



University of Porto
Faculty of Engineering

Automatic Assessment of Equestrian Pain

Maria Francisca Pessanha de Meneses Ribeiro dos Reis
MSc in Bioengineering

Supervisor: Prof. Dr. Albert Ali Salah (University of Utrecht)

Co-supervisor: Prof. Dr. Jaime Cardoso (University of Porto)

June 2020

Contact information:

Francisca Pessanha

email: up201405995@fe.up.pt

© University of Porto. All rights reserved.

This document was typeset with $\text{\LaTeX}$ using the CVR thesis template.

Abstract

Recognition of pain in equines is essential for their welfare. However, since there is no verbal communication, this assessment depends solely on the ability of the observer to locate visible signs of pain. The use of grimace scales is proven to be efficient in detecting pain but is time-consuming. Also, it depends on the level of training of the annotators and, therefore, validity is not easily ensured. So, there is a clear advantage to automating this pain assessment process. This work provides a system for pain prediction in horses, based on grimace scales. The pipeline automatically determines the quantitative pose of the horse head and finds landmarks on horse faces before classification. Considering the scarcity of animal faces datasets, already widely available for humans, a data augmentation method is proposed, focusing on generating realistic 3D models based on 2D annotated images. Additionally, a pain estimation model is introduced, assessing the pain score for each facial region-of-interest. In general, the data augmentation method improved the performance of both quantitative pose estimator and landmark detector, showing the potential of this methodology for data augmentation in diverse datasets. The pain estimation system overcame the baseline (a majority vote classifier), but the unbalanced in the pain levels represented in the dataset will have a high impact on the results.

Acknowledgements

I would like to extend my gratitude to all the professors that taught me how to do outstanding research, with a special thank you to Prof. Dr. Albert Ali Salah and Prof. Dr. Jaime Cardoso for the support into developing this dissertation, even if kilometres apart.

Thank you to my friends all over the world for keeping my spirits up during this time and to all the new friends that made the Netherlands feel like home so quickly. Most of all, thank you to my mother for all the support and love over the years, my "Cambridge family" for making me the best version of myself and my adventure buddy for making me leave my comfort zone. I couldn't have reached this far without you all.

"We have to speak up on behalf of those who cannot speak for themselves."

– Peter Singer, Animal Liberation

Contents

Abstract	i
Acknowledgements	iii
Contents	vii
List of figures	ix
List of tables	xi
Nomenclature	xiii
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	2
1.3 Outline of the Dissertation	2
1.4 Contributions	2
2 Face-based Pain Assessment	5
2.1 Pain Assessment in Humans	5
2.2 Pain Assessment in Equines	5
2.3 Automatic Pain Assessment in Animals	10
3 Automatic Landmark Detection	13
3.1 Automatic Landmark Detection in Humans	13
3.2 Holistic methods	14
3.3 Constrained Local Models	20
3.4 Regression-based methods	22

3.4.1	Direct Regression Methods	22
3.4.2	Cascaded Regression Methods	23
3.4.3	Deep-Learning Based Methods	27
3.5	Face Alignment in Animals	29
4	Data Augmentation	33
4.1	Camera Models	33
4.2	3D models based on 2D images	37
4.2.1	Proposed Method	38
5	Experimental Evaluation	45
5.1	The Horse and Donkey Faces Dataset	45
5.1.1	Landmarks annotations	46
5.1.2	Pose distribution	47
5.1.3	Pain annotations distribution	48
5.2	Pose Estimation	49
5.2.1	Methods	49
5.2.2	Experimental Results	50
5.3	Landmarking	52
5.3.1	Approach	52
5.3.2	Experimental Results	54
5.4	Pain Estimation	58
5.4.1	Methods	58
5.4.2	Results	59
6	Conclusions and Future Work	63
	Appendices	65
	Appendix A What about donkeys?	67
A.1	Results	67
	Bibliography	69

List of Figures

2.1	Horse Grimace Pain Scale (HGS).	7
2.2	Example images of the pain score sheet used in the present work.	9
2.3	Setups used to capture continuous video footage of mice for pain assessment.	10
2.4	Pipeline for automatic pain level estimation in sheep.	11
3.1	Active Shape Model fitting along a sampled profile to find the strongest edge	15
3.2	Active Shape Model fitting along a sampled profile to find the best fit of gray-level model.	15
3.3	Multi-resolution implementation of the Active Shape Model.	16
3.4	Landmark detection using an Active Shape Model.	17
3.5	Example of the variables extracted from a training image by the Active Appearance Model.	17
3.6	Effect of varying the first four facial appearance model parameter in the Appearance Active Model.	19
3.7	Multi-resolution search from displaced position using Active Appearance Model.	19
3.8	Joint Modes of Shape and Texture Variation in Constrained Local Models.	20
3.9	Constrained Local Model search algorithm.	21
3.10	Conditional Regression Forest for Landmark Detection.	24
3.11	Explicit Shape Regression.	25
3.12	Supervised Descent Model.	27
3.13	Cascade Convolutional network for Landmark Detection.	28
3.14	3D Dense Face Alignment.	28
3.15	Qualitative results produced by CALE on our CatsDogs dataset.	29

3.16	Qualitative examples of landmarks localisation improvements made by PI-ERT.	30
3.17	Network Architecture for animal facial keypoint detection.	30
4.1	Pinhole camera geometry.	33
4.2	The Euclidean transformation between the world and camera coordinate frames.	35
4.3	Camera model depth estimation.	37
4.4	Images showing eight dolphins from which an 8-parameter morphable model was built.	38
4.5	Full horse and head model with respective axis.	38
4.6	Examples of deformation and further texture transfer of the 3D model. . .	41
4.7	Examples of synthetic images produced using the described method. . .	43
5.1	Proposed pipeline for horse pain estimation based on facial features . . .	45
5.2	Faces extracted from each subset with marked points of interest.	46
5.3	Reduced landmarking system.	47
5.4	Quantitative head pose axis in horses.	47
5.5	Distribution of the pose values in the updated dataset.	48
5.6	Distribution of pain scores on the dataset.	49
5.7	Visual representation of the error in pose estimation.	52
5.8	Illustration of the eye-nostril distance used for normalization.	53
5.9	Landmark location results using ERT and a baseline mean shape model. . .	55
5.10	Average MNE distribution for <i>roll</i> , <i>yaw</i> , <i>pitch</i> values, respectively	57
5.11	Normalisation of the regions-of-interest for the "frontal" class.	59
5.12	Agreement in the annotations made by three specialists in 1655 images of horse faces.	61

List of Tables

2.1	EQUUS-FAP Score sheet.	6
2.2	Score sheet for facial pain score assessment in still images, adapted from the EQUUS-FAP and HGS.	8
5.1	Pose estimation in Horses.	50
5.2	Quantitative pose estimation results in the test set transfer learning from the model trained on the 300W-LP dataset.	51
5.3	Quantitative pose estimation results in the test set transfer learning from the model trained on the Sheep dataset.	51
5.4	Quantitative pose estimation results in the test set transfer learning from the model trained on the Sheep dataset with an augmented training set ($\alpha = 0.5$).	52
5.5	Mean Normalized Error and Success Rate in Horses using ERT, SDM and a baseline mean shape model.	54
5.6	Mean Normalized Error per region-of-interest in Horses.	54
5.7	Mean Normalized Error in the reduced landmarking system in the test set.	56
5.8	Mean Normalized Error per region-of-interest (ROI) in the test set with the reduced landmarking system.	56
5.9	Mean Normalized Error per region-of-interest (ROI) in the test set after data augmentation.	58
5.10	Performance of the pain estimation models considering three classes.	59
5.11	Performance of the binary pain estimation models.	60
A.1	Pose estimation in Donkeys.	67
A.2	Mean landmark location error in Donkeys.	68

Nomenclature

Holistic and Constrained Local Models

(s_x, s_y)	Scaling
(t_x, t_y)	Translation
$\bar{g}$	Mean texture vector in the model frame
$\bar{x}$	Mean shape in the model frame
θ	Rotation
$g = \{g_0, \dots, g_N\}$	Set of N texture vectors in the model frame
G	Texture example in the image frame
g_m	Texture model frame
g_s	Texture example in the model frame
T	Set of rigid shape transformations - translation, rotation and scaling
T_u	Set of texture transformations - scaling and intensity offsets
X	Shape example with n ordered landmarks in the image frame
x	Shape with n ordered landmarks in the model frame

Regression Models

$\hat{S} = \{\hat{S}_0, \dots, \hat{S}_N\}$	Set of N ground truth shapes
$\mathcal{P} = \{P_0, \dots, P_i\}$	Set of i patches extracted from a single image
S_i^t	Shape in the iteration t for the example i in the set S
$x^* = \{x_0^*, \dots, x_N^*\}$	Set of N ground truth shapes with n ordered landmarks

Chapter 1

Introduction

1.1 Motivation

The recognition and quantification of pain in equines is essential to maintain their welfare and improve their convalescence (1). However, contrary to humans, where pain assessment is facilitated through verbal communication, in animals, this process depends on the observer's ability to locate and quantify the pain, based on perceptible behaviour and physiological patterns.

Several studies have found a correlation between pain and behaviour changes in equines, such as aggressiveness, reluctance to move, vocalization and diminished socialisation (2). However, to study more subtle changes, it is useful to analyze the facial expressions of these animals (3). This method has been extensively used in other species, such as mice (4), rabbits (5) and sheep (6) with promising results. Several frameworks have been proposed for horse pain estimation, the most important being the Horse Grimace Scale (HGS) (7) and the Equine Utrecht University Scale for Facial Assessment of Pain (EQUUS-FAP) (8; 9).

Although the use of grimace scales to assess pain is proven to be efficient, it requires the training of observers and the manual assessment of the pain score for each facial region described. There is a clear necessity for automation. Recent progress on action unit based estimation of sheep pain (10), using Sheep Pain Facial Expression Scale (SPFES) (6), illustrates the potential of this method. The foremost application is the development of training programs for recognizing pain in equines.

Hence, the primary aim of this work is the development of an automatic equestrian pain assessment system based on facial expressions. The model proposed should be robust to colour differences in the face and the existence or absence of a bridle in the equine's head. The end-goal of this project is the development of a complete pipeline for implementation in stables and farms using computer vision methods for face detection, landmark location, and posterior pain assessment. The continuous monitoring of signs of pain in equines would constitute a valuable tool to study disease progression, the effect of medication and improve the time response of the care-keepers, minimizing both the animal suffering and the economic impacts of the disease.

1.2 Objectives

In the present section, we describe the main goals of the project, starting with the definition and collection of data followed by each step of the proposed pipeline:

- Description of the dataset available and definition of the annotation process. If necessary, perform additional annotation tasks, obtaining web-based data.
- Development of a pose estimation model to predict the quantitative equine head pose for further development of a pose-informed pain assessment algorithm.
- Design of a landmark detection system, able to locate important facial areas.
- Proposal of a data augmentation method and evaluation of the impact of its impact on the two previous points.
- Implementation of a pain assessment model, based on facial expressions analysis.

1.3 Outline of the Dissertation

This document is organized into 6 chapters, focus on the contextualization of the automatic pain assessment problem and the solutions introduced for each step, in particular, pose estimation, automatic landmark detection, data augmentation and pain assessment.

Chapter 1 corresponds to the motivation and lists the main objectives of the work.

An overview of pain assessment in humans, followed by a description of grimace scales for equines and automatic pain assessment in animals is presented in Chapter 2.

The technical background to automatic facial landmark detection is introduced in Chapter 3, with a summary of the challenges of landmarking in humans and a concise description of different solutions proposed. Additionally, a review of the work developed until the date in animal landmark detection is made.

Data augmentation techniques based on the development of 3D models are introduced in Chapter 4, with an overview of the camera models paradigm and a review of work made in animals to develop 3D models based on 2D landmarks. Additionally, the method proposed in the present project is described.

Further, in Chapter 5, results for head pose estimation and landmark detection are presented, evaluating the effects of the data augmentation. Additionally, the performance of the pain estimation system is evaluated.

Lastly, a summary of the key points of the previously stated chapters is made, proposing future work for additional development (Chapter 6).

1.4 Contributions

The main contributions of the present work are:

- Development of a horse and donkey dataset with manually annotated landmarks and feature-level, detailed pain score ground truth, given by a veterinarian expert.

- Implementation of a method for accurate head pose detection, both quantitative and qualitative and automatic landmark detection.
- Introduction of a novel method for data augmentation, generating more realistic 3D models based on 2D landmarks.
- Proposal of a hierarchical system for pose-informed automatic pain prediction on horse faces.

In the context of this dissertation, a paper was submitted to the *International Workshop on Automated Assessment of Pain*, at the “15th International Conference on Automatic Face and Gesture Recognition” (11).

Chapter 2

Face-based Pain Assessment

In this chapter, face-based pain assessment in humans is introduced, followed by a detailed description of grimace scales in equines, presenting the adapted Equine Utrecht University Scale for Facial Assessment of Pain (EUUS - FAP) used in this project. Lastly, an overview of previous work made towards automatic pain assessment in animals is introduced, showcasing the relevance of the area and the main challenges encountered.

2.1 Pain Assessment in Humans

Systematic ways to measure facial behaviours can provide relevant information about human physical and mental health, as well as presenting an objective way to assess emotions. These methodologies are normally based on a Facial Action Coding System (12), describing action units (AUs) related to the underlying facial muscles and evaluating changes in expression for each of them. Although verbal communication facilitates the assessment of pain in humans, some circumstances hinder this method (i.e. severely ill, young children or speech impediments) which motivated the design of pain assessment scales based on human facial expressions (13; 14; 15).

So, approaches for automatic pain estimation based on pain scales are emerging, using computer vision techniques to classify the AUs appearance (16; 17; 15). Note that the most accurate ways to detect AUs use video as input, leveraging spatio-temporal cues (18) due to the subtleties of facial movements.

2.2 Pain Assessment in Equines

Following a similar approach, grimace scales were developed for several species, usually focusing on a specific cause of pain, for example, frequent illnesses or common surgery procedures.

In horses, abdominal pain is one of the most frequently diagnosed diseases, being associated with a high incidence and mortality rate (19). For this reason, assessment tools to help identify colic pain would have a high impact on the quality of patient care and overall equine welfare. Note that different types and sources of pain will manifest differently in the horse's face, and so, to create a reliable system the source of the discomfort

should be specified. In this context, van Loon *et al.* (9) proposed the Equine Utrecht University Scale for Facial Assessment of Pain (EQUUS-FAP) for horses suffering from acute colic. This scoring system (Table 2.1) describes various states for each facial areas, giving them an individual score from 0 to 2 and summing them to get a final score.

Table 2.1: Score sheet of the Equine Utrecht University Scale for Facial Assessment of Pain (EQUUS-FAP) (9).

<i>Data</i>	<i>Categories</i>	<i>Score</i>
Head	Normal head movement/interested in environment	0
	Less movement	1
	No movement	2
Eyelids	Opened, sclera can be seen in case of eye/head movement	0
	More opened eyes or tightening of eyelids;	1
	An edge of the sclera can be seen 50% of the time	
	Obviously more opened eyes or obvious tightening of eyelids; Sclera can be seen >50% of the time.	2
Focus	Focussed on environment	0
	Less focussed on environment	1
	Not focussed on environment	2
Nostrils	Relaxed	0
	A bit more opened	1
	Obviously more opened;	2
	Nostril flaring and possibly audible breathing	
Corners mouth/lips	Relaxed	0
	Lifted slightly	1
	Obviously lifted	2
Muscle tone head	No fasciculations	0
	Mild fasciculations	1
	Obvious fasciculations	2
Flehming and/or yawning	Not seen	0
	Seen	2
Teeth grinding and/or moaning	Not heard	0
	Heard	2
Ears	Orientation towards sound;	
	Clear response with both ears or ear closest to source	0
	Delayed/reduced response to sounds	1
	Backwards/no response to sounds	2
Total		... / 18

However, the dataset used in this project is composed of still images, and so, it is not possible to evaluate movement-dependent features. This fact will limit significantly the EQUUS-FAP, incapacitating the assessment of the “Head”, “Focus”, “Flehming and/or yawning”, “Teeth grinding and/or moaning” and “Ears” scores.

With a different focus, Costa *et al.* (7) proposed a grimace scale for pain assessment in horses undergoing castration. This procedure is performed routinely with studies showing evidence of acute and chronic pain after it. In contrast with colic pain, perceived as severely painful, the castration post-procedural pain is mild, which will reflect on the Horse Grimace Pain Scale (HGS) defined (Figure 2.1).



Figure 2.1: Horse Grimace Pain Scale with images and explanations for each of the 6 facial action units. Each AU is scored according to whether it is not present (score of 0), moderately present (score of 1) and obviously present (score of 2) (Image from (7)).

Having both HGS and EQUUS-FAP scales into account, a adapted scale was proposed for pain score assessment in still images (20) (Table 2.2). Example images of this grimace scale are presented in Figure 2.2.

Table 2.2: Score sheet for facial pain score assessment in still images, adapted from the EQUUS-FAP and HGS (20).

