Radboudumc

Radboud Universiteit

Master Artificial Intelligence

Face classification of patients with intellectual disability

Master Thesis

August, 2020

Written by
Fien Ockers
s4618262
Radboud University

Internal supervision
prof. dr. M.A.J. van Gerven
Radboud University

External supervision
dr. Bert de Vries
drs. Lex Dingemans
Radboudumc

Abstract

Diagnosing a genetic syndrome can be difficult as extracted genetic data is not always decisive. To reach a diagnosis, doctors can compare the facial characteristics of a patient with previously diagnosed patients. This process is subjective and therefore doctors would benefit from an objective model to compare these faces. This research aims to compare the performance of several models for the task of syndrome vs. control face classification. Five different models have been used for syndrome vs. control classification, including one ensemble model. Three models use either a face representation based on a neural network, a 3D landmark representation or a morphometric representation based on these 3D landmarks. The last model is a Hybrid model which is based on previous research done at the Radboudumc and this model also has the best performance for most of the 12 syndromes included in this project. Future research can focus on reducing the computation time of the Hybrid model and making its predictions more explainable.

1 Introduction

Approximately 2% of the general population is affected by Intellectual Disability (ID), a neurodevelopmental disorder [1, 2]. Intellectual Disability is defined by the incomplete development of the mind during the developing stages of childhood, leading to impairment of several skills that contribute to the general level of intelligence, e.g. cognition, language, motor or social abilities [1].

1.1 Diagnosis

In about 30-40% of the cases, ID is caused by a genetic syndrome [3]. A genetic syndrome is the result of a mutated gene and as there are numerous different genes, there is a wide variation of frequent and rare syndromes. A genetic syndrome often comes with specific facial dysmorphisms. Diagnosing a genetic syndrome as early as possible is important to prevent potential health problems and inform parents whether the disorder is inheritable [4, 5].

Due to recent developments of next-generation sequencing techniques, it is possible to extract a lot of information from the DNA of a person [3]. However, for each gene, there are numerous mutated variants and not all of them are known to be either benign or pathogenic. In other words, whether a gene mutation is considered to be the cause of a specific syndrome or not. Thus, there are a lot of cases where the mutated gene is a variant of unknown significance (VUS). In these cases, the doctors already have a suspicion for a specific syndrome, but as the genetic data cannot support that suspicion yet, they take the facial characteristics of the patient into account. The doctors look at the facial characteristics of the to be diagnosed patient and compare them with patients that were previously diagnosed with that specific syndrome. In this way, doctors can either confirm or reject the diagnosis of their current presumed syndrome.

1.2 Computer vision model

Although most genetic syndromes come with these facial dysmorphisms, it can be difficult for clinical experts to diagnose a patient correctly based on these facial characteristics alone. Besides the fact that there are a lot of rare genetic diseases, these facial characteristics can also differ between patients with regard to their gender, age and ethnicity. Thus, reaching a correct diagnosis is subjective to the experience of the doctor with this specific syndrome and the consistency of the syndrome's facial characteristics across different patients [6].

To support doctors in deciding on a specific diagnosis based on facial dysmorphisms, it would be beneficial if there was an objective model. This model would then automate the process of extracting craniofacial features from an image, comparing them with the existing known cases of this specific syndrome and coming up with a result in the form of a probability or decision. To be

able to do this, a computer vision model is needed that can perform this task of face classification. This project aims to create this model in collaboration with the Radboudumc.

1.3 Face classification

The process of face classification usually consists of three steps. First, the face is detected, then the features are extracted from the face, and lastly, these features are used for classification. There are multiple well-performing methods to solve the task of face detection [7, 8], and there are also multiple different simple and complex classifiers that can be used to classify the face representations.

The task of classifying a face based on whether a specific syndrome is present or not, could be solved by using some sort of machine learning model, for example, a convolutional neural network. However, there is not enough data available to train a model from scratch on this task. Since there are no public data sets of syndromic patients available and the amount of data present at the Radboudumc is limited. Thus, some form of transfer learning, i.e. training a feature extraction model on a different domain and fine-tuning it on the actual domain, is necessary.

Hence, it is important to know what tasks related to face classification have already been tackled by previous researchers and could be used in this project. To be able to perform face classification, there are at least two components needed. Firstly, some sort of face representation that is based on a 2D image, as that is the format of the present data, and secondly, a way to classify these face representations based on their labels.

1.3.1 Face representation

Recently, there have been many developments in the field of computer vision with regards to the tasks of face recognition and verification [9, 10]. Face recognition is the task of identifying the person in an image based on a list of identities, and face verification is the task of classifying whether the same person is present in two different images. Research has resulted in multiple well-performing models that succeed in recognising and verifying faces with high scores on the data set Labelled Faces in The Wild [11], which is the benchmark data set for both of these tasks. These face recognition and verification models are usually trained with millions of images of thousands of different people using a private data set [9, 10].

The tasks of face recognition and verification are useful because it requires a model to come up with an internal face representation. If this model is, for example, a neural network, then this face representation can be easily extracted by removing the final classification layer. When these models are applied to a different domain with transfer learning, the last classification layer is removed and the rich face representation is used in combination with a classification model.

1.3.2 Classification models

As said above, most models that perform well on the Labelled Faces in the Wild data set [11] are neural networks and thus have an integrated classification layer [9, 10]. Besides this way of classification, it is also possible to use the extracted face representations in combination with other classification models, like a k-Nearest Neighbor (k-NN) model, a Support Vector Machine (SVM) or Random Forest.

1.4 Syndrome Classification

For the task of syndrome classification, there are in general three approaches found in the existing literature [4]. First of all, a face can be classified based on whether a syndrome is present or not, in which case the patient is a healthy individual. Secondly, a face can be classified based on

which syndrome is present, chosen from a set of predefined syndromes. And lastly, a face can be classified on whether a specific syndrome is present or not. As stated before, this last approach, a binary classification for a specific syndrome, i.e. a syndrome vs. control classification, is the most applicable for this project, as doctors usually already have a suspicion of a specific syndrome. However, relevant research concerning one of the two other classification tasks will be discussed as well to get a complete overview and to see which techniques could be useful.

One way of obtaining a face representation based on an image is to annotate the image with certain landmarks. These landmarks can be annotated manually or generated automatically and can be used in several ways to create a face representation.

1.4.1 Gabor wavelets

One form of processing found landmarks, is to apply Gabor wavelets to the patches around these landmarks [12, 13, 14, 15]. Gabor wavelets are filters that are modelled after the receptive field of simple cells in the primary visual cortex in mammals [16]. They can have different spatial sizes and orientations, and they are capable of detecting edges in an image. Gabor wavelets could be seen as a predecessor of convolutional layers in a neural network and are also lightning invariant. Concatenating the analysed textures around the annotated landmarks leads to a face representation.

There has also been some research that combines Gabor wavelets and morphometric methods [17]. In morphometric methods, not the textures surrounding landmarks are taken into account, but the distances and angles between the landmarks are calculated, normalised and used as face representation [18, 19].

1.4.2 Local Binary Patterns

Another way to analyse the textures surrounding landmarks is to apply Local Binary Patterns (LBPs) [20, 21]. In an LBP, a pixel is labelled in a binary way according to the value of its surrounding pixels [22]. For each surrounding pixel, the value of the original pixel that is being evaluated is subtracted. Depending on whether this difference is positive or negative, a 1 or 0 is added to the binary description of the original pixel. Again, concatenating these binary representations leads to a face representation.

1.5 Explainability

Aside from creating a model that could guide doctors in diagnosing patients, it would also be interesting to know why that model came to a certain conclusion. Therefore, it would be insightful to know which extracted features from an image weighed most in the decision. In that way, doctors could learn from the model and understand its predictions. Hence, an attempt will be made to explain the predictions of some of the models used in this project.

1.6 Aim

This project will focus on experimenting with different models for the task of syndrome. vs control face classification for a specific syndrome. The used models will be compared with each other, to see whether one model can be found that performs best for all of the 12 included syndromes in this project.

2 Methods

As this project focuses on the syndrome vs. control classification of images of faces, different methods of face representations and classification models will be described in this section.

2.1 Data

The available data for this project is either extracted from published papers or created in the Radboudumc. For each syndrome, a data set is created with patient images and control images, to be able to classify in a binary way. All the images were cropped around the face of the patient and any existing digits or other indicators on the images were removed. If possible, images were rotated so that the eyes were on a horizontal level. Lastly, all the images were padded to be a square by extending the background. There was some variety in the data for both the patient and control data set. For example, not all patients looked straight into the camera with their eyes open, sometimes parts of the face were either covered by hands or hair, and the facial expression varied from crying to smiling.

2.1.1 Syndromes

In total, the data of 12 syndromes was taken into account for this project. In Table 1 an overview is presented of the amount of syndromic data available, including the age range and the number of males and females present.

Syndrome	**Nr of images**	**Age range**	**Male**	**Female**
ADNP	N = 33	0 - 35	N = 17	N = 16
ANKRD11	N = 25	2 - 38	N = 16	N = 9
CDK13	N = 30	2 - 54	N = 9	N = 21
DEAF1	N = 19	2 - 25	N = 12	N = 7
DYRK1A	N = 16	1 - 29	N = 10	N = 6
EHMT1	N = 39	0 - 41	N = 18	N = 21
FBXO11	N = 17	0 - 17	N = 13	N = 4
KDVS	N = 75	0 - 46	N = 35	N = 40
SON	N = 18	1 - 34	N = 9	N = 9
WAC	N = 12	3 - 23	N = 4	N = 8
YY1	N = 10	1 - 39	N = 4	N = 6
22q11	N = 48	0 - 54	N = 28	N = 20

Table 1: Overview of syndrome data.

2.1.2 Controls

The control data set was mainly made in the Radboudumc itself and consisted of patients with an unknown form of ID. It was decided to use these images as a control set, as this mimics the real-world application of this classification problem the best. The assumption was made that there is no overlap between the syndromes present in the control set and the chosen syndromes. In Table 2 an overview is presented of the amount of data available, including the age range and the number of males and females present.

	Nr of images	**Age range**	**Male**	**Female**
Controls	N = 370	0 - 52	N = 223	N = 147

Table 2: Overview of control data.

2.1.3 Control selection

A control set was created for each of the 12 syndromes. Per syndrome, one control patient was selected for each syndromic patient, such that there was an even distribution of both labels. These control patients were selected based on gender, age and ethnicity, as was done in the development of the Hybrid model [23]. For the images that were extracted from papers, it was not always known how old the patients were on the image, thus in these cases, the age was estimated. If a control patient with the same age could not be found, a control patient was searched for in a larger age range. This age range runs from a third younger than the syndromic patient to a third older. This range was chosen as ageing has a smaller effect on facial characteristics for relatively older patients than younger ones. For example, there is a more observable difference in facial characteristics between a one and two years old, than between an 18 and 19 years old. If no control patient could be found in this age range, the syndromic patient was excluded from the data set.

2.2 Models

In this project, five different models are used to experiment with. The face representation and classifier used for each model is described below, as well as the experiments that are done with these models.

2.2.1 Deepface model

The Deepface model is developed by Facebook and is a convolutional neural network that uses deep learning to perform the task of face verification [9]. Their approach is two-fold. First, they have a way to extract and align all the faces from the data set, and second, they have a way to perform the actual face verification.

In short, the face alignment happens by detecting 6 fiducial points on the 2D image of the face. These fiducial points are then used to warp the images to appear frontal, using a 3D model of a generic face.

Layer (type)	Output Shape	Param #
C1 (Conv2D)	(None, 142, 142, 32)	11648
M2 (MaxPooling2D)	(None, 71, 71, 32)	0
C3 (Conv2D)	(None, 63, 63, 16)	41488
L4 (LocallyConnected2D)	(None, 55, 55, 16)	62774800
L5 (LocallyConnected2D)	(None, 25, 25, 16)	7850000
L6 (LocallyConnected2D)	(None, 21, 21, 16)	2829456
F7 (Flatten)	(None, 7056)	0
F8 (Dense)	(None, 4096)	28905472
D9 (Dropout)	(None, 4096)	0
F10 (Dense)	(None, 8631)	35361207

Total params: 137,774,071
Trainable params: 137,774,071
Non-trainable params: 0

Figure 1: Structure of the Deepface model for face recognition, with 8631 different identities.

