Facial Action Unit Detection with Capsules

Anonymous WACV submission

Paper ID ****

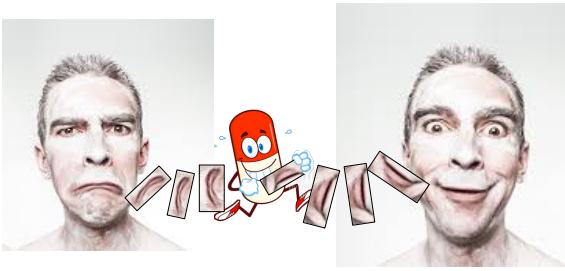

Figure 1. Turning that frown upside down: We propose that capsules can model action units as individual part deformations

Abstract

In this paper we motivate the use of capsule networks for facial action unit detection. We argue that action unit activations may be seen as local part deformations - for example AU 1,2, and 4 are deformations of the part 'eyebrows'. Different part deformations in a regular convolutional network must be modeled and represented as separate neurons. However, with capsule networks a part can be represented by a single capsule, and its deformations can be modeled by its direction. We test this hypothesis by creating a capsule network for action unit recognition. We find that capsule networks are indeed able to model action units and local part deformations as well. These lead to state-of-the-art results on the BP4D and DISFA datasets. We analyse the learned capsules' properties and find that capsule magnitude correlates with expression intensity and that capsule pose captures varied attributes such as face size, lighting, pose, and skin color. Finally we use activation gradient ascent to visualize capsule direction, and find that a single capsule can represent multiple deformations of the same part, while a single convolution neuron does not.

1. Introduction

Facial Action Coding System (FACS) is a system to define and name facial movements by their appearance on the face. Informed by the underlying muscular structure of a face, FACS annotation can be reliably used for describing as well as identifying facial expressions and is therefore not as subjective as grimace scales. There are 24 main facial action units to describe the human face. Additionally, action units can also be coded for intensity on a 4 to 5 point scale. While, extremely useful, manual action unit coding is a cumbersome process that can only be carried out by trained experts. Due to this hurdle, automatic action unit detection is an important problem for computer vision research.

Facial action unit detection requires identifying subtle deformations on parts of the face. Consequently, features that capture local movements around key parts of the face have been used to train machine learning systems. For example, in [1], a seminal work on emotion and facial action unit understanding, Gabor features were extracted around keypoints of the face to capture local muscle deformations. Developing features that capture part deformations well has also motivated more recent work [43, 19] where separate convolutional filters are trained to correspond to different parts of the face. The motivation behind these works is similar – the better we are able to model how parts of a face look and change, the better we can detect action units.

Recently, capsule networks [29, 13] were introduced. There are two primary reasons why capsule architectures have advantages over regular convolutional neural networks. Intuitively, neurons in a CNN can represent an attribute of the input image – such as the presence of an eye – and the activation of a neuron represents a confidence value in whether that attribute can be found in the image. With capsules the expressive power is increased - its activation can represent a confidence value in its presence or absence, and its direction can represent properties of the attribute. For example, the direction of the capsule can indicate how rotated the eye is, whether it is open or close, etc. The second advantage of capsules is that the additional representative capacity allows for complex routing procedures. The pose of a capsule can be used to determine how it is propagated through the network. This is in contrast to neuron activations in a CNN that are propagated solely on the basis of its scalar value. As a result, the routing procedure followed

by capsule networks can mimic the effect of a much deeper convolution neural network, trained with various data transformed augmentation techniques.

We believe that the higher expressive power capsules afford to each visual attribute of an image can directly translate to better modeling of local part deformations. Action unit activations can be seen as local part deformations, and therefore capsules can be better at detecting and modeling action units. To give a naive parallel example, if a capsule learns to detect lips, its pose can represent the type of deformation the lips are in – so different capsule poses can end up representing action units 12, 14, 15. At the same time, for a convolutional network, if a neuron comes to be associated with lips, it cannot express the pose the lips are in. It can simply be either active - indicating the presence of lips – or inactive – indicating that the lips are not present. In order to represent lips in AU-12 or in AU-15, the network must learn to associate separate neurons.

Contributions. Our main contributions in this paper are as follows:

We present results that indicate that capsules are indeed better than CNNs at modeling local part deformations - and therefore action units. Previous work [29, 13] have shown that capsules can model global deformations - so the network can generalize well across image level deformations (affine transformations of MNIST), or viewpoint deformations (azimuth changes on small-NORB). In this paper, we show that capsules can also capture small deformations well - and may therefore be extremely useful in other areas such as fine-grained classification.

We develop a capsule network for action unit detection that gives state-of-the-art results across two large action unit datasets. On BP4D dataset we outperform the closest baseline architecture by 14.1% in AUC. We replicate similar performance gains on DISFA. We additionally present results on emotion recognition for CK+ dataset, and find that our architecture generalizes well.

To the best of our knowledge, capsule networks have not been used to perfom facial action unit detection before. To this end, we thoroughly analyse and visualize the learned capsule networks. We visualize the effect of changing capsule magnitude as well as capsule direction via a reconstruction network. We find that capsules are able to model face pose, shape, lighting, and skin color, and that capsule magnitude is correlated with action unit or emotion intensity. Finally, we use activation maximizing gradient ascent to visualize capsule features and compare them with regular convolution networks. We find that capsules are able to model part deformations as changes in capsule direction, where as individual convolution neurons are not able to model multiple part deformations.

