Reconstruction of Emotion Using Transfer Learning on Visual Features

Mengtian Jin
Stanford University
mtjin@stanford.edu

Sherine Zhang
Stanford University
sherinez@stanford.edu

Kaylee Zhang
Stanford University
kayleez@stanford.edu

Abstract

Our project proposes a deep neural network model to predict emotion cognition based on visual features of human faces. We adpot transfer learning by implementing and experimenting various pre-trained models to extract generic features. We then use Transformer as an encoder layer and LSTM as a decoder layer, which captures the time-series features, to predict human emotions over time. We train and test our model on Stanford Emotional Narratives Dataset (SENDv1). After the experiments, we find that using VGG-pretrained model on ImageNet dataset, Transformer encoder layer and LSTM decoder layer generates the best average CCC score, 0.175, and the best CCC score on a single video, 0.955.

1. Introduction

As we are approaching into the age of Artificial Intelligence (AI), we are emphasizing more and more on Human-centered AI. How can AI better serve us human-beings? This project starts from this point of view, and mainly focuses on the aspect of modelling emotional cognition. Understanding and reconstructing human emotions is one important element of Human-centered AI because it improves the interaction between humans and AIs. For example, if robots at your home can learn your spontaneous emotions accurately, they can not only serve your needs, but also in a more user-friendly way, just like how your best friend treats you.

We are interested in modeling emotion cognition and using the model to predict human emotion valence based solely on images that capture human faces. There have been many recent works on emotion recognition via machine learning approaches, and the most common research focuses include recognizing emotions from facial expressions, linguistics, body gestures and the combination of these different modalities. Our project models emotional cognition on video frames of human faces by utilizing deep learning techniques. The deep network model consists of embedding layers that convert each image to an embedding vector, encoding layers and decoding layers to output valence ratings. Pre-trained models such as VGGNet and ResNet on large dataset are utilized to improve the model performance. Face detection and facial landmark extraction algorithms are added to obtain better feature vectors.

The dataset we use contains short interview videos, from which we extract two images per frame at a rate of 30 fps and use the raw pixel values of the images as input to our model. We then feed the images to Transformer and LSTM in order to get the final rating outputs. The baseline model takes only the raw images as inputs. We also experiment with a few more architectures which involve feature extraction using pre-trained models. Instead of the raw pixels, we input the feature vectors obtained from pre-trained models (sizes of the vectors vary on the model used) to the encoder-decoder model. The model outputs an array of predictions of emotion valence for each video. The predicted ratings range from 0 to 1 with 0 being most negative and 1 being most positive. We obtain one rating in each 0.5 second. CCC are used to evaluate the accuracy of our prediction.

2. Related Works

A recent work on video-based emotion recognition was done by Kaya et al [4]. They introduced a multimodal approach for video-based emotion recognition based on the data provided by the Emotion Recognition in the Wild (EmotiW) Challenge. Their work extended the framework [3] of the combination of multiple visual features with audio over summarizing functionals by employing deep convolutional neural network (CNN) features and investigating appropriate transfer learning strategies. The overall pipeline starts with the detection of the face and face alignment. The face images then go through two parallel channels. One channel applies image purification followed by dense feature extraction using Scale Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG), Local Phase Quantization (LPQ), Local Binary Patterns (LBP) and its Gabor extension (LGBP). The other channel exploits a pre-trained VGG-Face model followed by a multi-stage fine-tuning on FER 2013 dataset. Finally, the two parallel channels with different learned features and the audio fea-

tures from the original dataset are used as inputs for kernel extreme learning machine (ELM) and partial least squares (PLS), which serve as classification models to generate predictions. In their work, incorporating transfer learning and using VGG-Face model solve the potential issue of CNNs that it requires very large amounts of data to avoid overfitting.