The face verification and recognition happens by using a 9 layer convolutional neural network, of which the architecture is visible in Figure 1. The last layer of the network determines whether the model is used for face recognition, i.e. the last layer has the same number of units as different identities present in the data, or face verification, i.e. the last layer is binary.

In this structure, both fully connected convolutional layers are included, as well as locally connected convolutional layers. A locally connected layer differs from the default fully connected convolutional layer as it learns a different set of filters at each location.

Knowing that all of the input is already aligned, due to the previous face alignment step, researchers reasoned that the spatial stationarity assumption necessary for convolutional layers cannot hold. This assumption does not hold because each spatial region in the data has different local properties, for example, there are more edges present in the area of the eyes than the cheeks. Thus, the researchers customised the architecture of their model, based on the properties of the data.

The first three layers serve as a form of preprocessing to extract low-level features, whilst the locally connected layers are used to extract more high-level features which are then combined in the raw 4096-dimensional face representation that is the result of layer F8.

The Deepface model is trained on the Social Face Classification data set, which is a private data set of Facebook. This data set consists of 4.4 million faces from ± 4,000 different persons. Neither the data set nor the code of this research is made publicly available. However, there are some open-source implementations available on GitHub and there are other, smaller, data sets which can be used to train this model.

The Deepface model [9] is used to create a face representation. As the data already consists of cropped and squared faces, the decision was made to only use the network to obtain the face representation and skip the face alignment step. A publicly available implementation of the Deepface convolutional neural network, as well as pre-trained weights, have been taken from GitHub[1]. This model is trained on the VGGFace2 data set [24] which consists of more than 3.3 million images of ± 9,000 identities.

The original model was written using Tensorflow 1.0, so the code has been rewritten so it is compatible with Tensorflow 2.0. As the model was trained on the VGGFace2 data set for face recognition, the output had ± 9,000 dimensions. Thus, the last two layers (a dropout and a dense layer) were removed to obtain the 4096-dimensional raw face representation.

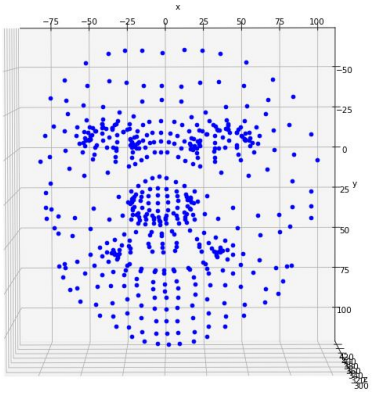
The Deepface face representation is used in combination with a k-Nearest Neighbours classifier to result in a probability or a prediction. The k value is set to three, as a low, uneven, number makes sense as there is not a lot of data available. The decision was made to not train the Deepface model itself for classification, as there was not enough data to do this.

2.2.2 PointNet model

For this project, the university was able to use and test the capabilities of the model called FaceReader, developed by the company VicarVision[2]. Besides being able to classify emotions, this model is capable of locating 510 3D points on a face, based on a 2D image. It does this by using Active Appearance Models [25], which is a model to map 3D shapes to an image using key points and that is trained by VicarVision with a private data set of annotated images. It should be noted that FaceReader cannot always detect these points. If there was no landmark representation found, the image was excluded from the data set. An example of this 3D landmark representation can be seen in Figure 2.

[1] `https://GitHub.com/swghosh/DeepFace`
[2] `https://www.vicarvision.nl/products/facereader/`

(a) Example image from `https://generated.photos/faces`.



(b) 3D landmarks of example image.

Figure 2: Example image and its generated 3D landmarks.

As this face representation is 3D, a different kind of classifier has to be used. The PointNet model takes 3D points as input and gives a probability per class as output [26]. The architecture of this model can be seen in Figure 3. The architecture consists of a classification network, which is relevant for this project, and a segmentation network, which will not be used, but will be explained shortly for completeness.

First of all, the model should be invariant to any rotations of the 3D input as that is not related to the class. To correct for this, the model includes two transformations with t-nets. A t-net is again a small neural network which includes 1D convolutional layers, max-pooling and a dense function.

Second, these 3D points that form a shape are unordered as they do not have a specific order in which they have to be evaluated. All possible permutations of these 3D points should give the same output. To handle this problem, PointNet transforms the input using a symmetry function. The symmetry function is a commutative operation which results in the same vector, despite the order of the input. The symmetry function consists of multi-layer perceptron (mlp) and a max-pooling function.

Last, the 3D points should not only be considered as isolated points, as the local structure and distances between points are relevant for the shape. As the classification network only results in one vector which contains the global features, another technique should be applied to obtain the local feature vectors. Hence, the segmentation network combines global features with per point

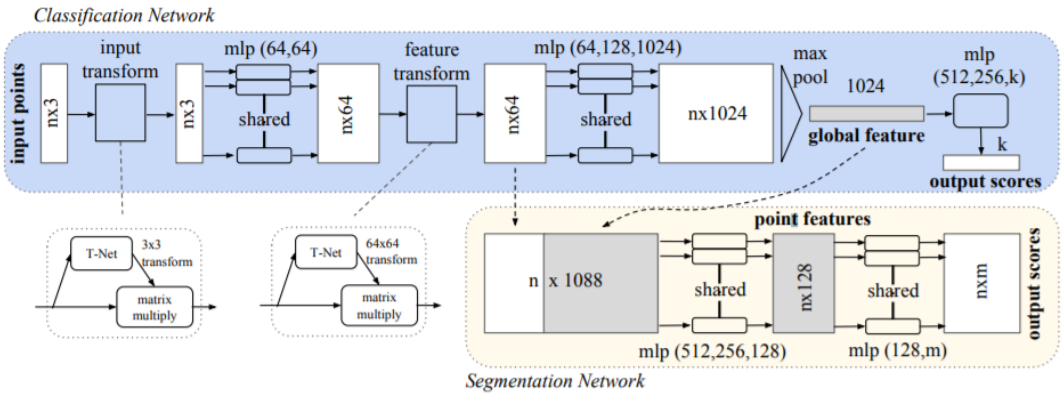


Figure 3: Architecture of the PointNet model [26].

features, again performs a feature transformation, which leads to class probabilities for each of the 3D points. However, this segmentation network will not be used in this project.

2.2.3 Distance model

Besides directly using the 510 3D landmarks from the FaceReader model as a representation for classification, the distances between these 3D points can be used as well. Inspired by previously mentioned morphometric methods [18, 19], the pairwise distances of these 3D points were calculated and concatenated. As using all of the 510 points would lead to ± 130k features, which is quite a lot computation wise, the decision was made to split the face in half. This decision was made after consultation with the Radboudumc and makes sense as you would expect both halves of the face to be mostly symmetric in their facial dysmorphisms. The face was split across the vertical middle line, leading to two halves of 270 points, where the exact middle itself was included in both sides. An example of this split can be seen in Figure 4.

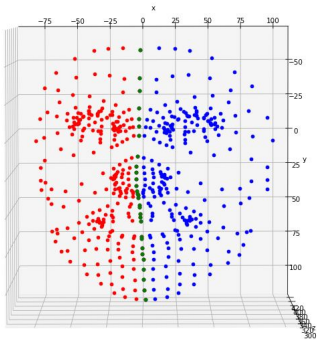


Figure 4: Split FaceReader landmark representation. The green vertical middle line is included in both of the halves of the face.

The pairwise distances of these 270 3D points were calculated, leading to ± 36k features per side and concatenated to ± 72k features in total per image. These distances were classified by a Random Forest. The decision was made to use a Random Forest as this classifier can return the importance per feature. This can hopefully lead to some insight in which features are most decisive in this classification task.

2.2.4 Ensemble model

The three previously mentioned models will be combined to see whether this Ensemble model performs better than the models separately. Following the idea of ensemble learning, where multiple weak learners can combine into one strong learner. In this Ensemble model, the mean will be taken of all probabilities and the mode of the predictions. Only the images that have a FaceReader landmark representation are taken into account in this Ensemble model.

2.2.5 Hybrid model

FaceNet

FaceNet is a model developed by Google and also consists of a convolutional neural network that uses deep learning to solve the tasks of face recognition and verification [10]. However, the researchers did not put the focus on the architecture of this network, but on the loss used. They wanted to come up with a low-dimensional face representation that could be put in Euclidean space in such a way that faces with the same identity are located close together and faces with different identities far apart. Hence, the convolutional neural network was treated as a black box and they used the triplet loss to obtain the desired properties of the final embedding. As input for the model, a tight bounding box is drawn around detected faces, but other than scaling and translating, no face alignment or preprocessing is used.

The embedding is the 128-dimensional output of the convolutional neural network after an L2 normalisation layer. During training, the triplet loss was applied to the embedding. The triplet loss is calculated for a tuple of 3 different faces, called the anchor, the positive and the negative. The anchor and the positive have the same identity and the negative has a different one. The triplet loss then ensures that the distance between the anchor and the positive is minimised in Euclidean space, and the distance between the anchor and the negative is maximised.

FaceNet is trained on a private data set from Google of 100-200 million images of ± 8 million different identities. Again, neither the data set nor the code of this research is made publicly available.

Openface
An example of a publicly available implementation of the FaceNet model is the Python library Openface [27]. This library contains a pre-trained model which also uses the triplet loss and has roughly the same architecture as the original model from FaceNet. Since a smaller publicly available data set is used for training, and this specific library focuses on mobile applications, which of course limits the computation power, a smaller version of the model was used, with fewer trainable parameters.

CFPS
A different well-performing approach combines morphometric and texture-based features to put the face representations in a multidimensional space called the Clinical Face Phenotype Space (CFPS) [28]. These face representations are acquired by automatically detecting 9 landmarks in a face, extending these to 36 using Active Appearance Models (AAMs), and extracting two different kinds of vectors from these landmarks. The first vector is the appearance vector that consists of a concatenation of the pixel intensities around the 9 initial landmarks. The second vector is the shape vector that consists of the normalised pairwise distances between all the 36 landmarks. Both of these feature vectors were concatenated and consequently, a PCA was performed, to reduce the representation to a 340-dimensional feature vector.

Hybrid model
Recent research done at the Radboudumc concerning syndrome classification has combined two of the previously mentioned models into a Hybrid model [23]. In this model, the Python library Openface [27] has been used as well as the CFPS representations [28]. This combination makes sense as both of these methods aim to put the face representation in a multidimensional Euclidean space. Both face representations were concatenated, leading to a combined 468-dimensional face representation.

This model was evaluated by measuring the Clustering Improvement Factor (CIF) [23, 28]. The CIF is used to evaluate how well a group of positives is clustered within a group of negatives. This CIF score was compared, by using a statistical test, to the CIF you would expect when all face representations are labelled randomly, whilst maintaining the original ratio of positives and negatives. The results of this evaluation method for the Hybrid model were promising in comparison with each of these models alone. However, no attempt has been made to apply this Hybrid model to syndrome vs. control classification, as is the aim of this project.

This model combines two models which are both written in a different programming language. The Openface model is written in Python and the CFPS model in MATLAB. A pipeline combining these two models has been made and is available for use at the Radboudumc. However, one downside of this model, is that it is computationally expensive, and analysing a single image can take up to one hour, which is of course impractical. Therefore, the previously calculated face representations of most of the data present at the Radboudumc will be used. This leads to some small difference in the number of samples used per syndrome, in comparison with the number of samples used for the other models.

As the Hybrid model combines the Openface [27] and CFPS [28] representation, which are both aimed at putting face representation in a multidimensional Euclidean space, the k-NN classifier will be used, again with k set to three.

