EXPLAINING DNN BASED FACIAL EXPRESSION CLASSIFICATIONS

by

Kaya ter Burg

Submitted to the Artificial Intelligence Program
in partial fulfillment of the requirements for the degree of
Bachelor of Science

Undergraduate Program in Artificial Intelligence
Utrecht University
2021

EXPLAINING DNN BASED FACIAL EXPRESSION CLASSIFICATIONS

APPROVED BY:

dr. Heysem Kaya .....
(Thesis Supervisor)

dr. Aleksey Nazarov .....

DATE OF APPROVAL: DD.MM.YYYY

ACKNOWLEDGEMENTS

First and foremost, my gratitude goes to my supervisor, dr. Heysem Kaya. This thesis could not have been what it is now without his help, guidance and enormous efforts. I would like to thank Denis Dresvyanskiy and Elena Ryumina for sharing their insights into CNNs and the fine-tuning of such models. Finally, I give my thanks to my father for always listening patiently to my ramblings.

ABSTRACT

EXPLAINING DNN BASED FACIAL EXPRESSION CLASSIFICATIONS

Classifying facial expressions is a vital part of developing systems capable of aptly interacting with users. In this field, the use of deep-learning models has become the standard. However, the inner workings of these models are unintelligible, which is an important issue when deploying them to high-stakes environments. Recent efforts to generate explanations for emotion classification systems has been focused on this type of models. In this study, an alternative way of explaining the decisions of a more conventional model based on geometric features is presented. I develop a geometric-features-based deep neural network (DNN) and a convolutional neural network (CNN). After calculating the fidelity and accuracy scores of the explanations, I find that they approximate the DNN well. From the performed user study, it becomes clear that the explanations increase the understanding of the DNN and that they are preferred over the explanations for the CNN, which are more commonly used. I argue that the use conventional models is better suited for high-stakes decisions than black-box models, which is shown using the new explanation method. All scripts are available at: https://github.com/kayatb/GeomExp.

NEDERLANDSE ABSTRACT

HET VERKLAREN VAN OP DNN-GEBASEERDE CLASSIFICATIES VAN GEZICHTSUITDRUKKINGEN

Het classificeren van gezichtsuitdrukkingen is een onmisbaar deel van het ontwikkelen van systemen die op een passende wijze met gebruikers kunnen interacteren. In dit onderzoeksveld is het gebruik van deep-learning modellen de standaard geworden. Echter, hoe deze modellen tot hun keuzes komen, is ondoorgrondelijk. Dit is een groot probleem voor het gebruik van dit soort modellen in omgevingen waarbij modelbeslissingen ver strekkende gevolgen kunnen hebben. Recent onderzoek naar het genereren van verklaringen voor emotie-classificatie modellen is gefocust op dit type modellen. In dit onderzoek wordt een alternatieve manier van verklaringen genereren voor een conventioneler model gebaseerd op geometrische features gepresenteerd. Ik ontwikkel een diep neuraal netwerk (DNN) dat gebruik maakt van geometrische features en een convolutioneel neuraal netwerk (CNN). Na het berekenen van de fidelity en accuracy scores van de verklaringen, kan geconcludeerd worden dat de verklaringen het DNN goed benaderen. Uit het uitgevoerde gebruikersonderzoek blijkt dat de verklaringen het begrip van het DNN verhogen en dat deze verklaringen geprefereerd worden boven de verklaringen voor het CNN, die vaker gebruikt worden. Ik beargumenteer dat het gebruik van meer conventionele modellen beter zou zijn voor omgevingen met grote belangen, wat wordt getoond met behulp van de nieuwe verklaringen. Alle code is te zien op: https://github.com/kayatb/GeomExp.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
NEDERLANDSE ABSTRACT v
LIST OF FIGURES viii
LIST OF TABLES ix
LIST OF ACRONYMS/ABBREVIATIONS x
1. INTRODUCTION 1
2. BACKGROUND AND RELATED WORK 4
2.1. Background on Explainable AI 4
2.1.1. SHAP 6
2.1.2. Grad-CAM 6
2.2. XAI in Affective Computing and FER 7
3. PROPOSED METHOD 9
3.1. Geometric Features based DNN modeling 9
3.1.1. Geometric Feature Extraction 9
3.1.2. SHAP based Explanation Generation 10
3.2. End-to-end CNN modeling 12
3.3. User Survey 12
4. EXPERIMENTAL SETTING 14
4.1. Dataset 14
4.2. DNN-based System Development 14
4.2.1. Hyperparameter Tuning 15
4.2.2. Splitting on Pose 17
4.2.3. Feature Selection 18
4.3. CNN-based System Development 20
4.3.1. Preprocessing 20
4.3.2. Fine-tuning 20
4.4. User Survey Construction 21

4.4.1. Environment 21
4.4.2. Questions 21
4.4.3. Hypotheses 23
4.4.4. Statistical Tests 24
5. RESULTS 25
5.1. Experimental Results for Geometric Features based DNN models . . . 25
5.1.1. Comparative Results Using Pose-based Models 25
5.1.2. Feature Selection Results 25
5.1.3. Final DNN Configuration 26
5.2. Geometric Feature Explanation Results 26
5.3. Experimental Results for CNN models 27
5.4. Comparing Explanations for the DNN and CNN 28
5.5. User Study Results 29
5.6. Discussion 32
6. CONCLUSION 35
REFERENCES 38
APPENDIX A: ALL GEOMETRIC FEATURES 46
APPENDIX B: USER STUDY 48
B.1. Research description 48
B.2. Consent Form 48
B.3. General Questions 49

LIST OF FIGURES

Figure 3.1. DNN and CNN explanation generation pipelines 9

Figure 3.2. Example geometric feature depictions 11

Figure 5.1. Fidelity, accuracy, and relative cumulative SHAP weight plots for the pose-based models. 26

Figure 5.2. Examples of generated explanations for the DNN and CNN models 28

Figure A.1. Landmarks with their corresponding numbers 47

LIST OF TABLES

Table 5.1. Accuracy comparison of the pose-based and non-pose based models. 25

Table 5.2. Feature selection accuracy scores 26

Table 5.3. DNN hyperparameter configurations 27

Table 5.4. Statistical tests on question set 1 30

Table 5.5. Statistical tests on question set 2 31

Table 5.6. Final test set accuracy scores 32

Table A.1. All geometric features 46

LIST OF ACRONYMS/ABBREVIATIONS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DNN	Deep Neural Network
FER	Facial Emotion Recognition
FR	Frontal
FS	Feature Selection
FSFS	Forward Sequential Feature Selection
Grad-CAM	Gradient-weighted Class Activation Mapping
HL	Half Left
HR	Half Right
KDEF	Karolinska Directed Emotional Faces
LIME	Local Interpretable Model-Agnostic Explanations
LRP	Layer-wise Relevance Propagation
ReLU	Rectified Linear Unit
RFE	Recursive Feature Elimination
SCS	System Causability Scale
SGD	Stochastic Gradient Descent
SHAP	SHapley Additive exPlanation
SUS	System Usability Scale
XAI	eXplainable Artificial Intelligence

1. INTRODUCTION

The field of affective computing concerns itself with giving computers the ability to examine and understand human affects and form their own human-like affects [1]. These notions are essential for creating empathetic computers that can interact appropriately with users, e.g. in situations such as (mental) health care, education, caring for the elderly, etc.

One of the key elements of affective computing is emotion recognition. Emotions can be recognized using acoustic, visual and linguistic modalities [1]. In visual modality, emotions are mainly recognized from faces, under Facial Expression Recognition (FER), which is the area of focus for this research. In the context of using visual information (i.e. images and videos), FER can be categorized into two different approaches: conventional and deep-learning [2]. Traditionally, FER was done in three main steps: component detection, feature extraction, and finally the emotion classification.

More recently, there has been a surge in the usage of deep-learning models for image- and video-based tasks, including but not limited to FER. Especially the use of convolutional neural networks (CNN) has increased immensely [3] and they have dramatically outperformed conventional models [4]. When using these models, the networks can extract features on their own and the images can thus be fed into the network directly instead of extracting the features beforehand. That means such models are not limited to human-extracted features.

Due to CNNs being so powerful, they have become the dominant approach for state-of-the-art methods of affective computing and FER [2]. However, these models are also extremely opaque. CNNs belong to the class of black-box models, which means they cannot be interpreted by humans. Even if one looks at all the internal workings of such a model, one could still not comprehend what abstractions the model has learned and why it makes the decisions it makes.

This is a problem. It is important that we know what is happening inside of the models we use. Artificial Intelligence (AI) models are getting more and more involved in our daily lives. Especially in high-stakes environments (e.g. the medical domain, education, mental health care) it is important that we understand what is happening underwater, as the model's decisions can have far-reaching consequences. Moreover, if we want the models to be adopted by human users, the users need to establish trust in the model, which can be improved when they understand what is happening inside the model, see for example [5, 6].

This is where the field of explainable artificial intelligence (XAI) comes into play. The broad goal of XAI is to make models more interpretable for humans [7]. In the context of affective computing and FER, researchers have started implementing XAI methods for the models they use [8, 9] and even challenges for explainable affective computing has been organized [10, 11].

The research done on XAI in the context of FER has been focused on state-of-the-art models such as CNNs. The more conventional approaches, however, have not been subjected to explainability methods yet. Consequently, a direct comparison between the interpretability of CNNs and conventional models has not been made.