2. Related Work

2.1. Capsule Networks

Capsule networks were first proposed in [29]. The network replaces scalar neurons with higher dimension capsules - so that activation and neuron attributes can be modeled jointly. In addition, capsule direction or pose can be used to route capsules between higher layers - which replaces pooling based routing in convolutional networks. In [13], the authors propose vector capsules whose magnitude represents the activation of a capsule. Iterative routing is done using a simple agreement between lower and higher level capsule directions. In [13] the authors introduce matrix capsules, where a separate value represents the capsule activation. Routing is done using an EM algorithm, such that the probability distributions of higher and lower level capsules between consecutive layers are in agreement. Furthermore, the authors introduce convolutional capsules, whereas [29] only worked with fully-connected capsules. In this paper, we use vector capsules, with dynamic routing, and work with fully-connected capsules only.

2.2. Facial Action Unit Understanding

Papers in action unit understanding have focused on two broad sub problems - action unit intensity estimation, and action unit detection.

A number of traditional non-deep approaches improve action unit understanding by exploiting the co-occurence patterns between action units - either by developing a learning model that can help capture inter-AU relations [34, 37, 7], by developing a model based on prior knowledge of AU relationships and semantics [32, 20], or by using a data-driven approach to learn important AU relationships [46]. In particular, [42] jointly learns to identify important patches, and positive and negative correlations between action units for understanding action units.

Traditional approaches have also learned action units by assistance from facial keypoints - features extracted around keypoints are used for action unit detection. Some examples of such approaches are [7, 18, 3, 33, 36].

Deep learning has also been applied to the problem of action unit detection with great success [9, 10, 14, 11, 43, 19]. Amongst these, two papers in particular require discussion. In [43] the authors develop a 'region layer' that splits the incoming convolution map into a grid and develops separate convolutional maps for each grid section. The resulting map is concatenated spatially and propagated through the network. In a similar vein, [19] also explicitly design their deep network to develop features for parts of a convolutional map - however, unlike [43] the spatial regions that are broken up are based on facial keypoint locations and their correlations with action units. Both methods explore a similar idea - to develop separate features for parts of a

face - as is based on the intuition that different areas of the face correspond to different AU activations that require their own unique set of features for identification.

ECCV papers [5, 30]. They both do facial action unit recognition.

CVPR papers [40, 28] do weakly supervised au recognition. [12] optimizes filter size per au by either expanding or contracting filter sizes from a base size over training. the model relies on separate models for each action unit. [39] uses expression indenpendant and expression dependant prior knowledge about action units to learn au classifiers without direct supervision. [41] does au intensity estimation.

2.3. Expressions

Facial expression prediction is a well-explored topic of research in computer vision. We primarily focus on action unit detection, but also show qualitative and quantitative results on expression detection. Some approaches that do not use deep learning are, [1, 8, 31, 45], of which [1] is of particular note for creating a pipeline based on extracting features around facial keypoints, detecting action units, and fusing action unit detections temporally for emotion detection.

A number of papers also explore emotion understanding in a simple deep feed-forward classification network setting [23, 16, 26, 44, 6]. [21, 22] attempt to enforce AU understanding to the end of emotion classification. Of these, [16] is notable for impressive results on expression detection and demonstrating the importance of data augmentation for the task of expression understanding. [6] is also an important paper that proposes a two-stage training pipeline to transfer VGG-Face [27] features for the task of expression classification. Also noteworthy is [15], which uses facial keypoint locations over time to train a network that is meant to capture temporal deformations alongside a traditional image-based CNN. Lastly, [17] proposes an encoder-decoder type architecture that learns from pairs of neutral/non-neutral expressions to develop features that are discriminative for expression classification.

CVPR [35].

3. Approach

Our network comprises two modules - a capsule network that outputs action unit capsules, and a reconstruction network that takes concatenated action unit capsules, and is trained to reconstruct the input image. During training, the capsules for all classes apart from the ground-truth classes are zeroed-out and used as input to the reconstruction network. In this way, the reconstruction network does not, directly, affect classification accuracy.

We train our network with color inputs of size 96×96. We train with three routing iterations.

The capsule network architecture comprises of two convolution layers, with 64 and 128 filters, and kernel size of 5. Each is followed by max-pooling and ReLu. The convolution layers are followed by a primary caps layer with 32 capsules of dimension 8, filter size 7, and stride 3. The resulting activation map is then fully connected to our class capsule layer with n capsules, each with dimension 32, where n is the number of output classes.

Our reconstruction network comprises of 3 fully connected linear layers of dimension 512, 1024, and 1024. The last layer is reshaped to 32×32 and then bilinearly upsampled to 96×96. In experiments we refer to this model as 'Ours'.

Additionally, we propose a larger capsule network with VGG convolution layers as its base. The network is identical to VGG-16 up to the end of its convolution layers. The last max-pooling layer is removed. This is followed by 32 primary capsules of size 8, kernel size of 3 and stride of 2. The class capsules have dimension 32. The reconstruction network comprises of three fully-connected layers with dimensions 512, 1024, and 9408. The output is resized to 56×56 and then bilinearly upsampled to 224 – the input image size. We show results of this model with the convolution layers initialized with both Imagenet pretrained weights (Ours-VGG) and VGG-Face pretrained weights (Ours-VGGF). We found Ours-VGGF was prone to overfitting. We therefore add spatial dropout at 70% after the last convolution layer.