2.3 Experiments

Because of the small amounts of data available, there will be no static split into train, test and validation data. For each syndrome, and for each model, a Leave One Out cross-validation is used. Thus, a split is made into train and test set, where the test set only contains one sample and this process is repeated for all samples. The class probabilities of this sample are saved and later compared with all the true labels, leading to a ROC curve and an area under the curve (aroc) score, as well as the specificity (spec) or the true negative rate, defined as $spec = \frac{TN}{TN+FP}$, and the sensitivity (sens) or the true negative rate, defined as $sens = \frac{TP}{TP+FN}$.

The simple classifiers used, k-NN and Random Forest, only need to be fit to the data. The PointNet model, however, is trained for 4 epochs on the training data. These syndrome vs. control experiments are run three times per syndrome to ensure that the scores are not due to a lucky pick of easy to classify control patients. Hence, the mean of three trials is reported, as well as the standard deviation to see how stable the results are. For completeness, the number of unique controls for each trial run per syndrome is reported as well.

Besides these experiments for syndrome vs. control classification, there will also be some syndrome vs. syndrome classification experiments. Although this type of classification is not the goal in this project, it can be beneficial to examine these results as they might give some insight in the robustness of the performance of a model and to see whether there might be a bias in one or more sets of data.

2.4 Visualisations

As mentioned before, it would be beneficial if there is some kind of visualisation of which features are most decisive and thus most relevant for the classification. This could give some insight into the decision making of the model, as well as serving as a check for biases in the data. This was possible for two of the models, the Deepface model and the Distance model.

2.4.1 Deepface model

As the Deepface model is a neural network, it is possible to plot the activation per patient in each of the convolution layers, summed over all the filters. The average of the activation of all patients for a specific syndrome, and the chosen control patients, will be shown per layer. This might give some insight in which areas of the face lead to higher activation and might be important in the classification decision.

2.4.2 Distance model

Since a Random Forest classifier has been chosen for this model, it is possible to visualise the most important feature according to the classifier. This feature importance is based on the decrease in node Gini impurity, weighted by the probability of reaching that specific node. The importance of all features returned by the model sum up to one. The features that have an accumulated importance of at least 0.8, with a maximum of 30 features, are visualised, to get some insight in which distances are most important.

2.5 Practical application

Ideally, the best performing model or models would be integrated into a practical tool that doctors can actually use for guiding their diagnosis of a specific syndrome. Doctors would be able to use this tool by uploading an image of a patient, selecting the syndrome they are interested in and then they would receive a prediction or probability for that specific syndrome. If the uploaded images would be sent to the Radboudumc, they could be used to increase the database and further

improve upon the models.

In this project a prototype has been made for this tool, using Python and the library Flask[3]. Research will continue at the Radboudumc after this project to see how this tool can be made available within the Radboudumc and later on also for doctors outside the Radboudumc.

[3] https://flask.palletsprojects.com/en/1.1.x/

3 Results

3.1 Syndrome vs. Control classification

In Table 4 the mean results of three runs of the syndrome vs. control experiments are shown. The number of samples mentioned refers to the number of syndromic patients included in the set, so the total set, including controls, is twice as large. The best performing model, looking at all the three evaluation metrics, is shown in bold. Besides reporting the aroc value, the ROC curves have also been plot. In Figure 5 the ROC curves of the third run for the syndromes ADNP and EHMT1 are visible. The other syndromes are visible in Figure 14 and 15 in the Appendix.

The Deepface model has in, general, quite a bad performance. Only for some syndromes, like ANKRD11 and EHMT1 it has an acceptable performance, looking at all the three evaluation metrics.

The PointNet model has quite a varying performance. For the syndromes, DEAF1, EHMT1 and 22q11, it performs relatively well, whilst the performance for the syndromes ADNP, ANKRD11, DRYK1A, FBXO11, WAC is bad, with the sensitivity score even below 0.5. A general trend can be seen of a lower sensitivity value than the specificity value.

The Distance model performs well for the syndromes CDK13, EHMT1, WAC, YY1 and 22q11. For the other syndromes, the performance is bad with often a sensitivity value lower than 0.5.

The Ensemble model performs well for the syndromes CDK13, EHMT1 and YY1. The general idea of ensemble learning, where combined weak learners result in a strong and robust learner, clearly did not happen here, as the performance is not evidently higher than of the models separately and also the performance still fluctuates just as much as the single models.

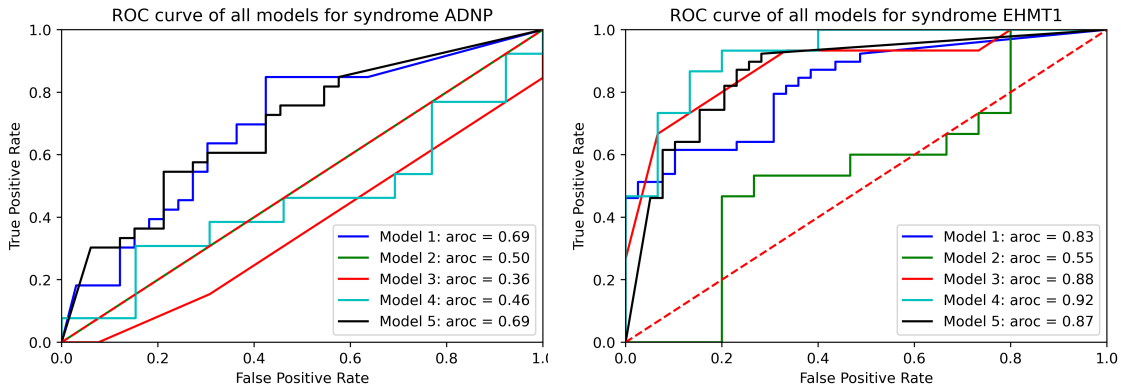
The Hybrid model performs best overall, as it has the best performance for most of the syndromes. Even for the syndromes it did not have the best performance for, it still had quite a decent performance, except for the syndrome YY1.

There is a general trend visible that some syndromes are easier to classify for most of the models than others. Examples of easy to classify syndromes are EHMT1, YY1 and 22q11 and examples of more difficult to classify syndromes are ANDP, DYRK1A and FBXO11.

Syndrome	**Run 1**	**Run 2**	**Run 3**	**Total**
ADNP	24	24	24	33
ANKRD11	13	14	14	25
CDK13	14	16	14	30
DEAF1	8	12	8	19
DYRK1A	9	11	12	16
EHMT1	16	19	16	39
FBXO11	8	11	11	17
KDVS	31	38	32	75
SON	11	12	13	18
WAC	7	8	7	12
YY1	5	6	5	10
22q11	25	25	25	45

Table 3: Number of unique control patients chosen per run.

In the appendix an overview is presented in Table 5 of the standard deviation of the scores for these three trials, to see how robust the models perform when different sets of control patients are chosen. Most models perform quite stable as the standard deviation is low, with some exceptions here and there. Especially the PointNet model has the most varying scores as it has the highest average standard deviation.



(a) ADNP

(b) EHMT1

Figure 5: ROC curves of ADNP and EHMT1 of run 3. Model 1 refers to the Deepface model, Model 2 refers to the PointNet mode, Model 3 refers to the Distance model, Model 4 refers to the Ensemble model and Model 5 refers to the Hybrid model.

For each trial, new control patients were chosen, but due to the age, gender and ethnicity constraints, there is some overlap in the control set for the three runs. In Table 3 an overview is presented of unique controls per run, so these control patients were not included in any of the two other runs, as well as the total number of controls chosen per run.

3.2 Syndrome vs. Syndrome classification

For each of the five models, a syndrome vs. syndrome classification task has been run. For each pair of syndromes, one was seen as the syndrome data set and the other as the control data set. The number of selected controls for the different models is visible in Figure 19 in the Appendix. If there were not enough samples in total, the threshold was set at 2 patient samples and 2 control samples, a zero is displayed.

In Figure 6 the performance of the Deepface model is displayed. A lot of aroc scores are around 0.5, which is chance level. The Deepface model performed well for the syndromes ANKRD11 and EHMT in the syndrome vs. control classification task, as can be seen in Figure 4. However, in the syndrome vs. syndrome classification task, the Deepface model only performed relatively well for the syndrome ANKRD11.

In Figure 7 the performance of the PointNet model is displayed. It is visible that for quite some pairs of syndromes, not enough samples were present, as there are a lot of zeros in the top right corner. This due to the fact that not all of the syndromic patients have a FaceReader landmark representation and thus these scores could not be calculated. The same pattern is visible for the Distance model in Figure 8 and for the Ensemble model in Figure 9. The PointNet model did not perform well for the syndrome pairs that did have enough samples. Even for the syndromes DEAF1, EHMT1 and 22q11, which had a high performance in the previous classification task, there are no high aroc scores.

In Figure 8 the performance of the Distance model is displayed. The Distance model performed well on the syndromes CDK13, EHMT1, WAC, YY1 and 22q11 in the previous task, but here it is visible that the model only performs well for the WAC syndrome. The other syndromes have highly fluctuating performances.

	Deepface model	**PointNet model**	**Distance model**	**Ensemble model**	**Hybrid model**
ADNP	N = 33 aroc = 0.612 spec = 0.737 sens = 0.455	N = 13 aroc = 0.668 spec = 0.944 sens = 0.383	N = 13 aroc = 0.420 spec = 0.528 sens = 0.398	N = 13 aroc = 0.609 spec = 0.867 sens = 0.218	**N = 33** **aroc = 0.734** **spec = 0.757** **sens = 0.576**
ANKRD11	N = 25 aroc = 0.809 spec = 0.707 sens = 0.773	N = 19 aroc = 0.589 spec = 0.800 sens = 0.362	N = 19 aroc = 0.716 spec = 0.712 sens = 0.581	N = 19 aroc = 0.854 spec = 0.892 sens = 0.580	**N = 21** **aroc = 0.835** **spec = 0.714** **sens = 0.857**
CDK13	N = 30 aroc = 0.559 spec = 0.478 sens = 0.600	N = 16 aroc = 0.780 spec = 0.876 sens = 0.595	N = 16 aroc = 0.722 spec = 0.753 sens = 0.607	N = 16 aroc = 0.786 spec = 0.733 sens = 0.629	**N = 30** **aroc = 0.870** **spec = 0.733** **sens = 0.778**
DEAF1	N = 19 aroc = 0.499 spec = 0.526 sens = 0.614	**N = 15** **aroc = 0.743** **spec = 0.794** **sens = 0.657**	N = 15 aroc = 0.448 spec = 0.501 sens = 0.319	N = 15 aroc = 0.728 spec = 0.727 sens = 0.567	N = 19 aroc = 0.666 spec = 0.614 sens = 0.772
DYRK1A	N = 16 aroc = 0.694 spec = 0.750 sens = 0.479	N = 10 aroc = 0.508 spec = 0.700 sens = 0.267	N = 10 aroc = 0.525 spec = 0.633 sens = 0.400	N = 10 aroc = 0.520 spec = 0.767 sens = 0.300	**N = 16** **aroc = 0.750** **spec = 0.688** **sens = 0.646**
EHMT1	N = 39 aroc = 0.843 spec = 0.846 sens = 0.632	N = 15 aroc = 0.704 spec = 0.667 sens = 0.733	N = 15 aroc = 0.849 spec = 0.867 sens = 0.689	**N = 15** **aroc = 0.901** **spec = 0.822** **sens = 0.844**	N = 39 aroc = 0.882 spec = 0.855 sens = 0.761
FBXO11	N = 17 aroc = 0.580 spec = 0.627 sens = 0.392	N = 15 aroc = 0.473 spec = 0.503 sens = 0.433	N = 15 aroc = 0.444 spec = 0.544 sens = 0.362	N = 15 aroc = 0.492 spec = 0.546 sens = 0.340	**N = 17** **aroc = 0.593** **spec = 0.529** **sens = 0.588**
KDVS	N = 75 aroc = 0.740 spec = 0.582 sens = 0.778	N = 52 aroc = 0.494 spec = 0.484 sens = 0.522	N = 52 aroc = 0.463 spec = 0.593 sens = 0.342	N = 52 aroc = 0.628 spec = 0.587 sens = 0.516	**N = 70** **aroc = 0.748** **spec = 0.676** **sens = 0.681**
SON	N = 18 aroc = 0.718 spec = 0.592 sens = 0.648	N = 10 aroc = 0.807 spec = 0.892 sens = 0.558	N = 10 aroc = 0.515 spec = 0.592 sens = 0.492	N = 10 aroc = 0.797 spec = 0.700 sens = 0.575	**N = 18** **aroc = 0.698** **spec = 0.630** **sens = 0.685**
WAC	N = 12 aroc = 0.418 spec = 0.639 sens = 0.278	N = 9 aroc = 0.665 spec = 0.843 sens = 0.477	**N = 9** **aroc = 0.828** **spec = 0.764** **sens = 0.732**	N = 9 aroc = 0.722 spec = 0.764 sens = 0.542	N = 12 aroc = 0.790 spec = 0.694 sens = 0.778
YY1	N = 10 aroc = 0.683 spec = 0.567 sens = 0.767	N = 7 aroc = 0.748 spec = 0.794 sens = 0.587	**N = 7** **aroc = 0.883** **spec = 0.801** **sens = 0.849**	N = 7 aroc = 0.801 spec = 0.746 sens = 0.786	N = 10 aroc = 0.545 spec = 0.433 sens = 0.567
22q11	N = 45 aroc = 0.756 spec = 0.503 sens = 0.778	N = 16 aroc = 0.759 spec = 0.824 sens = 0.681	**N = 16** **aroc = 0.603** **spec = 0.673** **sens = 0.562**	N = 16 aroc = 0.622 spec = 0.666 sens = 0.541	N = 45 aroc = 0.793 spec = 0.720 sens = 0.729

