Facial Action Unit Detection With Transformers

Geethu Miriam Jacob
Rakuten Institute of Technology
geethu.jacob@rakuten.com

Björn Stenger
Rakuten Institute of Technology
bjorn.stenger@rakuten.com

Abstract

The Facial Action Coding System is a taxonomy for fine-grained facial expression analysis. This paper proposes a method for detecting Facial Action Units (FAU), which define particular face muscle activity, from an input image. FAU detection is formulated as a multi-task learning problem, where image features and attention maps are input to a branch for each action unit to extract discriminative feature embeddings, using a new loss function, the center contrastive (CC) loss. We employ a new FAU correlation network, based on a transformer encoder architecture, to capture the relationships between different action units for the wide range of expressions in the training data. The resulting features are shown to yield high classification performance. We validate our design choices, including the use of CC-loss and Tversky loss functions, in ablative experiments. We show that the proposed method outperforms state-of-the-art techniques on two public datasets, BP4D and DISFA, with an absolute improvement of the F1-score of over 2% on each.

1. Introduction

Facial expressions are a primary means of conveying nonverbal information. Some expressions are universally understood, such as exhilaration or disappointment. However, expressions are specific to individuals, which motivates a person-independent representation in the form of the Facial Action Coding System (FACS) [1]. This system describes a taxonomy of facial action units (FAU) for encoding facial expressions, based on the observed activation of muscles or muscle groups, such as *Brow Lowerer* or *Cheek Raiser*. Each facial expression can be mapped using this system, and automatic FACS analysis has been applied in domains such as healthcare, entertainment, and photography [2]. The task of FAU detection can be formulated as a multi-label binary classification problem for detecting each action unit [3, 4], with some works taking into account the degree of FAU activation [5, 6, 7].

Since action units are defined by particular muscle activations, the spatial extent of each action unit is limited. We therefore employ attention maps to focus on regions of interest. Our attention model builds on the idea of Attention Branch Networks [8], which introduce a branch structure for estimating attention, and have been shown to perform well on image classification and other visual tasks [8]. In our case, separate attention maps are learnt for each action unit. Additionally, since the FAUs are related with each other, the feature embeddings learnt using the attention maps are input to a correlation unit, using a transformer encoder [3, 9, 10].

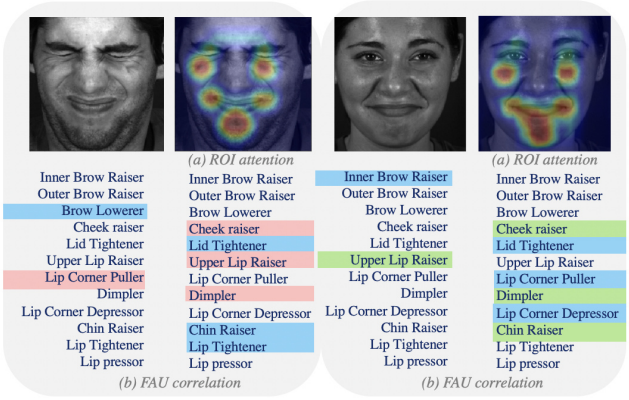

Figure 1: Overview. *Two levels of attention are captured in the proposed approach. (a) The first level focuses on the spatial regions using an Attention Branch Network [8] and (b) the second level of attention captures the relation between FAU feature embeddings using a transformer encoder. For each image, highly correlated FAUs are shown in same color.*

Figure 1 illustrates the two types of relationships we capture using our model. We model the spatial regions of individual action units using attention maps (Figure 1 (a)). We then capture the relationship between different action units by self-attention, using transformer encoders (Figure 1 (b)). In the figure, the spatial attention as well as the FAU correlations for two images are shown. The full list of AUs are listed to illustrate the AU correlations predicted by our model. Two sample active AUs are highlighted in the left and the corresponding correlated AUs (based on self-attention) are shown with the same color on the right in each

image. For example, in the first image, we show the correlated AUs of the *Brow Lowerer* and *Lip Corner Puller*. The correlated AUs of *Brow Lowerer*, as predicted by our model, are *Lid Tightener, Chin Raiser* and *Lip Tightener*. Similarly, the correlated AUs of *Lip Corner Puller* for the same image are *Cheek Raiser, Upper Lip Raiser* and *Dimpler*.

Detecting action units can be viewed as related, but separate tasks. Multi-task learning [11] is able to take advantage of such task relationships, motivating this approach in our architecture. The overall framework of our approach is illustrated in Figure 2. The method performs attention learning on face images for detecting active action units. Features are extracted with a pre-trained Inception network [12] and are passed to attention and multi-task (per-AU embeddings) modules, respectively. The attention module estimates attention maps for each action unit and predicts the active action units. The multi-task module branches the generated attention maps into one channel per action unit. A novel loss term, center contrastive loss (E_{CC}) is used to make the features as discriminative as possible. The extracted discriminative features (encoded with relative positional encoding) from the multi-task module are passed to an AU correlation unit based on transformers [10]. We adopt a Tversky loss function [13] to improve the performance of our model. The combined loss function for the attention module, the multi-task module and the AU correlation module is optimized directly, allowing end-to-end model training.

In summary, we propose a new model architecture for facial action unit detection, by (1) combining attention branch networks in a multi-task setting for focusing on the spatial regions of action units and merging branches for individual action units, (2) an action unit correlation module using a transformer encoder, (3) using a Tversky loss function for handling multi-label classification, and (4) using a new center contrastive loss term to learn discriminative features. We evaluate the resulting model on public datasets, showing state of the art performance, and evaluate the design choices using ablative studies.

