PAttNet: Patch-attentive deep network for action unit detection

Itir Onal Ertugrul[1]
iertugru@andrew.cmu.edu
Laszlo A. Jeni[1]
laszlojeni@cmu.edu
Jeffrey F. Cohn[2]
jeffcohn@pitt.edu

[1] Robotics Institute
Carnegie Mellon University
Pittsburgh, PA, USA

[2] Department of Psychology
University of Pittsburgh
Pittsburgh, PA, USA

Abstract

Facial action units (AUs) refer to specific facial locations. Recent efforts in automatic AU detection have focused on learning their representations. Two factors have limited progress. One is that current approaches implicitly assume that facial patches are robust to head rotation. The other is that the relation between patches and AUs is pre-defined or ignored. Both assumptions are problematic. We propose a patch-attentive deep network called PAttNet for AU detection that learns mappings of patches and AUs, controls for 3D head and face rotation, and exploits co-occurrence among AUs. We encode patches with separate convolutional neural networks (CNNs) and weight the contribution of each patch to detection of specific AUs using a sigmoid patch attention mechanism. Unlike conventional softmax attention mechanisms, a sigmoidal attention mechanism allows multiple patches to contribute to detection of specific AUs. The latter is important because AUs often co-occur and multiple patches may be needed to detect them reliably. On the BP4D dataset, PAttNet improves upon state-of-the-art by 3.7%. Visualization of the learned attention maps reveal power of this patch-based approach.

1 Introduction

Facial action is a non-verbal way to communicate intention, emotion and physical state [20]. The most comprehensive method to annotate facial action is the anatomically-based Facial Action Coding System (FACS). Action units defined in FACS correspond to facial muscle movements that individually or in combination can describe nearly all possible facial expressions. Automated detection of AUs has become a crucial computer vision problem.

The human face is more structured than many other natural images and different face regions have different local statistics [26]. Variation in local statistics stems from both structural features and transient facial muscle contraction and relaxation. Facial action units (AUs) correspond to anatomically-based muscle contractions that result in deformations in appearance. For instance, tightening of the eye aperture results from contraction of the inner portion of the orbicularis oculi muscle, which is AU7. For this reason, some facial regions are more important than others to detect specific AUs [25]. These observations have motivated region learning, specifically *patch learning*, for AU detection.

Patches have been defined in one of two principal ways. One is with respect to fixed grids [11]. The other is centered around facial landmarks [25]. Both approaches assume that patches are invariant to head rotation. That is, when the head moves or rotates, patches are assumed to maintain consistent semantic correspondence. This assumption often is violated. Faces look very different from different poses. Because most registration techniques treat the face as a 2D object, they are unable to accommodate 3D head rotation.

Another problem is that mappings between AUs and patches are defined a priori, and the mappings often fail to exploit co-occurrences among AUs. Some AUs frequently co-occur, while others decrease the activity of others. AU6 (cheek raiser) and AU12 (oblique lip-corner puller) occur together in Duchenne smiles and in pain expressions. AU24, which presses the lips together, inhibits dropping of the jaw (AU27). Appearance changes in different facial regions are likely to contribute to the prediction of co-occurring AUs. For this reason, it is important to weight the significance of patches to detection of specific AUs. While some patch-based AU detection methods do not focus on weighting the contribution of each patch [26], a few of them aim to select informative regions using regularization on the shallow representation of patches [25] or using pre-defined attention masks in CNN [7, 16] often ignoring AU correlations. In none of these approaches are attention maps learned.

We propose a patch-attentive deep network for AU detection, called PAttNet, that jointly learns patch representations and weights them for AU detection. We first apply 3D registration to reduce changes from head movement and preserve facial actions that would be distorted by change in pose. Then, we crop local patches that contain the same facial parts across frames and that are informative for detection of specific AUs. We encode patches with individual CNNs and obtain local representations. Inspired by the recent success of attention mechanisms in various tasks including neural machine translation [13], text classification [22], object detection [15], we introduce an attention mechanism to weight the importance of patches in detecting specific AUs. Since our network is trained in an end-to-end manner, the network itself learns i) encoding of patches and ii) the degree of attention to those patches to maximize AU detection. Unlike the state-of-the-art attention approaches, which employ softmax activation function to 'select' where to attend, we propose sigmoidal attention to allow networks to attend to multiple patches when needed.

