FAN-Trans: Online Knowledge Distillation for Facial Action Unit Detection

Jing Yang
y.jing2016@gmail.com

Jie Shen
jie.shen07@imperial.ac.uk

Yiming Lin
yl1915@ic.ac.uk

Yordan Hristov
yshristov@gmail.com

Maja Pantic
maja.pantic@gmail.com

1University of Nottingham, UK
2Imperial College London, UK

Abstract

Due to its importance in facial behaviour analysis, facial action unit (AU) detection has attracted increasing attention from the research community. Leveraging the online knowledge distillation framework, we propose the “FAN-Trans” method for AU detection. Our model consists of a hybrid network of convolution and transformer blocks to learn per-AU features and to model AU co-occurrences. The model uses a pre-trained face alignment network as the feature extractor. After further transformation by a small learnable add-on convolutional subnet, the per-AU features are fed into transformer blocks to enhance their representation. As multiple AUs often appear together, we propose a learnable attention drop mechanism in the transformer block to learn the correlation between the features for different AUs. We also design a classifier that predicts AU presence by considering all AUs’ features, to explicitly capture label dependencies. Finally, we make the attempt of adapting online knowledge distillation in the training stage for this task, further improving the model’s performance. Experiments on the BP4D and DISFA datasets demonstrating the effectiveness of proposed method.

1. Introduction

Facial behaviour is a natural and effective way to convey emotions, sentiments, and mental states in face-to-face communications. Due to its great potential in human-robot interaction, digital marketing, and psychological and behavioural research, automatic facial behaviour analysis has attracted increasing attention from both the academic community and the industry. Among different representations, facial action units (AU) provide the most comprehensive, expressive and objective descriptors for facial behaviours. They are defined over muscle movements according to the anatomy of human faces. As such, a robust AU detection method is important in facial behaviour understanding.

In Facial Action Coding System (FACS) [6], AUs are determined by appearance changes (i.e., geometry shape, textures) caused by facial muscle movements on a human face. Those changes are subtle, local and connected. For example, due to the underlying facial anatomy, AU1 and AU2 usually appear together because they are controlled by same muscle. In order to align with this area, the network design is desired to capture local AU features and consider the property of AU co-occurrences.

Facial landmarks, representing semantic key-points on a human face, are recognized as AU active locations. They are often used to crop the region of interest (ROI) for AUs [35, 33, 3] to reduce distraction from unrelated facial areas. Nevertheless, even after aligning faces to a common reference frame, precise AU localisation remains a challenge due to head pose and view variations, which will adversely affecting cropping ROI for AUs. Thus, we propose to extract intermediate representations from a pre-trained facial landmark detector for AU features learning. Due to the nature...
of face alignment task, these features are face specific and landmark-focused.

To consider the AU co-occurrences, previous works often used a standalone module to explicitly model the label or feature correlation between different AUs [27, 25, 11]. Recently, motivated by transformer network’s effectiveness in learning the correlations between distant patches in the image classification task [5, 24], TransAU [8] and TAM [20] pioneered in applying it to facial AU detection. Built upon this idea, we further introduce a learnable binary attention module to the transformer block to enhance its capability. This leads to more discriminative AU features.

Seminal works [35, 3, 17, 18, 11, 22, 8] are dedicated to architecture or loss designs to tackle the aforementioned issues, but to the best of our knowledge, none has tried to improve the learning procedure using online knowledge distillation (OKD). For AU detection, no high capacity model is readily available because increasing the number of parameters does not necessarily lead to higher accuracy due to the over-fitting problem caused by the limited amount of training data [25]. To address this, we propose to use OKD [30, 31, 10, 12] in our method. In contrast to two-stage knowledge distillation [7], OKD can boost the model’s accuracy without requiring a pre-trained teacher. It has achieved impressive results in tasks like object classification [10, 2], human pose estimation [12] but it has not been explored in the context of facial AU detection.

To this end, we formulate the AU detection task as a multi-label classification problem within the online knowledge distillation framework. Using a pre-trained face alignment network as the feature extractor, we only add a small subnet to learn per-AU features, as the intermediate feature maps produced by the face alignment network already provide rich shape and contextual information. To model AU co-occurrences, we propose a learnable attention drop mechanism on the self-attention module in the transformer block and significantly enhances the model’s performance by decreasing homogeneity among AU features. Additionally, we analyse two classifiers with one to predict per AU’s activation on a single AU’s feature and the other predicts all AUs’ existence based on it. We show that the latter achieves superior performance as it also learns the co-occurrences of AUs in the label space. Last but not least, we apply the OKD framework with diverse classifiers in the training stage to further improve the model’s accuracy, without incurring additional cost at inference time.