2. Prior Work

Early work on action unit detection used landmarks to define regions of interests and subsequently train classifiers such as neural networks [14] or Support Vector Machines [15]. Image patches around detected landmarks and multi-label classification are learned jointly in JPML [16]. Facial landmarks have also been used to generate attention maps in different ways. For example, in EAC-Net [17] a single, fixed, attention map is created for each image, by combining all regions associated with action units. These attention maps are multiplied with CNN features to help focusing on the regions of interest. JAA-Net [18] jointly estimates the location of landmarks and the presence of action

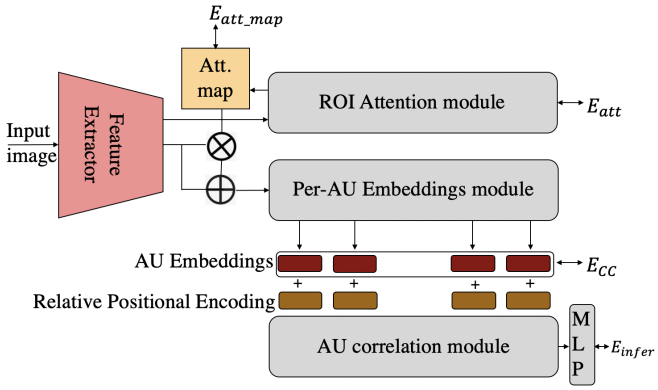

Figure 2: Proposed architecture. *The model includes ROI attention module, per-AU embeddings module, AU correlation module and a classification module. The full loss function includes attention map loss, $E_{\mathrm{att\_map}}$, center contrastive loss, E_{CC}, the label loss obtained from the attention module, E_{att}, and that from the classification module, E_{infer}.*

units. Landmarks are used to compute the attention map for each action unit separately. The recent ARL [19] uses hierarchical region learning to include various structure and texture information for FAUs in different local regions with the help of attention maps. Given the success of these approaches, we also follow an attention based approach. We provide supervision in the estimation of attention maps using landmarks as opposed to other methods. This helps in accurately focussing on the ROI relevant to each action unit. Also, in contrast to the fixed attention maps in the EAC-Net model, we predict the attention maps during inference time.

Multilabel FAU detection with imbalanced classes. Some action units, such as *Cheek Raiser* and *Lip Corner Puller* appear very frequently, whereas other action units, such as *Nose Wrinkler* and *Upper Lip Raiser* appear rarely, making FAU datasets imbalanced. SRERL [3] designed an adaptive cross entropy loss function for imbalanced data training based on the proportion of positive and negative samples in the training set. Weights used in [20] are inversely proportional to the ratio of positives in the total number of observations for each AU class in training. JAA-Net [18] uses a multi-label dice coefficient loss and weights based on the occurrence rate. Our method improves on this loss by using Tversky loss [13], which offers the flexibility in controlling the trade-off between false negatives and false positives. In contrast to other approaches, we use soft balance sampling weights [21], which offers flexibility in deciding whether to use class-aware or class-unaware weights.

Structure learning for AU detection. Several works explicitly take the relationships of AUs into account *e.g.* by using a Conditional Random Field (CRF) on top of a CNN [6]. A RNN model that learns representations and structure of

the input image in an end-to-end fashion is proposed in [20]. Similarly, AU correlation is modeled in [9] by using a knowledge graph of AU relationships and a Gated Graph Neural Network. Both [9] and [20] learn the spatial relationships of the AUs in face images, whereas the correlations of occurrence and intensity of AUs are learnt in [6]. The correlation between AUs is the motivation for introducing a AU correlation module based on transformer encoder [10] in our approach. Most recently, the work in [22] proposed a simple self-attention mechanism for facial features. The proposed method differs in two main points: we input discriminant features to the transformer, and we use full transformer encoders for estimating FAU correlations, rather than just self-attention.

Multi-task learning. Various multi-task architectures have been proposed for identifying facial attributes such as gender, hair color, eye glasses and hats. For example, the work in [23] trains individual networks on local regions for each attribute, then fine-tunes them using a joint loss function. In contrast, we avoid the need of attribute-specific image cropping. The work in [24] proposes an efficient algorithm for multi-objective optimization applied to multi-task learning, yielding good results on several tasks, including facial attribute prediction. The work in [25] introduces connections between all layers of task-specific networks, leading to improved attribute prediction. Multi-task learning is used in [26] to train attributes in groups based on their location and blur images outside these focus areas. Emotion recognition and facial action unit detection are considered two separate tasks in [27]. The work in [28] jointly learn landmarks and expressions. Other work aims to increase pose independence by combining pose regression with AU prediction [29]. These methods show that multi-task learning helps solving multiple related tasks. We therefore consider the estimation of separate AU embeddings as multiple related tasks.

3. Methodology

Our proposed framework, consists of four main modules, including a pre-trained feature extraction module, a ROI attention module, a multi-task module (per-AU embedding module) which extracts discriminative AU features and an AU correlation module, see Figure 2. The attention module focuses on regions of interest corresponding to each facial action unit. The multi-task module branches into N_{AU} different tasks, where N_{AU} is the number of action units, and each task is provided with the extracted features along with the attention map corresponding to the AU. Discriminative AU features are extracted from the multi-task module using the proposed center contrastive loss. The AU correlation module utilizes a transformer encoder to model the relationships between the discriminative AU features. The