Table 4: Results of the mean of three trials of syndrome vs. control classification with 12 syndromes.

In Figure 9 the performance of the Ensemble model is displayed. Just like the three models this Ensemble model is based on, this model did not perform well for most of the syndromes. It performs relatively well for the ANKRD11 and KDVS, but still varies quite a lot.

In Figure 10 the performance of the Hybrid model is displayed. It is visible that for more than half of the syndrome pairs, the aroc score is above 0.5, which shows that the Hybrid model also performs best in the syndrome vs. syndrome classification task.

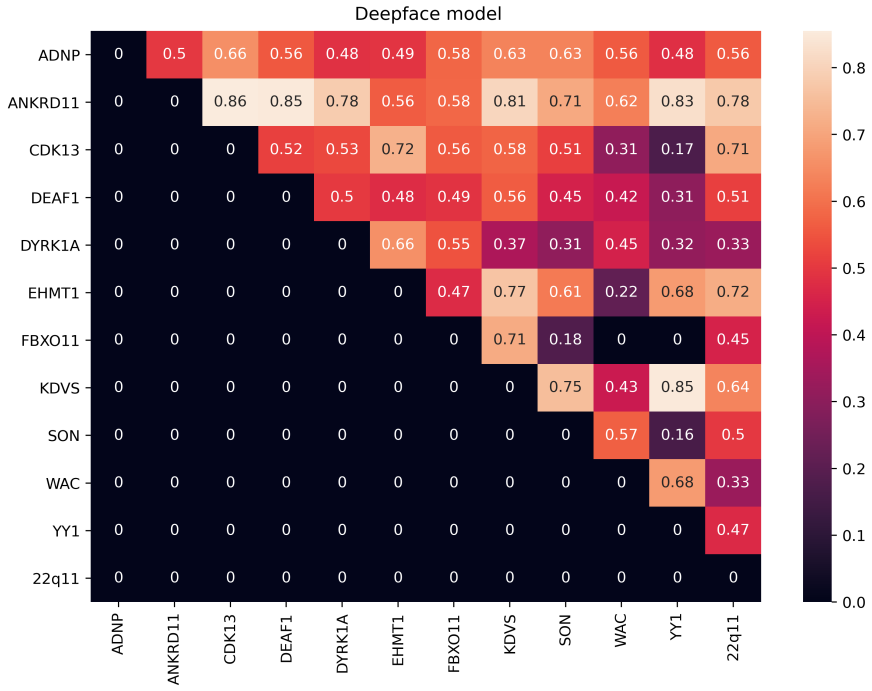


Figure 6: Aroc scores for the Deepface model for Syndrome vs. Syndrome classification.

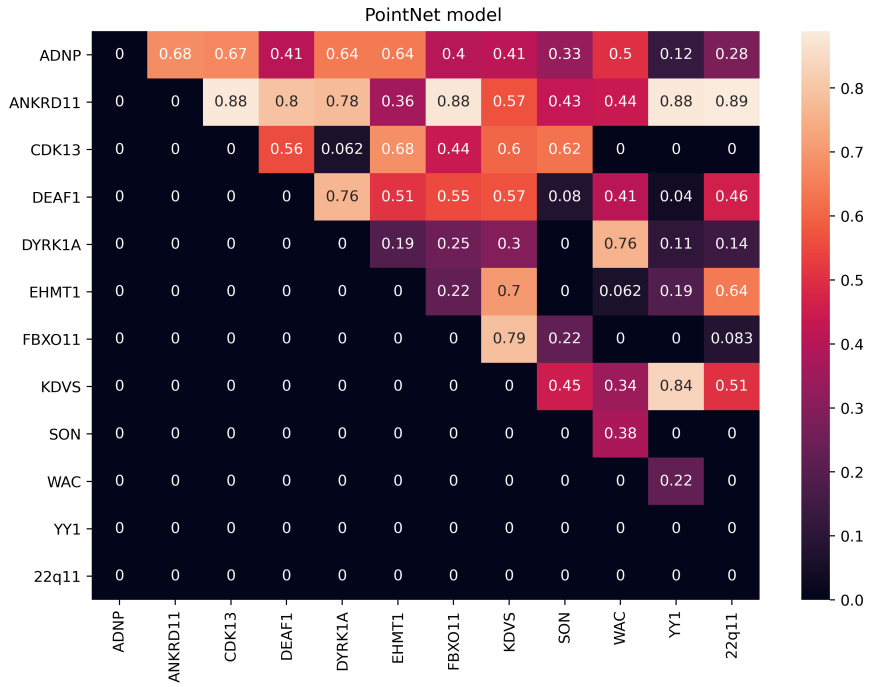


Figure 7: Aroc scores for the PointNet model for Syndrome vs. Syndrome classification.

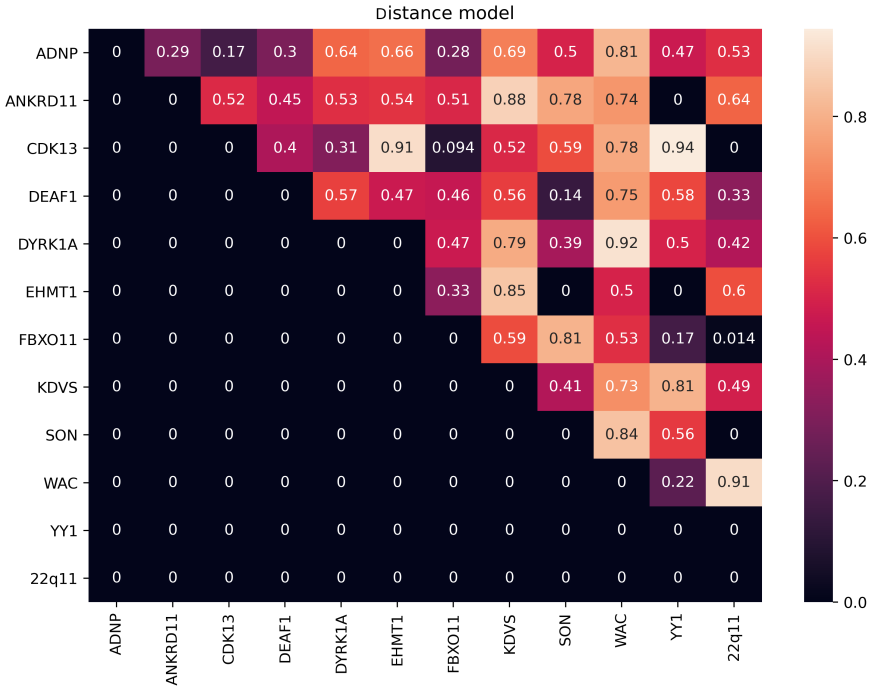


Figure 8: Aroc scores for the Distance model for Syndrome vs. Syndrome classification.

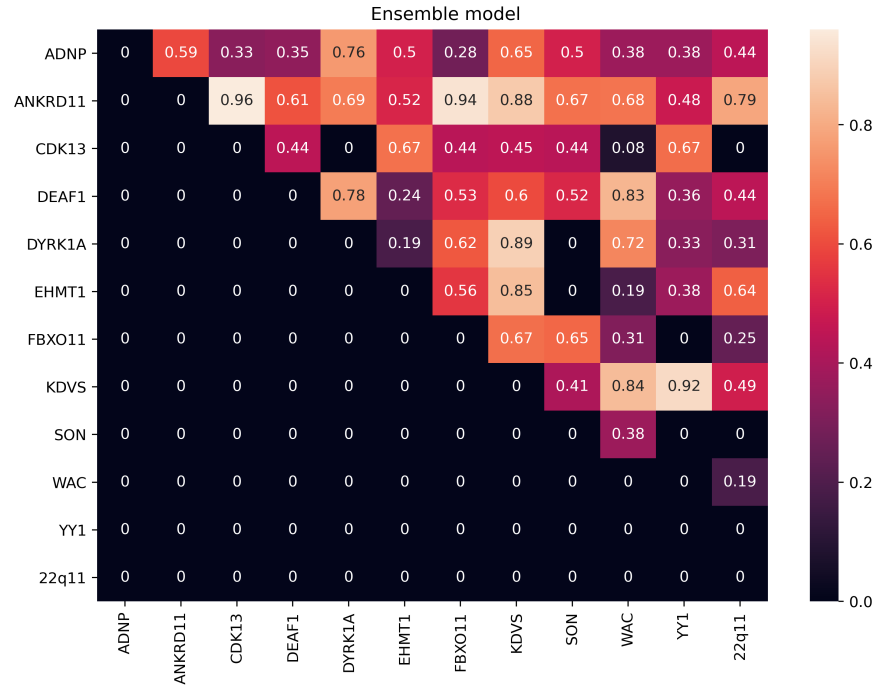


Figure 9: Aroc scores for the Ensemble model for Syndrome vs. Syndrome classification.