We use the margin loss from [29] to train our networks. The original loss for class c:

$$L_c = T_c \max(0, m^+ - ||v_c||)^2 + \lambda(1 - T_c) \max(0, ||v_c|| - m^-)^2$$

where $T_c = 1$ iff class c is present, $m^+ = 0.9$ and $m^- = 0.1$, v_c is the class capsule, and λ is a downweighting term for negative samples set to 0.5.

For single class classification (such as expression classification) a softmax operation is applied across all $||v_c||$. However, for the multiclass classification setting (action unit detection), the softmax is not applied. Note that due to the capsule squashing operation, the magnitude of all output capsules still lies between 0 and 1.

For action unit detection, the occurrence of different AUs is highly imbalanced. We therefore modify the margin loss to work with a class specific weight:

$$L_c = w_c(T_c \max(0, m^+ - ||v_c||)^2 + \lambda(1 - T_c) \max(0, ||v_c|| - m^-)^2)$$

where w_c is the inverse of the frequency of an action unit occurrence, normalized across all classes to sum to one.

The margin loss is then averaged across all action unit instances in a batch, and added with average reconstruction loss for the batch. We use mean square error to supervise the reconstruction network. The final loss term is:

$$L_{final} = L_{cls} + \alpha L_{recon}$$

AU	LSVM[43]	JPML[42]	DRML[43]	CPM[36]	CNN+LSTM[4]	FVGG[19]	ROI[19]	FERA[14]	Ours	Ours-VGG	Ours-VGGF
1	23.2	32.6	36.4	43.4	31.4	27.8	36.2	28.0	46.8	40.0	**47.3**
2	22.8	25.6	41.8	40.7	31.1	27.6	31.6	28.0	29.1	27.7	**39.9**
4	23.1	37.4	43.0	43.4	71.4	18.3	43.4	34.0	**52.9**	42.2	52.8
6	27.2	42.3	55.0	59.2	63.3	69.7	77.1	70.0	75.3	76.1	**77.9**
7	47.1	50.5	67.0	61.3	77.1	69.1	73.7	78.0	77.6	71.8	**79.9**
10	77.2	72.2	66.3	62.1	45.0	78.1	**85.0**	81.0	82.4	81.8	84.0
12	63.7	74.1	65.8	68.5	82.6	63.2	87.0	78.0	85.0	87.3	**88.1**
14	64.3	65.7	54.1	52.5	72.9	36.4	62.6	**75.0**	65.7	63.5	67.2
15	18.4	38.1	36.7	34.0	33.2	26.1	45.7	20.0	33.7	36.1	**49.2**
17	33.0	40.0	48.0	54.3	53.9	50.7	58.0	36.0	60.6	62.1	**65.4**
23	19.4	30.4	31.7	39.5	38.6	22.8	38.3	41.0	36.9	35.3	**47.7**
24	20.7	42.3	30.0	37.8	37.0	35.9	37.4	-	43.1	44.3	**55.1**
Avg	35.3	45.9	48.3	50.0	53.2	43.8	56.4	51.7	57.4	55.7	**62.9**

Table 1. F1-Frame results on BP4D dataset. Our method outperforms all methods that are trained from scratch and even outperforms FVGG despite not using any external data.

AU	LSVM[43]	APL[43]	DRML[43]	FVGG[19]	Ours	Ours-VGG	Ours-VGGF
1	10.8	11.4	17.3	32.5	17.6	15.7	15.7
2	10.0	12.0	17.7	24.3	18.8	25.7	25.7
4	21.8	30.1	37.4	61.0	50.1	41.3	41.3
6	15.7	12.4	29.0	34.2	44.8	52.8	52.8
9	11.5	10.1	10.7	1.67	21.6	40.7	40.7
12	70.4	65.9	37.7	72.1	65.1	70.1	70.1
25	12.0	21.4	38.5	87.3	68.8	62.5	62.5
26	22.1	26.9	20.1	07.1	45.4	47.7	47.7
Avg	21.8	23.8	26.7	40.2	41.5	44.6	44.6

Table 2. F1-Frame results on DISFA dataset. Our method outperforms all methods that follow a similar training protocol as ours.

where α is a weight parameter, L_{recon} is the average reconstruction loss and L_{cls} is the average margin loss. We set α to bring the order of the average reconstruction loss in the same order of magnitude as the averaged margin loss at the beginning of training. We set α to $1e-7$ at for the smaller network and $1e-8$ for the larger VGG based model.

4. Experiments

4.1. Action Unit Detection

4.1.1 Datasets

We present results on two widely-used datasets.

BP4D [38]: The dataset contains 328 videos of 31 subjects while completing eight different tasks designed to elicit emotion. Frames are annotated with 12 different action units. In total there are a little less than 140000 frames that we can use. Following common procedure, we do 3 fold cross validation on subjects, train on 2 folds, and test on the third. Results are collated across folds and reported.

DISFA [25]: 26 subjects are recorded while watching videos. Action units and their intensity are annotated for each frame. Similar to BP4D we conduct 3 fold cross validation, and collate results across folds.

For both datasets we detect and align faces using [2]. Images are randomly horizontally flipped, rotated, scaled, translated, cropped, and pixel augmented for data augmentation.

4.1.2 Metrics

We report F1-Frame score, as well as AUC. The F1 score is the harmonic mean of precision and recall, and used by AU detection methods to report results. AUC is the area under the curve of the receiver operating curve and captures the relationship between true and false positives.

