IT UNIVERSITY OF COPENHAGEN

Affect Detection of Infants using Vision Transformer

Bachelor Thesis

Petya Petrova
pety@itu.dk

Supervisors

Martin Lund Trinhammer (mlut@itu.dk)
Stella Grasshof(stgr@itu.dk)

GitHub Repository

BSc in Data Science
IT University of Copenhagen
May, 2025

Contents

1 Introduction[1]

The only way infants can express their affect is through their faces, however, it could be challenging for new parents to read them, which could lead to missed cues in caregiving. Knowing how to read these affects is also important in healthcare, as they can help in identifying conditions such as autism (Carpenter et al. 2021). The research has proven that children with autism disorder tend to display neutral facial expressions more frequently at a very early age, compared to typically developed children.

Yet, the automation detection of infant faces is still an unexplored area compared to adult faces. Adult facial expressions are documented and used a lot in different vision models, evidenced by large datasets such as the AffectNet (Mollahosseini, Hasani, and Mahoor 2017). Traditional approaches mainly include Convolutional Neural Networks (CNNs), which rely heavily on a large amount of datasets. Recently, some research has introduced Vision Transformers (ViTs) and their better performance when it comes to facial detection tasks. Such transformers have the ability to focus on the most relevant regions of an image by using the self-attention layer, enabling them to capture meaningful patterns even when the data is limited.

This project aims to create a model that can detect infants' affects from images, classifying them into positive, negative or neutral affects by using the City Faces database. It uses a Vision Transformer ViT pre-trained on adult facial expressions. By applying the SmoothGrad technique, the goal is to understand the motives behind that transformer's decisions. Such a model can help new parents and healthcare professionals to better understand infants.

2 Related Work

The automation of facial recognition in infants is challenging because they have different proportions of their faces compared to adults, less texture, fewer wrinkles, and furrows (Onal Ertugrul et al. 2023). Infant facial expressions are described using Facial Action Units (AU) including: AU4 (brow lowerer),AU6 (cheek raiser),AU12 (lip corner puller), AU20 (lip stretcher), AU1 (inner brow raiser), AU3 (inner brows drawn together), AU9 (upper lip raiser), AU28 (lip suck).
However, the lack of labeled infant facial data makes it even more challenging to develop accurate models Fatema (2021). Two of the most well-annotated infant facial expressions datasets are the following: MIAMI- a database of spontaneous behavior and CLOCK (Craniofacial microsomia: Longitudinal Outcomes in Children pre-Kindergarten). .
To address this, tools like PyAfar have been developed. PyAFAR is a Python library created for automated facial action unit recognition in both adults

[1]Parts of the code in this project were assisted by using Generative AI

and infants that was build on the AFAR (Automated Facial Action Recognition)(Hinduja et al. 2023). PyAfar and Afar refer to the same system, with Pyfar being the Python implementation. It is a tool for analyzing facial expressions. It uses ResNet50 models trained on BP4D+ dataset for adults and MIAMI and CLOCK datasets for infants(Hinduja et al. 2023).
Other tools such as Py-Feat was created for facial Action Unit detection that can be adapted to different dataset (Cheong et al. 2023).
A key challenge in the field remains the lack of infant facial expressions data sets. Hausmann et al. 2022 also highlights this issue and it proposes techniques called data augmentation, such as Mosaic, to overcome that. It combines four different images into one during training.
Facial recognition and affect detection are very related tasks because detecting affects starts with recognizing facial features. Traditionally, CNNs, Convolution Neural Networks, were the most common approaches when it comes to facial detection (Trigueros, Meng, and Hartnett 2018). By using deep learning, there is no need for developing different features, because the model learns them by itself. The key limitation of this deep learning method is that it requires a large amount of data (Trigueros, Meng, and Hartnett 2018).
A more recent study shows that Vision Transformers (ViTs) are a slightly more effective solution for such a task like facial detection than CNNs (Rodrigo, Cuevas, and García 2024). In a performance comparison across emotion categories, ViT slightly outperforms the other models, as the table below suggests, with the highest mean of accuracy of 59.2% among the other CNN models. This model offers the best overall consistency across emotions, particularly excelling in recognizing neutral, happy, and angry affects, however still struggles with digust and fear emotion; like the other models too. Even though some models show higher accuracy on some of the categories, ViTs are the most balanced model because it has the highest mean and a moderate standard deviation.