The contributions of this paper are:

- 3D face registration of the input images to eliminate distortion from 3D rigid motion, which ensures that cropped patches contain the same aligned facial regions across frames.
- An end-to-end trainable patch-attentive deep network that learns to attend to specific patches for the detection of specific AUs.
- A sigmoidal attention mechanism that allows multiple patches to contribute to the prediction of specific AUs.
- Relative to state of the art, an increase of 3.7 % performance in F1-score.

2 Related Work

Patch learning: Traditional AU detection methods are based on i) extracting appearance [1, 6, 9] or geometric features [4, 12] from the whole face and ii) obtaining shallow representations as histograms of these features, thus ignoring the specificity of facial parts to AUs

[18]. Deep approaches using whole face to train CNNs [14] also ignore the specificity of facial parts. More recent approaches focus on obtaining local representations using *patch learning*. Some of these approaches divide the face image into uniform grids [11, 26, 27] while others define patches around facial parts [3] or facial landmarks [25]. Among them, Liu *et al.* [11] divide a face image into non-overlapping patches and categorize them into common and specific patches to describe different expressions. Zhong *et al.* [27] identify active patches common to multiple expressions and specific to an individual expression using a multi-task sparse learning framework. Zhao *et al.* [26] use a regional connected convolutional layer that learns specific convolutional filters from sub-areas of the input. Corneanu *et al.* [3] crop patches containing facial parts, train separate classifiers for each part and fuse the decisions of classifiers using structured learning. Zhao *et al.* [25] describe overlapping patches centered at facial landmarks, obtain shallow representations of patches and identify informative patches using a multi-label learning framework. These studies generally preprocess their frames to remove roll rotation. None of the aforementioned studies perform a 3D face registration to remove pitch and yaw rotation. Hence, patches cropped from different frames are likely to contain variable facial regions under pose.

Regional attention: As described in FACS [5], AUs relate to specific regions of human faces. Motivated by this fact, recent studies aim to highlight information obtained from specific facial regions to detect specific AUs. Zhao *et al.* [25] employ patch regularization to eliminate the effect of non-informative shallow patch representations. Taheri *et al.* [19] learn a dictionary per AU using local features extracted from predefined AU semantic regions on faces performing that AU. Jaiswal *et al.* [7] use a pre-defined binary mask created to select a relevant region for a particular AU and pass it to a convolutional and bi-directional Long Short-Term Memory (LSTM) neural network. Li *et al.* [10] design an attention map using the facial key points and AU centers to enforce their CNN-based architecture to focus more on these AU centers. Sanchez *et al.* [16] generate heatmaps for a target AU, by estimating the facial landmarks and drawing a 2D Gaussian around the points where the AU is known to cause changes. They train Hourglass network to estimate AU intensity. Shao *et al.* [17] employ an initial attention map, created based on AU centers and refine it to jointly perform AU detection and face alignment. These studies have mechanisms to enforce their models to focus on pre-defined regions. They do not have a learned attention mechanism, in which the network decides where to attend itself for each AU.

3 Method

Figure 1 shows the components of the proposed PAttNet architecture. First, we perform dense 3D registration from 2D videos (a). Then, we crop patches containing local facial parts and encode each patch using a separate CNN architecture (b). We employ a sigmoidal attention mechanism to weight the contribution of each patch to detect specific AUs (c). Finally, using the final face encoding, we detect 12 AUs (d). In the following, we describe in detail, the different components of the proposed PAttNet approach.

3.1 3D Face Registration

We track and normalize videos using ZFace [8], a real-time face alignment software that accomplishes dense 3D registration from 2D videos and images without requiring person-specific training. ZFace performs a canonical 3D normalization that minimizes appearance