<i>Data</i>	<i>Categories</i>	<i>Score</i>
Ears	Both ears turned forwards	0
	At least one ear lateral position or further to backwards	1
	Both ears turned backwards	2
Orbital Tightening	Relaxed	0
	A bit tightening of the eyelids	1
	Obviously tightening of eyelid / eye closed	2
Angulated upper eyelid	Relaxed	0
	A bit more visible	1
	Obviously more visible	2
Visibility of the sclera	Sclera is not visible	0
	An edge of the sclera is visible	1
	Obviously more visible	2
Corners mouth / lip	Relaxed	0
	Lifted a bit	1
	Obviously lifted / strained	2
Nostrils	Relaxed	0
	A bit more opened	1
	Obviously more opened (dilated mediolaterally)	2
<i>Total</i>		<i>... / 10</i>

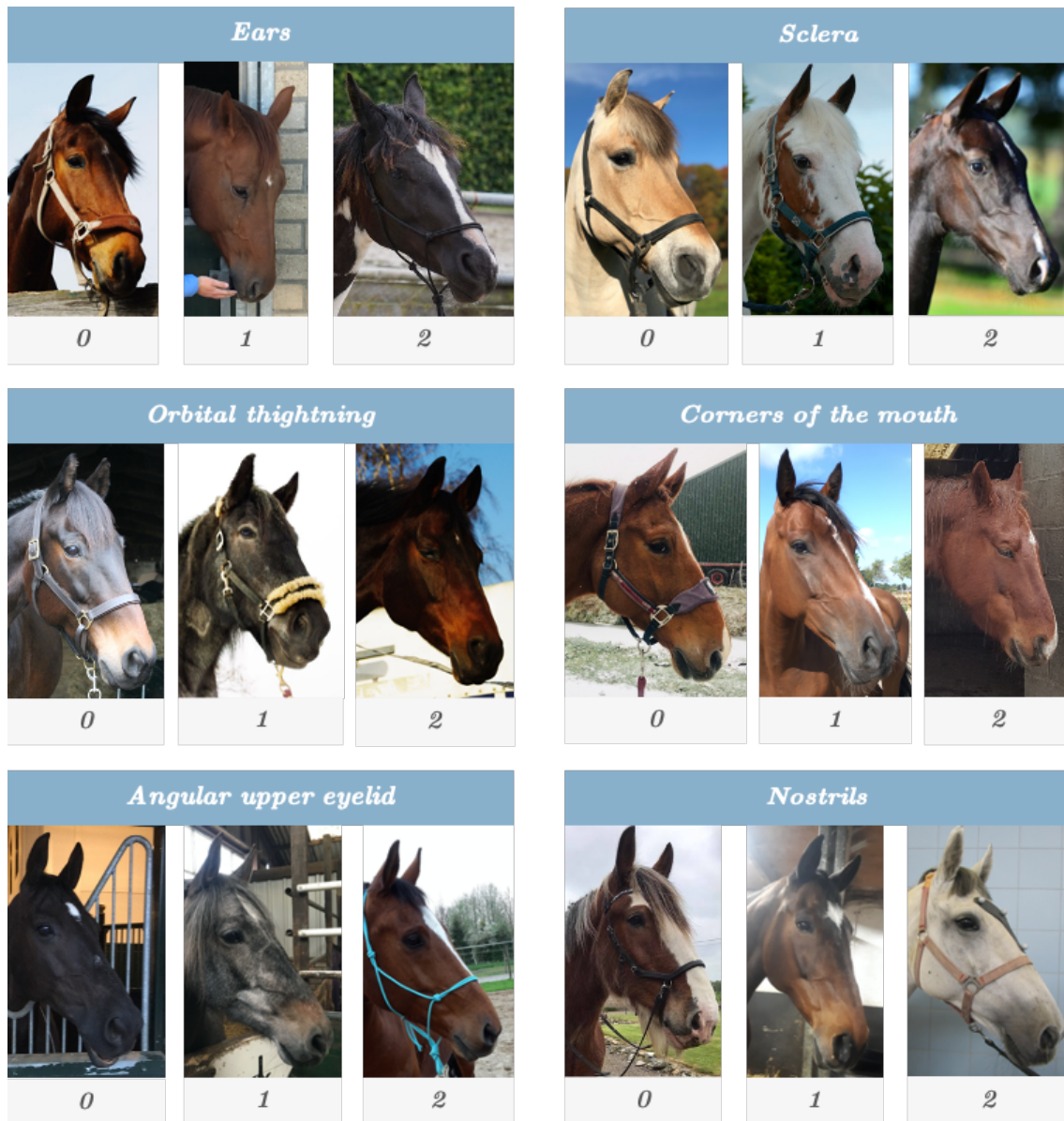


Figure 2.2: Example images of the pain score sheet used in the present work.

2.3 Automatic Pain Assessment in Animals

Although it is still the usual practice, manual classification is very time-consuming and can introduce unwanted bias. For this reason, automatic approaches for the pain annotation process are very appealing. In this context, a partially automated system was proposed by Sotocinal *et al.* (21) aiming to extract *scoring-ready* frames from videos of mice. However, this would still require pain manual assessment, a problem that was later tackled (22) using a convolutional neural network based on Inception V3 model. This method got a greater proportion of “pain” frames following a laparotomy surgery when compared to sham surgery or post-surgical analgesic, suggesting a correlation between the classification of “pain” and “no pain” status in mice. However, the model only considered the overall appearance of the face in a controlled environment (Figure 2.3), not having into account the information contained in grimace scales.

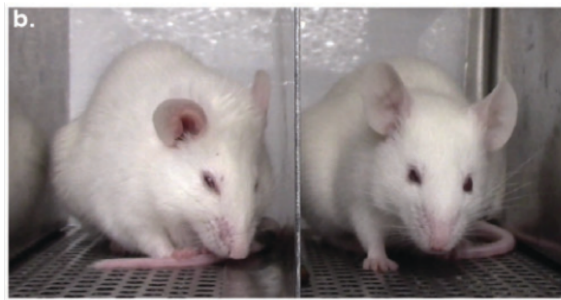


Figure 2.3: Setups used to capture continuous video footage of mice. Mice face towards the visual cliff for most of the recording session. (Image from (21)).

In a more similar line of work, Mahmoud *et al.* (10), showed the potential of an automatic pain assessment system in sheep, combining the pain prediction for several facial action units described in the Sheep Pain Facial Expression Scale (SPFES). To detect the regions-of-interest (ROIs) described in the scale, 8 facial landmarks were located using a modified version of Ensemble of Regression Trees (ERT) (23). However, the limited number of keypoints restricted the definition of ROIs. Further work on the estimation of facial landmarks was developed by Hewitt *et al.* (24) using 25 points and adding a pose estimation step to the pipeline, which lead to improvements for landmarking faces with extreme poses. The main limitation of both proposals was the face detection step with both, Viola-Jones (25; 10) and the Histogram of Oriented Gradient-Support Vector Machine model (HOG-SVM) (26; 24) proven to be insufficient to detect faces with the variety of head poses necessary. A complete pipeline was recently proposed (27), combining a fine-tuned SSD-Mobilenet model for face detection, with a CNN-based quantitative pose estimation system followed by a pose-informed landmark detection. HOGs features, as well as geometric features and the quantitative pose values, were used to train a binary SVM classifier, adapted to different head rotations and consequent self-occlusion (Figure 2.4).

Following the approach introduced by Mahmoud (10), previous work in horses suggested a classification model based on a combination of features, namely thinning, colour histograms and HOG (20). However, the extracted features were not sufficiently discriminative to achieve satisfactory performance. Nevertheless, this work showcases some of the challenges of pain assessment in horses, specifically, the scarcity of an-

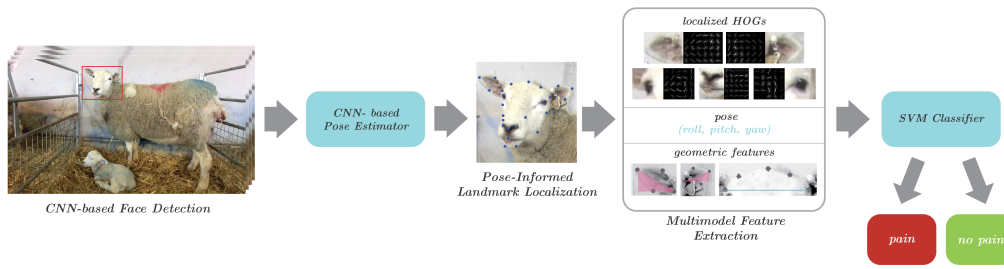


Figure 2.4: Pipeline for automatic approach for disease progression monitoring (Image from (27)).

notated data and high variations in colour and overall appearance between individual horses and between breeds. Additionally, the pose played a big part in the face appearance, with self-occlusion being an aggravating factor. Lastly, although the pain assessment model implemented was based on the location of the landmarks for the extraction of regions-of-interest, a landmarking system wasn't introduced. Further work in pain estimation in equines was proposed (28), extracting HOG features, SIFT features (29), LBP features (30) and VGG16 features (31) from the images and used it as the input of an SVM to predict the pain level of each region-of-interest described in the adapted grimace scale (Table 2.2).

In summary, grimace scales proved to be efficient at quantifying pain in animals, in particular, equines. However, the use of these scales is very labour intensive, seeing clear advantages in its automation. Previous work made in the animal pain assessment field showed the potential of this approach, inspiring the development of an automatic pain assessment system for equines. However, considering the grimace scales are based on changes in precise regions-of-interest in the face, such as eyes or ears, landmark detection is crucial for the correct extraction of each region for further classification. In the next chapter, automatic landmark detection methods are explored, evaluating the unique problems associated with the extension of human-based methods into animal faces.

Chapter 3

Automatic Landmark Detection

Landmarking in animals emerged as an extension of automatic landmark detection in humans, becoming important to introduce the challenges and potential approaches of the latter, to understand the prior. The methods presented were separated into three major categories: Holistic methods, Constrained Local Model methods and Regression-based methods (32). Lastly, previous work made on automatic landmark detection is described.

3.1 Automatic Landmark Detection in Humans

Accurate face alignment techniques are the basis of various face analysis applications in humans, such as face recognition, age estimation, and expression analysis. However, the automatic location of facial features is still challenging, with multiple factors adding to its complexity. The main factors that compromise the performance of the model are as follows (33):

- **Variability:** Human landmarks appearance will vary widely between individuals, being influenced by several features, namely, the hair, age, head pose, skin colour, and facial expressions.
- **Acquisition conditions:** the overall acquisition conditions of the dataset, such as illumination and resolution, will have a large influence in the landmark appearance and posterior detection.
- **Number of landmarks and their accuracy requirements:** Millborrow *et al.* (34) described an increase in the mean fit with the increment of the number of landmarks. This result was obtained using an Active Shape Model (35), described in the following section, confirming that the fitting of a landmark will facilitate the fitting of the others. Regarding the accuracy requirements, reference landmarks, such as the eyes and nose need to be detected with higher accuracy since they are often used to guide the location of secondary landmarks with less prominent features.

3.2 Holistic methods

Holistic methods learn a combined appearance and shape model during training and this model is used to fit landmarks to a testing image. This approach is associated with the Active Appearance Model (AAM), proposed by Cootes *et al.* (36) which combines a shape variation model and an appearance variation model.

The shape variation model is based on the Active Shape Model (ASM) (35), proposed by the same author, a statistical method that aims to fit a deformable shape to an object in the example. To train an ASM, it is mandatory to have a set of images annotated for the main points, applying a Procrustes analysis (37) to the shape x_i to iteratively obtain the optimal mean shape. For this purpose, the quality measurement used is the distance between each training shape and the mean shape (Equation 3.1), iterating towards its minimization. Therefore, after an initial alignment of all the shapes to the origin, a set of rigid transformations $T((t_x, t_y), \theta, (s_x, s_y))$ - translation, rotation, scale - is applied in each step with re-estimation of the mean shape until convergence.

$$D = \sum |x_i - \bar{x}|^2 \quad (3.1)$$

Considering a set of shapes align into a common coordinate frame, similar and plausible examples can be generated by modelling this distribution. A shape x_i combines the 2D coordinates of each of the n points of interest, and so is $2n$ dimensional, hindering the optimization problem. Therefore, a Principal Component Analysis (PCA) is used, transforming the cloud of points into a $2n - 4D$ space. Note that this model still holds the information of the original data, allowing its approximation using Equation 3.2.

$$x \approx \bar{x} + Pb \quad (3.2)$$

Here:

- $P = (p_1|p_2|...|p_t)$ contains the t eigenvectors of the covariance matrix of the data
- $b = P^T(x - \bar{x})$ is a t dimensional vector that defines the set of parameters of the deformable model.

When applying an ASM into an image, a rough approximation of the shape is made, initializing the model in the image frame:

$$X = T_{X_i, Y_i, s, \theta}(\bar{x} + Pb) \quad (3.3)$$

with the shape parameters b equal to zero (mean shape). The fitting occurs through an iterative approach (37):

- 1) **Examine a region of the image around each point X_i and find the best nearby match for the point Y_i ,** this is the new position of the point X_i . The definition of *best nearby match* can vary and should be adapted to the type of image analyzed. For instance, if the shape in question has strong edges, a good match would be the nearest strong edge (Figure 3.1). However, the strongest edges are not always the correct prediction and so, another approach would be to learn the statistical profiles of the contours in the trained model and measuring the Mahalanobis distance between the point profile and this model template (Figure 3.2).

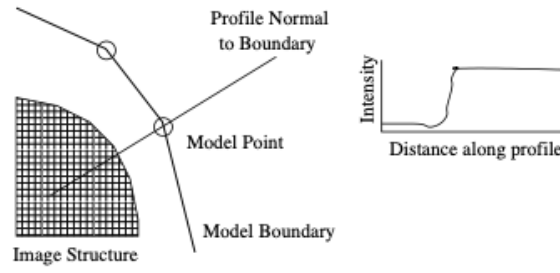


Figure 3.1: At each model point sample along a profile normal to the boundary (Image from (37))

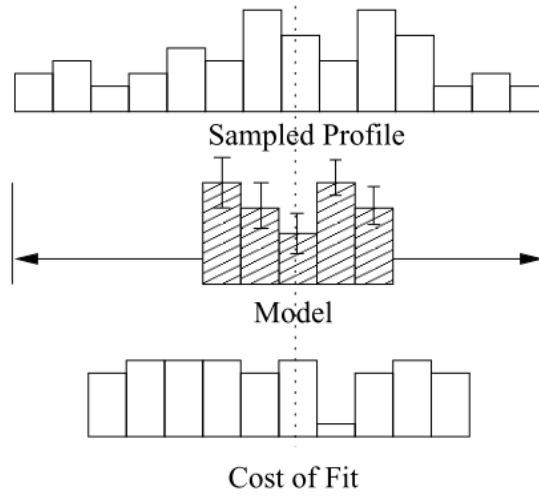


Figure 3.2: Search along sampled profile to find best fit of gray-level model (Image from (37)).

2) Update the parameters (X_t, Y_t, s, θ, b) to best fit the new found points Y .

- (a) Project the newly found points Y into the model frame by inverting the transformation T :

$$y = T_{X_t, Y_t, s, \theta}^{-1}(Y) \quad (3.4)$$

- (b) Project y into the tangent plane to $\bar{x}$ by scaling:

$$y' = y / (y \cdot \bar{x}) \quad (3.5)$$

- (c) Update the model parameters to match the y' , this is the best nearby match in the image frame after transformation into the model frame.

$$b = P_T / (y' - \bar{x}) \quad (3.6)$$

3) Apply constraints to the parameters, b , to ensure plausible shape. This is, limit so $|b_i| < 3\sqrt{\lambda_i}$ with λ_i corresponding to the eigenvalues.

4) Repeat until converges

Although the previous points describe the overall procedure, in practice it is used as a multi-resolution implementation. This method consists in searching for the best nearby match in versions of the image with successively finer resolution, updating the predictions in each step. For this, a Gaussian image pyramid is built, with each level L containing pixels 2^L times bigger than the ones on the original image (Figure 3.3). Therefore, the coarser levels pixels will represent more of the image, allowing for a rough estimate, and, going down the level it is possible to make successively more precise estimations having the previous one as a starting point. So, to get a final nearby match Y_i the following protocol is performed:

- 1) **Set** $L = L_{max}$
- 2) **While** $L \geq 0$
 - (a) Compute model point positions in the image at level L
 - (b) Search at n_s points (number of sample points either side of current point) on profile either side each current point
 - (c) Update pose and shape parameters to fit the model to a new point
 - (d) Return to (2a) unless more than p_{close} (Desired proportion of points found withing $n_s/2$ of the current point) of the points are found close to the current position, or N_{max} iterations have been applied at this resolution.
 - (e) If $L > 0$ then $L \rightarrow (L - 1)$
- 3) **Final result is given by the parameters after convergence at level 0.**

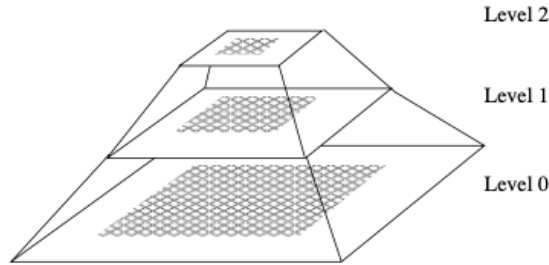


Figure 3.3: Multi - resolution implementation: A Gaussian image pyramid is formed by repeated smoothing and sub-sampling (Image from (37)).

The complete ASM algorithm, with a multi-resolution implementation, generates interesting results, being able to locate features in the face as seen in Figure 3.4. However, it requires a good initialization of the shape model, which demands approximate knowledge of the object location and it is not suitable for detecting objects with very diverse shapes. This, in the case of human faces, will implicate the training of separate models to detect widely different facial expressions. The introduction of the appearance factor in this model, creating the Active Appearance Model, came to solve the last point referred, making it possible to match all the classes of an object with a combined model (36).

For this purpose after performing the Procrustes analysis in the training set as described previously, the images are wrapped in a way that the points match to the mean shape

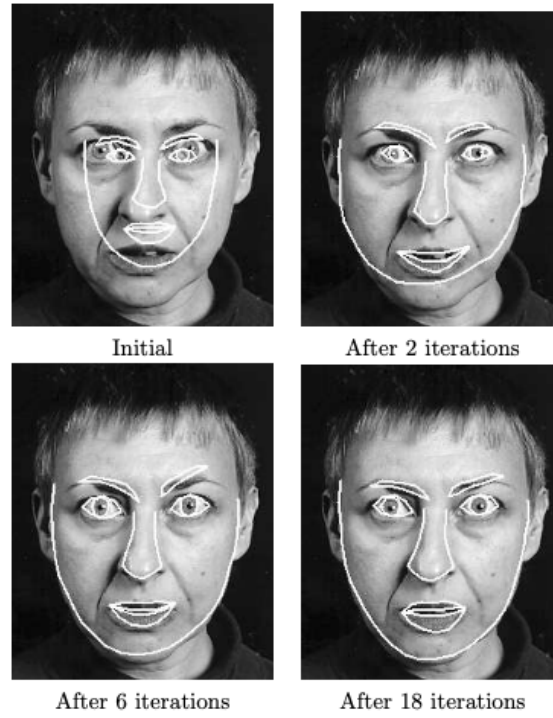


Figure 3.4: Landmark detection using an Active Shape Model (Image from (37)).

obtained. Following, a texture vector, $\mathbf{g}$, is computed for each image, succeeded by a normalization ($g \rightarrow (g - \mu_g)/\sigma_g$). Hence, for each image, there will be a correspondent point cloud and a shape-free patch (Figure 3.5).

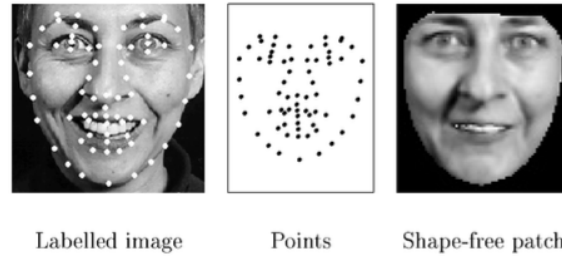


Figure 3.5: Example of the variables extracted from a training image by the Active Appearance Model (Image from (36)).

Then, the correlation between the two models (texture and shape) is learned, generating a combined model:

$$\begin{aligned} x &= \bar{x} + Q_s c \\ g &= \bar{g} + Q_g c \end{aligned} \tag{3.7}$$

Where x corresponds to the shape, g corresponds to the texture in a mean shaped patch and Q_s, Q_g are the matrices that describe the variation derived by the training set. The parameter of the deformable model is c present in both equations and, together with the shape transform t , defines the position of the model points in the image. Following

a similar updating process as the one described in ASM, the shape will be iteratively updated by a set of rigid transformations T . Regarding the texture in the image frame, it will be changed by scaling and applying offsets to the pixel intensities, $G = T_u(g) = (u_1 + 1)G + u_2$, where u is the vector of transformation.

