D 2014

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

HUMAN-CENTRIC SENSING AND UNDERSTANDING USING VISION AND WI-FI SIGNALS

LEONARDO GOMES CAPOZZI

TESE DE DOUTORAMENTO APRESENTADA
À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM
ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Human-Centric Sensing and Understanding Using Vision and Wi-Fi Signals

Leonardo Gomes Capozzi

Doctoral Program in Electrical and Computer Engineering

Supervisor: Ana Rebelo, PhD

Second Supervisor: Jaime dos Santos Cardoso, PhD

June 3, 2026

Human-Centric Sensing and Understanding Using Vision and Wi-Fi Signals

Leonardo Gomes Capozzi

Doctoral Program in Electrical and Computer Engineering

Approved in public examination by the Jury:

President: Professor Luís Miguel Pinho de Almeida
Referee: Professor Alexandre Bernardino
Referee: Professor Hugo Pedro Martins Carriço Proença
Referee: Professor Paula Maria Marques de Moura Gomes Viana
Referee: Professor Luís Filipe Pinto de Almeida Teixeira
Supervisor: Doctor Ana Maria Silva Rebelo

June 3, 2026

Resumo

A análise do comportamento e da identidade centrada no ser humano é um problema central na área da visão por computador. Apresenta aplicações fundamentais nos domínios da segurança e vigilância, da monitorização automática e dos sistemas de transporte inteligentes. Embora a área de deep learning tenha conduzido a avanços significativos no reconhecimento de ações, na re-identificação de pessoas e na estimação da pose humana, subsistem ainda vários desafios por resolver. Estes desafios resultam, entre outros fatores, da limitação e enviesamento dos conjuntos de dados, da sensibilidade às condições ambientais, da presença de oclusões e de preocupações relacionadas com a privacidade.

A primeira parte da tese centra-se no reconhecimento de ações humanas baseado em visão, com particular enfoque em cenários no interior de veículos. É estudada a influência de características dos conjuntos de dados, tais como a resolução de entrada, o rácio de aspeto, a obstrução do campo de visão e a taxa de imagens por segundo, fornecendo orientações para a utilização e implementação em contextos reais. De forma a mitigar o enviesamento associado ao ator, resultante da reduzida diversidade de sujeitos, são analisadas várias estratégias de treino, incluindo a utilização de características baseadas na pose, com o objetivo de promover representações invariantes ao ator durante o processo de aprendizagem.

A segunda parte da tese incide sobre o problema da reidentificação de pessoas. É proposto um enquadramento flexível que permite aprender máscaras dependentes da imagem, realçando as regiões mais discriminativas em termos de identidade. Para melhorar a robustez em cenários com oclusões, é apresentada uma arquitetura end-to-end de múltiplos ramos, juntamente com uma estratégia de aumento de dados baseada em oclusões artificiais. Avaliações realizadas em múltiplos conjuntos de dados demonstram um desempenho competitivo ou superior em cenários de re-identificação de pessoas holística, parcial e com oclusão.

A parte final da tese explora a estimação da pose humana e o reconhecimento de ações de forma preservadora da privacidade, recorrendo a sinais Wi-Fi. É realizada uma exploração sistemática de hiperparâmetros com o objetivo de identificar configurações arquiteturais adequadas, bem como um conjunto de experiências para analisar limitações relacionadas com os dados, nomeadamente a variação da taxa de amostragem, do tamanho da janela temporal e do passo da janela deslizante. Para melhorar a eficiência do modelo, são utilizadas duas estratégias distintas para selecionar os sub-portadores mais relevantes do sinal CSI, sendo o subconjunto selecionado utilizado no treino e avaliação do modelo. Os resultados indicam que os modelos treinados com um número reduzido de sub-portadores apresentam desempenho competitivo face ao modelo de referência, com ganhos significativos em eficiência computacional. Para melhorar a qualidade da estimação, é ainda proposta uma estratégia de aumento de dados baseada na adição de ruído, extraído de distribuições normal e uniforme, aos dados de treino, com o objetivo de reduzir o overfit. Embora quantidades excessivas de ruído conduzam à degradação da precisão devido à perda de informação relevante, a adição de pequenas quantidades de ruído resulta em melhorias no desempenho.

Em síntese, esta tese apresenta um estudo unificado da análise do comportamento e da identidade

centrada no ser humano nos domínios do vídeo e do Wi-Fi. Entre os principais desafios abordados encontram-se a robustez, o enviesamento dos dados, as oclusões, a interpretabilidade e a privacidade, com o objetivo de melhorar o desempenho destes sistemas em condições reais de utilização.

Palavras-chave: Reconhecimento de Atividades, Atenção, Veículos Autónomos, Visão por Computador, Aumento de Dados, Aprendizagem Profunda, Generalização de Domínio, Explicabilidade, Extração de Características, Reidentificação de Pessoas Holística, Estimação da Pose Humana, Deteção de Oclusões, Reidentificação de Pessoas com Oclusão, Reidentificação de Pessoas Parcial, Reidentificação de Pessoas (Re-ID), Robustez, Vision Transformer, Estimação da Pose Humana com Wi-Fi, Análise de Sinais Wi-Fi

Abstract

Human centered behaviour and identity analysis is a core problem in computer vision. It has key applications in the fields of security and survaillance, automatic monitoring and intelligent transportation systems. Although deep learning has led to significant advances in action recognition, person re-identification and human pose estimation, there are some challenges that still need to be addressed. The challenges stem from limited and biased datasets, sensitivity to environmental conditions, occlusions, and privacy concerns.

The first part of the thesis focuses on vision-based human action recognition with a particular focus on in-vehicle scenarios. The influence of dataset characteristics such as input resolution, aspect ratio, field of view obstruction, and input frame-rate are studied, providing guidance for real-world use and deployment. To address actor bias arising from limited subject diversity, several training strategies are studied, including pose-based features, with the goal of promoting actor-invariant representations during training.

The second part of the thesis focuses on person re-identification. A flexible framework is introduced to learn image-dependent masks that highlights the most identity-discriminative regions. To improve robusness under occlusion, an end-to-end multi-branch architecture, and an artificial occlusion-based data augmentation strategy is proposed. Comprehensive evaluations across multiple datasets demonstrate competitive or superior performance across holistic, partial and occluded person re-identification scenarios.

The final part of the thesis explores privacy-preserving human pose estimation and action recognition using Wi-Fi. We perform a hyperparameter sweep to find reasonable architecture hyperparameters, and perform a set of experiments to find limitations regarding the data, such as varying the sampling rate, the sliding window size, and the sliding window step size. To improve the efficiency of the model we use two different strategies to select the most relevant sub-carriers of the CSI signal, and use the selected subset to train and evaluate the model. We concluded that the models trained with fewer sub-carriers are competitive with the baseline but better in terms of computational efficiency. To improve the estimation of the model we also propose a noise-based data augmentation strategy, which consists of adding noise drawn from normal and uniform distributions to the training data to reduce overfitting. While excessive amounts of noise degrade the estimation accuracy due to the loss of information, small amounts of noise lead to better results.

Overall, this thesis presents a unified study of human-centered behavior and identity analysis across the video and Wi-Fi domains. Some of the challenges we address include robustness, data bias, occlusions, interpretability, and privacy, which aims to improve the performance of these systems in real-world conditions.

Keywords: Activity Recognition, Attention, Autonomous Vehicles, Computer Vision, Data Augmentation, Deep Learning, Domain Generalization, Explainability, Feature Extraction, Holistic Person Re-Identification, Human Pose Estimation, Occlusion Detection, Occluded Person Re-Identification, Partial Person Re-Identification, Person Re-Identification (Re-ID), Robustness,

Vision Transformer, Wi-Fi Human Pose Estimation, Wi-Fi Signal Analysis

Acknowledgements

I would like to start by thanking my supervisors, Dr. Ana Rebelo and Prof. Jaime Cardoso, for their guidance, knowledge, professionalism and constant availability throughout these years. Having good supervisors is one of the most important pillars of a PhD, and I could not have asked for a better supervising team to guide me through this journey. They were always available when I needed help, and for that I am extremely grateful. Thank you for the opportunity to work with you.

In addition to my supervisors, this work could not have been possible without the support of my two affiliated institutions, the Faculty of Engineering at the University of Porto and INESC TEC. I would also like to thank the Portuguese Foundation for Science and Technology for the financial support through the PhD Grant 2021.06945.BD, which made this PhD possible.

To all my colleagues and friends from the Visual Computing and Machine Intelligence (VCMI) research group. I learned so much from you, and I am deeply thankful for the support and the memories over the years.

To my friends and family, who gave me unconditional support, patience, and understanding throughout these years. Their encouragement, kindness, and belief in me were essential, especially during moments of doubt and exhaustion. I want to thank my father, Arlindo, my mother, Maria Angela, my sister, Leonela, my grandmother, Angela Maria, and my grandfather, Giuseppe, for their unconditional love and support. Words cannot describe how grateful I am. I am blessed to have a loving family that supports me in everything I do, and that I know will always have my back. To Inês Queiroz, for being my constant source of love, peace and comfort. Your presence makes the hardest days lighter and the joyful moments even brighter. I could not have asked for a better partner and I am forever grateful for being by your side.

Leonardo Capozzi

Contents

List of Figures

List of Tables

List of Acronyms

AVs	Autonomous Vehicles
BN	Batch Normalisation
BCE	Binary Cross Entropy
CSI	Channel State Information
CV	Computer Vision
CNN	Convolutional Neural Network
CE	Cross Entropy
CMC	Cumulative Match Characteristic
DL	Deep Learning
DNN	Deep Neural Network
ECG	Electrocardiogram
xAI	Explainable Artificial Intelligence
FOV	Field of View
FCT	Foundation for Science and Technology
FPS	Frames per Second
GELU	Gaussian Error Linear Unit
Grad-CAM	Gradient-Weighted Class Activation Mapping
GCN	Graph Convolution Network
GPU	Graphics Processing Unit
HAR	Human Action Recognition
HPE	Human Pose Estimation
INESC TEC	Institute for Systems and Computer Engineering, Technology and Science
ITS	Intelligent Transportation Systems
LN	Layer Normalisation
LiDAR	Light Detection and Ranging
LSTM	Long Short-Term Memory
mAP	Mean Average Precision
MPJPE	Mean Per Joint Position Error
MB	Megabyte
MFLOPS	Million Floating Point Operations

MMIT	Moments in Time
MSA	Multi-Head Self-Attention
MLP	Multi-Layer Perceptron
NLP	Natural Language Processing
PCK	Percentage of Correct Keypoints
Re-ID	Person Re-Identification
RECPAD	Portuguese Conference on Pattern Recognition
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
RGB	Red Green Blue
RCNN	Region-Based Convolutional Neural Network
SAM	Skeleton-Point Adjacency Matrix
STD	Standard Deviation
SDGs	Sustainable Development Goals
ViViT	Video Vision Transformer
ViT	Vision Transformer
VCMI	Visual Computing and Machine Intelligence
Wi-Fi	Wireless Fidelity

Part I

Prologue

Chapter 1

Introduction

1.1 Context and Motivation

Understanding human behaviour and identity from observational data is a central problem in Computer Vision (CV), with applications spanning security and surveillance, Intelligent Transportation Systems (ITS) and healthcare. Tasks such as Human Action Recognition (HAR), Person Re-Identification (Re-ID), and Human Pose Estimation (HPE) enable machines to reason about what people are doing and who they are, forming the basis for automated monitoring. In recent years, the rapid adoption of Deep Learning (DL) techniques has led to substantial performance gains across these tasks, often surpassing traditional handcrafted approaches. Despite this progress, significant challenges remain when deploying such systems in real-world scenarios [1].

A major limitation of deep learning-based human-centered analysis lies in its strong dependence on large, diverse, and well annotated datasets. In many practical settings, particularly constrained or safety-critical environments, the availability of such data is limited. Datasets may contain a small number of subjects, short durations, or insufficient variability in appearance, viewpoint, and environmental conditions[2]. As a result, models trained under these conditions often suffer from poor generalization, becoming sensitive to nuisance factors such as camera configuration, actor identity, lighting conditions, or occlusions. Addressing these limitations is essential for developing reliable systems that operate robustly outside controlled laboratory conditions.

These challenges are especially evident in in-vehicle environments, where monitoring passenger behaviour is becoming increasingly important with the emergence of autonomous and semi-autonomous vehicles. Applications range from ensuring passenger safety and comfort to detecting dangerous or violent behaviour. However, collecting large-scale, diverse in-vehicle datasets is difficult due to privacy concerns, safety constraints, and the cost of data acquisition. Consequently, models trained for in-vehicle action recognition are particularly susceptible to actor bias, where the learned representations inadvertently encode identity-specific characteristics rather than the underlying action. [3] This bias leads to poor performance when models are exposed to previously unseen individuals, motivating the need for training strategies that promote actor-invariant representations and improved generalization.

Beyond behaviour analysis, identity modeling through person re-identification plays a crucial role in security and surveillance systems. Person re-identification aims to match individuals across cameras or over time despite changes in viewpoint, illumination, pose, and background. Although deep learning has significantly advanced the state of the art, person re-identification remains challenging in realistic conditions, especially when individuals are partially visible or occluded by objects or other people. These scenarios are common in crowded environments and require models that are robust to missing or corrupted visual information. Attention mechanisms and data augmentation strategies have emerged as promising directions, but designing flexible and effective approaches that generalize across different levels of visibility remains an open problem [4].

Human pose estimation provides an intermediate and structured representation of human motion that is valuable for both action recognition and higher-level behavioural analysis. Traditional pose estimation approaches rely heavily on visual data acquired from Red Green Blue (RGB) cameras, Light Detection and Ranging (LiDAR), or radar sensors. While effective, these modalities present important limitations: vision-based methods are sensitive to lighting variations and occlusions, and LiDAR and radar require specialized and costly hardware. These limitations have motivated growing interest in alternative sensing modalities that can enable human understanding while mitigating some of these challenges[5], [6].

In this context, Wireless Fidelity (Wi-Fi) based sensing has emerged as a promising, and privacy-preserving alternative for human pose estimation and action recognition. By exploiting variations in Wi-Fi Channel State Information (CSI) caused by human motion, it is possible to infer pose and behaviour without capturing visual data. Recent advances in deep learning, particularly transformer-based architectures, have enabled more effective modeling of the complex spatio-temporal patterns present in Wi-Fi signals. However, this domain introduces its own challenges, including sensitivity to data configuration, limited interpretability of learned representations, and the need for computationally efficient models suitable for real-time deployment [7].

Another recurring challenge across both visual and Wi-Fi based domains is robustness under limited data. Overfitting remains a major concern when training deep models on small datasets. Data augmentation techniques are commonly employed to mitigate this issue, yet their effectiveness depends strongly on the type and magnitude of perturbations introduced, as well as on the underlying model architecture. Understanding how augmentation strategies, such as noise injection, interact with different learning paradigms is therefore essential for improving generalization without degrading performance [8], [9].

Motivated by these challenges, this thesis investigates robust human-centered behaviour and identity analysis across different sensing modalities. By studying the impact of data properties, addressing actor bias and occlusion, improving model interpretability and efficiency, and exploring privacy-preserving alternatives to visual sensing, this work aims to bridge the gap between high-performing research models and deployable real-world systems.

1.2 Research Aims

The aim of this thesis is to investigate robust deep learning methodologies for human-centric sensing tasks under challenging, data-constrained, and privacy-sensitive conditions. The work focuses on human action recognition, person re-identification, and human pose estimation across visual and wireless data. Particular emphasis is placed on understanding the impact of data properties, mitigating bias, the impact of occlusions, improving generalisation, and enhancing interpretability and efficiency of Deep Neural Network (DNN) in real-world scenarios such as in-vehicle monitoring and indoor environments.

The specific research aims of this thesis are as follows:

- To analyse the impact of data characteristics (e.g., resolution, aspect ratio, field of view, frame rate) on the performance of deep learning models for human action recognition in constrained environments.
- To address data scarcity and actor bias in in-vehicle human action recognition by proposing and evaluating methods that reduce dependence on actor-specific features and improve generalisation to unseen individuals.
- To improve person re-identification performance under partial and occluded conditions through novel data augmentation strategies and multi-branch end-to-end architectures.
- To investigate privacy-preserving alternatives to vision-based sensing by leveraging Wi-Fi channel state information for human pose estimation using transformer-based architectures.
- To study the interpretability and efficiency of deep learning models in the Wi-Fi domain using Explainable Artificial Intelligence (xAI) techniques, identifying the most important signal components for accurate pose estimation.
- To enhance model generalisation and robustness through data augmentation strategies, including artificial occlusions and noise injection, while analysing their effects on performance and overfitting.

1.3 United Nations Sustainable Development Goals

This thesis contributes to several United Nations Sustainable Development Goals (SDGs) by advancing intelligent, efficient, and privacy-aware technologies for human action recognition, person re-identification, and human pose estimation. The proposed methods have direct relevance to safety, well-being, sustainable urban environments, and technological innovation, particularly in real-world and resource-constrained scenarios such as vehicles and indoor spaces.

The research presented in this thesis aligns with the following SDGs:

- **Goal 3: "Good Health and Well-Being":** The development of reliable human action recognition and pose estimation systems supports applications in healthcare, assisted living,

and violence detection, contributing to improved safety, monitoring, and well-being in both private and public environments.

- **Goal 9: "Industry, Innovation and Infrastructure":** This work advances deep learning methodologies, including attention mechanisms, transformer-based architectures, and data-efficient training strategies, promoting innovation in intelligent sensing systems and supporting the development of robust AI-driven infrastructure.
- **Goal 11: "Sustainable Cities and Communities":** The proposed solutions for in-vehicle monitoring, indoor action recognition, and surveillance-related tasks contribute to safer urban environments, while exploring non-invasive sensing alternatives that reduce reliance on traditional camera-based systems.
- **Goal 16: "Peace, Justice and Strong Institutions":** Improvements in person re-identification, violence detection, and security-oriented systems support the development of technologies that enhance public safety and security, while the exploration of privacy-preserving sensing modalities helps promote responsible and ethical use of AI.

1.4 Publications

During his doctoral studies, the author collaborated and researched in deep learning applied to human action recognition, person re-identification and pose estimation. The publications that resulted (directly and indirectly) from the doctoral research are enumerated below, clustered by type and in reverse chronological order:

8. Capozzi, L., Sequeira, A.F., Teixeira, L., Oliveira, H.P., Rebelo, A. (2026). Video Vision Transformers for Privacy-Preserving Human Pose Estimation via Wi-Fi and Noise Augmentation. [submitted, waiting for decision]

7. Capozzi, L., Ferreira, L., Gonçalves, T., Rebelo, A., Cardoso, J.S., Sequeira, A.F. (2026). Deciphering the Silent Signals: Unveiling Frequency Importance for Wi-Fi-Based Human Pose Estimation with Explainability. In: Gonçalves, N., Oliveira, H.P., Sánchez, J.A. (eds) Pattern Recognition and Image Analysis. IbPRIA 2025. Lecture Notes in Computer Science, vol 15938. Springer, Cham. [10]

6. L. Capozzi, J. S. Cardoso and A. Rebelo, "End-to-End Occluded Person Re-Identification With Artificial Occlusion Generation," in IEEE Access, vol. 13, pp. 146020-146031, 2025, doi: 10.1109/ACCESS.2025.3600385 [11]

5. Ribeiro Pinto, João & Carvalho, Pedro & Pinto, Carolina & Sousa, Afonso & Capozzi, Leonardo & Cardoso, Jaime. (2022). Streamlining Action Recognition in Autonomous Shared Vehicles with an Audiovisual Cascade Strategy. 10.5220/0010838900003124. [12]

4. Capozzi, Leonardo and Barbosa, Vítor and Pinto, Carolina and Pinto, João Ribeiro and Pereira, Américo and Carvalho, Pedro M. and Cardoso, Jaime S., "Toward Vehicle Occupant-Invariant Models for Activity Characterization," in IEEE Access, vol. 10, pp. 104215-104225, 2022, doi: 10.1109/ACCESS.2022.3210973 [13]

3. Castro, Eduardo; Ferreira, Pedro M.; Rebelo, Ana; Rio-Torto, Isabel; Capozzi, Leonardo; Ferreira, Mafalda Falcão; Gonçalves, Tiago; et al. "Fill in the blank for fashion complementary outfit product Retrieval: VISUM summer school competition". Machine Vision and Applications 34 1 (2022): http://dx.doi.org/10.1007/s00138-022-01359-x. [14]

2. L. Capozzi, P. Carvalho, A. Sousa, C. Pinto, J. R. Pinto and J. S. Cardoso, "Impact of Visual Noise in Activity Recognition Using Deep Neural Networks - An Experimental Approach," 2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning (PRML), Chengdu, China, 2021, pp. 331-337, doi: 10.1109/PRML52754.2021.9520734. [15]

1. Capozzi, L., Pinto, J.R., Cardoso, J.S., Rebelo, A. (2021). Optimizing Person Re-Identification Using Generated Attention Masks. In: Tavares, J.M.R.S., Papa, J.P., González Hidalgo, M. (eds) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2021. Lecture Notes in Computer Science(), vol 12702. Springer, Cham. [16]

At a national level, the work developed during the doctoral studies resulted in four short papers presented at the Portuguese Conference on Pattern Recognition (RECPAD). The publications are enumerated below, clustered by type and in reverse chronological order:

4. T. Gonçalves, L. Capozzi, A. Rebelo, and J. S. Cardoso, “Deep Image Segmentation based on Mutual Information: A Study,” in 30th Portuguese Conference in Pattern Recognition (RECPAD), (Covilhã, Portugal), 2024 [17]

3. L. Capozzi, T. Gonçalves, J. S. Cardoso, and A. Rebelo, “Towards Fingerprint Presentation Attack Generation using Generative Adversarial Networks,” in 30th Portuguese Conference in Pattern Recognition (RECPAD), (Covilhã, Portugal), 2024 [18]

2. T. Gonçalves, L. Capozzi, A. Rebelo, and J. S. Cardoso, “Optimisation of Deep Neural Networks using a Genetic Algorithm: A Comparative Study,” in 29th Portuguese Conference in Pattern Recognition (RECPAD), (Coimbra, Portugal), 2023 [19]

1. L. Capozzi, T. Gonçalves, J. S. Cardoso, and A. Rebelo, “Analysis of Neurons’ Information in Deep Spiking Neural Networks using Information Theory,” in 29th Portuguese Conference in Pattern Recognition (RECPAD), (Coimbra, Portugal), 2023 [20]

1.5 Involvement in the Scientific Community

This section summarises the author’s participation in the Scientific Community.

1.5.1 Collaborations

The doctoral work presented in this thesis contains collaborations with members from the Visual Computing and Machine Intelligence (VCMI) Research Group of the Centre of Telecommunications and Multimedia at Institute for Systems and Computer Engineering, Technology and Science (INESC TEC). The author collaborated with researchers from the group in topics and projects related to the thesis.

1.5.2 Scientific Events

During his doctoral studies the author was involved in the organization of the following scientific events:

- INvicta school of VIsion, Computational intelligence, and patTern Analysis (INVICTA) Spring School - the author was a Member of the Organizing Team (2024);
- VISion Understanding and Machine intelligence (VISUM) Summer School - the author was a Member of the Project Committee (2021, 2022).

The author also participated in the following competitions and events:

- IEEE Video and Image Processing Cup (VIP Cup) on In-bed Pose Estimation
- IJCB2021 - Human Identification at a Distance 2021
- EUREKATHON 2020 - Challenging data for Zero Hunger

1.5.3 Scientific Projects

The author was a member of the following projects, where he developed research work related to his PhD:

- CONVERGE: Telecommunications and computer vision convergence tools for research infrastructures.
- AURORA: Characterise the activity and emotions of passengers inside vehicles.
- EasyRide: Emotional monitoring of vehicle occupants.

1.5.4 Student Supervision

The author proposed and supervised the following INESC TEC Summer Internships (in reverse chronological order):

3. Beatriz Arcipreste, Cláudia Ribeiro, José Macedo, "Classificação de atividade humana em vídeos", INESC TEC, 2022 - Leonardo Capozzi, Tiago Gonçalves.

2. Rodrigo Azevedo, Raul Viana, João Rôla, "Activity classification using noisy video streams for autonomous vehicles scenarios", INESC TEC, 2021 - Leonardo Capozzi, Afonso Sousa.

1. Rafael Cristino, Renata Lei, "Reconhecimento de hoteis para combater a exploração infantil e o tráfico de pessoas", INESC TEC, 2021 - Leonardo Capozzi, Ana Rebelo.

1.5.5 Teaching

The author worked as an Invited Teaching Assistant in the following courses (in reverse chronological order):

5. Invited Teaching Assistant during the 2nd Semester of 2025-2026 on the course Desenho de Algoritmos (Algorithm Design in C++).

4. Invited Teaching Assistant during the 1st Semester of 2025-2026 on the course Programação (Programming in C).

3. Invited Teaching Assistant during the 2nd Semester of 2024-2025 on the course Estruturas de Dados e Algoritmos (Data Structures and Algorithms in C++).

2. Invited Teaching Assistant during the 1st Semester of 2024-2025 on the course Programação (Programming in C).

1. Invited Teaching Assistant during the 1st Semester of 2023-2024 on the course Programação (Programming in C).

1.6 Document Structure

This thesis is organized into five main parts, progressing from foundational context and background to methodological contributions and concluding reflections.

Part I (Prologue) introduces the scope and objectives of the thesis. Chapter 1 presents the overall context and motivation, outlines the research aims, summarizes the scientific contributions and related activities, and describes the structure of the document. Chapter 2 provides the necessary background and conceptual foundations for the thesis. It reviews key topics including video-based action recognition, person re-identification and occluded person re-identification, human pose estimation, and WiFi-based human sensing.

Part II (Action Recognition using Video) focuses on action recognition in visual data, with an emphasis on robustness under degraded and constrained sensing conditions. Chapter 3 presents an experimental study on the impact of visual noise factors such as reduced resolution, field-of-view obstruction, and frame rate on deep neural networks for action recognition. This chapter provides empirical insights into the sensitivity of action recognition models to realistic perturbations commonly encountered in vehicle interiors. Chapter 4 builds upon these findings by proposing methods toward vehicle occupant-invariant activity characterization, exploring representation biases,

adversarial learning strategies, bilevel optimization, and the integration of pose information to improve generalization across individuals.

Part III (Person Re-Identification) addresses identity-aware perception under partial observability. Chapter 5 introduces a person re-identification framework based on generated attention masks, aiming to enhance discriminative feature learning by focusing on salient and visible regions. Chapter 6 extends this line of work to the more challenging setting of occluded person re-identification, presenting an end-to-end approach that incorporates artificial occlusion generation and specialized architectural components to improve robustness under severe occlusion.

Part IV (Pose Estimation and Action Recognition using Wi-Fi) explores privacy-preserving alternatives to vision-based sensing. Chapter 7 investigates explainability in Wi-Fi based pose estimation by analyzing the importance of different frequency components, providing insights into the relationship between wireless signals and human motion. Chapter 8 presents a Wi-Fi based human pose estimation framework that leverages video vision transformer architectures to map Wi-Fi CSI to pose representations. Chapter 9 examines data augmentation strategies for Wi-Fi signal data, addressing data generalization challenges in Wi-Fi based systems.

Finally, **Part V (Epilogue)** concludes the thesis. Chapter 10 summarizes the main contributions and reflects on the broader implications of the work.

1.7 Funding

This work has received funding from the Portuguese Foundation for Science and Technology (FCT) through the PhD Grant 2021.06945.BD

Chapter 2

Background

2.1 Human-Centered Behaviour and Identity Analysis: Overview

2.1.1 Definitions and Scope

The study of human-centered behavior and identity encompasses a range of tasks that vary semantically, temporally and have different modeling objectives. The concepts of action, activity, gesture and pose are central to this field, and each of them represent different levels of behaviour granularity.

An *action* is defined as a short duration motion event that can be recognized within a small time window. Examples include walking, sitting, standing up, waving, or jumping. They are charecterized by distinctive motion patterns, and are commonly used as the basic units in action recognition works. An *activity* is composed of higher-level behaviours, consisting of multiple actions. They tipically occur over longer time durations, and have temporal and contextual dependencies between the different actions that are part of that activity. Unlike action recognition systems, activity recognition systems require modeling long temporal dependencies and relationships, and the action composition of the activity being performed.

A *gesture* is composed of fine-grained and intentional movements of a certain body part (e.g., hands, arms, upper body). They are commonly used in communication and interaction contexts, typically have shorter time durations, and are much more subtle than full-body actions. Gesture recognition requires precise motion modeling and sensitivity to small temporal changes. A *pose* describes the position of the human body at any given moment. It is commonly represented using the joint positions or orientations, and it captures human kinematics in a structured form. Pose information serves as a low-level representation of human kinematics and is often used as an intermediate abstraction for higher-level tasks such as action recognition.

Another important distiction in human-centered analysis is between two complementary modeling objectives: *behaviour modeling* and *identity modeling*. Behaviour modeling focuses on motion patters and temporal dynamics, in tasks such as action recognition and activity recognition, where the goal is to detect what the individual is doing, regardless of the identity of the person. In other words, the objective is to be invariant to the individual performing them. Identity modeling, such as

person re-identification, focuses on recognizing or matching individuals across time, viewpoints, or sensing devices. The goal of identity modeling is to be invariant to changes in time, location, pose, motion, behaviour or anything else that is not related to the identity of the individual. These two objectives have different (and opposing) invariance requirements, where behaviour modeling tries to be invariant to the individual, while identity modeling tries to be invariant to everything not related to the identity of the person, making the disentanglement and separation of behaviour-related and identity-related features a key challenge.

Human centered analysis can be categorized by temporal scale. *Frame-level* analysis operates on individual time steps, i.e. a single frame, and it is commonly used for pose estimation as it emphasizes instantaneous spatial information, and doesn't rely on past events. *Clip-level* analysis consists of using short temporal windows typically containing a small number of frames, which capture motion over a short period of time. They are typically used for action recognition. *Sequence-level* analysis operates on long temporal sequences, which are required for more complex behaviours such as activities, or continuous monitoring.

Another important consideration is the number of subjects that are being analysed. *Single-person* scenarios are more common in controlled environments, such as laboratory experiments, while *Multi-person* scenarios involve multiple subjects in the same scene, sometimes interacting with each other, which pose different challenges such as occlusion between subjects, interactions between different people, identity ambiguity, and signal interference, depending on the type of data that is used.

2.1.2 Ethical and Practical Considerations

Human-centered behaviour and identity analysis raises many ethical and practical considerations that must be taken into account when acquiring and using data, and when deploying these systems into the real world. Such systems may capture sensitive personal information, as they involve observing or inferring human behaviour and identity. Vision-based sensing can reveal identifiable details, such as faces, appearence or surroundings. Behaviour detection can expose private habits or routines without explicitly trying to recognize the identity. Privacy preservation is one of the key motivators for alternative sensing modalities that do not include vision, as there are a number of concerns regarding the data acquired, such as potential misuse of the collected data or lack of consent.

There are some practical considerations that must be taken into account regarding data collection. Learning-based methods typically require large amounts of data, and its acquisition is time-consuming, labor-intensive and sometimes expensive. Data is often collected in controlled environments, which leads to a limited diversity in the number of subjects, environments or behaviours. These things make models biased and reduce their generalization capabilities in real-world conditions. Manual data annotation also poses a significant challenge as ground truth labels are sometimes noisy or inconsistent.

Another thing to consider is that performance in controlled settings may not transfer to real environments, as systems must handle environmental variation, such as lighting, layout, background activity, sensor noise, occlusions and interference. Computational cost, latency, and power consumption are also critical for real-world deployment, especially for real-time systems or edge deployment. Aditionally, these systems must comply with legal and regulatory privacy requirements. Together, these ethical concerns play a central role in the development, design, evaluation and deployment of human-centered behaviour and identity analysis systems.

2.2 Action Recognition Using Video

2.2.1 Problem Definition

Action recognition using video aims to automatically identify and classify human actions or activities from sequences of visual data. Given a video clip composed of a temporally ordered set of frames, the objective is to assign one or more action labels that best describe the action being performed. Formally, the task consists of learning a mapping from a video sequence, represented as a spatio-temporal signal, to a discrete action category drawn from a set of classes.

This problem is challenging due to the inherent complexity of human motion and the variability present in real-world scenarios. Actions can differ significantly in duration, speed, appearance, and execution style across individuals. In addition, factors such as camera viewpoint, lighting conditions, occlusions, and motion blur introduce noise into the visual data. These challenges become more difficult in unconstrained environments, where the conditions under which the video data is captured is not controlled.

From a modeling perspective, effective action recognition requires capturing both spatial information, which encodes the appearance of the scene and the actors, and temporal information, which describes how motion and interactions evolve over time. As a result, modern approaches typically rely on spatio-temporal representations that combine visual features across multiple frames. The task is commonly framed as a supervised learning problem, where models are trained on labeled video datasets and evaluated on their ability to generalise to unseen videos and actors.

In application domains such as surveillance, human-computer interaction, and in-vehicle monitoring, additional constraints arise, including limited computational resources, data scarcity, and the need for real-time processing. These constraints motivate the development of efficient and robust action recognition methods that can operate well under varying data quality and environmental conditions.

2.2.2 Challenges

Despite significant progress in recent years, action recognition using video remains a challenging problem due to a combination of data, modeling, and deployment-related factors. One of the primary challenges lies in the high variability of human actions. The same action can be performed with different speeds, styles, and levels of intensity by different individuals, leading to large

intra-class variation. Conversely, different actions may exhibit similar visual patterns, resulting in inter-class similarity that complicates discrimination.

Temporal modelling constitutes another major difficulty. Actions unfold over time and may vary significantly in duration, making it non-trivial to determine the temporal extent of an action and to capture the most informative motion cues. Fixed-length video clips or sampling strategies may omit relevant temporal information or include redundant frames, adversely affecting recognition performance.

Visual variations further complicate the problem. Changes in viewpoint, scale, illumination, background clutter, and camera motion can significantly alter the visual representation of an action. Occlusions, either caused by objects in the scene or by self-occlusion of the human body, may partially hide critical motion cues, leading to incomplete or ambiguous observations.

Data-related challenges also play a crucial role. High-performing deep learning models typically require large-scale, diverse, and well-annotated datasets, however, such datasets are often expensive and time-consuming to collect. In many application-specific scenarios, including in-vehicle environments, available datasets are small, imbalanced, or captured under constrained conditions, increasing the risk of overfitting and poor generalisation to unseen data or actors.

Finally, practical deployment constraints must be considered. Real-world systems often operate under limited computational budgets and strict latency requirements, particularly in embedded or real-time settings. Achieving a balance between recognition accuracy, robustness, and computational efficiency remains an open challenge, motivating research into data-efficient strategies, lightweight architectures, and adaptive inference mechanisms.

2.3 Person Re-Identification

2.3.1 Problem Definition

Person re-identification is the task of recognising and matching individuals across non-overlapping camera views or different time instances. Given an image or a sequence of images containing a person, the objective is to correctly associate that person with previously observed instances, despite variations in appearance, pose, and environmental conditions. Formally, the goal is to learn a mapping from the visual representation of a person to a feature space where images of the same individual are close together while images of different individuals are well separated.

This problem is particularly challenging due to several sources of visual variability. Differences in camera viewpoints, lighting conditions, and occlusions can significantly alter the appearance of the same person. Moreover, individuals may change clothing, carry objects, or adopt different poses over time, increasing intra-class variability. At the same time, different people may share similar appearances, such as clothing color or body shape, leading to high inter-class similarity and potential misidentifications.