AU	LSVM	JPML[42]	AlexNet	ConvNet	LCN	DRML	Ours	Ours-VGG	Ours-VGGF
1	20.7	40.7	34.9	49.4	51.9	55.7	65.7	60.3	**66.1**
2	17.7	42.1	25.8	51.3	50.9	54.5	56.0	54.9	**63.7**
4	22.9	46.2	36.1	47.4	53.6	58.8	70.2	62.7	**71.8**
6	20.3	40.0	48.3	52.2	53.2	56.6	71.3	**74.0**	73.7
7	44.8	50.0	54.3	64.8	63.7	61.0	60.6	58.7	**68.5**
10	73.4	75.2	54.3	61.4	62.4	53.6	70.8	**73.4**	70.1
12	55.3	60.5	50.0	60.2	61.6	60.8	74.6	**81.2**	80.8
14	46.8	53.6	47.7	29.8	**58.8**	57.0	56.7	56.9	57.5
15	18.3	50.1	34.9	50.6	49.9	56.2	59.4	60.9	**71.2**
17	36.4	42.5	48.5	53.5	48.4	50.0	66.1	67.4	**71.0**
23	19.2	51.9	40.5	49.5	50.3	53.9	61.6	60.5	**69.6**
24	11.7	53.2	31.7	52.5	47.7	53.9	67.6	68.3	**77.4**
Avg	32.2	50.5	42.2	51.8	54.4	56.0	65.0	64.9	**70.1**

Table 3. AUC scores on BP4D. We outperform all methods with a large margin.

AU	LSVM	APL	AlexNet	ConvNet	LCN	DRML	Ours	Ours-VGG	Ours-VGGF
1	21.6	32.7	47.8	44.2	44.1	53.3	57.3	56.0	56.0
2	15.8	27.8	52.1	37.3	52.4	53.2	58.9	61.5	61.5
4	17.2	37.9	44.0	47.9	47.7	60.0	70.4	63.2	63.2
6	08.7	13.6	44.3	38.5	39.7	54.9	67.6	70.2	70.2
9	15.0	64.4	48.7	49.5	40.2	51.5	66.0	67.5	67.5
12	93.8	94.2	55.3	54.8	54.7	54.6	77.1	79.0	79.0
25	03.4	50.4	50.2	48.4	48.6	45.6	75.7	72.2	72.2
26	20.1	47.1	45.8	45.8	47.0	45.3	69.2	67.2	67.2
Avg	27.5	46.0	49.1	45.8	46.8	52.3	67.8	67.1	67.1

Table 4. AUC scores on DISFA. We outperform all methods with a large margin.

4.1.3 Implementation Details

For 'Ours' on BP4D we train for ten epochs and learning rate of 1e-4. For finetuning on DISFA, we transfer convolution features only, and train at learning rate of 1e-4 for the first 5 epochs, and drop it to 1e-5 for the remaining 5 epochs.

For 'Ours-VGGF' we follow the procedure from ROI and fix all convolution layers up to the beginning of conv5. We finetune the conv5 layers at a 10 times lower learning rate, and train for 5 epochs. The capsule, and reconstruction layers are trained at 1e-4 learning rate. For 'Ours-VGG', we additionally finetune the first four convolution layers, and train for 10 epochs. For DISFA finetuning we initialize the convolution layers' weights with the best performing BP4D model and do XXX. We additionally present results without any BP4D transfer for DISFA (Ours-VGGFS, and Ours-VGGS) that uses the same training procedure as used for BP4D dataset.

4.1.4 Results

Results on BP4D dataset are shown in Table 1 for F1 and Table 3 for AUC. 'Ours' shows results after training our model from scratch and is comparable to all columns apart from ROI - which builds on VGG-Face features, and FVGG which is VGG-Face finetuned for AU detection. Our method outperforms all methods that are trained from scratch and even outperforms FVGG despite not using any external data. Overall, our best method outperforms the

Method	Accuracy
AURF[21]	92.2
AUDB[22]	93.7
Khorrami[16]	96.4
GCNet*[17]	97.28
FN2EN*[6]	96.8
Ours	93.7
Ours-Max	96.2

closest baseline on BP4D by 6.5% F1 score with external data, and our method without any external data outperforms the closest comparable baseline ([4]) by 4.2% despite not using any temporal information.

Following common procedure we present DISFA results after transferring from BP4D model in Table 2 for F1 and Table 4 for AUC. For all models we outperform the baselines significantly. We believe we are able to outperform baselines that often have more parameters because we are using capsule networks that are better equipped to model small facial deformations.

4.2. Emotion Detection

We also explore the use of capsules on the related task of emotion recognition on the Cohn-Kanade dataset [24]. We follow the established protocol of 10 fold cross validation, and average results across folds.

Figure **??** (left) shows our results on 8 emotion classification against several state-of-the-art methods. Our results are comparable to the state-of-the-art. We found that test results were prone to fluctuate throughout training, and due to the small dataset size, some folds were prone to overfit. We therefore also report the best test accuracy we achieve during training as 'Ours Max' to provide an idea of the upper limit our model may achieve with more careful training and hyperparameter searching.

4.3. Visualizing Capsules by Reconstruction

Every class capsule is a 32 length long vector. This vector can be modified by rescaling its magnitude, or altering its direction. For every altered version of an input image's correct class capsule, we can use reconstruction network to visualize the capsule.