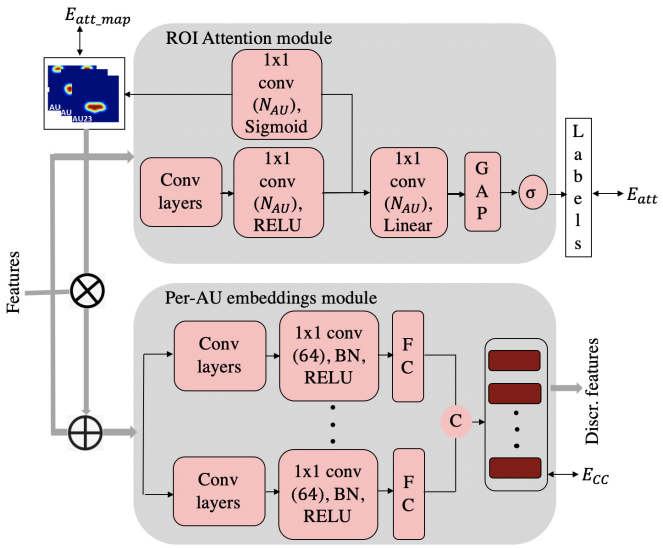

Figure 3: ROI attention and Per-AU embedding module. *The attention module and a multi-tasking module, which extracts discriminative features, are shown. The generated attention maps are compared with ground-truth maps using the loss, $E_{\text{att\_map}}$. Two other loss functions are the label loss obtained from the attention module, E_{att}, and the center contrastive loss, E_{CC}.*

network is trained in an end-to-end fashion by minimizing the following cost function:

$$E(\mathbf{x}) = \lambda_1 E_{\text{att}}(\mathbf{x}) + \lambda_2 E_{\text{infer}}(\mathbf{x}) + \lambda_3 E_{\text{att\_map}}(\mathbf{x}) + \lambda_4 E_{\text{CC}}(\mathbf{x}), \quad (1)$$

where $\mathbf{x}$ is the input image, $\lambda_1, ..., \lambda_4$ are hyper-parameters indicating the weights given to each cost term, $E_{\text{att}}(\mathbf{x})$ is the loss from the attention branch, $E_{\text{infer}}(\mathbf{x})$ is the loss between the predicted AU labels and the true labels, $E_{\text{att\_map}}(\mathbf{x})$ is the loss between the attention maps obtained from landmarks and the attention maps generated from the network, and $E_{\text{CC}}(\mathbf{x})$ is the center contrastive loss respectively. We found the optimal values of the hyper-parameters as, $\lambda_1 = \lambda_2 = 0.33$ and $\lambda_3 = \lambda_4 = 0.15$ through grid search. The following sections describe each module in more detail.

3.1. Pre-trained feature extraction

Previous approaches [3, 30, 18] use custom feature extractors. In this work we make use of pre-trained feature extractors for the problem of AU detection. We experimented with various pre-trained extractors and found that the features of InceptionV3 [12] provided the best performance among them. We take intermediate features (with a minimum resolution of 12) from the pre-trained models. Choosing intermediate layer features of the pre-trained models instead of the final layer features results in more accurate and explainable attention maps.

3.2. ROI Attention module

The aim of the attention module is to focus on the regions corresponding to active action units of the face image. As in the Attention Branch Network (ABN) [8], we use a separate network branch for estimating the attention maps and predicting the labels, see Figure 3. We learn the attention map using a robust Huber loss function to capture context better. The attention module consists of a few convolutional layers, a 1×1 convolution layer with N_{AU} filters (corresponding to the N_{AU} action units with output feature maps), an attention unit predicting N_{AU} attention maps, one for each FAU, a 1×1 convolution layer with N_{AU} filters, and a global average pooling (GAP) layer with a sigmoid activation function, yielding predictions of FAUs based on the predicted attention map. To focus on relevant ROIs, we provide supervision to the predicted attention maps. A ground-truth attention map is estimated for this purpose.

The ground-truth attention map is extracted as follows. We detect 66 facial landmarks [31] to initialize the attention maps of the AUs for a given input image. Landmarks specific to each action unit are defined similar to [17, 18]. We fit ellipses to landmarks as the initial regions of interest for each AU, smooth the image (Gaussian with $\sigma = 3$), thus obtaining N_{AU} attention maps. $E_{\text{att\_map}}$ constrains the attention map predicted by the attention module to be similar to the attention maps produced from action unit labels. We use a Huber loss function L_δ:

$$E_{\text{att\_map}} = L_\delta \left(F_M(f(\mathbf{x})) - A_m(\mathbf{x}) \right) , \tag{2}$$

where δ is the boundary between linear and quadratic loss, $f(\mathbf{x})$ is the output of the feature extraction stage, $F_M(f(\mathbf{x}))$ is the output of the attention module and $A_m(\mathbf{x})$ is the ground-truth attention map created from the action unit labels. The model was trained for different values of δ and the optimal value was found to be 0.5. $E_{\text{att}}(\mathbf{x})$ is the combination of weighted Tversky loss (WTL) and weighted cross-entropy loss (WCE) for this multi-label binary classification problem:

$$E_{\text{att}}(\mathbf{x}) = E^{\text{att}}_{wce}(\mathbf{x}) + E^{\text{att}}_{wtl}(\mathbf{x}). \tag{3}$$

AU detection datasets are generally imbalanced in nature. Hence, we use class weights along with the Tversky and cross entropy loss functions. We use soft-balance sampling for class weights [21]. The class weight of the i^{th} AU is given as $w_i = \frac{\sum_{i=1}^{N_{\text{AU}}} P_{s_i}}{P_{s_i}}$, where

$$P_{s_i} = \left(\frac{1}{N_{\text{AU}} * P_{n_i}}\right)^{\lambda} P_{n_i}, \quad P_{n_i} = \frac{n_i}{\sum_{j=1}^{N_{\text{AU}}} n_j},$$

and n_i is the number of images with the ith AU active and λ is a hyper-parameter, set to 0.7. Given the binary ground-truth labels $[y_{i0}, y_{i1}]$ associated with the input image $\mathbf{x}$ and the labels predicted by the attention branch of the model, $[p^{\text{att}}_{i0}, p^{\text{att}}_{i1}]$, the weighted binary cross-entropy loss and weighted Tversky loss associated with the ROI attention module are defined as:

$$\begin{aligned} E^{\text{att}}_{wce} &= \frac{1}{N_{\text{AU}}} \sum_i \sum_{j \in \{0,1\}} w_i y_{ij} p^{\text{att}}_{ij}, \\ E^{\text{att}}_{wtl} &= \frac{1}{N_{\text{AU}}} \sum_i \frac{w_i (\alpha p^{\text{att}}_{i1} y_{i0} + \beta p^{\text{att}}_{i0} y_{i1} + \epsilon)}{p^{\text{att}}_{i1} y_{i1} + \alpha p^{\text{att}}_{i1} y_{i0} + \beta p^{\text{att}}_{i0} y_{i1} + \epsilon}. \end{aligned} \tag{4}$$

Thus, the attention module predicts the attention map, which focuses on regions for each AU and passes them to the Per-AU Embedding module.

3.3. Per-AU Embeddings module

This module, designed as a multi-task network, estimates discriminative AU embeddings as output of N_{AU} tasks. Each branch of the module consists of a few convolution layers followed by a 1×1-convolution layer and a fully connected layer. The features from the fully connected layers of N_{AU} branches are provided as input to the AU correlation module. Discriminative AU features are obtained in this module using a novel center contrastive E_{CC} loss function and is defined as:

$$E_{\text{CC}}(\mathbf{x}) = \frac{\sum_{i=1}^{N_{\text{AU}}} \mathbb{1}_{y_{i1}==1} ||F^i_E(f(\mathbf{x})) - \mathbf{C}_i||^2}{\sum_{i=1}^{N_{\text{AU}}} \mathbb{1}_{y_{i1}\neq 1} ||F^i_E(f(\mathbf{x})) - \mathbf{C}_i||^2}, \tag{5}$$

where $F^i_E(f(\mathbf{x}))$ denotes the feature from i^{th} branch and $\mathbf{C}_j$ denotes the learned center of j^{th} feature of the faces with j^{th} action unit active. The formulation effectively reduces the intra-class variations and increases the inter-class variations. This loss function is a variant of the center loss [32]. Similar to the center loss, the centers are updated in each minibatch. After the weights of the model are updated for a minibatch, the centers are updated.

3.4. AU Correlation and Inference Module

The details of this module are illustrated in Figure 4. The AU correlation module estimates the relationships between the discriminative features. This unit consists of a transformer encoder [10], which takes the discriminative AU embeddings as input. The transformer encoder has two main components with normalization layers in between: Multi-Head Attention and Feed Forward Networks [10].

The features from the FAU correlation module are passed through a classifier with two fully connected layers to obtain the labels. A loss term similar to Equation 4 is used here, consisting of both WTL and WCE terms:

$$E_{\text{infer}}(\mathbf{x}) = E^{\text{inf}}_{wce}(\mathbf{x}) + E^{\text{inf}}_{wtl}(\mathbf{x}), \tag{6}$$

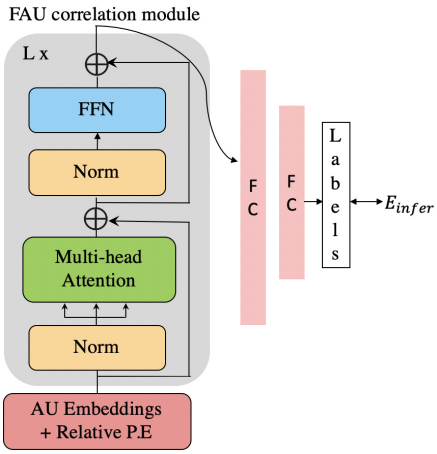

Figure 4: AU correlation learning. *Relationships between FAU embeddings are learnt using a transformer encoder. The features are passed to an MLP, which learns to predict the labels using the label loss* E_{infer}.

$$E^{\text{inf}}_{wce} = \frac{1}{N_{\text{AU}}} \sum_{i} \sum_{j \in \{0,1\}} w_i y_{ij} p^{\text{inf}}_{ij},$$

$$E^{\text{inf}}_{wtl} = \frac{1}{N_{\text{AU}}} \sum_{i} \frac{w_i(\alpha p^{\text{inf}}_{i1} y_{i0} + \beta p^{\text{inf}}_{i0} y_{i1} + \epsilon)}{p^{\text{inf}}_{i1} y_{i1} + \alpha p^{\text{inf}}_{i1} y_{i0} + \beta p^{\text{inf}}_{i0} y_{i1} + \epsilon}.$$

Here, $[p^{\text{inf}}_{i0}, p^{\text{inf}}_{i1}]$ denote the label of i^{th} action unit predicted by the classifier. The hyper-parameters α, β are set to 0.25 and 0.75, respectively, based on evaluation on a validation set. Note that we get two predictions, one from the attention branch and the other from the final classifier. The output from the classifier is taken as the predicted label during the time of inference.

4. Experimental Results

The proposed method is evaluated on three public datasets, BP4D [33, 34], DISFA [35], and EmotioNet [36, 37], which contain input images and action unit labels. For BP4D and DISFA, we set up experiments in line with prior work [3, 18]. We evaluate using three-fold cross-validation, and report mean values over the folds. The folds are the same as in prior work, for fair comparison. The BP4D dataset [33] contains images of 41 people (23 female, 18 male adults) of various ethnicity. In total, it includes approximately 140,000 face images with binary action unit labels (present or absent). As in prior work, we report the F1 score of 12 action units and use the same train/test splits as in [3, 18]. In order to avoid training and testing on images of the same person, different partitions contain images of different people.