Person re-identification is commonly framed as a supervised learning problem, where models are trained on labeled datasets consisting of multiple images per individual captured under different

conditions. The learned feature representations are then used to measure similarity between query and gallery images, often through distance metrics or ranking procedures. Recent approaches leverage deep learning to automatically extract discriminative features, incorporating strategies such as attention mechanisms, part-based modelling, and occlusion-aware representations to improve robustness.

The problem has broad applications in security, surveillance, and smart environments, where tracking and recognising individuals across camera networks is critical. Success in person re-identification requires models that are not only accurate but also robust to variations in appearance and efficient enough for large-scale deployment across multiple cameras and real-time settings.

2.3.2 Challenges

Person re-identification is a complex problem due to several inherent challenges related to visual variability, occlusion, and dataset limitations. One of the main difficulties is the large intra-class variation: the same individual can appear drastically different across camera views due to changes in pose, viewpoint, illumination, and background context. Additionally, people may change clothing, carry objects, or adopt different walking styles, further increasing the variability within the same identity.

At the same time, high inter-class similarity poses another challenge. Different individuals may share similar physical attributes, clothing colors, or accessories, making it difficult for models to distinguish between them solely based on appearance. This is especially problematic in crowded or large-scale camera networks where many identities may appear similar.

Occlusion is a frequent issue in real-world scenarios. People can be partially hidden by objects, other individuals, or the scene itself, which can obscure discriminative features necessary for correct identification. Traditional holistic models often struggle under these conditions, motivating the development of occlusion-aware and part-based approaches.

Data scarcity and imbalance add further complexity. While deep learning models require large amounts of labeled data, many Re-ID datasets are limited in size, contain few images per identity, or exhibit class imbalance. This can lead to overfitting and poor generalisation to unseen individuals or camera setups.

Finally, practical deployment imposes additional constraints. Models need to be computationally efficient to handle real-time or large-scale surveillance systems, and they must maintain robustness to domain shifts, such as new camera environments or varying acquisition conditions. These challenges collectively make person re-identification a demanding task that requires both robust feature representation and carefully designed learning strategies.

2.4 Pose Estimation and Action Recognition Using Wi-Fi Signals

2.4.1 Problem Definition

Pose estimation and action recognition using Wi-Fi signals are two related but distinct tasks that take advantage of wireless signal variations caused by human motion. Both tasks rely on indirect measurements such as CSI, which capture the impact of human movement on the surrounding wireless environment.

In Wi-Fi based pose estimation, the objective is to infer the spatial position of the human skeleton by predicting the positions of body joints. This involves mapping the temporal patterns in the Wi-Fi signals to a set of joint coordinates that represent the human posture, enabling applications such as activity monitoring, rehabilitation, or fall detection.

In Wi-Fi based action recognition, the goal is to classify human activities from sequences of Wi-Fi measurements. Each action, such as walking, sitting, or waving, produces characteristic patterns in the signal over time. The task is to learn a mapping from these spatio-temporal signal variations to a discrete set of action labels.

Although both tasks exploit the same type of input data, they address different objectives: pose estimation focuses on reconstructing detailed body geometry, while action recognition emphasizes understanding high-level human behaviors.

2.4.2 Challenges

Wi-Fi based pose estimation and action recognition each face unique challenges, which come from the indirect and noisy nature of wireless signals. For pose estimation, the main difficulties arise from limited spatial resolution, as Wi-Fi signals have wavelengths on the order of centimeters, which restricts the precision with which small or closely spaced joints can be localized. Environmental factors such as reflections, furniture, walls, and other obstacles further distort the signals, making it harder to reconstruct the position of the human body. Non-line-of-sight conditions add another layer of complexity, requiring models to infer joint positions from indirect signal paths. Temporal dynamics also pose challenges, as fast or subtle movements may be missed without sufficiently high sampling rates, while excessively high rates increase computational complexity.

For action recognition, the challenges include the variability of human motion across individuals, as people perform the same action in different ways, generating diverse signal patterns. Similar actions may produce overlapping signal signatures, increasing the risk of misclassification. Environmental factors, such as layout changes or additional moving objects, introduce additional noise and ambiguity. Furthermore, Wi-Fi measurements are inherently indirect, requiring models to extract discriminative spatio-temporal features from subtle variations in amplitude, or both amplitude and phase.

Across both tasks, practical deployment considerations impose additional constraints. Systems must be robust to environmental changes, generalize across different individuals, and operate efficiently in real time, often with limited computational resources. Balancing accuracy, robustness,

and efficiency remains a central challenge in developing reliable Wi-Fi based human sensing systems.

2.5 Relationship Between Vision-Based and Wi-Fi-Based Human Sensing

2.5.1 Common Problem Space

Vision-based and Wi-Fi based human sensing share a common problem space in that both aim to understand human behavior, motion, and pose from observable signals. In both cases, the objective is to extract meaningful information about human behaviour, body configurations, or identity from input data that encodes motion and/or identity over space and time.

Both modalities must handle similar sources of variability, including differences in individual motion styles, speed of movements, and environmental conditions. Occlusions, whether caused by other objects or body parts, are a challenge in vision-based systems, while in Wi-Fi based systems, non-line-of-sight conditions create other difficulties, although the person can still often be detected. Temporal modeling is critical in both cases, as human actions unfold over time and require capturing sequential dependencies to achieve accurate recognition.

Both approaches rely on feature extraction to translate pixel measurements in images or signal amplitudes and/or phases in Wi-Fi into embeddings suitable for estimation. This includes estimating poses, recognizing actions, or uniquely identifying individuals. In this sense, vision and Wi-Fi human sensing share common algorithmic frameworks, such as convolutional or transformer-based models for spatio-temporal pattern recognition, despite the differences in the nature of the input data.

2.5.2 Vision Versus Wi-Fi Sensing Modalities

Vision-based and Wi-Fi based human sensing differ fundamentally in the type of data they use and the information they can capture. Vision-based systems rely on cameras to acquire images or video sequences, providing high-resolution spatial and temporal information about human identity, pose, and movement. These systems are great at capturing fine-grained visual details, including body shape, clothing, and facial expressions, which can be directly interpreted by humans or processed by computer vision algorithms. Vision-based sensing is also widely used for person re-identification, where the goal is to recognize and match individuals across different camera views. However, vision-based systems are sensitive to lighting conditions, occlusions, and privacy concerns, which can limit their applicability in certain environments.

In contrast, Wi-Fi based sensing detects the impact of human motion on wireless signals. By analyzing variations in signal amplitude and/or phase, Wi-Fi sensing can infer human activity or estimate pose, without requiring direct visual contact. This modality is non-intrusive, works in low-light or obscured conditions, and preserves privacy since it does not capture identifiable visual features. However, Wi-Fi based systems face challenges due to indirect and noisy measurements,

lower spatial resolution, and environmental interference, which can complicate both pose estimation and action recognition.

Overall, while vision provides detailed and direct information, Wi-Fi offers privacy-preserving sensing that can operate in challenging conditions, highlighting the complementary nature of the two types of data for human sensing tasks including pose estimation, action recognition, and person re-identification.

Part II

Action Recognition using Video

Chapter 3

Impact of visual noise in activity recognition using deep neural networks - an experimental approach

Foreword on Author Contributions

The results of this work has been disseminated in an article in international conference:

[15] L. Capozzi, P. Carvalho, A. Sousa, C. Pinto, J. R. Pinto and J. S. Cardoso, "Impact of Visual Noise in Activity Recognition Using Deep Neural Networks - An Experimental Approach," 2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning (PRML), Chengdu, China, 2021, pp. 331-337, doi: 10.1109/PRML52754.2021.9520734.

3.1 Introduction

With the increased popularity of deep machine learning, data has gained augmented importance with many initiatives targeting the collection and preparation of datasets. However, in areas such as autonomous (shared) vehicles, these datasets are scarce, may not have the necessary characteristics, and are essentially private. This poses a problem for the research and development of models in topics for these scenarios since large amounts of data are needed to train robust models.

The recognition of emotions, activities, and unwanted behaviours plays a key role in autonomous shared vehicles [21]. Unwanted behaviour is an action or activity that can be described as undesirable in a given context due to its potentially harmful consequences. To this end, the set of human activities that can be categorised as unwanted differs according to the considered scenario. For example, eating or drinking in a shared car may be regarded as unwanted since it may cause discomfort to other passengers.

This paper analyses the impact of visual noise on deep neural network-based models for the recognition of a set of human activities, with three main contributions. First, given the general

unavailability of datasets targeting the proposed specific scenario, several public datasets were analysed, collected, and filtered to identify segments encompassing a subset that more closely resembles the target scenario. We studied the impact of different variations of input information (frame rate, resolutions, aspect ratio), both from a computational performance and model accuracy perspective. Also, the performance of the model under different types of visual noise, such as cropping and occlusions, was analysed since these are quite common in footage obtained from the inside of the vehicle where parts of the body might not be completely visible.

The remainder of this paper is organized as follows: the proposed methodology is presented in section 3.2; the data and training process are detailed in section 3.3; the conducted experiments are explained in section 3.4; section 3.5 presents and discusses the results; and conclusions are drawn in section 3.6.

3.2 Methodology

This paper presents the results of a study on the robustness of models for classifying human activity in the presence of different types of noise. In the last years, advances in deep learning methodologies have surpassed previous approaches to the problem, with 3D Convolutional Neural Network (CNN) showing enormous potential.

3D CNN are formed of 3D convolutions, which allow the network to process the temporal information of the input throughout the network [22], [23], [24], [25], [26], [27], [28], [29], [30], [31]. Before 3D networks, models consisted of multiple streams that processed the temporal component; however, these methods used 2D convolutions, which meant that the temporal information was added to the channels, sometimes limiting their performance. The output of a 2D convolution is always an image, which means that if the temporal component of a video is added to the channels, then the information is more easily lost. The output of a 3D convolution, on the other hand, is a video volume that preserves the temporal information of the input, making it better for video classification.

The model used in this study is based on the model proposed in [32], which follows the architecture presented in Figure 3.1. It is composed of the 3D ResNet50's convolutional encoder blocks, followed by an average pooling layer, dropout layer, and fully-connected layer with an input size of 2048 and an output size equal to the number of classes. The convolutional encoder is composed of one 3D convolutional layer containing 64 filters with a size of $7 \times 7 \times 7$, followed by a batch normalisation layer and Rectified Linear Unit (ReLU) activation function. The subsequent layers are three residual blocks of type A, four blocks of type B, six blocks of type C, and four blocks of type D. Each block contains three convolutional layers, with a filter size of $1 \times 1 \times 1$, $3 \times 3 \times 3$, and $1 \times 1 \times 1$, respectively. The first two convolutional layers of each block have 64 (type A), 128 (type B), 256 (type C), or 512 filters (type D), and are followed by a batch normalisation layer. The last convolutional layer of each block has four times the number of filters as the first two layers and is followed by a batch normalisation layer and ReLU activation function. More details on the network architecture are available on the original ResNet paper [33].

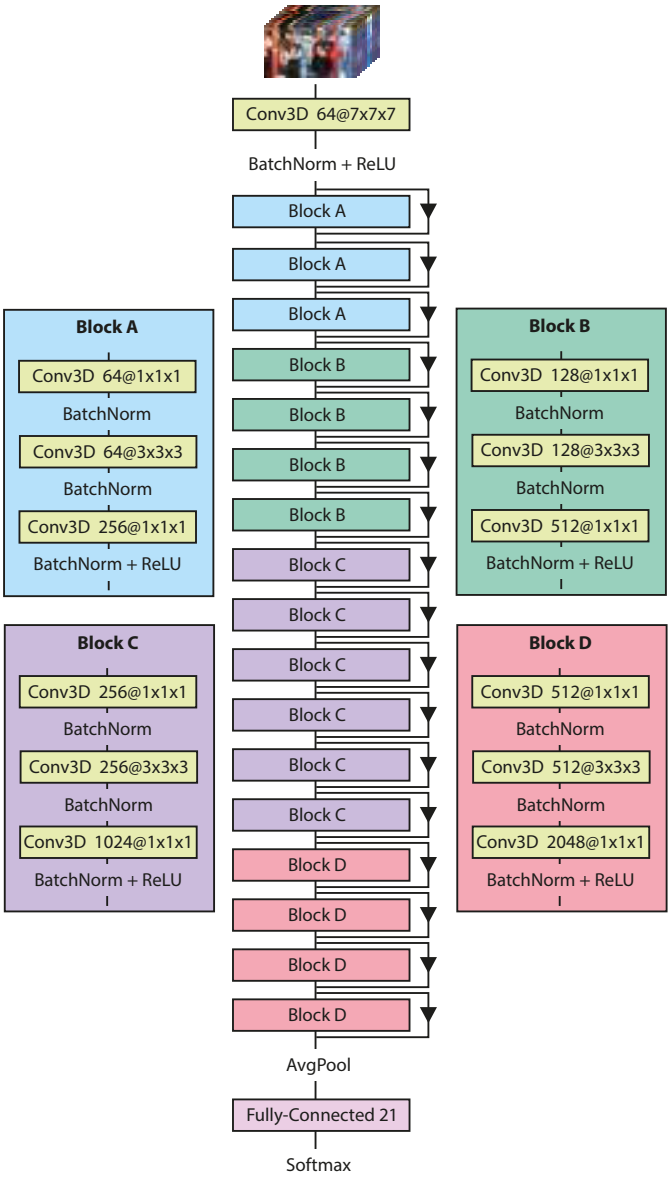

Figure 3.1: Network architecture (adapted from [32]).

Table 3.1: Summary of the datasets and corresponding classes used in the experiments.

Dataset	Number of videos used	Number of classes used	Resolution
Moments in Time [34]	10771	21	varied
HMDB51 [35]	814	7	varied
Hollywood [36], [37]	243	3	varied
MSR DailyActivity3D [38]	60	3	640x480
UWA3D Multiview [39]	107	2	320x240
SBU [40]	100	3	640x480

3.3 Experimental Setup

3.3.1 Data

Given the absence of public datasets specific to the target scenario, several datasets were analysed and filtered to identify an adequate subset.

The following were selected for the subsequent tests: Moments in Time (MMIT) [34]; HMDB51 [35]; Hollywood [36], [37]; MSR DailyActivity3D [38]; UWA3D Multiview [39]; SBU [40].

For training the model, 21 classes from the Moments in time were selected, considering those that are more closely related to the scenario of autonomous shared vehicles. For each dataset, we selected the classes that were in the set of 21 classes used to train the model, which allowed us to compute cross-dataset accuracies. Table 3.1 provides more detailed information about the datasets and the corresponding number of classes.

3.3.2 Training

The network used in this work (described in Section 3.2) was pre-trained on the Moments in Time dataset, and later fine-tuned with a subset of 21 classes from the Moments in Time dataset. These classes were chosen because they were the most relevant for the in-vehicle scenario (see Table 3.11 for the list of classes). All videos were resized to 224×224 resolution and then normalised using the mean and standard deviation from ImageNet, as these were the settings used on the pre-trained network. This pre-processing step was applied to all videos when training and testing the model. Since the base network was trained with 21 classes, we chose datasets that contained at least 2 of these 21 classes. This allowed us to test whether there was a drop in performance when videos from datasets other than the one it was trained on were presented to the model.

During training, the parameters of the 3D ResNet-50 layers were frozen, except for the final fully-connected layer. This was done to take advantage of the pre-training on the Moments in Time dataset [34].

The loss function used to train the model was cross-entropy, with a batch size of 32 videos, and the Adam optimiser with a learning rate of 1×10^{-4}.

3.4 Experiments

The following sub-sections describe a set of experiments intended to assess the impact of a given type of visual noise or input characteristics, namely: input resolution; frame rate; obstruction of the Field of View (FOV). The first two types are important from two different axes: noise and efficiency. Reducing the resolution or the frame rate means less information to be processed which can translate to less memory required and fewer operations (less computational complexity), but in turn, it may imply that relevant information is absent. This is an important trade-off, especially in autonomous (electric) vehicles, as computational resources are limited, and power consumption is a major concern. FOV obstructions, or occlusions, are important factors, favoured by the camera positioning in the vehicles and reduced space.

3.4.1 Input resolution

The base model was trained on videos downsized to 224×224 resolution; therefore, videos with a different aspect ratio would be stretched. It is relevant to assess if the videos' original aspect ratio would affect the performance of the model. Note that the network accepts different resolutions as it contains an adaptive average pooling layer. For the first experiment, the original aspect ratio was preserved by resizing the smaller dimension of the image to 224px and adjusting the other dimension accordingly. In a second experiment, the videos were inputted to the model without any resizing, i.e., keeping the original resolution.

The next experiment consisted of slightly adjusting the weights of the model trained on the MMIT dataset (finetuning). To accomplish this, we performed a small training session using the HMDB51 dataset and the Hollywood dataset. The idea was to allow the already trained network to adapt to the new datasets in order to get a performance gain.

3.4.2 FOV obstruction

Several tests were performed to verify the performance of the model in scenarios where some visual information is missing, since there are situations inside the vehicle where the field of view is obstructed, and the sensor can only see part of the person. To simulate these scenarios, parts of the video that may contain relevant information were cropped. More specifically, the following cases were analysed: cropping at the waist; cropping the right side of the body; cropping the left side of the body.

In order to automatically calculate where to crop the frames, the pose of each person in the frame was first calculated, extracting the corresponding bounding box and the coordinates of the main body parts. The pose extraction model [41] has the advantage of calculating 3D coordinates (it also estimates the distance from the camera). In situations where the individuals were very close

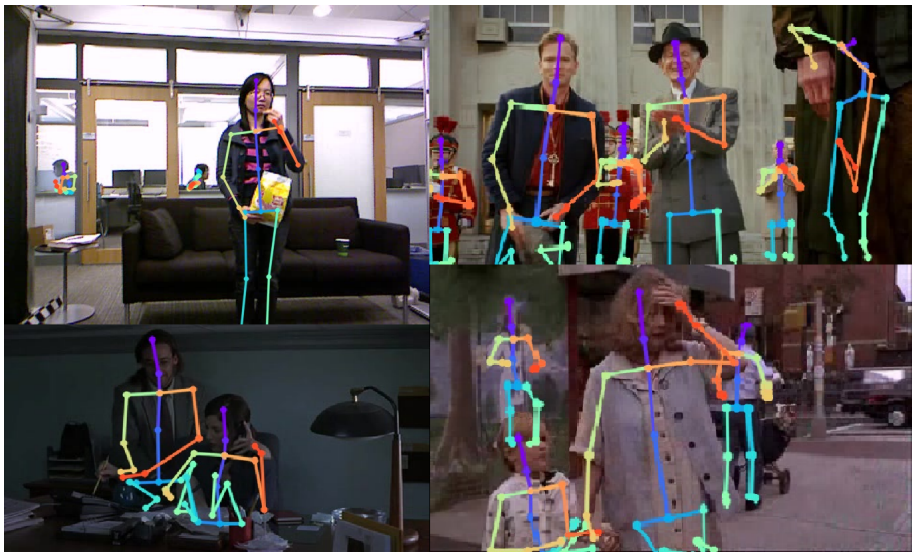

Figure 3.2: Key points calculated by the pose estimation model.

to the camera and parts of the body were not visible, e.g. the legs, the method randomly placed the key points corresponding to these parts on the screen. This did not have a negative effect as the key points used were visible on the screen in most of the videos. Figure 3.2 shows the key points predicted by this model.

Once the pose was determined, the crop of the image was calculated. In situations where there was only one person in the frame, calculating the crop was fairly straightforward as only that person needed to be considered. In situations where multiple people were visible, it was necessary to determine which of them were in the foreground (performing the actions) and which were in the background (irrelevant to the action recognition task). This process took into account the size of the largest person in the frame and assumed that all people above a certain threshold were in the foreground. When performing a right/left side crop, we cropped 30% of the rightmost/leftmost person in the frame. When performing a waist crop, we cropped at the waist of the person nearest to the bottom of the frame. Figure 3.3 shows examples of different crops.

In a subsequent test, the model was fine-tuned with data augmentation (randomly selecting right crop, left crop, waist crop, or keeping the original for each video) to check if there would be any performance gain. The model was fine-tuned using the MMIT, HMDB51, and Hollywood datasets. Data augmentation was used to avoid overfitting and improve the performance of models by increasing the amount of data, through the application of transformations to the original data. In this case, the transformation applied could be too drastic to improve accuracy, possibly removing information that is critical to make a prediction. This could have the opposite effect and hinder the performance of the model. To test whether this was the case, we checked the accuracy of the model that was fine-tuned on the Moments in Time dataset with data augmentation, on the other datasets.

Figure 3.3: Frames before and after cropping.

3.4.3 Input frame rate

We performed a series of tests to verify the impact that the input frame rate had on the model's accuracy. It is important to know if different frame rates have or not a large impact on the accuracy of the model, since using higher frame rates comes with associated computational costs, which is not ideal in situations where the computational power available is limited. In these experiments, we trained the model on the MMIT, HMDB51, and Hollywood datasets, with a training frame rate of 1, 2, 4, and 8 Frames per Second (FPS). This resulted in 12 different models, each trained with a different combination of dataset and frame rate. We tested the accuracy of each of the trained models using frame rates of 1, 2, 4, and 8 FPS.

3.5 Results

3.5.1 Input resolution

Table 3.2 shows the performance of the model under changes in resolution and aspect ratio. As expected, the performance of the model decreased slightly as it was presented with data that did not match the resolution of the data it had been trained with. Preserving the aspect ratio of the original input caused a slight decrease in performance and using the original higher resolution also did not add any benefit. The model also showed good accuracy on the datasets it was not trained on, demonstrating that it was able to apply what it learned from the Moments in Time dataset to other datasets.

Table 3.3 shows the performance of the model after finetuning the parameters. Performance increased considerably, since it was better adapted to the new data, rather than relying exceptionally on what it had previously learned from the Moments in Time dataset. In terms of resolution and aspect ratio changes, the results were consistent with those in Table 3.2 and showed a slight decrease in accuracy when using the original aspect ratio.

The use of lower resolutions can be an important advantage in autonomous vehicle scenarios as they reduce the computational complexity or, in other words, increase the amount of information that the model can process in the same period of time. Table 3.4 shows the number of frames processed by the model per second, using an NVIDIA GeForce GTX 1080 Graphics Processing Unit (GPU), depicting a significant gain. The advantage is clear, especially considering that, as previously shown, these resolution differences have a very small impact on model accuracy.

3.5.2 FOV obstruction

Table 3.5 summarises the accuracy of the model after performing the different crops to simulate obstructions in the field of view. For the MMIT, HMDB51, Hollywood, and SBU datasets, a slight drop in performance is observed due to cropping. However, the MSR DailyActivity3D and the UWA3D Multiview datasets did not suffer from such a drop. This is likely because they contain simpler studio-like scenarios; when we apply a crop, the action information is not completely lost, but in many cases, the background noise is removed, and therefore the accuracy does not decrease.

Table 3.2: Model accuracy with and without rescaling after training on MMIT

Dataset	Rescaling to 224x224px (%)	Rescaling to 224px preserving aspect ratio (%)	Original resolution (%)
MMIT	**51.49**	51.23	44.62
HMDB51	**69.53**	67.44	66.46
Hollywood	**52.67**	49.79	48.56
MSRDailyAct3D	**45.00**	**45.00**	41.67
UWA3D Mult	**70.09**	71.03	**70.09**
SBU	**50.00**	48.00	35.00

Table 3.3: Model accuracy with and without rescaling after fine-tuning on each dataset.

Dataset	Rescaling to 224x224px (%)	Rescaling to 224px preserving aspect ratio (%)	Original resolution (%)
HMDB51	**78.53**	77.91	77.91
Hollywood	**67.35**	65.31	65.31

Table 3.4: Processing capabilities of the model using different resolutions.

Resolution	Frames Per Second
224x224px	1043
300x224px	757
640x480px	134

Table 3.5: Model accuracy after training on MMIT.

Dataset	No crop (%)	Waist crop (%)	Right crop (%)	Left crop (%)
MMIT	**51.49**	44.88	45.40	45.65
HMDB51	**69.53**	54.05	57.25	55.04
Hollywood	**52.67**	51.44	46.50	50.21
MSRDailyAct3D	45.00	**71.67**	56.67	63.33
UWA3D Mult	70.09	**84.11**	72.90	78.50
SBU	50.00	**51.00**	43.00	38.00

Table 3.6 shows the performance of the model after finetuning. The performance increased considerably for the HMDB51 and Hollywood datasets, but there was not much gain for the MMIT dataset, as this was the dataset that the model was originally trained on.

Comparing Tables 3.5 and 3.7, we notice that the fine-tuned model performs worse on most of the datasets, which could mean that training with the full frames yields better performance than training with frames missing crucial information.

3.5.3 Input frame rate

Tables 3.8, 3.9 and 3.10 show the accuracy of the model using the MMIT, HMDB51 and Hollywood dataset, respectively in the presence of different input frame rates. We can observe that the model appears to have the best performance when it is trained and tested with the same frame rate. When the training frame rate is different from the testing frame rate, there is a slight performance decrease, which means that the model should be trained with a frame rate similar to the frame rate provided by the final system. We can also note that higher frame rates generally lead to higher accuracies, as the model is able to extract more precise information about the movement, particularly for faster actions. The performance increase from using a higher frame rate is not very large, which means that, in a system with limited computational resources, it might be advantageous to have a 2% to 5% decrease in accuracy for a boost in computational performance.

Tables 3.11, 3.12 and 3.13 show the best accuracies achieved in every class with different frame rates when testing the model with the MMIT, HMDB51 and Hollywood dataset, respectively. The

Table 3.6: Model accuracy after fine-tuning on each dataset with data augmentation.

Dataset	No crop (%)	Waist crop (%)	Right crop (%)	Left crop (%)
MMIT	**49.94**	44.10	45.78	45.40
HMDB51	**79.75**	65.03	71.78	62.58
Hollywood	**69.39**	61.22	55.10	55.10

Table 3.7: Model accuracy after fine-tuning on MMIT with data augmentation.

Dataset	No crop (%)	Waist crop (%)	Right crop (%)	Left crop (%)
MMIT	**49.94**	44.10	45.78	45.40
HMDB51	**69.16**	57.25	59.46	59.58
Hollywood	46.91	**51.03**	42.39	44.86
MSRDailyAct3D	51.67	**71.67**	53.33	60.00
UWA3D Mult	65.42	**81.31**	71.96	74.77
SBU	**49.00**	41.00	43.00	42.00

results also suggest that in most cases, higher frame rates lead to higher accuracies. However, there are a few classes that are outliers, such as singing, kissing, celebrating, laughing, and eating; these classes generally contain slow movements, which means that using a high frame rate does not give the model any additional information, making them more invariant to the used frame rate than classes with fast movements.

3.6 Conclusion

This chapter focused on the recognition of activities in an in-vehicle setting, measuring the performance of the model under different scenarios. Given the unavailability of data for the specific in-vehicle environment, we collected a set of publicly available action recognition datasets, and chose 21 classes that we found relevant for our scenario.

Several experiments were conducted, considering different input frame rates, resolutions, aspect ratios, as well as obstructions of the field of view.

Results showed that the accuracy of the model decreased slightly when training and testing on different resolutions. Moreover, it depicted good accuracy on the datasets it was not trained on. After fine-tuning the model with other datasets, a considerable increase in accuracy was observed, since the model adapted to the new data. Measuring the computation time it was possible to conclude that using lower resolutions enables a significant increase in the number of frames that the model can process each second with only a minor impact on accuracy.

Table 3.8: Model accuracy after training with the MMIT dataset using various frame rates.

Training \ Testing	1 fps	2 fps	4 fps	8 fps
1 fps	**48.64**	41.76	41.89	36.84
2 fps	50.19	51.49	**52.14**	50.97
4 fps	51.23	49.94	**53.44**	53.18
8 fps	50.84	49.94	53.31	**54.22**

Table 3.9: Model accuracy after training with the HMDB51 dataset using various frame rates.

Training \ Testing	1 fps	2 fps	4 fps	8 fps
1 fps	**79.51**	75.77	78.22	73.93
2 fps	76.87	80.80	**84.60**	**84.60**
4 fps	77.06	78.83	**86.81**	86.50
8 fps	73.93	78.83	84.72	**85.21**

Table 3.10: Model accuracy after training with the Hollywood dataset using various frame rates.

Training \ Testing	1 fps	2 fps	4 fps	8 fps
1 fps	**67.35**	59.18	61.22	51.02
2 fps	53.06	63.27	63.27	**67.35**
4 fps	63.27	**65.31**	**65.31**	**65.31**
8 fps	61.22	51.02	51.02	**75.51**

Table 3.11: The best accuracies achieved in every class of the MMIT dataset using different frame rates.

Class	1 fps	2 fps	4 fps	8 fps
fighting	45.00	**75.00**	60.00	55.00
punching	75.68	75.67	**83.78**	**83.78**
pushing	46.15	46.15	38.46	**53.85**
sitting	30.00	26.67	30.00	**36.67**
sleeping	56.84	49.47	**60.00**	53.68
coughing	21.43	28.57	21.43	**35.71**
singing	**76.17**	73.36	71.50	73.36
speaking	57.14	53.57	57.14	**64.29**
discussing	29.16	26.38	**33.33**	**33.33**
pulling	45.45	59.09	54.55	**63.64**
slapping	25.80	32.26	**35.48**	25.81
hugging	82.35	82.35	**88.23**	82.35
kissing	9.09	**27.27**	**27.27**	18.18
reading	14.29	14.28	**28.57**	**28.57**
telephoning	45.45	45.45	**54.55**	45.45
studying	64.71	58.82	**82.35**	58.82
socializing	15.00	20.00	20.00	**25.00**
resting	40.00	53.33	**66.67**	53.33
celebrating	**70.83**	**70.83**	**70.83**	66.66
laughing	**20.00**	**20.00**	**20.00**	**20.00**
eating	57.14	**71.43**	57.14	57.14

Table 3.12: The best accuracies achieved in every class of the HMDB51 dataset using different frame rates.

Class	1 fps	2 fps	4 fps	8 fps
punching	73.07	**84.61**	79.23	81.54
pushing	**92.59**	91.85	**92.59**	**92.59**
speaking	67.19	67.19	77.19	**83.75**
hugging	84.29	95.24	96.67	**100.00**
kissing	96.66	**100.00**	98.67	92.67
laughing	90.83	90.00	**94.58**	92.08
eating	71.66	78.33	87.77	**87.78**

Removing part of the visual information caused a slight decrease in performance for most datasets. Fine tuning the model using data augmented with different types of cropping overall resulted in worse performance, suggesting that training with full frames is better than training with frames missing crucial information.

Using different frame rates in training and testing could be useful. However, the results indicate that the model had the highest accuracy when training and testing on the same frame rates. This suggests that the model should be trained with a frame rate similar to the frame rate provided by the final system. We can see that higher frame rates generally lead to higher accuracy, since the model is presented with more precise information about the movement. The increase in accuracy from using higher frame rates is not very high, therefore using lower frame rates might be advantageous in systems with limited computational resources. Additionally, we can see that classes that contain slow movements (e.g. singing, kissing) do not see performance gains when using higher frame rates, as the model is not being presented with any additional information.

Table 3.13: The best accuracies achieved in every class of the Hollywood dataset using different frame rates.

Class	1 fps	2 fps	4 fps	8 fps
hugging	**45.45**	9.09	36.36	27.27
kissing	87.50	95.83	95.83	**100.00**
telephoning	71.43	57.14	71.43	**85.71**

Chapter 4

Towards vehicle occupant-invariant models for activity characterisation

Foreword on Author Contributions

The results of this work has been disseminated in an article in international journal:

[13] Capozzi, Leonardo and Barbosa, Vítor and Pinto, Carolina and Pinto, João Ribeiro and Pereira, Américo and Carvalho, Pedro M. and Cardoso, Jaime S., "Toward Vehicle Occupant-Invariant Models for Activity Characterization," in IEEE Access, vol. 10, pp. 104215-104225, 2022, doi: 10.1109/ACCESS.2022.3210973

4.1 Introduction

Successfully capturing passenger activity has far-reaching implications for both the user experience and safety features in Autonomous Vehicles (AVs). Without a driver responsible for the safety and integrity of the vehicle and its occupants, it is incumbent upon automated detection systems to monitor occupant well-being and actions and to detect potentially harmful behaviour or even violence. However, the multitude of possible actions that can be represented, the variability in how different individuals represent the same actions, the heterogeneity of sensors and types of information collected, and the influence of external factors still pose significant challenges to this task [12]. The current state of the art in action recognition is based on deep models, but their use for vehicle occupant action recognition is not without problems.

Training deep learning models requires significant amounts of data, which escalates with the complexity of the models to avoid overfitting. Moreover, the available datasets are usually split, with one part used for training and another part used for subsequent testing. Dataset availability and size can be critical when dealing with individuals engaged in different activities, both for legal reasons and because of the effort involved in preparation. If the dataset used to train a model contains only a small number of actors, the model can be biased toward them, i.e., to specific individuals and their unique characteristics [42]. This causes the model to perform poorly when

confronted with a different set of actors performing the same activities. In the context of this study, an actor is defined as a specific group of people in the vehicle (rather than a specific person). This means that different actors may have some individuals in common. In this work, we focus on the detection of violence and non-violence in the vehicle and describe and evaluate three methods that aim to counteract actor bias by removing actor-specific information from the features of the model, or by providing the model with data that is independent of the actors, such as information about posture, which contributes to a more robust model.

The presented research focuses on the monitoring of occupants of autonomous vehicles, more specifically on the detection of violence and non-violence. We believe that this scenario is particularly relevant since: (1) there is a strong focus on people with high heterogeneity; (2) to our knowledge, there are no available datasets for this scenario, leading to a high dependence of the models on the actors. Our main contributions are:

- Using a dataset that is domain-specific to the target scenario;
- The application of domain generalization methodologies;
- The application of regularization techniques for the specific scenario of the vehicle interior;
- The use of body posture to diversify data sources and consequently reduce actor bias.

In addition to the introduction, this paper contains 5 other sections. Section 4.2 presents the current state of the art for this problem. In section 4.3 the methods used in this study are reviewed. In section 4.4 the data and experimental settings used in this work are presented. In section 4.5 we analyse the results. Section 4.6 gives the conclusions from this work and presents ideas for future work.

4.2 Related Work

4.2.1 Video Action Recognition

Action classification requires models that focus on modelling spatio-temporal information. Early attempts at action recognition used compact video descriptors to extract handcrafted spatio-temporal features [43], [44], [45]. In recent years, the use of these techniques has declined, and deep learning-based methods have pushed the boundaries of the state-of-the-art in action recognition. However, despite their success, these models have raised concerns about their bias towards specific domain characteristics. This problem arises because deep learning methods rely on training data. This dependency leads to the developed models being biased towards, for example, actor features when the dataset used lacks diversity.

One of the most successful deep learning approaches is the two-stream model [46]. In these models, there are two separate deep Convolutional Networks (ConvNets) for processing RGB frames and optical flow, which are then combined by late fusion [47]. Based on this model, several approaches to action recognition have been proposed [48], [49], [50], [51]. Another approach that

is proving successful is the use of 3D CNN. These networks behave similarly to 2D CNN, but use 3D convolutions to extract features in spatial and temporal dimensions [52]. The C3D [53] network is an example of this type of approach, where the spatio-temporal features are learned to use 3D ConvNets trained on large-scale supervised video datasets. The use of large-scale video datasets was possible because C3D can handle video frames as input without any preprocessing. Other variants based on 3D CNN have been proposed [54], [55]. One worth mentioning is an architecture that combines the two-stream model with 3D ConvNet, called I3D [55], which served as the basis for the model created in this work and is still one of the best performing architectures for action recognition. One of the problems introduced by the use of 3D convolutions is the high computational cost. To solve this problem, several works have proposed to treat the spatial and temporal dimensions differently. Some proposed the decomposition of 3D convolutions into 2D spatial convolutions and 1D temporal convolutions, such as S3D [56], P3D [57], R(2+1)D [58]. Furthermore, SlowFast [59] shows that space and time should not be treated symmetrically. Therefore, it introduces a two-path structure to handle slow and fast motion separately.