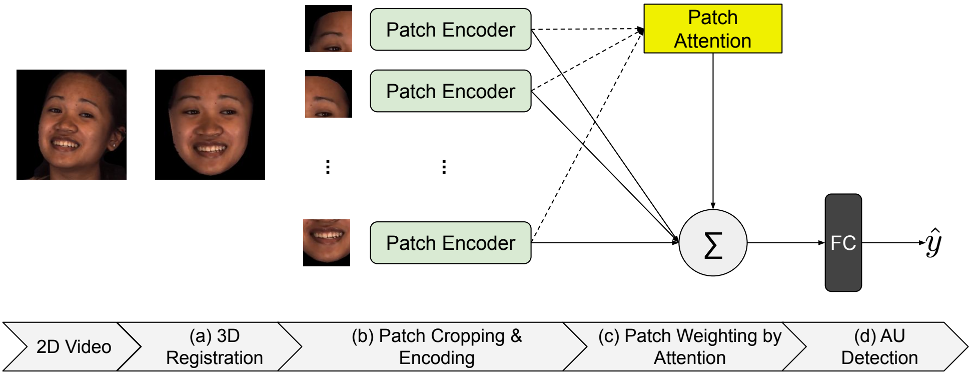

Figure 1: Proposed PAttNet approach. (a) A dense set of facial landmarks is estimated and a dense 3D mesh of the face is reconstructed. (b) Patches containing facial regions related to specific AUs are cropped and fed to different CNNs for encoding. (c) Patches are weighted by sigmoidal attention mechanism to detect specific AUs. (d) Face encodings are fed to a fully connected layer (FC) to detect AUs.

changes from head movement and maximizes changes from expressions. First, it uses dense cascade-regression-based face alignment to estimate a dense set of 1024 facial landmarks. Then a part-based 3D deformable model is applied to reconstruct a dense 3D mesh of the face. Face images are normalized in terms of pitch, yaw and roll rotation and scale and then centered. At the output of this step, video resolution is 512 × 512 with an interocular distance (IOD) of about 100 pixels.

3.2 Patch Cropping and Encoding

The 3D face registration step ensures that faces in all frames of all individuals are registered to the same template and that same landmarks (facial parts) in all frames are very close to each other. This step allows us to identify the locations of face parts and crop patches containing the same face parts for all frames.

Patch locations are identified using the domain knowledge of human FACS coders and based on the FACS manual [5]. We identify $N = 9$ patches given in Figure 2 with the aim to cover specific face parts that are deformed during the performance of specific AUs, namely right eyebrow (P_1), left eyebrow (P_2), right eye (P_3), region between eyebrows & nose root (P_4), left eye (P_5), right cheek & lip corner (P_6), nose & upper mouth (P_7), left cheek & lip corner (P_8) and mouth & chin (P_9). Then, we crop $N = 9$ patches using the same identified locations from all frames in the dataset. The size of each RGB patch is 100 × 100 pixels.

Convolutional Neural Networks (CNNs) are employed as patch encoders. We feed each of 9 patches to a different patch encoder so that each encoder aims to learn representations of local face parts. All of these 9 encoders have the identical architectures, which include 3 convolutional layers and 1 fully connected layer. At the output of each patch encoder, we obtain vector representations of local patches.

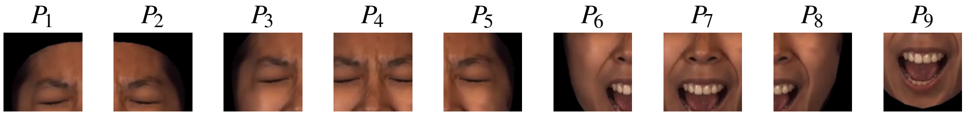

Figure 2: Cropped patches from 3D registered face images.

3.3 Patch Weighting by Sigmoidal Attention Mechanism

Different face patches contribute unequally to the face representation to predict AUs. In order to weight the contribution of patch encodings, we use an attention mechanism. An attention mechanism aggregates the representation of the informative patch encodings to form a face encoding. Let e_p be the encoding of patch p obtained at the output of CNN. First, patch encoding e_p is fed to a one-layer MLP to obtain hidden representation h_p of e_p as follows:

$$h_p = tanh(W_f e_p + b_f) \tag{1}$$

where W_f and b_f are the weight and bias parameters of the MLP, respectively. Then, the importance of each patch is measured by the similarity between h_p and a face level context vector c_f. In order to normalize the importance of patches to the range [0,1] and obtain attention weight α_p, we apply sigmoid function as follows:

$$\alpha_p = \frac{1}{1 + exp(-h_p^T c_f)} \tag{2}$$

Face level context vector c_f can be interpreted as the high level representation of fixed query *what are the informative patches to predict a specific AU?*. It is randomly initialized and learned during training. Finally, we obtain face encoding v as a weighted sum of patch encodings e_p as:

$$v = \sum_p \alpha_p e_p \tag{3}$$

Note that, it is typical to use softmax activation function for normalization in attention mechanisms employed in many NLP tasks. One such task is neural machine translation, where the network is trained to attend one word (or a few words, but not to the others) to obtain the corresponding translation of the word. Output of softmax function can be used to represent a categorical distribution. In our case, we aim to allow multiple patches to contribute to predict a specific AU. Therefore, instead of softmax, we used sigmoid activation function which allows for multiple selection with a collection of Bernoulli random variables.