Another paper by Ng et al [7] presented a two-stage transfer learning technique for emotion classification on small datasets. They started with pre-trained CNN architectures(AlexNet and vgg) on ImageNet. Considering the limitation of data size, they firstly used related generic face expression datasets (FER28, FER32) to fine-tune the CNNs, followed by their own dataset (EmotiW). One experiment included both the FER and the EmotiW as training data, with only the EmotiW validation data.

Yu et al [10] proposed methods on face detection by ensemble of three face detectors, the joint cascade detection and alignment (JDA) detector, the Deep-CNN-based (DCNN) detector and Mixtures of Trees (MoT). Mot was used as the last step where both JDA and DCNN fail. Besides using pre-trained models, they also utilized two loss criterion, the log-likelihood loss and hinge loss to learn the ensemble weights of multiple CNN models.

From the work of Kaya et al. and Ng et al., we learned that the integration of pre-trained model and variation of transfer learning played a key role for visual feature extraction in the pipeline and could improve the accuracy of classification. Therefore, we integrated the idea of transfer learning: implementing pre-trained models into our pipeline to extract generic features because of the relatively small dataset we have. This will be discussed in more details in the method section. Inspired by the idea of using dense feature extraction using SIFT, HOG, and LPQ to better learn the facial expressions such as the contours of the mouth or eyes over time, and then fitting these extra features into the training model by Kaya et al., we also used a very different pre-trained model, which focuses on detecting contours of facial features, to learn facial expression features.

3. Dataset and Features

The dataset we used for this project was Stanford Emotional Narratives Dataset (SENDv1)[1]. It contained 193 videos of 49 different participants describing their personal emotional stories. Each clip on average lasted 2 minutes and 15 seconds. The data set was split into 60% Train (117 videos), 20% Validation (38 videos) and 20% Test (38 videos). Most importantly, to test the ability of the models' generalization, 5 participants only appeared in the Validation set, and 6 only appeared in the Test set. In the later experiment, each video was watched by at least 20 independent observers and the observers provided ratings (ranging from 0 to 100) of how a narrator in the video felt in a continuous time frame.

Videos have a frame rate of 30. We extracted 2 images per second to capture the change of facial expressions of the target while keep the data to a reasonable size. Exact number of images per video depends on the video length, but approximately there are 240 images per video. For each extracted image, we used Cascade Classifier in OpenCV to crop out faces of the target, in order to clear out the non-informative background as much as possible. For the baseline model, we resized the images to 50 x 50 pixels and grayscaled them so that the resulting flattened vector is of size 2500. For pre-trained models, the size of images are adjusted accordingly. For example, VGG16 and ResNet152 take as input a $224 \times 224 \times 3$ image. We also normalize the pixel values of each image by subtracting the mean value and dividing by the standard deviation obtained from the entire training dataset.

The ratings were sampled every 0.5 second to match the extracted images, and then scaled down to a range of 0 to 1. To predict the continuous emotional ratings, we used the average observer rating as the ground truth.

4. Methods

We introduce different methods and architectures that we experimented in this section. As emotion ratings are continuous numerical values, we formulated our task as a regression problem to predict the emotional valence. Our network architectures are all consisted of three fundamental parts, namely, an embedding layer, an encoder and a decoder. We mainly focus on transfer learning with different feature embedding designs based on pre-trained models and feature extraction techniques. 1 briefly introduced the whole architecture.

4.1. Embedding Layer

The baseline model directly used OpenCV package to resize and center the raw data from SENDv1 dataset. It was then fed into a fully connected layer as inputs to the neural network model. To improve the baseline model, we adopted transfer learning, which is the learning that applies the already learned knowledge from a related task to a new task. We used VGG [8] and ResNet [2], which are pre-trained CNN architectures on ImageNet and VGGFace2[1] dataset to first extract generic features.

4.1.1 VGG Final Layer and Feature Layer

VGGNet is a convolutional neural network architecture that exploits small filter sizes and deep network structure to achieve a good classification result. Two best performing

[1]This dataset is not released.
Contact $desmond\_ong@ihpc.astar.edu.sg$ for the dataset.