Our method, coined FAN-Trans, has achieved a new state-of-the-art performance on public benchmarks. The visualization of attention maps learned by FAN-Trans is shown in Figure 1. Without explicitly ROI assigned for different AUs [8, 22, 11], FAN-Trans can learn where to focus based on only supervision from AU labels.

The main contributions of this paper are listed below.

1. To our best knowledge, we are the first to adopt OKD with diverse peers designed for AU detection, improving its performance via ensemble learning.
2. Instead of manually assigning pre-defined regions to different AUs, FAN-Trans builds upon a pretrained multi-scale face alignment features to automatically learn the spatial correspondence between AUs and the underlying facial parts.
3. We propose to exploit both feature and label correlations in solving the AU detection task. For features, we design a transformer block with a learnable binary mask to learn sub connections of AUs; For labels, we devise a classifier to predict single AU’s activation on features of all AUs.
4. Through extensive experiments, we demonstrate the effectiveness of our proposed method on two widely used benchmark datasets: BP4D [32] and DISFA [14].

2. Related works

Regional feature representation Since AUs are defined in FACS [6] as muscular activation on the face, the AU detection task can be formulated as a classification task on local features extracted around facial landmarks. The early works utilised handcraft features. A typical pipeline in [33] was to first align crop faces, then extract handcraft features in a predefined patch around landmarks, and finally enhance feature representation by fusing texture features with geometry features formed by landmark coordinates. Recently, deep models have been widely used to capture local appearance changes for AU detection. For example, Zhao et al. [35] designed a region layer to induce specific facial regions for identifying different AUs. Shao et al. [18] developed an end-to-end multi-task framework to jointly do AU detection and face alignment, and the heat-maps were used to predefine the ROI which was further refined in model optimization. The aforementioned works linked AU detection with the ROI features, which have proven to be notably effective for AU detection. Building upon this insight, we propose to extract informative face shape and contextual priors from a pre-trained face alignment model and let the network automatically assign AU features during optimization.

AU co-occurrences modeling Due to the underlying facial anatomy, activations of different AUs are often correlated. Therefore, instead of detecting each AU independently, learning the co-occurrences of AUs can be incorporated either into the network design, or as a standalone refinement step. Some works [27, 25] realize it by attaching an explicit module on initial predictions with probabilistic graphical models. For example, the early attempt [27] exploited the restricted Boltzmann machine to learn AU relations. Similarly, [25] appended conditional random field on the top of fully connected layer output to force AU de-
The architecture of the proposed FAN-Trans is shown in Figure 2. This section will describe each component and the rational behind in detail.

3.1. Landmark Attention Feature Extraction

Facial landmarks represent the semantically salient regions of a human face. Given that AUs are present in local regions around facial landmarks, previous approaches [35, 11] used landmarks to predefine ROI for AUs. Three drawbacks could be observed in such assignment: extra time cost, sensitivity to precision of landmarks, and difficulty to embrace unregistered AUs. To avoid these, we take intermediate features from a pre-trained facial landmark model and learn AU embeddings on them.

Concretely, we leverage the stacked-hourglass-based FAN [15, 1] model for this purpose, which has been explored as the feature extractor for face recognition [28] and face emotion recognition [23]. FAN was trained for landmarks localization with heat-maps (Gaussian circle peaking at keypoints) as supervision on a large corpus of facial data, covering the full range of poses. Unlike models trained on ImageNet with a classification task which are generic (i.e. not specific to faces) and coarse, FAN features (1) capture the finer grained aspects of the face — a direct consequence of the face alignment task; (2) are robust to appearance vary-
A number of AUs have linear layer on the 2nd dimension to transform F. Given an input tensor F, D, H, W describe the feature tensor, and its channel, height, width respectively. We drop the symbol of batch size for brevity.

3.2. A Hybrid Network of Convolution and Transformer modules

Based on the AU definition, a successful AU detector is supposed to learn features around AU active regions and consider the property of AU co-occurrences. We use a hybrid convolution + transformer structure for learning the AU features, in which convolution is to obtain abstract face representations from FAN. F, D, H, W describe the feature tensor, and its channel, height, width respectively. We drop the symbol of batch size for brevity.

As illustrated in Figure 2, the convolution module is composed of several convolution layers: one 1x1 convolution layer to decrease the channel dimension from Da to 0.25Da; four convolution layers followed by a max-pooling to reduce spatial size by 4 times. The advantages of this module are from two aspects: First, it makes the feature representations more abstract by enlarging the receptive field (the resolution decreases from 64 x 64 to 4 x 4); Second, it reduces the computational complexity by decreasing feature dimension as self-attention in transformer operates across sequential tokens by drawing pairwise interactions. Given an input tensor F, the output of convolution module is Fc ∈ Rega × Ho × Wa.