3.4 AU detection

Face encoding v is a high level representation of face and is used for AU detection. We apply ReLU to v for non-linearity and have a fully connected layer to predict the occurrence of AUs. We train individual networks for each AU. We apply sigmoid function and use weighted binary cross-entropy loss as follows:

$$L = y * -log(\hat{y}) * w_{pos} + (1-y) * -log(1-\hat{y}) \tag{4}$$

where y denotes actual AU occurrence, $\hat{y}$ denotes predicted AU occurrence. w_{pos} is the weight that is used for adjusting positive error relative to negative error.

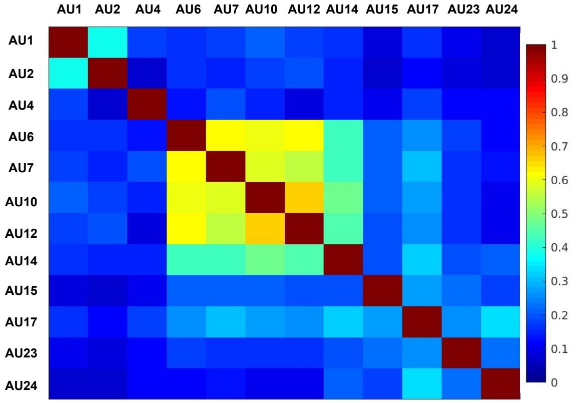

Figure 3: Co-occurrence matrix of AUs computed with Jaccard index.

4 Experiments

Dataset: BP4D is a manually FACS annotated database of spontaneous behavior containing 2D and 3D videos of 41 subjects (23 female and 18 male). Following previous research in AU detection, only 2D videos are used here. In BP4D, well-designed tasks initiated by an experimenter are used to elicit varied spontaneous emotions. Each subject performs 8 tasks. In total there are 328 videos of approximately 20 seconds each that have been FACS annotated manually. This results in about ~140.000 valid, manually FACS annotated frames. We include 12 AUs that occurred in more than 5% of the frames. Positive samples are defined as ones with intensities equal to or higher than A-level, and the remaining ones are negative samples. We visualize the co-occurrence matrix of AUs computed using Jaccard index in Figure 3. It can be observed that AU6, AU7, AU10, AU12 and AU14 co-occur frequently.

Network: We employ 32, 64, and 64 filters of 5×5 pixels in three convolutional layers with a stride of 1. After convolution, rectified linear unit (ReLU) is applied to the output of the convolutional layers to add non-linearity to the model. We apply batch normalization to the outputs of all convolutional layers. The network contains three maxpooling layers that are applied after batch normalization. We apply max-pooling with a 2×2 window such that the output of max-pooling layer is downsampled with a factor of 2. At the output of the fully connected layer, we obtain a patch encoding e_p of size 1×60, for each frame. In patch attention layer, we use the weight matrix W_f of size 60×60 and face level context vector c_f as 1×60. Attention layer output is a face encoding v of size 1×60, for each frame.

Training: We trained our architecture with mini-batches of 50 samples for 10 epochs. We used stochastic gradient descent (SGD) optimizer. Our models were initialized with learning rate of 1e-3, with a momentum of 0.9. In order to keep variability in the data, we used all of the available frames and did not subsample training frames to generate balanced dataset. For each AU, we assign w_{pos} to the ratio between the number of training frames excluding the AU and containing the AU. We perform a subject independent 3-fold cross-validation for BP4D dataset. Our folds include the same subjects as in [25].

Evaluation measures: We evaluate the performance of PAttNet on two common frame-based metrics: F1-score and AUC. F1-score is the harmonic mean of precision (P) and recall (R) $\frac{2RP}{R+P}$. AUC shows the success of classifier to rank frames with and without AU. For

Table 1: AU detection performances (F1-scores) on BP4D dataset. The best results are shown in bold and the second best results are shown underlined.