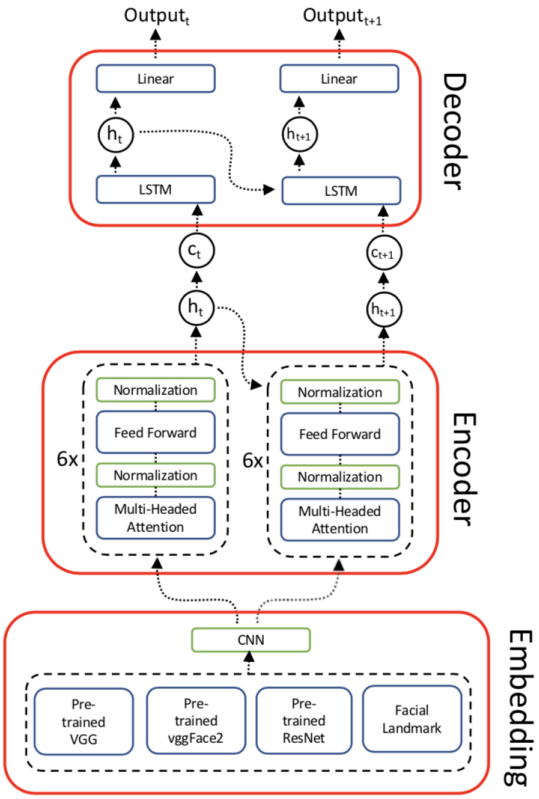

Figure 1. General Model Architecture

models, namely VGG16 and VGG19, are released. For the purpose of this project, we used VGG16 pre-trained model. The model contains 13 convolutional layers, followed by 3 fully connected layers, and output a vector of size 1000 to represent the image classes. Considering vgg-1000 is mainly for image classification instead of facial feature extraction, we also extracted features from the final intermediate fully connected layer called FC1, which might capture more complex structures that better fit our needs. FC1 layer gives a feature embedding size of 4096.

4.1.2 ResNet Final Layer and Feature Layer

ResNet is a 152-layer deep network using residual connections. It uses network layers to fit a residual mapping instead of a underlying mapping directly. Stacking residual blocks allows the very deep network to train without degrading. ResNet achieves highest accuracy with moderate efficiency on ImageNet, thus we believe that the use of pretrained ResNet at the beginning stage of our complete model can improve the performance. We utilized ResNet as the third model. Similarly to vgg-fc1, we also extracted the feature vector from the last second layer of ResNet as our feature embedding. It gives a feature embedding of size 2048.

4.1.3 VGGFace2

Inspired by the work of Kaya et al. [4] where they used a pre-trained VGG-Face model for feature extractions instead of on ImageNet, we implemented a pre-trained model on VGGFace2 [1]. ImageNet contains various classes of objects, which gives a good generality of images, however, because all of our images are human faces and we are more interested in predicting human emotions from faces, we believe that a pre-trained model on VGGFace2 could give a better features. Cao et al. [1] introduced a new large-scale face dataset named VGGFace2. This dataset has a large number of identities and also a large number of images for each identity. It covers a large range of pose, age and ethnicity. Therefore, we adopted their pre-trained SE-ResNet-50-256D on VGGFace2 to extract features that will be more specific to human faces. This model generated an embedding size of 256.

4.1.4 Facial Landmarks

Since pre-trained models mainly focus on object variations instead of facial expressions, they could fail to accurately capture emotion valence from facial images. We explored another method, the pre-trained facial landmark detector inside dlib library [5]. Facial landmarks are important facial structures in the face region: mouth, left and right eyebrow, left and right eye, nose, jaw, face edges, etc. The dlib detector we used estimates the location of 68 (x, y)-coordinates that map to facial structures. The 68 coordinates can outline the shape of facial landmarks. We then flatten 68 (x, y)-coordinates into a 1-d vector of size 136 and use it as the feature vector for each image.