The DISFA dataset [35] contains left and right views of 27 subjects, 130,815 frames in total, with AU intensities annotated in the range from 0 to 5. Following prior work [3, 18], we evaluate on 8 AUs and select only frames with intensity values greater than 2. The BP4D pre-trained network is fine-tuned on the DISFA dataset, also following the protocol in [3, 18].

The EmotioNet dataset [36] is a significantly larger dataset of Internet images, with a training set of approximately 950K images, automatically annotated by the algorithm in [36] and a validation set consisting of approximately 25,000 manually annotated images. There are 12 common AUs in the training and validation sets. The dataset is part of the EmotioNet challenge [38] with a withheld test set. We therefore analyze the performance of the proposed method on the validation set. We evaluate different training regimes: training the network pre-trained on ImageNet, training the network pre-trained on BP4D, training the network pre-trained on DISFA, and training the network pre-trained on both BP4D and DISFA.

Implementation details. Face regions are detected and resized to 224×224 grayscale images. Landmarks are used to initialize the attention maps for each AU. The size of the attention maps are 12×12 in order to match the dimensions of the convolution layer. Network weights are set using Xavier initialization [39] and optimized using Adam with hyper-parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The network was trained for 20 epochs per fold with learning rate 10^{-4} for BP4D and 10^{-5}, for DISFA, respectively. The features passed to our model are extracted from models pre-trained on ImageNet [40]. We compute the F1-score over a single fold of the BP4D dataset and select the hyper-parameters with the best score. For the test evaluation, we fix the hyper-parameters and evaluate by averaging over all three folds. We use the same hyper-parameters when testing on other datasets. Training the network takes approximately 8 hours on the BP4D and DISFA datasets and three days on EmotioNet. Inference takes 2ms per image using an RTX 2080Ti GPU.

4.1. Results on BP4D and DISFA datasets

Table 1 shows quantitative results on the BP4D dataset. We compare our method with published work [20, 3, 17, 9, 18, 19] in terms of F1-score. The results for other methods are taken from the papers. The methods in [17, 18] and [19] also include an attention mechanism, and [20, 3, 9] include correlations of action units. The proposed method performs better than state-of-the-art methods, with an average F1-score of 64.2.

Table 2 provides quantitative results on the DISFA dataset. The performance is evaluated for 8 action units, in line with prior work [20, 3, 17, 9, 18, 19]. The results for prior work is taken from the papers. In terms of F1-score, the proposed method performs best on half of the AUs, and,

Table 1: Results on the BP4D dataset. *Comparison with state-of-the-art methods using the F1-score metric (higher is better).*

AU	DSIN [20]	LP [9]	SRERL [3]	EAC [17]	JAA [18]	ARL [19]	Ours
1	**51.7**	43.4	46.9	39.0	47.2	45.8	**51.7**
2	40.4	38.0	45.3	35.2	44.0	39.8	**49.3**
4	56.6	54.2	55.6	48.6	54.9	55.1	**61.0**
6	76.1	77.1	77.1	76.1	77.5	75.7	**77.8**
7	73.5	76.7	78.4	72.9	74.6	77.2	**79.5**
10	79.9	83.8	83.5	81.9	**84.0**	82.3	82.9
12	85.4	87.2	**87.6**	86.2	86.9	86.6	86.3
14	62.7	63.3	63.9	58.8	61.9	58.8	**67.6**
15	37.3	45.3	**52.2**	37.5	43.6	47.6	51.9
17	62.9	60.5	**63.9**	59.1	60.3	62.1	63.0
23	38.8	**48.1**	47.1	35.9	42.7	47.4	43.7
25	41.6	54.2	53.3	35.8	41.9	55.4	**56.3**
Avg.	58.9	61.0	62.1	55.9	60.0	61.1	**64.2**

Table 2: Results on the DISFA dataset. *Comparison with state-of-the-art methods using the F1-score metric (higher is better).*

AU	DSIN [20]	LP [9]	SRERL [3]	EAC [17]	JAA [18]	ARL [19]	Ours
1	42.4	29.9	45.7	41.5	43.7	43.9	**46.1**
2	39.0	24.7	47.8	26.4	46.2	42.1	**48.6**
4	68.4	72.7	59.6	66.4	56.0	63.6	**72.8**
6	28.6	46.8	47.1	50.7	41.4	41.8	**56.7**
9	46.8	49.6	45.6	**80.5**	44.7	40.0	50.0
12	70.8	72.9	73.5	**89.3**	69.6	76.2	72.1
25	90.4	93.8	84.3	88.9	88.3	**95.2**	90.8
26	42.2	65.0	43.6	15.6	58.4	**66.8**	55.4
Avg.	53.6	56.9	55.9	48.5	56.0	58.7	**61.5**

Table 3: Pre-trained models. *We compare the F1-score of various models (ResNet50, InceptionNet (V3) and EfficientNet (B0)) trained on ImageNet data.*

Arch.	**ResNet50 [41]**	**EfficientNet [42]**	**InceptioNetV3 [12]**
F1-score	63.7	62.9	64.2

as on BP4D dataset, it shows the best average F1-score of 61.5. For both the datasets, our method is at least 2% better than the state-of-the-art methods.

4.2. Ablation Study

We carried out ablation studies on the BP4D dataset in order to evaluate different design choices and parameter settings of the proposed model. In particular, the contribution of the attention module, the multi-task architecture, the relevance of each cost term and the importance of the FAU correlation module are evaluated.