AU	LSVM	JPML [25]	DRML [26]	FVGG [10]	E-net [10]	LSTM [2]	ATF [24]	EAC Net [10]	DSIN [3]	JAA [17]	PAttNet
1	23.20	32.60	36.40	27.80	37.60	31.40	39.20	39.00	**51.70**	47.20	46.12
2	22.80	25.60	41.80	27.60	32.10	31.10	35.20	35.20	40.40	**44.00**	41.43
4	23.10	37.40	43.00	18.30	44.20	**71.40**	45.90	48.60	56.00	54.90	57.11
6	27.20	42.30	55.00	69.70	75.60	63.30	71.60	76.10	76.10	77.50	**77.91**
7	47.10	50.50	67.00	69.10	74.50	77.10	71.90	72.90	73.50	74.60	**76.15**
10	77.20	72.20	66.30	78.10	80.80	45.00	79.00	81.90	79.90	**84.00**	83.84
12	63.70	74.10	65.80	63.20	85.10	82.60	83.70	86.20	85.40	86.90	**88.41**
14	64.30	65.70	54.10	36.40	56.80	**72.90**	65.50	58.80	62.70	61.90	66.47
15	18.40	38.10	33.20	26.10	31.60	34.00	33.80	37.50	37.30	43.60	**51.19**
17	33.00	40.00	48.00	50.70	55.60	53.90	60.00	59.10	**62.90**	60.30	61.62
23	19.40	30.40	31.70	22.80	21.90	38.60	37.30	35.90	38.80	42.70	**44.11**
24	20.70	42.30	30.00	35.90	29.10	37.00	41.80	35.80	41.60	41.90	**57.31**
Avg.	36.68	45.93	47.69	43.81	52.08	53.20	55.40	55.58	58.86	60.00	**62.64**

each method, we computed average metrics over all AUs (denoted as Avg.).

5 Results

5.1 Comparison with the state-of-the-art

We compare the performance of PAttNet with those of state-of-the-art approaches namely Linear SVM (LSVM), Joint Patch and Multilabel learning (JPML) [25], Deep Region Multi-Label Learning (DRML) [26], Finetuned VGG Network, Network with enhancing layers (E-Net) [10], Network combining CNN and LSTM (LSTM) [2], Adversarial Training Framework (ATF) [24], Enhancing and Cropping Network (EAC Net) [10], Deep Structured Inference Network [3] and joint AU detection and face alignment (JAA) [17] in Table 1. For fair comparison, we excluded the studies which do not follow 3-fold protocol [21]. Results reflect that, our method gives the best F1-score for 6 of 12 AUs (AU6, AU7, AU12, AU15, AU23, and AU24) and the second best results for 4 AUs (AU4, AU10, AU14, and AU17). On average, our method outperforms all of the comparison approaches.

Since F1-score is affected by the skew in the labels and some action units are highly skewed, we also compute AUC results, which are not affected by the skew. Only a few studies report AUC values. In Table 2, we compare the performance of PAttNet with the state of the art approaches using AUC. PAttNet gives an average AUC of 72.68% over all AUs. For each AU, AUC is above 65%. PAttNet gives superior performance compared to all of the approaches reporting AUC.

5.2 Comparison with single patch and holistic face CNNs

Although AUs are identified from local patches, due to the co-occurring nature of multiple AUs, simultaneous changes occur in different parts of the face too. Therefore, AUs are better

Table 2: AU detection performances (AUC) on BP4D dataset. The best results are shown in bold.

AU	LSVM	JPML [25]	DRML [23]	PAttNet
1	20.70	40.70	55.70	**66.52**
2	17.70	42.10	54.50	**65.62**
4	22.90	46.20	58.80	**74.35**
6	20.30	40.00	56.60	**78.60**
7	44.80	50.00	61.00	**71.81**
10	73.40	75.20	53.60	**78.45**
12	55.30	60.50	60.80	**86.39**
14	46.80	53.60	57.00	**65.44**
15	18.30	50.10	56.20	**72.12**
17	36.40	42.50	50.00	**70.06**
23	19.20	51.90	53.90	**67.99**
24	11.70	53.20	53.90	**74.80**
Avg.	32.29	50.50	56.00	**72.68**

detected from combinations of local patches rather than the related local patch. To validate this, we trained CNNs using single patches for each AU. Moreover, due to the structured shape of face and locality of action units, learning representations of local parts (eyes, mouth, etc.) would be more informative than holistic representation of face to detect AUs. Previous patch-based approaches have shown superior results than holistic face based approaches. To validate this, we also trained CNN with frames of size 200×200 containing holistic face.