Furthermore, we implicitly allocate per AU embedding with a linear transformation. We first flatten 3D Fc across spatial dimension to 2D tensor Fc ∈ Rega × Ho × Wa, then apply a linear layer on the 2nd dimension to transform Ha × Wa to number of AUs Nau. This is formulated by:

\[ F_{au} = F_c \times W_{au}, \]  

where Wa ∈ RHa × Wa, and Fau ∈ RDa × Nau. By doing so, per AU feature is associated with Fc with learnable parameters optimized by the task loss. Every vector Da × 1 is considered as one AU’s representation.

The transformer module is exploited here to model the AU co-occurrences. A learnable position embedding is applied on Fc. The main component of the transformer [5] module is the stacked encode blocks (transblock) composed of multi-head self-attention (MHSA) and multi-layer projection (MLP) (see Figure 3). To utilize it, we apply transpose operation on Fau ∈ R Da × Nau to obtain Fau ∈ R Di × Nau × Dk as transformer here is to modeling connections among AU features. For MHSA module, linear layers are applied on Fau to obtain queries Q ∈ R Nau × Dq, keys K ∈ R Nau × Dk and values V ∈ R Nau × Dc (Dq == Dk), then the attention map A is calculated through:

\[ A = \text{softmax} \left( \frac{QK^\top}{\sqrt{D_q}} \right), \]  

where A ∈ R Nau × Nau with A[i,j] representing Q[i,:]'s correlation with K, and A[j,i] representation Q's correlation with K[i,:]. We further propose a learnable attention drop mechanism by multiplying A with learned binary mask M ∈ R Nau × 1 to drop some AUs connections with others.

\[ \overline{A}_{i,j} = M_i A_{i,j}. \]  

where \( \overline{A}_{i,j} \) is supposed to increase the difference among AU features:

3.3. Online Knowledge Distillation with Diverse Classifiers

The main structure is a two-branch architecture, which is illustrated in Figure 2. We feed Fau into two diverse peers (T1 and T2). We denote the outputs of T1 and T2 as F1 ∈ R Nu × Di and F2 ∈ R Nu × Di, respectively. We design the diverse classifiers as follows:

One-to-One classifier (C1o2o): We pass F1 through a fully connected layer Wo ∈ R Di × 1, and the prediction is computed as

\[ P_o = \text{F1} \times W_o, \]  

where Po ∈ R Nu × 1 with i-th element F1[i] predicting i-th AU’s activation.

One-to-Many classifier (C1o2m): We multiply F2 with a transformation matrix Wm ∈ R Di × Nu, and the prediction is calculated as

\[ P_m = \text{TopK}(F_2 \times W_m), \]
where $P_m \in \mathbb{R}^{1 \times N_{au}}$. The output of $F_2 \times W_m$ is $P_m \in \mathbb{R}^{N_{au} \times N_{au}}$ and TopK is a selection function. The i-th slice of $F_2[i, :]$ predicts the existence of all AUs. $P_m[i, :]$ denotes the i-th AU’s evidence from all AU features. TopK operation summarizes the highest k confidence in $P_m[i, :]$ to get the final prediction for i-th AU:

$$P_m[i] = \sum_{j=0}^{K_i} P_m[i, j],$$

where $K_i$ denotes the top-k index per i-th column. The illustration of $P_m$’s calculation is in Figure 4.

Figure 4. Steps for calculating $P_m$. We assume $N_{au} = 4$ for illustration purposes. $F_2[i, :]$ and $W_m[i, :]$ denote i-th AU’s feature and i-th AU’s prototype, respectively. Hence, i-th AU’s feature ($F_2[i, :]$) contributes to the logits of all AUs ($P_m[i, :]$) and all AUs’ features contribute to the logits of j-th AU ($P_m[\cdot; i]$). To avoid noise accumulation, $P_m[i]$ is aggregated (summed) from the highest k ($k=2$) values in $P_m[i, ::]$

Online knowledge distillation Based on the two-branch formulation, we have two transformer modules with the same number of transblocks but different AU classifiers. Both branches share the convolution module for compact feature extraction and per AU feature assignment. Inspired by previous works [10, 12], the ensemble weight generator is placed on $F_c$. Its structure is depicted in Figure 2 (Ensemble weights generator). Specifically, it contains two splits to capture features across different receptive fields, concatenation operation to enrich representation, and fully connected layer to produce weights $W_c \in \mathbb{R}^{2 \times N_{au}}$. Afterwards, the Softmax is used to normalize the weights. Finally, the ensemble weights $W_c$ are split into $W_1 \in \mathbb{R}^{1 \times N_{au}}$ and $W_2 \in \mathbb{R}^{1 \times N_{au}}$ and the ensemble target is computed by element-wise summation:

$$P_t = W_1 * P_o + W_2 * P_m^T.$$

The OKD is formulated as the Kullback-Leibler (KL) divergence loss between $P_t$ and the student’s prediction ($P_s \subseteq \{P_o, P_m\}$):

$$\mathcal{L}_{KD} = KL(P_s, P_t).$$

The multi-label classification loss is formulated as weighted binary cross entropy loss with ground-truth label:

$$\mathcal{L}_{cls} = - \sum_{i} w_i[y_i \log p_i + (1 - y_i) \log(1 - p_i)],$$

where $p_i$ denotes i-th AU’s prediction, which has three sources $P_o, P_m$, and $P_t$. $y_i$ is the ground-truth label for i-th AU (1 denotes AU appears and 0 denotes AU does not present). $w_i$ is the class weight for each AU based on AU’s occurrence to balance training [17].