In this study, I will explore the interpretability of models based on geometric features. Furthermore, I will compare the interpretability of such a model with that of a state-of-the-art CNN. I will attempt this by constructing two models: a deep neural network (DNN) based on geometric features extended from [12] and a CNN using transfer learning on a pre-trained model developed in [13]. Both models are trained on images of facial expressions and perform an emotion classification task. I will generate explanations for both the DNN and the CNN model. I will evaluate the quality of the DNN explanations using several XAI measures and compare these explanations with the explanations for the CNN. Moreover, the explanations shall be assessed and compared via a user study.

In short, the contributions of this thesis can be summarised as follows:

(i) Developing a new method of visually and textually explaining DNN predictions based on geometric features.
(ii) Making a direct comparison between the interpretability of a CNN and a DNN trained for an emotion classification task.
(iii) Performing a user study to evaluate and compare the quality of the explanations.

This study is organized as follows: first I will discuss background and related literature in Chapter 2. Next, in Chapter 3, I shall explain how the explanations and user study are constructed. Consequently, the experiments for developing the structures for both models and the user study shall be explicated in Chapter 4. After that I shall discuss the results these experiments yielded in Chapter 5. Lastly follows Chapter 6 with the conclusion.

2. BACKGROUND AND RELATED WORK

2.1. Background on Explainable AI

The field of explainable AI wants to make models more understandable for humans. This is important when we let AI models make important decisions. In terms of a classic example: we have a model in service of a bank that decides whether or not to give a person a loan. If someone is denied the loan, naturally that someone would like to know why they did not get it and what they need to change in order for them to obtain it (counter-factual). Additionally, we have ethical concerns: e.g. does the model look at racial features? If the bank uses a very complex model, we cannot know this just by looking at the model on its own. We need more explanations to gain the insights we want.

The term that stands at the centre of XAI is *interpretability*. There is little consensus on an exact definition for this. We use the same definition as [14] in terms of machine learning systems: *the ability to explain or to present in understandable terms to a human*. The main problem with complex state-of-the-art models such as CNNs is that their interpretability level is very low. These models belong to the collection of black-box models: the internal workings are unintelligible.

As to measurements to evaluate the quality of explanations, there are no formal methods for this yet [7, 15]. We can make a distinction between measurements that make use of human participants and those that do not [14].

Measurements that do not depend of human evaluation can be calculated automatically. One such metric is *fidelity* or *faithfulness*: how well does the explanation approximate the original model [16, 17]? This is measured in terms of accuracy, but with respect to the original model's predictions instead of the ground truth. This is an important metric, since an explanation that does not approximate the model well

does not tell us anything about the original model and is virtually useless.

The fidelity metric is distinct from *plausibility*: how convincing the explanation is for humans [17]? These two should not be mixed up, as they represent different things and should be calculated in different manners. Plausibility should be measured based on human evaluation, whereas fidelity should not. Both the fidelity and plausibility metric will used in this study.

Achieving interpretability can be done via two distinct paths: explaining an existing, black-box model (post-hoc explaining) or designing models that are inherently interpretable, see for example [18, 19]. In this study, I look at post-hoc explanations that are thus generated after the models have been constructed. Note that there are people who favour designing inherently interpretable models rather than post-hoc explanations, see [20].

Furthermore, there is a distinction between model-specific and model-agnostic explaination methods [15]. Model-specific explanations are those that only work for a specific type of model, e.g. only for CNNs. The opposite of this is model-agnostic methods. Such a method works for any type of model. It treats the model essentially as a black box and does need to look at any of the internal workings.

The last distinction is between local and global explanations [15]. Global explanations attempt to explain the whole model, whereas local explanations explain a subset of the data or even a single data point. Multiple local explanations can also be used to approximate a global explanation. In this study I will solely make use of local explanations on single data instances.

Next, I will discuss the two methods that are used as a basis to generate explanations in this study.

2.1.1. SHAP

SHAP (SHapley Additive exPlanation) [21] is a widely used method for generating explanations. Its main goal is to calculate the contribution of each feature to the prediction, thus explaining what features are the most important for a prediction. This is done using Shapley values, which have their foundation in game theory [22]. In short, importance of a feature f_i is calculated using a weighted average of the difference in prediction $f(S \cup f_i) - f(S)$, where S is a subset of the original feature set and the values of the complement set are assumed missing. SHAP also comes with some desirable properties: local accuracy (fidelity), missingness, and consistency.

I chose to use the SHAP method – more specifically the model-agnostic version of SHAP: KernelSHAP – over e.g. LIME [23], since the KernelSHAP implementation extends the heuristically driven LIME, but with the desirable properties of SHAP included.

SHAP can also be used for explaining CNNs, where the input does not consist of distinct features, but an image. In that case, SHAP groups pixels together as so-called 'super-pixels' and calculates the values with these super-pixels as features. However, this approach is computationally much more expensive than gradient based methods, e.g. Grad-CAM, which is described below.

SHAP is a model-agnostic, post-hoc method for generating explanations. It can thus work on any pre-made model.

2.1.2. Grad-CAM

Another prevalent method for explaining CNNs is Grad-CAM (Gradient-weighted Class Activation Mapping) [24]. It is a method to visualize class activation maps of the CNN. With that, one can see where the model is 'looking'. Grad-CAM works on the last convolutional layer of the model and uses the gradients that go into that layer

(dependent on the target concept one wants to show an explanation for). Based on this, a heatmap is generated, which can be superimposed on the original image, thus showing what parts of the image activated the network and what the model based its decision on.

Grad-CAM is a model-specific, post-hoc method. It is specifically made for explaining existing CNNs.

2.2. XAI in Affective Computing and FER

Like in other AI research areas, in affective computing and FER there has been an increase in research into explaining models as well. This research has been focused on CNNs for the most part, as this is the type of model that is the most prevalent in contemporary research.

In [8], Weitz et al. explained a CNN model that distinguishes pain from other emotions, such as happiness or disgust. For this they used the XAI method Layer-wise Relevance Propagation (LRP) [25]. They found that while this gives some insights into the model's decisions, it is not distinctive enough.

A challenge on explainability in computer vision was proposed in 2017 by Escalante et al. [10]. The main target of this challenge is to make an explainable model that examines videos of job candidates and gives a first impression in terms of the big five personality traits. An example submission is [26], where class activations and action units are used for explaining the predictions of their CNN model.

Both [27] and [28] use Shapley values to explain their models on sentiment analysis, although this analysis is in a different context than FER. Prajod et al. used LRP saliency maps to investigate whether a network has learned concepts (in this case action units), especially in the case where a network originally trained for emotion recognition is used as a base for transfer-learning a model to recognize pain [29].

Gund and Bharadwaj et al. propose a technique for extracting influential landmarks in [30]. They do this in the context of moving faces and use a CNN for the emotion classification. Then, class activation maps are used to find influential regions and from these, landmarks are extracted that are based on action units.

3. PROPOSED METHOD

I construct two different types of models: a DNN and a CNN. Both models work on the same dataset and perform a facial expression classification task. The models are first optimized for this particular task and then I generate explanations detailing why the model made certain decisions. Finally, I evaluate these explanations by calculating the fidelity (in case of the DNN), comparing them and performing a user study. The two complete pipelines can be seen in Figure 3.1.

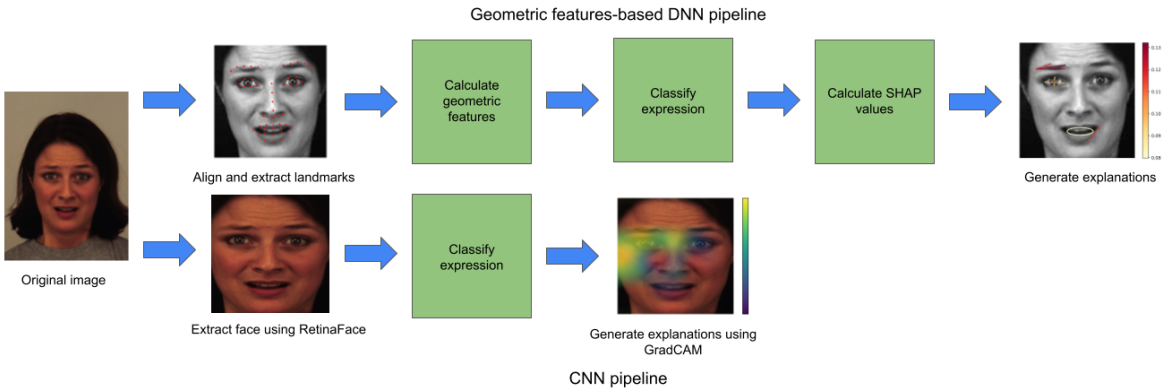

Figure 3.1. The two constructed pipelines. The top pipeline illustrates the process of the geometric features-based DNN and the bottom one illustrates the CNN process. The example image used in the pipeline is AF01AFS from the KDEF dataset.

3.1. Geometric Features based DNN modeling

3.1.1. Geometric Feature Extraction

The feature set used for the DNN is based on the geometric features from a previous study on an emotion classification task [12]. In the original study, Kaya et al. aligned and extracted landmarks from the images using Xiong and De la Torre's Supervised Descend Method [31]. Through this approach 49 fitted landmarks are obtained. From these landmarks, geometric features can be calculated. Geometric features quantify the geometric configurations that are constructed from elements such as points (as in this case), lines, etc. and can represent e.g. distances, areas, angles.

Originally, there were 23 hand-crafted geometric features. I extended this set to include the slope of the left and right eyebrow. Some features were originally averaged over the left and right parts of the face, which was reversed. The averaged features are less expressive and separate features for both parts of the face are more useful when explaining the model's decisions particularly on posed faces. Eventually, I ended up with a set of 40 features. See Appendix A for the full description of all features.