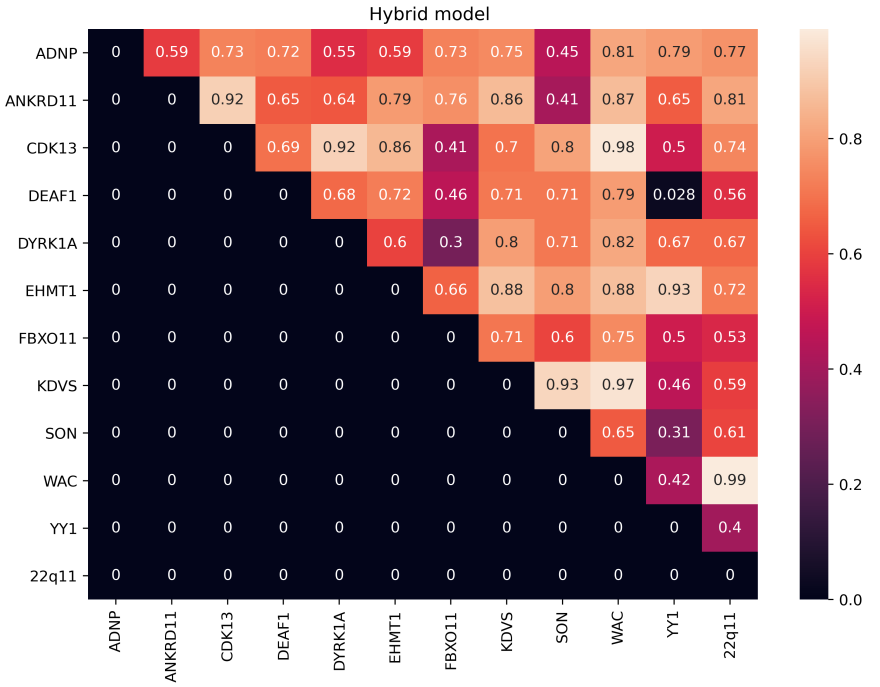
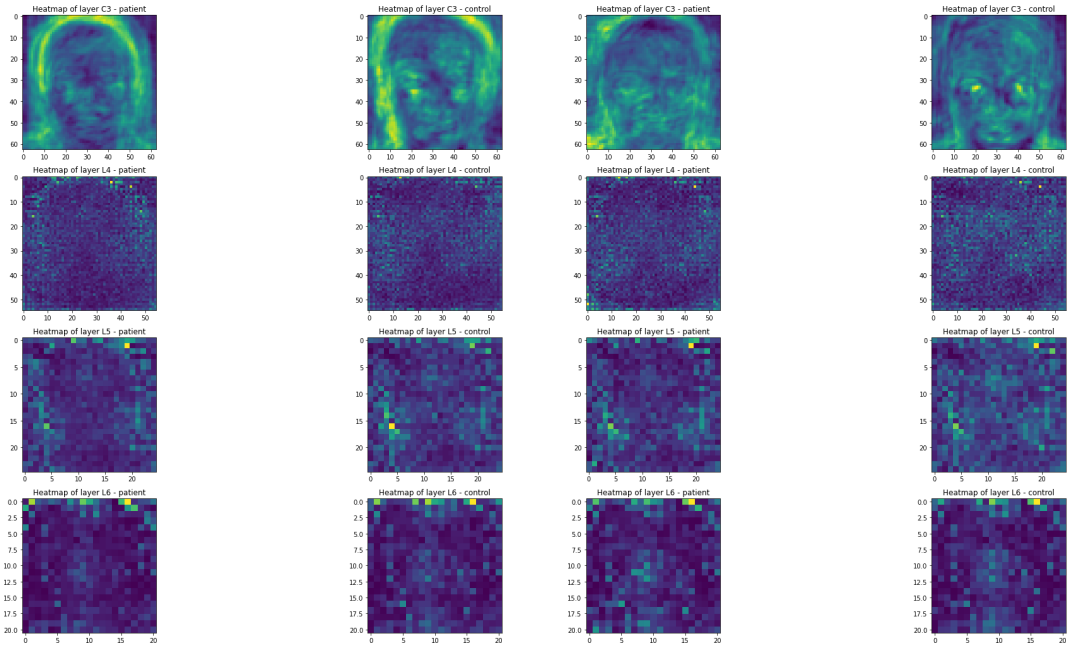


Figure 10: Aroc scores for the Hybrid model for Syndrome vs. Syndrome classification.



(a) ANKRD11: average activation of convolutional layers.

(b) WAC: average activation of convolutional layers.

Figure 11: Average activation of all the filters in different convolutional layers for the syndromes ANKRD11 and WAC.

3.3 Visualisations

For two of the used models in this project, an attempt has been made to visualise the decision making of the model to gather some insight in which features are most decisive.

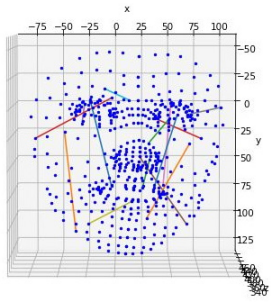
3.3.1 Deepface model

In Figure 11 the activation of each layer, averaged over the set of ANKRD11 and WAC patients and selected control patients, is visible. The other 10 syndromes are included in the Appendix in Figure 16, 17 and 18. The syndromes ANKRD11 and WAC are chosen as they are the syndromes for which the Deepface model performs best and worst, respectively, looking at Table 4.

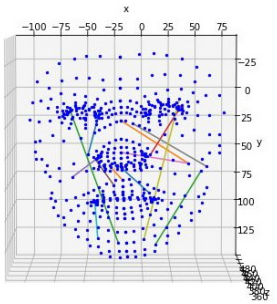
However, there is no clear difference in activation when comparing the average activation for the syndromic patients and the control patients, for both of the chosen syndromes in Figure 11. So, even when the Deepface model is performing well, plotting the activation for each layer does not give any clear insight into the features which bear the most importance.

3.3.2 Distance model

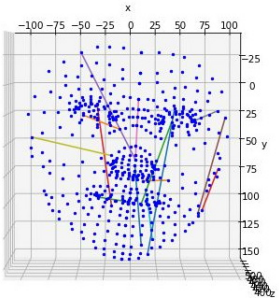
In Figure 12 the most important features are plotted. These features together have at least 0.8 importance in the decision of the Random Forest, with a maximum of 30 features. It is visible that a lot of syndromes only have between 10 and 15 features that carry some importance, which is not a lot, knowing that the total number of features is $\pm$ 72k. In many of the plots, the drawn features are, roughly, symmetric in both halves of the face, as was expected.



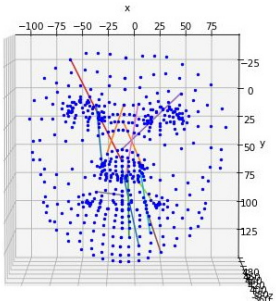
(a) ADNP: 14 features with 0.81 importance.



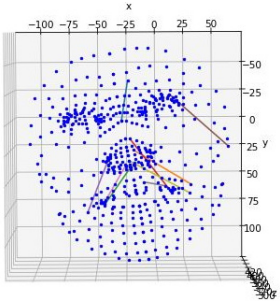
(b) ANKRD11: 13 features with 0.81 importance.



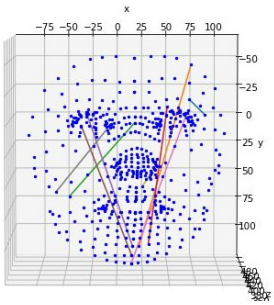
(c) CDK13: 14 features with 0.81 importance.



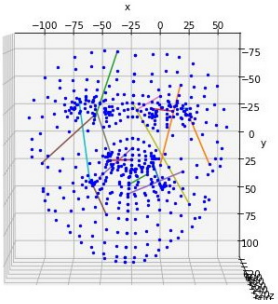
(d) DEAF1: 12 features with 0.82 importance.



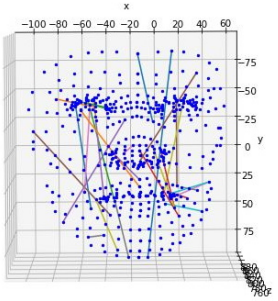
(e) DYRK1A: 9 features met 0.80 importance.



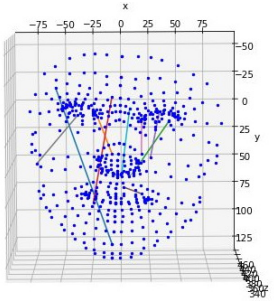
(f) EHMT1: 9 features with 0.83 importance.



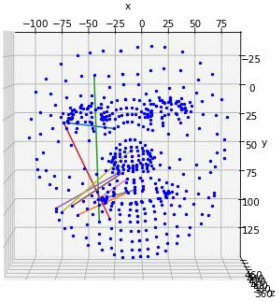
(g) FBXO11: 16 features with 0.81 importance.



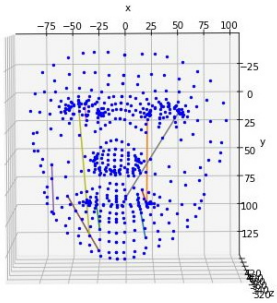
(h) KDVS: 30 features and 0.56 importance.



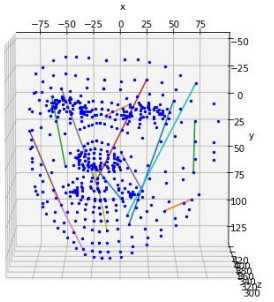
(i) SON: 10 features with 0.84 importance.



(j) WAC: 9 features with 0.90 importance.

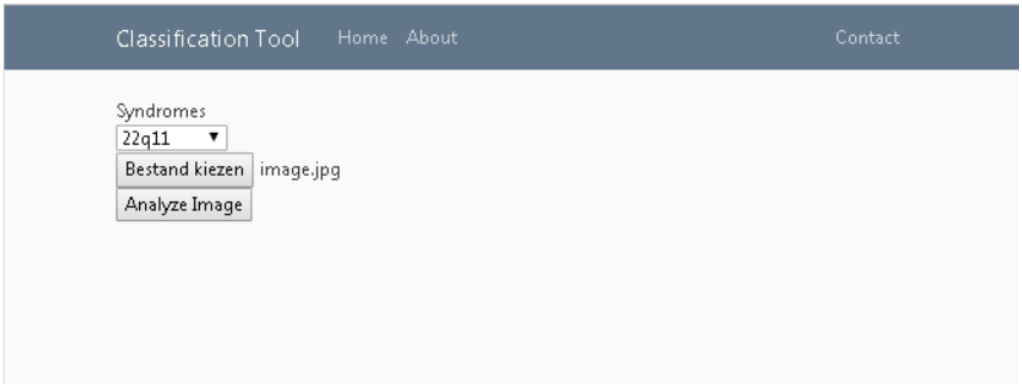


(k) YY1: 9 features with 0.90 importance.

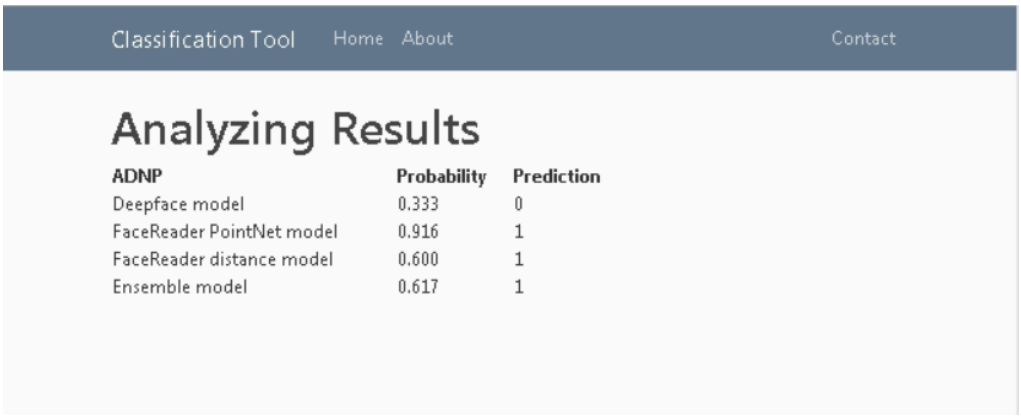


(l) 22q11: 13 features with 0.81 importance.

Figure 12: Plots of features that hold at least 0.8 importance in Random Forest classifier.



(a) Screenshot of Home page of the prototype.



(b) Screenshot of Analyze page of the prototype.

Figure 13: Screenshots of the prototype.

3.4 Prototype

In Figure 13 screenshots of the web tool made with Flask and Python are visible. Research will continue at the Radboudumc to develop this tool further, with the goal that doctors within the Rabdoudumc and outside this specific hospital can use this tool to help with diagnosing specific syndromes.

4 Discussion

In the future, it would probably be most practical if there was one model for this task of syndrome vs. control classification, such that all focus can be put on this single model. In this project, it has been seen that all models had varying performances over the 12 different syndromes. Hence, it might be difficult to find one model that is suitable for all syndromes, especially as there are even more syndromes than the 12 used in this project.