The models mentioned so far use random spatial crops of video frames as inputs during training. Since they do not explicitly focus on the human body, it is easy to overfit the scenes and objects in the video. Therefore, skeleton data was used to focus the action recognition on the human body. This has the advantage of being lightweight and free of scene cues. As for pose-based methods for action recognition, the main differences are the use of CNN or Graph Convolution Network (GCN). CNN-based [60], [61], [62], [63] methods represent the skeleton with a pseudo-image and thus recognize actions in the same way as image classification. Nevertheless, skeleton data is essentially a graph in non-Euclidean space with skeleton joints as vertices and bones as edges. Therefore, GCN-based [64], [65], [66] methods were proposed to capture joint interactions on the skeleton graphs, explicitly considering the adjacent relationship between joints in the non-Euclidean space.

4.2.2 Representation Biases

Representation bias is a problem in various image and video classification problems, and studies have been conducted to analyse and mitigate it. Li et al. mitigated scene, object and people biases by re-sampling the original video datasets [67], [68]. The re-sampling approach reduces representation bias, but it also reduces the number of training data, which is not desirable for deep learning methods. Another proposed method was to use adversarial losses for different scene types to mitigate scene biases [69]. Similarly, adversarial learning procedures allowed learning signer-invariant latent representations to be highly discriminating for sign recognition [42].

The primary source of representation bias is the dataset used for model training. Some studies have shown that video datasets for action recognition exhibit biases toward objects, scenes, and people [67], [68], [69], [70]. Video datasets for action recognition are mainly divided into generic and fine-grained action recognition datasets. Generic action recognition datasets provide the generic action recognition task, which attempts to classify various action categories in various domains. Such datasets include videos from different domains, such as daily activities, sports, and entertainment. Due to the wider diversity of domains in these datasets, the models trained on them

can recognise actions indirectly, i.e., by the presence of an object or a person. The use of such biased models leads to degenerate transferability and incorrectly recognizes novel actions in the same static cues. Popular datasets for generic action recognition are Kinetics [71], Moments in Time [72], ActivityNet [73], UCF-101 [74], and HMDB-51 [75]. Fine-grained action recognition datasets have been recently released, providing videos in a specific domain. Something-Something [76] includes the videos of fine-grained actions of human-object interactions. Jester [77] is a dataset for hand gesture recognition. Diving48 [67] is a video dataset in sports. These datasets overcome some problems associated with representation biases since the same domain's static cues, such as objects and scenes, are similar. Therefore, it is not easy to recognise actions based on only static cues. However, these datasets might face biases in domain static cues related to people. This happens especially in datasets with a small set of actors. In this study, the datasets used are fine-grained action recognition datasets. Nevertheless, the number of actors present is quite small. Therefore, several techniques were implemented to improve the generalisation capability regarding the actors' characteristics.

4.2.3 In-vehicle Occupant Activity Recognition

In-vehicle activity recognition is still relatively unexplored. Some proposals addressing this topic demonstrated that activity recognition might be a possible approach for monitoring the vehicle's occupants. Some proposals focused on violence detection to monitor occupants through anomaly detection of occupant's behaviour [78] or recognition of occupant's interactions [79]. Beyond detecting violence, the recognition of occupant's activity using different modalities has also been explored, specifically using audiovisual features [12]. Audio features appeared from applying this type of feature for the classification of group emotion [32]. This work focused primarily on the available hardware and energy consumption constraints associated with implementing an activity recognition system in a vehicle. Using an audio module and a cascading strategy demonstrated reductions in memory requirements and computational demands. This reduction was most significant when the audio module was the first processing block. This research paper continues these previous studies, focusing on a hypothesis previously identified of the possibility of bias in the results obtained so far. This hypothesis arose because the past studies used small datasets with a small set of actors.

4.3 Methodologies

In this section, the baseline method used in this work and the methods adapted to promote actor generalization are reviewed. The goal of the latter is to accurately predict violence and non-violence in videos of vehicle occupants, regardless of the actors present. Each of these methods aims to achieve this goal in different ways.

The Adversarial Learning method [42] uses an additional network that learns to classify the different actors in the training set and later uses that knowledge to remove actor information from the features of the network, making it less biased toward the actors.

The Bilevel Learning methodology [80] assigns a different weight to each mini-batch of the training set based on how much its gradients agree with the gradients of a validation mini-batch. This reduces the actor bias by giving more weight to training mini-batches that have similar gradients to validation mini-batches, minimizing the error on the validation set and leading to better generalization capabilities.

The use of pose information is also studied, both as a substitute and as a complement to the RGB information. We calculate the pose of each person in the frame and generate the keypoint map that is then given to the network [81]. Since the keypoint map does not contain any direct information about the actors, it helps to reduce actor bias.

4.3.1 Baseline

The baseline model used in this work is based on the model proposed in [32], presented in Figure 4.1. It uses RGB data for recognizing actions, and it has achieved results comparable to state-of-the-art methodologies. This model is a 3D CNN, which allows the network to take advantage and use the temporal information more efficiently [22], [23], [24], [25], [26], [27], [28], [29], [30], [31]. The output of 3D Convolutions is a video volume that retains the temporal information of the input, making them superior for video classification than standard 2D Convolutions, which lack the extra dimension used for the temporal component.

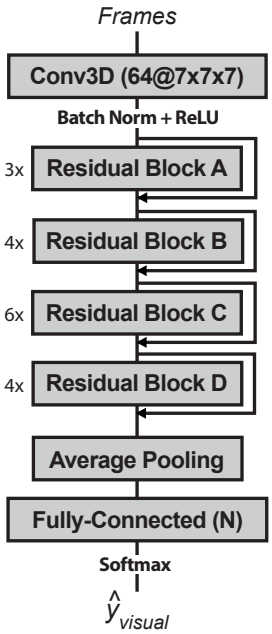

Figure 4.1: Schema of the network used for activity recognition (adapted from [32]).

The model is composed of the 3D ResNet50's convolutional encoder blocks, an average pooling layer, a dropout layer, and a fully-connected layer with 2048 inputs and 2 outputs (predicting violence or non-violence). The convolutional encoder has one 3D convolutional layer with 64 filters

of size $7 \times 7 \times 7$, a batch normalization layer and ReLU activation function. The following layers are three residual blocks of type A, four blocks of type B, six blocks of type C, and four blocks of type D. Residual blocks contain three convolutional layers, with a filter size of $1 \times 1 \times 1$, $3 \times 3 \times 3$, and $1 \times 1 \times 1$, respectively. The number of filters of the first two convolutional layers depends on the type of block. Type A contains 64 filters, type B contains 128 filters, type C contains 256 filters, and type D contains 512 filters. The first two convolutional layers are followed by a batch normalisation layer. The last convolutional layer of each block contains four times the number of filters of the first two layers and is followed by a batch normalisation layer and ReLU activation function. More details on the network architecture can be seen on the original ResNet paper [33].

The used network was pre-trained with videos of the Moments in Time dataset. During training, the layers of the network were frozen, except for the last 2 layers. Class weights were added to the cross-entropy loss function to combat class imbalances. The Adam optimizer was used with a learning rate of 1×10^{-4}. The work was developed using PyTorch. In all the experiments, we used the balanced accuracy to combat any class imbalances in the datasets.

4.3.2 Adversarial learning

To accurately predict violence and non-violence in video, independently of the actors that are present, the model should be trained in a way where the latent representations learned by the model preserve the information relative to the action, and discard the information relative to the actors, which may negatively impact the classification. To accomplish this, the methodology proposed in [42] was adapted. It presents a network architecture and an adversarial training objective, which addresses the signer-independent problem (actor-independent problem, in our case). The model consists of a feature extractor, which maps input videos to latent representations, and two classifiers. In our case, the network architecture is the same as our baseline model. The feature extractor is the inflated ResNet-50 network, and the two classifiers are dense layers. The action-classifier predicts violence or non-violence, and the actor-classifier predicts the actor present in the video.

During the learning process, the feature extractor is simultaneously trained to help the action-classifier while trying to fool the actor-classifier. Figure 4.2 shows an overview of the network architecture and the loss functions.

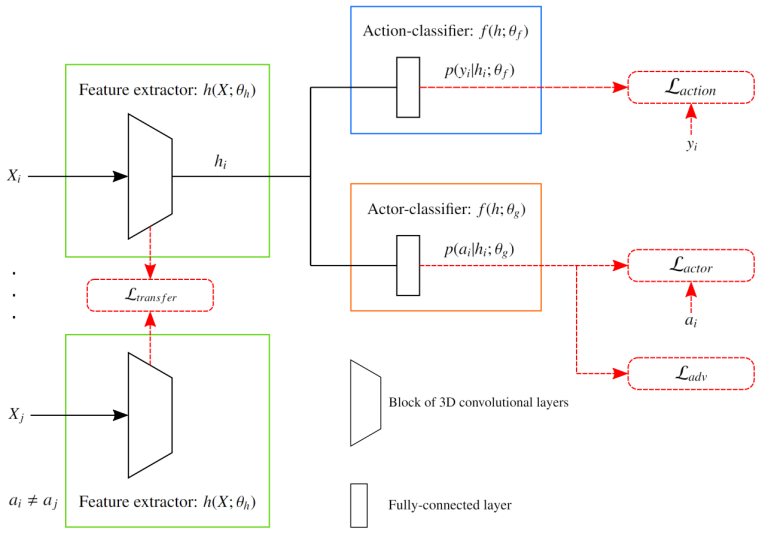

Figure 4.2: Architecture of the adversarial learning methodology (adapted from [42]).

Let $X = \{X_i, y_i, a_i\}_{i=1}^{N}$ represent a labeled dataset with N samples, where X_i represents the i-th video sample (a set of 8 concatenated RGB frames), y_i represents the action label (violence or non-violence), a_i represents the actor label and A represents the set of actor labels.

The feature extractor learns an encoding function $h(X;\theta_h)$, which using the parameters θ_h encodes an input video sample X into a latent representation h.

The action-classifier receives the latent representation h and learns a function $f(h;\theta_f)$, parameterized by θ_f, that gives the predicted probabilities $p(y|h;\theta_f)$ for each action class.

The actor-classifier learns a function $f(h;\theta_g)$, which using the parameters θ_g maps the latent representation h to the predicted probabilities $p(a|h;\theta_g)$ for each actor.

The actor-classifier is trained to minimize the negative log-likelihood of correct actor predictions:

$$\min_{\theta_g} \mathcal{L}_{actor}(\theta_h, \theta_g) = -\frac{1}{N}\sum_{i=1}^{N} \log p(a_i|h(X_i;\theta_h);\theta_g) \tag{4.1}$$

The feature extractor and the action-classifier are trained to minimize the negative log-likelihood of correct action predictions:

$$\min_{\theta_h,\theta_f} \mathcal{L}_{action}(\theta_h, \theta_f) = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_i|h(X_i;\theta_h);\theta_f) \tag{4.2}$$

Additionally, the predictions of the actor-classifier should be close to uniform, meaning that it is not capable of doing better than random guessing the actor identity. The following loss (eq. 4.3) adjusts the weights of the feature extractor to make the predictions of the actor-classifier close to uniform, and it is an adversarial loss with respect to the actor classification loss $\mathcal{L}_{actor}$:

$$\min_{\theta_h} \mathscr{L}_{adv}(\theta_h, \theta_g) = -\frac{1}{N|A|} \sum_{i=1}^{N} \sum_{a \in A} \log p(a|h(X_i;\theta_h);\theta_g) \tag{4.3}$$

In order to further encourage the actor invariance properties of the latent representation h, another loss $\mathscr{L}_{transfer}$ was added. It minimizes the distance between the hidden latent representations of different actors at each layer of the feature extraction network. The distance $D^{(m)}$ between the latent representations $h^{(m)}(\bullet;\theta_h)$ of actors a and t, at the m-th layer is calculated as follows:

$$D^{(m)}(a,t;\theta_h) = \left\| \frac{1}{N_a} \sum_{i:a_i=a} h^{(m)}(X_i;\theta_h) - \frac{1}{N_t} \sum_{j:a_j=t} h^{(m)}(X_j;\theta_h) \right\|_2^2, \tag{4.4}$$

where $||\bullet||_2$ is the $l2$-norm, and N_a and N_t represent the number of training examples of actors a and t, respectively. This assumes that the dataset is balanced in respect to the action labels for each actor. If this is not the case, each mini-batch used during training could be designed to fulfil this requirement.

To calculate the actor transfer loss at the m-th layer, the pairwise distances between all actors are summed:

$$\mathscr{L}^{(m)}_{transfer}(\theta_h) = \sum_{a \in A} \sum_{t \in A, t \neq a} D^{(m)}(a,t,\theta_h) \tag{4.5}$$

The final loss $\mathscr{L}_{transfer}$ is a weighted sum of the loss calculated at each layer of the feature extraction network, where $\beta^{(m)}$ is the weight attributed to the layer m, which controls the importance of that layer:

$$\mathscr{L}_{transfer}(\theta_h) = \sum_{m=1}^{M} \beta^{(m)} \mathscr{L}^{(m)}_{transfer}(\theta_h) \tag{4.6}$$

Therefore, the final objective can be written as:

$$\min_{\theta_h, \theta_f} \mathscr{L}(\theta_h, \theta_f, \theta_g) = \mathscr{L}_{action}(\theta_h, \theta_f) + \lambda \mathscr{L}_{adv}(\theta_h, \theta_g) + \gamma \mathscr{L}_{transfer}(\theta_h) \tag{4.7}$$

where λ and γ are weights attributed to each loss component to control their relative importance.

4.3.3 Deep Bilevel learning

In this study, the Deep Bilevel Learning [80] methodology is explored as another strategy to achieve actor generalization. Deep Bilevel Learning improves the training process by giving different weights to each mini-batch in the training set. These mini-batch weights favour the batches whose gradients match the gradients of the validation mini-batches, minimizing the error on the validation set and resulting in a model with better generalization capabilities. A batch size of 16 was used, and for each weight update, 4 training mini-batches and 1 validation mini-batch with the same class distributions were selected. Figure 4.3 shows an overview of the methodology.

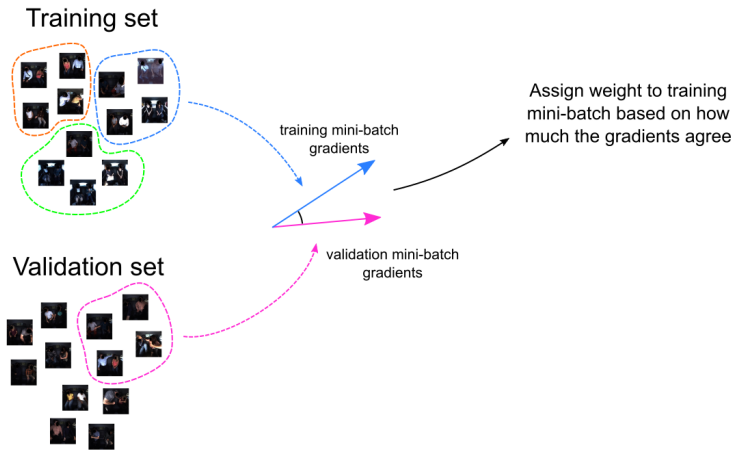

Figure 4.3: Overview of the Deep Bilevel Learning methodology (adapted from [80]).

The weights are calculated using the following function:

$$\forall i \in T^t, \omega_i \leftarrow \sum_{j \in V^t} \frac{\nabla l_j(\theta^t)^T \nabla l_i(\theta^t)}{|\nabla l_i(\theta^t)|^2/\hat{\lambda} + \hat{\mu}}, \hat{\omega} = \frac{\omega}{|\omega|_1}, \tag{4.8}$$

where T^t represents the collection of training mini-batches used at the t-th training iteration, V^t is the collection of validation mini-batches used at the t-th training iteration, θ^t are the model parameters at the t-th training iteration, $\nabla l_i(\theta^t)$ are the gradients of the i-th mini-batch in the training set, $\nabla l_j(\theta^t)^T$ are the gradients of the j-th mini-batch in the validation set, ω_i is the weight attributed to the i-th mini-batch in the training set, $\hat{\lambda}$ is an adjustable hyperparameter, and $\hat{\mu}$ is a term added to avoid divisions by zero.

A new gradient descent step can be calculated as follows:

$$\theta(\omega) = \theta^t - \varepsilon \sum_{i \in T^t} \hat{\omega}_i \nabla l_i(\theta^t), \tag{4.9}$$

where $\varepsilon\hat{\omega}_i$ can be interpreted as a learning rate specific to each training mini-batch. When the gradients of a mini-batch in the training set $\nabla l_i(\theta^t)$ point in the same direction as the gradients of a mini-batch in the validation set $\nabla l_j(\theta^t)$, then their inner product is $\nabla l_j(\theta^t)^T \nabla l_i(\theta^t) > 0$; when the gradients point in different directions (do not agree) the inner product is $\nabla l_j(\theta^t)^T \nabla l_i(\theta^t) \leq 0$ which gives a weight with a negative value or zero.

4.3.4 Using pose information

The baseline model uses RGB data for action recognition and has achieved results comparable to state-of-the-art methodologies.

A series of experiments were conducted to measure the effects of using pose information as a complement or alternative to RGB data [81]. Using pose information as an input can have some advantages, as the model receives data that is simpler and easier to interpret, and can also

help improve robustness to the actors. When using the raw RGB data, however, the model needs to extract and interpret more complex features, which can make the classification process more complex.

To calculate the key points of each person in a frame, a Keypoint R-CNN model with a ResNet-50-FPN backbone [82] is used. The model calculates the position of 17 key points, and whether they are visible or occluded. The key points calculated by the model are presented in Table 4.1 along with the corresponding assigned label (colour) and further illustrated in Figure 4.4.

Table 4.1: Key points calculated by the pose estimation model and their corresponding colour.

Keypoint	Colour (RGB)
nose	(128, 0, 0)
left eye	(0, 128, 0)
right eye	(128, 128, 0)
left ear	(0, 0, 128)
right ear	(128, 0, 128)
left shoulder	(0, 128, 128)
right shoulder	(128, 128, 128)
left elbow	(64, 0, 0)
right elbow	(192, 0, 0)
left wrist	(64, 128, 0)
right wrist	(192, 128, 0)
left hip	(64, 0, 128)
right hip	(192, 0, 128)
left knee	(64, 128, 128)
right knee	(192, 128, 128)
left ankle	(0, 64, 0)
right ankle	(128, 64, 0)

Figure 4.4: Key points calculated by the pose estimation model (Sunnyvale dataset from Bosch Car Multimedia).

To calculate the pose information for a certain video, we start by extracting each frame of the video. The pose information is extracted for each frame, consisting of the coordinates of the key points and the bounding box of each person in the frame.

For each frame, we generate an image with a black background and circles of a certain colour with a radius of 4 pixels, in the position of each key point. We added a different colour to each key point to label them, to make it possible for the model to distinguish each body part. Table 4.1 shows the key points and their corresponding colour. Figure 4.4 shows an example of an image and the corresponding key point map.

4.4 Data and Experimental Settings

As previously stated, the proposed methodologies are tested in a shared autonomous vehicle scenario. Given the lack of freely available appropriate data, two datasets from Bosch Car Multimedia were used, which will be designated Hanau03 Vito and Sunnyvale. Each of the datasets contains videos of people performing different actions inside the vehicle.

The Hanau03 Vito dataset contains 74 actors, and the videos were captured from above with a fisheye lens. Figure 4.5 shows a frame extracted from one of the videos.

Figure 4.5: Frame extracted from the Hanau03 Vito dataset from Bosch Car Multimedia.

The Sunnyvale dataset has 9 actors, and the videos were recorded from the front without a fisheye lens, making them more similar to the videos in the MMIT dataset, which was used to pre-train the video processing sub-module. Figure 4.6 shows a frame extracted from one of the videos.

Figure 4.6: Frame extracted from the Sunnyvale dataset from Bosch Car Multimedia.

In these experiments, we focus on detecting violence and non-violence. Each annotated video segment of the dataset was divided into sub-segments that will be referred to as samples. These samples are then used to train and test the model. Each sample has a duration of 1 second, and a frame-rate of 8 FPS, which means that for each prediction, the model receives 8 concatenated frames with a resolution of $224 \times 224 \times 3$ pixels. The number of samples extracted were 32739 for the Hanau03 Vito dataset, and 46838 for the Sunnyvale dataset.

Table 4.2 contains the number of samples that were extracted from the datasets, the number of actors, and the average number of samples per actor.

Table 4.2: Number of samples and actors in the Hanau03 Vito and Sunnyvale datasets.

	No. Samples	No. Actors	No. Samples/Actor (mean $\pm$ 2 $\times$ std)
Hanau03 Vito	32739	74	442 ± 968
Sunnyvale	46838	9	5211 ± 6828

As previously mentioned, an actor is defined as a specific group of people inside the vehicle (and not a specific person). This means that different actors could have some individuals in common.

When splitting the dataset into train set, validation set and test set, it was ensured that each specific person was not in different sets simultaneously, meaning that each of the train, validation, and test sets were completely independent of each other in terms of actors. The datasets were split in a way that kept the number of samples in each class relatively balanced. Table 4.3 and Table 4.4 show the dataset splits.

Table 4.3: Hanau03 Vito dataset train-validation-test split.

	Non-violence samples	Violence samples	No. Samples	No. Actors
Training Set	9752	11744	21496	49
Validation Set	3255	3372	6627	10
Test Set	1762	2854	4616	15
Total	14769	17970	32739	74

Table 4.4: Sunnyvale dataset train-validation-test split.

	Non-violence samples	Violence samples	No. Samples	No. Actors
Training Set	16351	16416	32767	5
Validation Set	5137	4864	10001	3
Test Set	3093	977	4070	1
Total	24581	22257	46838	9

Since the focus of this work is the detection of violence/non-violence the original classes of the datasets were grouped into those two categories. Table 4.5 and Table 4.6 show the original classes and their classification as violence or non-violence.

In addition to the Hanau03 Vito and Sunnyvale datasets three widely used and publicly available datasets are used: Moments in Time [34]; HMDB51 [75]; Hollywood [83]. When analysing the pose keypoint data it was noticed that the pose keypoints calculated for the Hanau03 Vito dataset were inaccurate in some situations due to the position of the camera (top view). Therefore, to further test the methodology that used the pose information, we decided to add these datasets. The advantages of these datasets are that they are all publicly available, and they contain actions that could be performed inside the vehicle.

The Moments in Time dataset is a large-scale action dataset. It contains one million 3-second videos and 339 classes. The HMDB51 dataset contains 6849 video clips from 51 action classes obtained from movies and web videos. The Hollywood dataset is a human action dataset which contains video samples obtained from 32 movies. Each sample has one or more labels corresponding to 8 action classes.

For each of the publicly available datasets, a subset of action classes was selected, based on the relevance for this study and for the context of in-vehicle action recognition.

Table 4.5: Action grouping as violence or non-violence for the Hanau03 Vito dataset.

Class	Violence
entering	No
leaving	No
blucke on	No
turning head	No
lay down	No
sleeping	No
stretching	No
changing seating position	No
changing clothes	No
reading	No
use mobile phone	No
making a call	No
posing	No
waving hand	No
drinking	No
eating	No
singing	No
pick up item	No
come closer	No
handshaking	No
talking	No
dancing	No
finger pointing	No
leaning forward	No
tickling	No
hugging	No
kissing	No
elbowing	Yes
provocative	Yes
pushing	Yes
protecting oneself	Yes
stealing	Yes
screaming	Yes
pulling	Yes
arguing/abuse	Yes
grabbing	Yes
touching	Yes
slapping	Yes
pushing	Yes
strangling	Yes
fighting	Yes
threatening weapon	Yes

Table 4.6: Action grouping as violence or non-violence for the Sunnyvale dataset.

Class	Violence
handshaking	No
hugging	No
talkRight	No
talkLeft	No
talking	No
arguing	Yes
touching	Yes
grabbing	Yes
pushing	Yes
kicking	Yes
slapping	Yes
elbowing	Yes
punching	Yes
fighting	Yes

4.5 Results

4.5.1 Adversarial learning

After training on the Hanau03 Vito dataset the baseline had an accuracy of 65.33% and the adversarial learning methodology had an accuracy of 62.42%. The actor classification accuracy during training was 2.04%. The actor accuracy should be close to random, as this indicates that the methodology is correctly removing the actor information from the features of the model. Table 4.7 shows the results.

Table 4.7: Model accuracy after training on the Hanau03 Vito dataset using the baseline and the adversarial learning methodology.

	Baseline Accuracy (%)	Adversarial Learning Accuracy (%)
Training Set	98.75	94.11
Validation Set	57.18	54.25
Test Set	65.33	62.42

After training on the Sunnyvale dataset the baseline had an accuracy of 82.81% and the adversarial learning methodology had an accuracy of 58.11%. The actor classification accuracy during training was 35.96%. Table 4.8 shows the results.

Table 4.8: Model accuracy after training on the Sunnyvale dataset using the baseline and the adversarial learning methodology.

	Baseline Accuracy (%)	Adversarial Learning Accuracy (%)
Training Set	99.25	98.84
Validation Set	89.30	77.62
Test Set	82.81	58.11

The obtained results indicate that the presented methodology did not improve the results of the baseline model. The hyperparameters used for the experiments were the ones proposed in the paper [42], and given that this problem uses a different dataset and a different network architecture, some tuning could be needed. Additionally, the datasets used are very small and contain a low number of samples per actor, which further increases the difficulty of the problem by making the network unable to correctly learn the distribution of each actor, making this methodology less effective. An interesting experiment would be to train the model from scratch, to be able to remove actor information from the first layers of the network, but more data would be required, since training the model from scratch with these datasets would result in an overfit model that would perform poorly when presented with new data.

4.5.2 Bilevel learning

After training on the Hanau03 Vito dataset, the model achieved an accuracy of 65.33% on the baseline and 68.19% on the bilevel learning methodology. Table 4.9 show the results.

Table 4.9: Accuracy after training on the Hanau03 Vito dataset using the baseline and the bilevel learning methodology.

	Baseline Accuracy (%)	Bilevel Learning Accuracy (%)
Training Set	98.75	91.70
Validation Set	57.18	69.77
Test Set	65.33	68.19

After training on the Sunnyvale dataset the model achieved an accuracy of 82.81% on the baseline and 86.62% on the bilevel learning methodology. Table 4.10 show the results.

Table 4.10: Accuracy after training on the Sunnyvale dataset using the baseline and the bilevel learning methodology.

	Baseline Accuracy (%)	Bilevel Learning Accuracy (%)
Training Set	99.25	92.15
Validation Set	89.30	89.96
Test Set	82.81	86.62

Although the gradients of the validation set are not used directly to train the model, there may be some data leakage during the training process, since more weight is given to the training mini-batches that have gradients similar to the gradients of the validation mini-batches. This is also a reason why the data is split into training, validation and testing.

On both datasets, the training set accuracy decreased when using bilevel learning, which means that the model is not overfitting as much to the training data. There is also an increase in accuracy in both the validation set and the test set.

4.5.3 Using pose information

Previously, the input of the model was the RGB data of the video frames. This section presents two experiments that aim to assess if inputting pose information into the model has any impact on the performance.

The first experiment consists of training the model using only pose information, where instead of giving the model the RGB frames, the model is given the generated key point map. The second experiment consists of training the model with the pose information as a complement to the RGB frames. To accomplish this, the RGB frames are concatenated with the key point maps before giving them to the model, which results in a total of 6 channels (labels in the key point map are colours expressed in RGB). To accommodate for this change, the number of input channels of the network is duplicated and the pre-trained weights are copied to the new channels. The first layer of the network is also unfrozen, as 3 more channels were added to the filters of this layer.

Table 4.11, Table 4.12 and Table 4.13 show the results on the Moments in Time, HMDB51 and Hollywood datasets, respectively.

Table 4.11: Results after training the model using the Moments in Time dataset, with RGB, pose only, and RGB + pose.

Class	RGB (%)	Pose (%)	RGB + Pose (%)
fighting	**60.00**	20.00	55.00
punching	72.97	8.11	**78.38**
pushing	38.46	30.77	**61.54**
sitting	23.33	3.33	**33.33**
sleeping	49.47	3.16	**53.68**
coughing	14.28	0.00	**28.57**
singing	**72.42**	58.88	68.69
speaking	**50.00**	45.24	32.14
discussing	26.38	11.11	**31.94**
pulling	36.36	9.09	**40.91**
slapping	29.03	9.68	**48.39**
hugging	**82.35**	5.88	**82.35**
kissing	**18.18**	0.00	**18.18**
reading	14.28	0.00	**28.57**
telephoning	**45.45**	0.00	**45.45**
studying	58.82	0.00	**76.47**
socializing	10.00	**15.00**	10.00
resting	46.66	13.33	**53.33**
celebrating	**66.66**	25.00	62.50
laughing	**20.00**	0.00	**20.00**
eating	**71.42**	0.00	**71.42**
all classes	43.17	12.31	**47.66**

Table 4.12: Results after training the model using the HMDB51 dataset, with RGB, pose only, and RGB + pose.

Class	RGB (%)	Pose (%)	RGB + Pose (%)
punching	67.31	43.46	**74.81**
pushing	91.48	67.78	**92.59**
speaking	63.44	40.47	**69.38**
hugging	**95.24**	72.62	89.76
kissing	**100.00**	42.67	90.67
laughing	83.75	72.29	**92.08**
eating	**78.33**	51.11	73.61
all classes	82.79	55.77	**83.27**

Table 4.13: Results after training the model using the Hollywood dataset, with RGB, pose only, and RGB + pose.

Class	RGB (%)	Pose (%)	RGB + Pose (%)
hugging	9.09	6.82	**20.45**
kissing	**91.67**	68.33	83.54
telephoning	57.14	36.07	**62.50**
all classes	52.63	37.07	**55.50**

Results show that using the pose information as an alternative to the RGB data does not translate into an improvement in performance. The only information that is given to the model is the position of the body parts in each frame, which makes the problem more difficult as there is less available information.

The results on the model trained with the RGB frames and the pose information suggest that there is an improvement for most classes. This could mean that there is an advantage to giving the model the pose information, as it can more easily extract relevant information about movement or the position of people in a given frame, improving the accuracy of the network.

There seems to be no advantage in using the pose information for the "hugging", "kissing" and "eating" classes, when comparing the classes in common between the Moments in Time and HMDB51 datasets. The Hollywood dataset also shared the same results, showing no improvement when using the pose information for the class "kissing". This might mean that some classes benefit more from using the pose information than others.

Table 4.14 and Table 4.15 show the results on the Hanau03 Vito and Sunnyvale datasets, respectively. Since there was the need to unfreeze the first layer of the network for the RGB + Pose experiment, we wanted to test if unfreezing the first layer on the other experiments would have any significant impact on the performance. Therefore, the tables present the results with the first layer frozen and unfrozen.

Table 4.14: Results after training the model using the Hanau03 Vito dataset, with RGB, pose only, and RGB + pose.

Class	RGB unfrozen (%)	RGB	Pose unfrozen (%)	Pose	RGB + Pose (%)
non-violence	65.83	**66.00**	59.93	62.59	45.51
violence	72.98	64.64	53.95	60.72	**74.38**
all classes	**69.41**	65.33	56.94	61.66	59.95

Table 4.15: Results after training the model using the Sunnyvale dataset, with RGB, pose only, and RGB + pose.

Class	RGB unfrozen (%)	RGB	Pose unfrozen (%)	Pose	RGB + Pose (%)
non-violence	21.11	74.62	**78.14**	70.22	37.31
violence	**96.11**	90.99	90.17	89.55	91.09
all classes	58.61	82.81	**84.16**	79.89	64.20

Once again, the model achieves reasonable results when using the pose information as a complement to the traditional RGB frames. It also seems that due to the more controlled scenario found in the Sunnyvale dataset, using only the pose information was enough to achieve results superior or comparable to the baseline (RGB column). The results on the Hanau03 Vito dataset did not show any improvements over the baseline model when using the pose information. This could be caused by the camera angle of the dataset (top view), since it makes it difficult to detect the keypoints, making them inaccurate and thus affecting the results.

Since the pose information only contains the position of the actor and its body parts, the model is less likely to become biased towards a certain actor. The downside is that the performance of the model is dependent on the accuracy of the keypoints.

4.6 Conclusions and Future Work

This chapter focused on methodologies that counteract actor bias, by removing specific actor information from the features of the model, or by using data that is actor independent, such as pose information.

The first methodology used an adversarial approach to remove actor-related information from the feature vectors of the model. Although the results did not show an improvement, some fine-tuning of the hyperparameters could give better results.

The second methodology assigned a weight to training mini-batches based on how much their gradients "agreed" with the gradients of the validation mini-batches. Results show an improvement over the baseline, and since the gradients of the validation set are taken into account (which contain different actors), this methodology is giving a preference to the training mini-batches that give the model better generalization capabilities.

The third methodology studied the impact of giving the model the pose information of the actors. Since the keypoints do not contain much information regarding the actor, we wanted to test if this was a viable strategy to reduce actor bias. Results show some improvements over the baseline on public datasets, and comparable results on the Hanau03 Vito and Sunnyvale datasets.

For future work, it would be interesting to explore the use of these methodologies together and see if there are any significant improvements. Since the use of these methodologies had results aligned or superior to our baseline model, we wonder if the combination of them could achieve even better results.

Part III

Person Re-Identification

Chapter 5

Optimizing Person Re-Identification using Generated Attention Masks

Foreword on Author Contributions

The results of this work has been disseminated in an article in a national conference:

[16] Capozzi, L., Pinto, J.R., Cardoso, J.S., Rebelo, A. (2021). Optimizing Person Re-Identification Using Generated Attention Masks. In: Tavares, J.M.R.S., Papa, J.P., González Hidalgo, M. (eds) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2021. Lecture Notes in Computer Science(), vol 12702. Springer, Cham.

5.1 Introduction

Person re-identification consists of matching a query image of an individual to other pictures of that same individual. The pictures are taken from a system of different cameras, each with a different view. Hence, this problem has important applications in security and surveillance systems. This task can be quite challenging, due to the differences in the position of the cameras, where the face of the person might not be visible, the possibility of occlusions, the large variation in poses, and differences in the background.

In the past, the problem of Re-ID was treated as a classification problem, and a common strategy was to use handcrafted features [84], [85], [86], [87], [88], [89]. This was due to the small size of the datasets available in the past. With the rising interest in solutions for this problem, more complex datasets have been created. These new datasets include more identities and more pictures per identity. The approach to the problem has also changed with the use of Deep Learning, where instead of using handcrafted features, the networks encode the images by extracting relevant information into feature vectors that can be used to compare different images, using simple distance metrics [90], [91], [92], [93], [94].

The use of the same datasets, with the same train-test splits, allows to benchmark different methods, in order to compare their performance. Measures such as the Mean Average Precision (mAP) and the Cumulative Match Characteristic (CMC) curve are commonly used to compare the results.

