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

Jing Yang [1]
y.jing2016@gmail.com

Jie Shen [2]
jie.shen07@imperial.ac.uk

Yiming Lin [2]
yl1915@ic.ac.uk

Yordan Hristov
yshristov@gmail.com

Maja Pantic[2]
maja.pantic@gmail.com

[1]University of Nottingham, UK [2]Imperial 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.

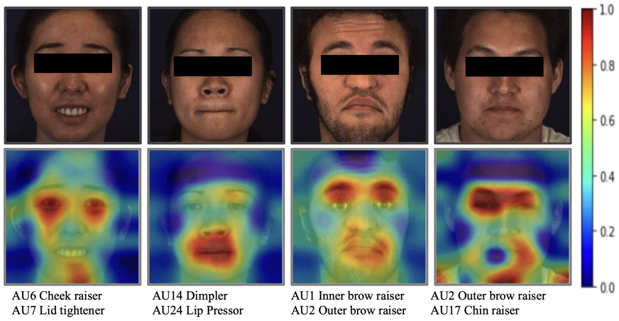


Figure 1. Visualization of attention maps of FAN-Trans via Score-CAM [26]. The result is achieved by placing the target layer on resolution 4×4. From top to bottom, each row denotes an input image, its spatial attention and activated action units, in respect. FAN-Trans shows a potential high concentration at the facial component that indicates its ability to learn where to focus without using explicit attention modules [8] or manual region allocations [22, 11]. For example, AU1 and AU2 emphasize the eyebrow, AU7 is around the eye while AU24 highlight the mouth.

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

† Corresponding author.

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 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-