In Table 3 we present results of CNNs trained using single patches ([P_1, P_9]). Results show that different patches give the best performance to detect different AUs. Brow raiser action units AU1 and AU2 are best detected from left eyebrow patch P_1 and second-best performances are obtained from patches covering eye and eyebrow regions(P_4 for AU1 and P_3 for AU2). P_4, which contains between eyebrows region & nose root best discriminates AU4 (inner brow lowerer). Notice that although AU6 (cheek raiser) is an eye AU, it is best detected from P_7 (mouth patch), due to the frequent co-occurrence of AU6 with AU12 during Duchenne smile. AU7 (lid tightener) is recognized well from patches P_2 containing outer eye corner and P_4 containing inner eye corners. AU10 is upper lip puller and is best recognized from P_7. Although AU12 and AU14 are lip corner AUs, their presence lead to a more salient visual change in mouth patch P_7 (which also contains lip corners). Therefore, they are better detected from P_7 compared to specific lip corner patches P_6 and P_8. AU15 (lip corner depressor) on the other hand, is best detected by the lip corner patch P_8. Chin raiser AU17, is similarly detected by P_7 and P_9 patches that contain mouth and chin regions, respectively. AU23 and AU24 are mouth AUs and are best detected by P_7 (mouth patch) as expected. Results show that in most cases expected patches give the best performance. However, when results are compared with the 3D registered holistic face results, all AUs except for AU7 and AU24 are better detected from holistic face. Additionally, PAttNet, which uses combination of local patches outperforms using holistic face for all but two AUs (AU15 and AU23).

Table 3: Performances (F1-scores) obtained using single patch CNN, holistic face CNN, PAttNet with softmax attention and our proposed PAttNet approach. The best results are shown in bold for both single patch CNNs (left) and other methods (right). The second best results are shown underlined for single patch CNNs.

AU	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	Holistic	PAttNet Softmax	PAttNet Sigmoid
1	**39.5**	37.2	38.1	38.5	36.8	26.5	23.4	22.7	21.9	45.3	37.24	**46.1**
2	**36.3**	31.9	35.5	27.3	32.7	21.5	14.5	19.6	18.9	37.2	28.4	**41.4**
4	41.9	44.4	44.7	**46.1**	43.3	27.8	33.3	33.0	31.8	55.5	41.5	**57.1**
6	67.3	69.0	70.7	71.0	71.1	73.0	**73.2**	72.9	72.1	76.3	73.4	**77.9**
7	72.9	**75.6**	71.2	75.5	73.9	70.3	69.2	66.7	70.2	74.0	69.8	**76.2**
10	73.7	75.2	77.6	78.6	78.7	**80.6**	80.3	79.3	78.5	82.2	79.3	**83.8**
12	74.1	75.1	78.2	78.5	81.2	82.5	**83.9**	83.7	83.2	87.3	82.0	**88.4**
14	55.2	52.3	56.4	55.6	54.8	57.7	**61.4**	59.1	61.0	65.6	58.7	**66.5**
15	28.1	26.5	28.7	28.6	26.8	42.0	41.4	**45.7**	43.1	**51.6**	32.7	51.2
17	37.9	37.8	38.1	42.9	40.3	55.0	**58.1**	57.0	58.0	61.3	58.5	**61.6**
23	24.3	22.6	24.8	26.6	23.4	38.7	**42.9**	40.0	38.3	**47.1**	39.8	44.1
24	18.3	18.8	13.8	17.9	16.1	47.5	**52.4**	41.7	51.5	50.8	49.2	**57.3**

5.3 Patch attention analysis

In this section, we first compare the AU detection results of using our proposed function sigmoid (PAttNet) and conventional activation function softmax (PAttNet Softmax) to weight the contributions of patches. Comparison of the last two columns of Table 3 shows that using softmax instead of sigmoid causes a significant drop in the performance for almost all AUs. When we force the network to attend one or a few patches, it cannot learn proper facial representation. These results are consistent with the assumption that even if AUs relate to specific facial regions, co-occurring nature of AUs cause the contribution of other facial regions to detect specific AUs.

We also visualize the attention maps formed using the learned attention weights of PAttNet and PAttNet Softmax in Figure 4. We obtain an attention map for each training fold and then average these maps to obtain the presented attention maps. In both maps, entries can take values between [0,1]. Cells with black color denote that the corresponding patch has high attention weight (is significant) to detect the corresponding AU for all of these folds whereas cells with white color denote that the related patch is not significant to detect the corresponding AU in any of the folds. Multiple patches contribute with varying weights to detect AUs.