When building feature vectors, we encountered that dlib algorithm failed to detect human face or return coordinates of facial landmarks. In this case, we used moving average with window size equals 5 seconds to impute the missing feature embedding. Before feeding into the encoding layer, we normalized coordinates by shifting every points so that the first coordinate of each image was always (0, 0). By doing so we removed the confounding effects of position variation of different human faces.

After using the Facial Landmarks to detect different parts in face, we did a visualization on the contour of each part. The visualization is shown in Figure 2. By looking at the visualization, we noticed that mouth is the key feature to reflect human emotions. The contour of mouth changes most when a person is happy or unhappy. Therefore, we also experimented the model with only mouth features extracted from Facial Landmarks as our embedding features. This made the input features to a small size and we can train the model relatively fast, but not lose too much key information.

Figure 2. Original Images and their Facial Landmarks

4.2. Encoding Layer

Since video lengths are different and our images were captured at 0.5-second time scale, we firstly appended zero vectors at the end to match the largest video length, which is the number of images captured per video. Prior to encoding layer, embedding vectors firstly went through 1D CNN layer so that each image is represented by a vector of size 256 and each video contains the same lengths of vectors. Our training batch is divided based on videos, thus appending videos to be the same length is necessary for model architecture.

Our encoding layer utilizes Transformer model. As the video were recorded by targets talking about their stories, images within a video are in fact highly connected over time and Transformer can capture such strong correlation based on its self-attention scores. We used a linear normalization with a dropout rate of 0.1 layer as the connection layer between sub encoder layers.

4.3. Decoding Layer

We used Long Short-Term Memory Networks (LSTM) as the decoder that takes the Transformer encoder output and hidden states output to predict the valence ratings that range from 0 to 1 indicating negative to positive emotions.

The sentiment analysis architecture in this project is essentially a one-to-one model. We firstly utilize pre-trained models to construct image embedding together with a 1-D CNN layer. Taking account of the connectivity of input images, Transformer is implemented as the encoder to achieve one-to-many procedure and vanilla LSTM performs as the decoder for many-to-one task. The deep learning architecture can take a single image as input and predict its valence score.

4.4. Evaluation

The optimization goal is to minimize the mean squared error of actual ratings and predicted ratings (i.e., $\text{MSE}(\hat{Y}_{1:T}, Y_{1:T}) = \sum_{t=1}^{T}(\hat{Y}_t - Y_t)^2$).

We used the Concordance Correlation Coefficient (CCC [6]) as our evaluation metrics. CCC measures the similarities of two time-dependent curves. The CCC for two time-series vectors X and Y is calculated as:

$$\begin{aligned} \text{CCC}_{XY} &\equiv 1 - \frac{E\left[(X-Y)^2\right]}{E\left[(X-Y)^2\right]|_{\text{setting } \rho_{XY}=0}} \\ &= \frac{2\rho_{XY}\sigma_X\sigma_Y}{\sigma_X^2 + \sigma_Y^2 + (\mu_X - \mu_Y)^2} \end{aligned} \tag{1}$$

where $\rho_{XY} \equiv \text{cov}(X,Y)/(\sigma_X\sigma_Y)$ is the correlation coefficient. μ and σ represents the average and standard deviation respectively. CCC measures agreement between two curves, where 1 means that the two time-series are perfectly correlated and 0 means that they are uncorrelated.