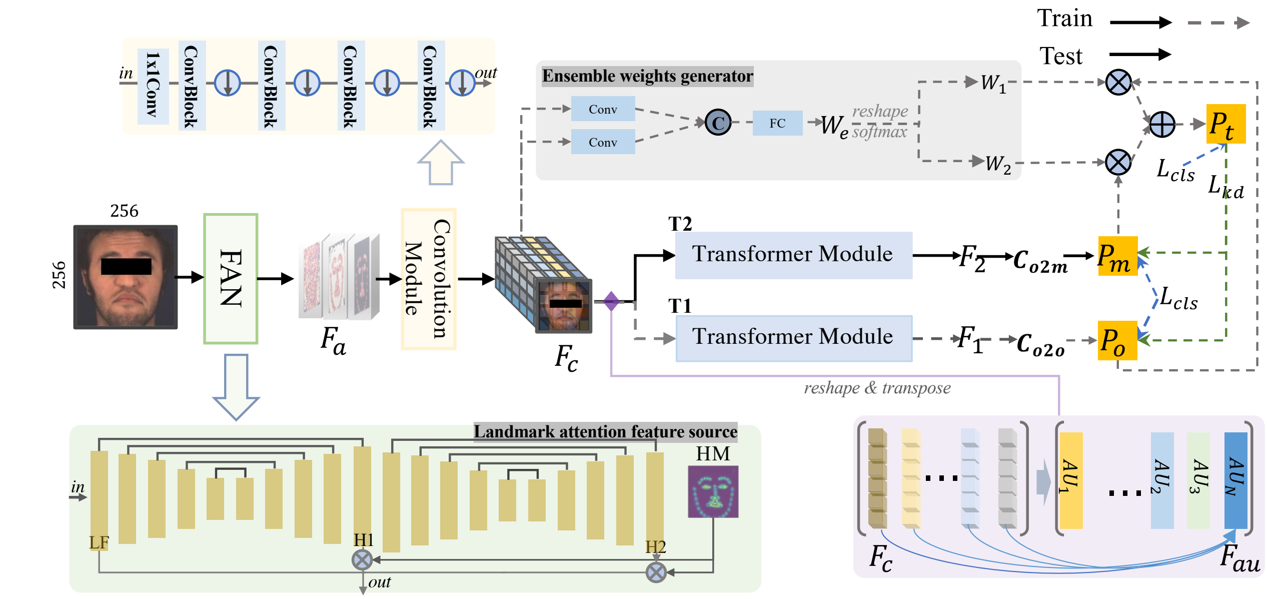


Figure 2. Overview of FAN-Trans. In the training stage, firstly, a face image is fed into a pre-trained face alignment model to extract feature maps $\mathbf{F}_a$. Secondly, $\mathbf{F}_a$ is passed through a convolution module to learn compact feature representation $\mathbf{F}_c$. Thirdly, $\mathbf{F}_c$ is reshaped and transposed to a sequence of vectors, and is further projected into a set of AU features $\mathbf{F}_{au}$ by a linear transformation. $\mathbf{F}_{au}$ is fed into two transformer branches with diverse classifiers C_{o2o} and C_{o2m} to get predictions $\mathbf{P}_o$ and $\mathbf{P}_m$. The ensemble target $\mathbf{P}_t$ is a weighted (W_1, W_2) combination of $\mathbf{P}_o$ and $\mathbf{P}_m$. Finally, all learnable parameters are optimized by three classification losses $\mathcal{L}_{cls}$ and two knowledge distillation losses $\mathcal{L}_{kd}$. Note that only a single pathway remains at inference time. All auxiliaries modules enclosed in dashed lines are discarded after training. By doing so, regional feature learning, AU-occurrences modeling, efficient training technique are integrated into one end-to-end trainable pipeline.

pendencies. Others [11, 8] concentrate on learning semantic correlation among AUs. For example, SRERL [11] fed AU features to Gated Graph Neural Network built on defined knowledge-graph in an offline manner. More recently, TransAU [8] introduced transformer [5] into the facial AU detection task due to its particular efficacy in capturing dependencies among distant patches. Our method uses the transformer but we go a step further by discriminating AU features with a learnable drop attention in self-attention module. Besides, our transformer network is trained with the automatically learned AU features instead of cropped features from pre-defined ROIs.

Online knowledge distillation Different from vanilla KD [7], which uses a pre-trained high capacity teacher network to guide the learning of a low-capacity student, there is no explicit teacher network in OKD. One category of OKD is to train with the ensemble output of the students sharing similar network configurations. For example, ONE [10] was the pioneer who constructed a single multi-branch network to let each branch learn from the ensemble distribution in the classification task. This idea was extended in the same or heterogeneous settings and applied in the human pose estimation task [12]. Our model is thus an instantiation of OKD in the specific context of AU recognition. We further maximize the capacity of OKD by increasing the divergence of peers.

3. FAN-Trans

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 variations from pose, illumination, color as the model was pretrained on a large variety of facial poses; **(3)** closely align with AU detection — having a good localization of the AU region correlates with higher AU accuracy.

As illustrated in Figure 2, we obtain source of AU features by combining intermediate features (LF, H1, H2) and heat-maps (HM) from the pre-trained FAN. Specially, we first aggregate heat-maps to a single plane, then multiply it with penultimate layer's outputs (H1,H2), and finally concatenate these high level features with the low-level features (LF) to obtain $\mathbf{F}_a \in \mathbb{R}^{D_a \times H_a \times W_a}$. We use "a" as a subscript to represent the features from FAN. $\mathbf{F}, D, H, W$ 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 $\mathbf{F}_a$ and transformer [29, 13] is to learn AU co-occurrences in feature space.

As illustrated in Figure 2, the convolution module is composed of several convolution layers: one 1x1 convolution layer to decrease the channel dimension from D_a to $0.25D_a$; 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 64×64 to 4×4); Second, it reduces the computational complexity by decreasing feature dimension as self-attention in transformer operates across sequential tokens by drawing pairwise interactions. Given a input tensor $\mathbf{F}_a$, the output of convolution module is $\mathbf{F}_c \in \mathbb{R}^{D_c \times H_c \times W_c}$.