3.1.2. SHAP based Explanation Generation

After the DNN has been constructed and trained on the data, I can generate explanations for its decisions. I do so using SHAP. For each image, the SHAP value of each feature is calculated. Next, I take the n features with the highest absolute value, i.e. the most important features. Each of those geometric features corresponds to several landmarks the feature was originally calculated from and with those, the geometric features can be plotted on the face. Since we have different types of geometric features, we end up with features that are displayed as a line, an angle, an ellipse, or aspect ratio. See Figure 3.2 for examples on how geometric features of different types are visualized. The features are coloured according to their SHAP value from yellow to red, with the more red the colour, the more important the feature was for the model's decision.

Accompanying these visualizations, I generate textual explanations. In these texts, I mention what the model's prediction for the image is and whether that is correct. If the prediction is incorrect, the true label is given. Furthermore, I list the names of the features that are plotted on the face in order of their importance. The sum of the SHAP values of the displayed features is calculated as a percentage of the sum of the SHAP values of all features and reported in the textual explanation.

I evaluate this method quantitatively by calculating both its *fidelity* and *explanation accuracy*. The fidelity is calculated by constructing new data points where only the top n features keep their original values and all other features are set to the training

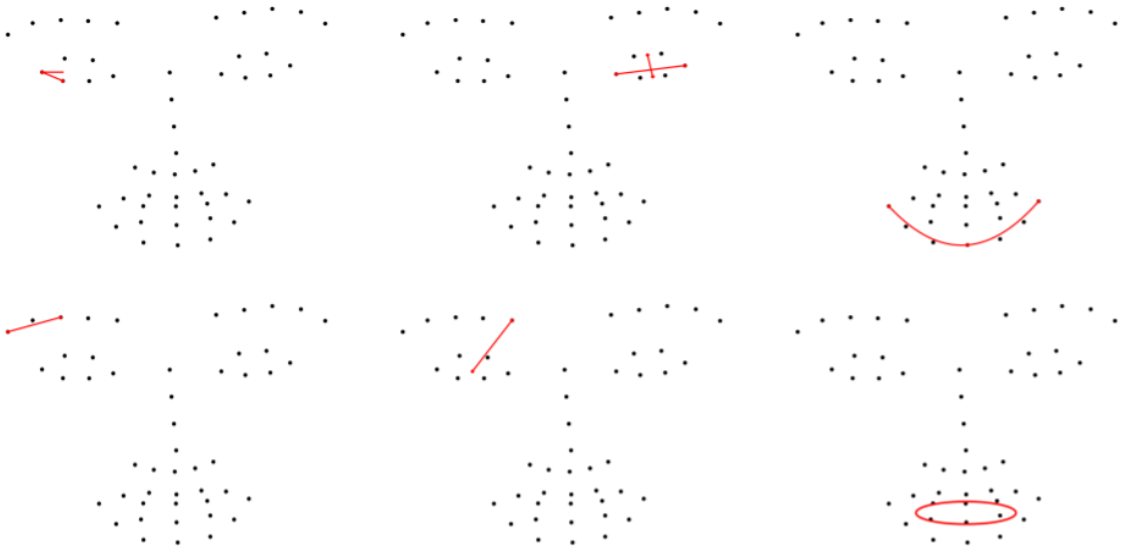

Figure 3.2. Examples of how geometric features are displayed based on the landmarks they originate from. From left to right, top to bottom, the geometric features displayed are: lower eye outer angles (L), eye aspect ratio (R), bottom lip curvature, outer mid eyebrow slope (L), eye - inner eyebrow distance (L), mouth opening / mouth width.

set average value for that feature (note that in case of feature value standardization, this value can be zero; note also that this is the way model-agnostic SHAP handles the *missing attributes*). I then let the model predict the class for this newly created data point and compare this to the model's prediction of the original data point. The percentage of predictions that stay the same as the original prediction is the fidelity score.

The *explanation accuracy* score is calculated in a similar fashion, only now the new predictions are not evaluated against the model's original prediction, but against the ground truth. Note that this is a distinct metric from the model's accuracy, which is simply the percentage of correctly classified examples. To avoid confusion, the accuracy measure for explanations shall be called the "explanation accuracy" from this point onward.

Furthermore, I calculate the relative cumulative SHAP weight for the top n features by dividing the sum of the SHAP values of those features by the sum of all features' SHAP values. A plot of the aforementioned measures helps the analysis of fidelity convergence.

3.2. End-to-end CNN modeling

For the CNN, I make use of a pre-trained model, since the used dataset is rather small and CNNs are very prone to overfitting. The model I use as a base was originally created by Dresvyanskiy et al. in [13]. They took a CNN model that was pre-trained on the VGGFace2 dataset [32] – which is mainly used for face recognition – and then further fine-tuned it on the Aff-Wild dataset [33]. Their model uses the same discrete seven emotions as I do, so no further alterations to the model's architecture were needed. I then fine-tune their model on the KDEF dataset. Thereafter, I generate explanations to gain insights into the model's decision making using Grad-CAM (implemented by [34]) and SHAP. Because of the (pixel attribution) nature of these types of explanations, one cannot calculate fidelity or a similar measure for these.

3.3. User Survey

In order to evaluate the plausibility of both model's explanations, I perform a user study where participants answer questions on their understanding of and trust in the models. The user study consists of two main parts: evaluating the geometric features based explanations and comparing the explanations for the DNN and the CNN. All questions on the explanations are posed in the form of statements together with a Likert item [35] from one to five (i.e. strongly disagree, disagree, neither agree nor disagree, agree, strongly agree). The questions can be found in Chapter 4.4.2.

In the first part, the participants will first see five example images from the test set with the original image and the probability distribution. The amount of examples where the model made a wrong prediction is chosen in accordance with the final model

test set accuracy score. After examining these images, they will answer ten questions regarding their understanding and trust in the model.

After answering these questions, a new batch of five example images will be shown. This time the participants will see five different images from the test set, but with the visual explanation images (i.e. the most important geometric features plotted on the face) and the accompanying textual explanation. Afterwards, they will answer the same ten questions as with the previous batch.

For the second part, the participants will be shown seven example images (one for each class) where the explanations for the DNN and the CNN predictions on the same image are shown side-by-side. Also shown are the probability distributions for each model prediction. Thereafter, the participants will answer ten questions where they compare both models and their respective explanations.

4. EXPERIMENTAL SETTING

Python version 3.8.2 was used for all the implementations. All models were trained using TensorFlow [36] and Keras [37].

4.1. Dataset

The dataset used for all models is the Karolinska Directed Emotional Faces (KDEF) dataset [38]. It consists of 4900 images of human faces displaying seven different emotional expressions. The expressions consist of the six basic emotions – anger, disgust, fear, happiness, sadness, surprise – as defined by Ekman [39], extended with neutral.

In total there were 70 participants (35 male, 35 female) who were each photographed twice for each of the expressions from five different angles (full left profile, half left profile, straight, half right profile, full right profile). Participants' faces were centered on a grid such that their eyes and mouths are positioned in fixed coordinates.

I omitted all pictures with the full left profile and full right profile orientation, since those poses are much more difficult to classify and the goal of this research is not to develop the most all-round facial emotion classifier.

Ultimately, I ended up with 1509 training set images (subject IDs 12 - 29), 504 validation set images (IDs 30 - 35) and 923 test set images (IDs 01 - 11). This split is used for all models. The class distribution was balanced.

4.2. DNN-based System Development

For the DNN, I use the 40 geometric features extracted from the images as input. All features values are standardized with the Scikit-learn StandardScaler [40]. Each

standardised feature value is computed by subtracting the mean of the training set and dividing this by the standard deviation of the training set. This is computed individually for each feature.

I construct a feed-forward neural network with the last layer being a dense layer with seven neurons using softmax activation. Between hidden layers, ReLU activation is used to obtain non-linearity. During training, I used the Adam optimizer [41].

4.2.1. Hyperparameter Tuning

When building a neural network, a set of hyperparameters must be tuned, such as the number of layers, the number of neurons per layer, learning rate, regularisation, etc. In context of machine learning, a hyperparameter is a setting of the model that is determined before training starts and that will affect the learning procedure. That means hyperparameters are distinct from the weights of the network, as those values will be learned during training.

The settings of these hyperparameters are very important, as they can have a huge impact on performance. A network with many layers and a large amount of neurons per layer will be able to approximate very complex functions. However, such a network is also very likely to overfit and will not generalise well to unseen data. Hence, one needs to balance the amount of layers and neurons such that on one hand, the network will not overfit on the idiosyncrasies of the training data, but on the other hand will be complex enough to approximate the function that describes the data. A similar argument can be made for other hyperparameters: they need to be tuned in order to let the model perform as optimal as possible.

One could tune these hyperparameters by hand, by changing the value of a single hyperparameter and observing the resulting validation set performance of the model. This approach comes with some major disadvantages. First of all, doing the tuning by hand requires a lot of time, especially if one wants to tune many different hyperparam-

eters and test a range of different values for each setting. Secondly, by changing only one hyperparameter at a time, you cannot take into account the interactions between parameters, such as between the amount of layers and the number of neurons per layer. That would mean you would need to tune such pairs of parameters side-by-side, which will increase the search space even further.

Then we arrive at grid search. For grid search, you need to specify the (range of) values for each hyperparameter you want to tune and grid search will then exhaustively search through each possible combination of values [42]. Grid search is subject to the *curse of dimensionality*, since the number of value combinations grows exponentially.

In [42], Bergstra and Bengio propose random search as a more efficient alternative to both grid search and manual search. Where grid search searches exhaustively through all combinations, random search selects combinations at random. Random search is more efficient than grid search, in particular in the case where only a few hyperparameters actually affect the model's performance. If random search gets the same amount of computation time as grid search, it can search through a larger search space, possibly obtaining a more optimal configuration of hyperparameters.