4.3.1 Magnitude

Since capsule magnitude represents the confidence of our network in a capsule class's presence in the image, we expect increasing capsule magnitude to create reconstructions that represent that class even more. In other words, we would expect the reconstruction of a 'surprise' capsule with less magnitude to show less surprise than a reconstruction of the same capsule scaled to a higher magnitude.

Figure 2. We show the effect of altering magnitude of a class capsule. As the magnitude or activation of a capsule increases, the intensity of the facial expression also increases.

Figure 3. Reconstructions of test images with increasing capsule magnitude (left to right) on the BP4D dataset

Figure 2 shows reconstructions of different expression capsules with increasing magnitude. From left to right the magnitude was increased from 0.1 to 0.9 at increments of 0.1. The extreme left image is the input image. For each of the capsules, we see the expression become more pronounced and exaggerated. For example, for happiness (row 1), the smile in subsequent reconstructions becomes wider. Similarly, for surprise (row 3), a dark spot resembling an open mouth begins to appear, eventually resembling a full jaw drop.

We repeat this experiment with our second network, this time increasing the magnitude of all action units that are present in the image. For BP4D Figure 3(top) increasing magnitude results in unique features of the image being exaggerated, such as skin color (3rd row), face and neck shape (4th and 5th row), or pose (2nd row). At the same time, certain AUs also become prominent. The open mouth smile in the 3rd row, the jaw drop in the 4th row, the furrowed brow in the 2nd row, and the raised chin in the 1st row, all become more obvious as magnitude is increased.

However, these results are not as clear as the ones from the expression network, and the changes in action units' prominence would be hard to see unless we knew what to look for. This can be due to the training data where expression changes are spontaneous and subtle and not posed and exaggerated as in the Cohn-Kanade dataset. They are unlikey to effect the reconstruction loss function enough to create a strong supervision signal. Additionally, there are multiple action units present in every image, so the net-

Figure 4. We reconstruct an image for its ground truth class while changing its capsule direction by altering the value of each of its 32 dimensions for the CK+ dataset. The capsule dimensions are associated with attributes as face scale (first row), face shape (second row), and lighting(third row).

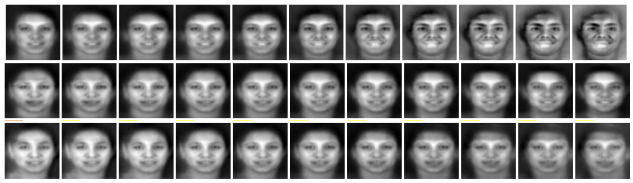

Figure 5. We reconstruct an image for its ground truth class while changing its capsule direction by altering the value of each of its 32 dimensions for the BP4D dataset. The first row shows face features and skin change. In the second, teeth appear. The neck in the last row indicates that the capsules have also learned pose attribute.

work may only prioritize the reconstruction of obvious action units that create large changes in the face, and not all action units.

4.4. Direction

We can also keep capsule magnitude stable, while changing its direction. For this, we vary the value of each of its 32 dimensions between -0.5 and 0.5, and reset the capsule magnitude to its original magnitude. In Figures 4 and 5 we show the effect of changing some capsule directions. The capsule dimensions are associated with attributes as varied as face shape, skin color, pose, or the visbility of teeth.

4.5. Visualizing by Gradient Ascent

An alternative method for visualizing capsule features is by activation maximization. In this approach the input image is treated as a learnable layer, and changed by gradient ascent for a particular optimization function. This optimization function can be the activation of a neuron in the network, the norm of a layer in the network, or any variations and combinations of the two. The core idea is that as the network modifies the input image to increase the objective function, say the activation of a neuron, the input image will begin to show the visual attribute that the neuron has learned to encapsulate. This process is popularly refered to as 'deep dreaming' - see [**?**] for an excellent overview.

Since the output of a capsule is a vector and not a scalar, we do gradient ascent on its magnitude which is analogous to doing gradient ascent on its activation. A naive application of this method is prone to create high frequency and non-sensical images. It is therefore necessary to regularize the input image. We use gaussian blurring and random jittering.

Figure 6. Deep dreaming expressions. The top row shows the mean image of each class, and the bottom row shows dreamed images for the class. Line correspond to lines that appear for each expression.