Using a model that is pre-trained on face verification and recognition for the task of syndrome vs. control classification did not lead to good results.This could be due to the data not being preprocessed in the same way as had been done for the original Deepface model, but it could also be because these two domains differ too much, making transfer learning impractical.

The 3D landmark representations that were used as face representations in this project did not result in a consistent performance. So, although this face representation is less susceptible to a bias than 2D images, it is not decisive for the syndrome vs. control classification task, as might have been expected. It also needs to be noted that the 3D landmark representation was not that robust, as quite a large number of images did not have a 3D landmark representation. This might be due to the variety in the data, with regards to partly covered or angled faces.

In this project, ensemble learning did not lead to better results. Hence, the usual reasoning behind ensemble learning, that multiple weak learners together can work as a strong learner, is not the case here. This could be because the models are not dissimilar enough.

It would be best if future research focuses on expanding the Hybrid model [23], as that model has been found to have the best results. More progress can be made concerning the computation time and explainability of the predictions it makes. As this model combines two face representations, relating the prediction of the model back to the most important features might be quite difficult.

From the syndrome vs. syndrome classification task, it became clear that there is no distinct bias in the syndromic data for the different models. No syndrome was performing well in the syndrome vs. control classification task that also performed well for all the models with regard to all the other syndromes in the syndrome vs. syndrome classification task. As the results of the syndrome vs. control classification task were also quite stable over the three different runs, there is only a small probability there is a hidden bias in the control data.

For two models, visualisations have been made to gather some insight into the most important features. However, making heatmaps of the activation did not result in a clear indication of the most important features. Plotting the most important distances was more clear, but it needs to be decided by doctors whether these features can be considered as actually relevant, or that they are still too general.

5 Conclusion

This project has compared several models for the task of syndrome vs. control face classification. It has been found that the Hybrid model has the best performance for most of the syndromes. The performance of all the used models fluctuates over the different syndromes. Hence, it will be quite challenging and possibly even too ambitious to come up with one model that performs well for all genetic syndromes. This research has used multiple different models, as well as face representations to explore what the possibilities are, and therefore adds value to the current body of research regarding this topic. Hopefully, this project can guide new research in the right direction of a model that performs well for all syndromes in a syndrome vs. control classification task.

References

[1] Pallab K Maulik, Maya N Mascarenhas, Colin D Mathers, Tarun Dua, and Shekhar Saxena. Prevalence of intellectual disability: a meta-analysis of population-based studies. *Research in developmental disabilities*, 32(2):419–436, 2011.

[2] Donna K Daily, Holly H Ardinger, and Grace E Holmes. Identification and evaluation of mental retardation. *American family physician*, 61(4):1059–1067, 2000.

[3] TC Hart and PS Hart. Genetic studies of craniofacial anomalies: clinical implications and applications. *Orthodontics & craniofacial research*, 12(3):212–220, 2009.

[4] Yaron Gurovich, Yair Hanani, Omri Bar, Nicole Fleischer, Dekel Gelbman, Lina Basel-Salmon, Peter Krawitz, Susanne B Kamphausen, Martin Zenker, Lynne M Bird, et al. Deepgestalt-identifying rare genetic syndromes using deep learning. *arXiv preprint arXiv:1801.07637*, 2018.

[5] Tracy Dudding-Byth, Anne Baxter, Elizabeth G Holliday, Anna Hackett, Sheridan O'Donnell, Susan M White, John Attia, Han Brunner, Bert de Vries, David Koolen, et al. Computer face-matching technology using two-dimensional photographs accurately matches the facial gestalt of unrelated individuals with the same syndromic form of intellectual disability. *BMC biotechnology*, 17(1):90, 2017.

[6] Juan J Cerrolaza, Antonio R Porras, Awais Mansoor, Qian Zhao, Marshall Summar, and Marius George Linguraru. Identification of dysmorphic syndromes using landmark-specific local texture descriptors. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 1080–1083. IEEE, 2016.

[7] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[8] 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.

[9] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[10] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[11] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[12] Hartmut S Loos, Dagmar Wieczorek, Rolf P Würtz, Christoph von der Malsburg, and Bernhard Horsthemke. Computer-based recognition of dysmorphic faces. *European Journal of Human Genetics*, 11(8):555–560, 2003.

[13] Stefan Boehringer, Tobias Vollmar, Christiane Tasse, Rolf P Wurtz, Gabriele Gillessen-Kaesbach, Bernhard Horsthemke, and Dagmar Wieczorek. Syndrome identification based on 2d analysis software. *European Journal of Human Genetics*, 14(10):1082–1089, 2006.

[14] Stefan Boehringer, Manuel Guenther, Stella Sinigerova, Rolf P Wurtz, Bernhard Horsthemke, and Dagmar Wieczorek. Automated syndrome detection in a set of clinical facial photographs. *American Journal of Medical Genetics Part A*, 155(9):2161–2169, 2011.

[15] Şafak Saraydemir, Necmi Taşpınar, Osman Eroğul, Hülya Kayserili, and Nuriye Dinçkan. Down syndrome diagnosis based on gabor wavelet transform. *Journal of medical systems*, 36(5):3205–3213, 2012.

[16] Tai Sing Lee. Image representation using 2d gabor wavelets. *IEEE Transactions on pattern analysis and machine intelligence*, 18(10):959–971, 1996.

[17] Tobias Vollmar, Baerbel Maus, Rolf P Wurtz, Gabriele Gillessen-Kaesbach, Bernhard Horsthemke, Dagmar Wieczorek, and Stefan Boehringer. Impact of geometry and viewing angle on classification accuracy of 2d based analysis of dysmorphic faces. *European journal of medical genetics*, 51(1):44–53, 2008.

[18] Ashwin B Dalal and Shubha R Phadke. Morphometric analysis of face in dysmorphology. *Computer methods and programs in biomedicine*, 85(2):165–172, 2007.

[19] Peter Hammond, Tim J Hutton, Judith E Allanson, Bernard Buxton, Linda E Campbell, Jill Clayton-Smith, Dian Donnai, Annette Karmiloff-Smith, Kay Metcalfe, Kieran C Murphy, et al. Discriminating power of localized three-dimensional facial morphology. *The American Journal of Human Genetics*, 77(6):999–1010, 2005.

[20] Juan J Cerrolaza, Antonio R Porras, Awais Mansoor, Qian Zhao, Marshall Summar, and Marius George Linguraru. Identification of dysmorphic syndromes using landmark-specific local texture descriptors. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 1080–1083. IEEE, 2016.

[21] Kurt Burçin and Nabiyev V Vasif. Down syndrome recognition using local binary patterns and statistical evaluation of the system. *Expert Systems with Applications*, 38(7):8690–8695, 2011.

[22] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[23] Roos van der Donk, Sandra Jansen, Janneke HM Schuurs-Hoeijmakers, David A Koolen, Lia CMJ Goltstein, Alexander Hoischen, Han G Brunner, Patrick Kemmeren, Christoffer Nellåker, Lisenka ELM Vissers, et al. Next-generation phenotyping using computer vision algorithms in rare genomic neurodevelopmental disorders. *Genetics in Medicine*, 21(8):1719–1725, 2019.

[24] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *International Conference on Automatic Face and Gesture Recognition*, 2018.

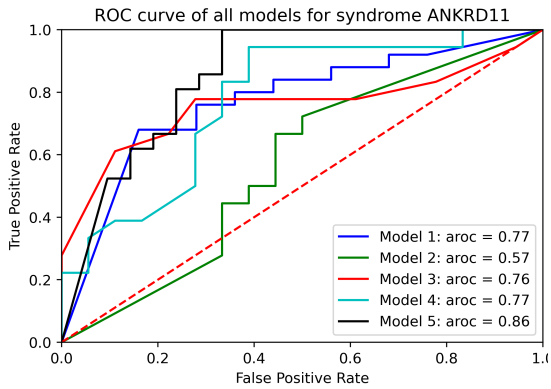
[25] Timothy F Cootes, Cristopher J Taylor, et al. Statistical models of appearance for computer vision, 2004.

[26] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

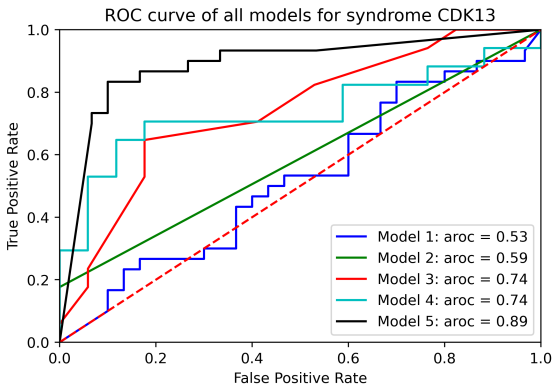
[27] Brandon Amos, Bartosz Ludwiczuk, Mahadev Satyanarayanan, et al. Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science*, 6:2, 2016.

[28] Quentin Ferry, Julia Steinberg, Caleb Webber, David R FitzPatrick, Chris P Ponting, Andrew Zisserman, and Christoffer Nellåker. Diagnostically relevant facial gestalt information from ordinary photos. *Elife*, 3:e02020, 2014.