Furthermore, we implicitly allocate per AU embedding with a linear transformation. We first flatten 3D $\mathbf{F}_c$ across spatial dimension to 2D tensor $\mathbf{F}_c \in \mathbb{R}^{D_c \times H_c W_c}$, then apply a linear layer on the 2rd dimension to transform $H_c W_c$ to number of AUs N_{au}. This is formulated by:

$$\mathbf{F}_{au} = \mathbf{F}_c \times \mathbf{W}_{au}, \tag{1}$$

where $\mathbf{W}_{au} \in \mathbb{R}^{H_c W_c \times N_{au}}$, and $\mathbf{F}_{au} \in \mathbb{R}^{D_c \times N_{au}}$. By doing so, per AU feature is associated with $\mathbf{F}_c$ with learnable parameters optimized by the task loss. Every vector $D_c \times 1$ 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 $\mathbf{F}_c$. 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 $\mathbf{F}_{au} \in \mathbb{R}^{D_c \times N_{au}}$ to obtain $\mathbf{F}_{au} \in \mathbb{R}^{\in N_{au} \times D_c}$ as transformer here is to modeling connections among AU features. For MHSA module, linear layers are applied on $\mathbf{F}_{au}$ to obtain queries $\mathbf{Q} \in \mathbb{R}^{N_{au} \times D_q}$, keys $\mathbf{K} \in \mathbb{R}^{N_{au} \times D_k}$ and values $\mathbf{V} \in \mathbb{R}^{N_{au} \times D_v}$ ($D_q == D_k$), then the attention map $\mathbf{A}$ is calculated through:

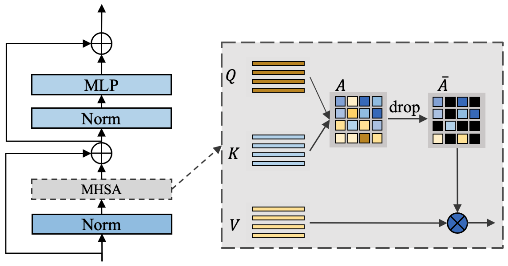


Figure 3. Encode block (transblock) in transformer module. We assume $N_{au} = 4$. For a 4x4 attention map $\mathbf{A}$, we explore different attention drop mechanisms on $\mathbf{A}$ to find the best choice $\overline{\mathbf{A}}$ for facial AU detection task.

$$\mathbf{A} = \mathrm{softmax}(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{D_q}}), \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{N_{au} \times N_{au}}$ with $\mathbf{A}[i,:]$ representing $\mathbf{Q}[i,:]$'s correlation with $\mathbf{K}$, and $\mathbf{A}[:,i]$ representation $\mathbf{Q}$'s correlation with $\mathbf{K}[i,:]$. We further propose a learnable attention drop mechanism by multiplying $\mathbf{A}$ with learned binary mask $\mathbf{M} \in \mathbb{R}^{N_{au} \times 1}$ to drop some AUs connections with others.

$$\overline{\mathbf{A}} = \mathbf{M} * \mathbf{A}. \tag{3}$$

where $\overline{\mathbf{A}}_{i,j} = \mathbf{M}_j \mathbf{A}_{i,j}$. This 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 $\mathbf{F}_{au}$ into two diverse peers ($\mathbf{T1}$ and $\mathbf{T2}$). We denote the outputs of $\mathbf{T1}$ and $\mathbf{T2}$ as $\mathbf{F}_1 \in \mathbb{R}^{N_{au} \times D_t}$ and $\mathbf{F}_2 \in \mathbb{R}^{N_{au} \times D_t}$, respectively. We design the diverse classifiers as follows:

One-to-One classifier (C_{o2o})**:** We pass $\mathbf{F}_1$ through a fully connected layer $\mathbf{W}_o \in \mathbb{R}^{D_t \times 1}$, and the prediction is computed as

$$\mathbf{P}_o = \mathbf{F}_1 \times \mathbf{W}_o, \tag{4}$$

where $\mathbf{P}_o \in \mathbb{R}^{N_{au} \times 1}$ with i-th element $\mathbf{F}_1[i]$ predicting i-th AU's activation.