To fit the model to an image is necessary to define an efficient adjusting pipeline, considering the full set of parameters of the model:

$$p^T = (c^T | t^T | u^T) \quad (3.8)$$

The iterative process is as follows:

1) Project the texture sample into the texture model frame:

$$g_s = T_u^{-1}(G) \quad (3.9)$$

Where g_s is the projection of the image intensities, G , in the texture frame described defined by T_u^{-1} .

2) Evaluate the error vector $r(p)$ and the current error, E :

$$\begin{aligned} r(p) &= g_s - g_m \\ E &= |r|^2 \end{aligned} \quad (3.10)$$

Where g_m is the current texture model, given by $g_m = \bar{g} + Q_g c$ and g_s is the image texture after being transformed into the texture frame. The model will change with the iterations, considering the update of the parameter c .

3) Compute the predicted displacements $\delta p = -Rr(p)$ where:

$$R = \left(\frac{\partial r^T}{\partial p} \frac{\partial r}{\partial p} \right)^{-1} \frac{\partial r^T}{\partial p} \quad (3.11)$$

This equation returns the displacement value δp that minimizes the error, $|r(p + \delta p)|^2$

- 4) Update the model parameters $p \rightarrow p + k\delta p$, where initially $k = 1$. This parameter will work as a learning rate, dictating the "speed" in which p changes.**
- 5) Update the model parameters - calculate the new points, X' , and respective model frame texture G' .**
- 6) Sample the image at the new points to obtain g_m**
- 7) Calculate a new error vector:**

$$r' = T_u^{-1}(G') - g'_m \quad (3.12)$$

- 8) If $|r'|^2 < E$, then accept the new estimative - there is a decrease in the error. Otherwise, try $k = 0.5$, $k = 0.25$, etc.**

9) Repeat until convergence

By varying the parameter c changes in age, pose, expression and identity are noticeable (Figure 3.6), which is indicative of the versatile of the model for face landmarking. The AAM model with multi-resolution implementation results in a robust system, able to successfully adapt to a variety of faces (Figure 3.7).

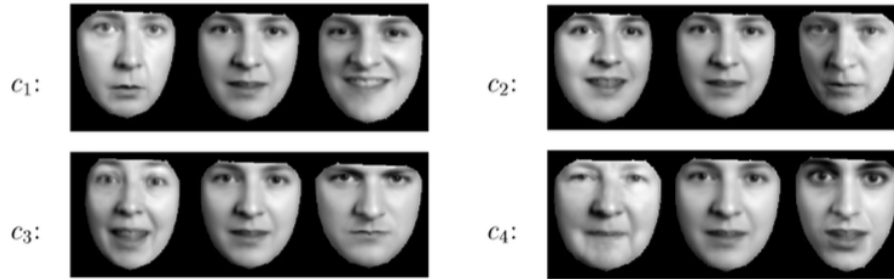


Figure 3.6: Effect of varying the first four facial appearance model parameter by ± 3 standard deviations from the mean (Image from (36)).

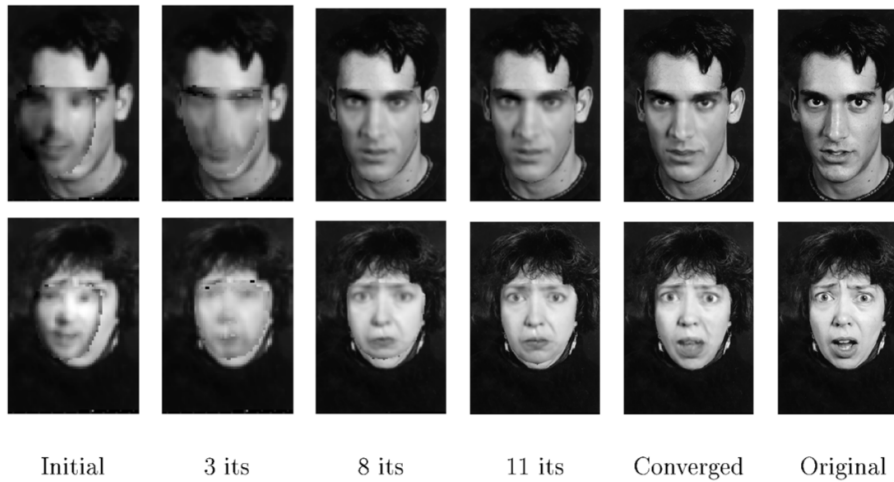


Figure 3.7: Multi-resolution search from displaced position using face model (Image from (36)).

On a final note, it is important to refer that variations of AAM have emerged focus on the fitting method improvement (32). The methods previously described can be classified as *analytic*, trying to solve the predicament as an optimization problem by minimizing a cost function. However, the use of gradient descent algorithms involves the calculation of Hessian and Jacobian matrices that are very computationally expensive. Hence, *learning-based* fitting can be a solution, offering a generally fast method, although less accurate. These approaches can use linear or non-linear regression, to predict the shape and appearance coefficients based on the image, requiring training images to learn the relationship between the coefficients and the appearance.

3.3 Constrained Local Models

Constrained Local Models designation, emerged from Cristinacce *et al.* work (38). In this, similar techniques to the ones described in Section 3.2 are used, however, it only defines texture templates for areas of interest. So, the final locations of the landmarks will be a combination of the independent local appearance information and the global facial shape.

To train an appearance model for each area of interest a training patch is defined around each feature, followed by normalization for both intensity and shape coordinates. This will result in two linear models, for shape and texture, accordingly:

$$\begin{aligned} x &= \bar{x} + P_s b_s \\ g &= \bar{g} + P_g b_g \end{aligned} \quad (3.13)$$

In which $\bar{x}$ is the mean shape, $\bar{g}$ is the mean normalised gray-level vector, P_s and P_g are a set of orthogonal modes of variation, b_s is a set of shape parameters and b_g is a set of gray-level parameters.

These template models are then combined using a PCA, obtaining a joint model similar to the one introduced in Equation 3.7:

$$b = P_c c = \begin{pmatrix} W_s b_s \\ b_g \end{pmatrix} \quad (3.14)$$

Where:

- $P_c = \begin{pmatrix} P_{cs} \\ P_{cg} \end{pmatrix}$ is the orthogonal matrix computed using PCA. It can be divided into two matrices P_{cs} and P_{cg} that compute the shape and texture parameters, respectively, given a joint parameter vector c .
- W_s is a weight vector, that accounts for the differences between the shape and texture units.

Again, similarly to what was presented in Figure 3.6, the variation of the parameters of c will lead to a big change in pose and identity, as seen in Figure 3.8.

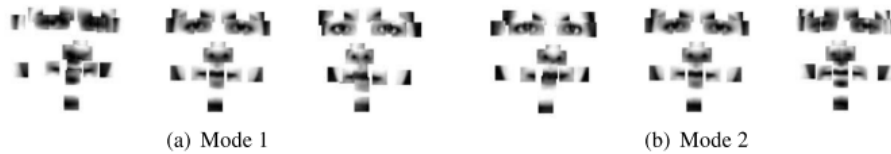


Figure 3.8: Joint Modes of Shape and Texture Variation by ± 3 standard deviations from the mean. (Image from (38)).

To fit the model to a new image, firstly it is necessary to input an initial set of feature points. Afterwards, the searching process starts (Figure 3.9):

1. **Fit the joint model to the current set of feature points to generate a set of templates.** Regarding the shape, to obtain the position vector in the image frame, X , a transformation from the shape model frame will be made:

$$X \approx T_t(\bar{x} + P_s b_s) \quad (3.15)$$

From these points, a set of CLM templates is generated represented by patches with a fixed rectangular shape around each landmark.

2. **Use the shape constrained search method to predict a new set of feature points**
By applying the templates to the image, a set of responses, $I_i(X_i)$ is computed, returning a confidence score for that location. When applying this classifier for detection, different locations are tested and a confidence score is returned. All the parameters of Equation 3.16 can be concatenated into $p = (t^T | b_s^T)^T$, with t corresponding to the similarity parameters of the transform T_t . So, X can be defined as a function of these parameters, is the final goal it's optimization:

$$f(p) = \sum_{i=1}^n I_i(X_i) + K \sum_{j=1}^s \frac{-b_j^2}{\lambda_j} \quad (3.16)$$

The first term is the sum of the response of each feature template over all the points and the second term is an estimate of the log-likelihood of the shape given the shape parameters b and the eigenvalues λ . The K weight will define the relative importance between a good shape and a high feature response. The function is optimized for each interaction using the Nelder-Mead simplex algorithm (39).

3. **Repeat until convergence**

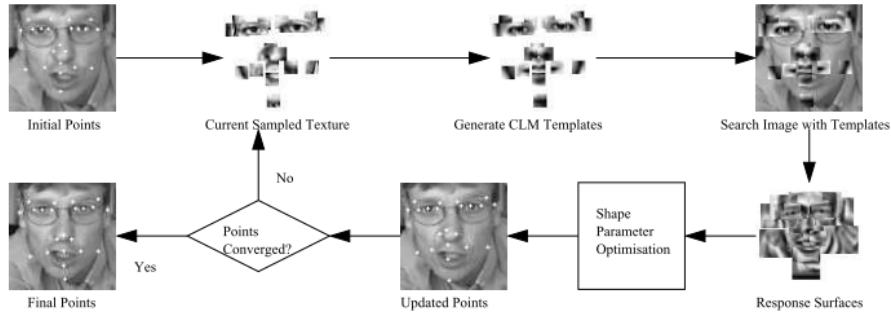


Figure 3.9: CLM search algorithm (Image from (38)).

When compared with the AAM approach that preceded this method, the CLM leads to an increase in accuracy and robustness being also more adequate for face tracking. However, different types of appearance and shape can be explored, using this strategy as a baseline. A diversity of variations can be made by altering the local appearance model from a classifier to a regression-based model (40), changing the type of appearance features used, for instance, to Histogram of Oriented Gradients (HOG) features (41), Scale-invariant feature transform (SIFT) features (42) or deep-learning-based features (43), or by exploring probabilistic face shape model instead of the deterministic approach used (44).

3.4 Regression-based methods

The regression-based methods can use holistic or local appearance information to define a regression model that best explains the data. They can be broadly divided into three categories: direct regression methods, cascaded regression methods, and deep-learning-based regression methods.

3.4.1 Direct Regression Methods

Direct regression methods will learn the direct mapping between the image appearance and the facial landmark location, with no need for initialization, a positive point considering the problems associated with a weak initialization referred to in the Section 3.2.

As an example of a regression method using local facial features, it will be described the use of a Conditional Regression Forest to predict the position of the landmarks (45). A Random Forest (46) is a combination of tree-structured classifiers, in which each tree will depend on a random vector, all independent of each other and with identical distribution. Each decision tree will cast a unitary vote for the most popular class in the data and a final result is obtained through a majority vote. Applying this simple classifier to the present Computer Vision problem, each tree T in the forest $\tau = \{T_t\}$ is built from a different, randomly selected, set of training images. Then, a set of square patches is randomly extracted from each image with each patch $P_i = \{(I_i, D_i)\}$ where I_i is the patch appearance and D_i the displacement in relation to each landmark.

To train the model it was defined the following patch comparison feature:

$$f_\theta(P) = \frac{1}{|R_1|} \sum_{q \in R_1} I^a(q) - \frac{1}{|R_2|} \sum_{q \in R_2} I^a(q) \quad (3.17)$$

In which $\theta = (R_1, R_2, a)$, R_1 and R_2 are two rectangles withing the patch boundaries, and $a \in \{1, 2, \dots, C\}$ is the selected appearance channel. The appearance channels will contain the grey values of the raw image, the grey values of the normalized image and additional channels that represent a Gabor filter bank.

The training will then follow the framework presented in (46; 47):

1. Generate a pool of splitting candidates $\phi = (\theta, \tau)$ with τ being the threshold used.
2. Divide the set of patches $\mathcal{P}$ into two subsets $\mathcal{P}_L$ and $\mathcal{P}_R$ for each ϕ , corresponding to the two branches of a node.

$$\begin{aligned} \mathcal{P}_L(\phi) &= \{\mathcal{P} | f_\theta(P) < \tau\} \\ \mathcal{P}_R(\phi) &= \mathcal{P} \setminus \mathcal{P}_L(\phi) \end{aligned} \quad (3.18)$$

3. Select the splitting candidate ϕ which maximizes the evaluation function Information Gain (IG):

$$IG(\phi) = \mathcal{H}(\mathcal{P}) - \sum_{S \in \{L, R\}} \frac{|\mathcal{P}_S(\phi)|}{|\mathcal{P}|} \mathcal{H}(\mathcal{P}_S(\phi)) \quad (3.19)$$

Where $\mathcal{H}(P)$ is the defined class uncertainty measure, given by Equation 3.4.2. So, the information gain measures the reduction in the uncertainty (entropy) by splitting the data.

$$\begin{aligned}\mathcal{H}(\mathcal{P}) &= -\sum_{n=1}^N \frac{\sum_i p(c_n|\mathcal{P}_i)}{|\mathcal{P}|} \\ p(c_n|\mathcal{P}_i) &\propto \exp\left(-\frac{|d_i^n|}{\lambda}\right)\end{aligned}\tag{3.20}$$

Where $p(c_n|\mathcal{P}_i)$ is the probability of the patch $\mathcal{P}_i$ belonging to the feature point n . This probability will be one if the patch is in the position of the n -th facial feature and will be zero if this patch is very far from the feature point. The factor λ will control the steepness of this function with d_i^n being an offset in relation to the centroid of the patch i for a facial feature.

4. Create leaf l (terminal node) when maximum depth is reached or the information gain $IG(\phi)$ is below a predefined threshold. Otherwise, repeat the process recursively.

The conditional component is added by training a different tree for each pose (Figure 3.10). As seen in Section 3.1, head pose variations are a common problem in landmark detection which motivates the design of pose informed approaches. Since obtaining continuous ground truth for the head pose is very difficult, the authors used qualitative labels for the yaw angle, $w = \{-90, -45, 0, +45, +90\}$. Then, a similar regression forest was trained to classify simultaneously if the patch belonged to the background or foreground (considering that the face detection can be slightly wrong) and the pose of the face.

The introduction of the pose information in the landmark location system can be made in two ways: first predict the pose followed by testing in the regression forest specific for that pose (hard method) or, as seen in Figure 3.10, by combining the results of the different head pose based models selecting trees from each of them based on the head pose probability (soft method). Likewise, new features can be added to the model, creating conditional random forests based on age, gender or other appearance features that can improve the landmarking (48).

Defining regression models for mapping global features can be very complex due to the amount of variability in the face, as already seen in the transition from the Active Appearance Model to the Constrained Local Model. For this reason, the majority of methods in this area are deep-learning-based (49; 50).

3.4.2 Cascaded Regression Methods

The cascade regression methods will start with an initial guess, updating it through a series of regression functions until reaching a final result. So, in the training phase, instead of doing the direct mapping between the appearance and the landmark locations, each regression function will learn “on top” of the previous one, updating the shape based in the previous prediction to reach the final goal.

The Explicit Shape Regression algorithm proposed by Cao *et al.* (51) and inspired in the Cascaded Pose Regression (CPR) (52), describes a framework based in gradient boosting

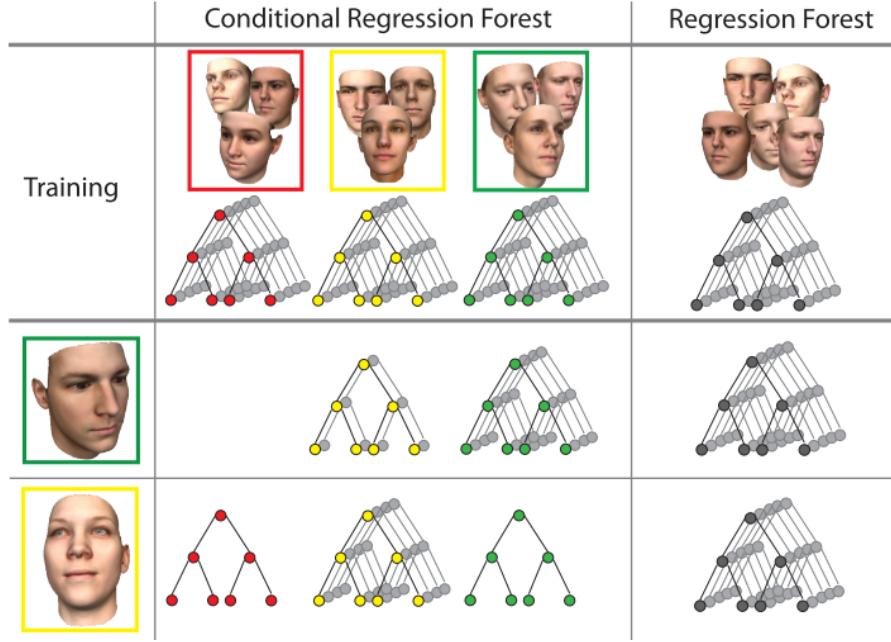


Figure 3.10: While a regression forest is trained on the entire training set and applied to all test images, a conditional regression forest consists of multiple forests that are trained on a subset of the training data illustrated by the head poses (coloured red, yellow, green). When testing on an image (illustrated by the two faces at the bottom), the head pose is predicted and trees of the various conditional forests (red, yellow, green) are selected to estimate the facial feature points (Image from (45)).

regression. In a first instance, the shape is normalized by aligning it to the mean shape, M_S (see the description of the mean shape in ASM - Section 3.2).

Considering N training samples, defined by the facial image, ground truth shape and initial shape $\{I_i, \hat{S}_i, S_i^0\}_{i=1}^N$ the regressors $(R^1, \dots, R^T)$ are sequentially learn in order to minimize the displacement error in the training set:

$$R^t = \underset{R}{\operatorname{argmin}} \sum_{i=1}^N \left\| y_i - R(I_i, S_i^{t-1}) \right\|_2 \quad (3.21)$$

$$y_i = M_{S_i^{t-1}} \circ (\hat{S}_i - S_i^{t-1})$$

With S_i^{t-1} being the shape estimated by the last regressor $(t-1)$, y_i the normalized regression target and $M_{S_i^{t-1}}$ the mean shape of the last iteration. This is, instead of mapping from a shape indexed feature to $(\hat{S}_i - S_i^{t-1})$, the difference between the ground truth and the last shape estimated, we used the normalized version of this, simplifying the regression task.

In the testing phase, the facial image I with the initial shape S^0 will go through each stage regressor, computing a normalized shape based on the image features, as seen in Equation 3.22.

$$S_i^t = S_i^{t-1} + M_{S_i^{t-1}}^{-1} \circ R^t(I_i, S_i^{t-1}) \quad (3.22)$$

Where the regressor R^t updates the previous shape S_i^{t-1} based on the facial image I_i . Regarding the regressor used, it was a *fern*, a regressor introduced by Ozuysal *et al.* (53) and also used by Dollar *et al.* (52), for the CPR. These regressors offer an alternative to randomized trees being faster and simpler to implement. Each fern consists of a small set of binary tests, giving the probability of a patch to belong to a class learned during training. In practice, what happens is that the regressors will learn increasingly more subtle aspects, as seen in Figure 3.11 where at the first regressor the main aspects learned are the pose and scaling while on the last the differences learned are based on small variations in the contours or facial expressions.