Still, random search is a brute-force method for tuning hyperparameters [43]. A more efficient way of doing the tuning requires a more intelligent algorithm. One such algorithm is the Hyperband algorithm [43]. Hyperband is a type of early stopping-based algorithm. This means the algorithm will run a configuration for a short amount of time and see how well it performs on the validation set. It will only select the configurations that performed the best for further testing. This makes Hyperband more efficient than other algorithms.

The optimal architecture of the network I construct is found by means of hyperparameter tuning using the Keras Tuner [37] with the included Hyperband algorithm. Each architecture is evaluated on the validation set accuracy, where accuracy is defined as the percentage of correctly classified data points.

On top of the standard Hyperband algorithm, I add an early stopping callback with patience 1 to the search, which in this case means a configuration will stop training once the validation set loss does not decrease for 1 epoch, the configuration will not be further trained. This is to increase efficiency and decrease overfitting. The following configurations for hyperparameters were tested:

- Number of hidden layers: 1 - 4.
- Number of neurons per layer: 32 - 512, in steps of 32.
- l_2 regularisation [44]: {0.1, 0.001, 0.0001}.
- Dropout [45] after each hidden layer: 0 - 0.9, in steps of 0.1.
- Learning rate: {0.1, 0.01, 0.001, 0.0001}.

4.2.2. Splitting on Pose

I explored whether it would be worthwhile to split the data on pose and train a model per subset, i.e. a separate dataset and model for half left, frontal, and half right. To this end, I trained four different models: one for each pose and one for the complete dataset. All the models' hyperparameters were optimized separately.

I then compared the validation set accuracy of the model trained on the complete dataset with the concatenated accuracy of the three other models. Concatenated accuracy is calculated by counting the number of correct predictions across all three models on their respective validation datasets and dividing this amount by the total number of instances in the complete validation dataset.

In order for the development of a complete pipeline using such a split on pose, one would also have to develop a pose classifier to automatically determine an image's face orientation.

4.2.3. Feature Selection

The complete geometric feature set consists of 40 features, but this set could contain redundant features. The goal of feature selection is to obtain a subset of features that describes the dataset, but without irrelevant or noisy features [46]. Redundant features are those that provide no extra information to the model (i.e. the feature is not needed for correct classification of the data points), but such features can cause noise and can thus introduce bias into the model. This affects how well the model generalizes and hence the performance on unseen data. On top of that, the smaller a feature set, the more efficient the computation time will be.

Evaluating all possible feature subsets from a set of n features would require evaluating 2^n subsets. This is not a viable solution as the number of features increases, so other techniques are needed for selection/elemination of features. In order to eliminate redundant features from the complete feature set, I used several feature selection algorithms to select a feature subset, trained and tuned models with that subset and compared validation set accuracy with the no-feature selection performance baseline.

The first technique I tested is forward sequential feature selection (FSFS), implemented in Scikit-learn. This is a wrapper method, which means it uses the model as a black box predictor and evaluates the performance on a certain feature subset [46]. FSFS starts with an empty set of features and at each iteration it adds the feature that yields the highest performance gain. This continues until the specified amount of features is reached.

Furthermore, I tested recursive feature elimination (RFE), proposed by Guyon et al. in [47], also implemented in Scikit-learn. RFE is an iterative process consisting of three steps: train a model, compute the ranking of the features and finally eliminate the feature with the lowest ranking. This process continues until the set of features is reduced to a certain amount. In my case, I used a logistic regression model to estimate the feature ranking, since that can be taken directly from the model's coefficients.

Another technique I tried was picking the n features with the highest global SHAP value (i.e. the most important features over the entire dataset) as the feature subset. For this, I first calculate the SHAP score for each feature by summing the absolute SHAP score for each feature for each data point across all seven classes. Then I rank the features according to this total score and take the top n features. This is a very basic feature selection method based on SHAP. It could be extended by taking inspiration from RFE. One could start with the complete feature set and calculate the SHAP scores and eliminate the feature with the lowest score. Then re-train the model with the smaller feature set and calculate the new SHAP scores. Keep iterating until enough features have been eliminated. It should be noted, however, that calculating SHAP scores is computationally inefficient and such a technique would therefore be slower than others.

The final technique consisted of hand-picking feature subsets based on domain knowledge. One can argue that features belonging to the left side of the face are more important to the half left model than for the half right model, and vice versa. Therefore, I constructed two feature subsets. Both contained the features that do not correspond to a particular side of the face (e.g. mouth width) and all features that correspond to either the left or the right side of the face. This last method is only tested on the half left model and the half right model, as the frontal model does not necessarily benefit from excluding features belonging to a particular side.

For each feature selection method, I test subsets with 5 - 35 features in steps of 5. Ultimately, for each model I pick the feature subset that yields the highest validation set accuracy.

4.3. CNN-based System Development

4.3.1. Preprocessing

In the study where the base-model was developed, the original images were detected and aligned using RetinaFace [48]. In order to give the model the most similar images as it was trained on as possible, I use the same alignment method for the KDEF dataset, using the implementation in [49]. Note that this is a different alignment method than used for the geometric features-based DNN. For the DNN, I needed not only a method that aligns the images, but also one that extracts the landmarks from the images in order to calculate the geometric features, whereas this is not needed for the CNN. On top of face extraction, the resulting images are also resized to 224 by 224 pixels.

For the CNN, we do not split the dataset on pose, as this would decrease the dataset size even further, which would make the model more prone to overfitting.

4.3.2. Fine-tuning

For the fine-tuning, I do not change anything about the original architecture of the model, as that model works with the same classes as I do. I do add a data augmentation layer to artificially increase the dataset size in order to reduce overfitting. The data augmentation consists of randomly rotating, shearing, zooming and horizontally flipping the images. I also added l_2 regularisation [44] to the layers.

I freeze the n bottom layers of the model and train only the unfrozen top layers. Again, I add an early stopping callback with various patience values. As learning rate/optimizer, I test both Adam and Stochastic Gradient Descent (SGD) with an exponential decay learning schedule.

4.4. User Survey Construction

4.4.1. Environment

All participants answered the questions independently on a computer. The user study was made using Google Forms. Before answering any questions, participants were informed of the nature of the study, what their task consisted of and what their answers could be used for. They had to give their consent to their answers being used in a research study before they could carry on answering questions. The consent form can be found in Appendix B.2.

The group of participants consisted of 12 persons. Every participant completed high school or a form of higher education. Most participants rated their level of knowledge on AI as neutral or better (on a scale of 1 - 5).

4.4.2. Questions

With the help of the user study, we want to quantify several qualities of the geometric features based explanations. On top of that, we want to compare those explanations with the ones for the CNN on a human-level. All questions are answered via a Likert item from 1 (Strongly Disagree) to 5 (Strongly Agree).

In [6], Hoffman et al. propose several checklists to evaluate the goodness, satisfaction and trust of explanations generated for AI systems. Goodness refers to how good an explanation is, determined by factors such as clarity and precision. Satisfaction is defined as: *"the degree to which users feel that they understand the AI system or process being explained to them"* [6]. Several of their proposed questions are used in the first set of questions for the user study.

The System Usability Scale (SUS) [50] is a widely used tool to measure the usability of a system. This scale can be adapted to instead of referring to a system's

usability, to refer to a system's explanatory power. Like is done in [51], where Holzinger et al. propose the System Causability Scale (SCS) which extends SUS to measure the quality of explanations in terms of causability, I extend SUS to be used in my user study. Several questions in the first question set are based on the SUS.

Ultimately, the first question set consists of the following:

1. The output representations help me understand how the model works. (adapted from [6])
2. The output representations of how the model works are satisfying. (adapted from [6])
3. The output representations are sufficiently detailed. (adapted from [6])
4. The output representations let me know how confident the model is for individual predictions.
5. The output representations let me know how trustworthy the model is. (adapted from [6])
6. I found the output representations unnecessarily complex. (adapted from [50])
7. I think I would need an expert to give me additional explanations. (adapted from [50])
8. The outputs of the model are very predictable. (adapted from [6])
9. The model can perform the task better than a novice human. (adapted from [6])
10. I am confident in the model. I believe it works well. (adapted from [6])

The second set of questions, is partly extended from the first set. Instead of referring to the explainability of a single model, these questions make a comparison between two models. Again, the questions deal with goodness, satisfaction and trust, but this time in terms of which explanation the user finds better on several aspects. Again, I took into account questions regarding the intelligibility, complexity, level of detail and trust in the explanations. In these questions, there is a consistent reference to "model 1" and "model 2". In all cases, model 1 refers to the CNN and model 2 refers to the DNN. The second question set consists of the following:

1. The explanations for model 1 are more understandable than those for model 2.
2. I trust model 1 more than model 2.
3. I would prefer the explanations of model 1 over those for model 2.
4. The explanations for model 1 are more detailed than those for model 2.
5. The explanations for model 1 are clearer on the model's accuracy than those for model 2.
6. The explanations for model 1 reflect the model's confidence on each prediction better than those of model 2.
7. Model 1's explanations are more unnecessarily complex than those of model 2.
8. The explanations for model 1 were more precise than those for model 2.
9. I would follow model 1's advice over that of model 2.
10. The outputs of model 1 were more predictable than those of model 2.