One of the most commonly used loss functions is the Triplet Loss, which modifies the embedding space to pull together pictures belonging to the same person, and to push apart pictures of different identities [95]. This loss is also used with a batch mining process, which makes training more effective by choosing triplets where the error of the network is higher [96].

Many works also use attention mechanisms to force the network to focus on less distinctive areas, in order to improve performance in situations where important information may be missing [90], [91], [92], [97], [98].

In [90], the authors propose a method that randomly drops part of the input images in order to force the network to include less distinctive areas in the creation of the feature maps. This leads to a more robust network capable of performing better in situations where information may be missing or occlusions might occur.

In order to improve results, Quispe and Pedrini [91] proposed a method that instead of dropping random parts of the image, created an activation map. This allowed them to drop areas of the image with highly distinctive information, in order to force the network to perform better in situations where information was missing, by using less distinctive information. Although this method calculates which parts of the image it should drop, it can only drop horizontal stripes, which is limited and might lead to suboptimal results.

In this paper, we present a more flexible method, capable of calculating such masks using an auxiliary segmentation model, which allows it to define masks of any shape. This also removes the constraint of deriving the mask directly from the activations, which allows us to calculate the mask in a less predetermined manner.

5.2 Proposed Methodology

The proposed method is comprised of two networks. The feature extraction network uses a ResNet-50, or a ResNet-101 backbone [99], pretrained on ImageNet [100]. This network receives as input an image and outputs a vector containing 512 features. This feature vector can be used to compare different pictures by computing their cosine distance. After the network is trained, pictures containing the same individuals will have features vectors with a small distance and pictures with different individuals will have feature vectors with a large distance. These feature vectors allow us to sort a gallery of images. This is achieved by taking a query image and computing its distance to each image in the gallery.

This method also makes use of a mask generation network, to make the feature extraction network more robust. The mask generation network is trained to output a mask that highlights the parts of the image that the feature extraction network is using to generate the feature vector. To make the feature extraction network more robust we hide 30% of the image, by choosing the pixels

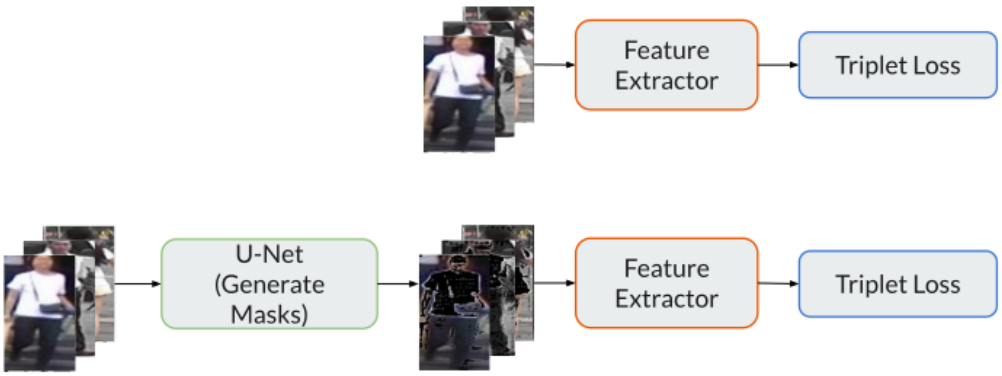

Figure 5.1: Training overview of the feature extraction model. Using the output from the U-Net we create a binary mask, setting 30% of the pixels with the highest activations to zero. The feature extractor receives the original images, and images with the masks applied. The triplet loss is calculated for each set, and a weighted sum of both losses gives the final loss of the feature extractor.

where the mask has the highest values. This forces the feature extraction network to extract relevant information from other parts of the image. We set the drop ratio to 30% as it was the recommended value in [90].

5.2.1 Network Architecture

5.2.1.1 Feature Extractor

The backbone is comprised of a ResNet-50 or ResNet-101 Network, pretrained on ImageNet. Using a pretrained backbone helps our model to more easily extract relevant features in an image, by using previously learnt filters. The last layer of the ResNet networks was removed, therefore the output from the backbone is a feature vector of shape 2048x1x1 (channels, height and width respectively). Then we added a convolutional layer with 512 filters, with a kernel size of 1×1 and stride 1×1, followed by Batch Normalisation (BN) and ReLU activation.

The output of the network is a template that can be used to match individuals. This template is a unidimensional vector containing 512 features, that describes the picture of the person that was passed to the network. The picture is matched against a gallery of pictures containing a variety of different individuals, by computing the cosine distance between the feature vectors and sorting the gallery based on the distance metric.

5.2.1.2 Mask Generator

The mask generation part of the model is performed by a U-Net [101], with an encoder that reduces the resolution at each level, and a decoder that increases the size back to its original resolution. The U-Net features skip connections between layers of the encoder and the decoder that have the same

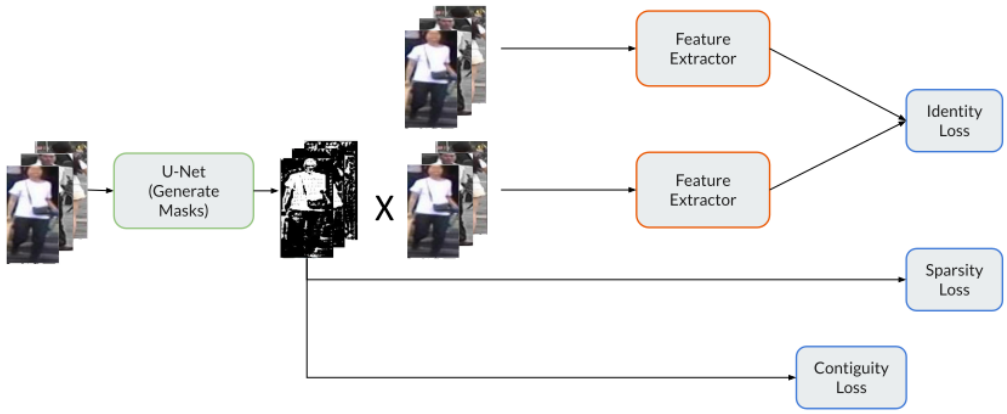

Figure 5.2: Training overview of the mask model. The identity loss promotes high mask activations to areas of the image relevant for the feature extractor. To avoid masks where all the values are equal to one, two more loss components are added. The goal of the sparsity loss is to force the mask model to hide parts of the image, and the contiguity loss promotes similar values for neighboring pixels.

resolution. This allows the transmission of information between layers, which avoids information loss during the encoding and decoding process.

The encoder is composed of 9 Convolutional layers and 4 Max-Pooling layers distributed over 5 resolution levels. Convolutional layers have $32-512$ filters, with a size of 3×3 and stride 1×1, each followed by a Batch Normalisation layer and a ReLU activation function. Max-Pooling layers use a window size of 2×2 and a stride of 2×2, which reduces the resolution by a factor of 2 at each level.

The decoder mirrors the architecture of the encoder with convolutional and deconvolutional layers. Convolutional layers have 32 to 512 filters, with a size of 3×3 and stride 1×1, each followed by Batch Normalisation and ReLU activation. Deconvolution Layers have $32-256$ filters with a size of 2×2 and stride 2×2, which increases resolution by a factor of 2.

The final layer is a Convolutional layer and has one filter with size 1×1 and stride 1×1, followed by a sigmoid activation function to have pixel values between 0 and 1 for the mask.

5.2.2 Loss

5.2.2.1 Feature Extractor

The feature extraction model uses the triplet loss, the goal of this loss function is to reduce the distance between feature vectors of pictures belonging to the same individual, and to increase the distance between feature vectors of pictures belonging to different individuals. In the experimental settings section, we go over the selection process of the triplets.

The triplet loss can be written as:

$$\mathscr{L}_{triplet}(F) = \mathbb{E}_{a,p,n}[max(||F(a)-F(p)||_2 - ||F(a)-F(n)||_2 + m, 0)]. \tag{5.1}$$

where F is the feature extractor, a is the anchor image, p is the positive image, n is the negative image and m is the margin.

When both models are being trained simultaneously we take a weighted sum of the triplet loss of the regular images and the triplet loss of the images with the mask applied.

$$\mathscr{L}_{extractor}(F) = \lambda_1 \mathscr{L}_{triplet}(F) + (1-\lambda_1)\mathscr{L}_{tripletwithmask}(F). \tag{5.2}$$

where $\mathscr{L}_{tripletwithmask}$ is the triplet loss applied to masked images (where some parts of the image has been removed to force the network to get relevant information from less distinctive areas).

5.2.2.2 Mask Model

In order to generate a mask that highlights the areas of the image that the feature extractor is using to calculate the feature vector, we use several losses combined.

The first component can be written as:

$$\mathscr{L}_{identity}(M) = \mathbb{E}_a[mse(F(a), F(M(a) \times a)]. \tag{5.3}$$

where mse is the mean squared error between two vectors, a is an image, M is the U-Net that generates the masks and F is the feature extractor. $M(a)$ is the mask generated for image a, and $M(a) \times a$ is the resulting image after applying the mask to image a.

The problem with this loss is that it will make the output of the mask model a mask filled with ones. This will make $M(a) \times a$ equal to a, minimizing the previous loss. This is not what we want, as our goal is to make the relevant pixels equal to one while making non-relevant pixels equal to zero. In order to achieve this we add a second loss component:

$$\mathscr{L}_{sparsity}(M) = \mathbb{E}_a[mean(M(a))]. \tag{5.4}$$

where $M(a)$ is the mask generated for image a. This loss function aims to reduce the mean value of the masks, in order to solve the previous problem.

In addition to both of these losses, we need to make the mask contiguous, in order to select an area of the image, and not single and separated pixels.

Hence, we add the third loss component:

$$\mathscr{L}_{contiguity}(M) = \mathbb{E}_a[\frac{1}{h \times w} \sum_{i,j} |M(a)_{i+1,j} - M(a)_{i,j}| + |M(a)_{i,j+1} - M(a)_{i,j}|]. \tag{5.5}$$

where $M(a)_{i,j}$ is the value of the mask at index (i, j). This loss calculates the difference of consecutive pixels of the mask (vertically and horizontally), then it calculates the absolute value and finally the mean.

The final loss function becomes:

$$\mathscr{L}_{mask}(M) = \lambda_2 \mathscr{L}_{identity}(M) + (1-\lambda_2)(\mathscr{L}_{sparsity}(M) + \mathscr{L}_{contiguity}(M)), \tag{5.6}$$

where $\mathscr{L}_{sparsity}$ and $\mathscr{L}_{contiguity}$ losses are adapted from [102].

5.3 Experimental Settings

5.3.1 Data

The proposed model was trained on three datasets that are commonly used in the person re-identification problem.

Market1501 dataset [103] contains data collected from six cameras, on an open environment, in Tsinghua University. It contains images of 1501 individuals. 751 individuals are used for training and 750 individuals for testing. There are a total of 12936 images in the training set, 19732 images in the test set and 3368 query images.

DukeMTMC-reID dataset [104], [105] is a subset of the DukeMTMC dataset. The original dataset contains 85-minute videos of high-resolution captured from 8 different cameras. It contains images of 1404 individuals. 702 individuals are used for training and 702 individuals for testing. This dataset contains 16522 images in the training set, 17661 images in the testing set and 2228 query images.

CUHK03 dataset [106] is comprised of images collected from The Chinese University of Hong Kong (CUHK) campus. It contains images from 1467 identities collected from 5 different pairs of camera views. This dataset contains 7368 images for training, 5328 images for testing and 1400 query images. This dataset has two versions: labelled and detected. Labelled means that the bounding boxes were labelled by a human. Detected means that the bounding boxes were estimated by a pedestrian detector. This dataset is prone to missing body parts, misalignments and occlusions, especially on the "detected" version, which makes it more challenging.

5.3.2 Data Augmentation

There are a couple of pre-processing steps that are applied to the images during training. These improve training, as they make the model more robust to changes and avoid overfitting. During training, images are resized to a resolution of 234×117 pixels (height and width, respectively). Then a random section of the image is cropped, with a size of 224×112 pixels. The image is then randomly flipped horizontally, with a probability of 0.5. After all these steps are applied we normalize the images using the mean and standard deviation from ImageNet. This is needed because we use a pretrained backbone, which was trained with normalized images.

5.3.3 Training

The proposed model was trained in three stages. The first stage consists of training the feature extraction model using $\mathscr{L}_{triplet}$. The second stage consists of training the mask model (while keeping the feature extraction model frozen) using $\mathscr{L}_{mask}$. The third stage consists of training both models simultaneously using $\mathscr{L}_{extractor}$ for the feature extractor model and $\mathscr{L}_{mask}$ for the mask generation model. After many experiments we set $\lambda_1 = 0.90$ and $\lambda_2 = 0.95$.

Since we are using the triplet loss we need a strategy to select the triplets that maximize the error of the network. Choosing triplets randomly is not a viable strategy, as the network can easily adapt itself to most pictures. This would result in a loss of progress, as the majority of the selected triplets would already have a difference in distances larger than the margin. To overcome this issue we used batch hard triplet mining [96].

Due to GPU memory constraints we used a batch size of 60 pictures (20 individuals and 3 pictures per individual).

We train the model for 50 epochs on the first stage, 25 epochs on the second stage and 100 epochs on the third stage.

For the feature extraction model we used the Adam optimizer with a learning rate of 2×10^{-4} as the default learning rate of 1×10^{-3} would make the pretrained ResNet-50 model unstable and led to a collapsed model, where every feature vector output by the model had the same values. The learning rate of 2×10^{-4} worked well for every dataset tested.

For the mask model we used the Adam optimizer with the default values.

5.4 Results and Discussion

To evaluate the performance of the model we used the mAP and the rank-1 accuracy. We computed these values for the test sets of the Market1501, DukeMTMC-reID dataset and CUHK03 dataset. There are two versions of the CUHK03 dataset, labelled (L) and detected (D).

Results show that the backbone has a considerable impact on the results, with the ResNet-101 achieving better results than the ResNet-50 across all datasets.

Our proposed methodology achieved competitive results to state-of-the-art methodologies on the rank-1 accuracy and the mAP score. The use of re-ranking [107] proved to be beneficial and improved the results among all test cases. Even though our method surpassed other approaches in some datasets when using re-ranking, it might not be a fair comparison since many of the methodologies did not publish their results using re-ranking. In order to further improve our approach we could add multiple streams (or networks), as in [90], [91]. This could make the training more stable, as we noticed some instability when training the mask model and the feature extractor simultaneously.

Figure 5.3 shows the masks that were generated by our mask generation model. We select 30% of the pixels with the highest values and hide them, therefore black areas represent zones of high importance to our feature extraction model, while white areas represent zones of low importance. During training, we hide the high importance zones, in order to force our feature extraction model to use other parts of the image to generate the feature vectors. In figure 5.3 we can see that the mask generation model correctly hides the person's body. Since we are limiting the area of the image that is hidden to 30% it generally masks the upper body, as this seems to be the most discriminative part.

Table 5.1: Comparison to state-of-the-art approaches. RK stands for re-ranking [107]. Note: Best values highlighted in bold.

Methods	Market-1501		DukeMTMC-reID		CUHK03 (L)		CUHK03 (D)	
	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
IDE [108]	72.5	46.0	67.7	47.1	22.2	21.0	21.3	19.7
PAN [109]	82.8	63.4	71.6	51.5	36.9	35.0	36.3	34.0
DPFL [110]	88.9	73.1	79.2	60.0	43.0	40.5	40.7	37.0
HA-CNN [97]	91.2	75.7	80.5	63.8	44.4	41.0	41.7	38.6
PyrNet [111]	95.2	86.7	87.1	74.0	71.6	68.3	68.0	63.8
Auto-ReID [112]	94.5	85.1	-	-	77.9	73.0	73.3	69.3
MGN [113]	**95.7**	86.9	88.7	78.4	68.0	67.4	66.8	66.0
DenSem [114]	**95.7**	87.6	86.2	74.3	78.9	75.2	78.2	73.1
MHN [115]	95.1	85.0	89.1	77.2	77.2	72.4	71.7	65.4
ABDnet [116]	95.6	88.2	89.0	78.5	-	-	-	-
SONA [117]	95.6	**88.6**	**89.2**	78.0	**81.8**	**79.2**	**79.1**	**76.3**
OSNet [118]	94.8	84.9	88.6	73.5	-	-	72.3	67.8
Pyramid [119]	**95.7**	88.2	89.0	**79.0**	78.9	76.9	78.9	74.8
Top-DB-Net [91]	94.9	85.8	87.5	73.5	79.4	75.4	77.3	73.2
ResNet-50								
Proposed	92.5	80.8	83.2	69.1	62.2	62.0	60.4	58.5
Proposed + RK	93.2	90.0	86.4	82.4	70.4	73.9	69.0	71.6
ResNet-101								
Proposed	92.8	82.2	84.6	70.9	65.9	64.7	63.5	61.3
Proposed + RK	93.6	90.4	86.8	82.5	72.1	75.7	70.9	73.8

5.5 Conclusion

The proposed model generates masks that represent areas of the image with important information for the identification process. In order to improve the performance of our feature extraction algorithm, we train it with original images, and images with missing information, which forces the model to use less distinctive areas to compute the feature vector.

Upon evaluation on three popular datasets, we show that the performance of the algorithm is comparable to the alternatives.

The process of removing information from the images could add noise and have the opposite effect that we want to achieve, making the training process unstable and decreasing the performance of the model. We tried to balance the losses to avoid this, but further efforts should be devoted to improving the architecture of the model, adding different streams to minimize the negative effects of removing information [90], [91].

Figure 5.3: Masks generated by our method (the hidden parts are areas of high importance).

Chapter 6

End-to-end Occluded Person Re-Identification with Artificial Occlusion Generation

Foreword on Author Contributions

The results of this work has been disseminated in an article in an international journal:

[11] L. Capozzi, J. S. Cardoso and A. Rebelo, "End-to-End Occluded Person Re-Identification With Artificial Occlusion Generation," in IEEE Access, vol. 13, pp. 146020-146031, 2025, doi: 10.1109/ACCESS.2025.3600385

6.1 Introduction

Person re-identification consists of matching images of individuals from different cameras, viewpoints, and locations. It is one of the most important computer vision problems and has several real-world applications, such as video surveillance, security, activity analysis and autonomous vehicles [85], [88], [89], [120], [121], [122]. In recent years, deep learning based methodologies have achieved remarkable results in Re-ID. This is due to the creation of large-scale Re-ID datasets, which allow to train deep learning based models, and also serve as a benchmark to compare different methodologies [96], [123], [124], [125], [126], [127], [128]. Although these methodologies achieve great results, they assume that the whole body of the individual is available for feature extraction. However, in real-world applications, parts of the individual's body are frequently hidden by different objects or obstacles. Therefore, the development of methodologies that take the problem of occlusions into account is of utmost importance.

Dai et al. [90] propose a methodology that improves the robustness of the model by randomly dropping parts of the feature map during training. When the dropped area contains a highly discriminative region, the network is forced to use less distinctive parts of the input image, improving the results in occluded scenarios. Quispe et al. [129] propose a similar approach, but instead of

dropping random parts, an activation map is created. This grants the ability to identify which parts of the input image contain the most valuable information, in order to ensure that the dropped part of the feature map contains important information. Similarly to the work of Dai et al. [90], this forces the network to use less distinctive information, improving the results when the input image contains occlusions. A limitation associated with these methodologies is that the dropped part of the feature map is always a horizontal stripe or a rectangular shape, which could have an impact on the results since in real-world scenarios occlusions can have many shapes and sizes.

Methodologies that use extra cues, such as pose information [130], [131], [132], [133] or human parsing [134], [135], seem to give the best results in the occluded person re-identification scenario. In most cases, an auxiliary external model extracts the body information, and then that information is used during the training of the model. However, these methodologies assume that the extracted body information is error-free, which is not the case in many instances, and this can degrade the performance of the model since the training data is noisy. If the auxiliary model is needed during inference there is also the problem of the big computational demands, which can pose some issues if the model is going to be used in a real-time scenario [136].

Other methodologies augment the data during training by adding generated artificial occlusions. These artificial occlusions help to make the model more robust by exposing the model to more realistic, and controlled scenarios during training. Wang et al. [136] create two occlusion sets by manually cropping backgrounds and occlusion objects from images in the training set. The images are manually cropped to avoid having body parts in the occlusion set. One of the downsides of this approach is that the occlusion set is dataset-specific, which can pose some limitations if we need to train the model on a different dataset. Additionally, there is the downside of having to manually crop the occlusion set. Chen et al. [137] generate occlusions by randomly selecting a training image and cropping a rectangular patch. In order to avoid cropping human bodies they crop the rectangular patch from the corners of the training image. The patch is then scaled and placed onto four locations of the input image. In this methodology, the occlusions are generated without manual annotation, however, the shape of the occlusions is always rectangular, and due to the scaling, the occlusions lack details, which limits the realistic aspect of this approach.

In order to deal with the issues presented above, we propose a jointly trained end-to-end model for occluded person re-identification, with an architecture that features three branches: the global branch, the feature dropping branch, and the occlusion detection branch. During training, we augment the data by adding two different types of artificial occlusions: Random Shape Occlusions and Random Object Occlusions, which ensures that our model is prepared to deal with missing information. Random Shape Occlusions consists of generating 6 pseudo-random points which are joined using Bézier curves, and filled with a random colour. In order to generate Random Object Occlusions, we create a small occlusion image dataset from publicly available images, which allows us to generate artificial occlusions during training, by randomly selecting an occlusion image and placing it on the input image, making the model robust towards occlusions and allowing it to later detect real occlusions.

The main contributions of this work are the following:

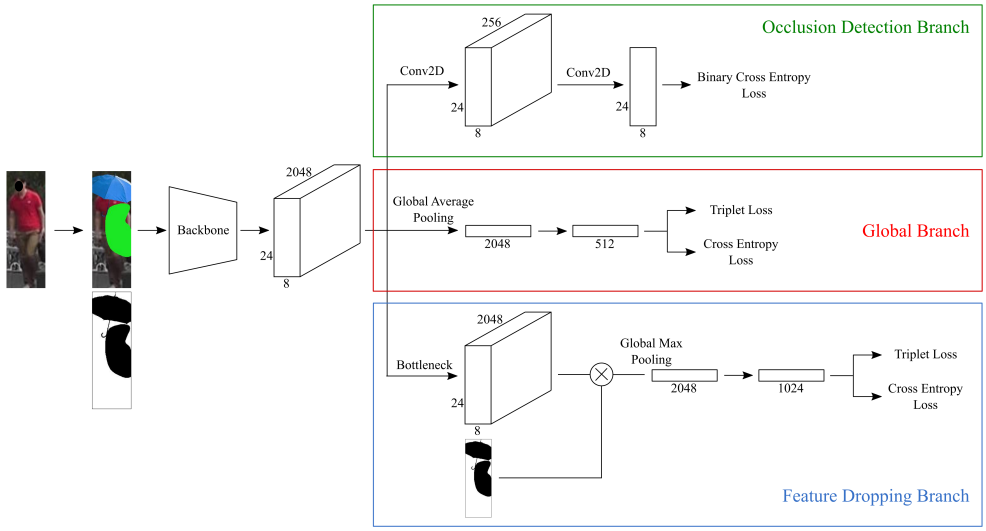

Figure 6.1: Overview of the network architecture. The method starts by applying an artificial occlusion to the input image, which is passed to a pre-trained backbone. The feature map returned by the backbone is passed to three different branches: the occlusion detection branch, the global branch, and the feature dropping branch. The occlusion detection branch outputs a mask that indicates which part of the input image is occluded. The global branch and the feature dropping branch output a descriptor of the input image. During training, the feature dropping branch applies the ground truth occlusion mask to the feature map. During testing, we use the mask output by the occlusion detection branch. To get the final descriptor of a pedestrian image in the test phase, we concatenate the features from the global branch and the feature dropping branch. The feature map sizes shown are for the ResNet-101 and ResNet-50 backbones. For details on other backbones, refer to section 6.3.

1. The development of an end-to-end methodology that does not need to rely on external methods, such as pose estimation algorithms;
2. Experiments on how different backbones affect the performance of the model;
3. The creation of a small dataset, using publicly available images, that is used to generate artificial occlusions during training, which eliminates the need to use occlusions that are dataset-specific;
4. The generation of occlusions using pseudo-random shapes, which helps improve the robustness of the model;
5. The addition of an occlusion detection branch in the architecture of our methodology, which is able to detect occlusions during inference.

6.2 Related Work

In this section, we briefly review methods related to holistic Re-ID and occluded Re-ID.

6.2.1 Holistic Person Re-Identification

Person re-identification consists of matching images of a person obtained from different camera views. Re-ID methods can be summarized into three different categories: hand-crafted descriptors [85], [88], [138], metric learning methods [89], [123], [124], [139], [140], [141] and deep learning methods [97], [142], [143], [144], [145], [146], [147], [148], [149], [150], [151], [152]. Currently, deep learning methodologies are dominant in the field of Re-ID. This is due to advances in GPU technology, and the availability of large-scale datasets that can be used to train the models. Some methods separate the input image or features into different sections in order to try to learn local information, and to achieve finer-grained feature matching by ensuring that the model is looking at a specific part of the image [143], [146], [153], [154]. Attention mechanisms have also been used, in order to ensure that the model focuses on the areas of the image where the person is [97], [151], [155]. However, these methods assume that the whole body is visible and do not take into account the presence of occlusions, which are very common in a realistic scenario, and this leads to a decreased performance.

6.2.2 Occluded Person Re-Identification

The study of occluded person re-identification is first explored by Zhuo et al. [156]. In occluded Re-ID, the training set and the gallery consists of holistic pedestrian images, and the query set is composed of occluded images, which makes this task much more challenging than holistic Re-ID due to the missing information. Wang et al. [136] propose a Feature Erasing and Diffusion Network to handle the challenges from occlusions. They also generate two occlusion sets by manually cropping occlusion objects and backgrounds from training images. Yang et al. [157] introduce a methodology that is robust to sparse and noisy pose information. The pose information is used to annotate the visibility of body parts, in order to suppress the use of features from occluded parts of the image. Chen et al. [137] propose an Occlusion Aware Mask Network, with an attention-guided mask module which requires occlusion labels. They generate occlusions by selecting a training image at random and cropping a rectangular patch from the corners of the image. The patch is then placed onto four locations of the input image. This produces precisely labelled occlusion images that can be used during training. Yan et al. [158] introduce a Re-ID methodology that learns discriminative single-scale global-level pedestrian features, by enforcing a new distance loss. They simulate occlusions by erasing parts of the input images. Li et al. [122] propose an end-to-end Part-Aware Transformer, with a transformer encoder-decoder architecture, which can detect highly discriminative parts of the image even when occlusions are present. Somers et al. [4] propose a body part-based ReID model that consists of two modules. The first module needs ground truth human parsing labels during training, which are obtained using a pose estimation model. The goal of this module is to predict body part attention maps. The second module produces the holistic and body part-based features which are used during inference.

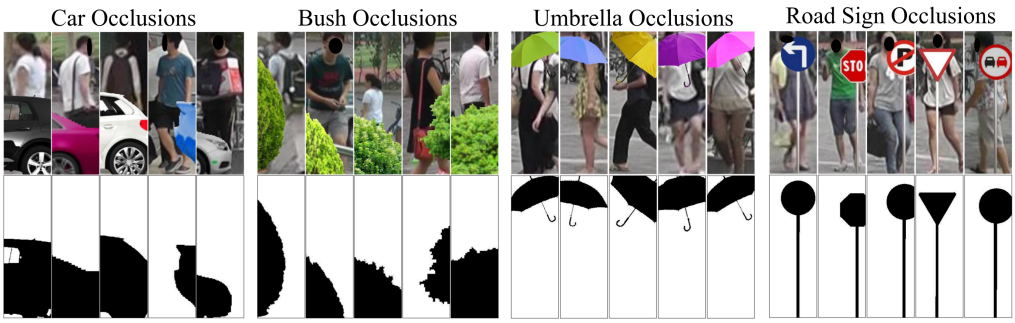

Figure 6.2: Examples of different types of object occlusions applied to training images from Market-1501 [103]. The bottom images are the occlusion masks.

6.3 Method

In this section, we present the proposed methodology in detail. The network architecture is illustrated in Fig. 6.1. The method starts by applying an artificial occlusion to the input image, which can either be a random shape occlusion, a random object occlusion or a combination of both. Then the image is passed to the backbone network. The feature map returned by the backbone is passed to three different branches: the occlusion detection branch, the global branch, and the feature dropping branch. The occlusion detection branch outputs a mask that indicates which part of the input image is occluded. The global branch and the feature dropping branch output a descriptor of the input image. During training, the feature dropping branch applies the ground truth occlusion mask to the feature map. During testing, we use the mask output by the occlusion detection branch. To get the final descriptor of a pedestrian image in the test phase, we concatenate the features from the global branch and the feature dropping branch.

6.3.1 Backbone network

We experiment with four different backbones, a ResNet-101 [99], a ResNet-50 [99], a MobileNetV3Large [159], and a VisionTransformerB16 [160], all pre-trained on ImageNet. We modify the ResNet networks slightly by not employing the down-sampling operation at the beginning of stage 4, which gives a larger feature map, of size $2048 \times 24 \times 8$, similarly to [90] and [129]. For the MobileNetV3Large and the VisionTransformerB16, we keep the original output size of $960 \times 12 \times 4$ and $768 \times 14 \times 14$, respectively.

6.3.2 Occlusion Detection Branch

The Occlusion Detection Branch receives the feature map returned by the backbone and applies a convolution layer, followed by batch normalisation and a ReLU activation function, reducing the feature map to a size of $256 \times 24 \times 8$ for the ResNet backbones, $256 \times 12 \times 4$ for the MobileNetV3Large, and $256 \times 14 \times 14$ for the VisionTransformerB16. Then, a second convolution

layer is applied, which returns the occlusion mask of size $1 \times 24 \times 8$ for the ResNet backbones, $1 \times 12 \times 4$ for the MobileNetV3Large, and $1 \times 14 \times 14$ for the VisionTransformerB16.

The goal of this part of the network is to identify which parts of the input image contain occlusions. During the testing stage, the output from this branch is given to the feature dropping branch to promote the use of relevant parts of the image.

6.3.3 Global Branch

The global branch starts by applying an average pooling operation on the feature map returned by the backbone. This gives a feature vector with a size of 2048 for the ResNet backbones, 960 for the MobileNetV3Large, and 768 for the VisionTransformerB16. This feature vector is further reduced to a dimension of 512 using a convolution layer, followed by batch normalisation and a ReLU activation function.

The global branch is used to extract features directly from the backbone. Global branches have already been successfully used to supervise the training of other branches [90], [113], [129], [143], [161].

6.3.4 Feature Dropping Branch

The feature dropping branch receives the feature map returned by the backbone and applies a ResNet Bottleneck Layer [99]. If the network is training, the ground truth occlusion mask is applied to the output of the Bottleneck Layer. If the network is in inference mode, the occlusion mask output by the occlusion detection branch is used. After applying the occlusion mask on the feature map, its dimensions are reduced using a max pooling layer, which returns a feature vector of size 2048 for the ResNet backbones, 960 for the MobileNetV3Large, and 768 for the VisionTransformerB16. The dimensions of the feature vector are once again reduced using a convolution layer, a batch normalisation layer and a ReLU activation function. The final size of the descriptor is 1024.

The objective of the feature dropping branch is to force the network to use less discriminative regions of the input image. When the artificial occlusions hide a highly discriminative area, the max pooling operation forces the network to use regions with weaker features, which in turn makes those features stronger [90]. During inference, the occlusion mask calculated by the Occlusion Detection Branch is applied to promote the network to use regions of the feature map that contain information about the individual.

6.3.5 Loss Functions

In order to optimise our model, we employ three different loss functions: the Binary Cross Entropy (BCE) Loss, the Cross Entropy (CE) Loss, and the Triplet Loss.

The binary cross entropy loss can be formulated as:

$$\mathscr{L}_{BCE} = BCE(\hat{y}, y), \tag{6.1}$$

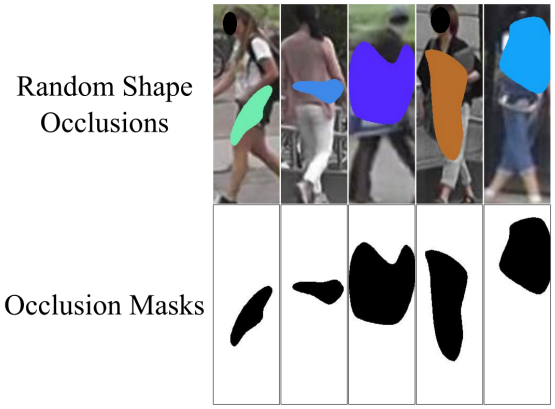

Figure 6.3: Examples of random shape occlusions applied to training images from Market-1501 [103].

where *BCE* represents the binary cross entropy loss, $\hat{y}$ is the prediction and y is the ground truth label.

The cross entropy loss can be written as:

$$\mathscr{L}_{CE} = CE(\hat{y}, y), \tag{6.2}$$

where *CE* is the cross entropy loss, $\hat{y}$ is the prediction and y is the ground truth label.

The triplet loss can be formulated as:

$$\mathscr{L}_{Triplet} = max(D(f_a, f_p) - D(f_a, f_n) + m, 0), \tag{6.3}$$

where f_a, f_p and f_n are the features extracted from the anchor, positive and negative samples, respectively, D is the L2-norm, and m is the margin.

The final objective can be written as:

$$\mathscr{L}_{total} = \mathscr{L}_{oBCE} + \mathscr{L}_{gCE} + \mathscr{L}_{gTriplet} + \mathscr{L}_{fCE} + \mathscr{L}_{fTriplet}. \tag{6.4}$$

where $\mathscr{L}_{oBCE}$ is the loss of the occlusion detection branch, $\mathscr{L}_{gCE}$ and $\mathscr{L}_{gTriplet}$ are the losses from the global branch, and $\mathscr{L}_{fCE}$ and $\mathscr{L}_{fTriplet}$ are the losses from the feature dropping branch.

6.4 Artificial Occlusion Generation

In order to make the model more robust, our methodology uses two types of artificial occlusions, which we refer to as Random Shape Occlusions, and Random Object Occlusion. In this section, we present how each type of occlusion is generated.

6.4.1 Random Shape Occlusion

The first type of artificial occlusion that is proposed is the random shape occlusion. The goal of this type of occlusion is to hide random information in the input image, in order to force the model to identify the individual using limited information. We generate random shapes to be less prone to overfitting, but also because it is better at mimicking real-world occlusions, since many times the occlusions that pedestrians are subjected to can have many shapes. In order to introduce even more variability, the generated shape is filled with a random colour.

In order to generate a random shape, we start by generating n points with random values for x and y in the range $[0, 1]$. The next step is to sort the points in a counterclockwise direction, making sure that there is a minimum distance of $\frac{1}{n}$ between consecutive points, and starting again if the condition fails. We use a value of 6 for n. Finally, the space between consecutive points is filled with 20 intermediate points calculated using a Bézier curve. Since all the initial points are in the range of $[0, 1]$ we can change the dimensions of the generated shape by multiplying the x and y coordinates by the desired values. Fig. 6.3 shows examples of random shape occlusions applied to training images from Market-1501 [103].

6.4.2 Random Object Occlusion

The second type of artificial occlusion that is proposed is the random object occlusion. The objects that are used belong to four different classes: cars, bushes, umbrellas and road signs. These classes were chosen because all of these objects are commonly found in public places, and they occlude the images in a way that is balanced, i.e., cars and bushes tend to hide the lower body, and umbrellas and road signs tend to hide the upper body.