Table 1: General accuracy of the models (in percent) across emotion categories Table 3 from (Rodrigo, Cuevas, and García 2024)

Model	**Neu**	**Hap**	**Sad**	**Sur**	**Fea**	**Dis**	**Ang**	**Mean**	**STD**
MobileNet	66.6	86.8	53.2	60.1	45.6	31.0	57.5	57.3	17.4
ResNet	73.8	84.1	48.0	60.2	38.2	24.4	54.0	54.7	20.4
XceptionNet	77.3	82.3	51.5	61.3	44.8	24.7	57.7	57.1	19.6
ViT	75.3	86.7	56.5	64.0	42.8	25.1	64.3	59.2	20.4
CLIP	79.7	85.1	44.1	45.1	25.8	9.4	43.8	47.5	25.8
GPT-4o-mini	72.6	79.5	56.8	62.9	27.6	34.9	62.1	56.6	17.5

3 Methods

3.1 Vision Transformer

A Vision transformer is a deep learning model suitable for vision tasks such as image processing (Vidhya 2023). The key concept about this type of model relies in the architecture and more specifically in the attention layer. That layer helps the model to decide which parts, called patches, of the images are more important by giving them more attention in specific areas. This key mechanism allows Vision Transformers to look at the whole image at once and find important patterns across the whole image (Vidhya 2023).

From a higher perspective, here are the steps of how a Vision Transformer actually works:

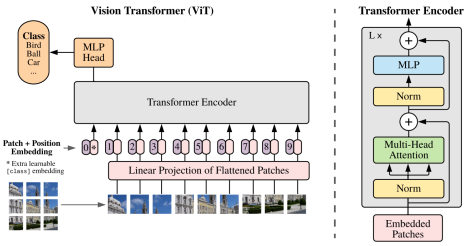

Figure 1: ViT architecture from Figure 1, Model Overview from (Dosovitskiy et al. 2020) paper

- **Steps**[2]

1. Patch Creation

The very first step is to split the input image (example 256x256 pixels) into patches in (16x16) pixels each. Then each patch is flattened into a vector.

2. Linear Projection (Embedding Patches)

Each patch is passed through a linear projection layer via a weight matrix. Simply put, each patch is multiplied by this matrix to turn it into an embedding. This way the patch flattened vector turns into a new smaller but more meaningful vector.

3. Adding Positional Encoding

Positional Encodings are added to each patch embedding. This way the model knows the position and the order of the patches, for example, that Patch 1 is

[2]These steps are adapted from (Correll 2023) and supplemented by clarifications obtained via Generative AI.

sitting next to Patch 2.

4. Classification Token [CLS] token
A classification token is created per each patch with random values, which is later updated based on the gathered information. The purpose of this token is to act as a summary of the entire image, containing the most valuable information, to make the final classification.

5. Pass the Positional Encoding and the CLS token through the 12 Transformer Blocks
Each of these 12 transformer blocks consists of two sublayers- the Multi-Head Self-Attention layer and the two-layer Multi-Layer Perception (MLP), a small neural network. The Multi-Head Self Attention Layer is the key concept of the ViT architecture. This layer calculates how strongly each patch connects to the other patches by calculating attention scores. These scores are calculated by scaling the dot product of queries (Q), keys (K), and values (V) of a token for each patch. Q, K, and V are vectors that represent the following: Query (Q)-what the patch is trying to find in rather patches, key (K)- what the patch represents to others, and value (V) the actual information that the patch can share with others

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

The attention equation (Equation 1) is taken from Figure 8 in (Carpenter et al. 2021).

Then, the scores are passed through a softmax function that converts the result into probabilities. Then these probabilities are used to compute the weighted sum of values for each patch. The end result shows which patches are the most important for each other. This allows the model to focus on the most relevant parts of the image.

6. Updating and improving patch information
Then another transformation is done by using the attention outputs and the original input, called the residual connection. Then that vector is normalized and passed through a small Neural Network called MLP.

7. Repeating through all Transformer blocks
This process of attention, normalizing the residual connection, and passing the vectors through the small neural network is passed across all 12 blocks.

8. CLS token updates
The CLS token is updated at every transformer block by gathering information from all other patches using the attention output. This is the key for the model to be able to build a summary of the entire image.

9. Predictions

The updated CLS tokens were needed for the model to make its final predictions by passing them through a fully connected layer and a softmax function, which spits the probabilities of each image being each category.

3.2 Smooth Grad

To see which pixels contributed the most for the transformer to classify an image, I used a technique called SmoothGrad. This method introduces noise to each image several times, and then it calculates how much the model's prediction would change (Smilkov et al. 2017). It has two hyperparameters: the noise and the sample size (Smilkov et al. 2017). Here are the steps of how this technique works:

- **Steps** - the steps and the formulas are adapted from (Smilkov et al. 2017)

1. Create noisy versions of the image

It creates many slightly different versions of the image by adding random noise

2. Pass each image through the model

Each of these created noisy images is passed through the model for predictions.

3. Collect gradients from the noisy images

Each collected gradient gives information about how important each pixel influencing the model's prediction, by using the formula below.

$$M_c(x_i) = \frac{\partial S_c(x_i)}{\partial x_i}$$

4. Average the gradients

All of these collected gradients are averaged, by using the formula below. This process highlights the pixels whose average gradient is consistently important across all noisy images.

$$\hat{M}_c(x) = \frac{1}{n} \sum_{i=1}^{n} M_c(x_i)$$

4 Datasets

The Vision Transformer (ViT) that I chose to work with was pre-trained on the combination of the following datasets: AffectNet, FER2013, and MMI. These datasets contribute to the model's representation of facial expression recognition on humans. Since the model was trained on facial expressions of adults, I had to fine-tune it on infants by using the City Infant Faces Database. Using a model pre-trained on adult faces is helpful because this way the model has a

strong starting point for recognizing human-like faces. While adult and infant faces may have differences in their faces in aspects like textures and proportions, they still share a lot of common facial features, such as eyes, mouth, and facial shapes. These shared aspects help the model to transfer its knowledge from adult faces to infants, making fine-tuning effective.