One-to-Many classifier (C_{o2m})**:** We multiply $\mathbf{F}_2$ with a transformation matrix $\mathbf{W}_m \in \mathbb{R}^{D_t \times N_{au}}$, and the prediction is calculated as

$$\mathbf{P}_m = \mathrm{TopK}(\mathbf{F}_2 \times \mathbf{W}_m), \tag{5}$$

where $\mathbf{P}_m \in \mathbb{R}^{1\times N_{au}}$. The output of $\mathbf{F}_2 \times \mathbf{W}_m$ is $\mathbf{P}_{\hat{m}} \in \mathbb{R}^{N_{au}\times N_{au}}$ and TopK is a selection function. The i-th slice of $\mathbf{F}_2[i,:]$ predicts the existence of all AUs. $\mathbf{P}_{\hat{m}}[:,i]$ denotes the i-th AU's evidence from all AU features. TopK operation summarizes the highest k confidence in $\mathbf{P}_{\hat{m}}[:,i]$ to get the final prediction for i-th AU:

$$\mathbf{P}_m[i] = \sum_{j=0}^{K_i} \mathbf{P}_{\hat{m}}[:,j], \tag{6}$$

where K_i denotes the top-k index per i-th column. The illustration of $\mathbf{P}_m$'s calculation is in Figure 4.

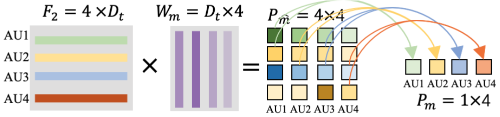


Figure 4. Steps for calculating $\mathbf{P}_m$. We assume $N_{au} = 4$ for illustration purposes. $\mathbf{F}_2[i,:]$ and $\mathbf{W}_m[:,i]$ denote i-th AU's feature and i-th AU's prototype, respectively. Hence, i-th AU's feature ($\mathbf{F}_2[i,:]$) contributes to the logits of all AUs ($\mathbf{P}_{\hat{m}}[i,:]$) and all AUs' features contribute to the logits of j-th AU ($\mathbf{P}_{\hat{m}}[:,j]$). To avoid noise accumulation, $\mathbf{P}_m[i]$ is aggregated (summed) from the highest k (k=2) values in $\mathbf{P}_{\hat{m}}[:,i]$.

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 $\mathbf{F}_c$. 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 $\mathbf{W}_e \in \mathbb{R}^{2\times N_{au}}$. Afterwards, the Softmax is used to normalize the weights. Finally, the ensemble weights $\mathbf{W}_e$ are split into $\mathbf{W}_1 \in \mathbb{R}^{1\times N_{au}}$ and $\mathbf{W}_2 \in \mathbb{R}^{1\times N_{au}}$ and the ensemble target is computed by element-wise summation:

$$\mathbf{P}_t = \mathbf{W}_1 * \mathbf{P}_o + \mathbf{W}_2 * \mathbf{P}_m^\top. \tag{7}$$

The OKD is formulated as the Kullback-Leibler (KL) divergence loss between $\mathbf{P}_t$ and the student's prediction ($\mathbf{P}_* \subseteq \{\mathbf{P}_o, \mathbf{P}_m\}$):

$$\mathcal{L}_{KD} = \mathrm{KL}(\mathbf{P}_*, \mathbf{P}_t). \tag{8}$$

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

$$\mathcal{L}_{cls} = -\sum_{i}^{N_{au}} w_i[y_i \log p_i + (1-y_i)\log(1-p_i))], \tag{9}$$

where p_i denotes i-th AU's prediction, which has three sources $\mathbf{P}_o$, $\mathbf{P}_m$, and $\mathbf{P}_t$. 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:

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \lambda\mathcal{L}_{kd}, \tag{10}$$

in which λ 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 JÂANet [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 pretrained and fixed while other learnable parameters are optimized using AdamW with hyper-parameters $\beta_1 = 0.9$ and