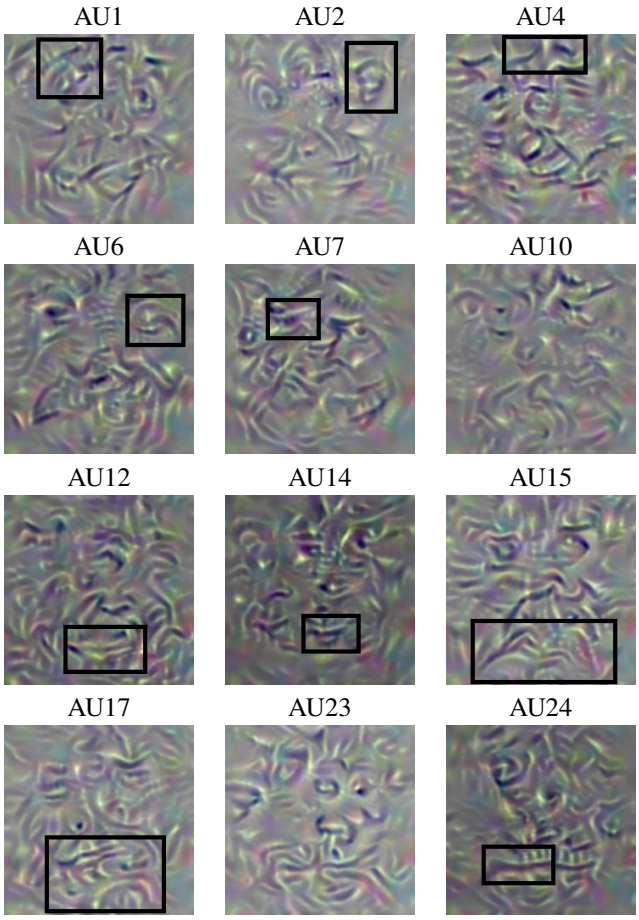

Figure 7. Deep dreaming AUs. The boxes show where we see the AUs. AU lookup table is Figure **??**. For AU 10 and AU 23 the results are diffcult to interpret.

4.5.1 Class Capsules

We first visualize the class capsules' properties for both the expression and action unit detection networks. For both net-

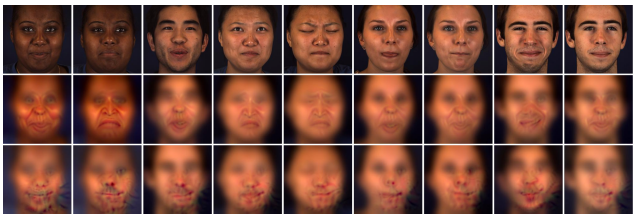

Figure 8. Comparing convolution and capsule units. The top row shows the input images. The second row shows the result of performing gradient ascent visualization on capsule 21on the mouth area. The bottom row shows the results on the same input images with convolution unit 498 on a finetuned convolution network. While a single capsule can have high magnitude with many different types of mouth positions, a single convolution capsule is only able to maximize activation for a thin upturned corner type mouth, regardless of the type of input image.

works we input a random noise image, and follow the procedure above to amplify the magnitude of each class in turn.

Figure 6 shows the results on the expression network. While the results are not natural - even scary - looking, they demonstrate that the expression capsules have indeed learned to identify correct attributes. For surprise the dark circle below the lips indicates that the capsules have learned to identify a jaw drop. The forehead lines indicate that the network is successfully identifying raised eyebrows by looking for forehead lines. Similarly, for happiness, the diagonal lines moving outward from the mouth corners resemble the shape of smiling lips, and the lines going diagonally down from each side of the nose resemble smile lines.

Figure 7 shows results on a VGG-Face finetuned action unit detection capsule network, with the attributes of the action unit surrounded with a black bounding box. The indicate that for almost every action unit the network is learning the correct characteristics. For example, for AU1, the network not only learns the inside raised eye brow shape, but also the corresponding folds in the skin around it. For AU 24, the network learns that the lips appear narrower and that lines apear around the mouth. For AU 6 the network focuses on the outside of the eyes where laugh lines appear as the outer cheek muscles are drawn up.

At the same time these results are enlightening because they indicate that the network has not completey decorrelated co-occuring but non-causal appearance changes. For example for AU 2, open and smiling mouths are also reconstructed by the network. AU 2 commonly occurs when people are surprised or happy, and so the corresponding mouth changes have also been learned by the network. These results indicate that the network is prone to falsely predict AU 2 when someone is laughing even if there eyebrows are not raised.

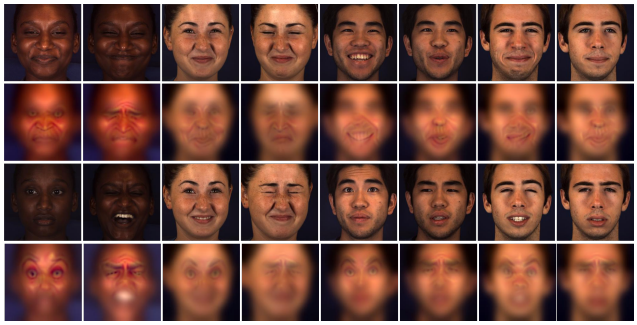

Figure 9. Activation maximization visualizing primary capsules. The first and third row show input test images. The second and fourth row show the result of activation maximizing for primary capsule 21 and 5 respectively. The results show that a single capsule is able to model multiple deformations of the mouth area and eyebrows, and activates differently for different action units.

Figure 10. Activation maximization on the mouth area for capsule 5. While this capsule is able to model changes in brow position well (Figure 9) it is unable to model differnt mouth positions and is 'sticky' towards a downturned mouth position.

4.5.2 Primary Capsules

In Section 4.3 we show that capsule magnitude is linked to how pronouced a expression or AU is, and its direction can be linked to different attributes a face or picture may have. This is done through the reconstruction network. However, class capsules are fully connected to the layer below and therefore do not help us understand how capsules are able to model local part deformations.

To gain insight in to how capsules model changes in local parts of the input image, we perform activation maximization on the magnitude of primary capsules' activations. More specifically, for each action unit, we use gradient ascent to maximize the capsule activation magnitude at a specific activation map location. This process is repeated for all action units. Unlike in Section 4.5.1, we do not use input images that are random noise since that does not let us control the direction in which capsules are activated, and then magnified. Furthermore, since primary capsules are applied to each image in a sliding window fashion, each capsule may have very different responses depending on where its receptive window overlaps with the input image.