- **City Infant Faces Database**

City Infant Faces Database is one of the very few data sets of infant faces. The goal of collecting these images was to support future research, as it can be a useful and valid tool, since there are many data sets of adult faces, but very few of infants (Webb, Ayers, and Endress 2018). The infants' images were collected by asking parents on social media, such as Facebook, if they were willing to send at least three images of their baby's face (0-12 months of age) showing the following affects: positive, neutral, and negative affect. The parents were asked to take photographs at the same time of the day, however, that was unsuccessful. They collected a total of 255 images, which were validated by 71 people, primarily student midwives (41), nurses (12), and the general public (18), with only six of these participants being male, with a mean age of 28 years. The images were assessed twice, to see if the perception of the images would change, on six dimensions:
- The expression is negative, neutral, or positive (Webb, Ayers, and Endress 2018).

- The Intensity, how strongly the participants felt about that expression on the scale of 1 to 5 (Webb, Ayers, and Endress 2018).

- The Genuineness, how genuine the participant felt the emotion on the scale of 1 to 5 (Webb, Ayers, and Endress 2018).

- The affective response- what emotion (positive, neutral, negative) the participant felt while looking at the image (Webb, Ayers, and Endress 2018).

- Strength of affective response- how strongly the participant felt that emotion on the scale of 1 to 5 (Webb, Ayers, and Endress 2018).

Based on the average scores, the images were categorized as positive, neutral, and negative. Only those having an agreement above 75% were kept, which resulted in 154 portrait images in total- 60 positive, 54 negative, and 40 neutral images.
All of the images were saved both in colour and black-and-white, however, the colour images were neither fully validated nor resized or normalized (Webb, Ayers, and Endress 2018). The Figure below shows a subset of the colour images and their labels.
One of the limitations of this dataset is that people rated the neutral images as being the most ambiguous, which led to the highest shift in perception and the

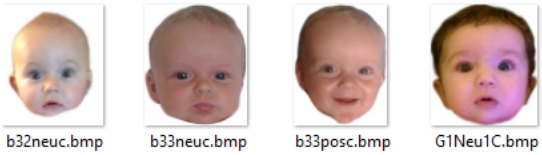

Figure 2: Example images from the City Infant Faces Database.

lowest level of agreement. Furthermore, they tried to gather images of infants from different ethnicities, but that was unsuccessful, which led to the majority of the infants being Caucasian (Webb, Ayers, and Endress 2018).

Expression	**Number**
Positive	58
Negative	52
Neutral	40

Table 2: Number of Infant Images per Affect Class Used for Fine-Tuning

- **AffectNet**

AffectNet is the largest facial expression dataset, consisting of more than 1,000,000 facial images from the Internet (Mollahosseini, Hasani, and Mahoor 2017). The idea behind the dataset came from a professor and a Phd student by searching for 1250 emotion-related keywords in six languages, using search engines such as Google, Bing, and Yahoo. Then these images were classified and labeled by humans, who had three training sessions, into the following categories: Neutral, Happy, Sad, Surprise, Fear, Disgust, Anger, Contempt, None, Uncertain, and Non-Face. The Non-Face category includes images that might contain the following: 1) No face in the image, 2) the image contains a watermark, 3) the face is a drawing, animation, or painted, and 4) the face does not have a normal or natural shape. The annotators include 12 full-time and some part-time workers at the University of Denver. One limitation of the data set is that each image was annotated by only one annotator. The data set was classified not only into the mentioned categories, but also each image was annotated with continuous values, such as valence, how positive or negative an emotion is, and arousal, how active or passive an emotion is. They have built a custom-made software that helped the annotators to label the images into both categories: categorical and continuous (Mollahosseini, Hasani, and Mahoor 2017). The distribution of the labels of these images can be seen on Table 3 below. The estimated average age of the faces on the images is 33.01 years. The percentage of correct matches between the queried emotion and the annotated result was the happy emotion with 48%, whereas the negative emotions had the lowest hit rate, around 2.7%.

Only a subset of the images, a total of 36,000, were annotated by two people, and their agreement was measured. The result of it shows that the annotators agreed the most on the happy images with 79.6% and the non-face images with 83.9%. The categories with the lowest agreement were mainly the negative images and the uncertain ones. In total, they agreed on only 60.7% of the whole subset of the dataset (Mollahosseini, Hasani, and Mahoor 2017).