As expected, we obtain a denser map with sigmoid attention since softmax tends to select sparse entries. For most of the AUs, the network learns to attend meaningful patches. In both maps, generally higher attention is observed in upper face patches to detect AUs of upper face region (AU1, AU2 and AU4). Similarly, higher attention is observed in mouth and lip corner patches to detect AUs of lower face region. In sigmoid attention map, P_4 (the patch containing inner brow region) has the maximum contribution to detect AU1 while P_6 (the patch containing the lip corner) and P_9 (the patch containing mouth and chin) are the most significant regions to detect AU12. Although AU6 (cheek raiser) is an eye AU, performing the action changes the appearance of cheeks. In addition, it generally co-occurs with AU12,

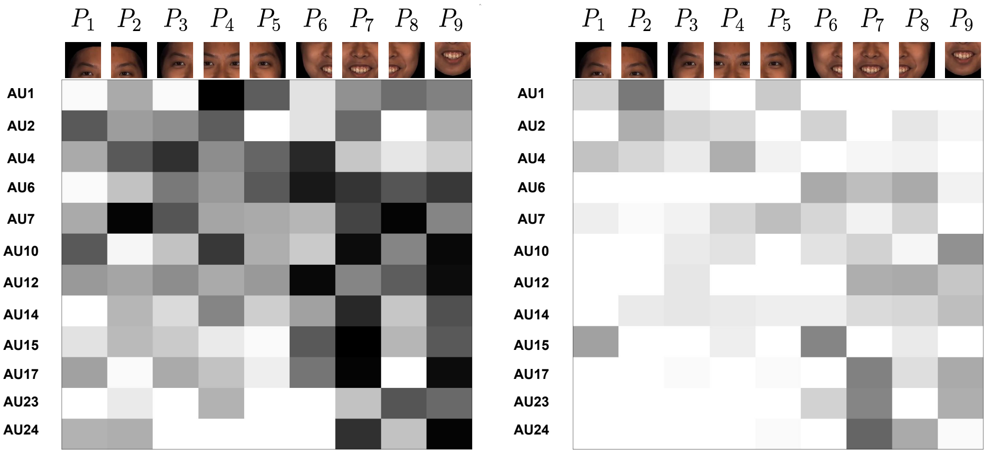

Figure 4: Average attention maps obtained using sigmoid attention (left) and using softmax attention (right). Attention weights of models obtained for each training fold are averaged. Attention weights are in [0,1]. White color represents no attention (0) and black color represents the maximum attention (1).

which also changes the appearance of lip corners and mouth. Therefore, the network learns to attend more to patches containing lip corner, cheek and mouth compared to the ones containing eyes. Similarly, the network highly attends P_2 and P_8 to detect AU7. While P_2 contains the facial parts whose appearances change with the AU7, P_8 is highly attended due to the correlations of AUs.

6 Conclusion

We have proposed a patch-attentive deep network called PAttNet, for AU detection. We first apply 3D face registration to remove the variation caused by the differences in pose and scale. Then, we crop patches containing important facial parts to detect specific AUs. After encoding each patch with CNN-based encoders, we weight the contribution of patch encodings using a patch attention mechanism. To allow multiple patches to contribute AU detection, we employ sigmoidal attention rather than the conventional softmax attention.

PAttNet outperforms state-of-the-art approaches on BP4D. Attention maps show that, with the help of sigmoidal attention PAttNet chooses to attend multiple patches and the most significant patches are meaningful. Softmax attention map is much sparser and softmax attention leads to lower AU detection performance. A future direction would be modeling spatiotemporal patch dynamics using LSTMs or 3D-CNNs and learning the importance of patches with a spatiotemporal attention mechanism to detect AUs.

7 Acknowledgments

This research was supported in part by NIH awards NS100549 and MH096951 and NSF award CNS-1629716.

References

[1] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. Openface 2.0: Facial behavior analysis toolkit. In *FG*, pages 59–66. IEEE, 2018.

[2] Wen-Sheng Chu, Fernando De la Torre, and Jeffrey F Cohn. Learning spatial and temporal cues for multi-label facial action unit detection. In *FG*, pages 25–32. IEEE, 2017.

[3] Ciprian Corneanu, Meysam Madadi, and Sergio Escalera. Deep structure inference network for facial action unit recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 298–313, 2018.

[4] Shichuan Du, Yong Tao, and Aleix M Martinez. Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences*, 111(15):E1454–E1462, 2014.