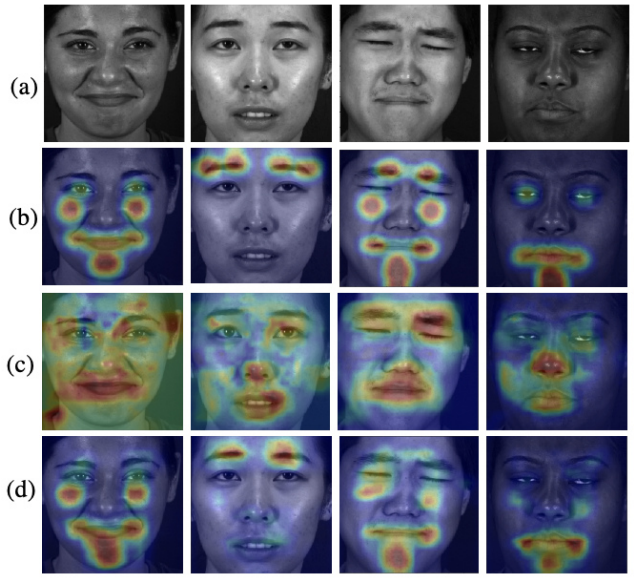

Figure 5: Effect of attention supervision. *(a) Input images (BP4D), combined attention maps of active action units (b) created from landmarks, (c) generated by the network without attention supervision, and (d) generated by our method with attention supervision.*

In order to assess the effect of different components we run the same experiments using variations of the proposed network, with and without the key components.

Table 3 shows the performance of our method with a few architectures for feature extraction. We tried our method with ResNet50 [41], EfficientNet [42] and InceptionNetV3 [12] architectures. In all the cases we take the intermediate features from the last layer which has a feature resolution of 12×12 or more. It is seen that InceptionV3 model gave the best performance. Inception architecture extracts features by convolution with different kernel sizes, thus looking at various resolutions. This helps is obtaining the features of FAUs occurring at multiple resolutions.

We test the contributions of the important components of our model, namely, pre-trained feature extractor (PT), multi-task architecture (MT), attention branch (AT), attention branch with attention map supervision (ATsup), center contrastive loss (CC) and transformer encoder (E). Table 4 shows the performance of various combinations of the components in terms of mean F1-score over all the FAUs.

The first test case is the use of pre-trained models and multi-task module. The baseline CNN consists of just the InceptionV3 [12]. Use of InceptionV3 model pre-trained on ImageNet (PT) improves the F1-score by 1.4%, whereas the use of a multi-task architecture with features extracted from pre-trained models (PT-MT), for each action unit improves

Table 4: Ablation study on BP4D. *In this experiment we compare models with different key components: Pretrained and multi-task architecture, attention (with and without attention map supervision), center contrastive loss, and transformer Encoder.*

Model	Pre-trained	Multi-task	Attention	E_{att_loss}	E_{CC}	Encoder	F1-score
baseline CNN	-	-	-	-	-	-	59.0
PT	✓	-	-	-	-	-	60.4
PT-MT	✓	✓	-	-	-	-	61.0
PT-MT-AT (ABN-MT [8])	✓	✓	✓	-	-	-	62.3
PT-MT-ATsup	✓	✓	✓	✓	-	-	62.5
PT-MT-CC	✓	✓	-	-	✓	-	62.6
PT-MT-AT-CC	✓	✓	✓	-	✓	-	62.6
PT-MT-ATsup-CC	✓	✓	✓	✓	✓	-	62.8
PT-MT-E	✓	✓	-	-	-	✓	63.4
PT-MT-CC-E	✓	✓	-	-	✓	✓	63.8
PT-MT-AT-E	✓	✓	✓	-	-	✓	63.5
PT-MT-ATsup-E	✓	✓	✓	✓	-	✓	63.6
PT-MT-AT-CC-E	✓	✓	✓	-	✓	✓	64.0
PT-MT-ATsup-CC-E	✓	✓	✓	✓	✓	✓	**64.2**

Table 5: Tversky and cross entropy loss. *In this experiment, we explore the performance in combinations of Weighted Tversky Loss (WTL) and Weighted Cross Entropy Loss (WCE) in both attention branch and MLP.*

E^{att}_{wtl}	E^{att}_{wce}	E^{inf}_{wtl}	E^{inf}_{wce}	F1-score
✓	-	✓	-	61.3
-	✓	-	✓	63.5
✓	-	✓	✓	64.0
-	✓	✓	✓	63.3
✓	✓	✓	-	61.4
✓	✓	-	✓	63.4
✓	✓	✓	✓	**64.2**

Table 6: Ablation study on transformer Encoder. *In this experiment, we explore the contributions of the components of the transformer.*

SHSA	MHSA	SHSA-FFN	Trig.P.E
63.2	63.8	63.5	63.9
Encoder-1	Encoder-2	Encoder-4	Encoder-5
64.1	64.1	63.3	63.7

the score by 2% (59.0 to 61.0).

The second test case is attention learning with (ATsup) and without (AT) attention map supervision. In the latter case, the cost function only includes two loss terms, E_{att} and E_{infer}. The supervision of attention comes with a overhead of computing facial landmarks. An increase of only 0.2% is caused with the attention supervision (PT-MT-ATsup) and our model performs very well even without the attention supervision. However, the attention branch with supervision improves the F1-score by 1.5% (61.0 to 62.5).

The third test case includes the center contrastive loss,

Table 7: Comparison with other loss functions. *We evaluate the existing design choices of loss functions and class weights.*

Design	Center loss [32]	Dice-loss [18]	JAA weights [18]	**Ours**
F1-score	63.6	64.0	63.3	**64.2**

E_{CC} (PT-MT-ATsup-CC) and is seen that E_{CC} improves the performance by 0.3%. As a final part of the ablation study, we studied the effect of transformer encoder. One observation is that the transformer encoder helps to significantly improve the results, even without the rest of the components. Note that the inclusion of just the transformer encoder (PT-MT-E) to the multi-task pre-trained model (PT-MT) improves the PT-MT-AT model by 2.4%. This shows the effect of the FAU correlation module in our method. All the components together yields the best results. All other components, such as parameters settings, training and evaluation procedure, remain unchanged in this experiment.