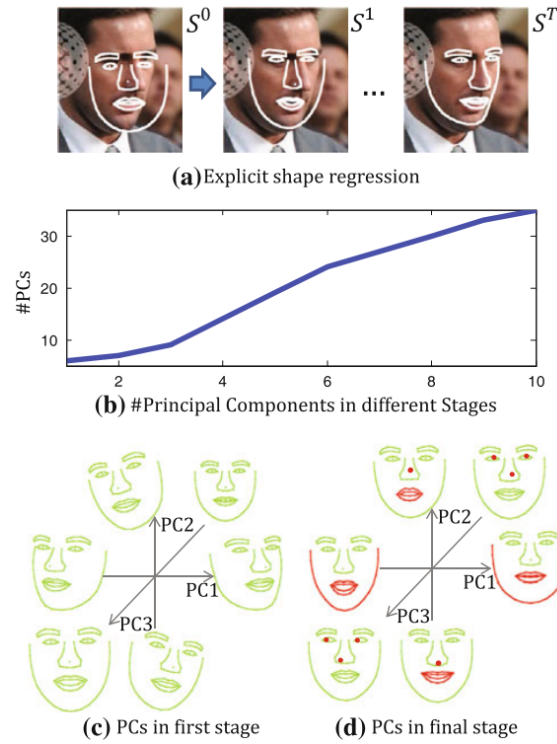


Figure 3.11: Shape constraint is preserved and adaptively learned in a coarse to fine manner in the boosted regressor. **(a)** The shape is progressively refined by the shape increments learnt by the boosted regressors in different stages. **(b)** Intrinsic dimensions of learnt shape increments in a 10-stage boosted regressor, using 87 facial landmarks. **(c), (d)** The first three principal components (PCs) of shape increments in the first and final stage, respectively (Image from (51)).

Another possible approach would be the use of an Ensemble of Regression Trees (54). In this work, the regressors used will be a regression tree, following a similar procedure to the one described before.

The optimization method used is very important to get a fast and robust system. Parameterized Appearance Models (PAMs) such as the AAM will often use the Gauss-Newton method to search over the parameter space while discriminative approaches, will establish a linear regression between the motion parameters and the appearance differences, as seen in the Section 3.4.1. Supervised Descent Method (55) will unify these two approaches.

Considering an image $d \in \mathbb{R}^{m \times 1}$ of m pixels, $d(x) \in \mathbb{R}^{p \times 1}$ indexes p landmarks in the image. h is a non-linear feature extraction function (for instance, SIFT (56)) and $h(d(x)) \in \mathbb{R}^{128p \times 1}$ in the case of extracting SIFT features. During training, the ground truth landmarks, x_* , are known so the face alignment problem can be described as follows:

$$f(x_0 + \Delta x) = \|h(d(x_0 + \Delta x)) - \phi_*\|_2^2 \quad (3.23)$$

Where $\phi_* = h(d(x_*))$ represents the SIFT values in the manually labelled landmarks. Since the SIFT operator is not differentiable the minimization of the previous equation using first and second derivatives would require numerical approximations of the Hessian ($H(x)$) and Jacobian ($J(x)$), which is very computationally expensive. For this reason, SDM learns a series of descent directions and re-scaling factors, in a way that produces a sequence of updates starting from x_0 and converging in x_* in the training data. By applying the second-order Taylor expansion with Newton-type method, the shape updates are calculated:

$$f(x_0 + \Delta x) \approx f(x_0) + J_f(x_0)^T \Delta x + \frac{1}{2} \Delta x^T H(x_0) \Delta x \quad (3.24)$$

Where $J_f(x_0)$ and $H(x_0)$ are the Jacobian and Hessian matrices of f (the function to minimize) evaluated at x_0 . Differentiating the Equation 3.25 with respect to Δx to get the first update for x (the x_0 was omitted for simplification):

$$\Delta x_1 = -H^{-1} J_f = -2H^{-1} J_h^T (\Phi_0 - \Phi_*) \quad (3.25)$$

Considering $R_0 = -2H^{-1} J_h^T$, the *descent direction* can be rewrite the step as:

$$\Delta x_1 = R_0 \Phi_0 + b_0 \quad (3.26)$$

With b_0 being the bias term, equal to $R_0 \Phi_*$ learned during training. However, it is unlikely to reach convergence with a single iteration so it's necessary to generalize the previous equation to update the landmark positions from the previous iteration:

$$\begin{aligned} x_k &= x_{k-1} - 2H^{-1} J_h^T (\Phi_{k-1} - \Phi_*) \\ x_k &= x_{k-1} + R_{k-1} \Phi_{k-1} + b_{k-1} \end{aligned} \quad (3.27)$$

Such that the successions of x_k will converge into x_* , for all the images of the training set, by learning a sequence of generic descent directions $\{R_k\}$ and bias terms $\{b_k\}$. The values of R_k and b_k that obtained by minimizing the well-known linear least square problem:

$$\underset{R_k, b_k}{\operatorname{argmin}} \sum_{d^i} \sum_{x_k^i} \left\| \Delta x_*^{ki} - R_k \Phi_k^i - b_k \right\|^2 \quad (3.28)$$

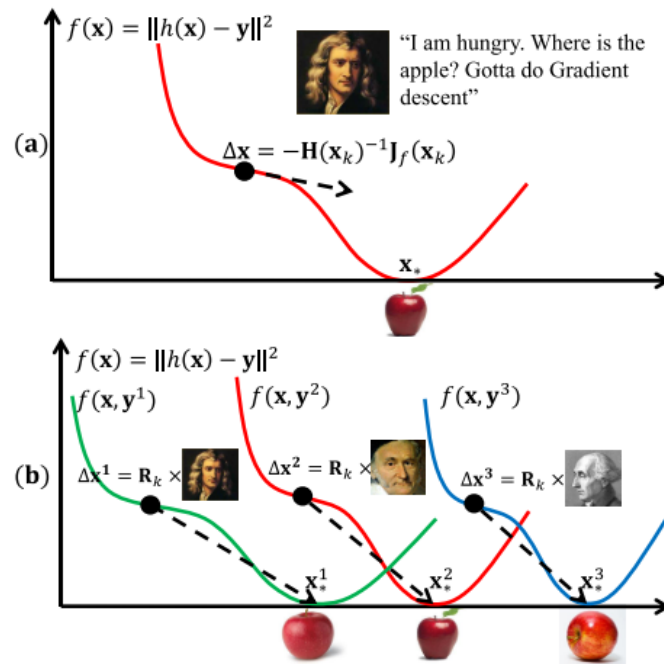


Figure 3.12: **a)** Using Newton's method to minimize $f(x)$. **b)** SDM learns from training data a set of generic descent directions $\{R_k\}$. Each parameter update (Δx_i) is the product of R_k and an image-specific component (y_i), illustrated by the 3 great Mathematicians. Observe that no Jacobian or Hessian approximation is needed at test time (Image from (55)).

The major difference, when comparing to the AAM, is that SDM has multiple step regressions as the AAM only has one (Figure 3.12). This leads to an increased in the performance, achieving state-of-art results for the facial feature detection problem.

However, the cascade methods, in general, will have some problems, namely, the initialization of the location of the landmark. Another issue will be choosing the number of cascaded predictions, not being clear when is the best time to stop it.

3.4.3 Deep-Learning Based Methods

With the increased popularity of Convolutional Neural Networks (CNN) as a way to solve computer vision problems, it has been suggested the application of the frameworks previously discussed in a deep learning approach. These methods can be classified as:

- *Pure-learning methods:* a CNN model predicts directly the landmark location in the facial images. In early work proposed by Sun *et al.* (57) a CNN was trained to locate five keypoints using a cascaded regression approach in a first-level to take the full face and make a keypoints' prediction based in global high-level features and on the texture context information. It then refines the initial estimation with two levels based on a small local region around each point (Figure 3.13). At each level multiple convolutional networks are fused to achieve higher accuracy, being the two last levels shallower, due to the decrease in complexity of the input. Further work, improved this methodology by applying multi-task learning

(50; 58), optimizing the main task, *facial landmark detection*, with auxiliary tasks such as *age estimation*, *pose estimation* or *gender classification*. Another optimization approach could be the increase of the number of points detected (59) or training a single CNN to mimic a cascade behaviour (e. g. Recurrent Neural Networks (60)) instead of placing multiple networks in a cascaded way.

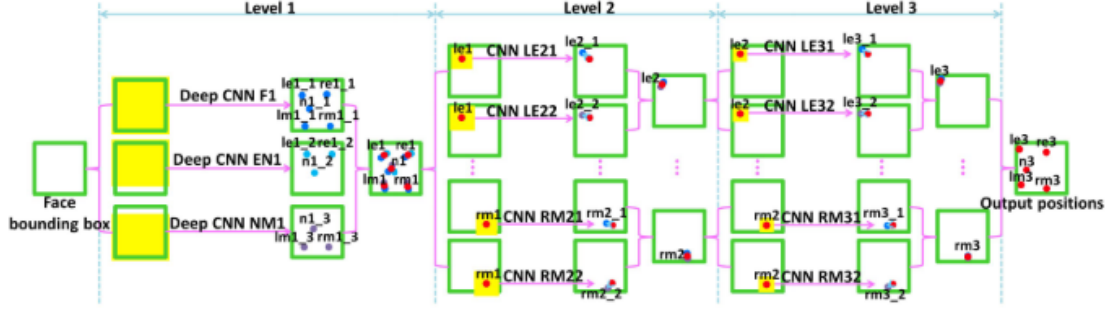


Figure 3.13: Three-level cascaded convolutional networks. The input is the face region returned by a face detector. The three networks at level 1 are denoted as F1, EN1, and NM1. Networks at level 2 are denoted as LE21, LE22, RE21, RE22, N21, N22, LM21, LM22, RM21, and RM22. Both LE21 and LE22 predict the left eye centre, and so forth. Networks at level 3 are denoted as LE31, LE32, RE31, RE32, N31, N32, LM31, LM32, RM31, and RM32. The green square is the face bounding box given by the face detector. Yellow shaded areas are the input regions of networks. Red dots are the final predictions at each level. Dots in other colours are predictions given by individual networks (Image from (57)).

- *Hybrid methods*: Hybrid models will predict 3D shape deformable model coefficients and the head pose, combining the CNN with a 3D vision. In (61; 62) a 3D Dense Face Alignment approach was proposed, in which a dense 3D face model, obtained from the 2D positions from the training set, is fitted to the image via a CNN (Figure 3.14).

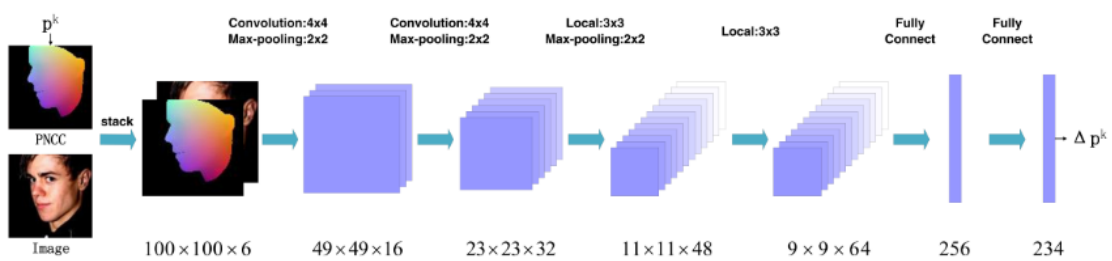


Figure 3.14: An overview of 3DDFA. At k th iteration, Net^k takes a medium parameter p^k as input, constructs the projected normalized coordinate code (PNCC), stacks it with the input image and sends it into CNN to predict the parameter update Δp^k . (Image from (61)).

3.5 Face Alignment in Animals

When extrapolating the existent knowledge to animal models new challenges emerge. The lack of annotated datasets, already widely available for human landmark detection, will be a big weighting factor, nevertheless, the morphology of the animal face will also introduce difficulties. Fur and colour variations, ear movement, as well as the accentuated depth changes with rotation, are some of the factors that differentiate this problem from human landmarking.

For cats and dogs, Bulat *et al.* (63), proposed a deep learning approach, applied in a dataset with 1511 images of cats and 1514 images of dogs. In the proposed method, Convolutional Aggregation of Local Evidence (CALE), a CNN returns the detections heat maps for individual facial landmarks, that are then aggregated with early CNN features through joint regression to refine the landmark location. The approach proved to be efficient in both animals and humans, achieving a Normalized Mean Error (NME) of 2.71 %. The normalization factor was the square root of the face size, calculated from the bounding box, for both cats and dog (Figure 3.15).



Figure 3.15: Qualitative results produced by CALE on our CatsDogs dataset (Image from (63)).

With a focus on human and sheep, Yang *et al.* (23) proposed a new feature extraction scheme called Triplet-Interpolated Features (TIF), used at each iteration of the cascaded shape regression framework (a modified version of the Ensemble of Regression Trees (54)). The method was applied in a dataset with 600 sheep faces and only 8 facial landmarks. Further work, (24) increased the number of landmarks to 25, applying a CNN model to estimate the head pose followed by a pose informed Ensemble of Regression Trees (PI-ERT), to predict the landmark location. The results obtained show an improvement when compared with the TIF method that preceded it, achieving an MNE of 4.5 %, normalized by the edge length of the bounding box (Figure 3.16)).

Regarding equines facial landmarks, Rashid *et al.* (64) explored the finetuning of networks implemented for human faces, adapting the animal images to correct the differences in shape between equines faces and human faces. So, the proposed method will have two phases, first, the warping of the image based on the human face shape followed by the detection of the keypoints in the warped image with a network pre-trained in human faces (Figure 3.17). The failure rate obtained was 8.36 %, being this metric defined as the percentage of errors higher than 10% of the face (bounding box) length.



Figure 3.16: Qualitative examples of landmarks localisation improvements made by PI-ERT. The right-most column shows an example where PI-ERT struggles with some landmarks due to inaccurate head pose estimation. Rows (from top to bottom): ground-truth, standard ERT and PI-ERT. (Image from (24)).

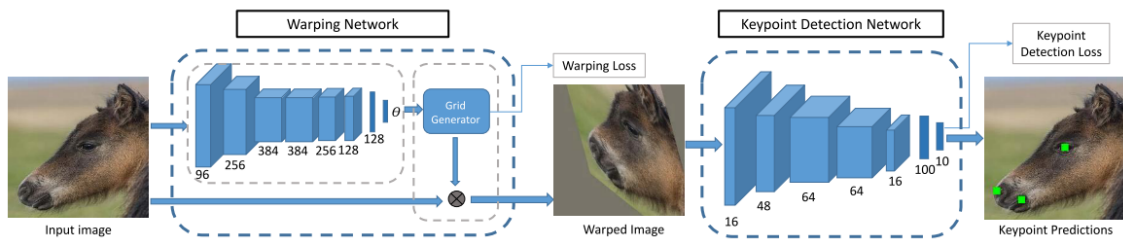


Figure 3.17: Network architecture for animal facial keypoint detection. During training, the input image is fed into the warping network, which is directly supervised using keypoint-annotated human and animal image pairs with a similar pose. The warping network warps the input animal image to have a human-like shape. The warped animal face is then passed onto the keypoint detection network, which finetunes a pre-trained human keypoint detection network with the warped animal images (Image from (64)).

As a final note, it's important to point out that the metrics used in the previous examples are all based on the bounding box/face length. However, in human landmark detection, the inter-ocular distance is commonly used as a normalization factor. Since the error

will be scaled by the inter-ocular distance it will be invariant to the variation of each individual face size and camera zoom, allowing the comparison of point to point errors between images.

In sum, although automatic landmark detection is already very established in the human face detection field, new challenges appear when extrapolating these methods for animal applications. Self-occlusion due to long nose, variations in the texture and colour of the fur and the differences between breeds of the same animal are some of the factors that will make this process particularly challenging. Besides, the annotated datasets available in this field are scarce, difficulting the training process. For this reason, new approaches for data augmentation would bring significant benefits to the research area, allowing the development of more accurate systems with the preexistent data. In the following chapter, data augmentation approaches based on 2D landmarks will be discussed proposing a novel 3D-based method for data augmentation from 2D images.

Chapter 4

Data Augmentation

In the present chapter, an introduction to camera models is made followed by a description of the state-of-art from 3D models based solely on 2D images (with landmark annotations). Then, the novel approach introduced in this dissertation for data-augmentation is described, using Thin-Plate-Splines to deform a simple 3D model based on 2D images. Lastly, examples of the resulting synthetic images are presented.

4.1 Camera Models

When applying 3D models to images, is important to understand how to map a 3D world into the 2D space. Starting with the simplest camera model, the *pinhole model* considers a central projection of points in space into a plane at $Z = f$, *focal plane* or *image plane*. In this scenario, a point $\mathbf{X} = (X, Y, Z)^T$ is mapped where the line joining the point $\mathbf{X}$ to the central projection, camera centre, meets the image plane, $\mathbf{x}$ (Figure 4.2).

So, the central projection mapping from world to image coordinates can be defined as:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \mapsto \begin{pmatrix} fX/Z \\ fY/Z \end{pmatrix} \quad (4.1)$$

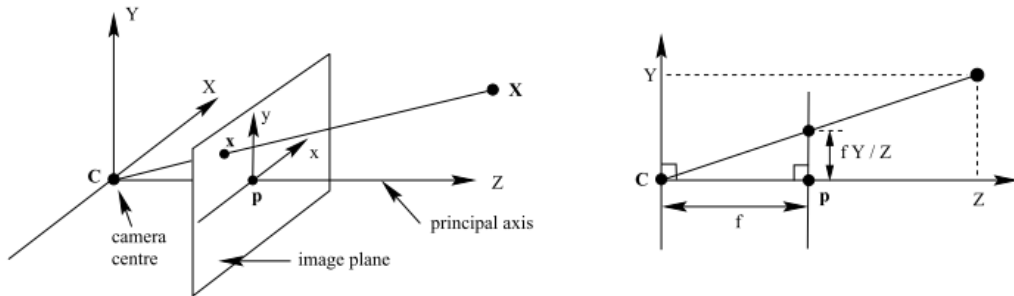


Figure 4.1: Pinhole camera geometry. C is the camera centre and p the principal point. The camera centre is here placed at the coordinate origin. Note the image plane is placed in front of the camera centre (Image from (65)).