4.4.3. Hypotheses

The hypotheses for questions 1, 2, 3, 5, 6, 8, 9, and 10 from the first question set are that the examples with explanations are evaluated with higher scores than the examples without and thus indicate that showing the explanations increase understanding of and trust in the model. The hypothesis for questions 4 is that the score would be less for the second batch, since the probability distributions are shown in the first batch, but not in the second batch. This is done on purpose, so that I can check whether participants have answered the questions in a serious manner. The hypothesis for question 7 is also that the score after the second batch is lower than after the first, since this question is phrased in a reverse manner from the others.

For question set 2, the hypotheses for questions 1, 2, 3, 4, 7, 8, 9, and 10 are that the score is below the expected median score of 3, which sits right in the centre of the 5-point scale I use, indicating that explanations for model 1 (the CNN) score lower than the explanations for model 2 (the DNN). For questions 5 and 6 I performed a two-sided test, since I present the probability distributions for both models, which can be used for answering these questions.

I will thus use one-sided tests everywhere, apart from questions 5 and 6 from question set 2, since for the other questions I only want to test whether the explanations for the DNN increase understanding and are preferred over the explanations for the CNN.

4.4.4. Statistical Tests

The questions in the user study are stated in the form of Likert items. The data obtained from Likert items are generally seen as ordinal. This means the responses have an order, but the distance between values is not necessarily equal. That is why some argue that parametric tests such as t-test or ANOVA cannot be used, but rather a non-parametric test should be used to determine the statistical significance of the results [52]. However, there is discussion surrounding this, see for example [53]. For the analysis of the user study results, I have decided to use non-parametric tests.

For the first and second batch of questions (i.e. question set 1), I used the Wilcoxon signed-rank test [54], which works on two related samples; in this case the first batch of examples without any explanations and the second batch with the explanations given. Both batches use the same questions, so I will perform a test of significance on each pair of questions. The null hypothesis for such a test is that both samples are taken from the same distribution.

The third batch of questions (i.e. question set 2) stands on its own. I have thus used the Mann-Whitney U test [55] to test the significance. This test works on two unrelated samples. For the second sample I use the expected outcome/median for each question: 3. This can be compared to a one-sample t-test. The null hypothesis here is for any selected scores s_1 from sample 1 and s_2 from sample 2, it holds that $Pr(s_1 > s_2) = Pr(s_2 > s_1)$.

5. RESULTS

5.1. Experimental Results for Geometric Features based DNN models

5.1.1. Comparative Results Using Pose-based Models

In Table 5.1 we can see that the concatenated accuracy scores of the three models split on pose is higher than the accuracy score of the model trained on the whole dataset. The separate accuracy score of the models trained on a single pose are also all higher than the no split model on the validation set.

Table 5.1. Accuracy comparison of the pose-based and non-pose based models. Overall represents the accuracy score obtained from concatenating the predictions from pose-based models.

Model	Validation set accuracy
Frontal	0.780
Half Left	0.790
Half Right	0.828
Overall	**0.794**
No split	0.756

5.1.2. Feature Selection Results

From Table 5.2 we can see that the frontal model has the highest performance with a subset of 30 features, decided by RFE. The half left model has three optimal feature sets consisting of 24 or 25 features. I decided to take the feature set picked by FSFS, as the training set accuracy was closest to the validation set accuracy for that set, indicating less overfitting than with the other two sets. For the half right model, a subset of 30 features picked by FSFS yields the best results.

Table 5.2. The highest validation set accuracy per algorithm for each pose model together with the amount of selected features. Handpicked means manually excluding right and left orientated features, for left and right poses, respectively.

	Frontal		Half Left		Half Right	
Algorithm	# Feats.	Accuracy	# Feats.	Accuracy	# Feats.	Accuracy
FSFS	25	0.7857	25	**0.8070**	30	**0.8448**
RFE	30	**0.7976**	25	**0.8070**	20	0.8276
SHAP	35	0.7857	35	0.7895	35	0.8362
Handpicked	-	-	24	**0.8070**	24	0.8017
No FS	40	0.7798	40	0.7895	40	0.8276

5.1.3. Final DNN Configuration

The final configuration for the three models obtained from the previous results can be seen in Table 5.3. These are the final models we use for the generation of the visual and textual explanations and for the comparison with the CNN model.

5.2. Geometric Feature Explanation Results

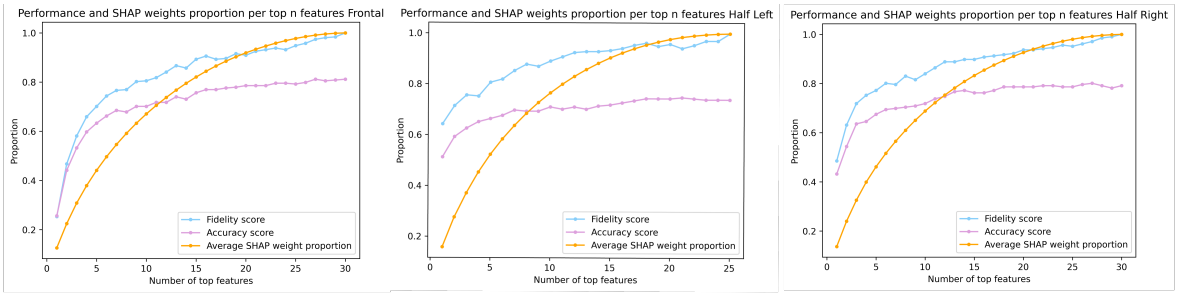

Figure 5.1. Fidelity and accuracy scores per top n SHAP features for the frontal, half left and half right model. Also given is the proportion of the SHAP weight of the top n features with regard to the total weight of all features. All scores are calculated on the test set.

In Figure 5.1 we can see that there is a steady increase in fidelity until 1.0 fidelity is reached when using all features (as should be expected). The first amount of features

Table 5.3. Final configurations of the DNN models after splitting on pose, finding the best feature subset using feature selection and hyperparameter tuning. HL: Half Left, HR: Half Right

Hyperparameter	Frontal	HL	HR
Number of hidden layers	2	1	1
Learning rate	0.001	0.001	0.001
No. neurons hidden layer 1	352	64	352
Regularisation rate hidden layer 1	0.01	0.01	0.001
Dropout hidden layer 1	0.3	0	0.6
No. neurons hidden layer 2	256	-	-
Regularisation rate hidden layer 2	0.01	-	-
Dropout hidden layer 2	0.8	-	-

where 0.8 fidelity is reached is with 9, 5, 6 features for the frontal, half left and half right model, respectively. This amount of features yields explanation accuracy scores of 0.7013, 0.6639, 0.6943 on the test set, respectively. In all three plots, the slope decreases with increasing number of features, showing a convergence trend. The explanation accuracy is equal to the final accuracy score of the model on the test set when all features are used (again, as expected).

5.3. Experimental Results for CNN models

For the CNN model, I trained the final model with the following data augmentation settings:

- rotation range: 50
- shear range: 0.5
- zoom range: 0.5
- horizontal flip

These settings give quite aggressive data augmentation, but this was necessary to combat the tendency of the model to overfit on the relatively small dataset.

Furthermore, I used the SGD optimizer with a learning rate schedule with exponential decay with an initial value of 0.01, a decay rate of 0.9 and decay step size of 10,000 (i.e. the learning rate goes down after this many steps). The regularisation value was set to 0.01 for each layer and the early stopping patience was 2. I only unfroze the top three layers for fine-tuning, the rest remained as they are in the base model. These top three layers consisted of the feed-forward classification layers.

This configuration obtained a training set accuracy of 0.9708 and a validation set accuracy of 0.7778.

5.4. Comparing Explanations for the DNN and CNN

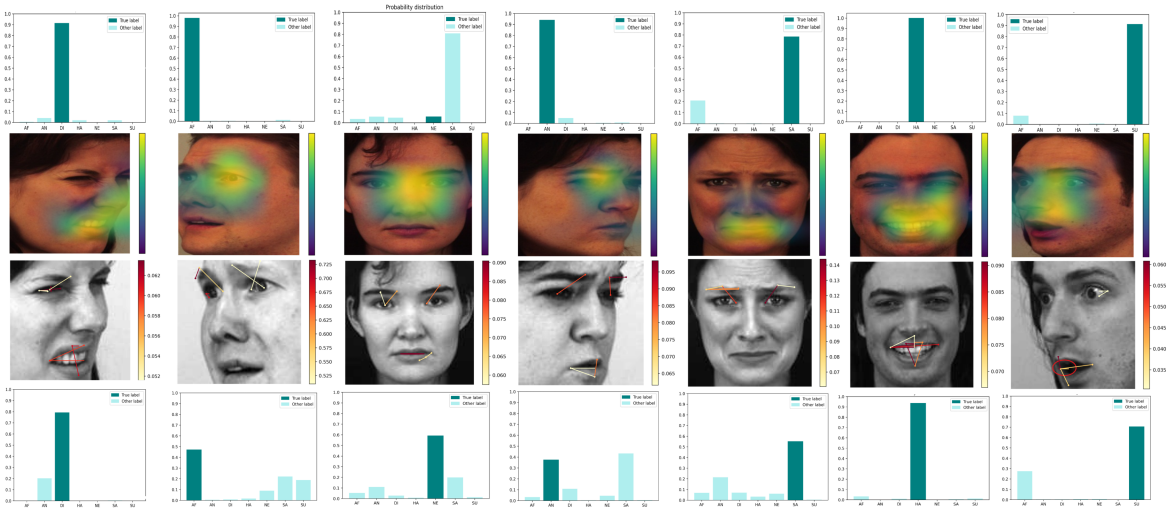

Figure 5.2. Example explanations for the CNN using Grad-CAM (second row) and DNN (third row) displaying all seven emotions. The images are the same and have the following codes: AF07DIHR, AM11AFHL, AF04NES, AF03ANHR, AF01SAS, AM01HAS, BM03SUHL, respectively. The top and bottom row give the corresponding probability distributions for the images shown.