Expression	**Number**
Neutral	80276
Happy	146198
Sad	29487
Surprise	16288
Fear	8191
Disgust	5264
Anger	28130
Contempt	5135
None	35322
Uncertain	13163
Non-Face	88895

Table 3: Number of Annotated Images in Each Category, Table 3 from (Mollahosseini, Hasani, and Mahoor 2017) paper

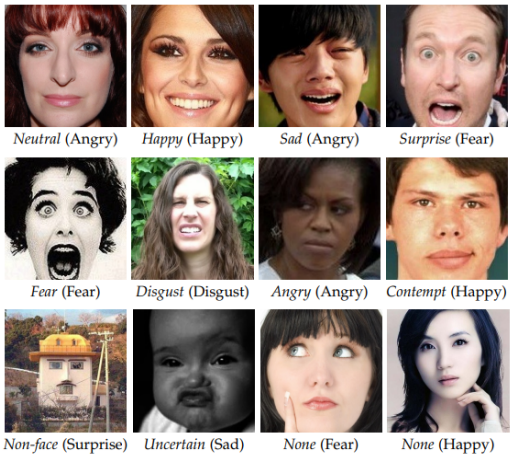

Figure 3: Example images from the AffectNet Database, Figure 3 from (Mollahosseini, Hasani, and Mahoor 2017) paper

- **FER2013**

FER 2013 is on the biggest gray-scale facial data set publicly available on the web (Mollahosseini, Hasani, and Mahoor 2017). It was created by querying 184 keywords related to human emotions, searching through Google. It contains over 35,887 images, categorized into 7 categories: anger, neutral, disgust, fear, happiness, sadness, and surprise. People agree only 65-68% of the time with the correct labels (Khaireddin and Chen n.d.). This dataset has several limitations, such as: 1) the images have low resolution (48x48 pixels), making it hard to detect facial details such as cheek movement; 2) the faces are unaligned, meaning that the faces are in different angles or positions. Due to these limitations, facial landmark annotations do not work well on these images (Mollahosseini, Hasani, and Mahoor 2017).

Figure 4: Example images from the FER2013 Database, Figure 5 from (Khaireddin and Chen n.d.) paper

- **MMI Facial Expression Database**

The MMI Facial Expression Database is a web-based, easily accessed application, containing a collection of over 1500 images, of both static (740) and image sequence (848) images (Pantic et al. 2005). This dataset was created to solve or improve the following issues: most facial expression datasets are not easily accessible, most include only static images or only videos, and most are missing profile views (Pantic et al. 2005). The images include both frontal and profile view of human faces of 19 male and female participants. This dataset is one of the most comprehensive sets because the image sequences enable analysis of the facial muscle activation, and it's ease for searching. All the images are true colour(24 bit) with 720x576 pixels. The data set includes frontal and profile views of a face, recorded with the help of a mirror. The length of the image sequences varies between 40 and 520 frames, including at least one or more neutral facial expressions. One-fourth of the images used have natural lighting,g and different backgrounds were used (Pantic et al. 2005).

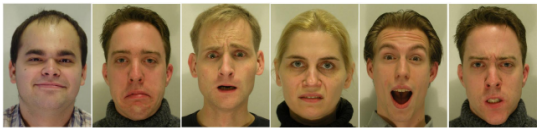

Figure 5: Example images from the MMI Database, Figure 1 from Pantic et al. (2005) paper

5 Experiments

5.1 Dataset choices

For this project, a total of 150 color images were used. The choice was made due to two factors:
1. The Vision Transformer I have decided to use was mainly trained on color images using the AffectNet and FER2013 data sets.
2. The Vision Transformers are more likely to perform better on color images due to the richer features (RGB) (Bissoonauth-Daiboo et al. 2023). The categories' split across these 150 images can be seen below in Table 2. The data was randomly split between train, development, and test data. The exact distribution of affect labels across the splits can be seen in Table 4. To prevent the model from data leakage, I had to make sure that the same infant face is in the same split. This had prevented the model from memorizing a face that could appear, for example, in both train and test splits. This could lead to unrealistically high performance of the model.

Label	**Train**	**Dev**	**Test**
Positive (Pos)	37	10	11
Negative (Neg)	34	10	8
Neutral (Neu)	29	5	6
Total	100	25	25

Table 4: Distribution of Affect Labels Across Train, Dev, and Test Sets

5.2 Model Selection

In this project, I decided to use a pre-trained Vision Transformer (ViT), trained on three different adult facial expression datasets. I have chosen this approach because of the following reason: even though adult faces may differ from infant faces, the pretrained model has already learned facial features such as the contours and the shape of a human face. This allowed the model to transfer the facial knowledge and adapt it to infants' expressions. This strategy also prevented the model from overfitting, since the City Infant Faces Database is quite small.