If the world and image points are represented by homogeneous vectors then the central projection will express a linear mapping between their homogeneous coordinate. This can be written as:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (4.2)$$

Where f is the focal length. Considering $\mathbf{X}$ the world point represented by the homogeneous 4-vector $(X, Y, Z, 1)^T$, $\mathbf{x}$ the image point represented by a homogeneous 3-vector and P the 3×4 homogeneous *camera projection matrix*, Equation 4.5 can be compressed into:

$$\mathbf{x} = P\mathbf{X} \quad (4.3)$$

With:

$$P = \text{diag}(f, f, 1)[I|0] \quad (4.4)$$

The previous models assume that the origin of coordinates at the *image plane* is at the *principal point*, this is, the point where the principal axis meets the image plane. However, this may not be the case, and so, an offset should be added. This can be written in homogeneous coordinates as:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & p_x & 0 \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (4.5)$$

Where $(p_x, p_y)^T$ are the coordinates of the principal point. Considering the *camera calibration matrix*, K , as:

$$K = \begin{bmatrix} f & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4.6)$$

The image point $\mathbf{x}$ can be written as:

$$\mathbf{x} = K[I|0]\mathbf{X}_{cam} \quad (4.7)$$

With $\mathbf{X}_{cam}$ being the world coordinates, assuming that the world referential is located at the origin of a Euclidean coordinate system with the principal axis of the camera pointing straight down the z-axis. However, the world coordinate frame can also be rotated in relation to the camera coordinate frame. This transformation can be defined as:

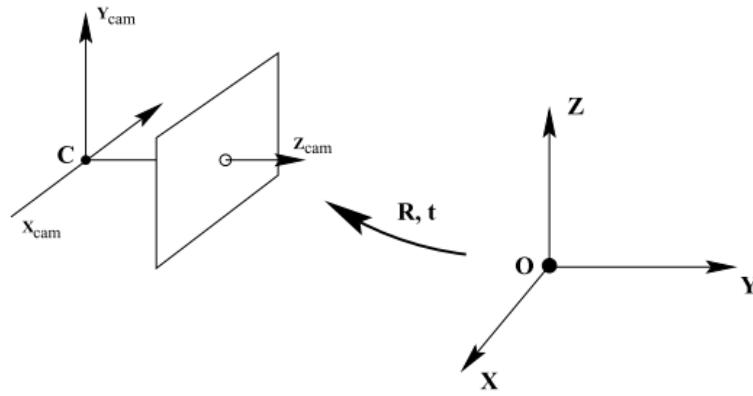


Figure 4.2: The Euclidean transformation between the world and camera coordinate frames. (Image from (65)).

$$\mathbf{X}_{cam} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \mathbf{X} \quad (4.8)$$

Where $\tilde{C}$ represents the coordinates of the camera centre in the world coordinate frame, R is a 3×3 rotation matrix representing the orientation of the camera coordinate frame and $\mathbf{X}$ is the point coordinates in the world coordinate frame.

Combining the Equations 4.7 and 4.8 leads to:

$$\mathbf{x} = KR[I | -\tilde{C}]\mathbf{X} \quad (4.9)$$

So, the general pinhole camera can be described as $P = KR[I | \tilde{C}]$. It has 9 parameters, three for K (f, p_x, p_y) related to the internal orientation of the camera (*internal parameters*), three for R and three for $\tilde{C}$ related to the camera position (*external parameters*).

To facilitate further calculations, it is convenient not to make the camera centre explicit, and so the world to image transformation can be expressed as $\tilde{\mathbf{X}}_{cam} = R\tilde{\mathbf{X}} + t$, with the camera matrix being:

$$P = K[R | t], \quad (4.10)$$

where $t = -R\tilde{C}$.

Until now it was assumed that the Euclidean coordinates had an equal scale in both axial directions, but that is not always the case. Being the number of pixels per unit distance in the image coordinate equal to m_x and m_y in x and y directions, the transformation from the world coordinates to pixel coordinates is obtained by multiplying on the left by an extra factor, leading to:

$$K = \begin{bmatrix} fm_x & & p_x m_x \\ & fm_y & p_y m_y \\ & & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \quad (4.11)$$

This will add an extra degree of freedom to the model.

Lastly, for generalisation, a new parameter is added to the calibration matrix K , the *skew* parameter:

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \quad (4.12)$$

The *skew* is induced by the angle between the axis of the sensor (lack of perpendicularity between axis) and will be zero for most normal cameras. So, the final camera, a *finite projective camera*, P , has 11 degrees of freedom.

As a complement, an introduction to back-projection of points is pertinent. The aim is to define the points in space that map a point $\mathbf{x}$ in the image. Two points of the ray, the camera center C and the point $P^+\mathbf{x}$, where P^+ is the pseudo-inverse of P , $P^+ = P^T(PP^T)^{-1}$. This point lays on the ray because it projects to $\mathbf{x}$, the image point. Defining the ray as the line that joins these two points:

$$\mathbf{X}(\lambda) = P^+\mathbf{x} + \lambda C \quad (4.13)$$

In the case of finite cameras (the center is not in infinity) $P = KR[I | -\tilde{C}] = M[I | M^{-1}p_4]$ with $M = KR$ and p_4 being the last column of P . So, $\tilde{C} = -M^{-1}p_4$. The image point $\mathbf{x}$ backprojects to a ray intersecting the plane at infinity at the point $D = ((M^{-1}\mathbf{x})^T, 0)^T$ so D provides a second point to the ray. So, the line that joins point D and $\tilde{C}$ can be defined as:

$$\mathbf{X}(\mu) = \mu \begin{pmatrix} M^{-1}\mathbf{x} \\ 0 \end{pmatrix} + \begin{pmatrix} -M^{-1}\mathbf{p}_4 \\ 1 \end{pmatrix} = \begin{pmatrix} M^{-1}(\mu\mathbf{x} - \mathbf{p}_4) \\ 1 \end{pmatrix} \quad (4.14)$$

Considering a camera matrix $P = [M | p_4]$ projecting a point $\mathbf{X} = (X, Y, Z, 1)^T = (\tilde{\mathbf{X}}^T, 1)^T$ in a 3-space to the image point $\mathbf{x} = w(x, y, 1)^T = P\mathbf{X}$. Let $C = (\tilde{C}, 1)^T$ be the camera center. Then $w = p^{3T}\mathbf{X} = p^{3T}(\mathbf{X} - C)$ since $PC = 0$ for the camera center C . However, $p^{3T}(\mathbf{X} - C) = m^{3T}p^{3T}(\tilde{\mathbf{X}} - \tilde{C})$ where m^{3T} is the principal ray direction. If the camera matrix is normalized, so that $\det M > 0$ and $\|m^3\| = 1$, then w can be interpreted as the depth of the point $\mathbf{X}$ from the camera center C in the direction of the principal ray (Figure 4.3).

$$\text{depth}(\mathbf{X}; P) = \frac{\text{sign}(\det M)w}{\|m^3\|^T} \quad (4.15)$$

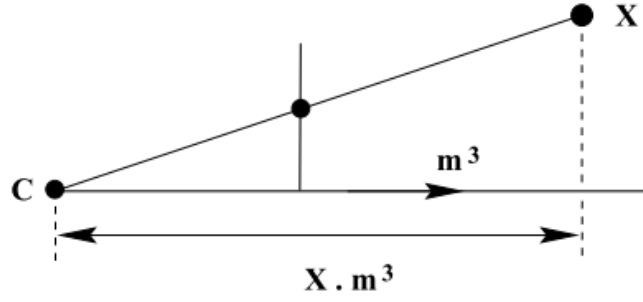


Figure 4.3: If the camera matrix $P = [M|p_4]$ is normalized so that $\|m^3\| = 1$ and $\det M > 0$, and $x = w(x, y, 1)^T = PX$, where $X = (X, Y, Z, 1)^T$, then w is the depth of the point X from the camera centre in the direction of the principal ray of the camera (Image from (65)).

4.2 3D models based on 2D images

There is a vast library of 3D-models for human faces, with data collected with tools such as multi-view stereo and commercial depth sensors. Several approaches tried to extrapolate this information for landmark detection on 2D images (62; 61), however, these methods require a 3D ground truth for each of the images or a pre-trained 3D morphable model. Unfortunately, when working with datasets of animals this information is not readily available. There are several reasons for this, such as the increased difficulty of collecting a 3D-scan from an animal, in particular, if different poses are desired, and the small dimension of the field of research, when compared with the human face analysis area. Therefore, the 3D models available in datasets such as TOSCA (66) have limited realism. Still, there is a large collection of images of animals on the internet. For this reason, deforming 3D models based on these images, with no need for a ground truth 3D model, is appealing. This concept was first introduced by Cashman *et al.* (67). In this paper, a 3D morphable model is defined based on a set of images, with silhouette and landmark annotations. Given an initial 3D model, this will be deformed to adapt to the information of the set of images of the same class (Figure 4.4). Following, Kanazawa *et al.* (68) improved the deformation strategy by having into account the local stiffness of each area, specific for the class. The final model will try to minimize the deformation energy, the location variation between the points in the 3D and 2D and the local stiffness, only distorting the less stiff “tets” (i.e. a tetrahedron of the mesh). Following, Zuffi *et al.* (69) produced a small dataset based on 3D scans of toy figures in arbitrary poses, and, after pose normalization, learned a statistical shape model to fit a combination of 2D keypoints and 2D silhouettes. In none of these works, texture transfer is implemented, with the final shape being a rough estimative of the silhouette and not a direct match point to point between the image and the projection.

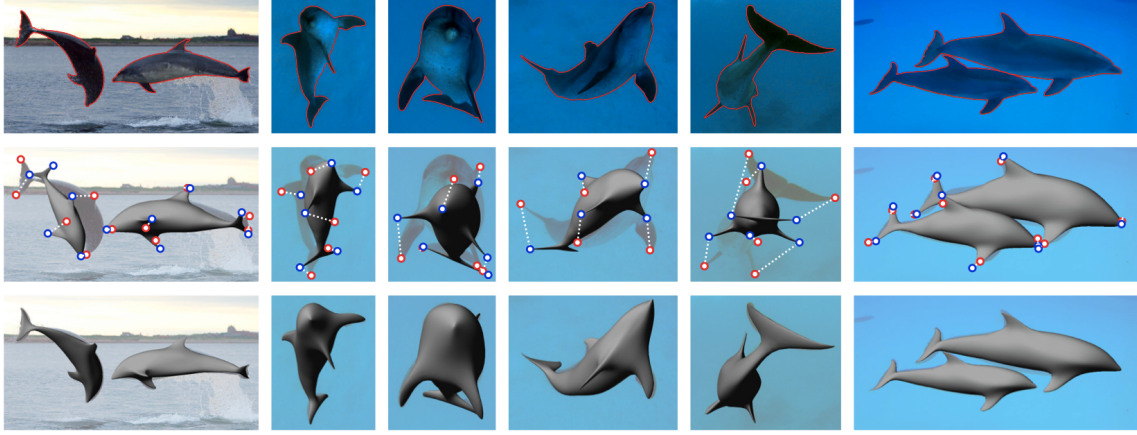


Figure 4.4: Images showing eight dolphins out of 32, from which an 8-parameter morphable model was built. Top row: input images with silhouette annotations. Middle row: The rigid dolphin prototype in initial position for the optimization proposed, showing user-provided point constraints (blue), each of which corresponds to a point in the image (shown in red). Bottom row: final morphable model reconstruction overlaid on the input images (Image from (67)).

4.2.1 Proposed Method

To create 3D heads with a similar texture and shape as the horses in the dataset, it is proposed the use of profile images and correspondent landmarks to deform a pre-existing 3D horse head model from the TOSCA dataset (Figure 4.5). Assuming the horse head is approximately symmetrical, the occluded side will have a similar shape and texture as the one visible.

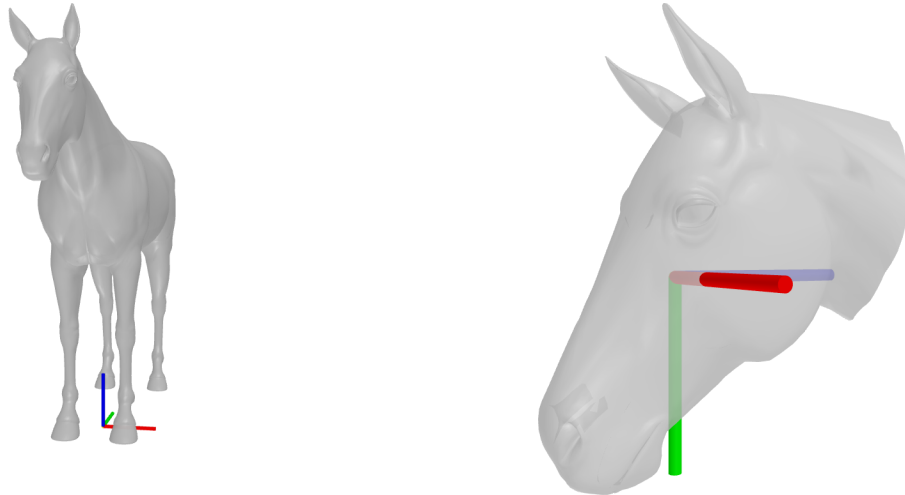


Figure 4.5: Full horse and head model with respective axis (red: x axis, green: y axis, blue: z axis).

For this purpose, the 3D model is annotated with the same landmarking system as the profile faces. A preprocessing of the edge 2D landmarks is made, using an edge detector around each landmark and replacing each point for the closest edge point in the

window. This step is necessary since the resolution of the landmarking is low and they often are deviated from the contour point intended, leading to issues for later texture transfer. Next, the head pose was found based on the 3D-2D point correspondence. The camera calibration matrix, K , was defined as:

$$K = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

Considering a field-of-view of 60 degrees, the focal length, f , was defined as:

$$f = \frac{x_0}{\tan(\frac{60}{2} * \frac{\pi}{180})} \quad (4.17)$$

With (x_0, y_0) corresponding to the image center.

The method used to solve the Perspective-n-Point problem was iterative, based on a Levenberg-Marquardt optimization (70; 71). The resulting rotation matrix, R , and translation vector, t , allowed the projection of the 3D model points into the image frame:

$$\begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = K[R|t]\mathbf{X} = K\mathbf{X}_{cam} \quad (4.18)$$

With (x, y) corresponding to the pixel coordinates of the world point $\mathbf{X}$ projected into the image.

Assuming the w of each ground truth landmark in the image is the same as the one of the correspondent projected 3D point, the image point in the camera frame is:

$$\mathbf{X}_{cam} = K^{-1} \begin{pmatrix} wx \\ wy \\ w \end{pmatrix} \quad (4.19)$$

The landmarks were then projected according to the symmetry plane, and a Thin-Plate-Spline (TPS) approach (72) was applied to deform the 3D model initialized according to the ground truth landmarks and their projections. This method interpolates surfaces over scattered data. This is done by having a fixed set of nodes in the plane, in this case, the landmarks, and trying to minimize the bending energy by warping the surface to fit the ground truth. This method is usually applied to 2D images but can be used in more dimensions. Here, it is used in 3D.

Let $P_1 = (x_1, y_1, z_1)$, $P_2 = (x_2, y_2, z_2)$, ..., $P_n = (x_n, y_n, z_n)$, be the n "source" points and $r_{ij} = |P_i - P_j|$ the distance between points i and j . The following matrices can be defined:

$$K_s = \begin{bmatrix} 0 & U(r_{12}) & \dots & U(r_{1n}) \\ U(r_{21}) & 0 & \dots & U(r_{2n}) \\ \dots & \dots & \dots & \dots \\ U(r_{n1}) & U(r_{n2}) & \dots & 0 \end{bmatrix}, n \times n \quad (4.20)$$

with $U(r) = r^2 \log r^2$.

$$P_s = \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ \dots & \dots & \dots & \dots \\ 1 & x_n & y_n & z_n \end{bmatrix}, 4 \times n \quad (4.21)$$

and

$$L = \left[\begin{array}{c|c} K_s & P_s \\ \hline P_s^T & O \end{array} \right], (n+4) \times (n+4) \quad (4.22)$$

Where the operator O is a 4×4 matrix of zeros.

Considering $V = (v_1, \dots, v_n)$ an n -vector with the “destination” points and $Y = (V \mid 0 \ 0 \ 0)^T$, a column vector of length $n+4$:

$$L^{-1}Y = (W \mid a_1 \ a_x \ a_y \ a_z)^T \quad (4.23)$$

With $W = (w_1, \dots, w_n)$ and the coefficients a_1 , a_x , a_y and a_z being the parameters of the transformation between the source and destination location. This transformation is described by:

$$f(x, y) = a_1 + a_x x + a_y y + a_z z + \sum_{i=1}^n w_i U(|P_i - (x, y, z)|) \quad (4.24)$$

Given a new set of points $M = (m_1, \dots, m_n)$, $r \times r$ the previous transformation can be apply:

$$K_d = \begin{bmatrix} U(|m_1 - P_1|) & \dots & U(|m_1 - P_n|) \\ U(|m_2 - P_1|) & \dots & U(|m_2 - P_n|) \\ \dots & \dots & \dots \\ U(|m_r - P_1|) & \dots & U(|m_r - P_n|) \end{bmatrix}, r \times n \quad (4.25)$$

$$P_d = \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ \dots & \dots & \dots & \dots \\ 1 & x_r & y_r & z_r \end{bmatrix}, 4 \times r \quad (4.26)$$

Being the new location of points M , given by:

$$M_d = K_d P_d (W \mid a_1 \ a_x \ a_y \ a_z)^T, r \times 3 \quad (4.27)$$

After deforming the 3D face model, the correspondence between the vertices in both sides of the symmetry plane was obtained, for further texture transfer. Then, the colour of each triangle of the mesh was defined as the mean between the colour of each of its



Figure 4.6: Examples of deformation and further texture transfer of the 3D model.

vertices and the triangle centroid, attributing the same colour to symmetrical triangles. The colour map of each annotated horse was saved allowing the interchange of texture between the different horse head shapes in the subset (Figure 4.6). In total, 29 colour maps and 3D shapes were collected.

To generate synthetic images with a common reference, the 3D points were converted to the world frame:

$$\mathbf{X} = [R|t]^{-1}\mathbf{X}_{cam} \quad (4.28)$$

The rotation matrix, R , was then defined as (73):

$$R(\alpha, \beta, \gamma) = R_z(\alpha)R_y(\beta)R_x(\gamma) \quad (4.29)$$

Where:

$$R_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.30)$$

$$R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \quad (4.31)$$

$$R_x(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix} \quad (4.32)$$

With a *yaw* being the counterclockwise rotation of α about the y -axis, a *pitch* being the counterclockwise rotation of β about the x -axis and a *roll* being the clockwise rotation of γ about the z -axis.

The background was then replaced with images extracted from *Flickr* under the tags: “farm”, “field”, “barn” and “stable” (Figure 4.7).