[5] P Ekman, WV Friesen, and JC Hager. Facial action coding system: Research nexus network research information. *Salt Lake City, UT*, 2002.

[6] Stefanos Eleftheriadis, Ognjen Rudovic, and Maja Pantic. Multi-conditional latent variable model for joint facial action unit detection. In *Proceedings of the IEEE international conference on computer vision*, pages 3792–3800, 2015.

[7] Shashank Jaiswal and Michel Valstar. Deep learning the dynamic appearance and shape of facial action units. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–8. IEEE, 2016.

[8] László A Jeni, Jeffrey F Cohn, and Takeo Kanade. Dense 3d face alignment from 2d video for real-time use. *Image and Vision Computing*, 58:13–24, 2017.

[9] Bihan Jiang, Michel F Valstar, and Maja Pantic. Action unit detection using sparse appearance descriptors in space-time video volumes. In *Face and Gesture 2011*, pages 314–321. IEEE, 2011.

[10] Wei Li, Farnaz Abtahi, Zhigang Zhu, and Lijun Yin. Eac-net: Deep nets with enhancing and cropping for facial action unit detection. *IEEE transactions on pattern analysis and machine intelligence*, 40(11):2583–2596, 2018.

[11] Ping Liu, Joey Tianyi Zhou, Ivor Wai-Hung Tsang, Zibo Meng, Shizhong Han, and Yan Tong. Feature disentangling machine-a novel approach of feature selection and disentangling in facial expression analysis. In *European Conference on Computer Vision*, pages 151–166. Springer, 2014.

[12] Simon Lucey, Ahmed Bilal Ashraf, and Jeffrey F Cohn. Investigating spontaneous facial action recognition through aam representations of the face. In *Face recognition*. IntechOpen, 2007.

[13] Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, 2015.

[14] Itir Onal Ertugrul, Jeffrey F Cohn, László A Jeni, Zheng Zhang, Lijun Yin, and Qiang Ji. Cross-domain au detection: Domains, learning approaches, and measures. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, 2019.

[15] Pau Rodríguez, Josep M Gonfaus, Guillem Cucurull, F XavierRoca, and Jordi Gonzàlez. Attend and rectify: a gated attention mechanism for fine-grained recovery. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 349–364, 2018.

[16] Enrique Sanchez, Georgios Tzimiropoulos, and Michel Valstar. Joint action unit localisation and intensity estimation through heatmap regression. In *BMVC*, 2018.

[17] Zhiwen Shao, Zhilei Liu, Jianfei Cai, and Lizhuang Ma. Deep adaptive attention for joint facial action unit detection and face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 705–720, 2018.

[18] Seyedehsamaneh Shojaeilangari, Wei-Yun Yau, Karthik Nandakumar, Jun Li, and Eam Khwang Teoh. Robust representation and recognition of facial emotions using extreme sparse learning. *IEEE Transactions on Image Processing*, 24(7):2140–2152, 2015.

[19] Sima Taheri, Qiang Qiu, and Rama Chellappa. Structure-preserving sparse decomposition for facial expression analysis. *IEEE Transactions on Image Processing*, 23(8): 3590–3603, 2014.

[20] Y-I Tian, Takeo Kanade, and Jeffrey F Cohn. Recognizing action units for facial expression analysis. *IEEE TPAMI*, 23(2):97–115, 2001.

[21] Zoltán Tősér, László A Jeni, András Lőrincz, and Jeffrey F Cohn. Deep learning for facial action unit detection under large head poses. In *European Conference on Computer Vision*, pages 359–371. Springer, 2016.

[22] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, 2016.

[23] Zheng Zhang, Jeff M Girard, Yue Wu, Xing Zhang, Peng Liu, Umur Ciftci, Shaun Canavan, Michael Reale, Andy Horowitz, Huiyuan Yang, et al. Multimodal spontaneous emotion corpus for human behavior analysis. In *CVPR*, pages 3438–3446, 2016.

[24] Zheng Zhang, Shuangfei Zhai, and Lijun Yin. Identity-based adversarial training of deep cnns for facial action unit recognition. In *BMVC*, 2018.

[25] 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 and holistic expression recognition. *IEEE Transactions on Image Processing*, 25(8):3931–3946, 2016.

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

[27] Lin Zhong, Qingshan Liu, Peng Yang, Junzhou Huang, and Dimitris N Metaxas. Learning multiscale active facial patches for expression analysis. *IEEE transactions on cybernetics*, 45(8):1499–1510, 2015.