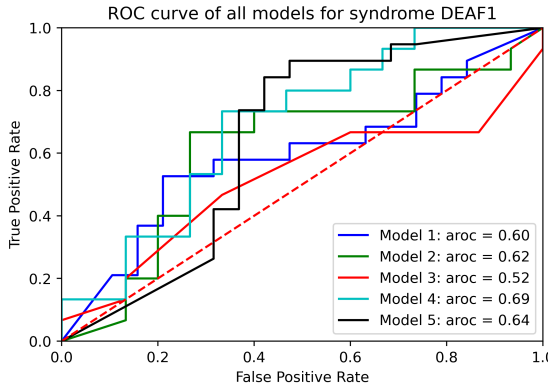
6 Appendix



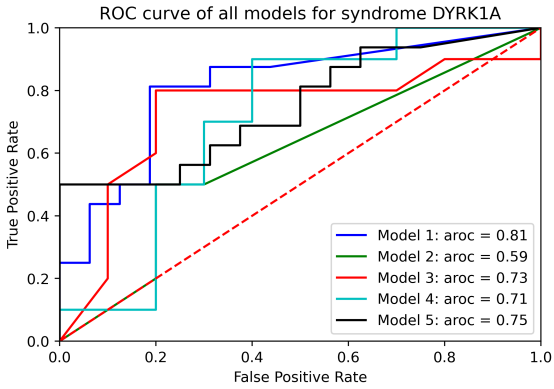
(a) ANKRD11



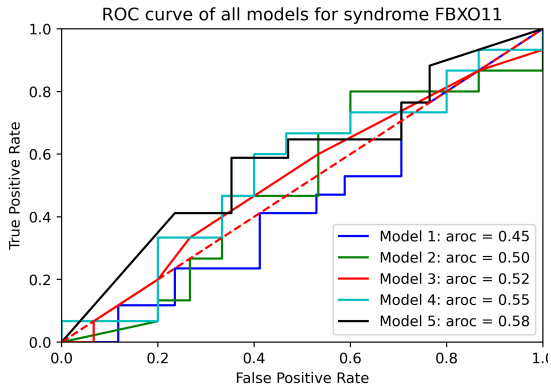
(b) CDK13



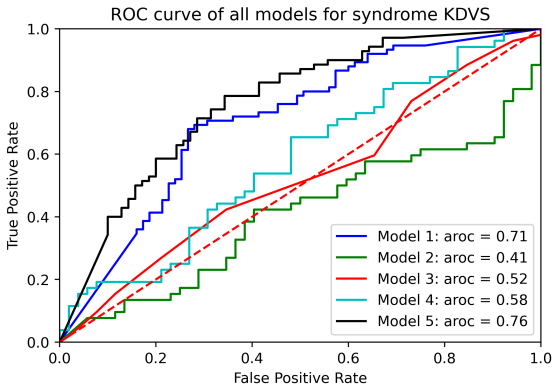
(c) DEAF1



(d) DYRK1A

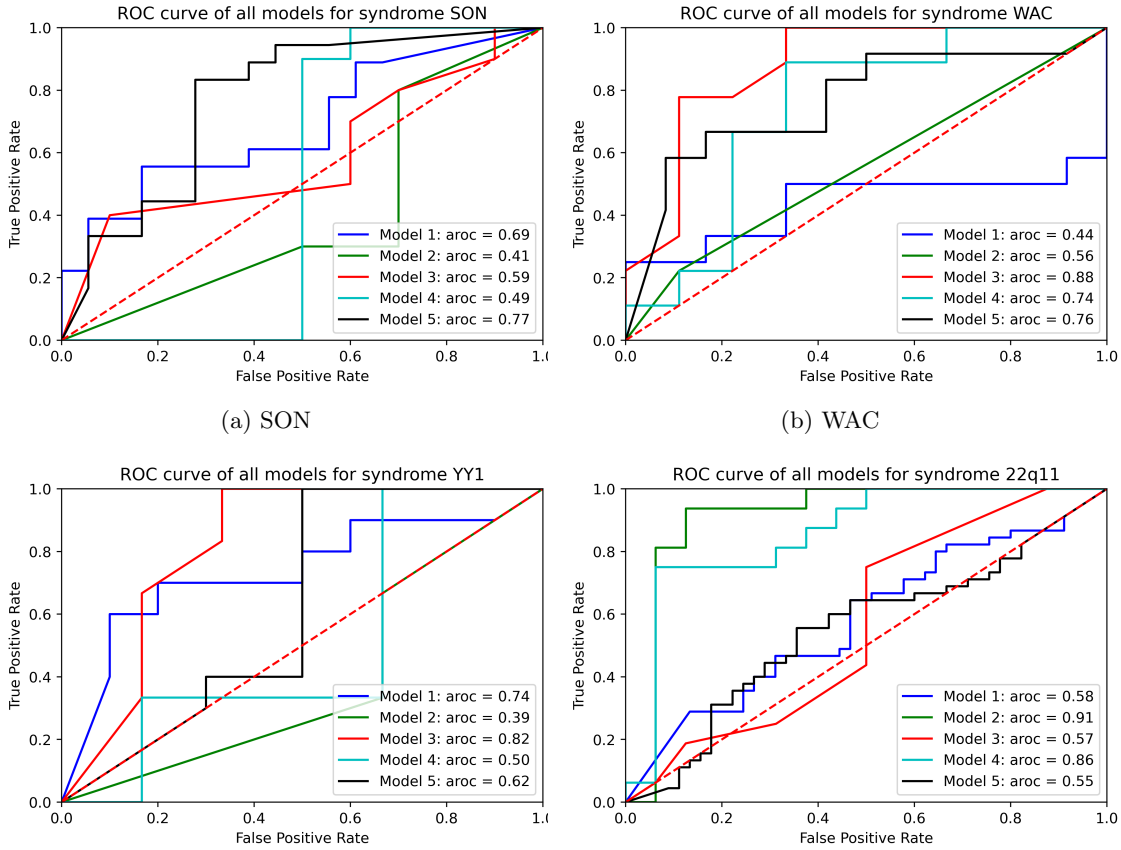


(e) FBXO11



(f) KDVS

Figure 14: ROC curves of run 3. Model 1 refers to the Deepface model, Model 2 refers to the PointNet mode, Model 3 refers to the Distance model, Model 4 refers to the Ensemble model and Model 5 refers to the Hybrid model.



(a) SON

(b) WAC

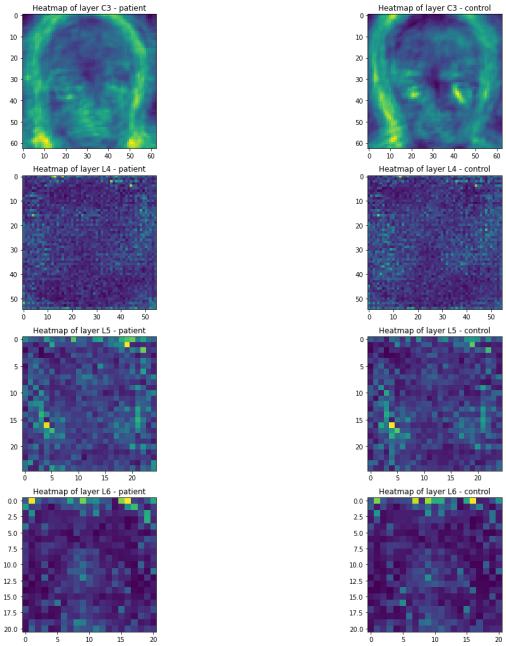
(c) YY1

(d) 22q11

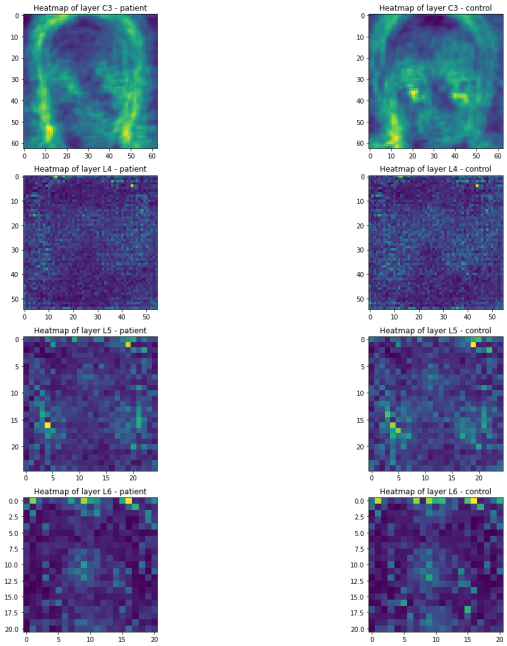
Figure 15: ROC curves of run 3. Model 1 refers to the Deepface model, Model 2 refers to the PointNet mode, Model 3 refers to the Distance model, Model 4 refers to the Ensemble model and Model 5 refers to the Hybrid model.

	Deepface model	**PointNet model**	**Distance model**	**Ensemble model**	**Hybrid model**
ADNP	N = 33 aroc = 0.070 spec = 0.046 sens = 0.046	N = 13 aroc = 0.192 spec = 0.096 sens = 0.464	N = 13 aroc = 0.073 spec = 0.125 sens = 0.097	N = 13 aroc = 0.105 spec = 0.119 sens = 0.256	N = 33 aroc = 0.070 spec = 0.053 sens = 0.109
ANKRD11	N = 25 aroc = 0.033 spec = 0.061 sens = 0.061	N = 19 aroc = 0.085 spec = 0.029 sens = 0.217	N = 19 aroc = 0.076 spec = 0.115 sens = 0.120	N = 19 aroc = 0.044 spec = 0.094 sens = 0.166	N = 21 aroc = 0.035 spec = 0.048 sens = 0.048
CDK13	N = 30 aroc = 0.032 spec = 0.051 sens = 0.067	N = 16 aroc = 0.192 spec = 0.061 sens = 0.313	N = 16 aroc = 0.049 spec = 0.092 sens = 0.118	N = 16 aroc = 0.176 spec = 0.126 sens = 0.143	N = 30 aroc = 0.017 spec = 0.034 sens = 0.102
DEAF1	N = 19 aroc = 0.086 spec = 0.139 sens = 0.031	N = 15 aroc = 0.060 spec = 0.077 sens = 0.081	N = 15 aroc = 0.063 spec = 0.090 sens = 0.047	N = 15 aroc = 0.032 spec = 0.011 sens = 0.088	N = 19 aroc = 0.069 spec = 0.061 sens = 0.080
DYRK1A	N = 16 aroc = 0.130 spec = 0.108 sens = 0.191	N = 10 aroc = 0.135 spec = 0.200 sens = 0.058	N = 10 aroc = 0.111 spec = 0.115 sens = 0.100	N = 10 aroc = 0.074 spec = 0.115 sens = 0.100	N = 16 aroc = 0.062 spec = 0.000 sens = 0.036
EHMT1	N = 39 aroc = 0.018 spec = 0.068 sens = 0.030	N = 15 aroc = 0.201 spec = 0.200 sens = 0.067	N = 15 aroc = 0.046 spec = 0.067 sens = 0.038	N = 15 aroc = 0.082 spec = 0.039 sens = 0.139	N = 39 aroc = 0.015 spec = 0.015 sens = 0.039
FBXO11	N = 17 aroc = 0.158 spec = 0.034 sens = 0.180	N = 15 aroc = 0.098 spec = 0.157 sens = 0.088	N = 15 aroc = 0.050 spec = 0.051 sens = 0.094	N = 15 aroc = 0.070 spec = 0.135 sens = 0.057	N = 17 aroc = 0.055 spec = 0.059 sens = 0.102
KDVS	N = 75 aroc = 0.028 spec = 0.020 sens = 0.050	N = 52 aroc = 0.035 spec = 0.053 sens = 0.065	N = 52 aroc = 0.068 spec = 0.073 sens = 0.049	N = 52 aroc = 0.059 spec = 0.036 sens = 0.044	N = 70 aroc = 0.022 spec = 0.068 sens = 0.030
SON	N = 18 aroc = 0.052 spec = 0.140 sens = 0.032	N = 10 aroc = 0.090 spec = 0.014 sens = 0.274	N = 10 aroc = 0.285 spec = 0.274 sens = 0.181	N = 10 aroc = 0.106 spec = 0.200 sens = 0.066	N = 18 aroc = 0.079 spec = 0.085 sens = 0.064
WAC	N = 12 aroc = 0.040 spec = 0.127 sens = 0.210	N = 9 aroc = 0.198 spec = 0.137 sens = 0.383	N = 9 aroc = 0.109 spec = 0.133 sens = 0.058	N = 9 aroc = 0.173 spec = 0.133 sens = 0.091	N = 12 aroc = 0.024 spec = 0.096 sens = 0.127
YY1	N = 10 aroc = 0.045 spec = 0.153 sens = 0.058	N = 7 aroc = 0.226 spec = 0.180 sens = 0.220	N = 7 aroc = 0.128 spec = 0.077 sens = 0.014	N = 7 aroc = 0.148 spec = 0.099 sens = 0.258	N = 10 aroc = 0.130 spec = 0.058 sens = 0.153
22q11	N = 45 aroc = 0.174 spec = 0.122 sens = 0.059	N = 16 aroc = 0.208 spec = 0.046 sens = 0.283	N = 16 aroc = 0.071 spec = 0.068 sens = 0.063	N = 16 aroc = 0.101 spec = 0.059 sens = 0.071	N = 45 aroc = 0.129 spec = 0.085 sens = 0.201

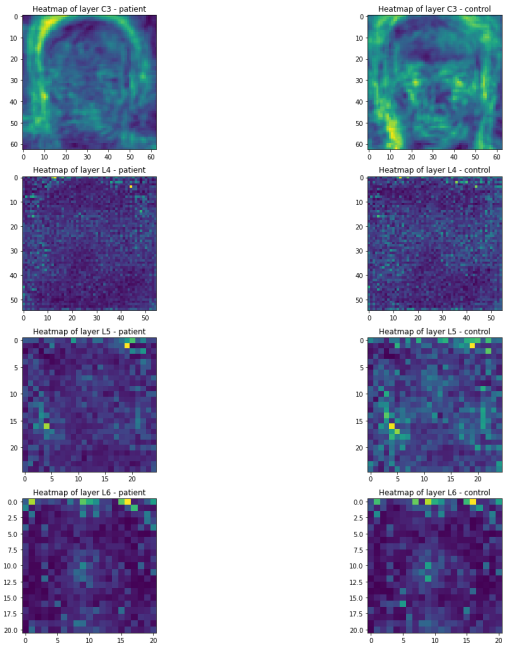
Table 5: Results of the standard deviation of three trials of syndrome vs. control classification with 12 syndromes.