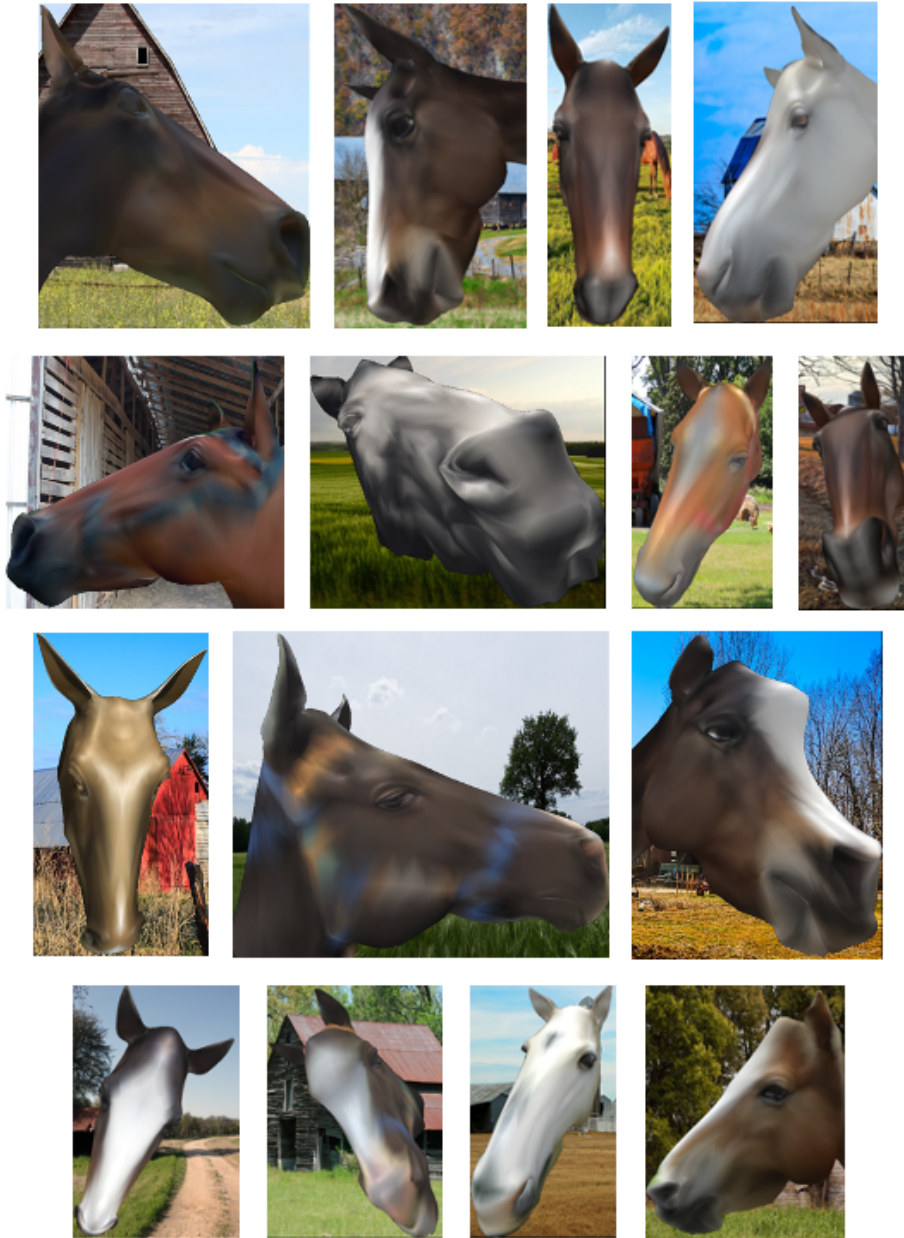


Figure 4.7: Examples of synthetic images produced using the described method. Examples of different head shapes with the different head pose with the same texture are presented, showing the effects of texture transfer.

Chapter 5

Experimental Evaluation

In the present chapter, different aspects of the equines dataset are described, including the origin of the images, the landmark annotation system and the quantitative head pose grouping. Then, pose estimation and landmark location systems are explored, evaluating the effects of data augmentation with synthetic images on the model performance. Lastly, a pain estimation model is described, evaluating the potential sources of error and suggesting future improvements. The proposed pipeline is presented in Figure 5.1.



Figure 5.1: Proposed pipeline for horse pain estimation based on facial features

5.1 The Horse and Donkey Faces Dataset

The dataset consists of a total of 1855 and 531 images of horses and donkey faces, respectively (20; 28). Moreover, all images have pain score and landmarks annotations following the criteria described in the following section. According to the origin of the images, the dataset was divided into three subsets (Figure 5.2):

- **Horses from the Netherlands - HFN (1521 images):** Images provided by horse owners all over the country. The photographs are very diverse, showcasing an extensive set of backgrounds, breeds, and image resolution.
- **Horses with clinically induced injuries - HWI (334 images):** The images were collected as part of a project running in the Faculty of Veterinary Medicine of the Utrecht University. In this study, clinically induced injuries were inflicted on a set

of horses and photographs were taken in different time-periods. The images have similar backgrounds with alike illumination and resolution. Additionally, there are multiple pictures of the same horse with different head poses and with several timestamps.

- **Donkeys from a Donkey Sanctuary - DFS (531 images):** Images provided by a donkey sanctuary in the UK. Similarly to what is described in the HWI subset the images have identical characteristics with multiple photographs of the same donkey.

During the following sections, the HWI and HFN subsets will be used as a combined dataset for horse pose estimation and landmark detection.

5.1.1 Landmarks annotations

The annotation schemes proposed in previous work (20) describe the head shape and facial features in great detail. Three different shapes were established, considering a qualitative evaluation of the head pose with 54, 44 and 45 annotated landmarks for frontal, tilted and profile view, accordingly (Figure 5.2).

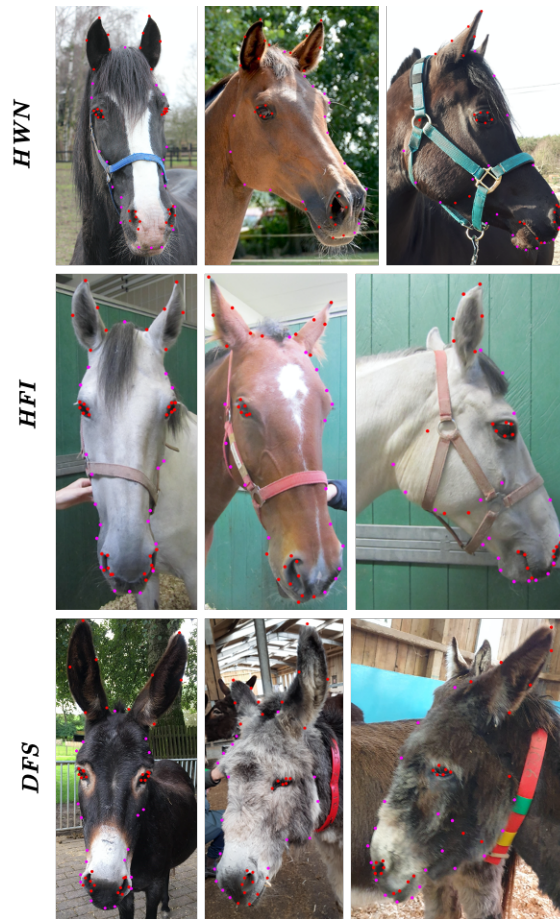


Figure 5.2: Faces extracted from each subset with marked points of interest. First column - Frontal view; Second column - Tilted view; Third column - Profile view; Red points indicate the landmarks of interest; Magenta points indicate outline landmarks.

5.1.2 Pose distribution

Using the weak perspective projection method described in Chapter 4 the ground truth for the quantitative head pose of each image was defined. The landmark annotations were divided into two sets “absolute landmarks”, in which their position doesn’t change with rotation, and “relative landmarks”, outline landmarks that will change with pose variations (Figure 5.3). Only the first set was used to estimate the head pose. The images in the dataset have different origins, with some being cropped, and so the estimates for the camera intrinsic parameters were not always accurate. For this reason, all observations were checked manually, excluding the inadequate images. The resulting dataset has 370 frontal images, 952 tilted images and 438 profile.



Figure 5.3: Reduced landmarking system with solely the “absolute landmarks” for each pose class. Blue: landmarks not included in the new landmarking system; Magenta: reduced landmarking system.



Figure 5.4: Quantitative head pose axis in horses (red: x axis, green: y axis, blue: $(-)$ z axis).

The pose distribution in each class is presented in Figure 5.5. Analysing the yaw values of each class (Figure 5.5) it is noticeable that the yaw values for the “frontal” class are restricted in the $[-25, 25]$ degrees range, the yaw values for the “profile” class have an absolute value predominantly in $[50, 90]$ degrees range and the “tilted” class overlaps with both “frontal” and “profile” classes, with a diverse set of yaw values, ranging mainly from an absolute value of 10 to 75 degrees. The $roll$ and $pitch$ distribution is similar in all classes, with higher variance in the profile class, likely due to the error associated with the head pose assessment since the 2D points used to predict it are in similar planes being the 3D-2D correspondence ambiguous.

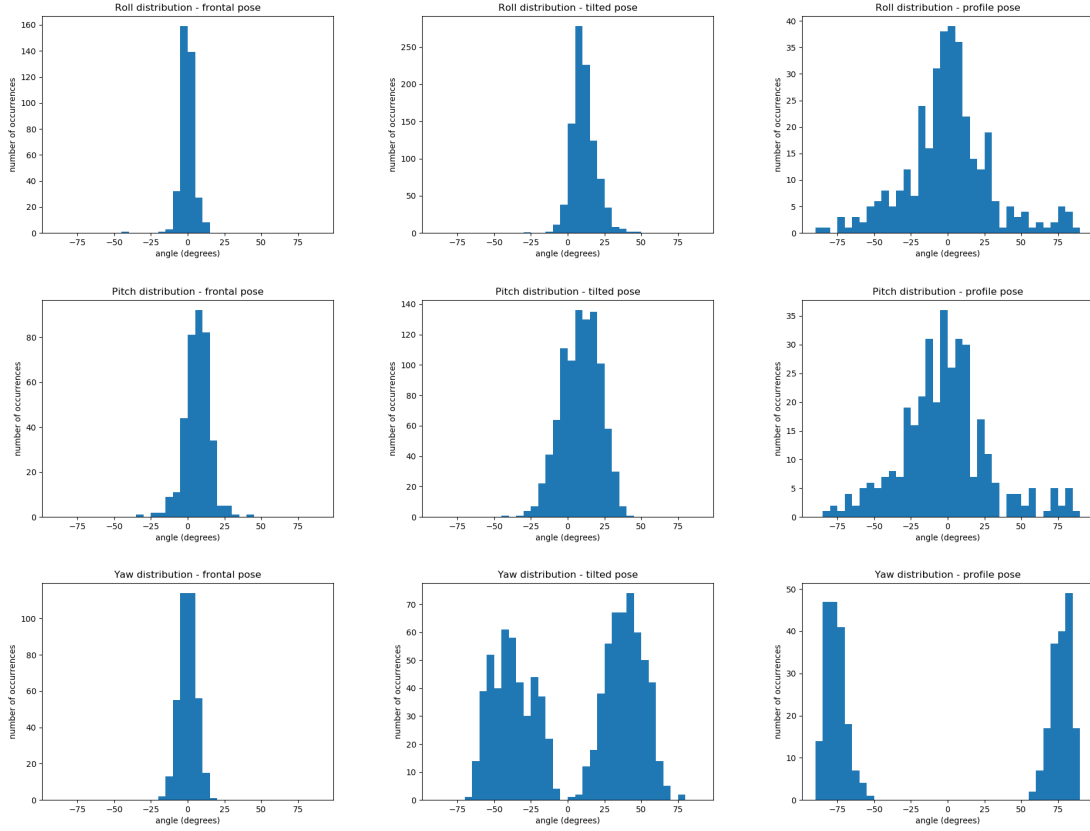


Figure 5.5: Distribution of the pose values in the updated dataset. From top to bottom: *roll*, *pitch*, *yaw* values. From left to right: “frontal”, “tilted” and “profile” classes.

5.1.3 Pain annotations distribution

The distribution of the pain scores following the grimace scale presented in Table 2.2 was evaluated. As observed in Figure 5.6, the dataset is highly unbalanced, with very few instances of class “2” (“pain obviously present”) in all regions-of-interest. The predominant class will alternate between “0” (“pain not present”) and “1” (“pain moderately present”).

Note that the used grimace scale is not explicitly head-pose dependent, except in the case of occluded regions of interest. Also, there is an overlap between the head pose of the three quantitative pose classes defined in previous work (28). As a result, a 3-fold cross-validation and a test set were defined, maintaining the proportion of pain score combinations and quantitative *yaw* values in each subset aiming to develop a unique pose-informed pain assessment model for each region-of-interest. The complete training set contained 259 frontal faces, 666 tilted faces and 243 profile faces, with the test set containing 111 frontal faces 286 tilted faces and 105 profile faces.

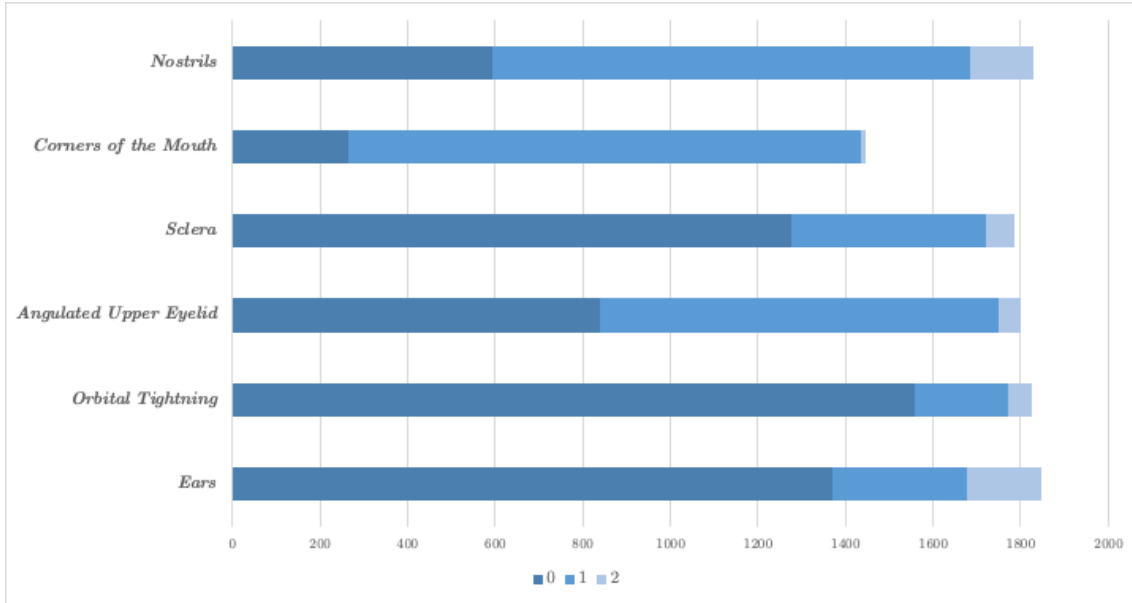


Figure 5.6: Distribution of pain scores on the dataset.

5.2 Pose Estimation

5.2.1 Methods

Head pose variations cause evident changes in the facial appearance of equines due to self-occlusion. For this reason, pose-informed methods for landmark detection and further pain estimation are pertinent, allowing a better description of the areas of interest visible from a specific pose. So, we propose the implementation of a 5-class pose classifier to distinguish frontal, tilted and profile faces, as well as the sign of their rotation. Based on results from the literature, Histogram of Oriented Gradients (HOG) features (26) were used with a Support Vector Machine classifier (SVM).

All horse faces were cropped according to their landmarks. For data augmentation purposes, tilted and profile faces were flipped horizontally reducing the data imbalance between the two tilted and profile classes when compared to the frontal class. After that, each face was resized to 128×128 and a 3-fold cross-validation was performed to adjust the HOG parameters and the classifier used. The higher weighted F1-score was achieved with 9 orientations, (8,8) pixels per cell and (4,4) cells per block, associated with a weighted SVM classifier with a linear kernel.

Knowing the quantitative head pose (i.e. the *yaw*, *pitch* and *roll* values) can be helpful to improve both landmark detection and pain estimation, since the face appearance varies widely with these values. For this purpose, a multi-loss convolutional neural network was used (74) for head pose estimation. This approach combines a ResNet50 architecture (75) with the Mean Squared Error and Cross Entropy Losses. Different amounts of data augmentation were tested, introducing synthetic images on the training set. A 3-fold cross-validation was performed, choosing the best model according to the error in the correspondent unaugmented validation set. Considering the differences in appearance between the synthetic and real images, it is desirable to maintain predominantly real images in the training set to avoid overfitting to the synthetic image's properties. For this

reason, the number of synthetic images generated per pose bin (n_{aug}) was defined as the number of images in the training set in that bin (n_{bin}), multiplied by an augmentation factor determined as a function of the maximum number of images in a pose bin in the training set (n_{max}). The pose bins with the maximum number of images will not be augmented. The data was augmented according to the *yaw* representation in the training set:

$$n_{aug} = n_{bin} \times \left[\left(\frac{n_{max}}{n_{bin}} \right)^\alpha - 1 \right] \quad (5.1)$$

The results presented refer to the performance on the unaugmented test set, with a total of 522 horse faces.

5.2.2 Experimental Results

The implemented qualitative pose classifier proved to be efficient in distinguishing the five pose classes, especially when considering the wide variety of head poses represented in the dataset. Table 5.1 summarizes the results in terms of precision, recall, and F1-Score. The errors observed were often related to ambiguous head poses or extreme angles, in particular, the *pitch* angle.

Table 5.1: Pose estimation evaluation on the test set containing 342 images of horse faces. The *yaw* angle sign is presented when needed. The performance is separately presented for each pose bin as well as the weighted average of each performance metric, according to the pose distribution. For the individual performance metrics, the class of interest was labelled as “positive” and all the others as “negative”, calculating the Precision, Recall, and F1-Score.

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Profile (−)	0.81	0.85	0.83
Tilted (−)	0.85	0.90	0.87
Frontal	0.90	0.76	0.82
Tilted (+)	0.91	0.95	0.93
Profile (+)	0.90	0.90	0.90
Weighted Average	0.88	0.88	0.88

Regarding the quantitative pose prediction, on a first instance, the possibility of fine-tuning the Hopenet model trained on the Sheep dataset (27) was evaluated versus the model trained on the 300W-LP dataset (76). The model was trained for 50 epochs and the best model was chosen based on the validation set results. The performance metrics used were the Mean Absolute Error (MAE), calculated in degrees (out of 180 degrees), the Pearson’s Correlation Coefficient (PCC) that measures the correlation between predictions and the ground truth and the Sign Agreement metric (SAGR). It is desirable to achieve a low MAE and high PCC and SAGR scores.

As expected, there is a significant improvement in the model’s performance when transfer learning from the sheep-based model (Tables 5.2 and 5.3), particularly in the *yaw* values. This is due to the Sheep model being trained with a focus on the *yaw* values, dividing the dataset and augmenting the data based on it. The sheep faces will have

similar behaviour as horse faces when changing pose, tackling the problems related to the elongated nose and consequent self-occlusion. Similarly to what happens in the present dataset, the diversity of *roll* and *pitch* angles is reduced and so the performance associated with these angles is lower.

Table 5.2: Quantitative pose estimation results in the test set transfer learning from the model trained on the 300W-LP dataset. A satisfactory performance will be defined by an low MAE and high PCC and SAGR scores.

	<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>	<i>Average</i>
MAE	38.53	12.38	10.07	20.33
PCC	0.09	0.22	0.09	0.13
SAGR	0.51	0.70	0.72	0.65

Table 5.3: Quantitative pose estimation results in the test set transfer learning from the model trained on the Sheep dataset. A satisfactory performance will be defined by an low MAE and high PCC and SAGR scores.