Methods	AU1	AU2	AU4	AU6	AU7	AU10	AU12	AU14	AU15	AU17	AU23	AU24	AVG
Baseline	51.6	39.0	60.0	72.8	79.0	79.1	87.2	62.0	49.3	58.1	48.9	51.1	61.5
C_T_C_{o2o}	55.8	44.9	56.9	77.8	75.6	82.8	87.5	61.3	48.7	61.9	48.2	52.5	62.8
C_T_C_{o2m}	52.7	47.0	56.8	75.0	75.3	82.2	88.0	63.1	51.9	64.3	49.5	53.7	63.3
C_T1_C_{o2o}_T1_C_{o2m}	55.7	42.5	60.9	76.4	76.7	83.6	86.6	62.4	47.4	65.8	49.5	57.9	63.8
C_T1_C_{o2o}_T2_C_{o2o}	55.1	46.8	58.8	77.5	74.7	83.4	87.2	63.4	48.9	65.9	50.2	56.9	64.1
C_T1_C_{o2m}_T2_C_{o2m}	58.0	46.8	59.9	76.5	76.0	83.6	87.3	60.3	49.9	66.1	49.9	56.6	64.2
C_T1_C_{o2o}_T2_C_{o2m}	55.4	46.0	59.8	78.7	77.7	82.7	88.6	64.7	51.4	65.7	50.9	56.0	**64.8**
C_T1_C_{o2o}_T2_C_{o2m}_F	49.4	46.0	61.2	77.4	77.6	83.0	88.4	65.9	50.3	63.6	50.3	52.2	63.8
C_T1_C_{o2o}_T2_C_{o2m}_R	51.8	44.8	58.7	77.6	77.4	82.5	87.6	64.7	51.7	66.0	49.7	56.2	64.1
C_T1_C_{o2o}_T2_C_{o2m}_C	54.0	43.1	60.3	77.2	78.1	84.3	86.7	63.2	51.6	65.0	48.5	56.9	64.1

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.

$\beta_2 = 0.999$ 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 λ 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 $\mathbf{F}_c$ 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} and C_T_C_{o2m} respectively. At this point, no knowledge distillation part is included.

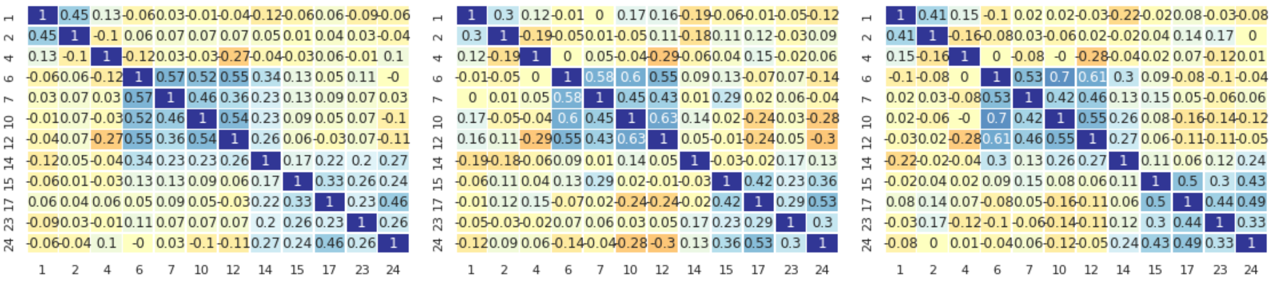
The results are in Table 1. Compared with Baseline, both C_T_C_{o2o} and C_T_C_{o2m} get higher F1-score in the majority of AUs with obvious improvement on AU2, AU6, AU17, AU23. Overall, in terms of the average F1-score, C_T_C_{o2o} and C_T_C_{o2m} exceeds Baseline by 1.3% and 1.8% separately. We believe 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 representation ability of FAN features.