The final network is trained in an end-to-end manner by minimizing the following cost function:

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{kd},$$

in which $\lambda$ is hyper-parameter to balance $\mathcal{L}_{cls}$ and $\mathcal{L}_{kd}$. It should be noted that only the $C_{o2m}$ branch is deployed at inference time.

4. Experiments

To validate the effectiveness of FAN-Trans, we conduct experiments on two widely used AU detection datasets: BP4D [32] and DISFA [14].

4.1. Implementation Details

Datasets BP4D [32] contains 41 participants with 23 females and 18 males who were involved in 8 spontaneous expressions sessions. In total, 328 videos with 140,000 frames are recorded and then annotated with 12 AUs (AU1, AU2, AU4, AU6, AU7, AU10, AU12, AU14, AU15, AU17, AU23, and AU24). We evaluate the models with subject exclusive 3-fold cross-validation protocol following existing works [17, 18, 11, 8], where two folds are for training while the remaining one is for testing.

DISFA [14] involves 27 participants with 12 females and 15 males. Each person is documented in a video. The entire dataset consists of over 100,000 frames with intensity annotation ranging from 0 to 5 on 12 AUs. Following the protocol in [17, 18, 11, 8], we select 8 AUs (AU1, AU2, AU4, AU6, AU9, AU12, AU25, and AU26) for subject exclusive 3-fold cross-validation, and use intensity 2 as a threshold to distinguish between positive and negative samples.

Evaluation criteria We evaluate the models with F1-score [9] that considers per AU’s precision and recall and is widely used in the multi-label classification task, especially when samples are imbalanced across categories. Here, we calculate the F1-score on 12 AUs for BP4D and 8 AUs for DISFA. Per-AU’s score indicates the performance of different models on individual AU while the average F1-score across all AUs exhibits one model’s overall performance.

Training details All images are cropped based on the detection box provided by RetinaFace [4]. We do not use facial landmarks to pre-process faces thus our input to the network is unaligned face images. Since we use FAN [1] to extract features, our network input is a 256x256 RGB face image. Similar to JÄANet [18], we use random rotation (+/- 15 degrees), horizon flipping, scaling (0.75 - 1.25) and centre shifting (-10 - 10) for data augmentation. FAN [1] is pre-trained and fixed while other learnable parameters are optimized using AdamW with hyper-parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, weight decay $\gamma = 0.0001$, and learning rate $\eta = 1 \times 10^{-4}$ with linear warmup for 100 steps and then cosine annealing for 400 steps in total.

6023
The network is trained for 12 epochs per fold with starting learning rate at 0.0001 and decaying 30% every 4 epochs. We use the timm library \(^1\) to implement our transformer with its parameter settings like MLP ratio 4 and head number = dim//64 (dim denotes feature dimension). For DISFA, following [18, 11], we use weights trained on BP4D as initialization and then fine-tune on DISFA. To calculate F1-score, we binarize the Sigmoid predictions with the threshold 0.5. The weight balance parameter \(\lambda\) is set to 0.2 after grid search. All the implementations are based on PyTorch [16].

### 4.2. Ablation Studies

We carry out ablation studies on BP4D to reckon the elements contributing to the efficacy of the proposed framework. In particular, the contributions of the transformer module, the One-to-Many classifier, the OKD instances, attention drop mechanism in the transformer are analysed. For fair comparison, all variants are trained with the same setting including data augmentation and training schedule.

**Effectiveness of Transformer** To verify the effectiveness of the transformer module, we remove the transformer module of the proposed framework by directly performing a multi-label classification on the feature maps produced by the convolution module (see \(F_1\) in Figure 2). The AU detection is done by first average pooling the 4-D dimensional tensor, then feeding the generated 1-D vector to a fully connected classifier. We call this variant Baseline. We implement another two variants by feeding the convolution module’s output to the identical transformer module with separate classifiers \(C_{o2o}, C_{o2m}\) for comparison. We call them \(C_{T,C_{o2o}}, C_{T,C_{o2m}}\) respectively. At this point, no knowledge distillation part is included.