	<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>	<i>Average</i>
MAE	12.50	11.68	9.33	11.16
PCC	0.89	0.24	0.21	0.46
SAGR	0.83	0.74	0.74	0.76

Next, we have evaluated the effect of data augmentation. Different values for α were tested (Equation 5.1), finding the best performance with $\alpha = 0.5$ (Table 5.4). Note that the model was chosen based on an unaugmented validation set, and consequently, with an overly augmented training set a big decrease in training loss is observed, accompanied by an increase in validation loss, due to the different origin of the images from each set. When compared to the results in Table 5.3 there is an improvement in performance in each angle, in particular, the *yaw* angle, with a decrease of MAE and an increase in both PCC and SAGR. Although the MAE for the *roll* and *pitch* angles is similar to the *yaw* one, their PCC is significantly lower. This fact can be explained by the smaller range of values for the *pitch* and *roll* (mainly between $[-25, 25]$ degrees instead of $[-90, 90]$ degrees) and the estimation error associated to the ground truth head pose, in particular for the “profile” class, described in Section 5.1.2. Considering the diversity of poses in the dataset, and the error associated with the ground truth pose estimation, the presented results are satisfactory, with a high signal agreement and correlation. A visual representation of the error is presented in Figure 5.7.

Table 5.4: Quantitative pose estimation results in the test set transfer learning from the model trained on the Sheep dataset with an augmented training set ($\alpha = 0.5$).

	<i>Yaw</i>	<i>Pitch</i>	<i>Roll</i>	<i>Average</i>
MAE	9.50	11.43	9.25	10.06
PCC	0.94	0.35	0.26	0.51
SAGR	0.86	0.75	0.76	0.79

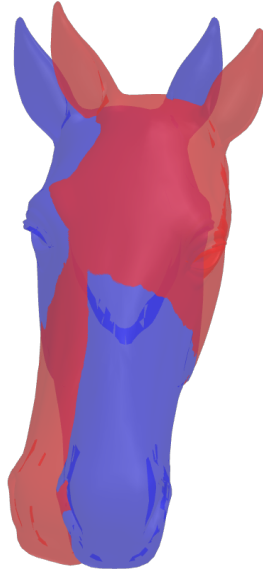


Figure 5.7: Visual representation of the error in pose estimation. Blue: a ground truth pose Red: visualized pose with the MAE for yaw, pitch and roll.

5.3 Landmarking

5.3.1 Approach

For landmarking, two state-of-the-art landmark detection algorithms are compared. These algorithms, Ensemble of Regression Trees (ERT) (54) and Supervised Descent Model (SDM) based on SIFT features (55), were originally developed for detecting landmarks on human faces.

To assess the landmarking algorithms, we assume that the face detection is correctly performed. The horse faces were cropped based on the ground truth landmark locations and resized according to the face proportions in each of the three classes (i.e. “frontal”, “tilted” and “profile”) of the training set. For each pose bin, the height of the face was scaled to 600 *px*, with the width being defined according to the training set aspect ratio. This resulted in 600 × 270 “frontal” images, 600 × 330 “tilted” images and 600 × 380 “profile” images. Furthermore, all faces were rotated according to the absolute *yaw* angle.

A 3-fold cross-validation was performed to adjust the number of perturbations to apply in each pose class. In this scenario, perturbations correspond to the addition of uniform noise to the shape within the bounding box. A performance plateau of the ERT

model was observed when increasing the number of perturbations above 30. In this scenario, Additionally, a mean shape model was calculated for each pose class based on the training set shapes.

In human landmark localization, automatically located landmarks within 10% of the inter-ocular distance are considered accurate (18; 15). Since there is not a standard normalization factor for landmark analysis in horses, the use of the distance between the centre of an eye and the centre of the underlying nostril is proposed for this purpose. The suggested accuracy threshold is 6% (Figure. 5.8). These two features are present in every pose of the horse face, unlike the two eyes, and their distance is sufficiently long to make it robust against errors. So, the performance metrics used in the present Section are the Mean Normalized Error (MNE), correspond to the Euclidean distance between the prediction and ground truth, normalized by the eye-nostril distance and the Success Rate (SR), referencing to the percentage of predictions with an error lower than 6%. Still, variation in pose (especially, changes in *pitch*) will distort the eye-nostril distance and, if dealing with a dataset with diverse breeds, the nose length in proportion to the face will vary.



Figure 5.8: Illustration of the eye-nostril distance used for normalization (green line). Limits of error surrounding a ground truth landmark (red dot) are given in yellow (2%) and in blue (6%) (Image from (28)).

Next, we evaluate the performance of the best performing model when using solely the “absolute landmarks”. These landmarks do not change with the head position (i.e. the outline landmarks are excluded, as moving the face changes the absolute position of the face contour). Data augmentation was implemented in this experiment, by generating synthetic images according to the MNE obtained in each validation fold of the cross-validation. The number of images generated for a particular *yaw* pose bin was:

$$n_{aug} = n_{bin} \times [(\frac{MNE_{bin}}{median_{MNE}})^\alpha - 1] \quad (5.2)$$

with the $median_{MNE}$ corresponding to the median of all the MNE for the *yaw* bins in each fold. Different experiments were made to achieve the most suitable equation, choosing the median MNE as the normalization factor, for instance instead of the minimum MNE, as the median is less sensitive to outliers. Additionally, various values for α were tested.

The results presented in the following section refer to the performance on the unaugmented test set, with a total of 111 frontal faces, 286 tilted faces, and 105 profile faces.

5.3.2 Experimental Results

The Ensemble of Regression Trees (ERT) model has shown promising results for the landmark localization, outperforming both the Mean Shape model and the Supervised Descent Model (Table 5.5). As expected, considering the qualitative head pose is unknown and the quantitative pose is only based on the *yaw* angle, the variations in the normalization distance with the *pitch* angle introduced variation in the final MNE. Overall, extreme angles, combined with a lack of representation of these angles in the dataset, resulting in incorrectly located landmarks. Additionally, not all outline landmarks are associated with strong changes in appearance, which leads to deviations on their prediction, visible in Figure 5.9. It's also important to note that there is a clear improvement in performance when applying a train-test split based on the *yaw* angle values, with a decrease in the MNE of 2% in both the ERT and SDM model and an improvement of 0.07 in the success rate of the ERT classifier. This fact reflects a decrease of the MNE for regions of interest.

Table 5.5: Mean Normalized Error and Success Rate in Horses using ERT, SDM and a baseline mean shape model. The values presented are the weighted average results for the test set for the three qualitative pose classes. The Success Rate indicates the percentage of landmarks with a location error less than 6% of eye-nostril distance.

	<i>ERT</i>	<i>SDM</i>	<i>Mean Shape</i>
MNE	0.070	0.087	0.130
Success Rate	0.590	0.430	0.210

Table 5.6: Mean Normalized Error per region-of-interest (ROI) in the test set. The highest error for each region of interest is highlighted. The missing values indicate that the ROI in question is not defined in the landmark system for that pose class.

	<i>Frontal</i>	<i>Tilted</i>	<i>Profile</i>
Ear(s)	0.056	0.061	0.100
Nostrils	0.083	0.071	0.069
Left eye	0.060	0.042	0.079
Right eye	0.051	-	-
Mouth	-	0.056	0.062

Analyzing the individual results for each region of interest (ROI) (Table 5.6) in combination with the examples presented in Figure 5.9, additional sources of error are observed.

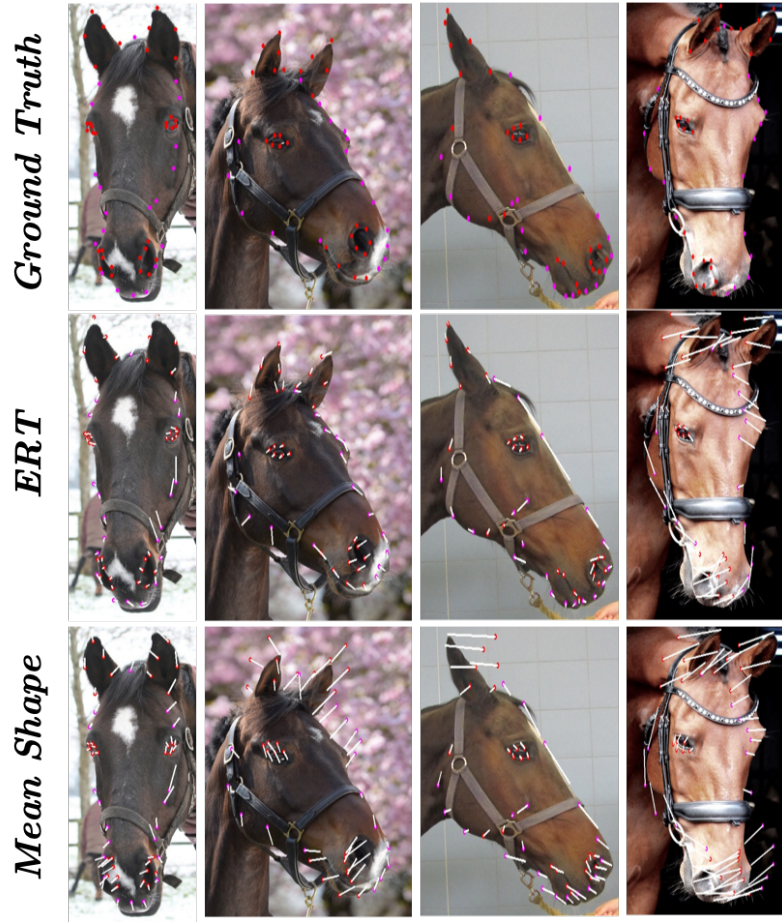


Figure 5.9: Examples of the landmarks located in the test set by the ERT model compared to the baseline mean shape model. The three pose bins are represented from left to right: frontal, tilted and profile, respectively. The last column shows an image with large errors due to extreme pitch angle. The white lines connect the predicted point with the ground truth landmark location.

Ear position varies greatly from horse to horse and with the pain level, so, although they commonly look straight and facing forwards, there are several examples where this ROI is rotated, in a low position near the head. These variations make the precise location of all the landmarks in the ears more challenging and so, a higher error in comparison with the other ROIs was expected, in particular in the profile pose. Regarding the nostrils, the subtle outer contour, when compared to the nostrils' dark interior, leads to landmark deviation towards it, visible in the second column of Figure 5.9. Similar to what was described for the nostrils, eye landmark detection is affected by the clear contrast between the outer eye and the pupil when compared with the outer eye landmarks. Lastly, the lack of a clear definition in the mouth landmarks result in deviations in the landmark location, similar to the outline landmarks.

Due to the difference between the normalization factor used and the ones applied in Section 3.5, it's not possible to directly compare with the literature. Thus, to compare the performance of our model with the animal landmark detection literature, we further normalized the errors with the mean edge length of the bounding box, similar to Hewitt

Table 5.7: Mean Normalized Error in the reduced landmarking system in the test set.

	<i>Frontal</i>	<i>Tilted</i>	<i>Profile</i>
ERT	0.066	0.063	0.068

et al. (24). The Mean Normalized Error achieved for the ERT model was 0.05, comparable to the ones presented in the related work.

Considering that the outline landmarks are highly variable and do not have a direct appearance correlation, an ERT model was trained with solely the “absolute landmarks”. This will allow the use of the synthetic images for data augmentation since the 3D model was annotated based on this invariable set of landmarks. The results obtained were comparable with the ones of the previous model (Tables 5.7 and 5.8) and so this landmark system was used for the following process. The “tilted” landmark system had the highest increased in MNE, likely due to its extremely diverse characteristics, with a big range of *yaw* angles.

Table 5.8: Mean Normalized Error per region-of-interest (ROI) in the test set with the reduced landmarking system. The highest error for each region of interest is highlighted. The missing values indicate that the ROI in question is not defined in the landmark system for that pose class.

	<i>Frontal</i>	<i>Tilted</i>	<i>Profile</i>
Ear(s)	0.062	0.064	0.092
Nostrils	0.078	0.077	0.060
Left eye	0.055	0.033	0.069
Right eye	0.051	-	-
Mouth	-	0.060	0.055

The MNE variation for each head pose was analysed in a 3-fold cross-validation. Large variability in the MNE per angle are observed (Figure 5.10).

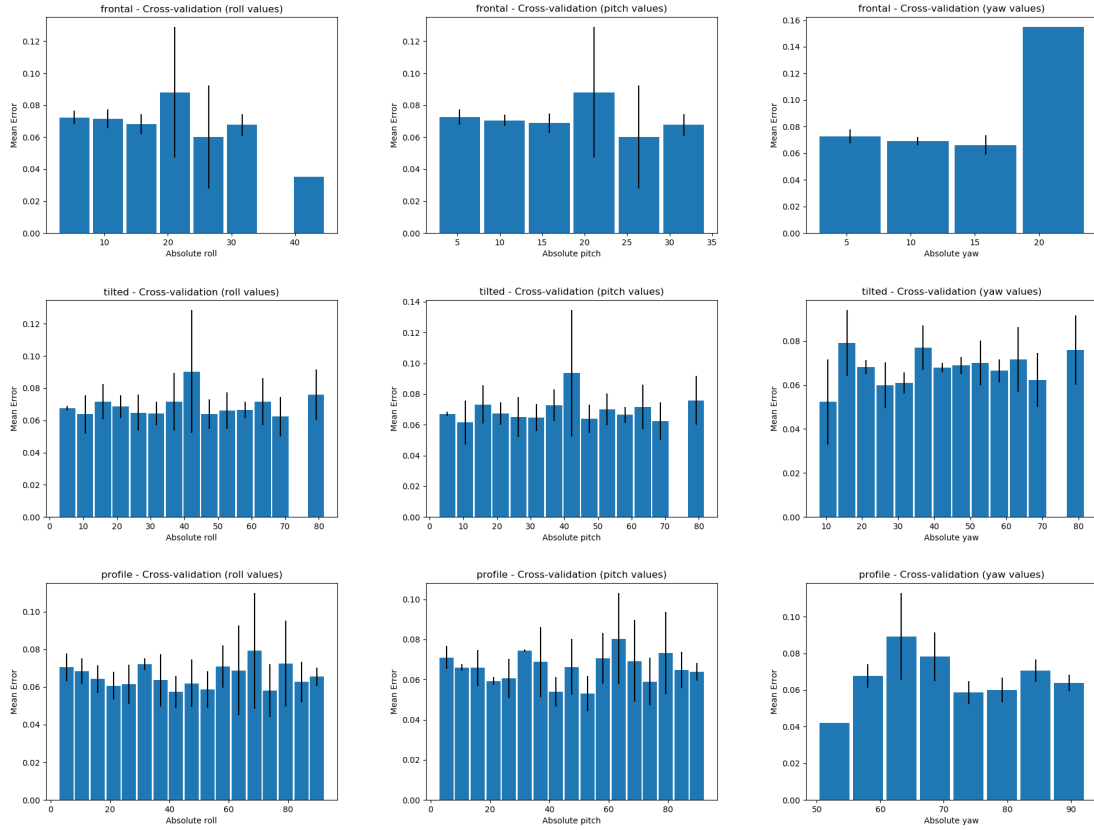


Figure 5.10: Average MNE distribution for the pose values. From the top to the bottom: “frontal”, “tilted” and “profile” classes. From left to right: *roll*, *yaw*, *pitch* values.

Different data augmentation factors are tested, experimenting with adding perturbations to the augmented data or not. A decrease on the 3-fold cross-validation MNE was observed when used an $\alpha = 2$ and 20 perturbations for the frontal faces, and $\alpha = 1.5$ and 10 perturbations for the profile and tilted (Table 5.9). Following, a full model was trained using the complete training set and augmenting the data according to the average MNE per *yaw* bin in the cross-validation.

Overall there is an improvement in the performance when training the “frontal” and “profile” model with a synthetic component, however, the increase is not as high as expected. The “tilted” model wasn’t improved. This can likely be explained by the fact the “tilted” class is the most diverse in terms of *yaw* angle, and so the MNE per bin, used for data augmentation, were not representative of the overall data behaviour. Also, the landmarking system used is not ideal for the range of head poses that this class includes. A global landmarking system could improve performance by using the quantitative pose to divide the images into classes.

Other potential sources of error for the ERT models, particularly after data augmentation, may be the strong structural constraints of the model as opposed to weak feature constraints and the inconsistencies in manual annotations, especially in small areas. These inconsistencies may introduce incompatibility between the automatic augmented

Table 5.9: Mean Normalized Error per region-of-interest (ROI) in the test set with the reduced landmarking system and data augmentation. The MNE that maintain or increase in relation to the previous model are highlighted. The missing values indicate that the ROI in question is not defined in the landmark system for that pose class.

	<i>Frontal</i>	<i>Tilted</i>	<i>Profile</i>
Ear(s)	0.058	0.066	0.094
Nostrils	0.075	0.079	0.057
Left eye	0.056	0.033	0.063
Right eye	0.046	-	-
Mouth	-	0.061	0.053

image annotations and the manual real image annotations.

Lastly, regarding the ears, the lack of “anatomical” points associated with the annotations for the base of the ears, will make these landmarks particularly difficult to assess, especially for the “tilted” and “profile” poses. The lack of realism of the proposed augmentation technique may as well limit the amount of information that can be learned from the synthetic images. Better data augmentation with independent ear movement and more realistic rendering can overcome the aforementioned problem.

5.4 Pain Estimation

5.4.1 Methods

Considering the full pipeline is aimed for a real-time application, the use of simple appearance and geometric features was explored, evaluating the performance of SVM models, trained with the quantitative pose, the local rotation angle of each region-of-interest (ROI), and HOG features (27). Since there is no clear division between the different qualitative head pose bins defined, we proposed a unique pose-informed model per ROI.

On a first instance, each ROI was normalised, rotating the ears and nostrils into a vertical position and the eyes and mouth into a horizontal position (Figure 5.11). The rotation angles were used for model training. Each ROI was resized based on the mean ratio in the training set, with the biggest side as 128 *px*. Different values for the HOG parameters (orientations, cells-per-block, pixels-per-cell) were tested in a 3 fold cross-validation, as well as different kernels for the SVM model (linear, rbf and polynomial). Each model was trained with the right and left ROI’s HOG features, angles and head pose. In case an ROI was not visible it was replaced by a (200,200,3) zero array, with rotation equal to zero. The best performance was achieved with SVM models with a linear kernel, (8,8) pixels per cell and (3,3) cells per block for the ears model, (2,2) cells per block for the orbital and sclera model, (1,1) cells per block for the eyelid and mouth models and (4,4) cells per block for the nostrils model.

As shown in Section 5.1.3 the dataset is highly unbalanced, and so, in addition to using a balanced SVM model, the performance metrics presented in the following section are weighted, with the metrics for each label being calculated separately and then finding



Figure 5.11: Normalisation of the regions-of-interest for the "frontal" class.

Table 5.10: Performance of the pain estimation models considering three classes.