One key reason for that choice is that the City Infant Faces Database is quite small, so using a model whose weights are already learned on visual facial patterns will improve the performance of the model. Secondly, the process of learning in Vision Transformers (ViT) is different compared to CNNs. Both models do not rely on predefined features, but they extract patterns directly from the data, however, they do not have the same architecture. ViTs are learning through patches of the image, so it takes into consideration all of the pixels, and thanks to the self-attention layer, it focuses only on the most relevant patches. According to this paper (Rodrigo, Cuevas, and García 2024) ViTs perform better compared to CNNs in terms of accuracy and in robustness. This means that ViTs will be more effective regarding significantly different data, such as the adult faces data and the infant faces.
After I had chosen to work with a Vision Transformer, I found that the Hugging Face platform provides excellent resources about how to use such models. My aim was to find a model that was trained or fine-tuned on the AffectNet dataset, as it is the largest available dataset on adult facial expressions. I came across three different models, where the AffectNet dataset was used. The architecture on two of them is based on Vision Transformers (ViT), and the other is based on Bidirectional Encoder representation from Image Transformers (BEiT).
I chose to work with the one that had the highest accuracy (84.3 %) and the lowest loss (0.4503) among them all. It was also trained additionally on two other adult facial expression datasets mentioned above in the Datasets section, which I believe will contribute to a better performance. Furthermore, it uses the standard ViT-Base-Patch16-224-in21k model, one of the most up-to-date making it easier to work with because the library supports that version. The comparison between these models can be seen in Table 5.

Criteria	**motheecreator**	3una	Mauregato
Accuracy	84.3%	73.4%	67.1%
Loss	0.4503	0.8122	0.9712
Datasets used	AffectNet + FER2013 + MMI	AffectNet	AffectNet
Training epochs	3	100	22
Transformers version	4.36.0	4.35.2	4.29.0
Pretrained base model	ViT-Base-Patch16-224-in21k	BEiT	ViT-Base-Patch16-224

Table 5: Comparison of models fine-tuned on AffectNet

5.3 ViT Fine Tuning

- **Data pre-processing**

The very first step was to open both the colour and the black and white images, simply for data exploration, and check the label distribution across both sets. The whole focus was on the colour images. I have manually split them into train, development, and test data, by making sure that the same infant face is in the same folder.

The next step was to find out the type of data structure that the model expects. I used the `load_dataset` built-in function to load the image sets separately, since I kept them in different folders. This function returned the data in the expected format for the model to be able to process- a dictionary containing two keys: the image and the label for it. The original dataset did not include labels in a separate file. The labels were part of the file names, therefore, they were missing in the mentioned dictionary. I created an extraction function that took the labels from the file names and mapped them back to the dictionary based on the images' names.

The model can read only digits from pixel values of each image, therefore, each image had to be pre-processed by the AutoImageProcessor. That is a class from the Hugging Face, whose main function is to resize all the images into 224 x 224 pixels, to normalize the pixel values, and convert these values into tensors in the right format. To check the distribution of the normalized pixel values from the tensors, I plotted these values for the first ten images from the training set. The most dominant value is 1, corresponding to the white background. The result can be seen in Appendix B. By doing so, the images at that moment were converted into tensors, but the labels were missing. I used a Pythorch Dataloader that can handle both the preprocessed tensors together with the labels from the `load_dataset` function. This allowed me to prepare the data as input for the model.

- **Model fine-tuning**

When I made sure that the data had the correct shape, I have loaded the model with the pre-trained weights. The model consists of three main parts: the embeddings, the encoder, and the classification layer. The embedding layer is responsible for splitting the tensors for each image into 16x16 patches, and then it converts them into a 768-dimensional feature vector per patch. The encoder in this model has 12 transformer blocks, responsible for the model to learn the relationships between parts of the images, through the attention layer. Since that model was fine-tuned on the AffectNet, which has seven categories, I had to change the classification head of the transformer to be three, so it matches the labels of the City Infant Faces Database. I decided to use the "CrossEntropyLoss" loss function, because it works on cases where the labels are several, like in this case. For an optimizer, I chose to use AdamW. It is the most used and recommended optimizer when working with Transformers, according to the (Hugging Face 2024).
The whole training process consisted of five epochs, and the data was split into batches of 32 images. During each batch, the model performs a forward pass and it calculates how far it is from the true label, using the loss function, and adjusts its weights, using the mentioned optimizer. To track the model's performance, the accuracy and the loss were tracked for each epoch. Then the weights of the model were saved for testing purposes afterward.

5.4 SmoothGrad

In this project, for each image, there were created 50 noisy versions and the values added to the pixels were drawn from a normal distribution. Then the average of all 50 gradients is calculated. These values are used for creating a heatmap, where the higher the average, the higher the influence. I used a color map to show that relation. The heatmap was overplayed with the original images, which showed the most influential pixels of a model's prediction.

