, "Anchor box optimization for object detection," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, mar 2020, pp. 1275–1283.

O. Zohar, K.-C. Wang, and S. Yeung, "PROB: probabilistic objectness for open world object detection," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2023, pp. 11 444–11 453.

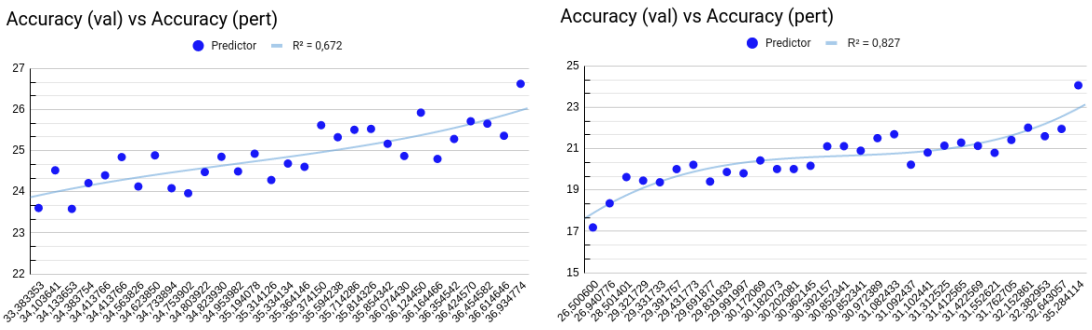
Appendix A

Behaviour of the Underspecification Predictors

Further observations supporting Chapter 6 on the connection between the epistemic uncertainty and the behaviour showcased by the predictors in a underspecification distribution are collected. The following reports concern procedures 1 through 3, namely, utilising Monte Carlo Dropout and Ensembles for uncertainty quantification as well as the standard entropy interpretation.

Firstly, the relationship between accuracy on validation and perturbation sets is examined in Figure A.1. Predictors with higher validation accuracy generally perform better in perturbation settings. Despite this, no consistent pattern emerges, suggesting that it remains difficult to predict which model will perform best in out-of-distribution.

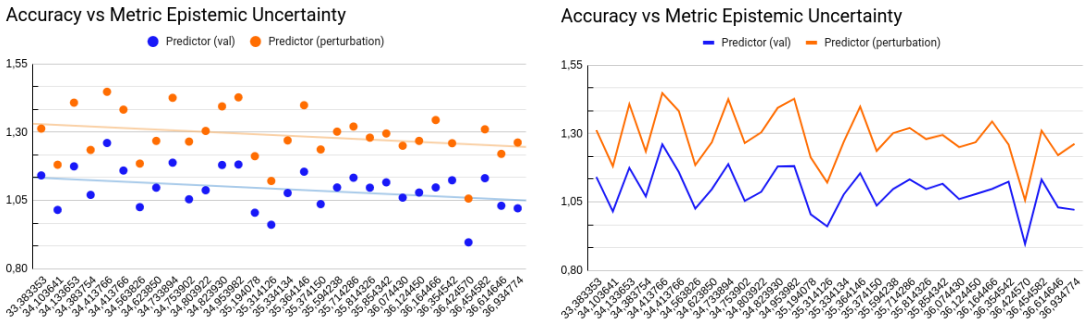
Next, the relationship between accuracy and metric epistemic uncertainty is reported in Figure A.2. While uncertainty is expected to decrease as accuracy increases, Figure A.2 shows significant variability. This aligns with the results obtained on the CIFAR procedures - although epistemic uncertainty offers insight into model information, depending on the method utilised for its quantification, estimates must be cautiously leveraged in underspecified models. Interestingly,



(a) Accuracy in the validation set versus in the perturbation set (*seed* variation).

(b) Accuracy in the validation set versus in the perturbation set (uniform xavier *weight initialisation*).

Figure A.1: Predictors with higher validation accuracy generally perform better in perturbation settings.