Effectiveness of classifiers C_T_C_{o2m} outperforms C_T_C_{o2o} by 0.5% in average F1-score. We argue C_T_C_{o2m} is superior to C_{o2o} 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 is captured by C_T_C_{o2m} but ignored in C_T_C_{o2o}. We assume that the C_{o2m} is sufficient in detecting the lower part of AUs because AU14 fosters inter connections. In addition, we compute the element-wise Euclidean distance between correlation matrices for more direct comparison. The distance between labels and predictions from C_{o2o} is 0.012 while it decreases to 0.008 for C_{o2m}, which further confirms the ability of C_{o2m} in learning AU associations from label statistics.

Improvement from OKD training We first test a variant C_T1_C_{o2o}_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} and 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}_T2_C_{o2o}, (II) C_T1_C_{o2m}_T2_C_{o2m}, (III) C_T1_C_{o2o}_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_{o2o} 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



(a) ground-truth labels

(b) predictions of C_T_C_{o2o}

(c) predictions of C_T_C_{o2m}

Figure 5. Correlation maps of (a) ground-truth labels, (b) predictions of C_T_C_{o2o} and (c)predictions of C_T_C_{o2m}, where each entry (i,j) is computed as the coefficient correlation between the i-th AU and the j-th AU.

spectively because the same feature descriptor hurts the diversity. This finding is supported by the analysis of branch diversity proposed in [12, 2].

From the results, we observe that (I) improves C_T1_C_{o2o} from 62.8% to 64.1% while (II) improves C_T1_C_{o2m} from 63.3% to 64.2%. Both of them are comparable to the SOTA performance (see Table 5). It is worth mentioning here that this is the *first* time that OKD is deployed on facial AU detection task, and it obtains significant performance gains.

(III) achieves the best performance. More concretely, the output of C_{o2o} in (III) gets 64.7% in average F1-score, the output of C_{o2m} in (III) achieves 64.8% as well as P_t. The superior performance shows that designing two branches with different classifiers boosts the peer diversity, which is effective in OKD [2]. We set default branch as 2 because we leveraged two different classifiers. We have tried increasing the branch to three which obtains another 0.1% gain.

Effectiveness of attention drop in Transformer In the following experiment, we dig into how the attention map influences the AU features. We consider four variants: (a) C_T1_C_{o2o}_T2_C_{o2m}_F, (b) C_T1_C_{o2o}_T2_C_{o2m}_R, (c) C_T1_C_{o2o}_T2_C_{o2m}_C and (d) C_T1_C_{o2o}_T2_C_{o2m}. The description is in Figure 3. In (a), a full attention map is used, which means all elements in $\mathbf{A}$ will be multiplied with $\mathbf{V}$ to produce new AU features. (b) deactivates the lower similarity in each row of $\mathbf{A}$. (c) deactivates the lower similarity in each column of $\mathbf{A}$. (d) learns a binary mask. As is shown in Table 1, (a), (b) and (c) improve the average F1-score of C_T_C_{o2m} from 63.3% to 63.8%, 61.4% and 64.1% respectively. It demonstrates that all attention mechanisms benefit the OKD framework. (b) and (c) outperforming (a) may be caused by the alleviation of over-fitting in the full attention map. The learnable attention drop stands out by increasing the F1-score in C_T_C_{o2m} from 63.3% to 64.8%.

Figure 6 illustrates the distribution of the pairwise cosine distance among $\mathbf{F}_{au}$. By comparison, the average similarity is decreased from 0.738 (a) to 0.566 (d). It shows attention drop will increase diversity among different AU representations.

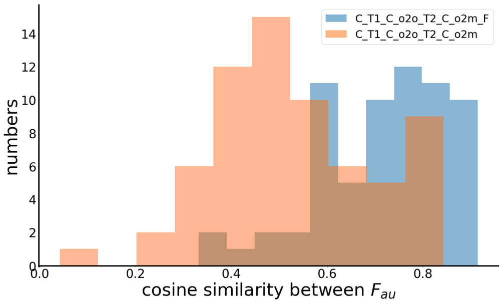


Figure 6. The pairwise cosine similarity distribution among AU features $\mathbf{F}_{au}$ before C_{o2m} for variant (a) and (d).