In Figure 5.2 example explanations of the CNN using Grad-CAM (second row) and DNN (third row) can be seen. I decided to omit the explanations generated for

the CNN using SHAP. These explanations were of a lower quality than those of Grad-CAM, since there was a lot of noise in the explanations with seemingly random pixels coloured. It should also be noted that explanations for the CNN generated by SHAP take dramatically longer to compute than those generated by Grad-CAM.

To exemplify the textual explanation, the top left visual DNN explanation is accompanied by the following text:

> This person's emotion is classified as DISGUST. This classification is CORRECT. The following 5 features, listed from most important to less important, contributed for 36.6% to the decision:
> 1. Left eye aspect ratio (ratio between eye width and eye height).
> 2. Angle from bottom mouth to left upper mouth.
> 3. Angle from left mouth corner to top of the mouth.
> 4. Distance between the centre of the left eye and the left inner eyebrow.
> 5. Left lower eye outer angle.

For the geometric features explanations, I decided to visualize 5 features, which seems like the minimum given the fidelity scores of these explanations. This could easily be extended to show more features.

5.5. User Study Results

For determining the significance of the results, I take $\alpha = 0.05$ for all questions. In Table 5.4 the results from the Wilcoxon signed-rank test on the question pairs from question set 1 can be found. Scores for questions 1, 3, and 4 have a significant difference between the first and second question batch.

Question 4 was a control question and the significantly lower scores for this question shows that participants filled in the questions seriously and that posterior distributions provide significant information about prediction confidence. The significantly lower scores for question 1 shows that the participants' understanding of the model has increased after they saw the explanations. The participants also think the explanations are more sufficiently detailed, indicated by the significantly lower scores for question 3.

Question 6 does not have a significantly higher score, which is a positive outcome. This indicates the participants did not find the explanations unnecessarily more complex than no explanation. Questions 5, 9, and 10 regard the trust in the model's performance. None of these questions show a significant increase in score after the participants saw the explanations. In line with the literature [56], this shows that building trust for a technology / model is not easy and may require longitudinal exposure to or testing of the technology.

Table 5.4. Results of the Wilcoxon signed-rank test on question pairs from question set 1. Question numbers are the same as used in Chapter 4.4.2. H_a refers to whether the alternative hypothesis was that the answers from the second sample would be greater than or less than those from the first sample. Reported are the W-value and p-value obtained from the tests. * indicates $p \leq \alpha$.

Question #	H_a	W-value	p-value
1	greater	0.0	0.001*
2	greater	6.5	0.100
3	greater	4.5	0.015*
4	less	39.5	0.021*
5	greater	16.5	0.415
6	greater	0.0	0.118
7	less	23.0	0.061
8	greater	14.5	0.198
9	greater	2.0	0.718
10	greater	9.0	0.327

The results of the Mann-Whitney U test on question set 2, where the explanations for the CNN and those for the DNN are compared, can be found in Table 5.5. The scores for questions 1, 3, 4, 7 and 8 are significantly below the median. Questions 5 and 6 do not show a significant difference in scores in either direction, which is to be expected. After all, the probability distributions for both models were given in all examples. Like before, these can be used as control questions.

The scores for questions 1, 4 and 8 are significantly lower, which shows that participants' found the explanations for model 2 (the DNN) more understandable, detailed, and precise than those for model 1 (the CNN). Moreover, the participants would also prefer the explanations for model 2 over those for model 1, as indicated by the significantly lower score for question 3. There is a significantly lower score for question 7 as well. That indicates that the participants found the explanations for model 2 to be more complex than those for model 1. Even so, it seems they would still prefer the more complex explanations.

For questions 2 and 9, we can see the same happening as in the previous tests: the questions regarding trust in the models do not show a significantly lower score. Also, like in the previous tests, there is no significantly lower score regarding the predictability of the model outputs (question 8 for the first set and 10 for the second one).

Table 5.5. Results of the Mann-Whitney U test on questions from question set 2. Question numbers are the same as used in Chapter 4.4.2. H_a refers to whether the alternative hypothesis was that model 1 was evaluated as worse than model 2 (less) or a two-sided test (unequal). * indicates $p \leq \alpha$.

Question #	H_a	U-value	p-value
1	less	42.0	0.031*
2	less	66.0	0.364
3	less	42.0	0.031*
4	less	18.0	0.0003*
5	unequal	48.0	0.105
6	unequal	54.0	0.154
7	less	36.0	0.011*
8	less	24.0	0.001*
9	less	66.0	0.364
10	less	9.0	0.327

5.6. Discussion

The final test set accuracy scores of the models and a few recent models on the KDEF set can be seen in Table 5.6. The state-of-the-art models were chosen by looking at recent research (anything published since 2018) where the KDEF dataset was used. It should be noted these do not necessarily use the same train-validation-test set split as we have used in this study.

Table 5.6. The test set accuracy scores for the final DNN models (and their concatenated score) and the CNN. Added are three example scores for recent models for the KDEF dataset as well.

Model	Test Set Accuracy
GEO-DNN Frontal (FR)	0.8117
GEO-DNN Half Left (HL)	0.7395
GEO-DNN Half Right (HR)	0.7913
GEO-DNN Combined (FR, HR, HL)	0.7832
CNN (FR, HR, HL)	0.7595
Puthanidam and Moh [57] (FR, HR, HL)	0.8086
Mahmud et al. [58] (FR)	0.8602
Kandeel et al. [59] (FR)	**0.8888**

In [57], a combination of image pre-processing and different CNN models was used. The preprocessing consisted of converting the images to greyscale, data augmentation, cropping the images using Haar feature-based cascade classifiers [60], and downsampling the images to reduce memory usage. The CNN that obtained the best results on the KDEF dataset was the model fine-tuned over the model initially trained by Yu and Zhang [61]. To obtain the result as reported in Table 5.6, they use only the frontal and half rotated images, as we have done.

The result in [58] was obtained by detecting the faces using the method from [60] as well, then segmenting the image intro four parts (i.e. right eye, left eye, nose,

mouth). Next, they extracted features from those segments using Gabor filters and these features are fed into a K-nearest neighbours model for classification. They used only the frontal oriented images from the KDEF set.

In [59], a CNN model was trained on only the frontal orientated images of the KDEF set. The faces were extracted from the images using the technique from [60], like the other examples. They trained CNN models with different architectures and used Grad-CAM and saliency maps to compare the different models on explainability.

My models do not perform as well as the state-of-the-art models, but this was not the main target of the study, where a sufficiently high accuracy is aimed. The geometric features based DNN performs somewhat better than the CNN model (based on the combined score of the three DNN models). This could partly be blamed on the size of the dataset. The CNN is more likely to outperform the DNN if I had used a more substantial amount of data.

For the geometric feature explanations, I get good results on the fidelity scores, with only approximately 25 - 30% of the complete feature set necessary to give a fidelity score of above 0.8 (i.e. 80% of the predictions stay the same when using only these feature values). The explanations approximate the model to a high extent.

For the CNN, explanations using Grad-CAM can be generated, which are based on the gradients inside the model when predicting an image. However, I cannot calculate measures like fidelity and accuracy for these explanations, caused by the nature of the calculations with which these explanations are constructed. Hereby, I do not have a direct measure to know how good the explanations actually approximate the CNN model. Even though we generate explanations of the model that seem to show where it looks, it is not certain if this is actually what the model bases its decisions on.

This issue is not present in our explanations for the geometric features based DNN. I have calculated fidelity and accuracy scores for these explanations, which show

the explanations stay true to the model to a high degree. Thus, the explanations indeed say something about the decision-making process of the model.

Furthermore, the explanations as generated for the DNN seem more precise than those for the CNN. By the nature of geometric features, they are inherently explainable. Therefore, visualizing them and giving their names should be enough to know what the model sees. For the CNN, this is not the case. Like in the left most example images from Figure 5.2, we can see the model roughly looks at the mouth, but what about the mouth the model sees, is still hard to grasp. With the geometric features, we can exactly pinpoint that the model is looking at e.g. mouth width or lip curvature.

Finally, the results obtained from the user study indicate that the explanations for the (geometric features based) DNN increase participants' understanding of the underlying model. They also found those explanations better than the more frequently used heatmap explanations for CNNs on points such as understandibility, preciseness and level of detail. Even though the DNN explanations were found to be more complex, the participants would still prefer those explanations. However, the trust in the models was not increased by the DNN explanations nor was there a significant difference between the trust in the DNN and the CNN.

6. CONCLUSION

In this work, I have developed an alternative way of explaining a model that classifies facial expressions. In particular, this method displays the most important geometric features (as calculated by SHAP) plotted on the original images. I developed both geometric features based DNN models and a CNN model. On the small KDEF dataset I used here, the DNN models outperforms the CNN model, although examples can be found in the literature where models have been developed that outperform all of my models.

A case can be made that the explanations of the geometric features are more trustworthy, as measures like fidelity can be calculated, which is not the case for the CNN explanations. The explanations for the DNN show what the model bases its decisions on more precisely than those for the CNN. Furthermore, human participants found the explanations for the geometric features-based DNN better overall than the Grad-CAM explanations for the CNN. This shows that the DNN explanations are not only preferred in terms of measurements like fidelity, but also based on human evaluation.

I argue that the more conventional methods for FER are better explainable than the state-of-the-art CNNs, as can be seen in the explanations I developed and compared with methods for visualizing the decision process for CNNs. This might not matter for mundane tasks, but for high-stakes decision processes, it is vital that we can understand what the model is doing. I believe it would be better to focus more on developing and using interpretable models or models that are better to explain in critical areas than the prevalent black-box models.