6 Results

6.1 Model Evaluation

After training the chosen Vision Transformer on the City Infant Faces data set, I evaluated its performance on using both datasets: the development and the test set, using the accuracy. The model performed quite well on the development set, by misclassifying only two out of 25 total images, which results in 92% accuracy. In both cases, the model predicted a neutral label when the true labels were either positive or negative, as shown in Figure 6

Criteria	**Development Set**	**Test Set**
Total Images	25	25
Misclassified Images	2	4
Misclassification Rate	8.00%	16.00%
Accuracy	92.00%	84.00%

Table 6: Performance comparison on the development and test datasets

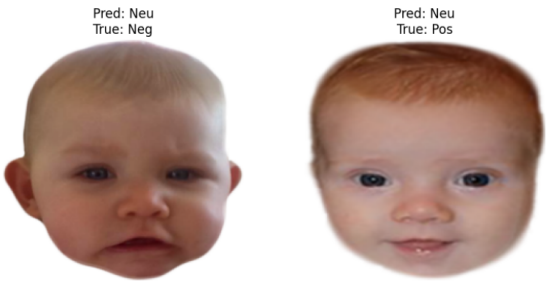

Figure 6: Misclassified images from the development set

After testing the model on the development data set, I tested it on the test set. The performance slightly dropped to 84%. The model struggled to classify four out of 25 images. Three images were predicted as neutral and one as positive, whereas all of them were actually negative, as shown in Figure 7

Overall, the model performed well on both sets, even though the performance dropped on the test set by 8%. The result is shown in the Table 6. This drop can suggest that the model is probably overfitted to the development data. This means that the model struggles to generalize unseen data. In the test data sets, the misclassification is mainly on the negative images. This means that the model struggles the most with generalizing the negative images, and it classifies them as neutral. This behaviour could be understandable, because even for a human eye it might be challenging to catch subtle sights of negative affects such as shown in the first image in Figure 7 and the third image in the Figure 7 below. The misclassification in the second image of the test data could be due to the fringe (bangs) since it is hiding the infant's forehead and eyebrows. The model is not trained to recognize the affect in such cases, since that infant is the only one having bangs out of all in the dataset.

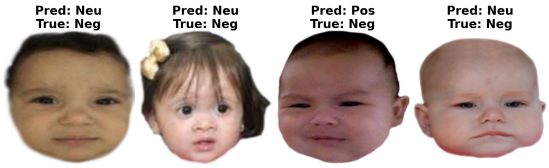

Figure 7: Misclassified images from the test set

6.2 SmoothGard heatmaps

The SmoothGard heatmaps show the most influential parts of the images- those that contribute the most to the model to make its prediction, and the result is shown in Appendix C.

- **Positive Images**

According to Zaharieva et al. (2024), the positive expressions of infants are associated with two main indicators, AU12 (lip corner raiser) and AU6 (cheek raiser). The SmoothGrad heatmaps support this. The heatmaps highlight the apple of the cheeks and the lip corners, which are the key factors in infant smiles. The result can be seen in Figure 8 below.

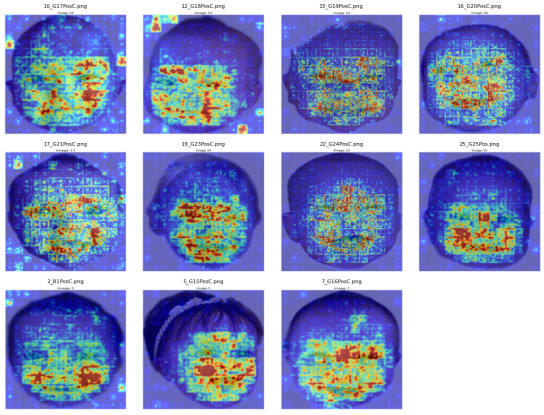

Figure 8: SmoothGrad Heatmaps for Positive Images

- **Neutral Images**

The study from Zaharieva et al. (2024) does not include any dominant features related to the neutral facial expressions. Based on the results from the heatmap, the main facial features for an image to be neutral are the T zone of an infant's face, which includes: forehead, nose bridge, and mouth area. These results can explain why the model tends to classify most of the images as neutral more often than other classes. Figure 9 below.

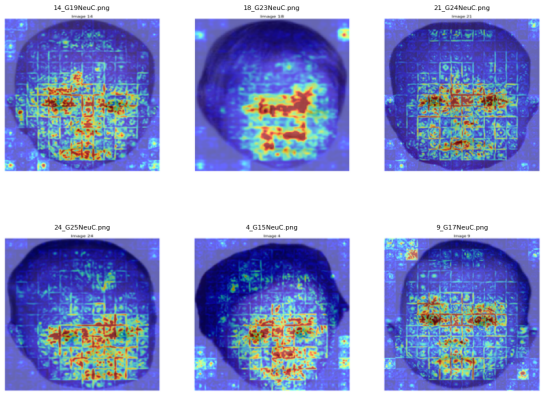

Figure 9: SmoothGrad Heatmaps for Neutral Images

- **Negative Images**

According to Zaharieva et al. (2024), the negative expressions of infants are

associated with several facial expressions: primarily characterized by AU20, AU25/25/27 (lip stretching, crying), which is AU17 (chin; pouting), AU6/7 (eye constriction from intense cry). The underlined files' names are all the negative images that were misclassified. When we exclude them, we can see that the parts contributing the most to the model's prediction are the following areas: pouting, stretched lips, and the corner of the mouth. These heatmaps support the AU-based framework from this paper: (Zaharieva et al. 2024) for negative facial expression in infants.Figure 10 below.