Instead, we use test images that have perfect predictions, and the highest confidence (or class activation magnitude) per test subject and per action unit. This allows us to get primary capsules that have variation in direction naturally, while also allowing us ignore shortcomings of the network (perfect predictions), and test across identities.

Spatial locations for each action unit are chosen based on which part of the face each action unit deforms. In Figure 9 we show examples of different action units becoming more pronounced with gradient ascent on the *same* primary capsule's magnitude. Furthermore, we find that the cosine distance between activations after performing gradient ascent. This shows that the same capsule is able to model different types of deformations as changes in capsule direction. However, this is not always true for the primary capsules, and Figure 10 shows examples of a capsule that is unable to model deformations around the mouth, but is, on the other hand, able to model changes around the brow.

For comparison, in Figure 8 we perform similar activation maximization visualization on convolution units. We finetune a VGG-Face network for action unit detection using the same training data. We use the activations of the last convolution layer after ReLU and max-pooling for this purpose since it allows us to have comparable receptive window size as the primary capsules, and lets us ignore convolution units that are turned off by ReLU and therefore irrelevant to the final prediction. We then forward the same selected test images through the network and record the convolution units with the highest activation at the specific spatial locations in the activation map for each action unit. We then perform gradient ascent on the activations of these particular convolution units at selected spatial locations. Since we use the post ReLU activations, it is not always possible to generate visualizations for all input test images, since the activation can be zeroed out. However, we do find that unlike capsules, individual convolution units are not able to model dramatically different deformations, and either do not activate or exaggerate a fixed type of attribute or attributes exclusively. This is not surprising given that convolution units have scalar outputs, while each capsules has a vector output that is capable of representing more complex information. As a result while a single capsule is able to have high activation with downturned, puckered, open, and smiling - among others - mouths, the convolution unit maximizes activation with thin mouths with a sharp right upturn regardless of the input image.

4.6. To do and questions

Add CVPR 2018 related work and baselines. Add disfa results and improved Ours-VGG results Add results showing overfitting on Ours vggf. Add cosine similarity numbers in an attractive way.

Questions: How to best show that after activation max, the capsules become more orthogonal when the resulting image is different, and less orthogonal when it is the same? Cosine similarity numbers show this, but i'm not sure how to add them to the figures. I was thinking of a bar under the images with the cosine similarity values before and after max.

Should we add ECCV18 numbers? Would it count as concurrent work because one of the wacv deadlines was before eccv?

I've removed all the localization routing experiments. Do you think that's ok?

5. Conclusions

In this paper, we tested the hypothesis that capsule networks are able to model local part deformations in faces well. We tested this hypothesis by using capsules for action unit detection, and found that capsules are indeed able to model action unit activations well. Our results demonstrated state-of-the-art results on action unit detection on two widely-used datasets. While previous work has shown that capsules are able to model global deformations, we showed that capsules can also capture local deformations. This indicates that capsule networks will also be useful for other tasks where parts of an object need to be modeled well - such as fine-grained classification, or human pose estimation and tracking. At the same time, our results can be useful for the action unit and facial expression understanding community. We propose a novel architecture for action unit detection, and push the state-of-the-art numbers in the area.

In the future, we plan to work on automatic animal facial expression understanding. For a setting such as animal facial expression understanding where data is scarce and difficult to both collect and annotate, it becomes critical to work with models such as capsule networks that are able to extract rich feature representations with fewer overall parameters. In addition, the added ease with which capsule properties can be visualized makes capsule networks an appropriate model working with limited, possibly noisily annotated data.

References

[1] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. Recognizing facial expression: machine learning and application to spontaneous behavior. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 568–573. IEEE, 2005. 1, 3

[2] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision*, 2017. 4

[3] W.-S. Chu, F. De la Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3515–3522. IEEE, 2013. 2

[4] W.-S. Chu, F. De la Torre, and J. F. Cohn. Modeling spatial and temporal cues for multi-label facial action unit detection. *arXiv preprint arXiv:1608.00911*, 2016. 4, 5

[5] C. A. Corneanu, M. Madadi, and S. Escalera. Deep structure inference network for facial action unit recognition. *ECCV*, 2018. 3

[6] H. Ding, S. K. Zhou, and R. Chellappa. Facenet2expnet: Regularizing a deep face recognition net for expression recognition. *Automatic Face and Gesture Recognition (FG)*, 2017. 3, 5

[7] S. Eleftheriadis, O. Rudovic, and M. 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. 2

[8] X. Feng, M. Pietikäinen, and A. Hadid. Facial expression recognition based on local binary patterns. *Pattern Recognition and Image Analysis*, 17(4):592–598, 2007. 3

[9] S. Ghosh, E. Laksana, S. Scherer, and L.-P. Morency. A multi-label convolutional neural network approach to cross-domain action unit detection. In *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*, pages 609–615. IEEE, 2015. 2

[10] A. Gudi, H. E. Tasli, T. M. den Uyl, and A. Maroulis. Deep learning based facs action unit occurrence and intensity estimation. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–5. IEEE, 2015. 2

[11] S. Han, Z. Meng, A.-S. Khan, and Y. Tong. Incremental boosting convolutional neural network for facial action unit recognition. In *Advances in Neural Information Processing Systems*, pages 109–117, 2016. 2