The geometric feature explanations in this study are based on a DNN, which admittedly is a black-box model as well. However, the explanations I have developed here are not limited to such a model and can be generalized to other types; the ex-

planations are model-agnostic given the geometric feature set. In other words, any model that can make use of geometric features could use the explanations. An intrinsically interpretable logistic regression model could also visualize its decisions in the same way. More recently developed interpretable models that can perform more on-par with deep-learning models such as [19] can also use this technique to visualize their decisions.

This study can be extended by using a more extensive dataset with less posed images than the KDEF dataset and seeing how the geometric features perform in such an environment. Furthermore, a dataset with more emotions than these seven could be tested.

The user study I performed is also limited in certain areas. Here I used a static questionnaire to evaluate the plausibility of the methods, but this could be extended in several ways. The questionnaire could be made more interactive, such that users can explore the explanations more on their own. A more concrete way of doing this is by letting the users increase and decrease the amount of visualized features in the geometric explanations on demand. Another approach would be to let users upload their own images and explore the explanations for the model classifications on this image. This would require a fully developed pipeline, where the face and landmarks extraction from the input image, the pose classification, geometric features calculation, model classification and finally explanation generation are included.

Another suggestion would be to make the questionnaire less individual and more interactive. In [62], Chromik et al. argue that participants' mental models of how systems work cannot easily be approximated by a static questionnaire. They found that a more interactive session, where participants are asked to explain their reasoning exposed where their understanding was lacking. Some participants may overestimate their understanding in the models.

From the user study it also became clear that trust in the models was not increased

by the explanations and the DNN was not trusted over the CNN. This could be caused by a limitation caused by the nature of the questions I posed regarding trust. Trust is a hard to obtain quality and may only emerge after someone has more experience with using a system [6, 56]. The user study here was most likely too limited to truly measure trust. Additionally, the predictability of the model outputs also did not show any significant difference. To measure this quality, one would probably have to give the participants a different type of task. An example could be what Doshi-Velez and Kim call a *Forward simulation/prediction task* in [14]: show participants an example image or explanation and ask what they think the model would have predicted for this instance, regardless of what the ground truth is.

In summary, the user study especially can be extended in quite a few ways and was limited in this work. Further research should be put in the human evaluation of the proposed method.

In conclusion, I have explored and laid the groundwork for a different way of explaining a system for FER. The first results regarding the quality and plausibility of these explanations look promising. Questions on several points still exists and are open for further examination.

REFERENCES

1. Tao, J. and T. Tan, "Affective computing: A review", *International Conference on Affective computing and intelligent interaction*, pp. 981–995, Springer, 2005.

2. Ko, B. C., "A brief review of facial emotion recognition based on visual information", *Sensors*, Vol. 18, No. 2, p. 401, 2018.

3. Wang, W., Y. Yang, X. Wang, W. Wang and J. Li, "Development of convolutional neural network and its application in image classification: a survey", *Optical Engineering*, Vol. 58, No. 4, p. 040901, 2019.

4. Krizhevsky, A., I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", *Advances in neural information processing systems*, Vol. 25, pp. 1097–1105, 2012.

5. Weitz, K., D. Schiller, R. Schlagowski, T. Huber and E. André, "'Do you trust me?' Increasing user-trust by integrating virtual agents in explainable AI interaction design", *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, pp. 7–9, 2019.

6. Hoffman, R. R., S. T. Mueller, G. Klein and J. Litman, "Metrics for explainable AI: Challenges and prospects", *arXiv preprint arXiv:1812.04608*, 2018.

7. Adadi, A. and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)", *IEEE Access*, Vol. 6, pp. 52138–52160, 2018.

8. Weitz, K., T. Hassan, U. Schmid and J. Garbas, "Towards explaining deep learning networks to distinguish facial expressions of pain and emotions", *Forum Bildverarbeitung*, pp. 197–208, 2018.

9. Gund, M., A. R. Bharadwaj and I. Nwogu, "Interpretable Emotion Classification

Using Temporal Convolutional Models", *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 6367–6374, IEEE, 2021.

10. Escalante, H. J., I. Guyon, S. Escalera, J. Jacques, M. Madadi, X. Baró, S. Ayache, E. Viegas, Y. Güçlütürk, U. Güçlü *et al.*, "Design of an explainable machine learning challenge for video interviews", *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 3688–3695, IEEE, 2017.

11. Escalante, H. J., H. Kaya, A. A. Salah, S. Escalera, Y. Güç, U. Güçlü, X. Baró, I. Guyon, J. C. Jacques, M. Madadi *et al.*, "Modeling, Recognizing, and Explaining Apparent Personality from Videos", *IEEE Transactions on Affective Computing*, 2020.

12. Kaya, H., F. Gürpinar, S. Afshar and A. A. Salah, "Contrasting and combining least squares based learners for emotion recognition in the wild", *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pp. 459–466, 2015.

13. Dresvyanskiy, D., E. Ryumina, H. Kaya, M. Markitantov, A. Karpov and W. Minker, "An Audio-Video Deep and Transfer Learning Framework for Multimodal Emotion Recognition in the wild", *arXiv preprint arXiv:2010.03692*, 2020.

14. Doshi-Velez, F. and B. Kim, "Towards a rigorous science of interpretable machine learning", *arXiv preprint arXiv:1702.08608*, 2017.

15. Arrieta, A. B., N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI", *Information Fusion*, Vol. 58, pp. 82–115, 2020.

16. Guidotti, R., A. Monreale, S. Ruggieri, F. Turini, F. Giannotti and D. Pedreschi, "A survey of methods for explaining black box models", *ACM computing surveys*

(CSUR), Vol. 51, No. 5, pp. 1–42, 2018.

17. Jacovi, A. and Y. Goldberg, "Towards Faithfully Interpretable NLP Systems: How should we define and evaluate faithfulness?", *arXiv preprint arXiv:2004.03685*, 2020.

18. Letham, B., C. Rudin, T. H. McCormick, D. Madigan *et al.*, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model", *Annals of Applied Statistics*, Vol. 9, No. 3, pp. 1350–1371, 2015.

19. Nori, H., S. Jenkins, P. Koch and R. Caruana, "Interpretml: A unified framework for machine learning interpretability", *arXiv preprint arXiv:1909.09223*, 2019.

20. Rudin, C., "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead", *Nature Machine Intelligence*, Vol. 1, No. 5, pp. 206–215, 2019.

21. Lundberg, S. and S.-I. Lee, "A unified approach to interpreting model predictions", *arXiv preprint arXiv:1705.07874*, 2017.

22. Shapley, L. S., "A value for n-person games", *Classics in game theory*, Vol. 69, 1997.

23. Ribeiro, M. T., S. Singh and C. Guestrin, "" Why should i trust you?" Explaining the predictions of any classifier", *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

24. Selvaraju, R. R., M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization", *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

25. Bach, S., A. Binder, G. Montavon, F. Klauschen, K.-R. Müller and W. Samek,

"On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation", *PLOS ONE*, Vol. 10, No. 7, pp. 1–46, 07 2015, https://doi.org/10.1371/journal.pone.0130140.

26. Ventura, C., D. Masip and A. Lapedriza, "Interpreting CNN models for apparent personality trait regression", *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 55–63, 2017.

27. Bobek, S., M. M. Tragarz, M. Szelażek and G. J. Nalepa, "Explaining Machine Learning Models of Emotion Using the BIRAFFE Dataset", L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz and J. M. Zurada (Editors), *Artificial Intelligence and Soft Computing*, pp. 290–300, Springer International Publishing, Cham, 2020.

28. Liew, W. S., C. K. Loo and S. Wermter, "Emotion Recognition Using Explainable Genetically Optimized Fuzzy ART Ensembles", *IEEE Access*, 2021.

29. Prajod, P., D. Schiller, T. Huber and E. André, "Do Deep Neural Networks Forget Facial Action Units?–Exploring the Effects of Transfer Learning in Health Related Facial Expression Recognition", *arXiv preprint arXiv:2104.07389*, 2021.

30. Gund, M., A. R. Bharadwaj and I. Nwogu, "Interpretable Emotion Classification Using Temporal Convolutional Models", *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 6367–6374, IEEE, 2021.

31. Xiong, X. and F. De la Torre, "Supervised Descent Method and Its Application to Face Alignment", *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 532–539, 2013.

32. Cao, Q., L. Shen, W. Xie, O. M. Parkhi and A. Zisserman, "VGGFace2: A Dataset for Recognising Faces across Pose and Age", *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pp. 67–74, 2018.

33. Kollias, D., P. Tzirakis, M. A. Nicolaou, A. Papaioannou, G. Zhao, B. Schuller, I. Kotsia and S. Zafeiriou, "Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond", *International Journal of Computer Vision*, Vol. 127, No. 6, pp. 907–929, 2019.

34. Korobov, M. and K. Lopuhin, "ELI5", `https://eli5.readthedocs.io/en/latest/`, 2016.

35. Likert, R., "A technique for the measurement of attitudes.", *Archives of psychology*, 1932.

36. Abadi, M., A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems", , 2015, `http://tensorflow.org/`, software available from tensorflow.org.

37. Chollet, F. *et al.*, "Keras", `https://keras.io`, 2015.

38. Lundqvist, D., A. Flykt and A. Öhman, "The Karolinska Directed Emotional Faces - KDEF", , 1998.

39. Ekman, P., "Basic emotions", *Handbook of cognition and emotion*, Vol. 98, No. 45-60, p. 16, 1999.

40. Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python", *the Journal of machine Learning research*, Vol. 12, pp. 2825–2830, 2011.