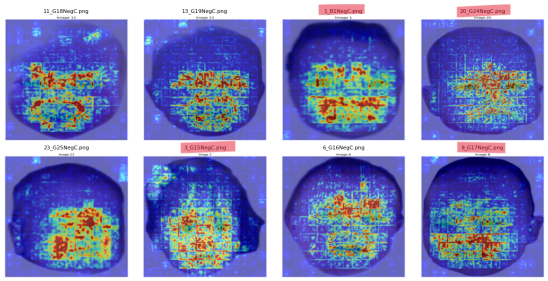

Figure 10: SmoothGrad Heatmaps for Negative Images

7 Discussion

7.1 Interpretation of Results

The Vision Transformer was tested on both datasets: the development and the test data. It performed well on both, but it performed slightly better on the development set. This means that the model's predictions are less reliable on unseen data. The model is biased towards the neutral class because it misclassifies the images as neutral. This could mean that the model struggles to detect the images, especially the ones that are ambiguous.

7.2 Model Bias

The model in general tends to misclassify only one main class, the negative class. That could be due to several reasons: it is hard to detect the subtle sighs of unhappiness even for a human eye; additionally, the negative class in the AffectNet data set is relatively smaller compared to the rest of the classes, which means that the model was not exposed to enough samples of negative images. In addition to that, the annotators labeling the images from the AffectNet dataset had more difficulty classifying the negative facial expressions. Together with the smaller size of negative samples, the labeling challenges can cause the model to misclassify the negative class.

7.3 Limitations

The biggest limitation in this project is the size of the dataset. It is quite small, which limits the model's performance, because it is hard to capture the full spectrum of each class. Also, the images lack diversity in terms of lighting, the position of the face, and infant ethnicities.
Another limitation of this study is the lack of pre-trained models specifically on infants' facial datasets. This limitation makes the task more challenging as it requires relying on less specialized models, which do not capture the unique features of an infant's face.

7.4 Future Work

To make the results more reliable and meaningful, there are several ways to build upon. The following improvements could lead to a better understanding of the model's behavior, strengths, and weaknesses:

- **Heatmap Analysis on more images**

In order to better understand the reason for misclassifying the negative images with neutral ones, an extended analysis by using SmoothGrad heatmaps could be done on the development test as well. By trying to get patterns for all the images on which the model was tested.

- **Train on black and white images**

Given the small size of the colorful images, the model might struggle to get patterns among the different labels. Therefore, having the model trained on more images is a great idea. As part of the City Infant Database, there is a greater amount of black and white images compared to the color ones. A larger dataset can increase the performance of the model by making the learned patterns more consistent.

8 Conclusion

To sum up, this project explored the use of a Vision Transformer for infants' facial expressions by using the instructions of the Hugging Face library. The model was chosen among three other models, all of them trained on the AffectNet- the largest adult facial expression dataset. The chosen model has the highest accuracy, the smallest loss, and was trained on several other facial datasets, together with the AffectNet dataset. The Vision Transformer was fine-tuned on the City Infant Database set, by using only the colour images. The model was tested on two datasets: the development and the test dataset, both containing 25 images each. The performance dropped slightly when tested on the test set by 8%. The model struggles to identify negative images as it misclassifies them as neutral.

To deeper analyze the decision of the model when predicting, a SmoothGrad technique was used together with heatmaps. For the positive images, the models seem to detect the expected activation of the action units, such as the lip corners and the cheek raiser. However, for the neutral images, the main facial features on the heatmap are the T zone of the infant. For the negative images, there are many more factors related to negative expressions such as lip stretching, chin, pouting, and eye construction. The model seems to catch them on a few images, however, that is still the most misclassified category. Further research needs to be done, such as using more images for both testing and training. To better understand why the model tends to predict neutral class and misclassifies the negative images, further analysis needs to be done.

9 Acknowledments

I would like to thank my supervisor, Martin Lund Trinhammer, for the continuous support and help during this project. His feedback and guidance contributed a lot for the outcome of this project.
I would also like to acknowledge Stella Grasshof for being part of the supervisory team.

10 Appendix

Appendix A: Model Links

In this project the following models were explored and can be found on the links below. All of them are publicly available at the Hugging Face webpage.