5. Results

5.1. Experiments and Results

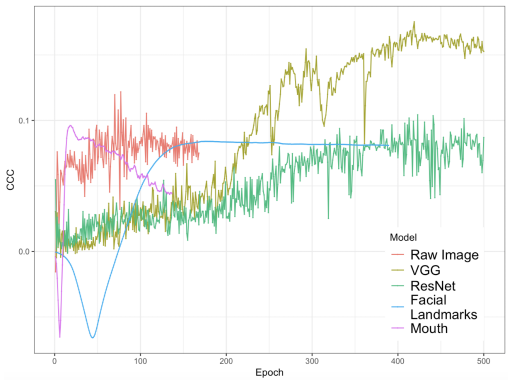

Figure 3. Plot of CCC score over number of epochs for five of the models. VGG-Feature and ResNet-Feature were not included since results were similar to VGG and ResNet.

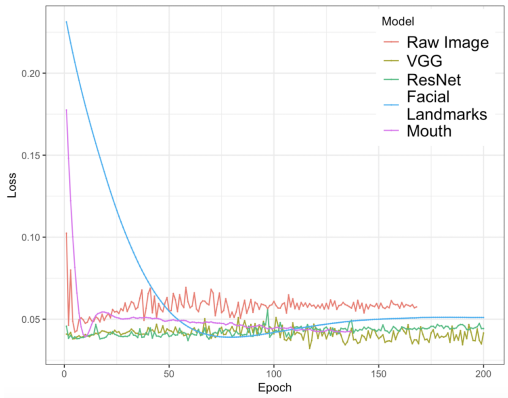

Figure 4. Plot of evaluation loss value over number of epochs for five of the models. VGG-Feature and ResNet-Feature were not included for the same reason as above.

Shown in Figure 3 and Figure 4, we plot CCC scores and MSE loss values against the epoch numbers for all models except VGG-Feature and ResNet-Feature. The results from these two models are very similar to that of VGG and

ResNet models. This is unsurprising since the Feature models' features are obtained from the fully connected layer directly above the final layer from where we extract inputs for VGG and ResNet and thus both layers share overlap information. For Raw-Image and Facial-Mouth models, we stop training after observing a stable state or a declining trend, and hence the early stopping appearance on both graphs. The evaluation loss either drops very quickly in below ten epochs, or starts with a low value and remains fairly static for the rest of epochs. Only Facial Landmarks Model shows a relatively smooth decreasing loss over time.

	Statistics		
	MSE Loss	CCC	Single Best CCC
Raw-Image	0.058	0.122	0.781
VGG	0.041	0.175	0.955
VGG-Feature	0.045	0.169	0.952
ResNet	0.049	0.104	0.953
ResNet-Feature	0.045	0.096	0.924
Facial-Landmarks	0.051	0.084	0.409
Facial-Mouth	0.042	0.096	0.452
VGGFace2	0.05	0.048	0.631

Figure 5. Result table listing the loss value, average CCC value of all videos, and the best CCC value for a single video for all models.

From Figure 5, we can clearly see that pre-trained VGG model on ImageNet gives the best prediction in terms of highest average CCC score, singel best CCC socre and loswet MSE Loss. VGG-Feature model has a comparable result as the VGG model, which has been explained above. ResNet and ResNet-Feature model do not have as good performance as the two VGG models and the Raw-Image model. This could be explained by the fact that ResNet trained on ImageNet achieves the best object classification accuracy, so out of the domain of object classification, the model does not generalize and apply to emotion prediction on visual features well. The Facial-Landmarks, Facial-Mouth and VGGFace2 are pre-trained models focusing on face data. Among these three models, Facial-Mouth generates the best result in terms of MSE Loss and average CCC score. This confirms that mouth is the most important feature to learn human emotions from faces, but some other features such as nose and face contour may be noises and deprecate the prediction. Facial-Mouth uses a very small dimension of feature embedding as generic feature input to our pipeline, so another advantage of this model is that it is relatively computationally cheap.