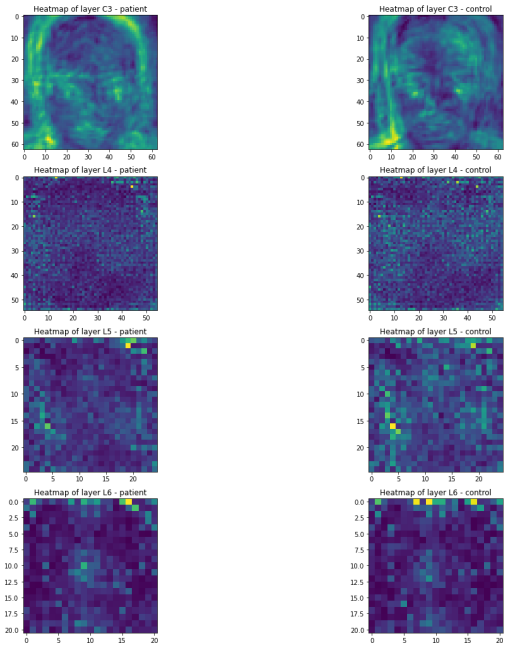
(a) ADNP: average activation of convolutional layers.



(b) CDK13: average activation of convolutional layers.

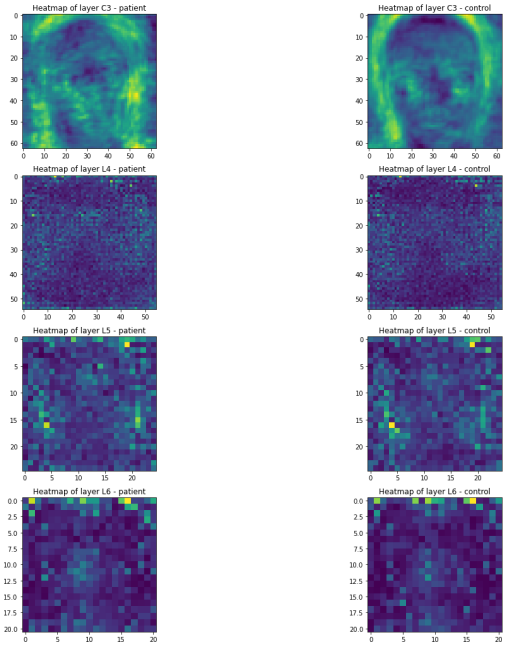


(c) DEAF1: average activation of convolutional layers.

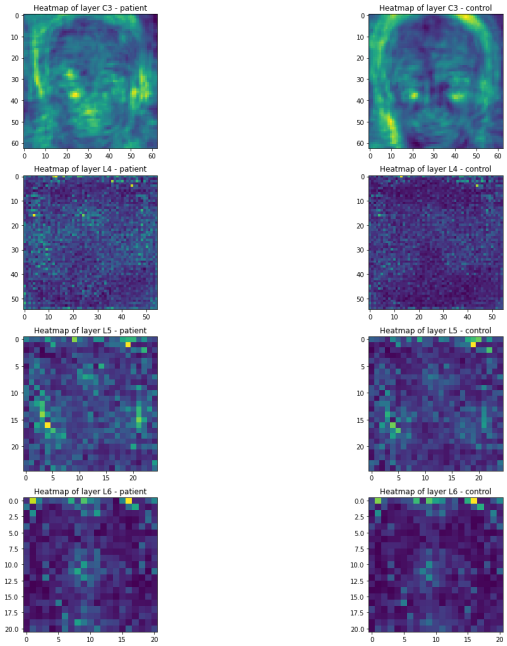


(d) DYRK1A: average activation of convolutional layers.

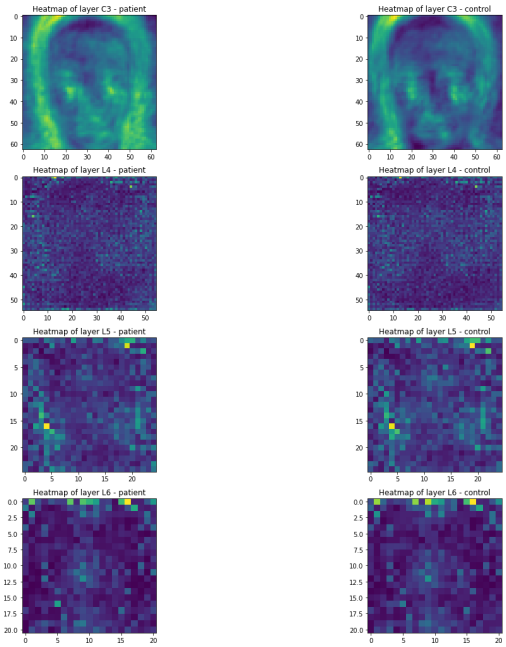
Figure 16: Overview 1/3 of all convolutional layer activation averaged per syndrome.



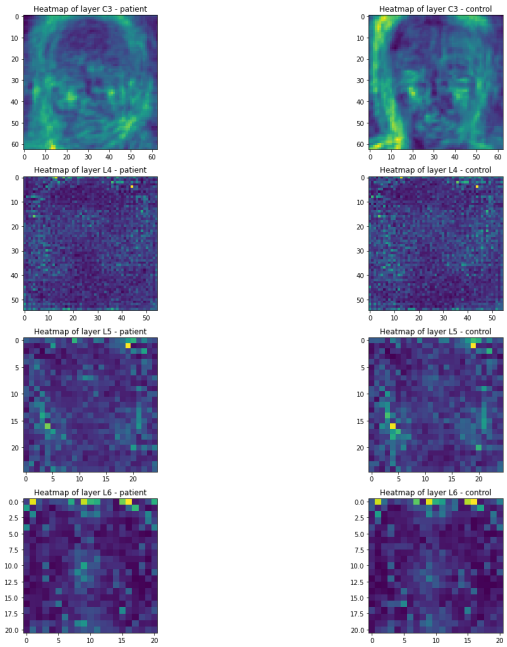
(a) EHMT1: average activation of convolutional layers.



(b) FBXO11: average activation of convolutional layers.

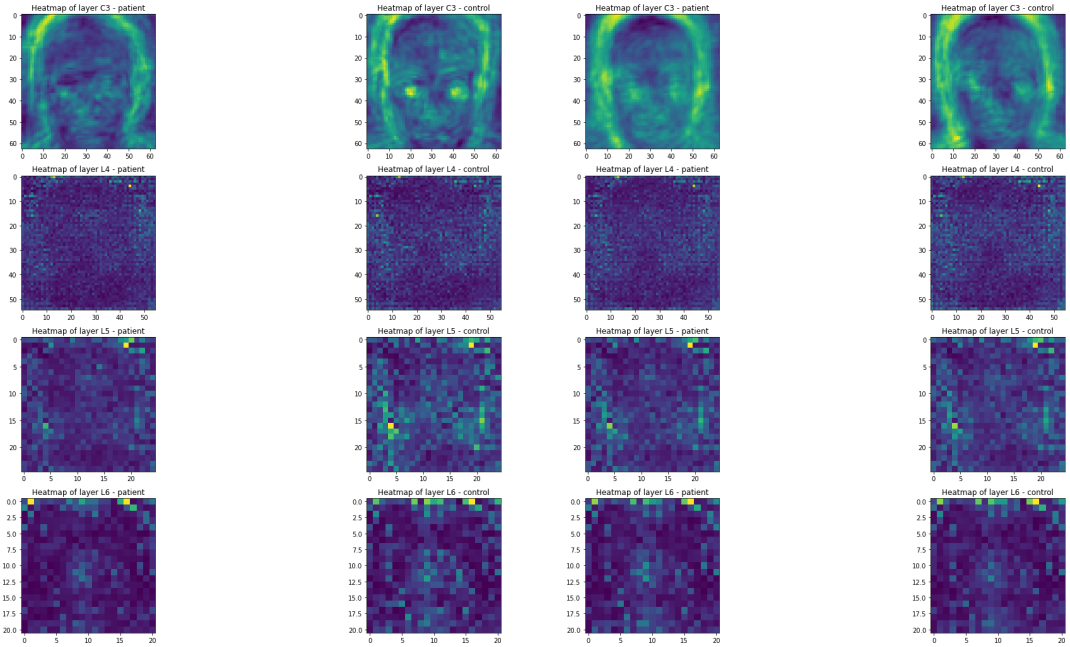


(c) KDVS: average activation of convolutional layers.



(d) SON: average activation of convolutional layers.

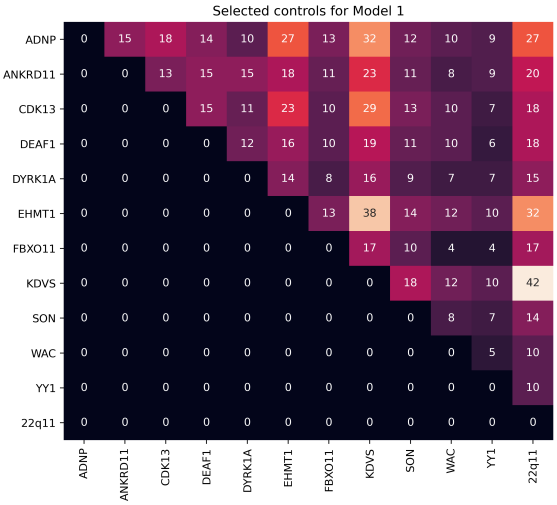
Figure 17: Overview 2/3 of all convolutional layer activation averaged per syndrome.



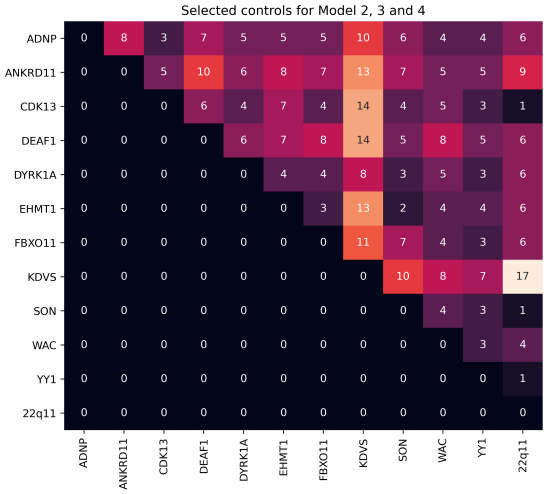
(a) YY1: average activation of convolutional layers.

(b) 22q11: average activation of convolutional layers.

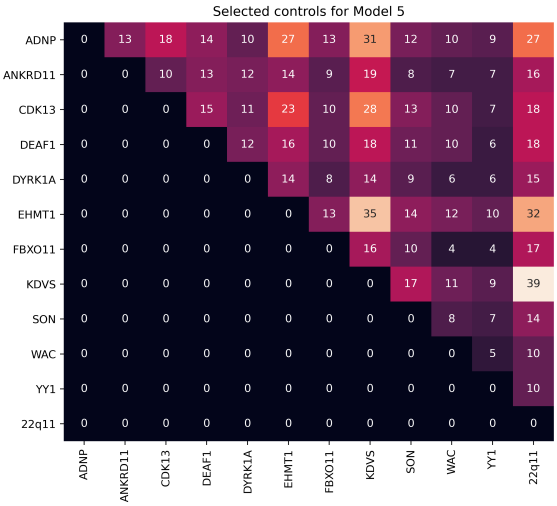
Figure 18: Overview 3/3 of all convolutional layer activation averaged per syndrome.



(a) Number of selected controls for the Deepface model.



(b) Number of selected controls for the PointNet model, the Distance model and the Ensemble model.



(c) Number of selected controls for the Hybrid model.

Figure 19: Overview of the number of selected controls for the Syndrome vs. Syndrome classification for all models.