The results are in Table 1. Compared with Baseline, both \(C_{T,C_{o2o}}\) and \(C_{T,C_{o2m}}\) get higher F1-score in the majority of AUs with obvious improvement on AU2, AU6, AU17, AU23. Overall, in terms of the average F1-score, \(C_{T,C_{o2o}}\) and \(C_{T,C_{o2m}}\) exceed Baseline by 1.3% and 1.8% separately. We believe the advancement comes from the more representative features learned by the transformer module. Besides, simple Baseline alone surpasses JAANet[17], DSIN [3] with more complex feature extractors, which reveals the strong representability of FAN features.

**Effectiveness of classifiers** \(C_{T,C_{o2o}}\) outperforms \(C_{T,C_{o2m}}\) by 0.5% in average F1-score. We argue \(C_{T,C_{o2m}}\) is superior to \(C_{o2m}\) because the former one considers the AU co-occurrences in both feature learning and classification stages while the latter only captures this characteristic in feature learning within the transformer module. This argument is evidenced by Figure 5 which presents the correlation coefficients between pairwise AUs from annotations, predictions of \(C_{o2o}\) and predictions of \(C_{o2m}\) separately. For example, AU14 is associated with AU6, AU7, AU10, AU12 in Figure 5(a). This pattern presents the correlation coefficients between pairwise AUs from annotations, predictions of \(C_{o2o}\) and predictions of \(C_{o2m}\) in learning AU associations from label statistics.

**Improvement from OKD training** We first test a variant \(C_{T1,C_{o2o}}, C_{T1,C_{o2m}}\) which shares the feature extraction (\(C_{T1}\)) but is fed into two classifiers \(C_{o2o}\) and \(C_{o2m}\). From the table, this variant gains performance over student alone (i.e. \(C_{T,C_{o2o}}, C_{T,C_{o2m}}\)).

Such improvement motivates us to explore key factors concerning the OKD: where to put split point and how to increase peers’ diversity. We set up three variants: (I) \(C_{T1,C_{o2o}}, C_{T2,C_{o2o}}\), (II) \(C_{T1,C_{o2m}}, C_{T2,C_{o2m}}\), (III) \(C_{T1,C_{o2o}}, C_{T2,C_{o2m}}\). The architecture for (III) is in Figure 2, in which the breakpoint for two branches is after the convolution module. (I) replaces \(C_{o2m}\) in T2 branch of (III) with \(C_{o2o}\) while (II) replaces \(C_{o2m}\) in T1 branch of (III) with \(C_{o2m}\). From the Table 1, all the variants obtain the average F1-score gains with (III) ranking 1st. (I) and (II) outperform \(C_{T1,C_{o2o}, T1,C_{o2m}}\) by 0.3% and 0.4% re-

---

\(^1\)https://github.com/rwightman/pytorch-image-models

---

### Table 1. Ablation studies on BP4D. We compare variants with various key components: with or without the transformer module, different classifiers, online knowledge distillation, and different attention drop mechanisms.