Ms	BL	BL-FAN+R18	BL-LF	BL-HM
F1-score	61.5	59.5	59.2	60.8

Table 2. Impact of FAN feature on BP4D (F1-score in %).

Encode No.	1	3	5	7	9
F1-score	63.8	64.4	64.8	**64.9**	64.1

Table 3. Impact of No. of TransBlock on BP4D (F1-score in %).

λ	0.05	0.1	0.2	0.5	1
F1-score	64.1	64.3	**64.8**	63.9	63.8

Table 4. Sensitivity to λ for FAN-Trans on BP4D (F1-score in %).

Impact of FAN feature We compare four variants in Table 2: BL is the baseline method; BL-FAN+R18 replaces FAN in BL with feature map of same resolution (64x64) from ResNet18 (after 2rd layer) pretrained on ImageNet; BL-LF is BL without LF; BL-HW is BL without the product of heatmap. Based on results, we conclude features pretrained on the alignment task are superior to generic features from a classification model. Additionally, although HM is inferior to LF for overall contribution, it complements LF by giving more attention around landmarks which are recognized as AU active areas.

Encode numbers in transformer module We set encode number in the transformer module as 5 in the main experiment to keep balance with convolution operations. Table 3 presents the performance influenced by the transBlock number in the transformer module on the BP4D. We test 5 cases with encode numbers as 1,3,5,7,9 respectively. From this table, we can see that if very few encodes are utilized, the performance will drop quickly. This is caused by the limited learning capacity in the transformer module. Be-

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

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.

Methods	AU1	AU2	AU4	AU6	AU9	AU12	AU25	AU26	AVG
DRML [35]	17.3	17.7	37.4	29.0	10.7	37.7	38.5	20.1	26.7
JAANet [17]	43.7	46.2	56.0	41.4	44.7	69.6	88.3	58.4	56.0
JÂANet [18]	**62.4**	**60.7**	67.1	41.1	45.1	73.5	90.9	**67.4**	63.5
DSIN [3]	42.4	39.0	68.4	28.6	46.8	70.8	90.4	42.2	53.6
SRERL [11]	45.7	47.8	59.6	47.1	45.6	73.5	84.3	43.6	55.9
UGN-B [19]	43.3	48.1	63.4	49.5	48.2	71.9	90.8	59.0	60.0
TransAU [8]	46.1	48.6	**72.8**	**56.7**	[50.0]	72.1	90.8	55.4	61.5
MONET [21]	55.8	[60.4]	68.1	[49.8]	48.0	[73.7]	[92.3]	[63.1]	**63.9**
FAN-Trans	[56.4]	50.2	[68.6]	49.2	**57.6**	**75.6**	**93.6**	58.8	[63.8]

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.

sides, when the number of encoding increases to 9, the performance will also drop because of too much flexibility within the two branches. Although encode number 7 gets a slightly better F1-score than encode number 5, considering the increased model complexity, we still use encode number 5 in our final model.

Sensitivity to hyper-parameter λ Basically, we test 5 weights for λ: 0.05, 0.1, 0.2, 0.5 and 1. Table 4 illustrates the impact of hyper-parameter λ 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 C_T1_C_{o2o}_T2_C_{o2m} by setting λ as 0.2 is FAN-Trans.

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*. JPML [34], DRML [35], JAANet [17], JÂANet [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.

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 to assign them very precisely without manual supervision. Exploring how to learn individual AU attentions associated with landmarks is a promising direction.

References

[1] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *ICCV*, 2017.

[2] Defang Chen, Jian-Ping Mei, Can Wang, Yan Feng, and Chun Chen. Online knowledge distillation with diverse peers. In *AAAI*, 2020.

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

[4] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *CVPR*, 2020.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020.

[6] Paul Ekman and Erika L Rosenberg. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.

[7] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv*, 2015.

[8] Geethu Miriam Jacob and Bjorn Stenger. Facial action unit detection with transformers. In *CVPR*, 2021.