Effect of loss functions. A few visual results on the effect of attention map supervision is shown in Figure 5. All the input images are from BP4D dataset. The first row shows the input images of different subjects. A combined attention map created from ground truth labels are shown in the second row for visual comparison with the predictions. The combined attention map is obtained by taking pixel-wise maxima over only the active AU maps. The predicted attention maps (combined) without supervised training and our proposed model are shown in the last two rows. In all the cases, the attention maps of only the active action units are taken for the purpose of visualization. The proposed method generates significantly more localized attention maps, focusing on the relevant face regions. Note the similarity to the combined attention map based on ground truth labels.

Table 8: Results on EmotioNet. *This dataset of ∼1M Internet images contains large appearance variation and training label noise. F1-score on the manually annotated validation set are reported.*

Train Strategies	Ours-Imagenet	Ours-BP4D	Ours-BP4D-DISFA
F1-score	45.7	**47.3**	47.1

We compare performances of combinations of Tversky loss and cross entropy loss for attention branch (E_{att}) as well as the classifier (E_{infer}). Table 5 shows the various combinations of Tversky and cross-entropy losses. It is seen that the losses of the FC layers from the encoder are contributing more to the performance. Also, WCE loss contributes more to the performance.

The other design choices on loss functions, such as center contrastive loss (E_{CC}), Weighted Tversky loss ($E_w tl$) and class weights based on soft-balance strategy (w_i) are shown in Table 7. The design choices taken similar to these by existing methods are Center loss [32], Dice loss [18] and class weights based on number of true positives [18]. It is seen that all three design choices leads to a better performance when compared to the existing choices.

Components of transformer encoder. Table 4 shows the performance of the FAU correlation module. The two main components of the transformer encoder are multi-head self attention and feed-forward networks. Relative Positional Encoding is used to encode the input features before passing it to the Encoder. We evaluated the contribution of different Encoder components, see Table 6. SHSA, MHSA shows the performances of single-head and multi-head self-attention without the feed forward layers. SHSA-FFN shows the performance of the transformer encoder with single-head self attention and feed-forward networks. MHSA performs the best among the three (63.8). Our method (MHSA-FFN) achieves an F1-score of 64.2, showing that feed forward networks as well as MHSA improve the performance. We also evaluated Trigonometric Positional Embedding for encoding the features (63.9). In terms of encoder layers, we found that a 3-layer model was performing best.

We also experimented with L2 and L1 losses, which resulted in F1-scores of 63.1 and 64.0, respectively, on the BP4D dataset.

4.3. Experiments on EmotioNet

In order to evaluate the model on facial expressions 'in the wild', we ran experiments on the larger and less constrained EmotioNet 2018 Challenge dataset [36]. It contains nearly one million Internet images with large variations in illumination, pose and occlusions. The training data of 950K images was annotated automatically with a reported accuracy of 80% [36]. The major challenge with this dataset is the severity of the class imbalance and the variation in the head pose and expression. Another challenge in addition to label noise is that landmark detection using the dlib library, which works reliably on the previous datasets, fails on approximately 3% of the EmotioNet images. We trained the model using various pre-trained models for 50 epochs on the training set, using the same parameters as in the previous experiments on the BP4D dataset. We evaluated on the 25K images in the validation set, for which manually annotated labels are provided. Note that the test data is not publicly available at the time of submission, so the results cannot directly be compared with previously reported results [38]. Table 8 shows the results of our experiments. We tested with ImageNet feature extractor, BP4D and BP4D-DISFA pretrained models. Based on the number of action units used, the number of branches in our multi-task module changes. For each dataset, we select the branches to be trained based on the AUs present in it. Some examples of the success cases on EmotioNet dataset are shown in Figure 6. We assume that data augmentation and face alignment can improve the performance of our model.

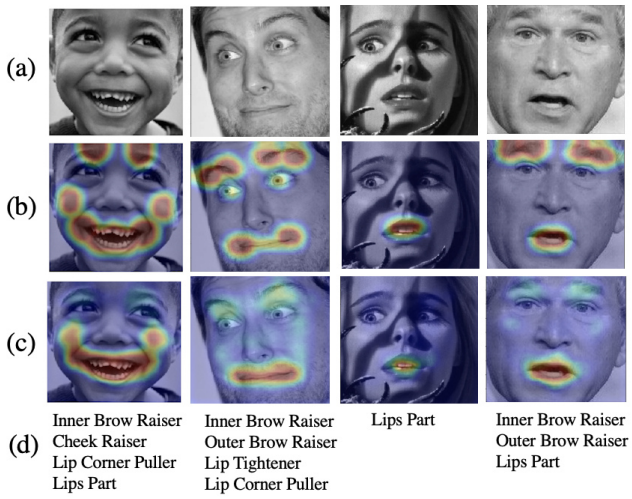

Figure 6: Qualitative results on EmotioNet. *(a) Input images, (b) combined attention map created from landmarks and (c) attention map (combined) predicted and (d) labels predicted.*

5. Conclusion

This paper proposed a new framework for facial action unit detection using an attention network and transformer-based FAU correlation model. Experiments showed that the combination of attention branch with supervision, using a multi-task approach with center contrastive loss and a transformer encoder to learn correlations leads to state-of-the-art results on the BP4D and DISFA datasets. Experiments on the larger EmotioNet dataset of web images, showed competitive results using the same architecture, and explored model variations for additional improvements.

References

[1] Rosenberg Ekman. "What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)". In: (1997).