5.2. Hyperparameter

For the encoder-decoder model, we tuned some hyperparameters including the learning rate, the hidden layer dimension, and the feature embedding dimension after CNN. We used a validation set to fine tune our model, tried different combinations and recorded the best set of hyperparameters to use. After experimenting, We decided to use the same number of hidden layer neurons (256) and same output dimension from CNN (256) for every model. However, different learning rates were applied to different models. For example, Facial Landmark models produce a more stable output under a learning rate of 10^{-5}, but VGG-related models yield better result with a learning rate of 10^{-4}. We used Adam as our optimizer. Since we only have 117 videos in the training set, a minibatch size of 25 was used to keep convergence relatively speedy as well as have enough data in one batch to perform less random updates.

5.3. Feature Visualization

A good way to analyze the results we got from different models was to visualize the generic features we extracted from each pre-trained model. We provided two approaches for the visualizations on extracted features. The first approach was to use a heat map visualizing the features of images in the same video over time to evaluate how much facial expression changes could be represented by the features. The second approach was to plot correlations between the extracted features of images in two different videos to evaluate how much facial differences of two persons could be captured by the features.

Figure 6 showed two examples of the heat map visualization. The top left heat map represented the extracted features of all images within one specific video (we took video No.4-111 as an example) from pre-trained VGG model on ImageNet dataset. Each row was an image captured from the video, and each column was the extracted feature values, so the rows represented the images over a continuous time frame. The top right heat map represented the extracted features of all images within one specific video from pre-trained VGG model on VGGFace2 dataset, and the bottom left heat map represented the extracted features from pre-trained Facial Landmarks model. The rows and columns had the same meanings as before. We were interested in looking at the the color changes along rows within each column. The color changes along rows indicated the feature changes of the face along time. By comparing the three heat maps, we could see that there were more variations in colors along rows in the top left and bottom left heat maps, which were generated by pre-trained VGG model on ImageNet and pre-trained Facial Landmarks model. The variations along rows were almost subtle in the top right heat map. This could explain the better performance of using pre-trained VGG on ImageNet and pre-trained Facial Landmarks model because they captured some facial expression changes over time, but the pre-trained VGG on VGGFace2 could not recognize facial expression changes accurately and thus generating similar feature values over time.

We also generated a heat map of the pixel values in the

raw images, shown in bottom right plot in Figure 6. There were large amount of variations along rows, which indicated the facial expression changes over time of an individual in the video. However, the heat map of the pixel values in the raw images presented a lot of noises that would affect the prediction.

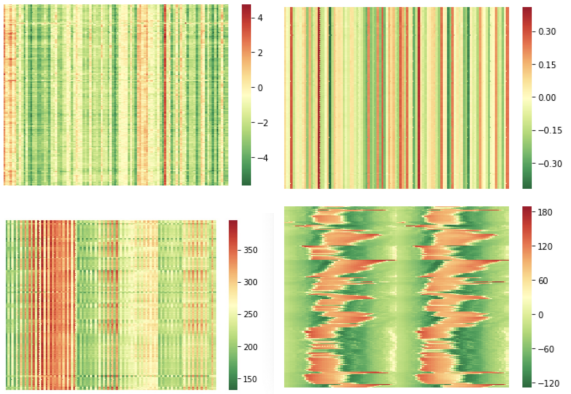

Figure 6. Heat map on extracted features, Top Left: extracted feature from pre-trained VGG model on ImageNet; Top Right: extracted features from pre-trained model on VGGFace2; Bottom Left: extracted features from pre-trained facial-landmarks model; Bottom Right: Heat map on pixel values of raw images

Figure 7 showed the correlation among feature embedding vectors of the first image extracted from 50 videos. The plot is symmetric and grid-(i ,j) indicated the value of correlation coefficients between the video i and video j. Lighter color corresponds to larger correlation. We found that in general, feature embeddings extracted from different videos are correlated as most correlation values fall into the range of 0.5 to 1 for pre-trained VGG model and facial landmark model. In particular, pre-trained vggFace2 model output almost same feature vectors cross different videos, and the correlations are above 0.9. The feature embedding from pre-trained VGG model gave the largest variation across different videos compared to vggFace2 and facial landmark models, which is also consistent with our model results.