We created a small dataset with publicly available images from these classes. The dataset contains 8 car images, 4 bush images, 4 umbrella images, and 7 road sign images. All of the images were manually cropped, in order to make the occlusion masks as accurate as possible. To avoid overfitting, we apply colour jitter, random horizontal flip, and choose the rotation, scale and position of the image at random with custom constraints for each class, in order to make the positioning of the objects as realistic as possible, and to avoid situations where the original image is excessively occluded. Fig. 6.2 shows examples of random object occlusions applied to training images from Market-1501 [103].

6.4.3 Random Shape Occlusion and Random Object Occlusion

Both of these types of artificial occlusions have the same goal, which is to force the network to learn to identify a pedestrian using limited information. The generation of random shapes is a flexible approach that provides a lot of variability, and the generation of occlusions using random objects is more realistic. Therefore, we decided to join both approaches so that they could complement each other. Notice that in cases where the two types of occlusions overlap, the random object is on top, in order to encourage the occlusion detection branch to detect real objects. Fig. 6.4 shows

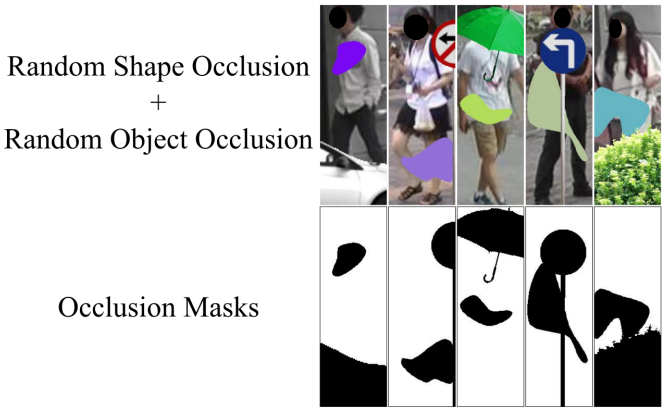

Figure 6.4: Examples of random shape occlusions and random object occlusions applied simultaneously to training images from Market-1501 [103].

examples of random shape occlusions and random object occlusions applied simultaneously to training images from Market-1501 [103].

6.5 Experiments

6.5.1 Datasets

We evaluate the performance of our methodology on two occluded Re-ID datasets: Occluded-Duke [131] and Occluded REID [156], two partial Re-ID datasets: Partial-REID [162] and Partial-iLIDS [163], and two holistic Re-ID datasets: Market-1501 [103] and DukeMTMC-reID [104].

Occluded-Duke [131] contains 15,618 training images, 17,661 gallery images, and 2,210 occluded query images. It was generated by selecting occluded images from DukeMTMC-reID [104]. All query images and 10% of gallery images are occluded. This means that when calculating the pairwise distance between query and gallery images at least one of the images is occluded.

Occluded-REID [156] is an occluded Re-ID dataset captured by mobile cameras. It contains 2,000 images belonging to 200 identities. Each identity has five full-body images and five occluded images with different types of occlusions from different viewpoints.

Partial-REID [162] is a partial Re-ID dataset that includes 600 images from 60 people. The gallery set contains 5 full-body images per person and the query set also contains 5 full-body images per person.

Partial-iLIDS [163] is a partial Re-ID dataset. It contains a total of 238 images from 119 people captured by multiple cameras in the airport, and some images contain people occluded by other individuals or luggage.

Market-1501 [103] is a holistic Re-ID dataset which consists of 1,501 identities captured by 6 cameras. The dataset is split into two parts: 750 identities are used for training and 751 identities

Table 6.1: Effect of the global branch (GLB), feature dropping branch (FDB), and occlusion detection branch (ODB) on the Occluded-Duke, Market-1501 and DukeMTMC-reID datasets. We use a pre-trained ResNet-101 [99] backbone, and apply both Random Shape Occlusions and Random Object Occlusions. Note: Best values highlighted in bold.

Methods	Occluded-Duke		Market-1501		DukeMTMC-reID	
	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
GLB	58.7	50.8	94.4	85.6	88.1	76.9
GLB + FDB	60.6	53.1	**94.9**	**87.4**	**88.6**	78.3
GLB + FDB + ODB	**60.8**	**53.2**	**94.9**	**87.4**	**88.6**	**78.4**

are used for testing. The training set contains 12,936 images, the query set contains 3,368 images, and the gallery set contains 19,732 images.

DukeMTMC-reID [104] is a holistic Re-ID dataset created from high-resolution videos from 8 different cameras. It is one of the largest pedestrian datasets. The images were cropped using hand-drawn bounding boxes. The training set contains 16,522 images, the query set contains 2,228 images and the gallery set contains 17,661 images.

6.5.2 Implementation Details

During training, we resize the input images to 384×128 [90], except for the VisionTransformerB16, which only accepts inputs of size 224×224, and augment them by applying random horizontal flip, random crop, random shape occlusion and random object occlusion. The images are normalised using the mean and standard deviation of ImageNet, since we are using pre-trained backbones.

Since the Occluded-REID, Partial-REID and Partial-iLIDS datasets do not contain a training set, we train the model using the Market-1501 [103] dataset and use colour jitter when applying data augmentation. This is how other occluded Re-ID methods tested their performance on these datasets, therefore, we do the same to compare the results.

As proposed by [96] we apply test-time augmentations by using different crops and flipping the input image. The final descriptor of an input image is the average of all the descriptors from different crops. We use the cosine distance to measure the similarity between different descriptors during testing.

Our network is trained using a batch size of 64 images. In each batch, there are 16 different individuals and 4 images per individual. A value of 0.1 is used for the label smoothing of the cross entropy loss. We use the BNNeck strategy proposed in [126]. When using the triplet loss, we adopt a batch-hard mining strategy, which means that for each anchor, we select the hardest positive and the hardest negative within that batch.

The networks that are used as backbones are pre-trained on ImageNet. We use the Adam optimiser with the learning rate initialised to $3.5e-5$, which is linearly increased to $3.5e-4$ over the first 30 epochs, and then decreased to $3.5e-5$ and $3.5e-6$ after 120 epochs and 210 epochs respectively, similarly to [126]. The training procedure is 360 epochs.

Table 6.2: Comparison with state-of-the-art methods on the Market-1501 [103] and DukeMTMC-reID [104] datasets. RK stands for re-ranking [107]. Note: Best values highlighted in bold, excluding RK results.

Methods	Market-1501		DukeMTMC-reID	
	Rank-1	mAP	Rank-1	mAP
DSA-reID [114]	95.7	87.6	86.2	74.3
BDB [90]	93.5	82.8	86.8	71.5
OSNet [118]	94.8	84.9	88.6	73.9
ABD [116]	95.6	88.2	89.0	78.5
BOT [126]	94.5	85.9	86.4	76.4
VPM [164]	93.0	80.8	83.6	72.6
IANet [165]	94.4	83.1	87.1	73.4
CAMA [166]	94.7	84.5	85.8	72.9
MHN-6 [115]	95.1	85.0	89.1	77.2
PGFA [131]	91.2	76.8	82.6	65.5
FPR [135]	95.4	86.6	88.6	78.4
HOReID [132]	94.2	84.9	86.9	75.6
CASN+PCB [167]	94.4	82.8	87.7	73.7
PVPM [130]	93.1	82.3	84.9	71.8
FED [136]	95.0	86.3	89.4	78.0
SGR [158]	**96.1**	**89.3**	**91.1**	**81.3**
PAT [122]	95.4	88.0	88.8	78.2
OAMN [137]	93.2	79.8	86.3	72.6
BPBreID [4]	95.1	87.0	89.6	78.3
ResNet-101				
Random Shape	**95.2**	**88.6**	**89.0**	**79.2**
Random Object	94.3	85.2	86.6	75.0
Both	94.9	87.4	88.6	78.4
Random Shape + RK	**95.6**	94.4	**91.4**	**89.6**
Random Object + RK	95.5	93.6	89.3	86.6
Both + RK	**95.6**	**94.6**	**91.4**	**89.6**
ResNet-50				
Random Shape	94.4	**86.8**	**87.6**	**77.2**
Random Object	93.9	83.3	86.2	73.2
Both	**94.7**	85.7	87.5	77.1
Random Shape + RK	**95.8**	94.0	**91.1**	88.9
Random Object + RK	95.4	93.0	88.4	85.9
Both + RK	95.4	**94.2**	90.6	**89.0**
MobileNetV3Large				
Random Shape	51.3	44.0	59.8	55.9
Random Object	**59.4**	47.1	**72.1**	**65.2**
Both	58.8	**48.6**	68.5	62.3
Random Shape + RK	58.8	59.8	58.1	61.2
Random Object + RK	67.4	65.6	**71.2**	**73.4**
Both + RK	**69.1**	**68.6**	69.5	70.8
VisionTransformerB16				
Random Shape	44.8	37.5	47.4	45.5
Random Object	**53.2**	39.6	**57.8**	52.1
Both	51.2	**40.1**	57.6	**52.8**
Random Shape + RK	50.7	50.1	46.1	49.4
Random Object + RK	**60.1**	54.9	56.3	57.2
Both + RK	57.3	**56.4**	**57.4**	**59.4**

For the evaluation of our methodology, we adopt the metrics that are most common in the Re-ID literature, which are the CMC curves and the mAP metrics.

6.5.3 Architecture Ablation Study

To test the effect of the branches in our network, we tested three different configurations: using only the global branch; using the global branch and the feature dropping branch; and using the global branch, the feature dropping branch, and the occlusion detection branch. When testing the network without the occlusion detection branch, we did not apply any mask on the feature dropping branch. We do not use the Occluded-REID, Partial-REID and Partial-iLIDS datasets in this ablation study, since they do not contain a training set, and we wanted to avoid cross-dataset experiments, as the difference in the domain could have an impact on the results. We use a pre-trained ResNet-101 [99] backbone, and apply both Random Shape Occlusions and Random Object Occlusions.

The results of the ablation study are presented in Table 6.1. In all of the datasets, we can see an increase in performance when the feature dropping branch is added, as this branch forces the network to use less discriminative regions of the input image. When the occlusion detection branch is added, we notice a small increase in performance on the occluded dataset (Occluded-Duke), but the performance remains almost the same on the holistic datasets. Since the holistic datasets do not contain any occlusions, the network is unable to take advantage of the occlusion detection branch. We decided to keep the occlusion detection branch for all datasets since it does not negatively impact the results, and we believe that having the ability to detect occlusions during inference could be useful in some situations.

6.5.4 Comparison between Random Shape Occlusions and Random Erasing Augmentation

To test the effectiveness of our proposed Random Shape Occlusions, we compare our results to another commonly used data augmentation technique called Random Erasing Augmentation [126], [170], which consists of hiding part of the image with a rectangle of a random size (we used the size ranges proposed in [126]) and filling the rectangle with the mean of the pixels of the image. Figure 6.5 shows a comparison between Random Shape Occlusions and Random Erasing Augmentation.

We study the effect of the two different fill strategies, a mean fill, as used in [126], and a random fill, where the colour used to fill the occlusion is selected randomly in the RGB colour space.

The effect of the shape of the occlusions is also studied, by comparing the rectangles used in [126] with our proposed Random Shape Occlusions.

For these experiments, we use a pre-trained ResNet-101 [99] backbone.

We do not use the Occluded-REID, Partial-REID and Partial-iLIDS datasets in this study to avoid cross-dataset experiments, since they do not contain a training set.

The results are shown in Table 6.4. They seem to indicate that regardless of the fill method used, Random Shape Occlusions lead to a better performance than Random Erasing Augmentation when using our methodology. This could be caused by the fact that using a random shape instead

Table 6.3: Comparison with state-of-the-art methods on the Partial-REID [162] and Partial-iLIDS [163] datasets. RK stands for re-ranking [107]. Note: Best values highlighted in bold.

Methods	Partial-REID		Partial-iLIDS	
	Rank-1	mAP	Rank-1	mAP
IDE [99]	57.0	53.6	68.9	72.4
PCB [143]	66.3	63.8	46.8	40.2
DSR [141]	73.7	68.1	64.3	58.1
SFR [168]	56.9	-	63.9	-
FPR [169]	81.0	76.6	68.1	61.8
RGA-S [151]	62.0	59.1	75.6	78.6
PGFA [131]	69.0	61.5	71.4	74.7
PVPM+Aug [130]	78.3	72.3	-	-
HOReID [132]	85.3	-	72.6	-
OAMN [137]	86.0	**77.4**	77.3	**79.5**
LKWS [157]	85.7	-	**80.7**	-
PAT [122]	**88.0**	-	76.5	-
ResNet-101				
Random Shape	80.3	76.4	**72.3**	**80.1**
Random Object	**81.3**	**79.7**	70.6	78.8
Both	**81.3**	78.7	70.6	78.6
Random Shape + RK	80.0	77.9	70.6	78.5
Random Object + RK	**83.0**	**82.1**	**71.4**	**79.0**
Both + RK	82.7	80.9	69.8	78.5
ResNet-50				
Random Shape	81.3	77.0	**76.5**	**82.0**
Random Object	80.3	78.3	71.4	78.4
Both	**82.3**	**79.3**	74.8	81.0
Random Shape + RK	82.0	79.2	**73.1**	**79.7**
Random Object + RK	80.7	80.8	64.7	74.6
Both + RK	**83.3**	**81.9**	70.6	78.3
MobileNetV3				
Random Shape	**80.3**	75.6	**69.8**	**78.0**
Random Object	80.0	**76.4**	68.9	77.7
Both	77.3	74.9	65.6	75.3
Random Shape + RK	**81.7**	77.7	**71.4**	**78.9**
Random Object + RK	79.7	**78.7**	70.6	78.6
Both + RK	77.7	76.9	65.6	74.3
VisionTransformerB16				
Random Shape	67.7	65.5	50.4	61.2
Random Object	70.0	**67.0**	**53.8**	**63.1**
Both	**70.3**	66.0	47.9	60.6
Random Shape + RK	68.3	67.4	47.1	58.8
Random Object + RK	70.7	**69.3**	**51.3**	**60.9**
Both + RK	**71.7**	68.6	47.9	59.4

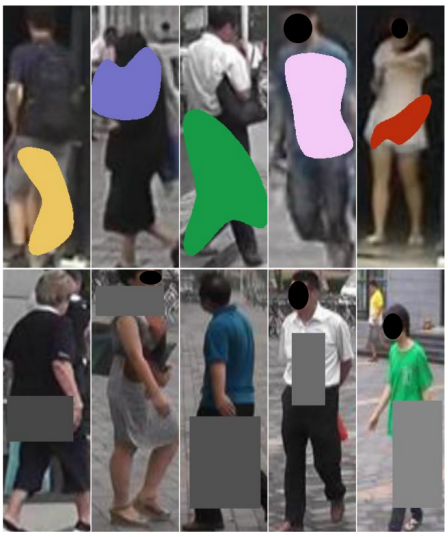

Figure 6.5: Examples of Random Shape Occlusions with a random colour fill (top) and Random Erasing Augmentation [126] with mean fill (bottom) to training images from Market-1501 [103].

of a rectangle forces the network to adapt to occlusions that are more similar to those found in the real world, where there is a lot of variability regarding their shape. When comparing different fill strategies, we find that the random fill seems to be superior to the mean fill, which could be due to less overfitting.

6.5.5 Computational Performance Analysis

To assess the efficiency and scalability of our architecture, we evaluate the computational performance of each module, based on three metrics: MFLOPS, number of parameters, and the size of the module in Megabyte (MB). The modules consist of the backbone, the global branch (GLB), the feature dropping branch (FDB), and the occlusion detection branch (ODB). Table 6.5 presents the characteristics of each of the modules.

MobileNet is the less computationally complex backbone, with 218.95 MFLOPS, 5.48 million parameters, and a size of 21.01 MB, making it well suited for resource-constrained environments. In contrast, Vision Transformer (ViT) has the highest computational complexity, with 11289.27 MFLOPS, 86.57 million parameters, and a size of 330.23 MB. ResNet-50 and ResNet-101 offer an intermediate level of complexity, making them well-suited for most applications.

The global branch (GLB), the feature dropping branch (FDB), and the occlusion detection branch (ODB) have a much lower computational overhead than the backbones. The global branch (GLB) has 1.05 MFLOPS, 395776 parameters, and a size of 1.52 MB. The feature dropping branch (FDB) has 2.10 MFLOPS, 791552 parameters, and a size of 3.04 MB. The occlusion detection branch (ODB) has 100.81 MFLOPS, 197633 parameters, and a size of 0.76 MB. We decided to keep the occlusion detection branch in our architecture since the computational overhead added by this module is small compared to the backbones. As shown in section 6.5.3, adding the occlusion

Table 6.4: Comparison between Random Erasing Augmentation [126] and Random Shape Occlusions. We use a pre-trained ResNet-101 [99] backbone. Note: Best values highlighted in bold.

	Methods	Occluded-Duke		Market-1501		DukeMTMC-reID	
		Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
Mean Fill	Random Erasing Augmentation	54.8	48.3	95.0	88.0	88.3	78.5
	Random Shape Occlusions	**56.2**	**49.7**	**95.3**	**88.4**	**88.5**	**79.0**
Random Fill	Random Erasing Augmentation	55.0	48.8	**95.2**	88.2	88.8	**79.2**
	Random Shape Occlusions	**56.7**	**50.0**	**95.2**	**88.6**	**89.0**	**79.2**

detection branch increases the accuracy in some cases, and it allows our architecture to generate occlusion masks.

6.5.6 Results

For each of the datasets, we trained the model using three different artificial occlusion generation strategies. We train the model using only random shape occlusions, using only random object occlusions, and also combining both approaches. We use four different backbones, a ResNet-101 [99], a ResNet-50 [99], a MobileNetV3Large [159], and a VisionTransformerB16 [160], all pre-trained on ImageNet.

The results on the Market-1501 and DukeMTMC-reID datasets are presented in Table 6.2. Similarly to other works, we compare our results with other Re-ID methodologies that deal with occlusions. Results on the Market-1501 and DukeMTMC-reID datasets indicate that the best data augmentation technique for holistic datasets (without re-ranking) seems to be random shape occlusions, achieving an accuracy of 95.2% / 88.6% Rank-1/mAP on the Market-1501 dataset using a ResNet-101 backbone, 94.4% / 86.8% Rank-1/mAP using a ResNet-50 backbone, 58.8% / 48.6% Rank-1/mAP using a MobileNetV3Large backbone and 51.2% / 40.1% Rank-1/mAP using a VisionTransformerB16 backbone. We achieve an accuracy of 89.0% / 79.2% Rank-1/mAP on the DukeMTMC-reID dataset using a ResNet-101 backbone, 87.6% / 77.2% Rank-1/mAP using a ResNet-50 backbone, 72.1% / 65.2% Rank-1/mAP using a MobileNetV3Large backbone and 57.6% / 52.8% Rank-1/mAP using a VisionTransformerB16 backbone. When comparing the results to other state-of-the-art methodologies, we can see that our approach surpasses most methods on the mAP metric and achieves comparable results using the Rank-1 metric. Although the best results in holistic datasets were achieved using random shape occlusions, the type of occlusion that achieved the best results for each dataset is not consistent. We believe that this is because improvements in the occluded scenarios are not as noticeable due to the lack of occlusions in holistic datasets.

The results on the Partial-REID and Partial-iLIDS datasets are presented in Table 6.3. Results on the Partial-REID and Partial-iLIDS datasets indicate that the best data augmentation technique for partial datasets depends on the type of data used. Random object occlusions seem to give the best results for the Partial-REID dataset, achieving an accuracy of 81.3% / 79.7% Rank-1/mAP using a ResNet-101 backbone, 82.3% / 79.3% Rank-1/mAP using a ResNet-50 backbone, 80.0% / 76.4% Rank-1/mAP using a MobileNetV3Large backbone and 70.0% / 67.0% Rank-1/mAP using

Table 6.5: Million Floating Point Operations (MFLOPS), Nº of parameters and Size (MB) of the modules used in our architecture. It contains 4 backbones, and 3 branches (global branch, feature dropping branch, and occlusion detection branch).

	Modules	MFLOPS	Nº Parameters	Size (MB)
Backbones	ResNet-50	6104.78	23508032	89.88
	ResNet-101	9751.27	42500160	162.53
	MobileNet	218.95	5483032	21.01
	ViT	11289.27	86567656	330.23
Branches	GLB	1.05	395776	1.52
	FDB	2.10	791552	3.04
	ODB	100.81	197633	0.76

a VisionTransformerB16 backbone. Random shape occlusions seem to work better with the Partial-iLIDS dataset, achieving an accuracy of 72.3% / 80.1% Rank-1/mAP using a ResNet-101 backbone, 76.5% / 82.0% Rank-1/mAP using a ResNet-50 backbone, 69.8% / 78.0% Rank-1/mAP using a MobileNetV3Large backbone and 53.8% / 63.1% Rank-1/mAP using a VisionTransformerB16 backbone. This could be because the type of occlusions found on the Partial-iLIDS dataset are from a different domain than the occlusions used during data augmentation. The results on the partial datasets are superior to state-of-the-art methods using the mAP metric, and comparable to other approaches using the Rank-1 metric. The model that is used on the partial datasets is trained on the Market-1501 dataset, making our methodology suitable for cross-dataset scenarios, which is more similar to real-world deployment, where the inference data might be from a different domain than the training data.

The results on the Occluded-Duke and Occluded-REID datasets are presented in Table 6.6. Results on the Occluded-Duke and Occluded-REID datasets indicate that random objects do improve the results, and depending on the dataset, there can be an advantage in combining both random object occlusions and random shape occlusions. The accuracy on the Occluded-Duke dataset is 60.8% / 53.2% Rank-1/mAP using a ResNet-101 backbone, 58.4% / 50.2% Rank-1/mAP using a ResNet-50 backbone, 58.8% / 48.6% Rank-1/mAP using a MobileNetV3Large backbone and 53.2% / 39.6% Rank-1/mAP using a VisionTransformerB16 backbone, which is aligned with other methodologies. The accuracy on the Occluded-REID dataset is 72.7% / 67.0% Rank-1/mAP using a ResNet-101 backbone, 71.7% / 66.0% Rank-1/mAP using a ResNet-50 backbone, 72.1% / 65.2% Rank-1/mAP using a MobileNetV3Large backbone and 57.6% / 52.8% Rank-1/mAP using a VisionTransformerB16 backbone.

In these experiments, we notice that the backbone network has a big impact on the results. The ResNet-101 and ResNet-50 seem to give the best results, which might be due to the larger feature map of size $2048 \times 24 \times 8$. The size of the backbone network could also play a role, as larger networks such as the VisionTransformerB16 might be more prone to overfitting, and smaller networks such as the MobileNetV3Large might not have enough complexity to extract the relevant features.

For each of the datasets we also include the results using re-ranking [107]. When using this

Table 6.6: Comparison with state-of-the-art methods on the Occluded-Duke [131] and Occluded-REID [156] datasets. RK stands for re-ranking [107]. Note: Best values highlighted in bold.

Methods	Occluded-Duke		Occluded-REID	
	Rank-1	mAP	Rank-1	mAP
PCB [143]	42.6	33.7	41.3	38.9
AdO [171]	44.5	33.2	-	-
DSR [141]	40.8	30.4	72.8	62.8
SFR [168]	42.3	32	-	-
OSNet [118]	33.7	20.1	-	-
ASB [126]	61.1	48.7	67.5	62.5
PGFA [131]	51.4	37.3	-	-
PVPM+Aug [130]	-	-	70.4	61.2
PVPM [130]	47.0	37.7	70.4	61.2
GASM [172]	-	-	74.5	65.6
HOReID [132]	55.1	43.8	80.3	70.2
ISP [173]	62.8	52.3	-	-
FED [136]	68.1	56.4	**86.3**	**79.3**
LKWS [157]	62.2	46.3	81.0	71.0
OAMN [137]	62.6	46.1	-	-
SGR [158]	**69.0**	**57.2**	78.5	72.9
PAT [122]	64.5	53.6	81.6	72.1
BPBreID [4]	66.7	54.1	76.9	68.6
ResNet-101				
Random Shape	56.7	50.0	60.4	59.0
Random Object	**61.9**	50.4	**72.7**	**67.0**
Both	60.8	**53.2**	70.6	65.1
Random Shape + RK	61.9	64.3	58.9	63.8
Random Object + RK	68.5	67.5	71.2	73.6
Both + RK	**69.6**	**71.4**	**71.6**	**73.7**
ResNet-50				
Random Shape	51.1	44.4	63.1	58.8
Random Object	57.7	46.8	**71.7**	**66.0**
Both	**58.4**	**50.2**	71.5	**66.0**
Random Shape + RK	55.9	58.5	60.7	64.5
Random Object + RK	63.9	64.3	**74.5**	75.4
Both + RK	**68.1**	**69.1**	73.6	**76.0**
MobileNet				
Random Shape	51.3	44.0	59.8	55.9
Random Object	**59.4**	47.1	**72.1**	**65.2**
Both	58.8	**48.6**	68.5	62.3
Random Shape + RK	58.8	59.8	58.1	61.2
Random Object + RK	67.4	65.6	**71.2**	**73.4**
Both + RK	**69.1**	**68.6**	69.5	70.8
ViT				
Random Shape	44.8	37.5	47.4	45.5
Random Object	**53.2**	**39.6**	**57.8**	52.1
Both	51.2	40.1	57.6	**52.8**
Random Shape + RK	50.7	50.1	46.1	49.4
Random Object + RK	**60.1**	54.9	56.3	57.2
Both + RK	57.3	**56.4**	**57.4**	**59.4**

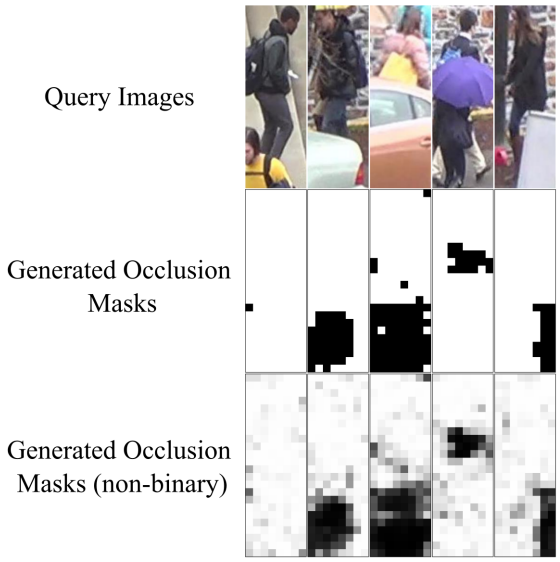

Figure 6.6: Occlusion masks returned by the occlusion detection branch, using query images from Occluded-Duke [131]. The model was trained using random shape occlusions and random object occlusions simultaneously.

methodology, we notice an improvement in the results for every dataset, except for the Partial-iLIDS [163] dataset, which could be caused by the small number of images.

Fig. 6.6 shows the output of the occlusion detection branch on query images from the Occluded-Duke dataset, using the model that was trained using both random shape occlusions and random object occlusions. If we analyse the figure, it is clear that the model is able to correctly detect occlusions, even though the images used to create the artificial occlusions were publicly available images, which were not related to the dataset. Additionally, the small size of the occlusion dataset is a testament to the generalisation capabilities of our method.

6.6 Conclusion

In this chapter, we propose an end-to-end methodology for occluded Re-ID that uses two types of artificial occlusion generation. Random shape occlusions help improve the robustness of the model, and random object occlusions provide a more realistic augmentation strategy, which proved to be effective on occluded datasets. Regarding holistic datasets, our method performed on par with state-of-the-art methodologies. The results on the partial datasets show similar performance to state-of-the-art methodologies on the Rank-1 metric and superior performance to state-of-the-art approaches on the mAP metric. Our method also had comparable results to the state-of-the-art on the occluded datasets. Overall, the results show that our methodology performs favourably against state-of-the-art methodologies. We also perform an ablation study comparing our proposed Random Shape Occlusions to the more commonly used Random Erasing Augmentation, to show the effectiveness of our approach.

Part IV

Pose Estimation and Action Recognition using Wi-Fi

Chapter 7

Deciphering the Silent Signals: Unveiling Frequency Importance for Wi-Fi-Based Human Pose Estimation with Explainability

Foreword on Author Contributions

The results of this work has been disseminated in an article in an international conference:

[10] Capozzi, L., Ferreira, L., Gonçalves, T., Rebelo, A., Cardoso, J.S., Sequeira, A.F. (2026). Deciphering the Silent Signals: Unveiling Frequency Importance for Wi-Fi-Based Human Pose Estimation with Explainability. In: Gonçalves, N., Oliveira, H.P., Sánchez, J.A. (eds) Pattern Recognition and Image Analysis. IbPRIA 2025. Lecture Notes in Computer Science, vol 15938. Springer, Cham.

7.1 Introduction

The investment and advancement of wireless technologies motivated several research studies on wireless signals, namely Wi-Fi, for indoor human activity detection in diverse contexts (e.g., smart healthcare or homes, security, and industry). Apart from having an established infrastructure in such contexts, Wi-Fi is non-invasive, has low cost, and provides ubiquitous coverage in the indoor setup [5]. Taking advantage of the deep learning paradigm in related tasks (e.g., human body parsing [174] and pose estimation [175]), the community started exploring the application of these models in human activity recognition based on Wi-Fi information [176]. However, the increase in data availability and computational motivated the development of (deep) machine learning methods with enhanced predictive power, but also more complex, with black-box behaviour. The field of xAI appeared as a potential solution to understand and audit such methods, hoping to bring some

light to what the community knows about their inner workings. Although the community often uses interpretability and explainability interchangeably, we focus on the definition of xAI as a three-stage process: pre-model (i.e., analysing the data before training any model), in-model (i.e., ensuring that the model will respect user-defined constraints or conditions, being interpretable by design), and post-model (i.e., using independent methods to assess an already trained model, without assuming we know its inner workings) [177]. In this paper, we build a transformer-based deep neural network (DNN) to estimate human pose from Wi-Fi signals. Specifically, our goal is to predict the key point coordinates of the human skeleton from the Wi-Fi CSI. Hoping to understand better how the model is functioning, we perform a systematic study in which we used the attention matrix of the transformer to select the sub-carriers the model is paying attention to (i.e., with selections of n high or low sub-carriers) and compare its performance against the baseline case (i.e., with all the sub-carriers) and the random (i.e., with a random selection of n sub-carriers).

7.2 Background and State of the Art

Wang et al. [178] proposed a deep learning method (Person-in-WiFi) that aims at fine-grained person perception by mapping Wi-Fi signal data to compute skeleton key points coordinates and the body segmentation mask. Guo et al. [179] created an in-house dataset (publicly available) and employed a custom-made CNN (Wi-Pose) to extract the skeleton key points from CSI. Jiang et al. [180] presented a method based on a recurrent neural network (RNN) to predict 3D human pose (skeletons composed of the joints on both limbs and torso of the human body) from Wi-Fi signals (WiPose). Yu et al. [181] designed a deep learning framework that combines image (i.e., an initial optical image) and Wi-Fi data to compute the body skeleton key points and generate human pose images. Geng et al. [182] developed a DNN that maps the phase and amplitude of Wi-Fi signals to UV coordinates within several human regions and reported competitive results (when comparing to image-based approaches). Ren et al. [183] proposed a deep 3D human pose estimation system (GoPose) that uses CNN and RNN to analyse and process Wi-Fi signal data. Zhou et al. [184], [185] developed multi-modal DNNs with an encoder-decoder architecture and a multi-head attention mechanism (PerUnet, CSI-former) to fuse pose features and Wi-Fi CSI data to estimate human pose, and released the Wi-Pose dataset, which consists of 5 GHz Wi-Fi CSI, along with the corresponding images (not included in the public version of the dataset because of personal privacy and portrait rights), and skeleton point annotations (generated with AlphaPhose [186], [187]). Recently, Yan et al. [188] extended the work of Wang et al. [178] and introduced the Person-in-WiFi 3D, an end-to-end transformer-based encoder-decoder model designed for estimating multi-person 3D poses using Wi-Fi signals, along with the release of a new publicly available large dataset. Gian et al. [189] aimed at lightweight 3D human pose estimation with a multi-branch CNN empowered by a selective kernel attention (SKA) mechanism that dynamically adjusts kernel sizes according to input data characteristics, promoting adaptability without increasing complexity.

Using xAI techniques to analyse 1D signals is still a recent topic. Pinto and Cardoso [190] were pioneers in this field of research: they used post-model approaches to verify (and visualise) if

the QRS complex was the most relevant component of the Electrocardiogram (ECG) for biometrics and found that, despite its importance, the QRS complex shared relevance with other heartbeat components, a finding that could be useful for model regularisation in future work directions. Interestingly, the biomedical community has been devoting special attention to the ECG regarding heart disease detection: Anand et al. [191] and Majhi and Kashyap [192] used SHapley Additive exPlanations (SHAP) [193] to assess the results and the functioning of their models. Chen and Lee [194] made a similar approach to the classification and analysis of vibration signals, in which they transformed the 1D signals into 2D images with a short-time Fourier transform, and used Gradient-Weighted Class Activation Mapping (Grad-CAM) to assess the relevant features by the model. Recently, we noticed that the community is starting to devote some attention to the potential of xAI in signal-processing applications. For instance, Elango and Landry [195] studied the application of post-model xAI techniques (e.g., SHAP) to assess the quality of Global Navigation Satellite Systems (GNSS) signals, and Gizzini et al. [196] used xAI to support channel estimation in wireless communication, designing a framework (XAI-CHEST) that aims to identify the relevant model inputs by inducing high noise on the irrelevant ones. Naturally, with the advent of Natural Language Processing (NLP) models and voice-guided personal assistants, these techniques have also played an interesting role in audio signal analysis [197].

7.3 Methodology

7.3.1 Data

In this work, we use the Wi-Pose dataset [184]. The data was captured using Wi-Fi devices and cameras, and the pose key point information was extracted by the creators using Alphapose [198]. This dataset includes 12 different actions performed by 12 volunteers. In the publicly available version, Zhou et al. [184] provide the Wi-Fi channel state information and the pose annotation extracted from the images (they do not share the images because of privacy constraints). The original dataset is composed of $166,600$ packets, which are then divided into $132,847$ packets for training and $33,753$ packets for testing. Each packet contains 2 tensors: a $30\times5\times3\times3$ CSI tensor, and a 3×18 pose annotation tensor. The CSI tensor has 30 sub-carriers, 5 time steps, 3 transmitting antennas and 3 receiving antennas. The pose tensor has 18 key points, and each key point contains the x and y coordinates, and the confidence value (x,y,c). Notice that for each pose annotation tensor there are 5 Wi-Fi time steps: the Wi-Fi sampling rate is 100Hz and the camera sampling rate is 20Hz, which gives 5 Wi-Fi samples for each camera frame. In this work, our goal is to estimate the position of each of the 18 pose key points. Following Zhou et al. [184], we calculate and use the skeleton-point adjacency matrix, which contains the information about the position of each key point, and the relative distance between each of them. The Skeleton-Point Adjacency Matrix (SAM) is calculated using the original 3×18 matrix $(x_i,y_i,c_i), i\in[1,18]$, and expanding the last dimension. The SAM consists of a $3\times18\times18$ matrix $(x_{i,j},y_{i,j},c_{i,j}), (i,j\in[1,18])$, and it is calculated according to Equation 7.1.