[9] László A Jeni, Jeffrey F Cohn, and Fernando De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine association conference on affective computing and intelligent interaction*, 2013.

[10] Xu Lan, Xiatian Zhu, and Shaogang Gong. Knowledge distillation by on-the-fly native ensemble. In *NeurIPS*, 2018.

[11] Guanbin Li, Xin Zhu, Yirui Zeng, Qing Wang, and Liang Lin. Semantic relationships guided representation learning for facial action unit recognition. In *AAAI*, 2019.

[12] Zheng Li, Jingwen Ye, Mingli Song, Ying Huang, and Zhigeng Pan. Online knowledge distillation for efficient pose estimation. In *ICCV*, 2021.

[13] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021.

[14] S Mohammad Mavadati, Mohammad H Mahoor, Kevin Bartlett, Philip Trinh, and Jeffrey F Cohn. Disfa: A spontaneous facial action intensity database. *TAC*, 2013.

[15] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[16] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[17] Zhiwen Shao, Zhilei Liu, Jianfei Cai, and Lizhuang Ma. Deep adaptive attention for joint facial action unit detection and face alignment. In *ECCV*, 2018.

[18] Zhiwen Shao, Zhilei Liu, Jianfei Cai, and Lizhuang Ma. Jaanet: joint facial action unit detection and face alignment via adaptive attention. *IJCV*, 2021.

[19] Tengfei Song, Lisha Chen, Wenming Zheng, and Qiang Ji. Uncertain graph neural networks for facial action unit detection. In *AAAI*, 2021.

[20] Wenyu Song, Shuze Shi, and Gaoyun An. Facial action unit detection based on transformer and attention mechanism. In *ICIG*, 2021.

[21] Gauthier Tallec, Arnaud Dapogny, and Kevin Bailly. Multi-order networks for action unit detection. *TAC*, 2022.

[22] Yang Tang, Wangding Zeng, Dafei Zhao, and Honggang Zhang. Piap-df: Pixel-interested and anti person-specific facial action unit detection net with discrete feedback learning. In *ICCV*, 2021.

[23] Antoine Toisoul, Jean Kossaifi, Adrian Bulat, Georgios Tzimiropoulos, and Maja Pantic. Estimation of continuous valence and arousal levels from faces in naturalistic conditions. *Nature Machine Intelligence*, 2021.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[25] Robert Walecki, Vladimir Pavlovic, Björn Schuller, Maja Pantic, et al. Deep structured learning for facial action unit intensity estimation. In *CVPR*, 2017.

[26] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *CVPRW*, 2020.

[27] Ziheng Wang, Yongqiang Li, Shangfei Wang, and Qiang Ji. Capturing global semantic relationships for facial action unit recognition. In *ICCV*, 2013.

[28] Jing Yang, Adrian Bulat, and Georgios Tzimiropoulos. Fanface: a simple orthogonal improvement to deep face recognition. In *AAAI*, 2020.

[29] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Keypoint localization via transformer. In *ICCV*, 2021.

[30] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, 2019.

[31] Linfeng Zhang, Zhanhong Tan, Jiebo Song, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Scan: A scalable neural networks framework towards compact and efficient models. *NeurIPS*, 2019.

[32] Xing Zhang, Lijun Yin, Jeffrey F Cohn, Shaun Canavan, Michael Reale, Andy Horowitz, Peng Liu, and Jeffrey M Girard. Bp4d-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. *IVC*, 2014.

[33] Yong Zhang, Weiming Dong, Bao-Gang Hu, and Qiang Ji. Classifier learning with prior probabilities for facial action unit recognition. In *CVPR*, 2018.

[34] Kaili Zhao, Wen-Sheng Chu, Fernando De la Torre, Jeffrey F Cohn, and Honggang Zhang. Joint patch and multi-label learning for facial action unit detection. In *CVPR*, 2015.

[35] Kaili Zhao, Wen-Sheng Chu, and Honggang Zhang. Deep region and multi-label learning for facial action unit detection. In *CVPR*, 2016.