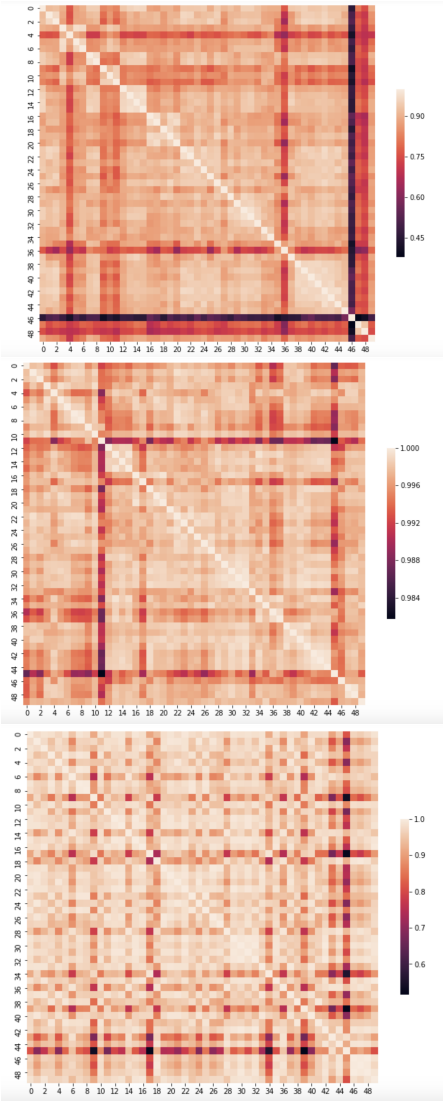

Figure 7. Correlation map on extracted features cross different videos, Top: extracted feature from pre-trained VGG model on ImageNet; Middle: extracted features from pre-trained model on VGGFace2; Bottom: extracted features from pre-trained facial-landmarks model

6. Discussion

We will analyze our results based on three aspects. First, we will focus on the effect of extracted features on our CCC scores. Second, we will discuss the limitations of using only visual modality to predict emotions. Finally, we will compare our results with human benchmark and discuss the challenges in emotion recognition.

6.1. Extracted Features

We experimented with 7 pre-trained models on different datasets. These pre-trained models were used as embedding layers to extract generic features. We already provided some sample visualizations on the extracted features from different models, shown in section 5.2. The major difficulty we encountered was that all of the pre-trained models on different datasets could only capture the feature of subtle facial changes over time at a limited level. As we briefly mentioned in section 5.2, if we looked at the plots in Figure 6, we noticed that the color variations in each column were not obvious. Among the three models, Facial Landmarks model and VGG on ImageNet model had the largest variations along columns, and VGGFace2 model had the least variations along columns. Pre-trained VGG model on ImageNet focused on classifying different objects, and pre-

trained VGG on VGGFace focused on classifying individual identities. These two models did not have such specific focus on the learning human facial features. For the other pre-trained models, similar reasons applied.

Facial Landmarks model focused on extracting the coordinates of eyes, brows, nose and mouth as features. Figure 2 presented 4 original images and their facial landmarks. The most left image showed positive emotion, and the second image showed negative emotion and the other two images correspond to neutral emotions. When a person was smiling, the features extracted would be quite different from the features when he/she was not smiling. We can recognize the difference of mouth shapes in the first and second image, but the second one does not different much from the two neural images. The algorithm seems mainly aims to extract the position of facial landmarks instead of precisely outlining their shapes.

6.2. Single Modality

Recall the data collection procedure, observers watched narrators describing their emotional stories and gave continuous valence ratings. Observers received, processed and combined information from three modalities in total, visual, acoustic and linguistic script whereas we only included visual input for this task. We used the linguistic information on the same dataset before and achieved CCC score above 0.4 on validation set [9]. Imagine two scenarios, reading emotional stories and watching narrative videos without sound, it is reasonable to infer that one can conjecture emotions of others more accurately based on scripts than silent videos. It further suggested in the context of storytelling in wild, linguistic and acoustic features usually deliver more emotional information than visual and acoustic features.