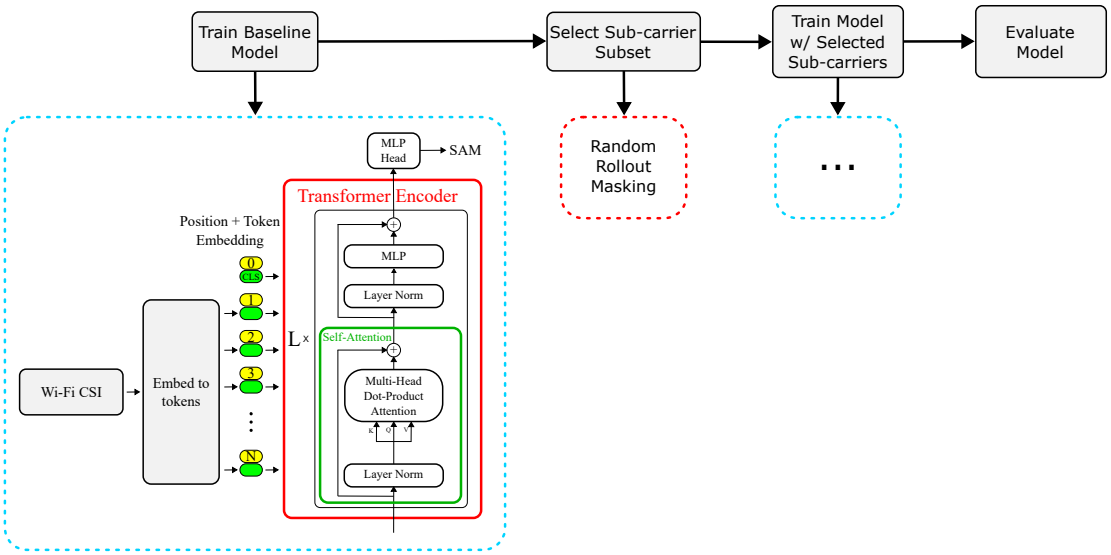

Figure 7.1: Pipeline of our proposal.

$$p'_{i,j} = \begin{cases} p_i - p_j, & \text{if } i \neq j \\ p_i, & \text{if } i = j \end{cases}, p \in [x, y, c] \tag{7.1}$$

7.3.2 Proposal

In this work, we tackle the task of pose estimation from Wi-Fi signals using a transformer-based architecture [199]. We encode each sub-carrier into tokens (an essential step to use this data as input to the transformer) and analyse which sub-carrier the model is paying more attention to. Based on the previous assessment, we sought to improve the computational performance of the model by using only the sub-carriers that contain the most relevant information (i.e., that have higher attention scores). We employ three different methods to select the sub-carriers: **random** selection, **rollout** attention-based selection [200], and **masked** attention-based selection (see Fig. 7.1).

7.3.2.1 Tokenisation of the Wi-Fi Signal

Transformer-based language models typically encode discrete words or characters (organised temporally) into tokens. Vision transformers [201], [202] encode images or videos into tokens by dividing them into patches. CSI samples contain spatial information, given by the transmitter and receiver antennas, and temporal information (CSI samples are time-series data). Hence, we start by reshaping the original signal of size $30 \times 5 \times 3 \times 3$ (i.e., #sub-carriers $\times$ #time $\times$ #transmitters $\times$ #receivers) into 30×45, which gives us 30 tokens with 45 features each. We pass the tokens through a linear layer, resulting in a shape of 30×128. We add a classification token, and a randomly initialised learnable positional embedding, resulting in an input tensor of size 31×128.

7.3.2.2 Architecture

Our architecture consists of 12 encoder layers that process the tokens generated with the CSI samples. Each encoder layer is composed of a multi-head self-attention module with 2 heads and a multilayer perceptron with a hidden dimension of 512. The multi-head self-attention module and the multilayer perceptron layers have skip connections. We select the classification token output by the encoder and pass it to a linear layer that gives the estimated skeleton-point adjacency matrix. The output of the network is a tensor of shape $2 \times 18 \times 18$, $(x_{i,j}, y_{i,j}), (i, j \in [1, 18])$.

7.3.2.3 Implementation Details

The network is trained using a batch size of 512. We use the Adam optimizer with the learning rate initialized to $5e-4$, which is decreased to $5e-5$ and $5e-6$ after 800 epochs and 1100 epochs respectively. The training procedure takes 1400 epochs.

7.3.2.4 Sub-carrier Selection

To understand which part of the CSI signal is more important, we perform a set of experiments in which we select a subset of the sub-carriers and check how the accuracy of the model is affected. We also check the computational performance increase from using fewer sub-carriers. Firstly, we train a baseline model using all available sub-carriers, and then use this model to select the sub-carriers.

- *Random*: This method consists of selecting a subset of 5, 3, or 1 sub-carriers at random. Since the selection is random we do not use the baseline model trained with all the sub-carriers.
- *Rollout*: The second method is the rollout attention method [200]. Starting from the baseline model, we leveraged the multi-head self-attention mechanism in the model architecture, we applied the rollout attention algorithm to get a proxy of the relevance of each sub-carrier using the attention weights. We ranked all sub-carriers by importance based on the principle that the most relevant sub-carriers have higher attention scores. Finally, we selected the top 5, 3, or 1 most relevant sub-carriers.
- *Masking*: The third and final selection method is the masking method. Starting from the baseline model, we leveraged the multi-head self-attention mechanism in the model architecture. We individually masked each sub-carrier by cancelling its attention coefficient within the attention matrix on the first Transformer Block and assessed its impact on model performance. We ranked all sub-carriers by importance based on the principle that the most relevant sub-carriers exert a higher negative impact when removed. Finally, we selected the top 5, 3, or 1 most relevant sub-carriers.

7.3.2.5 Loss Function

The loss function used in this work is similar to the one used by Zhou et al. [184]. It calculates the mean squared error between the estimated key points and the ground truth, and it gives a different weight to each key point using the Alphapose [198] confidence. It is given by Equation 7.2:

$$L = \frac{1}{2 \times N_i \times N_k} \sum_{i=1}^{N_i} \sum_{k=1}^{N_k} \left(G_c^{'i,k} \times \left(\left(S_x^{'i,k} - G_x^{'i,k} \right)^2 + \left(S_y^{'i,k} - G_y^{'i,k} \right)^2 \right) \right), \tag{7.2}$$

where N_i is the number of key points, with $N_i = 18$, N_k is the number of samples in the batch, $G_c^{'i,k}$ is the Alphapose [198] confidence of key point i belonging to sample k, $S_x^{'i,k}$ is the prediction for the x coordinate of key point i belonging to sample k, $G_x^{'i,k}$ is the ground truth for the x coordinate of key point i belonging to sample k, $S_y^{'i,k}$ is the prediction for the y coordinate of key point i belonging to sample k, $G_y^{'i,k}$ is the ground truth for the y coordinate of key point i belonging to sample k.

7.3.2.6 Testing Protocols

Similarly to other works [184], we test the accuracy of the predicted key points by calculating the percentage of correct key points ($PCK@a_j$). To calculate if a key point is correct we start by calculating the size of the torso of the person that the key point belongs to. This is given by Equation 7.3:

$$TD_k = \sqrt{\left(G_k^{RS_x} - G_k^{LH_x} \right)^2 + \left(G_k^{RS_y} - G_k^{LH_y} \right)^2}, \tag{7.3}$$

where TD_k is the size of the torso of person k, $G_k^{RS_x}$ is the ground truth x coordinate of the right shoulder of person k, $G_k^{LH_x}$ is the ground truth x coordinate of the left hip of person k, $G_k^{RS_y}$ is the ground truth y coordinate of the right shoulder of person k, and $G_k^{LH_y}$ is the ground truth y coordinate of the left hip of person k. Then we need to calculate the distance between the predicted key point and the ground truth. The key point is considered correct when the ratio between the distance of the predicted key point and the ground truth, and the size of the torso is below the Percentage of Correct Keypoints (PCK) threshold. This is given by Equation 7.4:

$$PCK_i@a_j = \frac{1}{N_k} \sum_{k=1}^{N_k} \delta \left(\frac{||S_k^i - G_k^i||}{TD_k} \leq a_j \right), \tag{7.4}$$

where PCK_i is the percentage of correct key points of joint i, with $i \in [1, 18]$, a_j is the PCK threshold, N_k is the number of samples in the dataset, δ is a function that is 1 when the condition is true and 0 otherwise, S_k^i is the (X, y) coordinates of the predicted key point i belonging to person k, G_k^i is the (x, y) coordinates of the ground truth key point i belonging to person k. To calculate the PCK for all key points, we average the PCK value for each key point, according to Equation 7.5:

$$PCK@a_j = \frac{1}{N_i} \sum_{i=1}^{N_i} PCK_i@a_j \tag{7.5}$$

We also evaluate the model using the Mean Per Joint Position Error (MPJPE) [188] between the estimated key point and the ground truth, given by Equation 7.6:

$$MPJPE = \frac{1}{N_i \times N_k} \sum_{i=1}^{N_i} \sum_{k=1}^{N_k} \sqrt{\left(S_x^{'i,k} - G_x^{'i,k}\right)^2 + \left(S_y^{'i,k} - G_y^{'i,k}\right)^2}, \tag{7.6}$$

where N_i is the number of key points, with $N_i = 18$, N_k is the number of samples in the dataset, $S_x^{'i,k}$ is the prediction for the x coordinate of key point i belonging to sample k, $G_x^{'i,k}$ is the ground truth for the x coordinate of key point i belonging to sample k, $S_y^{'i,k}$ is the prediction for the y coordinate of key point i belonging to sample k, $G_y^{'i,k}$ is the ground truth for the y coordinate of key point i belonging to sample k.

7.4 Results and Discussion

Tables 7.1–7.3 present the metrics (i.e., PCK, MPJPE) obtained for different numbers of selected sub-carriers ($n \in [5,3,1]$), with two different methods (i.e., rollout, masking), Alphapose confidences (conf. $\in [0,0.6]$) and selection strategies (i.e., high, low, random and baseline). The pattern observed in the results seems to apply to selected sub-carriers and sub-carrier selection method (rollout, masking). When we train a model with the n higher attention sub-carriers, the model tends to achieve better performance (PCK, MPJPE) results, confirming our hypothesis that if the model is paying more attention (i.e., learning higher attention weights) to a given sub-carrier, that probably means that sub-carrier is more informative of the pose of the humans in the Wi-Fi surroundings. Interestingly, results seem to improve when we select n random sub-carriers instead, which, although a bit counter-intuitive, might suggest that a combination of sub-carriers of high- and low-attention is slightly better than using just the highest or lowest attention sub-carriers. Overall results seem to improve when we include only the key points that have an Alphapose confidence greater or equal to 0.6, a phenomenon we consider intuitive since increasing the confidence threshold means that the quality of the pose estimation data will be higher. Nevertheless, both high- and random-attention models seem to compete well against the baseline models (that use all 30 available sub-carriers), meaning that concerning computational efficiency metrics (i.e., frames-per-second processed by the algorithms), presented in Table 7.4, one could still prefer to deploy any of the methods that use a limited number of sub-carriers. These results relate to the field of xAI because we are using the attention weights of the transformer and using this information to reduce the size of the input by selecting the sub-carriers with higher (or lower) attention weights and re-training the model with these to understand if its predictive performance is the same (or closely similar). Our experiments can be perceived almost like a sensitivity analysis, in which the idea is to understand how to change the input (in our case, reducing its size) to maintain the model's predictive performance. Moreover, the fewer sub-carriers we use, the easier it will be to understand why those are more important than others. In Fig. 7.2, we present the subsets of selected sub-carriers according to each different selection method (rollout, masking). Each dot indicates whether a certain sub-carrier was selected.

Table 7.1: Metrics (PCK, MPJPE) obtained for 5 **selected sub-carriers** with different methods (rollout, masking), Alphapose confidences (conf. $\in [0, 0.6]$) and selection strategies (high, low, random and baseline). Results presented for the rollout and the masking methods correspond to 10 and 5 runs, respectively.

Method	**Metric**	**Conf.**	**Selected Sub-carriers** ($n = 5$)			
			High	Low	Random	Baseline
Rollout	PCK ↑	0	0.3083 ± 0.0064	0.2910 ± 0.0112	0.3244 ± 0.0059	0.3166 ± 0.0038
		0.6	0.4436 ± 0.0118	0.4171 ± 0.0125	0.4617 ± 0.0082	0.4524 ± 0.0056
	MPJPE ↓	0	37.11 ± 1.00	38.86 ± 1.24	34.23 ± 0.89	34.69 ± 0.46
		0.6	26.59 ± 1.03	29.32 ± 1.33	24.32 ± 0.85	24.31 ± 0.47
Masking	PCK ↑	0	0.3220 ± 0.0086	0.3076 ± 0.0164	0.3223 ± 0.0067	0.3171 ± 0.0034
		0.6	0.4605 ± 0.0107	0.4385 ± 0.0229	0.4585 ± 0.0088	0.4540 ± 0.0035
	MPJPE ↓	0	34.83 ± 1.48	36.16 ± 2.21	34.46 ± 1.08	34.57 ± 0.37
		0.6	24.69 ± 1.29	26.45 ± 2.33	24.52 ± 1.01	24.20 ± 0.35

Table 7.2: Metrics (PCK, MPJPE) obtained for 3 **selected sub-carriers** with different methods (rollout, masking), Alphapose confidences (conf. $\in [0, 0.6]$) and selection strategies (high, low, random and baseline). Results presented for the rollout and the masking methods correspond to 10 and 5 runs, respectively.

Method	**Metric**	**Conf.**	**Selected Sub-carriers** ($n = 3$)			
			High	Low	Random	Baseline
Rollout	PCK ↑	0	0.2815 ± 0.0191	0.2773 ± 0.0134	0.3153 ± 0.0083	0.3166 ± 0.0038
		0.6	0.4069 ± 0.0279	0.3984 ± 0.0182	0.4505 ± 0.0116	0.4524 ± 0.0056
	MPJPE ↓	0	40.59 ± 2.58	41.28 ± 2.21	35.48 ± 1.66	34.69 ± 0.46
		0.6	30.57 ± 2.85	31.87 ± 2.35	25.75 ± 1.62	24.31 ± 0.47
Masking	PCK ↑	0	0.3092 ± 0.0020	0.2996 ± 0.0093	0.3141 ± 0.0099	0.3171 ± 0.0034
		0.6	0.4448 ± 0.0017	0.4279 ± 0.0129	0.4481 ± 0.0137	0.4540 ± 0.0035
	MPJPE ↓	0	36.74 ± 0.55	37.26 ± 1.19	35.77 ± 1.87	34.57 ± 0.37
		0.6	26.43 ± 0.37	27.33 ± 1.25	25.98 ± 1.77	24.20 ± 0.35

Table 7.3: Metrics (PCK, MPJPE) obtained for 1 **selected sub-carrier** with different methods (rollout, masking), Alphapose confidences (conf. $\in [0, 0.6]$) and selection strategies (high, low, random and baseline). Results presented for the rollout and the masking methods correspond to 10 and 5 runs, respectively.

Method	**Metric**	**Conf.**	**Selected Sub-carriers** ($n = 1$)			
			High	Low	Random	Baseline
Rollout	PCK ↑	0	0.2425 ± 0.0069	0.2353 ± 0.0043	0.2392 ± 0.0078	0.3166 ± 0.0038
		0.6	0.3541 ± 0.0081	0.3387 ± 0.0074	0.3456 ± 0.0118	0.4524 ± 0.0056
	MPJPE ↓	0	46.76 ± 0.53	47.84 ± 0.58	47.07 ± 1.14	34.69 ± 0.46
		0.6	37.00 ± 0.49	38.64 ± 0.75	37.73 ± 1.03	24.31 ± 0.47
Masking	PCK ↑	0	0.2379 ± 0.0071	0.2344 ± 0.0064	0.2426 ± 0.0074	0.3171 ± 0.0034
		0.6	0.3499 ± 0.0072	0.3389 ± 0.0075	0.3514 ± 0.0103	0.4540 ± 0.0035
	MPJPE ↓	0	47.37 ± 0.79	47.09 ± 0.89	46.93 ± 1.45	34.57 ± 0.37
		0.6	37.44 ± 0.78	37.43 ± 0.89	37.38 ± 1.09	24.20 ± 0.35

Table 7.4: Frames per second (FPS) using 1, 3, 5, and 30 sub-carriers using an AMD Ryzen 7 3700X processor.

# Sub-carriers	1	3	5	30
FPS	258.9	244.7	236.4	199.5

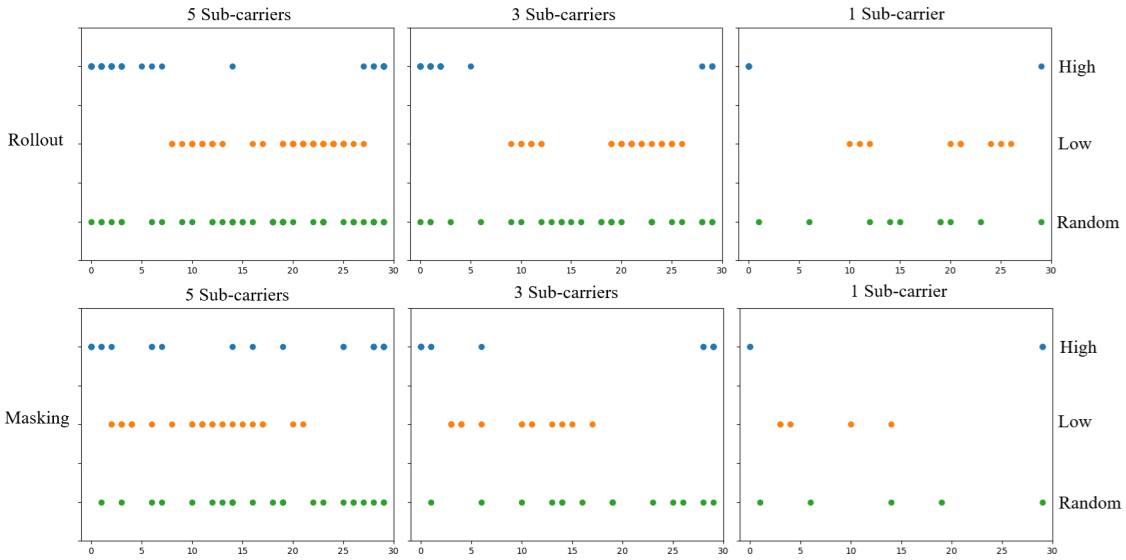

Figure 7.2: Subset of sub-carriers selected from the 30 available sub-carriers, using rollout- or masking-attention when choosing 5, 3, or 1 sub-carriers (10 runs for rollout, 5 runs for masking).

7.5 Conclusions and Future Work

In this chapter, we study a transformer-based model that estimates human pose from Wi-Fi signals, perform a systematic comparison to assess which signal sub-carriers are more important to the model's predictive performance and retrain a new model with the latter. Our results suggest that the models trained with fewer (but relevant) sub-carriers are competitive with the baseline (trained with all sub-carriers) but better in terms of computational efficiency (i.e., processing more data per second). We believe that this work contributes to the field of xAI by using two different methods to select the sub-carriers (one based on attention and another based on masking the inputs), and analysing the model's predictive performance after being trained with the selected sub-carriers. Future work should be devoted to integrating the image data into the pipeline and understanding the model's behaviour when we add this new data modality. Naturally, with the images, a step forward would be to map the selected sub-carriers into explanations (saliency maps) in the image space to have a sense of the location of the image features that are related to relevant sub-carriers: a possible solution could be based on a cross-attention strategy.

Chapter 8

Video Vision Transformers for Privacy-Preserving Human Pose Estimation via Wi-Fi

Foreword on Author Contributions

This work has been submitted and is currently awaiting decision:

Capozzi, L., Sequeira, A.F., Teixeira, L., Oliveira, H.P., Rebelo, A. (2026). Video Vision Transformers for Privacy-Preserving Human Pose Estimation via Wi-Fi and Noise Augmentation. [submitted, waiting for decision]

8.1 Introduction

In the last few years, there have been many advancements in human pose estimation. It consists of identifying the position of human joints in 2D or 3D using data that can belong to different domains. These advancements were driven by applications such as autonomous driving, surveillance, augmented reality, and the recognition of human behaviours. Generally, the data used to estimate the pose can be obtained from different sources, such as cameras (RGB), LiDAR and radar.

The most common type of data is RGB images or video, which is obtained using cameras. The main advantages of this type of data are that it is easy to acquire and cameras do not have a high cost. On the other hand, they are susceptible to occlusions, glare and poor lighting, which decrease the quality of the data and the accuracy of the predictions. There are also privacy concerns, preventing the use of cameras in non-public places. This issue is critical for healthcare applications, where automatic monitoring could be used to ensure the safety and well-being of patients or elderly people, especially in situations where they are transferred back to their homes but still need to be monitored. Most people are not comfortable being filmed inside their home, since it violates their privacy, but if we could solve the privacy issues by using other types of data, it could be especially

useful for the elderly, as it would allow them to be monitored, ensuring their well-being, while also allowing them to be more independent.

The use of LiDAR and radar has also been explored, but their main limitation is their price. These are professional equipment that are very expensive for the average person or small business, which makes it very unlikely for them to be used on a mass scale.

Wi-Fi has some advantages relative to RGB that could make it a better choice in some situations. Firstly, it solves the privacy issues associated with image capture, which makes it a great choice for healthcare use, and ensures the safety of people inside their homes. Occlusions and lighting have little effect on the signal, which is very important for indoor detection since poor lighting and occlusions are very common due to furniture or walls. Another advantage is that Wi-Fi routers are very cheap compared to other types of equipment, and most people already have one at home, which facilitates the scalability of this technology.

The main contributions of this work are the following:

1. The use of Video Vision Transformers in the Wi-Fi domain;
2. A study of 3 different architectures;
3. Conducting a Hyperparameter sweep to find an optimal architecture;
4. Experiments with different Tubelet sizes;
5. Evaluating the impact of the Sliding Window Size and Sliding Window Step;
6. Assessing how changes in Wi-Fi sample frequency (Hz) affect the performance of the model.

8.2 Related Work

8.2.1 Human pose estimation using visual data

Human pose estimation has been extensively studied in the image domain. Methods such as AlphaPose [198], Hourglass networks [203], Mask R-CNN [204] and HR-Net [205] follow a top-down approach, where they first detect regions of interest using networks such as Yolo [206] and Faster RCNN [207], and then use a single person pose estimator to detect the keypoints for each region of interest. These methods achieve a high accuracy on public datasets, but their performance relies greatly on the quality of the data, and poor lighting conditions or occlusions can have an impact on the accuracy of these methods. Additionally, the use of visual data can raise privacy concerns and limit its ability to be used in real-world scenarios.

8.2.2 Human pose estimation using Wi-Fi

Human pose estimation using Wi-Fi is a relatively new research area which aims to reduce the privacy concerns associated with the use of images and facilitate pose detection in situations where occlusions are present. Wang et al. [208] develop a fully convolutional network (WiSPPN) to

estimate the pose of a single person. It uses data acquired from a Wi-Fi transmitter with 3 antennas, a Wi-Fi receiver with 3 antennas, and a synchronised camera used for keypoint annotation. Jiang et al. [209] propose a methodology that is able to reconstruct 3D skeletons of the human body using Wi-Fi signals. Their method takes a 3D velocity profile as input, which enables it to separate static objects in the environment. They use a Recurrent Neural Network (RNN) and a smooth loss. Ren et al. [210] develop a system that estimates 3D skeletons by combining signal separation and joint movement modelling. Ren et al. [183] propose the use of the 2D angle-of-arrival spectrum to estimate the position of 3D keypoints. Their architecture consists of a combination of CNN and Long Short-Term Memory (LSTM), which allows the model to extract both spatial and temporal features. Wang et al. [178] develop a deep learning methodology using CNN, which estimates keypoints and performs body segmentation using 1D Wi-Fi signals. Guo et al. [211] convert the Channel State Information into images, which are then given as input to a convolutional neural network. The Wi-Fi signal is acquired using 3 transceivers and is synchronised with a camera to obtain the annotations. Yu et al. [212] develop a framework that uses a CNN network to estimate keypoints using Wi-Fi signals. The predicted keypoints supervise a Generative Adversarial Network that generates human pose images. Geng et al. [213] use a Modality Translation Network that receives the amplitude and phase in the CSI domain, and transforms it into a feature map in the image domain. An Region-Based Convolutional Neural Network (RCNN) is used to estimate the keypoints using the feature map in the image domain. Zhou et al. [6] develop a methodology that uses an encoder-decoder architecture with the multi-head attention mechanism to encourage the network to pay more attention to relevant features in the input signal. The data used in this work is publicly available (except for the images, due to privacy concerns). Yan et al. [7] extended the work of Wang et al. [178]. They acquire the Wi-Fi signals using a greater number of devices, which captures spatial reflections from multiple people. It uses a Transformer-based architecture to estimate 3D keypoints of multiple people. D. Gian et al. [214] propose the use of a multi-branch convolutional neural network and a selective kernel attention mechanism, which is able to dynamically adjust the size of the kernel, promoting adaptability without increasing the complexity of the model.

8.3 Methodology

The methodology used in this work is based on the ViT [160] and the Video Vision Transformer (ViViT) [201], applied to a different data type, which in our case is Wi-Fi CSI data. The Vision Transformer adapts the original transformer architecture to work with 2D images. It starts by extracting N non-overlapping patches of the image, $x_i \in \mathbb{R}^{h \times w}$ and applying a linear projection. The resulting tensor is flattened into 1D tokens, $z_i \in \mathbb{R}^d$. The input of the transformer encoder is the following:

$$\boldsymbol{z} = [z_{cls}, \boldsymbol{E}x_1, \boldsymbol{E}x_2, ..., \boldsymbol{E}x_N] + \boldsymbol{p} \tag{8.1}$$

where $\boldsymbol{E}$ is a linear projection equivalent to a 2D convolution. A classification token z_{cls} is prepended to the sequence of tokens. This token is an optional, learnable parameter that is used by the final classification layer. Given that the self-attention operations of a transformer are invariant to permutations, a learned positional embedding, $\boldsymbol{p} \in \mathbb{R}^{N \times d}$, is added to the input tokens. The encoder consists of L transformer layers, and each layer ℓ is composed of Multi-Head Self-Attention (MSA), Layer Normalisation (LN), and Multi-Layer Perceptron (MLP) blocks, as shown by the following equations:

$$\boldsymbol{y}^{\ell} = \mathrm{MSA}\left(\mathrm{LN}\left(\boldsymbol{z}^{\ell}\right)\right) + \boldsymbol{z}^{\ell} \tag{8.2}$$

$$\boldsymbol{z}^{\ell+1} = \mathrm{MLP}\left(\mathrm{LN}\left(\boldsymbol{y}^{\ell}\right)\right) + \boldsymbol{y}^{\ell} \tag{8.3}$$

The MLP block consists of two linear layers separated by a Gaussian Error Linear Unit (GELU) activation function. The dimension of the tokens, d, remains fixed throughout all layers of the encoder. If the classification token was added to the input, $z^{L}_{cls} \in \mathbb{R}^{d}$ is used to give the final prediction, otherwise, a global average pooling of all tokens, $\boldsymbol{z}^{L}$ is used. A linear classifier gives the final prediction of the model.

8.3.1 Embedding Wi-Fi signal

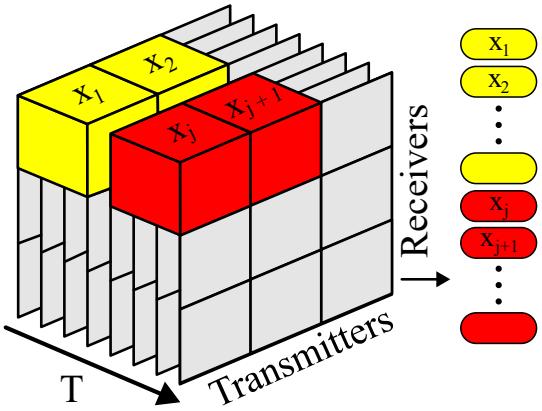

Figure 8.1: Mapping the Wi-Fi signal into a sequence of tokens.

In order to map the Wi-Fi signal into a sequence of tokens, we follow a similar strategy to [201], where the Wi-Fi channel state information $S \in \mathbb{R}^{C \times T \times H \times W}$ is mapped to a sequence of tokens $\tilde{z} \in \mathbb{R}^{nc \times nt \times nh \times nw}$, where C, T, H and W are the sub-carriers, time steps, transmitting antennas and receiving antennas, respectively. Then the positional embedding is added and the sequence of tokens is reshaped into $\mathbb{R}^{N \times d}$, to obtain z, which is the input to the transformer.

Figure 8.1 shows how the Wi-Fi channel state information is mapped into a sequence of tokens. We extract non-overlapping, spatio-temporal "tubes" from the input signal, and linearly project each "tube" into $\mathbb{R}^{d}$. This type of mapping is first explored by [201], which is an extension into 3D

of ViT's embedding [160]. This method corresponds to a 3D convolution, where the kernel size and the stride are set to the same value. A tubelet dimension of $t \times h \times w$ results in $n_t = \lfloor T/t \rfloor$ tokens extracted from the temporal dimension, $n_h = \lfloor H/h \rfloor$ tokens extracted from the height dimension, and $n_w = \lfloor W/w \rfloor$ tokens extracted from the width dimension.

Using a tubelet dimension of $1 \times h \times w$ results in the same tokenisation strategy as ViT [160], where each patch is extracted from a single frame, without fusing spatio-temporal information during tokenisation.

8.3.2 Architectures

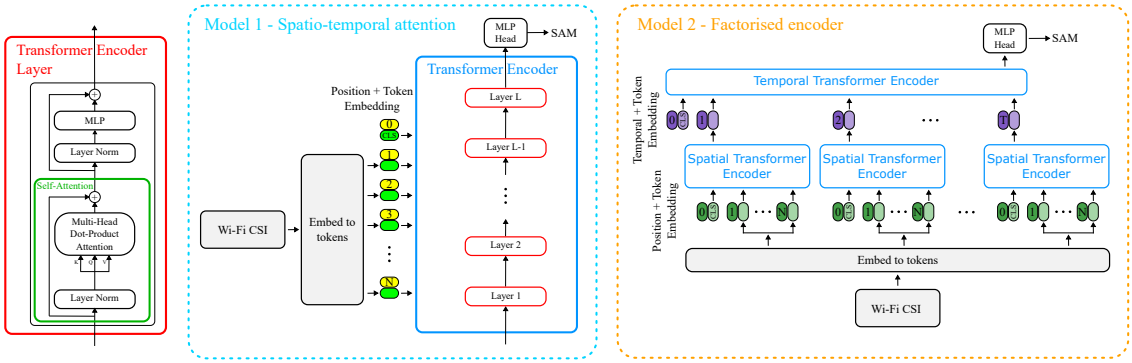

Figure 8.2: Architectures used in this work, based on [201]. Model 1: Spatio-temporal attention processes all spatio-temporal tokens simultaneously, by passing them through a transformer encoder. Model 2: Factorised encoder is composed of a spatial transformer encoder and a temporal transformer encoder. The spatial transformer encoder processes tokens belonging to the same temporal index. The temporal encoder then aggregates and processes tokens from different temporal indices. This model follows a "late fusion" approach, where the signal is first processed independently for each time index and then aggregated.

8.3.2.1 Model 1: Spatio-temporal attention

This model is based on one of the architectures proposed by Arnab et al. [201]. It processes all spatio-temporal tokens extracted from the Wi-Fi signal, z^0, by passing them through the transformer encoder. Each transformer encoder layer is composed of Layer Normalisation, Multi-Head Self-Attention, and MLP layers, as seen in Figure 8.2. Transformer layers compute pairwise interactions between all tokens, and as a result, long-range dependencies across the entire signal are captured throughout the whole network. However, this has a computational cost, where the computational complexity is quadratic with respect to the number of tokens. This is particularly important with time-series data, such as Wi-Fi, where the number of tokens grows linearly with the number of time steps, motivating the use of more efficient architectures.

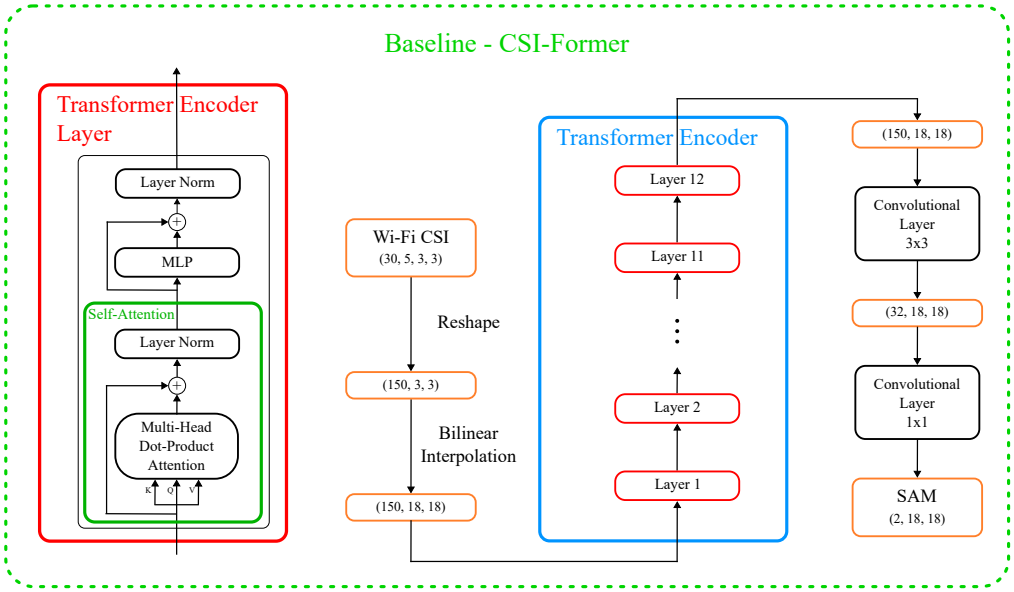

Figure 8.3: Baseline network used in this work [6]. It is composed of 12 transformer encoder layers and 2 convolutional layers for estimating keypoints based on the input Wi-Fi CSI.

8.3.2.2 Model 2: Factorised encoder

This model is based on one of the architectures proposed by Arnab et al.[201]. As shown in Figure 8.2, this architecture is composed of two different transformer encoders. The first is the Spatial Transformer Encoder, which processes interactions between tokens belonging to the same temporal index. After L_s layers, a representation for each temporal index is obtained, $h_i \in \mathbb{R}^d$. It contains the encoded classification tokens $z_{cls}^{L_s}$, if it was added, or a global average pooling of the tokens output by the spatial encoder, z^{L_s}. These representations, h_i, are concatenated into $H \in \mathbb{R}^{n_t \times d}$, and passed to the temporal encoder, which consists of L_t transformer encoder layers. It then processes the interactions between tokens from different temporal indices. The output of the temporal encoder is a classification token $z_{cls}^{L_t}$, if it was added, or a global average pooling of the tokens. The keypoints are estimated using an MLP Head, which receives as input the output of the temporal encoder.

This architecture follows a "late fusion" approach, where spatial features are extracted independently for each temporal index and then aggregated.

8.3.3 Loss Function

The loss function used in this work is described by Equation 7.2.