	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>F1-score MV</i>
<i>Ears</i>	0.66	0.69	0.67	0.64
<i>Nostrils</i>	0.56	0.56	0.55	0.56
<i>Orbital</i>	0.76	0.77	0.77	0.80
<i>Eyelid</i>	0.45	0.46	0.45	0.38
<i>Sclera</i>	0.55	0.58	0.57	0.56
<i>Mouth</i>	0.63	0.66	0.65	0.65

the average weighted value.

The final model was training on the complete training set with 1168 images and evaluated the trained model on the test set with 502 images. The number of occurrences of each ROI is variable.

5.4.2 Results

High pain level ("2") class has much fewer examples compared to moderate pain ("1") and no pain ("0") classes. We present the performance on both 3-class pain estimation task using all three classes (Table 5.10), and the binary pain estimation task combining class "1" and "2" one class (Table 5.11). Note that the problem of unbalanced data is not solved entirely with the binarization of the problem. The weighted F1-score results are comparable or superior to the baseline majority vote classifier. However, the F1-score of the underrepresented class was significantly lower than the one of the majority class, suggesting that further improvements should be made.

Potential explanations for this fact are the overall ambiguity of the classifications, with

Table 5.11: Performance of the binary pain estimation models.

	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>F1-score MV</i>
<i>Ears</i>	0.72	0.73	0.72	0.64
<i>Nostrils</i>	0.56	0.56	0.56	0.56
<i>Orbital</i>	0.78	0.77	0.78	0.80
<i>Eyelid</i>	0.48	0.48	0.48	0.41
<i>Sclera</i>	0.59	0.61	0.60	0.56
<i>Mouth</i>	0.64	0.66	0.65	0.66

significant disagreement among experts (Figure 5.12), the noticeable data imbalance in most classes, even after combining class “1” and “2” and the subtlety of the changes to perceived. Lastly, the dataset has a lot of different breeds with different face morphology and proportions which makes the comparison between facial features more difficult. As an example, it is clear that different breeds can have very distinct nostril shapes which makes it difficult to assess if they are “relaxed” or “open” when using a mixed dataset.

Previously published results on this dataset were not replicable (11).



Figure 5.12: Agreement in the annotations made by three specialists in 1655 images of horse faces. The annotations from “expert 2” and “expert 3” were compared with the ones from “expert 1”, used to train the models. From top to bottom: agreement for “no pain”, agreement for “moderate pain” and agreement for “obvious pain”.

Chapter 6

Conclusions and Future Work

Continuous monitoring of pain in equines would provide a helpful tool for both clinicians and care-keepers, namely, aiding the evaluation of disease progression and assessing the effect of medication. Although existing grimace scales proved to be accurate at quantifying the levels of pain, based solely on the horse facial expressions, having to manually assess pain scores for multiple action units can be a drawback to their use. So, a grimace scale based system for pain detection would allow an efficient pain estimation with the advantage of detecting subtle changes in it, more noticeable when using a facial expression approach instead of behavioural analysis. However, even though a lot of work has been done in emotion analysis in humans, including pain, building a similar model for equines reveals new challenges, namely, self-occlusions, variability in breed and colours and data scarcity.

Automatic landmark detection is important to identify the regions of interest presented in the grimace scale, in this case, EQUUS-FAP. Yet, variations in the head pose, in particular, the *yaw* angle, will lead to significant changes in visibility and overall head silhouette. For this reason, a simple HOG based classifier is proposed aiming to divide the horse faces according to their qualitative pose. Further, a CNN-based quantitative pose estimation system was implemented. Afterwards, the facial landmarks were located using a Pose-informed Ensemble of Regression Trees system. Additionally, a novel data augmentation system was proposed, deforming a simple 3D-horse head model according to 2D landmarks with texture transfer from the images. This allowed the creation of diverse synthetic images with precise landmarking and known quantitative pose. Lastly, a pose-informed pain estimation system was developed, introducing an SVM model for each region-of-interest trained based on geometric features, the head pose and the HOG features extracted from the bounding box defined by the landmarks.

The 5-class pose classifier achieved promising results in horses with a 0.88 F1-Score, implying there is a clear change in appearance with head pose variations. Further, after data augmentation, the CNN-based quantitative pose estimator achieved a high performance for the *yaw* angle, with only 9.51 degrees as Mean Absolute Error. For the landmarking system, the mean normalized error was 0.07. The data augmentation decreased the mean normalized error in the majority of regions-of-interest. Potential sources of error, are the subtle appearance associated with several landmarks, especially when near a strong edge, and the variations of the *pitch*, not taken into account when

performing the qualitative pose division. Additionally, the shape constraints of the model may be too strong for the variations in head morphology observed in the dataset due to different breeds. The pain estimation system overcame the baseline, yet, the strong data imbalance between “pain” and “no pain” annotations introduces limitations to its performance. Additionally, the level of disagreement among experts in the “1” and “2” pain scores is high, on average 44% and 48% respectively, suggesting some ambiguity in the pain assessment.

As future work, it is recommended the implementation of a unique landmarking scheme common to all poses. This would allow the use of the quantitative pose value to define the most suitable division for landmark detection, in opposition to the current qualitative pose class definition (“frontal”, “tilted” and “profile”). Additionally, obtaining a more realistic 3D model for the horse face, either by 3D scanning realistic miniatures or collecting data from real horses, would constitute an interesting challenge, reducing the difference in appearance between synthetic and real images. Also, it is suggested the optimization of the 3D material appearance by using fur-like textures and different light settings coherent with the background. Regarding pain estimation, more data should be collected with a known ground truth for the real state of the pain of each horse. This could be achieved, for instance, by collecting the current medical condition of each subject. This data should be gathered having into account the present pain score distribution in the dataset, to focus on the underrepresented classes. Besides, having such information about the medical condition of each horse would allow the definition of a threshold for the global pain score, training a single classifier based on the region-of-interests, to assess if the horse is in pain or not.

Appendices

Appendix A

What about donkeys?

The present dataset has around three times more horse faces than donkey faces, which makes the idea of applying a horse-based model in donkeys very appealing, especially considering the similarities between their Action Units and the common grimace scale (EQUIFAP), described for equines. To evaluate the potential of extrapolating a horse-based model to donkey images, preliminary tests were made by applying the trained models on the donkey dataset.

A.1 Results

The steep drop in results for both pose detection (Table A.1) and landmarking (Table A.2), reveals clear differences in the face proportions, caused by the distinct ear sizes and nose length. To improve the performance of the model for donkeys, it is possible to “donkify” the horse faces by defining a Thin Plate Spline-based deformation field from horse to donkey, based on the facial landmarks. Alternatively, experiments can be made with mixed training.

Table A.1: Pose estimation on the donkey face dataset, containing 534 images, using a HOG-SVM model trained with the horse face dataset. The *yaw* angle sign is presented when needed. The performance is separately presented for each pose bin as well as the weighted average of each performance metric, according to the pose distribution. For the individual performance metrics, the class of interest was labeled as “positive” and all the others as “negative”, calculating the Precision, Recall, and F1-Score based on that.

	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Profile (−)	0.61	0.25	0.35
Tilted (−)	0.49	0.43	0.46
Frontal	0.06	0.67	0.11
Tilted (+)	0.46	0.52	0.49
Profile (+)	0.55	0.23	0.32
Weighted Average	0.52	0.36	0.40

Table A.2: Landmarking results for the donkey face dataset containing 534 images, using an ERT model trained on the horse face dataset. The success rate indicates the percentage of landmarks with a location error below 10% of the eye-nostril distance. Donkey faces are smaller, so the threshold of error can be increased.

	<i>Mean Error</i>	<i>Success Rate</i>
Frontal	0.27	0.07
Tilted	0.32	0.07
Profile	0.37	0.08
Weighted Average	0.33	0.08

Bibliography

- [1] KB Gleerup and Casper Lindegaard. Recognition and quantification of pain in horses: A tutorial review. *Equine Veterinary Education*, 28(1):47–57, 2016.
- [2] FH Ashley, AE Waterman-Pearson, and HR Whay. Behavioural assessment of pain in horses and donkeys: application to clinical practice and future studies. *Equine veterinary journal*, 37(6):565–575, 2005.
- [3] Karina B Gleerup, Björn Forkman, Casper Lindegaard, and Pia H Andersen. An equine pain face. *Veterinary anaesthesia and analgesia*, 42(1):103–114, 2015.
- [4] Dale J Langford, Andrea L Bailey, Mona Lisa Chanda, Sarah E Clarke, Tanya E Drummond, Stephanie Echols, Sarah Glick, Joelle Ingrao, Tammy Klassen-Ross, Michael L LaCroix-Fralish, et al. Coding of facial expressions of pain in the laboratory mouse. *Nature methods*, 7(6):447–449, 2010.
- [5] Stephanie CJ Keating, Aurelie A Thomas, Paul A Flecknell, and Matthew C Leach. Evaluation of emla cream for preventing pain during tattooing of rabbits: changes in physiological, behavioural and facial expression responses. *PloS one*, 7(9):e44437, 2012.
- [6] Krista M McLennan, Carlos JB Rebelo, Murray J Corke, Mark A Holmes, Matthew C Leach, and Fernando Constantino-Casas. Development of a facial expression scale using footrot and mastitis as models of pain in sheep. *Applied Animal Behaviour Science*, 176:19–26, 2016.
- [7] Emanuela Dalla Costa, Michela Minero, Dirk Lebelt, Diana Stucke, Elisabetta Canali, and Matthew C Leach. Development of the horse grimace scale (hgs) as a pain assessment tool in horses undergoing routine castration. *PLoS one*, 9(3):e92281, 2014.
- [8] Johannes PAM van Loon and Machteld C Van Dierendonck. Monitoring acute equine visceral pain with the equine utrecht university scale for composite pain assessment (equus-compass) and the equine utrecht university scale for facial assessment of pain (equus-fap): a scale-construction study. *The Veterinary Journal*, 206(3):356–364, 2015.
- [9] Johannes PAM van Loon and Machteld C Van Dierendonck. Monitoring equine head-related pain with the equine utrecht university scale for facial assessment of pain (equus-fap). *The Veterinary Journal*, 220:88–90, 2017.

- [10] Marwa Mahmoud, Yiting Lu, Xijie Hou, Krista McLennan, and Peter Robinson. Estimation of pain in sheep using computer vision. In *Handbook of Pain and Palliative Care*, pages 145–157. Springer, 2018.
- [11] Hilde I Hummel, Francisca Pessanha, Albert Ali Salah, T van Loon, and Remco C Veltkamp. Automatic pain detection on horse and donkey faces. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)(FG)*, pages 717–724.
- [12] Rosenberg Ekman. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.
- [13] Ahmed Bilal Ashraf, Simon Lucey, Jeffrey F Cohn, Tsuhan Chen, Zara Ambadar, Kenneth M Prkachin, and Patricia E Solomon. The painful face–pain expression recognition using active appearance models. *Image and vision computing*, 27(12):1788–1796, 2009.
- [14] Patrick Lucey, Jeffrey F Cohn, Kenneth M Prkachin, Patricia E Solomon, and Iain Matthews. Painful data: The unbc-mcmaster shoulder pain expression archive database. In *Face and Gesture 2011*, pages 57–64. IEEE, 2011.
- [15] Patrick Lucey, Jeffrey F Cohn, Iain Matthews, Simon Lucey, Sridha Sridharan, Jessica Howlett, and Kenneth M Prkachin. Automatically detecting pain in video through facial action units. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(3):664–674, 2010.
- [16] Gwen Littlewort, Jacob Whitehill, Tingfan Wu, Ian Fasel, Mark Frank, Javier Movellan, and Marian Bartlett. The computer expression recognition toolbox (cert). In *Face and gesture 2011*, pages 298–305. IEEE, 2011.
- [17] Xiaojing Xu, Kenneth D Craig, Damaris Diaz, Matthew S Goodwin, Murat Akcakaya, Büşra Tuğçe Susam, Jeannie S Huang, and Virginia R de Sa. Automated pain detection in facial videos of children using human-assisted transfer learning. In *International Workshop on Artificial Intelligence in Health*, pages 162–180. Springer, 2018.
- [18] Albert Ali Salah, Nicu Sebe, and Theo Gevers. Communication and automatic interpretation of affect from facial expressions. In *Affective Computing and Interaction: Psychological, Cognitive and Neuroscientific Perspectives*, pages 157–183. IGI Global, 2011.
- [19] Sakha Mehdi and Vatandost Mohammad. A farm-based prospective study of equine colic incidence and associated risk factors. *Journal of Equine Veterinary Science*, 26(4):171–174, 2006.
- [20] Bram Jonkers. *Equine Utrecht University Scale for Automated Recognition in Facial Assessment of Pain – EQUUS-ARFAP (MSc Thesis)*. Universiteit Utrecht, 2018.
- [21] Susana G Sotocina, Robert E Sorge, Austin Zaloum, Alexander H Tuttle, Loren J Martin, Jeffrey S Wieskopf, Josiane CS Mapplebeck, Peng Wei, Shu Zhan, Shuren

- Zhang, et al. The rat grimace scale: a partially automated method for quantifying pain in the laboratory rat via facial expressions. *Molecular pain*, 7:1744–8069, 2011.
- [22] Alexander H Tuttle, Mark J Molinaro, Jasmine F Jethwa, Susana G Sotocinal, Juan C Prieto, Martin A Styner, Jeffrey S Mogil, and Mark J Zylka. A deep neural network to assess spontaneous pain from mouse facial expressions. *Molecular pain*, 14:1744806918763658, 2018.
- [23] Heng Yang, Renqiao Zhang, and Peter Robinson. Human and sheep facial landmarks localisation by triplet interpolated features. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8. IEEE, 2016.
- [24] Charlie Hewitt and Marwa Mahmoud. Pose-informed face alignment for extreme head pose variations in animals. In *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 1–6. IEEE, 2019.
- [25] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [26] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *European conference on computer vision*, pages 428–441. Springer, 2006.
- [27] Francisca Pessanha, Marwa Mahmoud, and Krista McLennan. Towards automatic monitoring of disease progression in sheep: A hierarchical model for sheep facial expressions analysis from video. In *15th IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE, 2020.
- [28] Hilde Hummel. *Analysing Horse and Donkey Faces for Measuring Pain Expressions (MSc Thesis)*. Universiteit Utrecht, 2020.
- [29] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- [30] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.
- [31] Xiangyu Zhang, Jianhua Zou, Kaiming He, and Jian Sun. Accelerating very deep convolutional networks for classification and detection. 2015.
- [32] Yue Wu and Qiang Ji. Facial landmark detection: A literature survey. *International Journal of Computer Vision*, 127(2):115–142, 2019.
- [33] Oya Çeliktutan, Sezer Ulukaya, and Bülent Sankur. A comparative study of face landmarking techniques. *EURASIP Journal on Image and Video Processing*, 2013(1):13, 2013.
- [34] Stephen Milborrow and Fred Nicolls. Locating facial features with an extended active shape model. In *European conference on computer vision*, pages 504–513. Springer, 2008.

- [35] Tim Cootes, ER Baldock, and J Graham. An introduction to active shape models. *Image processing and analysis*, 243657:223–248, 2000.
- [36] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence*, 23(6):681–685, 2001.
- [37] Timothy F Cootes and Christopher J Taylor. Active shape models—‘smart snakes’. In *BMVC92*, pages 266–275. Springer, 1992.
- [38] David Cristinacce and Timothy F Cootes. Feature detection and tracking with constrained local models. In *Proc. BMVC*, volume 1, page 3, 2006.
- [39] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.
- [40] David Cristinacce and Timothy F Cootes. Boosted regression active shape models. In *BMVC*, volume 2, pages 880–889. Citeseer, 2007.
- [41] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2879–2886. IEEE, 2012.
- [42] Peter N Belhumeur, David W Jacobs, David J Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2930–2940, 2013.
- [43] Yue Wu and Qiang Ji. Discriminative deep face shape model for facial point detection. *International Journal of Computer Vision*, 113(1):37–53, 2015.
- [44] Michel Valstar, Brais Martinez, Xavier Binefa, and Maja Pantic. Facial point detection using boosted regression and graph models. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2729–2736. IEEE, 2010.
- [45] Matthias Dantone, Juergen Gall, Gabriele Fanelli, and Luc Van Gool. Real-time facial feature detection using conditional regression forests. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2578–2585. IEEE, 2012.
- [46] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [47] Juergen Gall, Nima Razavi, and Luc Van Gool. An introduction to random forests for multi-class object detection. In *Outdoor and large-scale real-world scene analysis*, pages 243–263. Springer, 2012.
- [48] Heng Yang and Ioannis Patras. Privileged information-based conditional regression forest for facial feature detection. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–6. IEEE, 2013.
- [49] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. Continuous conditional neural fields for structured regression. In *European conference on computer vision*, pages 593–608. Springer, 2014.

- [50] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Learning deep representation for face alignment with auxiliary attributes. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):918–930, 2015.
- [51] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. Face alignment by explicit shape regression. *International Journal of Computer Vision*, 107(2):177–190, 2014.
- [52] Piotr Dollár, Peter Welinder, and Pietro Perona. Cascaded pose regression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1078–1085. IEEE, 2010.
- [53] Mustafa Ozuysal, Pascal Fua, and Vincent Lepetit. Fast keypoint recognition in ten lines of code. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. Ieee, 2007.
- [54] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. pages 1867–1874, 2014.
- [55] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013.
- [56] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [57] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3476–3483, 2013.
- [58] Rajeev Ranjan, Vishal M Patel, and Rama Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):121–135, 2017.
- [59] Erjin Zhou, Haoqiang Fan, Zhimin Cao, Yuning Jiang, and Qi Yin. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. pages 386–391, 2013.
- [60] George Trigeorgis, Patrick Snape, Mihalios A Nicolaou, Epameinondas Antonakos, and Stefanos Zafeiriou. Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4177–4187, 2016.
- [61] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. pages 146–155, 2016.
- [62] Amin Jourabloo and Xiaoming Liu. Large-pose face alignment via cnn-based dense 3d model fitting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4188–4196, 2016.
- [63] Adrian Bulat and Georgios Tzimiropoulos. Convolutional aggregation of local evidence for large pose face alignment. 2016.

- [64] Maheen Rashid, Xiuye Gu, and Yong Jae Lee. Interspecies knowledge transfer for facial keypoint detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6894–6903, 2017.
- [65] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [66] Michael Bronstein. *Numerical Geometry of Non-Rigid Shapes-Farthest Point Sampling*. 2009.
- [67] Thomas J. Cashman and Andrew W. Fitzgibbon. What shape are dolphins? building 3D morphable models from 2D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):232–244, 2013.
- [68] Angjoo Kanazawa, Shahar Kovalsky, Ronen Basri, and David Jacobs. Learning 3D deformation of animals from 2D images. *Computer Graphics Forum*, 35(2):365–374, 2016.
- [69] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6365–6373, 2017.
- [70] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.
- [71] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
- [72] Fred L. Bookstein. Principal Warps: Thin-Plate Splines and the Decomposition of Deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989.
- [73] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.
- [74] Nataniel Ruiz, Eunji Chong, and James M. Rehg. Fine-grained head pose estimation without keypoints. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, volume 2018-June, pages 2155–2164, 2018.
- [75] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.
- [76] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1021–1030, 2017.