In addition, we took screenshots of the video every half second and use the extracted images as our features, however, there is no guarantee that we can fully capture facial expressions with two images per second. Explicit emotional expression can last less than half second. Scripts and acoustic input are both able to fully recover the whole story, but discrete images fail to do so. Moreover, facial expressions are not always consistent with innate feelings or emotions for adults. Lack of information on what participants are talking about, it is more challenge to speculate their emotions.

6.3. Human Evaluation

The human baseline CCC score for emotion cognition on this dataset is 0.46, by using acoustic, linguistic and visual modalities. This score is not very high either even using three modalities. Recognizing human emotions is a challenging task. The raw observer scores show large variations, and we use the average scores as the ground truth. This high variance on the original scores and subjective scores add even more difficulties for predicting emotions.

7. Conclusion and Future Work

Predicting emotion cognition is an important task. Accurate prediction can improve the interaction between AIs and human beings and thus AIs can better serve us human beings. Our project adopts transfer learning, and uses Transformer as encoder layers and LSTM as decoder layer to predict emotions. Our major work focuses on experimenting different pre-trained models to extract generic features that works best for our purpose. However, we encounters several difficulties that prevent us from achieving satisfying result. First of all, the extracted generic features from the existing pre-trained models do not represent our dataset well because most of them can not capture the subtle facial changes over time. Second, using one visual modality limits the emotion prediction since visual features alone usually can not tell a full story about human's emotion. Finally, the human baseline CCC score by using three modalities is not high either. Subjective ratings introduce high variance of the original emotional scores and the biased ground truth emotion ratings contribute to more difficulties.

From our explorations and analysis of the results, we have several suggestions for future work. One major improvement we can make for our project is to fine tune the model, instead of just using pre-trained model as embedding layers. Due to limited computing power, we had a hard time both training the model from scratch and fine tuning the pre-trained models using our own dataset. Therefore, as for future work, with increased computing power, we can fine tune the model and extract intermediate layers that appear earlier in the model for better and more complex feature representation that encodes more graphical data than classification information. Another improvement would be to use a combination of existing modalities: visual, acoustic and linguistic. Fusion network could be applied to combine the learned emotion information from each modality for a better integration.

8. Contributions and Acknowledgements

Each team member equally contributed to this project. Most of the works were done together during our group meetings, which included brain-storming, built the overall pipeline and analyzed the results. Some individual works included that Mengtian researched on VGGFace2 model and generated feature visualizations, Kayleen worked on extracting intermediate feature layers from pre-trained VGG and ResNet model and plotted feature visualizations, and Sherine researched on Facial Landmarks model and did most of the data pre-processing. We also equally split the write-up so each team member contributes to the final report in equal amount.

References

[1] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. *CoRR*, abs/1710.08092, 2017.

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

[3] H. Kaya, F. Gürpinar, S. Afshar, and A. A. Salah. Contrasting and combining least squares based learners for emotion recognition in the wild. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15, pages 459–466. ACM, 2015.

[4] H. Kaya, F. Grpnar, and A. Salah. Video-based emotion recognition in the wild using deep transfer learning and score fusion. *Image and Vision Computing*, 02 2017.

[5] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees.

[6] L. I.-K. Lin. A concordance correlation coefficient to evaluate reproducibility. *Biometrics*, pages 255–268, 1989.

[7] H.-W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler. Deep learning for emotion recognition on small datasets using transfer learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15, pages 92–99, 11 2015.

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] Z. Wu, X. Zhang, and X. Zhang. Emonet: Rescontruction of emotion as people read using deep neural network with attention. 3 2018.

[10] Z. Yu and C. Zhang. Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15, pages 435–442, 11 2015.