- **motheecreator**: Hugging Face page
- **3una**: Hugging Face page
- **Mauregato**: Hugging Face page

Appendix B: Normalized Pixels Distribution for the first ten images from the training set

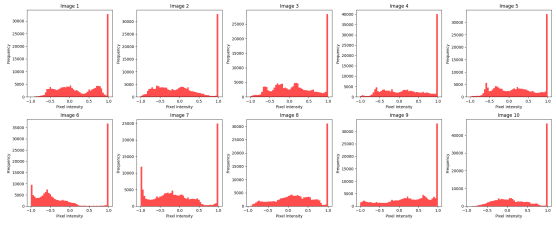

Figure 11: Pixel Values Distribution

Appendix C: SmoothGrad heatmaps

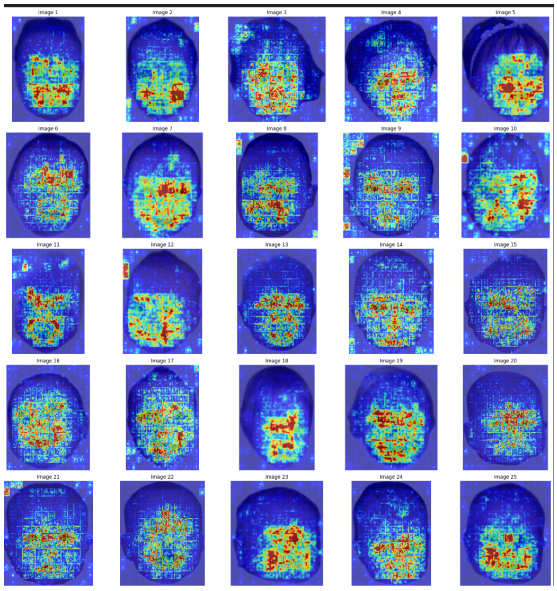

Figure 12: SmoothGrad heatmaps

11 Reference

References

Bissoonauth-Daiboo, Preeti et al. (2023). "Endoscopic Image Classification using Vision Transformers". In: *Proceedings of the 2023 7th International Conference on Advances in Artificial Intelligence*, pp. 128–132.

Carpenter, Kimberly LH et al. (2021). "Digital behavioral phenotyping detects atypical pattern of facial expression in toddlers with autism". In: *Autism Research* 14.3, pp. 488–499.

Cheong, Jin Hyun et al. (2023). "Py-feat: Python facial expression analysis toolbox". In: *Affective Science* 4.4, pp. 781–796.

Correll, Mark (2023). *Building a Vision Transformer Model from Scratch*. Accessed: 2025-05-05. URL: `https://medium.com/correll-lab/building-a-vision-transformer-model-from-scratch-a3054f707cc6`.

Dosovitskiy, Alexey et al. (2020). "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929*.

Fatema, Umme (2021). "Infants' Facial Emotion Recognition". Master's thesis. University of Nevada, Las Vegas. URL: `https://digitalscholarship.unlv.edu/thesesdissertations/4141/`.

Hausmann, Jacqueline et al. (2022). "Robust neonatal face detection in real-world clinical settings". In: *arXiv preprint arXiv:2204.00655*.

Hinduja, Saurabh et al. (2023). "Pyafar: Python-based automated facial action recognition library for use in infants and adults". In: *2023 11th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, pp. 1–3.

Hugging Face (2024). *Efficient Training of Transformers*. Accessed: April 15, 2025. URL: `https://huggingface.co/docs/transformers/v4.42.0/perf_train_gpu_one`.

Khaireddin, Y and Z Chen (n.d.). "Facial emotion recognition: State of the art performance on FER2013. arXiv 2021". In: *arXiv preprint arXiv:2105.03588* ().

Mollahosseini, Ali, Behzad Hasani, and Mohammad H Mahoor (2017). "Affectnet: A database for facial expression, valence, and arousal computing in the wild". In: *IEEE Transactions on Affective Computing* 10.1, pp. 18–31.

Onal Ertugrul, Itir et al. (2023). "Infant AFAR: Automated facial action recognition in infants". In: *Behavior research methods* 55.3, pp. 1024–1035.

Pantic, Maja et al. (2005). "Web-based database for facial expression analysis". In: *2005 IEEE international conference on multimedia and Expo*. IEEE, 5–pp.

Rodrigo, Marcos, Carlos Cuevas, and Narciso García (2024). "Comprehensive comparison between vision transformers and convolutional neural networks for face recognition tasks". In: *Scientific reports* 14.1, p. 21392.

Smilkov, Daniel et al. (2017). "Smoothgrad: removing noise by adding noise". In: *arXiv preprint arXiv:1706.03825*.

Trigueros, Daniel Sáez, Li Meng, and Margaret Hartnett (2018). "Face recognition: From traditional to deep learning methods". In: *arXiv preprint arXiv:1811.00116*.

Vidhya, Analytics (2023). *Introduction to Vision Transformers (ViT)*. Accessed: 2025-05-05. URL: `https://www.analyticsvidhya.com/blog/2023/05/introduction-to-vision-transformers-vit/`.

Webb, Rebecca, Susan Ayers, and Ansgar Endress (2018). "The City Infant Faces Database: A validated set of infant facial expressions". In: *Behavior research methods* 50, pp. 151–159.

Zaharieva, Martina S et al. (2024). "Automated facial expression measurement in a longitudinal sample of 4-and 8-month-olds: Baby FaceReader 9 and manual coding of affective expressions". In: *Behavior research methods* 56.6, pp. 5709–5731.