[2] Michael A Sayette et al. "A psychometric evaluation of the facial action coding system for assessing spontaneous expression". In: *Journal of Nonverbal Behavior* (2001).

[3] Guanbin Li et al. "Semantic Relationships Guided Representation Learning for Facial Action Unit Recognition". In: *AAAI*. 2019.

[4] Kaili Zhao, Wen-Sheng Chu, and Honggang Zhang. "Deep region and multi-label learning for facial action unit detection". In: *CVPR*. 2016, pp. 3391–3399.

[5] Mohammad H Mahoor et al. "A framework for automated measurement of the intensity of non-posed facial action units". In: *CVPRW*. 2009.

[6] Robert Walecki et al. "Deep structured learning for facial action unit intensity estimation". In: *CVPR*. 2017.

[7] Yong Zhang et al. "Joint Representation and Estimator Learning for Facial Action Unit Intensity Estimation". In: *CVPR*. 2019.

[8] Hiroshi Fukui et al. "Attention branch network: Learning of attention mechanism for visual explanation". In: *CVPR*. 2019.

[9] Xuesong Niu et al. "Local Relationship Learning With Person-Specific Shape Regularization for Facial Action Unit Detection". In: *CVPR*. 2019.

[10] Ashish Vaswani et al. "Attention is all you need". In: *NeurIPS*. 2017.

[11] Richard A. Caruana. "Multitask Learning: A knowledge-based source of inductive bias". In: (1993).

[12] Christian Szegedy et al. "Rethinking the inception architecture for computer vision". In: *CVPR*. 2016.

[13] Xiaoya Li et al. "Dice Loss for Data-imbalanced NLP Tasks". In: *ACL*. 2020.

[14] Ying-Li Tian, Takeo Kanade, and Jeffrey F Colin. "Recognizing action units for facial expression analysis". In: *Multimodal interface for human-machine communication*. 2002.

[15] Michel Valstar and Maja Pantic. "Fully automatic facial action unit detection and temporal analysis". In: *CVPRW*. 2006.

[16] Kaili Zhao et al. "Joint patch and multi-label learning for facial action unit and holistic expression recognition". In: *IEEE Transactions on Image Processing* 25.8 (2016).

[17] Wei Li et al. "Eac-net: Deep nets with enhancing and cropping for facial action unit detection". In: *T-PAMI* (2018).

[18] Zhiwen Shao et al. "Deep adaptive attention for joint facial action unit detection and face alignment". In: *ECCV*. 2018.

[19] Zhiwen Shao et al. "Facial action unit detection using attention and relation learning". In: *Transactions on Affective Computing* (2019).

[20] Ciprian Corneanu, Meysam Madadi, and Sergio Escalera. "Deep structure inference network for facial action unit recognition". In: *ECCV*. 2018.

[21] Junran Peng et al. "Large-Scale Object Detection in the Wild from Imbalanced Multi-Labels". In: *CVPR*. 2020.

[22] Zhilei Liu et al. "Relation modeling with graph convolutional networks for facial action unit detection". In: *ICMM*. 2020.

[23] Yuechuan Sun and Jun Yu. "Deep Facial Attribute Detection in the Wild: From General to Specific". In: *BMVC*. 2018.

[24] Ozan Sener and Vladlen Koltun. "Multi-task learning as multi-objective optimization". In: *NeurIPS*. 2018.

[25] Yuchun Fang et al. "Dynamic Multi-Task Learning with Convolutional Neural Network". In: *IJCAI*. 2017.

[26] Sara Atito Aly and Berrin Yanikoglu. "Multi-Label Networks for Face Attributes Classification". In: *ICMEW*. 2018.

[27] Chu Wang et al. "Multi-Task Learning of Emotion Recognition and Facial Action Unit Detection with Adaptively Weights Sharing Network". In: *ICIP*. 2019.

[28] Terrance Devries, Kumar Biswaranjan, and Graham W Taylor. "Multi-task learning of facial landmarks and expression". In: *Canadian Conference on Computer and Robot Vision*. 2014.

[29] Yuqian Zhou, Jimin Pi, and Bertram E Shi. "Pose-independent facial action unit intensity regression based on multi-task deep transfer learning". In: *FG*. 2017.

[30] Wei Li et al. "EAC-Net: A region-based deep enhancing and cropping approach for facial action unit detection". In: *FG*. 2017.

[31] Davis E King. "Dlib-ml: A machine learning toolkit". In: *JMLR* (2009).

[32] Yandong Wen et al. "A discriminative feature learning approach for deep face recognition". In: *ECCV*. 2016.

[33] Xing Zhang et al. “BP4D-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database”. In: *Image and Vision Computing* 32.10 (2014).

[34] Michel F Valstar et al. “Fera 2015-second facial expression recognition and analysis challenge”. In: *FG*. 2015.

[35] S Mohammad Mavadati et al. “DISFA: A spontaneous facial action intensity database”. In: *Transactions on Affective Computing* (2013).

[36] C. F. Benitez-Quiroz, Ra. Srinivasan, and A. M. Martinez. “EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild”. In: *CVPR*. 2016, pp. 5562–5570.

[37] C Fabian Benitez-Quiroz et al. “Emotionet challenge: Recognition of facial expressions of emotion in the wild”. In: *arXiv preprint arXiv:1703.01210* (2017).

[38] A. M. Martinez et al. *EmotioNet Challenge*. http://cbcsl.ece.ohio-state.edu/EmotionNetChallenge/index.html. (accessed: March 1, 2020).

[39] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *AISTATS*. 2010.

[40] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *CVPR*. 2009.

[41] Kaiming He et al. “Deep residual learning for image recognition”. In: *CVPR*. 2016.

[42] Mingxing Tan and Quoc Le. “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”. In: *ICML*. 2019.