<table>
<thead>
<tr>
<th>Methods</th>
<th>AU1</th>
<th>AU2</th>
<th>AU4</th>
<th>AU6</th>
<th>AU7</th>
<th>AU10</th>
<th>AU12</th>
<th>AU14</th>
<th>AU15</th>
<th>AU17</th>
<th>AU23</th>
<th>AU24</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>51.6</td>
<td>39.0</td>
<td>60.0</td>
<td>72.8</td>
<td>79.0</td>
<td>87.2</td>
<td>62.0</td>
<td>49.3</td>
<td>58.1</td>
<td>48.9</td>
<td>51.1</td>
<td>51.1</td>
<td>61.5</td>
</tr>
<tr>
<td>(C_{T,C_{o2o}})</td>
<td>55.8</td>
<td>44.9</td>
<td>56.9</td>
<td>77.8</td>
<td>75.6</td>
<td>82.8</td>
<td>87.5</td>
<td>61.3</td>
<td>48.7</td>
<td>61.9</td>
<td>48.2</td>
<td>52.5</td>
<td>62.8</td>
</tr>
<tr>
<td>(C_{T,C_{o2m}})</td>
<td>52.7</td>
<td>47.0</td>
<td>56.8</td>
<td>75.0</td>
<td>75.3</td>
<td>82.2</td>
<td>88.0</td>
<td>63.1</td>
<td>51.9</td>
<td>64.3</td>
<td>49.5</td>
<td>53.7</td>
<td>63.3</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T1,C_{o2m}})</td>
<td>55.7</td>
<td>42.5</td>
<td>60.9</td>
<td>76.4</td>
<td>76.7</td>
<td>83.6</td>
<td>86.6</td>
<td>62.4</td>
<td>47.4</td>
<td>65.8</td>
<td>49.5</td>
<td>57.9</td>
<td>63.8</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T2,C_{o2o}})</td>
<td>55.1</td>
<td>46.8</td>
<td>58.8</td>
<td>77.5</td>
<td>74.7</td>
<td>83.4</td>
<td>87.2</td>
<td>63.4</td>
<td>48.9</td>
<td>65.9</td>
<td>50.2</td>
<td>56.9</td>
<td>64.1</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T2,C_{o2m}})</td>
<td>58.0</td>
<td>46.8</td>
<td>59.9</td>
<td>76.5</td>
<td>76.0</td>
<td>83.6</td>
<td>87.3</td>
<td>60.3</td>
<td>49.9</td>
<td>66.1</td>
<td>49.9</td>
<td>56.6</td>
<td>64.2</td>
</tr>
<tr>
<td>(C_{T1,C_{o2m}, T2,C_{o2m}})</td>
<td>55.4</td>
<td>46.0</td>
<td>59.8</td>
<td>78.7</td>
<td>77.7</td>
<td>82.7</td>
<td>88.6</td>
<td>64.7</td>
<td>51.4</td>
<td>65.7</td>
<td>50.9</td>
<td>56.0</td>
<td>64.8</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T2,C_{o2m}})</td>
<td>49.4</td>
<td>46.0</td>
<td>61.2</td>
<td>77.4</td>
<td>77.6</td>
<td>83.0</td>
<td>88.4</td>
<td>65.9</td>
<td>50.3</td>
<td>63.6</td>
<td>50.3</td>
<td>52.2</td>
<td>63.8</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T2,C_{o2m}})</td>
<td>51.8</td>
<td>44.8</td>
<td>58.7</td>
<td>77.6</td>
<td>77.4</td>
<td>82.5</td>
<td>87.6</td>
<td>64.7</td>
<td>51.7</td>
<td>66.0</td>
<td>49.7</td>
<td>56.2</td>
<td>64.1</td>
</tr>
<tr>
<td>(C_{T1,C_{o2o}, T2,C_{o2m}})</td>
<td>54.0</td>
<td>43.1</td>
<td>60.3</td>
<td>77.2</td>
<td>78.1</td>
<td>84.3</td>
<td>86.7</td>
<td>63.2</td>
<td>51.6</td>
<td>65.0</td>
<td>48.5</td>
<td>56.9</td>
<td>64.1</td>
</tr>
</tbody>
</table>

\(\beta_2 = 0.999\) without weight decay. The network is trained for 12 epochs per fold with starting learning rate at 0.0001 and decaying 30% every 4 epochs. We use the timm library \(^1\) to implement our transformer with its parameter settings like MLP ratio 4 and head number = dim//64 (dim denotes feature dimension). For DISFA, following [18, 11], we use weights trained on BP4D as initialization and then fine-tune on DISFA. To calculate F1-score, we binarize the Sigmoid predictions with the threshold 0.5. The weight balance parameter \(\lambda\) is set to 0.2 after grid search. All the implementations are based on PyTorch [16].
The page contains a detailed discussion on the effectiveness of attention drop in Transformer models. The text is divided into several main sections:

1. **Effectiveness of attention drop in Transformer**
   - The text explains how the inclusion of attention mechanisms can improve the performance of Transformer models, particularly in the context of facial AU detection.
   - It mentions that attention mechanisms benefit different classifiers, with some cases showing significant improvements, while others may not.
   - The text also highlights the use of pre-trained features, noting that they can boost peer diversity.
   - An experiment is described where the performance of the OKD framework is evaluated, with results indicating that attention drop can increase diversity among different AU representations.

2. **Impact of Feature Numbers**
   - The text discusses the impact of encoding numbers in the transformer module on the F1-score in percentages.
   - Table 2 provides a comparison of F1-score for different numbers of encodes, showing that the performance can drop quickly if very few encodes are utilized.
   - The table also includes a comparison of baseline methods, with BL-HM achieving the highest F1-score.

3. **Impact of No. of TransBlock**
   - Table 3 presents the impact of the number of TransBlocks on the BP4D dataset, with F1-score in percentages.
   - The table shows that increasing the number of TransBlocks can improve performance, with cases showing gains of 0.5%.

4. **Sensitivity to $\lambda$ for FAN-Trans on BP4D**
   - Table 4 illustrates the sensitivity to $\lambda$ for FAN-Trans on BP4D, with F1-score in percentages.
   - The table shows that the performance varies with different values of $\lambda$, with cases showing gains of 64.8%.

The text is rich with diagrams and tables that illustrate the experimental results and show the distribution of AU active areas. The diagrams include correlation maps and pairwise cosine similarity distributions, which are used to support the findings.


### 4.3. Comparison with State-of-the-Art methods

We compare FAN-Trans with published AU detection techniques including the methods focusing on attention or regional feature i.e. JPAI [34], DRML [35], JAA[Net [17], JAA[Net [18], methods taking the relationship of AUs into account i.e. DSIN [3], SRERL [11], UGN-B [19] and very lately proposed works TransAU [8] and MONET [21]. The results for other methods are taken from the papers [8, 22]. Table 5 shows the performance comparison on the BP4D [32]. The proposed method performs better than the SOTA methods with an average F1-score 64.8%.