8.4 Experiments

8.4.1 Data

In this work, we use the Wi-Pose dataset [6]. The data was captured using Wi-Fi devices and cameras, and the pose keypoint information was extracted by the creators using Alphapose [198]. There are 12 different actions performed by 12 volunteers. The authors provide the Wi-Fi channel state information and the pose annotation extracted using the images. The original dataset is composed of 166,600 packets, which are then divided into 132,847 packets for training and 33,753 packets for testing. Each packet contains 2 tensors: a 30x5x3x3 CSI tensor, and a 3x18 pose annotation tensor. The CSI tensor has 30 sub-carriers, 5 time steps, 3 transmitting antennas and 3 receiving antennas. The pose tensor has 18 keypoints, and each keypoint contains the x and y coordinates, and the confidence value (x, y, c). Notice that for each pose annotation tensor, there are 5 Wi-Fi time steps. This is because the Wi-Fi sampling rate is 100Hz and the camera sampling rate is 20Hz, which gives 5 Wi-Fi samples for each camera frame.

After analysing the dataset, we noticed that there was a cutoff in the Wi-Fi signal after 5 seconds, therefore, we decided to only consider the first 5 seconds of each clip to avoid Wi-Fi signals containing no information. After this change, the number of packets was reduced to 111,586 packets.

The dataset creators split the packets using a window size of 5 and a step size of 5 samples (see Figure 8.5). Instead of stepping 5 samples, we propose stepping 1 sample, which increases by a factor of 5 the number of packets in the dataset. The number of packets using a step size of 1 and a window size of 5 is 553,322. This also has the advantage of reducing overfitting, since we increase the amount of distinct packets.

The goal of this work is to estimate the position of each of the 18 keypoints, but many works have shown that it is easy to overfit if we only try to predict the position of the keypoints. Therefore, following Zhou et al. [6] we calculate and use the SAM, which not only contains the information about the position of each keypoint, but also the relative distance between each of them. The SAM is calculated using the original 3×18 matrix $(x_i, y_i, c_i), i \in [1, 18]$, and expanding the last dimension. The SAM consists of a $3 \times 18 \times 18$ matrix $(x_{i,j}, y_{i,j}, c_{i,j}), (i, j \in [1, 18])$, and it is calculated in the following way:

$$x'_{i,j} = \begin{cases} x_i - x_j, & \text{if } i \neq j. \\ x_i, & \text{if } i = j. \end{cases} \tag{8.4}$$

$$y'_{i,j} = \begin{cases} y_i - y_j, & \text{if } i \neq j. \\ y_i, & \text{if } i = j. \end{cases} \tag{8.5}$$

$$c'_{i,j} = \begin{cases} c_i \times c_j, & \text{if } i \neq j. \\ c_i, & \text{if } i = j. \end{cases} \tag{8.6}$$

8.4.2 Baseline

The baseline model used in this work is CSI-Former [6], which uses a teacher-student strategy. The teacher network is Alphapose [198], and it is used to generate ground truth pose annotations from images. The student network is based on the Transformer architecture, and it starts by reshaping and interpolating the CSI data to match the dimensions of the ground truth pose annotations. Then the data is passed through a 12-layer transformer encoder and two convolutional layers, which give the final pose estimation. They created the publicly available dataset used in this work, which consists of the Wi-Fi channel state information and the ground truth pose annotations extracted from images using Alphapose.

8.4.3 Testing Protocol

To evaluate how accurate the models are, we use the percentage of correct key points ($PCK@a_j$), described in Section 7.3.2.6.

8.4.4 Implementation Details

This work is implemented using PyTorch. We train the network using a batch size of 512. The network is randomly initialised and trained from scratch using the Adam Optimiser, with an initial learning rate of $5e-4$. The learning rate is decreased to $5e-5$ after 800 epochs and $5e-6$ after 1100 epochs. The training procedure is 1400 epochs, and we save the last model since the dataset used does not contain a validation set.

8.4.5 Architecture Hyperparameter Sweep

In order to find an optimal model architecture and to see the impact that each of the hyperparameters has on the performance of the model, we perform a hyperparameter sweep. The data used has a dimension of $30 \times 5 \times 3 \times 3$, with a step size of 5 samples, which is the same as CSI-Former [6]. We use model 1 with a tubelet size of $1 \times 1 \times 1$, and PCK@5 to evaluate the model.

The hyperparameters that we are optimising during the sweep, along with their possible values, are the following:

1. Token Dimension $\in \{64, 128, 256, 512\}$
2. MLP Dimension $\in \{64, 128, 256, 512\}$
3. Number of Heads $\in \{2, 4, 8\}$
4. Number of Layers $\in \{2, 4, 6, 8, 10, 12\}$

We use Bayesian search to find the optimal hyperparameter combination, which in contrast to grid search and random search, makes informed decisions based on the accuracy of the models that are tested during the search.

We perform a total of 45 runs, which can be visualised in Figure 8.4.

The best hyperparameter combinations can be seen in Table 8.1.

We notice that the best results are obtained with a Token Dimension of 256 or 128, an MLP Dimension of 512, a Number of Heads equal to 2 (although other sizes do not seem to have a big impact on the results) and a Number of Layers of 12 or 10.

Focusing on run #4 we notice that the hyperparameters are similar to those suggested in the ViT [160] paper, where the MLP Dimension is 4 times the size of the Token Dimension, and the Number of Layers is also 12. Using a Token Dimension of 128 instead of 256 improves computational performance without drastically decreasing the accuracy. Therefore, we adopt the hyperparameters of run #4 in the following experiments.

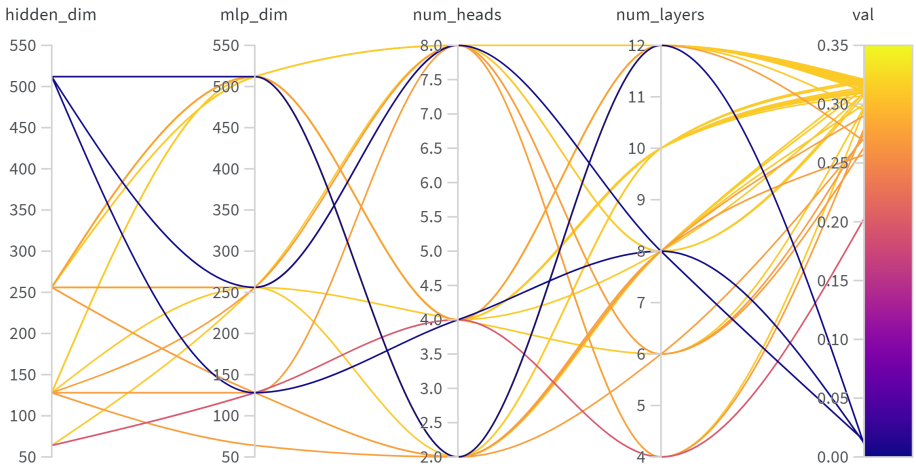

Figure 8.4: Relationship between hyperparameters used and accuracy achieved.

Table 8.1: Top-5 models after performing the hyperparameter sweep. The data used has a dimension of $30 \times 5 \times 3 \times 3$, with a step size of 5 samples, which is the same as CSI-Former [6]. We use model 1 with a tubelet size of $1 \times 1 \times 1$, and PCK@5 to evaluate the model.

Run #	Token Dim	MLP Dim	Nº of Heads	Nº of Layers	PCK@5
1	256	512	2	12	31.85
2	256	512	4	12	31.84
3	256	512	2	10	31.82
4	128	512	2	12	31.71
5	128	512	2	8	31.52

8.4.6 Tubelet Size

We train both models using different tubelet sizes to see the impact it has on performance. All models are trained from scratch. The input size is 30x8x3x3, which corresponds to 30 wireless sub-carriers, 8 time steps at a frequency of 100Hz, 3 transmitters and 3 receivers. The input

size is the same for all experiments in this section. The input size needs to be divisible by the tubelet size, therefore, we train 8 different configurations, varying the time dimension and the transmitter/receiver dimensions.

The results are shown in Table 8.2. We can see that increasing the tubelet size improves the results. We also notice that models using smaller tubelet sizes seem to overfit more, as they have a lower training loss than models using larger tubelet sizes, but their PCK@5 is worse. Our experiments show that Model 2 is better than Model 1, as it achieves a better PCK@5, while at the same time having a larger training loss, which indicates that there is not as much overfitting to the training data.

Table 8.2: Comparison between different tubelet sizes, using an input size of 30x8x3x3. We use PCK@5 to evaluate the model. Note: M1 is Model 1 and M2 is Model 2.

Tubelet Size	M1 loss	M2 loss	M1 PCK @ 5	M2 PCK @ 5
1x1x1	9.34	10.23	30.74	33.71
2x1x1	10.83	12.77	32.35	34.28
4x1x1	13.35	14.69	33.37	34.48
8x1x1	14.18	17.85	33.79	35.62
1x3x3	9.66	10.28	32.11	33.25
2x3x3	10.59	11.85	34.23	35.06
4x3x3	15.32	14.09	35.01	35.92
8x3x3	17.11	18.63	35.03	36.44

8.4.7 Sliding Window Size

In order to measure the impact that the temporal window has on the model, we try 4 different sizes for the sliding window. The values shown in the first column of Table 8.3 correspond to the number of time steps that we use in the temporal window. The Wi-Fi signal has a frequency of 100Hz, which means that the temporal window size in ms is 50, 150, 250, and 550, respectively. For this experiment, we use a tubelet size of 5x1x1.

Looking at the result, we notice that increasing the time window makes the model worse. This seems counterintuitive at first since the model should be able to predict the keypoints more accurately as it has more contextual temporal information. But if we analyse the training loss, we can see that while the accuracy of the predictions drops with the increase of the time window, the training loss decreases. This suggests that the model is overfitting to the training data, causing it to have poor generalisation capabilities, therefore reducing the accuracy of the predictions. A solution to this would be to increase the amount of data, either by collecting more or artificially increasing the number of samples using data augmentation techniques.

8.4.8 Sliding Window Step

The dataset used in this work aligns one image frame to five CSI frames, which gives a total of 111,586 samples. Due to the limited amount of data, we try using a step size of 1, which increases

Table 8.3: Comparison between different time window sizes, using a tubelet size of 5x1x1. We use PCK@5 to evaluate the model. Note: M1 is Model 1 and M2 is Model 2.

Time Window size	M1 loss	M2 loss	M1 PCK @ 5	M2 PCK @ 5
5	16.01	16.81	34.12	35.22
15	11.41	14.08	32.03	33.38
25	9.92	11.63	31.53	32.43
55	8.61	10.09	30.22	32.48

the number of samples to 553,322. We do this to avoid overfitting since applying traditional data augmentation techniques, such as those used with images, is not possible with CSI data. Figure 8.5 shows how samples with different step sizes are collected.

The input size is 30x5x3x3, which corresponds to 30 wireless sub-carriers, 5 time steps, 3 transmitters and 3 receivers. For these experiments, we use a tubelet size of 5x1x1. Analysing the results in Table 8.4, we can see that there is an improvement on the PCK@5 metric when using a step size of 1. We believe the additional samples that are given to the model allow for better generalisation.

Table 8.4: Comparison between different step sizes using a tubelet size of 5x1x1. We use PCK@5 to evaluate the model. Note: M1 is Model 1 and M2 is Model 2.

Step size	M1 loss	M2 loss	M1 PCK @ 5	M2 PCK @ 5
1	16.01	16.81	34.12	35.22
5	17.00	20.22	33.56	35.16

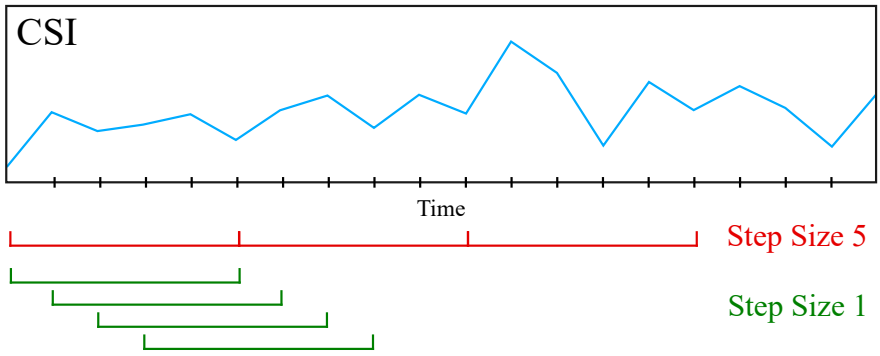

Figure 8.5: Comparison between different step sizes.

8.4.9 Sampling Rate

We perform some experiments regarding the CSI sampling rate to see the impact on the keypoint prediction accuracy. We choose frequencies ranging from 2Hz to 100Hz. The tubelet size used in these experiments is 1x3x3, to reduce the number of tokens and the GPU memory required.

The results are presented in Table 8.5, and they show an increase in keypoint prediction accuracy as the frequency decreases. This can seem contradictory, as the model should have a better performance when it has access to more information. However, looking at the training loss, we can see that having more information (a higher frequency) allows the model to be more accurate on the training set. The problem is that the model is not able to generalise well, and the accuracy decreases on the testing set. This could be because increasing the frequency and therefore the size of each sample causes the model to overfit more to the training data, as it has more information available. This is problematic, since a model that is overfit to the training data is more sensitive and more negatively impacted by the slight domain differences between the training set and the testing set, as can be seen by the decrease in keypoint prediction accuracy. Increasing the amount of data and collecting data in different conditions, with many different actors performing a large collection of actions, could be beneficial for the generalisation capabilities of the model.

Table 8.5: Comparison between different Wi-Fi sample frequencies. We use PCK@5 to evaluate the model. Note: M1 is Model 1 and M2 is Model 2.

Wi-Fi Freq (Hz)	Input Size	M1 loss	M2 loss	M1 PCK @ 5	M2 PCK @ 5
100	100x3x3	6.24	7.18	29.05	28.96
50	50x3x3	6.82	8.49	28.64	30.36
25	25x3x3	8.25	9.59	31.38	32.53
20	20x3x3	8.21	9.67	31.70	33.33
10	10x3x3	10.46	11.33	34.20	34.52
5	5x3x3	12.94	13.37	36.45	36.02
2	2x3x3	14.94	16.39	38.01	37.91

8.4.10 Architecture Comparison

We compare three different architectures: the baseline architecture, based on CSI-Former [6], Model 1 (Spatio-temporal attention), and Model 2 (Factorised encoder) [201]. The input size is 30x5x3x3, which corresponds to 30 wireless sub-carriers, 5 time steps, 3 transmitters and 3 receivers, and a tubelet size of 5x1x1 for models 1 and 2.

Table 8.6 shows the results for each of the architectures. The best results were achieved using model 2, with a PCK @ 5 of 35.22, followed by model 1, with a PCK @ 5 of 34.12, and finally CSI-Former, with a PCK @ 5 of 27.96.

The main difference between the baseline model, and model 1 and 2, is the processing of the Wi-Fi CSI before the transformer encoder. The baseline performs a bilinear interpolation of the input data to match the dimensions to the size of the SAM. This increases the size of the input data, but does not add any additional information, which leads to a decrease in computational performance, since the number of input tokens also increases, and does not seem to improve the accuracy. Model 1 and model 2, on the other hand, extract non-overlapping, spatio-temporal "tubes" from the input signal, which are then linearly projected before giving them to the transformer encoder. This linear projection is also used in the Vision Transformer [160].

Table 8.6: Comparison between the Baseline, Model 1 and Model 2, using an input size of 30x5x3x3, which corresponds to 30 wireless sub-carriers, 5 time steps, 3 transmitters and 3 receivers, and a tubelet size of 5x1x1. We use PCK@5 to evaluate the models. Note: The results on the baseline are from our implementation of the method, since the source code was not provided.

Model	Loss	PCK @ 5
Baseline (CSI-Former)	20.70	27.96
Model 1 (Spatio-temporal attention)	16.01	34.12
Model 2 (Factorised encoder)	16.81	35.22

8.5 Conclusions

In this chapter, we focus on the use of video vision transformers in the Wi-Fi domain. We perform a hyperparameter sweep to find the optimal architecture hyperparameters, and we also perform experiments regarding the data, such as varying the sampling rate, the sliding window size, and the sliding window step size. We find that the optimal hyperparameters are similar to those found in the ViT model [160]. We find that bigger tubelet sizes lead to better results, as well as shorter time windows, a smaller step size, and a lower Wi-Fi sample frequency. For future work, we would like to experiment with other publicly available datasets.

Chapter 9

Impact of Noise Augmentation on Wi-Fi-Based Human Pose Estimation and Action Recognition

Foreword on Author Contributions

This work has been submitted and is currently awaiting decision:

Capozzi, L., Sequeira, A.F., Teixeira, L., Oliveira, H.P., Rebelo, A. (2026). Video Vision Transformers for Privacy-Preserving Human Pose Estimation via Wi-Fi and Noise Augmentation. [submitted, waiting for decision]

9.1 Introduction

Wi-Fi-based sensing has recently emerged as a promising alternative to vision-based approaches for human pose estimation and action recognition. By exploiting variations in Wi-Fi CSI caused by human motion, these systems can infer fine-grained information about human behaviour without relying on cameras, offering advantages in terms of privacy, robustness to lighting conditions, and deployment flexibility. As a result, Wi-Fi-based sensing has gained increasing attention in applications such as smart environments and healthcare monitoring.

Despite these advantages, Wi-Fi-based models are still sensitive to noise and variations introduced by environmental factors, and changes in deployment conditions. These sources of variability can negatively impact generalization performance, particularly when models are trained on small or relatively clean datasets. Improving robustness to such variations remains an important challenge for Wi-Fi-based human sensing systems.

Data augmentation is a widely adopted strategy to mitigate overfitting and improve generalization in deep learning models. While augmentation techniques are extensively studied in computer vision, their application to Wi-Fi sensing has received comparatively less attention. Typical image

data augmentation techniques such as crops, translations and rotations cannot directly be used for Wi-Fi, motivating the need for alternatives.

In this chapter, we investigate the effect of noise-based data augmentation on Wi-Fi-based human pose estimation and action recognition. We add controlled amounts of noise into the CSI data during training and evaluate its impact across three different architectures and two datasets. By analyzing different noise distributions and noise magnitudes, we aim to better understand how noise augmentation influences model robustness and performance.

9.2 Related Work

Data augmentation is a well-established technique to improve generalization and avoid overfitting in deep learning models, particularly when training data is limited. In computer vision, common augmentation strategies include geometric transformations such as crops, translations and rotations, or other types of transformations, such as color jitter.

Although Wi-Fi data augmentation is a relatively unexplored topic, there are still some contributions. Hyunggeun Mo and Seungku Kim [215] use sliding windows and time warping as data augmentation for Wi-Fi human identification. Julian Strohmayer and Martin Kampel [216] propose the use of four data augmentations techniques applied to CSI amplitude spectrogram: randomCircularRotation, randomResizedCrop, randomAmplitude, and randomContrast. Weiying Hou and Chenshu Wu [217] propose a methodology called RFBoost, which is a plug-and-play module that encompasses different data augmentation techniques. Wen et al. [218] develop a data augmentation method using Generative AI, with a transformer-based diffusion model to generate high-quality CSI data.

9.3 Methodology

In this section we present the data used to train and evaluate the models, what our proposed data augmentation technique consists of, the architectures used, and the training and evaluation details.

9.3.1 Data

In this work we use two datasets: the Wi-Pose dataset [184], previously described in Section 7.3.1, and the Person-in-WiFi 3D dataset [188].

The Person-in-WiFi 3D dataset was acquired with the help of 7 volunteers, with heights between 160-177cm and weights between 55-70kg. They performed 8 different actions in 3 locations: an office, a classroom, and a corridor. The actions were the following: reaching out, raising hands, bending over, stretching, sitting down, lifting legs, standing, and walking. The actors performed these actions in each location for 40 seconds. They recorded 152 clips with 40 seconds each, including 32 single-person, 48 two-person, 48 three-person, and 24 four-person scenarios. Given the 3 locations, this resulted in a total of 456 CSI and RGB-D clips. Each clip contains a total of

600 frames (550 for training, and 50 for testing). The multi-person 3D keypoints were generated using the Kinect Body Tracking SDK, however, the performance on some of the clips was not entirely accurate, therefore the dataset was cleaned, and the samples where the SDK failed were discarded. The resulting dataset has over 97000 samples.

9.3.2 Noise Injection

In order to avoid overfitting and improve the results of the models, we propose using noise as a data augmentation technique. Our intuition is that by adding noise the model will be able to generalize better to unseen data, by becoming more robust to changes.

We tested two different types of noise: noise obtained from a normal distribution, and noise obtained from an uniform distribution. We can adjust the amount of noise that we add to the data by choosing different values for the Standard Deviation (STD) of the normal distribution, or the range of the uniform distribution. In order to have consistency across datasets we start by calculating the standard deviation of the training dataset ($\sigma_{dataset}$) and then multiply it by the ratio (r) that we want to use. A ratio of 0 is equivalent to using the original data without modifications, which gives us our baseline models. This will give us the final STD ($\sigma = r \times \sigma_{dataset}$) or range that will be used during training. In the case of the normal distribution we use a mean of 0 and a STD of σ, and for the uniform distribution we use a range of $[-\sigma, \sigma]$. After the amount of noise is selected we generate a set of random numbers from the corresponding distribution and add them to the training data. During testing no noise is added to the data.

We decided to test these two types of noise since the normal distribution has an unlimited range with values more concentrated around the mean, and the uniform distribution has a limited range with uniformly distributed values, and we wanted to test if this had any sort of impact on the results.

9.3.3 Architectures

In this work we use the same architectures that were used in Chapters 7 and 8, and we also use the methodology developed by [188] using the original source code and dataset. For the previously used models, we used the Transformer architecture in Figure 7.1, and the ViViT architecture referred to as Model 1 - Spatio-temporal attention, described in Figure 8.2.

The main difference between those architectures is the tokenization strategy used on the input CSI signal. Each of the strategies are described in detail in Sections 7.3.2.1 and 8.3.1, for the Transformer, and the ViViT architectures, respectively. For the Transformer architecture the CSI data is flattened, and then passed through a linear layer. For the ViViT architecture the CSI data is tokenized by passing it through a 3D Convolution layer, where the kernel size and the stride are set to the same value.

The output of the model is the SAM, in the case of pose estimation, or a tensor of size $N_{classes}$, where $N_{classes}$ is the number of action classes in the dataset.

For the pose estimation models we use the same implementation details described in Section 8.4.4.

Due to the long training time of the Person-in-WiFi 3D methodology [188] we start by training a model from scratch using a ratio of 0.01, and then fine-tune that model for 100 epochs for each of the other ratios. Since we were unable to achieve the results reported in the original paper, and the model was not converging correctly, we used the model that was trained with a ratio of 0.01, which seemed to be more stable, and had results closer to the original paper, for the rest of the experiments.

The action recognition models are trained for a total of 300 epochs, with the Adam Optimiser, and an initial learning rate of $5e-4$. The learning rate is decreased to $5e-5$ after 100 epochs and $5e-6$ after 200 epochs. The sliding window size used for the action recognition models is 100, which corresponds to a CSI tensor of size $30 \times 100 \times 3 \times 3$. We decided to increase the sliding window size for action recognition to give the model more context in the temporal dimension, since actions happen over longer time periods. For reference, the CSI tensor used for pose estimation has a size of $30 \times 5 \times 3 \times 3$.

For the ViViT architecture we use a tubelet size of $5 \times 1 \times 1$ for the pose estimation model, and a tubelet size of $10 \times 3 \times 3$ for the action recognition model.

9.3.4 Loss Function

The loss function used for the pose estimation model is described by Equation 7.2. For the Person-in-WiFi 3D methodology we use the loss function in the original implementation. The loss function used by the action recognition models is the Cross-Entropy Loss.

9.3.5 Testing Protocol

To evaluate how accurate the pose estimation models are, we use the percentage of correct key points ($PCK@a_j$), described in Section 7.3.2.6. To evaluate the Person-in-WiFi 3D methodology we use the MPJPE, described in Equation 7.6. To evaluate the action recognition models we calculate the accuracy per class (number of correctly classified samples divided by the total number of samples), and then calculate the global accuracy by averaging the accuracies across all classes.

9.4 Results and Discussion

The pose estimation results are presented in Table 9.1, and the action recognition results are presented in Table 9.2. The results of the experiments are aligned with our expectations, where slowly increasing the ratio seems to improve the results, until the amount of noise that is added to the data becomes too large, to the point that it becomes detrimental and the accuracy decreases again.

The model also seems to be less prone to overfitting, as can be seen by the simultaneous increase in training loss value and test accuracy. When adding noise to the data the model starts to have a worse performance on the training set, as can be seen by the increase in the loss value, which is a sign that the model is not overfitting as much to the training data, leading to a more robust and

generalizable model. The increase in performance is not as noticeable for action recognition, which we believe is due to the fact that the baseline model already has a high accuracy, and therefore less room for improvement. The use of normal distributions versus uniform distributions does not seem to have a big impact on the results.

Our baseline model for the Person-in-WiFi 3D methodology [188] has a higher MPJPE than what was reported in the original paper. We used the provided code without any modifications, but were unable to reproduce the results. Nevertheless, when applying our noise data augmentation technique, the results improved considerably, and were closer to the original reported value of 107.2.

Table 9.1: Pose estimation performance using different ratios. For the Transformer and ViViT architectures we used the Wi-Pose dataset [184], and for the Person-in-WiFi 3D architecture we used the Person-in-WiFi 3D dataset [188].

Ratio	**Transformer**				**ViViT**				**Person-in-WiFi 3D**			
	Normal		Uniform		Normal		Uniform		Normal		Uniform	
	PCK@5 ↑	Loss ↓	PCK@5 ↑	Loss ↓	PCK@5 ↑	Loss ↓	PCK@5 ↑	Loss ↓	MPJPE ↓	Loss ↓	MPJPE ↓	Loss ↓
0.00	0.31685	12.24868	0.31685	12.24868	0.34124	16.01424	0.34124	16.01424	880.35	3.1300	880.35	3.1300
0.01	0.31485	12.25131	0.31640	12.60573	0.33944	15.78155	0.33995	16.44822	120.28	1.2644	120.59	1.2319
0.05	0.31524	12.53269	0.31369	12.61389	0.34147	16.58889	0.33665	16.09343	121.62	1.2975	**119.25**	1.2650
0.10	0.31810	13.18259	0.31266	12.62024	0.34193	17.91318	0.34635	16.52099	**118.02**	1.3474	**119.25**	1.2830
0.15	0.32218	13.91511	0.31456	13.06239	0.34733	19.70521	0.34292	17.60245	119.44	1.4385	120.07	1.3063
0.20	0.32311	14.81370	0.31628	13.52973	0.34698	20.82833	0.34663	18.26443	121.02	1.4773	121.15	1.3334
0.50	0.32900	23.20849	0.32653	17.00948	0.34659	30.06286	0.34975	23.14275	129.87	2.2326	124.31	1.6934
0.75	0.34327	33.10980	0.32943	20.98216	**0.35208**	41.26341	0.34489	27.78020	144.73	2.9490	131.15	2.1408
1.00	**0.34517**	49.11285	0.33304	25.62325	0.34808	56.85434	**0.35081**	35.03422	176.45	3.8061	137.78	2.4046
1.25	0.34436	65.90693	0.34010	32.64342	0.34736	78.04866	0.35003	39.96779	214.76	4.5808	156.95	2.8714
1.50	0.33828	95.28187	0.33625	40.58332	0.33641	104.94761	0.34978	49.44440	265.32	5.4023	182.09	3.3537
1.75			0.34322	49.91386	0.33159	137.43253	0.34735	58.18072	302.95	6.2746	165.42	3.8897
2.00			**0.34659**	61.34390			0.34470	71.08899	337.97	7.5822	209.62	4.4796
2.25			0.34153	74.85415			0.34396	82.57658			231.78	4.7826

Table 9.2: Action recognition accuracy using different ratios. For these experiments we used the Wi-Pose dataset [184].

Ratio	**Transformer**				**ViViT**			
	Normal		Uniform		Normal		Uniform	
	Accuracy ↑	Loss ↓	Accuracy ↑	Loss ↓	Accuracy ↑	Loss ↓	Accuracy ↑	Loss ↓
0.00	0.89236	4.48E-08	0.89236	4.48E-08	0.91767	5.90E-09	0.91767	5.90E-09
0.01	0.89194	2.33E-08	0.88475	1.47E-07	0.91559	3.14E-09	0.91624	1.35E-08
0.05	0.88655	4.21E-08	0.89677	1.41E-07	0.91809	2.89E-09	0.90964	6.63E-09
0.10	0.89868	1.60E-08	0.88855	3.39E-08	0.91440	1.82E-08	0.90476	4.18E-09
0.15	0.89214	7.16E-08	**0.90391**	2.55E-08	0.91452	3.89E-09	0.91714	1.51E-09
0.20	0.89654	5.04E-08	0.89402	6.71E-08	**0.92154**	6.40E-09	0.92094	1.73E-09
0.50	0.89998	1.13E-07	0.89797	7.10E-08	0.91649	1.06E-08	0.91812	6.85E-09
0.75	**0.90404**	3.40E-07	0.89949	1.15E-07	0.91987	7.45E-08	0.90796	7.57E-09
1.00	0.90321	7.83E-07	0.89734	2.64E-07	0.91761	1.43E-07	0.91837	9.50E-09
1.25	0.89445	5.50E-06	0.89773	2.37E-07	0.92021	1.20E-06	0.91434	1.96E-08
1.50	0.89597	2.62E-05	0.89470	4.69E-07	0.91342	6.30E-06	**0.92109**	3.90E-06
1.75	0.89364	5.82E-05	0.89450	1.10E-05	0.91996	3.18E-05	0.91425	2.19E-07
2.00			0.90123	1.70E-06			0.92089	1.42E-06
2.25			0.89866	3.28E-05			0.91664	1.00E-06

9.5 Conclusion

This chapter explores the use of noise as a data augmentation technique for Wi-Fi CSI. We experiment with different amounts of noise using three different architectures, and two dataset, to estimate human pose and recognize human actions. Overall, the results seem to be aligned with what we expected, where small amounts of noise improve the performance of the model, and too much noise becomes detrimental and decreases the performance.

Part V

Epilogue

Chapter 10

Conclusions

This thesis investigated deep learning methodologies for human-centric perception tasks under challenging real-world conditions, including low amounts of data, actor bias, occlusion, limited computational resources, and privacy constraints. The work was organised as a collection of research papers, each addressing a specific problem within the broader context of human action recognition, person re-identification, and human pose estimation, using both vision-based and Wi-Fi based sensing modalities. This chapter summarises the main conclusions drawn from each contribution.

The main contributions and conclusions of this thesis are summarised below:

- **Chapter 3:** This chapter analysed the performance of deep learning models for activity recognition in in-vehicle environments, with particular focus on the impact of input resolution, aspect ratio, field of view, and frame rate. Due to the lack of dedicated in-vehicle datasets, a set of publicly available action recognition datasets was curated and adapted to the target scenario. Experimental results showed that models are generally robust to moderate resolution changes and can generalise to unseen datasets, especially after fine-tuning. Importantly, using lower resolutions and frame rates enabled substantial computational savings with only minor accuracy degradation, making such configurations suitable for real-time and resource-constrained systems. The study also demonstrated that training and testing at similar frame rates leads to the best performance, while higher frame rates provide diminishing returns for actions characterised by slow movements.

- **Chapter 4:** This chapter addressed the problem of actor bias, which arises when models overfit to the appearance and motion patterns of a small number of individuals. Three complementary strategies were investigated: adversarial removal of actor-specific information, gradient-based mini-batch weighting guided by validation data, and the use of actor-independent pose representations. While the adversarial approach did not consistently outperform the baseline, the gradient-based weighting strategy led to measurable improvements by prioritising training samples that enhanced generalisation to unseen actors. The use of pose information yielded improvements on public datasets and comparable performance

on real in-vehicle datasets, indicating its potential as a bias-reduction mechanism in scenarios with limited actor diversity.

- **Chapter 5:** This chapter proposed a novel method for generating attention masks that highlight the most informative regions of an image for person re-identification. By training the feature extraction network on both original images and images with missing information, the model was encouraged to rely on less distinctive but more robust visual cues. Experimental evaluation on three benchmark datasets showed that the proposed approach achieved superior Rank-1 accuracy compared to competing methods, while maintaining comparable mAP. The results demonstrated that learning to cope with missing information can improve identification robustness, although excessive information removal may introduce noise and destabilise training.

- **Chapter 6:** This chapter introduced an end-to-end architecture for occluded person re-identification, combining global features, feature dropping, and occlusion detection with two types of artificial occlusion augmentation. Random shape occlusions improved general robustness, while object-based occlusions provided more realistic training scenarios and proved particularly effective for occluded datasets. The proposed method achieved competitive performance on holistic datasets and strong results on partial and occluded datasets. An ablation study further demonstrated the superiority of the proposed occlusion strategies over commonly used random erasing techniques.

- **Chapter 7:** This chapter investigated the interpretability and efficiency of transformer-based Wi-Fi pose estimation models by analysing the importance of individual sub-carriers. Using attention-based rollout and input masking strategies, the most relevant sub-carriers were identified and used to retrain the model. The resulting models achieved accuracy comparable to the baseline while significantly improving computational efficiency. This work contributes to explainable artificial intelligence by providing insight into how transformer models exploit wireless signals for pose estimation and demonstrating that performance can be maintained using a reduced, more informative input space.

- **Chapter 8:** This chapter explored the use of video vision transformers for estimating human pose from Wi-Fi channel state information. A comprehensive hyperparameter sweep was conducted to identify suitable architectural configurations, alongside an analysis of key data-related factors such as sampling rate, temporal window size, and window step size. The results indicated that optimal configurations closely resemble those used in vision transformers, and that larger tubelet sizes, shorter temporal windows, smaller step sizes, and lower sampling frequencies lead to improved performance. These findings provide practical guidance for designing transformer-based models in the Wi-Fi domain.

- **Chapter 9:** This chapter examined the effect of noise-based data augmentation on Wi-Fi based pose estimation and action recognition models. By injecting noise drawn from normal and uniform distributions during training, the models were encouraged to learn more robust

representations. Experimental results showed that moderate levels of noise reduced overfitting and improved generalisation, while excessive noise led to information loss and degraded performance.

References

[1] M. Kulbacki et al., "Intelligent video analytics for human action recognition: The state of knowledge," *Sensors*, vol. 23, no. 9, 2023, ISSN: 1424-8220. DOI: 10.3390/s23094258 [Online]. Available: https://www.mdpi.com/1424-8220/23/9/4258

[2] P. Pareek and A. Thakkar, "A survey on video-based human action recognition: Recent updates, datasets, challenges, and applications," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 2259–2322, 2021.

[3] H. Duan, Y. Zhao, K. Chen, Y. Xiong, and D. Lin, "Mitigating representation bias in action recognition: Algorithms and benchmarks," in *European Conference on Computer Vision*, Springer, 2022, pp. 557–575.

[4] V. Somers, C. De Vleeschouwer, and A. Alahi, "Body part-based representation learning for occluded person re-identification," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 1613–1623.

[5] I. Ahmad, A. Ullah, and W. Choi, "WiFi-Based Human Sensing With Deep Learning: Recent Advances, Challenges, and Opportunities," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 3595–3623, 2024. DOI: 10.1109/OJCOMS.2024.3411529

[6] Y. Zhou, C. Xu, L. Zhao, A. Zhu, F. Hu, and Y. Li, "Csi-former: Pay more attention to pose estimation with wifi," *Entropy*, vol. 25, no. 1, p. 20, 2022.

[7] K. Yan, F. Wang, B. Qian, H. Ding, J. Han, and X. Wei, "Person-in-wifi 3d: End-to-end multi-person 3d pose estimation with wi-fi," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 969–978.