41. Kingma, D. P. and J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

42. Bergstra, J. and Y. Bengio, "Random search for hyper-parameter optimization.", *Journal of machine learning research*, Vol. 13, No. 2, 2012.

43. Li, L., K. Jamieson, G. DeSalvo, A. Rostamizadeh and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization", *The Journal of Machine Learning Research*, Vol. 18, No. 1, pp. 6765–6816, 2017.

44. Ng, A. Y., "Feature selection, L 1 vs. L 2 regularization, and rotational invariance", *Proceedings of the twenty-first international conference on Machine learning*, p. 78, 2004.

45. Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", *Journal of Machine Learning Research*, Vol. 15, No. 56, pp. 1929–1958, 2014, `http://jmlr.org/papers/v15/srivastava14a.html`.

46. Chandrashekar, G. and F. Sahin, "A survey on feature selection methods", *Computers & Electrical Engineering*, Vol. 40, No. 1, pp. 16–28, 2014.

47. Guyon, I., J. Weston, S. Barnhill and V. Vapnik, "Gene selection for cancer classification using support vector machines", *Machine learning*, Vol. 46, No. 1, pp. 389–422, 2002.

48. Deng, J., J. Guo, Y. Zhou, J. Yu, I. Kotsia and S. Zafeiriou, "Retinaface: Single-stage dense face localisation in the wild", *arXiv preprint arXiv:1905.00641*, 2019.

49. Zheng, E., "Batch Face", `https://github.com/elliottzheng/batch-face`, 2020.

50. Brooke, J., "SUS: A quick and dirty usability scale", *Usability Eval. Ind.*, Vol. 189, 11 1995.

51. Holzinger, A., A. Carrington and H. Müller, "Measuring the quality of explanations: the system causability scale (SCS)", *KI-Künstliche Intelligenz*, pp. 1–6, 2020.

52. Jamieson, S., "Likert scales: How to (ab) use them?", *Medical education*, Vol. 38, No. 12, pp. 1217–1218, 2004.

53. Norman, G., "Likert scales, levels of measurement and the "laws" of statistics", *Advances in health sciences education*, Vol. 15, No. 5, pp. 625–632, 2010.

54. Wilcoxon, F., "Individual Comparisons by Ranking Methods", *Biometrics Bulletin*, Vol. 1, No. 6, pp. 80–83, 1945, `http://www.jstor.org/stable/3001968`.

55. Mann, H. B. and D. R. Whitney, "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other", *The Annals of Mathematical Statistics*, Vol. 18, No. 1, pp. 50 – 60, 1947, `https://doi.org/10.1214/aoms/1177730491`.

56. Davis, B., M. Glenski, W. Sealy and D. Arendt, "Measure Utility, Gain Trust: Practical Advice for XAI Researchers", *2020 IEEE Workshop on TRust and EXpertise in Visual Analytics (TREX)*, pp. 1–8, IEEE, 2020.

57. Puthanidam, R. V. and T.-S. Moh, "A Hybrid approach for facial expression recognition", *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication*, pp. 1–8, 2018.

58. Mahmud, F., B. Islam, A. Hossain and P. B. Goala, "Facial region segmentation based emotion recognition using K-nearest neighbors", *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, pp. 1–5, IEEE, 2018.

59. Kandeel, A. A., H. M. Abbas and H. S. Hassanein, "Explainable Model Selection of a Convolutional Neural Network for Driver's Facial Emotion Identification", A. Del Bimbo, R. Cucchiara, S. Sclaroff, G. M. Farinella, T. Mei, M. Bertini, H. J. Escalante and R. Vezzani (Editors), *Pattern Recognition. ICPR International Workshops and Challenges*, pp. 699–713, Springer International Publishing, Cham,

2021.

60. Viola, P. and M. Jones, "Rapid object detection using a boosted cascade of simple features", *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, Vol. 1, pp. I–I, IEEE, 2001.

61. Yu, Z. and C. Zhang, "Image based static facial expression recognition with multiple deep network learning", *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pp. 435–442, 2015.

62. Chromik, M., M. Eiband, F. Buchner, A. Krüger and A. Butz, "I Think I Get Your Point, AI! The Illusion of Explanatory Depth in Explainable AI", *26th International Conference on Intelligent User Interfaces*, IUI '21, p. 307–317, Association for Computing Machinery, New York, NY, USA, 2021, https://doi.org/10.1145/3397481.3450644.

APPENDIX A: ALL GEOMETRIC FEATURES

Table A.1. Table containing the full geometric feature set, extended from [12]. * indicates the feature was added here and not used in the original paper. Distance based features are normalized by face height. Landmark numbers can also be found in Figure A.1.

Feature #	Description	Landmarks	Feature type
1	Eye aspect ratio (L)	[19, 24]	Distance
2	Eye aspect ratio (R)	[25, 30]	Distance
3	Mouth aspect ratio	[31, 34, 37, 40]	Distance
4	Upper lip angle (L)	[31, 34]	Angle
5	Upper lip angle (R)	[34, 37]	Angle
6	Nose tip - mouth corner angle (L)	[16, 31]	Angle
7	Nose tip - mouth corner angle (R)	[16, 37]	Angle
8	Lower lip angle (L)	[31, 41]	Angle
9	Lower lip angle (R)	[37, 39]	Angle
10	Eyebrow slope (L)	[0, 4]	Angle
11	Eyebrow slope (R)	[5, 9]	Angle
12	Lower eye outer angles (L)	[19, 24]	Angle
13	Lower eye inner angles (L)	[22, 23]	Angle
14	Lower eye outer angles (R)	[28, 29]	Angle
15	Lower eye inner angles (R)	[25, 30]	Angle
16	Mouthe corner - mouth bottom angle (L)	31, 40]	Angle
17	Mouth corner - mouth bottom angle (R)	[37, 40]	Angle
18	Upper mouth angles (L)	[33, 40]	Angle
19	Upper mouth angles (R)	[35, 40]	Angle
20	Curvature of lower-outer lips (L)	[31, 41, 42]	Curvature
21	Curvature of lower-outer lips (R)	[37 - 39]	Curvature
22	Curvature of lower-inner lips (L)	[31, 40, 41]	Curvature
23	Curvature of lower-inner lips (R)	[37, 39, 40]	Curvature
24	Bottom lip curvature	[31, 37, 40]	Curvature

Table A.1 (cont.)

Feature #	Description	Landmarks	Feature type
25	Mouth opening / mouth width	[43 - 48]	Distance
26	Mouth up/down	[34, 40, 44]	Distance
27	Eye - middle eyebrow distance (L)	[0, 4, 19, 22]	Distance
28	Eye - middle eyebrow distance (R)	[5, 9, 25, 28]	Distance
29	Eye - inner eyebrow distance (L)	[4, 19, 22]	Distance
30	Eye - inner eyebrow distance (R)	[5, 25, 28]	Distance
31	Inner eye - eyebrow centre (L)	[2, 22]	Distance
32	Inner eye - eyebrow centre (R)	[7, 25]	Distance
33	Inner eye - mouth top distance (L)	[22, 34]	Distance
34	Inner eye - mouth top distance (R)	[25, 34]	Distance
35	Mouth width	[31, 37]	Distance
36	Mouth height	[34, 40]	Distance
37	Upper mouth height	[34, 44, 47]	Distance
38	Lower mouth height	[40, 44, 47]	Distance
39	Outer mid eyebrow slope (L)*	[0, 2]	Slope
40	Outer mid eyebrow slope (R)*	[7, 9]	Slope

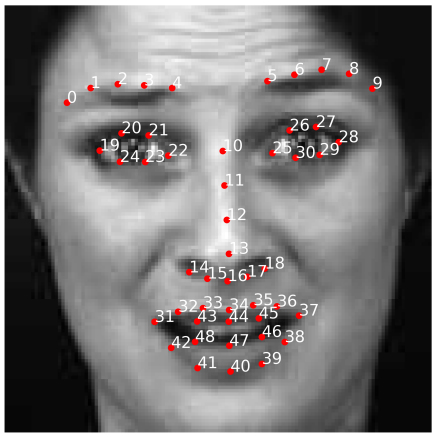

Figure A.1. All landmarks with their corresponding numbers annotated. The numbers correspond to the numbers in Table A. Example image is AF01AFS from the KDEF dataset.

APPENDIX B: USER STUDY

Below the texts and general questions in the user study are shown.

B.1. Research description

This research is done in the context of my Bachelor thesis Artificial Intelligence at Utrecht University. In this research, I want to analyze and evaluate new ways of explaining uninterpretable machine learning models. The purpose of this survey is to quantify the quality of automatic explanations (e.g. in terms of clarity and plausibility) generated for two types of Deep Neural Network models trained to predict facial expressions.

Your task is to evaluate and compare different explanation methods for two machine learning models. This will be done using closed questions. No personal data is required or being collected. The survey takes 6-8 minutes to complete.

B.2. Consent Form

The participant states:

- I voluntarily agree to participate in the research project.
- I agree that I will not be paid for my participation.
- I have been informed of the nature of the research project.
- I understand that statistical data gathered from this survey can be used in a scientific publication.
- I understand that my participation will remain anonymous.
- I agree that my data can be shared with other researchers to answer possible other research questions.

B.3. General Questions

What is the highest degree or level of school you have completed?

- No degree
- Elementary school
- High school
- MBO
- HBO
- Bachelor's degree
- Master's degree
- Doctorate degree

I am very knowledgeable on the subject of Artificial Intelligence (AI).

- 1 - Strongly disagree
- 2 - Disagree
- 3 - Neutral
- 4 - Agree
- 5 - Strongly agree