Table 6 compares the performance of our proposed FAN-Trans with SOTA methods on the DISFA [14]. It can be seen that our method obtains a 63.8% average F1-score. It is 2% better than TransAU [8] which also deploys transformer in model design.

#### Table 5. Comparison of the classification results (F1-score in %) with other methods on BP4D. Bold numbers indicate the best performance; Bracketed numbers indicate the second best.

<table>
<thead>
<tr>
<th>Methods</th>
<th>AU1</th>
<th>AU2</th>
<th>AU4</th>
<th>AU6</th>
<th>AU7</th>
<th>AU10</th>
<th>AU12</th>
<th>AU14</th>
<th>AU15</th>
<th>AU17</th>
<th>AU23</th>
<th>AU24</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPML [34]</td>
<td>32.6</td>
<td>25.6</td>
<td>37.4</td>
<td>42.3</td>
<td>50.5</td>
<td>72.2</td>
<td>74.1</td>
<td>65.7</td>
<td>38.1</td>
<td>40.0</td>
<td>30.4</td>
<td>42.3</td>
<td>45.9</td>
</tr>
<tr>
<td>DRML [35]</td>
<td>36.4</td>
<td>41.8</td>
<td>43.0</td>
<td>55.0</td>
<td>67.0</td>
<td>66.3</td>
<td>65.8</td>
<td>54.1</td>
<td>33.2</td>
<td>48.0</td>
<td>31.7</td>
<td>30.0</td>
<td>48.3</td>
</tr>
<tr>
<td>JAA[Net [17]</td>
<td>47.2</td>
<td>44.0</td>
<td>54.9</td>
<td>77.5</td>
<td>74.6</td>
<td>[84.0]</td>
<td>86.9</td>
<td>61.9</td>
<td>43.6</td>
<td>60.3</td>
<td>42.7</td>
<td>41.9</td>
<td>60.0</td>
</tr>
<tr>
<td>JAA[Net [18]</td>
<td>53.8</td>
<td>47.8</td>
<td>58.2</td>
<td>[78.5]</td>
<td>75.8</td>
<td>82.7</td>
<td>88.2</td>
<td>63.7</td>
<td>43.3</td>
<td>61.8</td>
<td>45.6</td>
<td>49.9</td>
<td>62.4</td>
</tr>
<tr>
<td>DSIN [3]</td>
<td>51.7</td>
<td>40.4</td>
<td>56.0</td>
<td>76.1</td>
<td>73.5</td>
<td>79.9</td>
<td>85.4</td>
<td>62.7</td>
<td>37.3</td>
<td>62.9</td>
<td>38.8</td>
<td>41.6</td>
<td>58.9</td>
</tr>
<tr>
<td>SRERL [11]</td>
<td>46.9</td>
<td>45.3</td>
<td>55.6</td>
<td>77.1</td>
<td>[78.4]</td>
<td>83.5</td>
<td>[87.6]</td>
<td>63.9</td>
<td>[52.2]</td>
<td>[63.9]</td>
<td>47.1</td>
<td>53.3</td>
<td>62.1</td>
</tr>
<tr>
<td>UGN-B [19]</td>
<td>[54.2]</td>
<td>[46.4]</td>
<td>56.8</td>
<td>76.2</td>
<td>76.7</td>
<td>82.4</td>
<td>86.1</td>
<td>64.7</td>
<td>51.2</td>
<td>63.1</td>
<td>48.6</td>
<td>53.6</td>
<td>63.3</td>
</tr>
<tr>
<td>TransAU [8]</td>
<td>51.7</td>
<td>49.3</td>
<td>61.0</td>
<td>77.8</td>
<td>[79.5]</td>
<td>82.9</td>
<td>86.3</td>
<td>67.6</td>
<td>51.9</td>
<td>63.0</td>
<td>43.7</td>
<td>56.3</td>
<td>64.2</td>
</tr>
<tr>
<td>MONET [21]</td>
<td>54.5</td>
<td>45.0</td>
<td>61.5</td>
<td>75.9</td>
<td>78.0</td>
<td>84.5</td>
<td>87.6</td>
<td>[65.1]</td>
<td>[54.8]</td>
<td>60.5</td>
<td>53.0</td>
<td>53.2</td>
<td>64.5</td>
</tr>
<tr>
<td><strong>FAN-Trans</strong></td>
<td><strong>55.4</strong></td>
<td><strong>46.0</strong></td>
<td>[59.8]</td>
<td><strong>78.7</strong></td>
<td><strong>77.7</strong></td>
<td><strong>82.7</strong></td>
<td><strong>88.6</strong></td>
<td><strong>64.7</strong></td>
<td><strong>51.4</strong></td>
<td><strong>65.7</strong></td>
<td>[50.9]</td>
<td><strong>56.0</strong></td>
<td><strong>64.8</strong></td>
</tr>
</tbody>
</table>

#### Table 6. Comparison of the classification results (F1-score in %) with other methods on DISFA. Bold numbers indicate the best performance; Bracketed numbers indicate the second best.