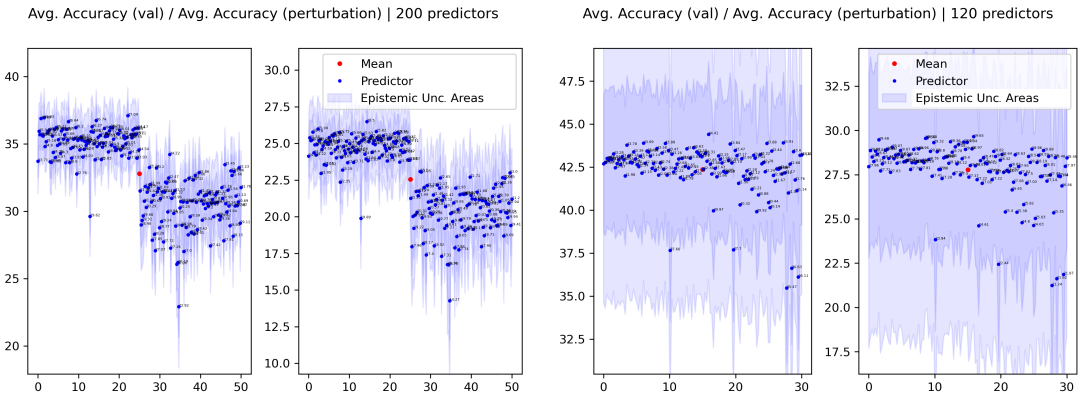
(a) Accuracy versus the metric epistemic uncertainty for validation and perturbation tests (*seed* variation).

(b) Accuracy versus the metric epistemic uncertainty for validation and perturbation tests (*seed* variation) with the points connected.

Figure A.2: Epistemic uncertainty follows the same shape even in perturbation, simply with a steady increase.

as shown in Figure A.2b, each predictor maintains its validation pattern in perturbation tests, albeit with slightly higher uncertainty.

Finally, Figure A.3 highlights the metric epistemic uncertainty for each predictor. The Monte Carlo Dropout method shows a narrower range compared to the Ensemble method, that provides a broader estimation of predictor accuracy. Ensembles may offer a faster method to estimate underspecification by sampling a few points from the distribution, aiding during model development.



(a) Sampled underspecification distribution with 200 predictors. The metric epistemic uncertainty, calculated with Monte Carlo Dropout, is plotted from each predictor.

(b) Sampled underspecification distribution with 120 predictors. The metric epistemic uncertainty, calculated with ensembles, is plotted from each predictor.

Figure A.3: Sampled underspecification distributions in the Monte Carlo Dropout and Ensemble experiments. The metric epistemic uncertainty is not averaged and is plotted from each obtained predictor.

Appendix B

Uncertainty over Predictors

Predictors generated by training the same architecture under different conditions, namely under *seed*, *batch size* and *retraining from checkpoint* variations, feature different predictions of uncertainty. Table B.1 demonstrates the uncertainty results obtained from two divergent predictors over the considered data modalities, namely validation and light, moderate and heavy perturbations.

As it can be observed, both predictors utilise different seeds and batch sizes. The variation observed in the uncertainty estimates is not as considerable as expected, with the highest distance found in *moderate* perturbation. For the *heavy* perturbation mode, estimates are equal. For the remainder modalities, whilst the aleatoric component is estimated rather similarly, the epistemic parcel shows more variation. This is coherent with expectation since the aleatoric component of uncertainty originates in the data, meaning that all predictors should sport increasingly similar estimates. In the case of the epistemic parcel, since it depends on the model's knowledge and learning outcomes, further variance should be seen. Confirming other observations, the *moderate* perturbation also showcases in this experiment that it produces higher disagreement within the model, akin to a threshold upon which the architecture is unable to process with consistency.

Table B.1: Uncertainties estimated by two different predictors, *seed2-batch4* and *seed333* over the considered data modalities.

seed2-batch4	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
Aleatoric	3.73	3.75	3.75	3.89
Epistemic	3.67	3.40	3.29	4.89

seed333	Validation	Light Perturbation	Moderate Perturbation	Heavy Perturbation
Aleatoric	3.64	3.76	3.84	3.89
Epistemic	3.79	3.55	3.45	4.89