[12] S. Han, Z. Meng, Z. Li, J. O'Reilly, J. Cai, X. Wang, and Y. Tong. Optimizing filter size in convolutional neural networks for facial action unit recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[13] S. Hinton, Geoffrey E snd Sabour and N. Frosst. Matrix capsules with em routing. In *ICLR*, 2018. 1, 2

[14] S. Jaiswal and M. Valstar. Deep learning the dynamic appearance and shape of facial action units. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–8. IEEE, 2016. 2, 4

[15] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2983–2991, 2015. 3

[16] P. Khorrami, T. Paine, and T. Huang. Do deep neural networks learn facial action units when doing expression recognition? In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 19–27, 2015. 3, 5

[17] Y. Kim, B. Yoo, Y. Kwak, C. Choi, and J. Kim. Deep generative-contrastive networks for facial expression recognition. *arXiv preprint arXiv:1703.07140*, 2017. 3, 5

[18] S. Koelstra, M. Pantic, and I. Patras. A dynamic texture-based approach to recognition of facial actions and their temporal models. *IEEE transactions on pattern analysis and machine intelligence*, 32(11):1940–1954, 2010. 2

[19] W. Li, F. Abtahi, and Z. Zhu. Action unit detection with region adaptation, multi-labeling learning and optimal temporal fusing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 2, 4

[20] Y. Li, J. Chen, Y. Zhao, and Q. Ji. Data-free prior model for facial action unit recognition. *IEEE Transactions on affective computing*, 4(2):127–141, 2013. 2

[21] M. Liu, S. Li, S. Shan, and X. Chen. Au-aware deep networks for facial expression recognition. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–6. IEEE, 2013. 3, 5

[22] M. Liu, S. Li, S. Shan, and X. Chen. Au-inspired deep networks for facial expression feature learning. *Neurocomputing*, 159:126–136, 2015. 3, 5

[23] P. Liu, S. Han, Z. Meng, and Y. Tong. Facial expression recognition via a boosted deep belief network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014. 3

[24] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010. 5

[25] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn. Disfa: A spontaneous facial action intensity database. *IEEE Transactions on Affective Computing*, 4(2):151–160, 2013. 4

[26] A. Mollahosseini, D. Chan, and M. H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016. 3

[27] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. 3

[28] G. Peng and S. Wang. Weakly supervised facial action unit recognition through adversarial training. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[29] S. Sabour, N. Frosst, and G. E. Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, pages 3859–3869, 2017. 1, 2, 3

[30] Z. Shao, Z. Liu, J. Cai, and L. Ma. Deep adaptive attention for joint facial action unit detection and face alignment. *ECCV*, 2018. 3

[31] K. Sikka, T. Wu, J. Susskind, and M. Bartlett. Exploring bag of words architectures in the facial expression domain. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 250–259. Springer, 2012. 3

[32] Y. Tong, W. Liao, and Q. Ji. Facial action unit recognition by exploiting their dynamic and semantic relationships. *IEEE transactions on pattern analysis and machine intelligence*, 29(10), 2007. 2

[33] M. F. Valstar, T. Almaev, J. M. Girard, G. McKeown, M. Mehu, L. Yin, M. Pantic, and J. F. Cohn. Fera 2015-second facial expression recognition and analysis challenge. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–8. IEEE, 2015. 2

[34] Z. Wang, Y. Li, S. Wang, and Q. Ji. Capturing global semantic relationships for facial action unit recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3304–3311, 2013. 2

[35] H. Yang, U. Ciftci, and L. Yin. Facial expression recognition by de-expression residue learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[36] J. Zeng, W.-S. Chu, F. De la Torre, J. F. Cohn, and Z. Xiong. Confidence preserving machine for facial action unit detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3622–3630, 2015. 2, 4

[37] X. Zhang and M. H. Mahoor. Simultaneous detection of multiple facial action units via hierarchical task structure learning. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 1863–1868. IEEE, 2014. 2

[38] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard. Bp4d-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. *Image and Vision Computing*, 32(10):692–706, 2014. 4

[39] Y. Zhang, W. Dong, B.-G. Hu, and Q. Ji. Classifier learning with prior probabilities for facial action unit recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[40] Y. Zhang, W. Dong, B.-G. Hu, and Q. Ji. Weakly-supervised deep convolutional neural network learning for facial action unit intensity estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[41] Y. Zhang, R. Zhao, W. Dong, B.-G. Hu, and Q. Ji. Bilateral ordinal relevance multi-instance regression for facial action unit intensity estimation. In *CVPR*, June 2018. 3

[42] K. Zhao, W.-S. Chu, F. De la Torre, J. F. Cohn, and H. Zhang. Joint patch and multi-label learning for facial action unit detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2207–2216, 2015. 2, 4

[43] K. Zhao, W.-S. Chu, and H. Zhang. Deep region and multi-label learning for facial action unit detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3391–3399, 2016. 1, 2, 4

[44] X. Zhao, X. Liang, L. Liu, T. Li, Y. Han, N. Vasconcelos, and S. Yan. Peak-piloted deep network for facial expression recognition. In *European Conference on Computer Vision*, pages 425–442. Springer, 2016. 3

[45] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. N. Metaxas. Learning active facial patches for expression analysis. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2562–2569. IEEE, 2012. 3

[46] Y. Zhu, S. Wang, L. Yue, and Q. Ji. Multiple-facial action unit recognition by shared feature learning and semantic relation modeling. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 1663–1668. IEEE, 2014. 2