<table>
<thead>
<tr>
<th>Methods</th>
<th>AU1</th>
<th>AU2</th>
<th>AU4</th>
<th>AU6</th>
<th>AU7</th>
<th>AU10</th>
<th>AU12</th>
<th>AU14</th>
<th>AU15</th>
<th>AU17</th>
<th>AU23</th>
<th>AU24</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRML [35]</td>
<td>17.3</td>
<td>17.7</td>
<td>37.4</td>
<td>29.0</td>
<td>10.7</td>
<td>37.7</td>
<td>38.5</td>
<td>20.1</td>
<td>26.7</td>
<td>38.3</td>
<td>58.4</td>
<td>60.0</td>
<td></td>
</tr>
<tr>
<td>JAA[Net [17]</td>
<td>43.7</td>
<td>46.2</td>
<td>56.0</td>
<td>41.4</td>
<td>44.7</td>
<td>69.6</td>
<td>88.3</td>
<td>58.4</td>
<td>56.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAA[Net [18]</td>
<td><strong>62.4</strong></td>
<td><strong>60.7</strong></td>
<td>67.1</td>
<td><strong>41.1</strong></td>
<td>45.1</td>
<td>73.5</td>
<td>90.9</td>
<td><strong>67.4</strong></td>
<td>63.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSIN [3]</td>
<td>42.4</td>
<td>39.0</td>
<td>68.4</td>
<td>28.6</td>
<td>46.8</td>
<td>70.8</td>
<td>90.4</td>
<td>42.2</td>
<td>53.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRERL [11]</td>
<td>45.7</td>
<td>47.8</td>
<td>59.6</td>
<td>47.1</td>
<td>45.6</td>
<td>73.5</td>
<td>84.3</td>
<td>43.6</td>
<td>55.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UGN-B [19]</td>
<td>43.3</td>
<td>48.1</td>
<td>63.4</td>
<td>49.5</td>
<td>48.2</td>
<td>71.9</td>
<td>90.8</td>
<td>59.0</td>
<td>60.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TransAU [8]</td>
<td>46.1</td>
<td>48.6</td>
<td><strong>72.8</strong></td>
<td><strong>56.7</strong></td>
<td>[50.0]</td>
<td>72.1</td>
<td>90.8</td>
<td>55.4</td>
<td>61.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONET [21]</td>
<td>55.8</td>
<td>[60.4]</td>
<td>68.1</td>
<td>[49.8]</td>
<td>48.0</td>
<td>[73.7]</td>
<td>[92.3]</td>
<td>[63.1]</td>
<td><strong>63.9</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>FAN-Trans</strong></td>
<td>[56.4]</td>
<td><strong>50.2</strong></td>
<td>[68.6]</td>
<td>49.2</td>
<td><strong>57.6</strong></td>
<td><strong>75.6</strong></td>
<td><strong>93.6</strong></td>
<td>58.8</td>
<td>[63.8]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sensitivity to hyper-parameter \( \lambda \) Basically, we test 5 weights for \( \lambda \): 0.05, 0.1, 0.2, 0.5 and 1. Table 4 illustrates the impact of hyper-parameter \( \lambda \) in the proposed framework. From these results, we can see that our method is affected by the weights to balance the distillation loss and the classification loss. The too small or too large value will deteriorate the performance gains. Thus, a grid search technique contributes to the best model.

Finally, the best configuration is \( C_1, C_{o2a}, C_{o2m} \) by setting \( \lambda \) as 0.2 is FAN-Trans.

5. Conclusion and Discussion

**Conclusion** In this work, we propose FAN-Trans for facial AU detection which can learn representative AU features and correlations among both AU features and AU labels in an OKD framework. FAN-Trans takes advantage of the multi-scale face alignment feature maps to learn AU features from AU active regions with heatmap attention. It uses transformer to model AU co-occurrences with a learnable binary mask to drop self attention in order to discriminate different AU features. It customizes OKD with diverse classifiers designed for AU detection. Experiments show its advantages over SOTA methods.

**Discussion** FAN-Trans is built on a pre-trained face alignment model for extraction of AU features. Thus the performance of face alignment will influence the model performance. Besides, as is show in Figure 1, although the face alignment features provide a strong representation and the attention area is roughly around key points, it is still hard to assign them very precisely without manual supervision. Exploring how to learn individual AU attentions associated with landmarks is a promising direction.
References