[8] J. Strohmayer and M. Kampel, "Data augmentation techniques for cross-domain wifi csi-based human activity recognition," in *IFIP International Conference on Artificial Intelligence Applications and Innovations*, Springer, 2024, pp. 42–56.

[9] W. Hou and C. Wu, "Rfboost: Understanding and boosting deep wifi sensing via physical data augmentation," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 8, no. 2, pp. 1–26, 2024.

[10] L. Capozzi, L. Ferreira, T. Gonçalves, A. Rebelo, J. S. Cardoso, and A. F. Sequeira, "Deciphering the silent signals: Unveiling frequency importance for wi-fi-based human pose estimation with explainability," in *Pattern Recognition and Image Analysis*, N. Gonçalves, H. P. Oliveira, and J. A. Sánchez, Eds., Cham: Springer Nature Switzerland, 2026, pp. 285–296, ISBN: 978-3-031-99568-2.

[11] L. Capozzi, J. S. Cardoso, and A. Rebelo, "End-to-end occluded person re-identification with artificial occlusion generation," *IEEE Access*, vol. 13, pp. 146 020–146 031, 2025. DOI: 10.1109/ACCESS.2025.3600385

[12] J. R. Pinto, P. Carvalho, C. Pinto, A. Sousa, L. G. Capozzi, and J. S. Cardoso, "Streamlining Action Recognition in Autonomous Shared Vehicles with an Audiovisual Cascade Strategy," in *17th International Conference on Computer Vision Theory and Applications (VISAPP)*, Online, Feb. 2022.

[13] L. Capozzi et al., "Toward vehicle occupant-invariant models for activity characterization," *IEEE Access*, vol. 10, pp. 104 215–104 225, 2022. DOI: 10.1109/ACCESS.2022.3210973

[14] E. Castro et al., "Fill in the blank for fashion complementary outfit product retrieval: Visum summer school competition," *Machine vision and applications*, vol. 34, no. 1, p. 16, 2023.

[15] L. Capozzi, P. Carvalho, A. Sousa, C. Pinto, J. R. Pinto, and J. S. Cardoso, "Impact of visual noise in activity recognition using deep neural networks - an experimental approach," in *2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning (PRML)*, 2021, pp. 331–337. DOI: 10.1109/PRML52754.2021.9520734

[16] L. Capozzi, J. R. Pinto, J. S. Cardoso, and A. Rebelo, "Optimizing person re-identification using generated attention masks," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, J. M. R. S. Tavares, J. P. Papa, and M. González Hidalgo, Eds., Cham: Springer International Publishing, 2021, pp. 248–257, ISBN: 978-3-030-93420-0.

[17] T. Gonçalves, L. Capozzi, A. Rebelo, and J. S. Cardoso, "Deep image segmentation based on mutual information: A study," in *30th Portuguese Conference in Pattern Recognition (RECPAD)*, Covilhã, Portugal, 2024.

[18] L. Capozzi, T. Gonçalves, J. S. Cardoso, and A. Rebelo, "Towards fingerprint presentation attack generation using generative adversarial networks," in *30th Portuguese Conference in Pattern Recognition (RECPAD)*, Covilhã, Portugal, 2024.

[19] T. Gonçalves, L. Capozzi, A. Rebelo, and J. S. Cardoso, "Optimisation of deep neural networks using a genetic algorithm: A comparative study," in *29th Portuguese Conference in Pattern Recognition (RECPAD)*, Coimbra, Portugal, 2023.

[20] L. Capozzi, T. Gonçalves, J. S. Cardoso, and A. Rebelo, "Analysis of neurons' information in deep spiking neural networks using information theory," in *29th Portuguese Conference in Pattern Recognition (RECPAD)*, Coimbra, Portugal, 2023.

[21] P. Augusto, J. S. Cardoso, and J. Fonseca, "Automotive interior sensing - towards a synergetic approach between anomaly detection and action recognition strategies," in *Fourth IEEE International Conference on Image Processing, Applications and Systems (IPAS 2020)*, Genova, Italy, Dec. 2020, pp. 162–167. DOI: 10.1109/IPAS50080.2020.9334943

[22] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," *CoRR*, vol. abs/1705.07750, 2017. arXiv: 1705.07750. [Online]. Available: http://arxiv.org/abs/1705.07750

[23] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "C3D: generic features for video analysis," *CoRR*, vol. abs/1412.0767, 2014. arXiv: 1412.0767. [Online]. Available: http://arxiv.org/abs/1412.0767

[24] M. E. Kalfaoglu, S. Kalkan, and A. A. Alatan, *Late temporal modeling in 3d cnn architectures with bert for action recognition*, 2020. arXiv: 2008.01232 [cs.CV].

[25] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "Multi-fiber networks for video recognition," *CoRR*, vol. abs/1807.11195, 2018. arXiv: 1807.11195. [Online]. Available: http://arxiv.org/abs/1807.11195

[26] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," *CoRR*, vol. abs/1812.03982, 2018. arXiv: 1812.03982. [Online]. Available: http://arxiv.org/abs/1812.03982

[27] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" *CoRR*, vol. abs/1711.09577, 2017. arXiv: 1711.09577. [Online]. Available: http://arxiv.org/abs/1711.09577

[28] A. J. Piergiovanni, A. Angelova, A. Toshev, and M. S. Ryoo, "Evolving space-time neural architectures for videos," *CoRR*, vol. abs/1811.10636, 2018. arXiv: 1811.10636. [Online]. Available: http://arxiv.org/abs/1811.10636

[29] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," *CoRR*, vol. abs/1904.02811, 2019. arXiv: 1904.02811. [Online]. Available: http://arxiv.org/abs/1904.02811

[30] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," *CoRR*, vol. abs/1711.11248, 2017. arXiv: 1711.11248. [Online]. Available: http://arxiv.org/abs/1711.11248

[31] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning for video understanding," *CoRR*, vol. abs/1712.04851, 2017. arXiv: 1712.04851. [Online]. Available: http://arxiv.org/abs/1712.04851

[32] J. R. Pinto et al., "Audiovisual classification of group emotion valence using activity recognition networks," in *Fourth IEEE International Conference on Image Processing, Applications and Systems (IPAS 2020)*, Genova, Italy, Dec. 2020, pp. 114–119. DOI: 10.1109/IPAS50080.2020.9334943

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 770–778.

[34] M. Monfort et al., "Moments in time dataset: One million videos for event understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–8, 2019, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2019.2901464

[35] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[36] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8. DOI: 10.1109/CVPR.2008.4587756

[37] M. Marszałek, I. Laptev, and C. Schmid, "Actions in context," in *IEEE Conference on Computer Vision & Pattern Recognition*, 2009.

[38] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1290–1297. DOI: 10.1109/CVPR.2012.6247813

[39] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. S. Mian, "Histogram of oriented principal components for cross-view action recognition," *CoRR*, vol. abs/1409.6813, 2014. arXiv: 1409.6813. [Online]. Available: http://arxiv.org/abs/1409.6813

[40] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, IEEE, 2012.

[41] G. Moon, J. Y. Chang, and K. M. Lee, "Camera distance-aware top-down approach for 3d multi-person pose estimation from a single RGB image," *CoRR*, vol. abs/1907.11346, 2019. arXiv: 1907.11346. [Online]. Available: http://arxiv.org/abs/1907.11346

[42] P. Ferreira, D. Pernes, A. Rebelo, and J. Cardoso, "Learning signer-invariant representations with adversarial training," Jan. 2020, p. 117. DOI: 10.1117/12.2559534

[43] I. Laptev, "On space-time interest points," *International journal of computer vision*, vol. 64, no. 2, pp. 107–123, 2005.

[44] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *CVPR 2011*, 2011, pp. 3169–3176. DOI: 10.1109/CVPR.2011.5995407

[45] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 3551–3558. DOI: 10.1109/ICCV.2013.441

[46] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., vol. 27, Curran Associates, Inc., 2014.

[47] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732. DOI: 10.1109/CVPR.2014.223

[48] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal residual networks for video action recognition," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16, Barcelona, Spain: Curran Associates Inc., 2016, pp. 3476–3484, ISBN: 9781510838819.

[49] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal multiplier networks for video action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7445–7454. DOI: 10.1109/CVPR.2017.787

[50] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4694–4702. DOI: 10.1109/CVPR.2015.7299101

[51] G. Chéron, I. Laptev, and C. Schmid, "P-cnn: Pose-based cnn features for action recognition," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3218–3226. DOI: 10.1109/ICCV.2015.368

[52] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013. DOI: 10.1109/TPAMI.2012.59

[53] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4489–4497. DOI: 10.1109/ICCV.2015.510

[54] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6546–6555. DOI: 10.1109/CVPR.2018.00685

[55] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4724–4733. DOI: 10.1109/CVPR.2017.502

[56] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., Cham: Springer International Publishing, 2018, pp. 318–335, ISBN: 978-3-030-01267-0.

[57] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5534–5542. DOI: 10.1109/ICCV.2017.590

[58] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459. DOI: 10.1109/CVPR.2018.00675

[59] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6201–6210. DOI: 10.1109/ICCV.2019.00630

[60] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015, pp. 579–583. DOI: 10.1109/ACPR.2015.7486569

[61] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3d action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4570–4579. DOI: 10.1109/CVPR.2017.486

[62] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1963–1978, 2019. DOI: 10.1109/TPAMI.2019.2896631

[63] H. Duan, Y. Zhao, K. Chen, D. Shao, D. Lin, and B. Dai, *Revisiting skeleton-based action recognition*, 2021. arXiv: 2104.13586 [cs.CV].

[64] D. K. Duvenaud et al., "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[65] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[66] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 140–149. DOI: 10.1109/CVPR42600.2020.00022

[67] Y. Li, Y. Li, and N. Vasconcelos, "Resound: Towards action recognition without representation bias," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018.

[68] Y. Li and N. Vasconcelos, "Repair: Removing representation bias by dataset resampling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019.

[69] J. Choi, C. Gao, J. C. E. Messou, and J.-B. Huang, "Why can't i dance in the mall? learning to mitigate scene bias in action recognition," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019.

[70] D.-A. Huang et al., "What makes a video a video: Analyzing temporal information in video understanding models and datasets," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7366–7375. DOI: 10.1109/CVPR.2018.00769

[71] W. Kay et al., "The kinetics human action video dataset," *CoRR*, vol. abs/1705.06950, 2017. arXiv: 1705.06950. [Online]. Available: http://arxiv.org/abs/1705.06950

[72] M. Monfort et al., "Moments in time dataset: One million videos for event understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 502–508, 2020. DOI: 10.1109/TPAMI.2019.2901464

[73] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 961–970. DOI: 10.1109/CVPR.2015.7298698

[74] K. Soomro, A. R. Zamir, and M. Shah, *Ucf101: A dataset of 101 human actions classes from videos in the wild*, 2012. arXiv: 1212.0402 [cs.CV].

[75] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: A large video database for human motion recognition," in *2011 International Conference on Computer Vision*, 2011, pp. 2556–2563. DOI: 10.1109/ICCV.2011.6126543

[76] R. Goyal et al., "The "something something" video database for learning and evaluating visual common sense," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5843–5851. DOI: 10.1109/ICCV.2017.622

[77] J. Materzynska, G. Berger, I. Bax, and R. Memisevic, "The jester dataset: A large-scale video dataset of human gestures," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 2874–2882. DOI: 10.1109/ICCVW.2019.00349

[78] P. Augusto, J. S. Cardoso, and J. Fonseca, "Automotive interior sensing - towards a synergetic approach between anomaly detection and action recognition strategies," in *2020 IEEE 4th International Conference on Image Processing, Applications and Systems (IPAS)*, Genova, Italy, 2020, pp. 162–167. DOI: http://dx.doi.org/10.1109/IPAS50080.2020.9334942

[79] M. Pinto, "Automotive interior sensing: Human interaction recognition," M.S. thesis, Faculdade de Engenharia da Universidade do Porto, 2021.

[80] S. Jenni and P. Favaro, "Deep bilevel learning," *CoRR*, vol. abs/1809.01465, 2018. arXiv: 1809.01465. [Online]. Available: http://arxiv.org/abs/1809.01465

[81] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "Potion: Pose motion representation for action recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7024–7033. DOI: 10.1109/CVPR.2018.00734

[82] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," *CoRR*, vol. abs/1703.06870, 2017. arXiv: 1703.06870. [Online]. Available: http://arxiv.org/abs/1703.06870

[83] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *IEEE Conference on Computer Vision & Pattern Recognition*, 2008.

[84] Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith, "Learning locally-adaptive decision functions for person verification," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3610–3617. DOI: 10.1109/CVPR.2013.463

[85] S. Liao, Y. Hu, Xiangyu Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2197–2206. DOI: 10.1109/CVPR.2015.7298832

[86] A. J. Ma, P. C. Yuen, and J. Li, "Domain transfer support vector ranking for person re-identification without target camera label information," in *2013 IEEE International Conference on Computer Vision*, 2013, pp. 3567–3574. DOI: 10.1109/ICCV.2013.443

[87] S. Pedagadi, J. Orwell, S. Velastin, and B. Boghossian, "Local fisher discriminant analysis for pedestrian re-identification," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3318–3325. DOI: 10.1109/CVPR.2013.426

[88] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, "Salient color names for person re-identification," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 536–551, ISBN: 978-3-319-10590-1.

[89] W. Zheng, S. Gong, and T. Xiang, "Reidentification by relative distance comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 653–668, 2013. DOI: 10.1109/TPAMI.2012.138

[90] Z. Dai, M. Chen, S. Zhu, and P. Tan, "Batch feature erasing for person re-identification and beyond," *CoRR*, vol. abs/1811.07130, 2018. arXiv: 1811.07130.

[91] R. Quispe and H. Pedrini, "Top-db-net: Top dropblock for activation enhancement in person re-identification," *arXiv preprint arXiv:2010.05435*, 2020.

[92] Y. Shen, H. Li, T. Xiao, S. Yi, D. Chen, and X. Wang, "Deep group-shuffling random walk for person re-identification," *CoRR*, vol. abs/1807.11178, 2018. arXiv: 1807.11178.

[93] Y. Sun, L. Zheng, W. Deng, and S. Wang, "Svdnet for pedestrian retrieval," *CoRR*, vol. abs/1703.05693, 2017. arXiv: 1703.05693.

[94] L. Zhao, X. Li, J. Wang, and Y. Zhuang, "Deeply-learned part-aligned representations for person re-identification," *CoRR*, vol. abs/1707.07256, 2017. arXiv: 1707.07256.

[95] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018.

[96] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *CoRR*, vol. abs/1703.07737, 2017. arXiv: 1703.07737.

[97] W. Li, X. Zhu, and S. Gong, "Harmonious attention network for person re-identification," *CoRR*, vol. abs/1802.08122, 2018. arXiv: 1802.08122.

[98] J. Si et al., "Dual attention matching network for context-aware feature sequence based person re-identification," *CoRR*, vol. abs/1803.09937, 2018. arXiv: 1803.09937.

[99] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. arXiv: 1512.03385.

[100] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[101] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015. arXiv: 1505.04597.

[102] I. Rio-Torto, K. Fernandes, and L. F. Teixeira, "Understanding the decisions of cnns: An in-model approach," *Pattern Recognition Letters*, vol. 133, pp. 373–380, 2020, ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2020.04.004 [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865520301240

[103] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1116–1124. DOI: 10.1109/ICCV.2015.133

[104] E. Ristani, F. Solera, R. S. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," *CoRR*, vol. abs/1609.01775, 2016. arXiv: 1609.01775. [Online]. Available: http://arxiv.org/abs/1609.01775

[105] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by GAN improve the person re-identification baseline in vitro," *CoRR*, vol. abs/1701.07717, 2017. arXiv: 1701.07717. [Online]. Available: http://arxiv.org/abs/1701.07717

[106] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 152–159. DOI: 10.1109/CVPR.2014.27

[107] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1318–1327.

[108] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," *CoRR*, vol. abs/1610.02984, 2016. arXiv: 1610.02984.

[109] Z. Zheng, L. Zheng, and Y. Yang, "Pedestrian alignment network for large-scale person re-identification," *CoRR*, vol. abs/1707.00408, 2017. arXiv: 1707.00408.

[110] Y. Chen, X. Zhu, and S. Gong, "Person re-identification by deep learning multi-scale representations," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017, pp. 2590–2600. DOI: 10.1109/ICCVW.2017.304

[111] N. Martinel, G. L. Foresti, and C. Micheloni, "Aggregating deep pyramidal representations for person re-identification," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019, pp. 1544–1554. DOI: 10.1109/CVPRW.2019.00196

[112] R. Quan, X. Dong, Y. Wu, L. Zhu, and Y. Yang, "Auto-reid: Searching for a part-aware convnet for person re-identification," *CoRR*, vol. abs/1903.09776, 2019. arXiv: 1903.09776. [Online]. Available: http://arxiv.org/abs/1903.09776

[113] G. Wang, Y. Yuan, X. Chen, J. Li, and X. Zhou, "Learning discriminative features with multiple granularities for person re-identification," *CoRR*, vol. abs/1804.01438, 2018. arXiv: 1804.01438. [Online]. Available: http://arxiv.org/abs/1804.01438

[114] Z. Zhang, C. Lan, W. Zeng, and Z. Chen, "Densely semantically aligned person re-identification," *CoRR*, vol. abs/1812.08967, 2018. arXiv: 1812.08967. [Online]. Available: http://arxiv.org/abs/1812.08967

[115] B. Chen, W. Deng, and J. Hu, "Mixed high-order attention network for person re-identification," *CoRR*, vol. abs/1908.05819, 2019. arXiv: 1908.05819. [Online]. Available: http://arxiv.org/abs/1908.05819

[116] T. Chen et al., "Abd-net: Attentive but diverse person re-identification," *CoRR*, vol. abs/1908.01114, 2019. arXiv: 1908.01114. [Online]. Available: http://arxiv.org/abs/1908.01114

[117] B. (Xia, Y. Gong, Y. Zhang, and C. Poellabauer, "Second-order non-local attention networks for person re-identification," *CoRR*, vol. abs/1909.00295, 2019. arXiv: 1909.00295. [Online]. Available: http://arxiv.org/abs/1909.00295

[118] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Omni-scale feature learning for person re-identification," *CoRR*, vol. abs/1905.00953, 2019. arXiv: 1905.00953. [Online]. Available: http://arxiv.org/abs/1905.00953

[119] F. Zheng, X. Sun, X. Jiang, X. Guo, Z. Yu, and F. Huang, "A coarse-to-fine pyramidal model for person re-identification via multi-loss dynamic training," *CoRR*, vol. abs/1810.12193, 2018. arXiv: 1810.12193. [Online]. Available: http://arxiv.org/abs/1810.12193

[120] T. Zhang, C. Xu, and M.-H. Yang, "Robust structural sparse tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 473–486, 2019. DOI: 10.1109/TPAMI.2018.2797082

[121] T. Zhang, C. Xu, and M.-H. Yang, "Learning multi-task correlation particle filters for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 365–378, 2019. DOI: 10.1109/TPAMI.2018.2797062

[122] Y. Li, J. He, T. Zhang, X. Liu, Y. Zhang, and F. Wu, "Diverse part discovery: Occluded person re-identification with part-aware transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2898–2907.

[123] M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2288–2295. DOI: 10.1109/CVPR.2012.6247939

[124] S. Liao and S. Z. Li, "Efficient psd constrained asymmetric metric learning for person re-identification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3685–3693. DOI: 10.1109/ICCV.2015.420

[125] H. Dong, P. Lu, S. Zhong, C. Liu, Y. Ji, and S. Gong, "Person re-identification by enhanced local maximal occurrence representation and generalized similarity metric learning," *Neurocomputing*, vol. 307, pp. 25–37, 2018.

[126] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang, "Bag of tricks and A strong baseline for deep person re-identification," *CoRR*, vol. abs/1903.07071, 2019. arXiv: 1903.07071. [Online]. Available: http://arxiv.org/abs/1903.07071

[127] Z. Wang, L. He, X. Gao, and Y. Huang, "Multi-scale spatial-temporal network for person re-identification," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 2052–2056. DOI: 10.1109/ICASSP.2019.8683716

[128] Z. Wang et al., "Robust video-based person re-identification by hierarchical mining," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 12, pp. 8179–8191, 2022. DOI: 10.1109/TCSVT.2021.3076097

[129] R. Quispe and H. Pedrini, "Top-db-net: Top dropblock for activation enhancement in person re-identification," in *2020 25th International conference on pattern recognition (ICPR)*, IEEE, 2021, pp. 2980–2987.

[130] S. Gao, J. Wang, H. Lu, and Z. Liu, "Pose-guided visible part matching for occluded person reid," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 744–11 752.

[131] J. Miao, Y. Wu, P. Liu, Y. Ding, and Y. Yang, "Pose-guided feature alignment for occluded person re-identification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019.

[132] G. Wang et al., "High-order information matters: Learning relation and topology for occluded person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6449–6458.

[133] H. Tan, X. Liu, B. Yin, and X. Li, "Mhsa-net: Multihead self-attention network for occluded person re-identification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 11, pp. 8210–8224, 2022.

[134] H. Huang, X. Chen, and K. Huang, "Human parsing based alignment with multi-task learning for occluded person re-identification," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6. DOI: 10.1109/ICME46284.2020.9102789

[135] L. He, Y. Wang, W. Liu, H. Zhao, Z. Sun, and J. Feng, "Foreground-aware pyramid reconstruction for alignment-free occluded person re-identification," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 8450–8459.

[136] Z. Wang, F. Zhu, S. Tang, R. Zhao, L. He, and J. Song, "Feature erasing and diffusion network for occluded person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 4754–4763.

[137] P. Chen et al., "Occlude them all: Occlusion-aware attention network for occluded person re-id," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 11 813–11 822. DOI: 10.1109/ICCV48922.2021.01162

[138] B. Ma, Y. Su, and F. Jurie, "Covariance descriptor based on bio-inspired features for person re-identification and face verification," *Image and Vision Computing*, vol. 32, no. 6-7, pp. 379–390, 2014.

[139] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 403–412.

[140] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1318–1327.

[141] L. He, J. Liang, H. Li, and Z. Sun, "Deep spatial feature reconstruction for partial person re-identification: Alignment-free approach," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7073–7082.

[142] X. Liu et al., "Hydraplus-net: Attentive deep features for pedestrian analysis," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 350–359.

[143] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 480–496.

[144] M. S. Sarfraz, A. Schumann, A. Eberle, and R. Stiefelhagen, "A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 420–429.

[145] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Pose-driven deep convolutional model for person re-identification," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3960–3969.

[146] L. Zheng, Y. Huang, H. Lu, and Y. Yang, "Pose-invariant embedding for deep person re-identification," *IEEE transactions on image processing*, vol. 28, no. 9, pp. 4500–4509, 2019.

[147] J. Liu, B. Ni, Y. Yan, P. Zhou, S. Cheng, and J. Hu, "Pose transferrable person re-identification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4099–4108. DOI: 10.1109/CVPR.2018.00431

[148] J. Xu, R. Zhao, F. Zhu, H. Wang, and W. Ouyang, "Attention-aware compositional network for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2119–2128.

[149] M. M. Kalayeh, E. Basaran, M. Gökmen, M. E. Kamasak, and M. Shah, "Human semantic parsing for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1062–1071.

[150] C. Song, Y. Huang, W. Ouyang, and L. Wang, "Mask-guided contrastive attention model for person re-identification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1179–1188. DOI: 10.1109/CVPR.2018.00129

[151] Z. Zhang, C. Lan, W. Zeng, X. Jin, and Z. Chen, "Relation-aware global attention for person re-identification," in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 2020, pp. 3186–3195.

[152] Z. Wang, L. He, X. Gao, and J. Shen, "Robust person re-identification through contextual mutual boosting," *arXiv preprint arXiv:2009.07491*, 2020.

[153] L. Wei, S. Zhang, H. Yao, W. Gao, and Q. Tian, "Glad: Global-local-alignment descriptor for pedestrian retrieval," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 420–428.

[154] H. Zhao et al., "Spindle net: Person re-identification with human body region guided feature decomposition and fusion," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 907–915. DOI: 10.1109/CVPR.2017.103

[155] C.-P. Tay, S. Roy, and K.-H. Yap, "Aanet: Attribute attention network for person re-identifications," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7134–7143.

[156] J. Zhuo, Z. Chen, J. Lai, and G. Wang, "Occluded person re-identification," in *2018 IEEE international conference on multimedia and expo (ICME)*, IEEE, 2018, pp. 1–6.

[157] J. Yang et al., "Learning to know where to see: A visibility-aware approach for occluded person re-identification," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 11 865–11 874. DOI: 10.1109/ICCV48922.2021.01167

[158] C. Yan, G. Pang, J. Jiao, X. Bai, X. Feng, and C. Shen, "Occluded person re-identification with single-scale global representations," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 11 855–11 864. DOI: 10.1109/ICCV48922.2021.01166

[159] A. Howard et al., "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.

[160] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[161] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1335–1344. DOI: 10.1109/CVPR.2016.149

[162] W. Zheng, X. Li, T. Xiang, S. Liao, J. Lai, and S. Gong, "Partial person re-identification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4678–4686. DOI: 10.1109/ICCV.2015.531

[163] W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *CVPR 2011*, 2011, pp. 649–656. DOI: 10.1109/CVPR.2011.5995598

[164] Y. Sun et al., "Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 393–402.

[165] R. Hou, B. Ma, H. Chang, X. Gu, S. Shan, and X. Chen, "Interaction-and-aggregation network for person re-identification," *CoRR*, vol. abs/1907.08435, 2019. arXiv: 1907.08435. [Online]. Available: http://arxiv.org/abs/1907.08435

[166] W. Yang, H. Huang, Z. Zhang, X. Chen, K. Huang, and S. Zhang, "Towards rich feature discovery with class activation maps augmentation for person re-identification," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1389–1398. DOI: 10.1109/CVPR.2019.00148

[167] X. Qian, Y. Fu, T. Xiang, Y.-G. Jiang, and X. Xue, "Leader-based multi-scale attention deep architecture for person re-identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 371–385, 2020. DOI: 10.1109/TPAMI.2019.2928294

[168] L. He, Z. Sun, Y. Zhu, and Y. Wang, "Recognizing partial biometric patterns," *arXiv preprint arXiv:1810.07399*, 2018.

[169] T. Ruan, T. Liu, Z. Huang, Y. Wei, S. Wei, and Y. Zhao, "Devil in the details: Towards accurate single and multiple human parsing," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 4814–4821.

[170] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, 2020, pp. 13 001–13 008.

[171] H. Huang, D. Li, Z. Zhang, X. Chen, and K. Huang, "Adversarially occluded samples for person re-identification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5098–5107. DOI: 10.1109/CVPR.2018.00535

[172] L. He and W. Liu, "Guided saliency feature learning for person re-identification in crowded scenes," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Cham: Springer International Publishing, 2020, pp. 357–373, ISBN: 978-3-030-58604-1.

[173] K. Zhu, H. Guo, Z. Liu, M. Tang, and J. Wang, "Identity-guided human semantic parsing for person re-identification," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, Springer, 2020, pp. 346–363.

[174] T. Zhou, Y. Yang, and W. Wang, "Differentiable Multi-Granularity Human Parsing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 7, pp. 8296–8310, 2023. DOI: 10.1109/TPAMI.2023.3239194

[175] R. A. Guler, N. Neverova, and I. Kokkinos, "DensePose: Dense Human Pose Estimation in the Wild," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2018, pp. 7297–7306. DOI: 10.1109/CVPR.2018.00762

[176] J. Yang et al., "SenseFi: A library and benchmark on deep-learning-empowered WiFi human sensing," *Patterns*, vol. 4, no. 3, p. 100 703, 2023, ISSN: 2666-3899. DOI: https://doi.org/10.1016/j.patter.2023.100703

[177] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine Learning Interpretability: A Survey on Methods and Metrics," *Electronics*, vol. 8, no. 8, 2019, ISSN: 2079-9292. DOI: 10.3390/electronics8080832 [Online]. Available: https://www.mdpi.com/2079-9292/8/8/832

[178] F. Wang, S. Zhou, S. Panev, J. Han, and D. Huang, "Person-in-wifi: Fine-grained person perception using wifi," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5452–5461.

[179] L. Guo, Z. Lu, X. Wen, S. Zhou, and Z. Han, "From Signal to Image: Capturing Fine-Grained Human Poses With Commodity Wi-Fi," *IEEE Communications Letters*, vol. 24, no. 4, pp. 802–806, 2020. DOI: 10.1109/LCOMM.2019.2961890

[180] W. Jiang et al., "Towards 3D Human Pose Construction Using WiFi," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '20, London, United Kingdom: Association for Computing Machinery, 2020, ISBN: 9781450370851. DOI: 10.1145/3372224.3380900

[181] C. Yu et al., "WiFi-Based Human Pose Image Generation," in *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, 2022, pp. 1–6. DOI: 10.1109/MMSP55362.2022.9949562

[182] J. Geng, D. Huang, and F. D. la Torre, *DensePose From WiFi*, 2022. arXiv: 2301.00250 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2301.00250

[183] Y. Ren, Z. Wang, Y. Wang, S. Tan, Y. Chen, and J. Yang, "Gopose: 3d human pose estimation using wifi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 2, pp. 1–25, 2022.

[184] Y. Zhou, C. Xu, L. Zhao, A. Zhu, F. Hu, and Y. Li, "CSI-Former: Pay More Attention to Pose Estimation with WiFi," *Entropy*, vol. 25, no. 1, 2023, ISSN: 1099-4300. DOI: 10.3390/e25010020

[185] Y. Zhou, A. Zhu, C. Xu, F. Hu, and Y. Li, "PerUnet: Deep Signal Channel Attention in Unet for WiFi-Based Human Pose Estimation," *IEEE Sensors Journal*, vol. 22, no. 20, pp. 19 750–19 760, 2022. DOI: 10.1109/JSEN.2022.3204607

[186] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional Multi-person Pose Estimation," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2353–2362. DOI: 10.1109/ICCV.2017.256

[187] Y. Xiu, J. Li, H. Wang, Y. Fang, and C. Lu, "Pose Flow: Efficient Online Pose Tracking," in *BMVC*, 2018.

[188] K. Yan, F. Wang, B. Qian, H. Ding, J. Han, and X. Wei, "Person-in-WiFi 3D: End-to-End Multi-Person 3D Pose Estimation with Wi-Fi," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2024, pp. 969–978.

[189] T. D. Gian, T. Dac Lai, T. Van Luong, K.-S. Wong, and V.-D. Nguyen, "HPE-Li: WiFi-Enabled Lightweight Dual Selective Kernel Convolution for Human Pose Estimation," in *Computer Vision – ECCV 2024*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., Cham: Springer Nature Switzerland, 2025, 93–111, ISBN: 978-3-031-72751-1.

[190] J. R. Pinto and J. S. Cardoso, "Explaining ECG Biometrics: Is It All In The QRS?" In *2020 International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2020, pp. 1–5.

[191] A. Anand, T. Kadian, M. K. Shetty, and A. Gupta, "Explainable AI decision model for ECG data of cardiac disorders," *Biomedical Signal Processing and Control*, vol. 75, p. 103 584, 2022, ISSN: 1746-8094. DOI: https://doi.org/10.1016/j.bspc.2022.103584 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809422001069

[192] B. Majhi and A. Kashyap, "Explainable AI-driven machine learning for heart disease detection using ECG signal," *Applied Soft Computing*, vol. 167, p. 112 225, 2024, ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2024.112225 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494624009992

[193] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

[194] H.-Y. Chen and C.-H. Lee, "Vibration Signals Analysis by Explainable Artificial Intelligence (XAI) Approach: Application on Bearing Faults Diagnosis," *IEEE Access*, vol. 8, pp. 134 246–134 256, 2020. DOI: 10.1109/ACCESS.2020.3006491

[195] A. Elango and R. J. Landry, "XAI GNSS—A Comprehensive Study on Signal Quality Assessment of GNSS Disruptions Using Explainable AI Technique," *Sensors*, vol. 24, no. 24, 2024, ISSN: 1424-8220. [Online]. Available: https://www.mdpi.com/1424-8220/24/24/8039

[196] A. K. Gizzini, Y. Medjahdi, A. J. Ghandour, and L. Clavier, "Towards Explainable AI for Channel Estimation in Wireless Communications," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 7389–7394, 2024. DOI: 10.1109/TVT.2023.3345632

[197] A. Akman and B. W. Schuller, "Audio Explainable Artificial Intelligence: A Review," *Intelligent Computing*, vol. 3, p. 0074, 2024. DOI: 10.34133/icomputing.0074

[198] H.-S. Fang et al., "Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 6, pp. 7157–7173, 2022.

[199] A. Vaswani et al., "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 6000–6010, ISBN: 9781510860964.

[200] S. Abnar and W. Zuidema, "Quantifying Attention Flow in Transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 4190–4197. DOI: 10.18653/v1/2020.acl-main.385 [Online]. Available: https://aclanthology.org/2020.acl-main.385/

[201] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6836–6846.

[202] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *International Conference on Learning Representations (ICLR)*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[203] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, Springer, 2016, pp. 483–499.

[204] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[205] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5693–5703.

[206] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[207] 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, 2016.

[208] F. Wang, S. Panev, Z. Dai, J. Han, and D. Huang, "Can wifi estimate person pose?" *arXiv preprint arXiv:1904.00277*, 2019.

[209] W. Jiang et al., "Towards 3d human pose construction using wifi," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[210] Y. Ren, Z. Wang, S. Tan, Y. Chen, and J. Yang, "Winect: 3d human pose tracking for free-form activity using commodity wifi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 4, pp. 1–29, 2021.

[211] L. Guo, Z. Lu, X. Wen, S. Zhou, and Z. Han, "From signal to image: Capturing fine-grained human poses with commodity wi-fi," *IEEE Communications Letters*, vol. 24, no. 4, pp. 802–806, 2019.

[212] C. Yu et al., "Wifi-based human pose image generation," in *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, IEEE, 2022, pp. 1–6.

[213] J. Geng, D. Huang, and F. De la Torre, "Densepose from wifi," *arXiv preprint arXiv:2301.00250*, 2022.

[214] T. D. Gian, T. Dac Lai, T. Van Luong, K.-S. Wong, and V.-D. Nguyen, "Hpe-li: Wifi-enabled lightweight dual selective kernel convolution for human pose estimation," in *European Conference on Computer Vision*, Springer, 2024, pp. 93–111.

[215] H. Mo and S. Kim, "A deep learning-based human identification system with wi-fi csi data augmentation," *IEEE Access*, vol. 9, pp. 91 913–91 920, 2021. DOI: 10.1109/ACCESS.2021.3092435

[216] J. Strohmayer and M. Kampel, "Data augmentation techniques for cross-domain wifi csi-based human activity recognition," in *Artificial Intelligence Applications and Innovations*. Springer Nature Switzerland, 2024, pp. 42–56, ISBN: 9783031632112. DOI: 10.1007/978-3-031-63211-2_4 [Online]. Available: http://dx.doi.org/10.1007/978-3-031-63211-2_4

[217] W. Hou and C. Wu, "Rfboost: Understanding and boosting deep wifi sensing via physical data augmentation," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 8, no. 2, May 2024. DOI: 10.1145/3659620 [Online]. Available: https://doi.org/10.1145/3659620

[218] J. Wen et al., "Generative ai for data augmentation in wireless networks: Analysis, applications, and case study," *IEEE Wireless Communications*, pp. 1–10, 2025. DOI: 10.1109/MWC.2025.3610574