# A Multi-label Convolutional Neural Network Approach to Cross-Domain Action Unit Detection

Sayan Ghosh[1], Eugene Laksana[1], Stefan Scherer[1], Louis-Philippe Morency[2]

Institute for Creative Technologies, University of Southern California[1]
12015 E Waterfront Dr, Los Angeles, CA, USA
{sghosh, elaksana, scherer}@ict.usc.edu

Language Technologies Institute, Carnegie Mellon University[2]
5000 Forbes Ave, Pittsburgh, PA, USA
morency@cs.cmu.edu

***Abstract*—Action Unit (AU) detection from facial images is an important classification task in affective computing. However most existing approaches use carefully engineered feature extractors along with off-the-shelf classifiers. There has also been less focus on how well classifiers generalize when tested on different datasets. In our paper, we propose a multi-label convolutional neural network approach to learn a shared representation between multiple AUs directly from the input image. Experiments on three AU datasets- CK+, DISFA and BP4D indicate that our approach obtains competitive results on all datasets. Cross-dataset experiments also indicate that the network generalizes well to other datasets, even when under different training and testing conditions.**

***Keywords—Convolutional Neural Networks; Action Units; Cross-dataset transfer***

## I. INTRODUCTION

One of the most important classification tasks in affective computing is the automatic detection of expressions from facial images. Humans communicate through exchange of verbal and non-verbal messages in interactions. Facial expressions are an important component of human communication dynamics, and prior research shows that they convey valuable information about emotional state and social interaction [1]. Recognition of facial expressions, such as anger, fear, joy or surprise is facilitated by the detection of individual, or a combination of facial muscle movements, which are commonly referred to as Action Units (AUs) [2]. The FACS (Facial Action Coding System) introduced by Ekman et al. [1] is a principled standard to categorize facial muscle movements. The manual encoding of FACS labels is highly time consuming and expensive, which necessitates the automatic detection of AUs using machine learning algorithms.

There has been a great deal of work on automated approaches for recognition of AUs from still images of faces, as well as from videos [3]. AU recognition can be posed as a classification task, where the presence/absence of an AU is a binary decision, or a regression task where the intensity of each AU is estimated. In our paper, we focus on the subject independent classification of AU presence/absence from videos. Most existing approaches [3][4] employ standard off-the-shelf classifiers with feature extractors such as SIFT (Scale-Invariant Feature Transform), HoG (Histogram of Gradients), and LBP (Local Binary Patterns) along with a classifier such as SVM (Support Vector Machine) for detecting action units. Further, the recognition of AUs is posed as a set of binary classification problems, where the detection of each AU is done separately.

When evaluating across AU technologies, it is important to test across domains and datasets to evaluate generalization. Unfortunately, the conventional methodology is to train and test on the same dataset. While there has been some effort to investigate the cross-domain performance of AU detection algorithms [5][6] there has been no formal study of their generalization properties when different datasets are used for training and testing. It would be useful to characterize the performance under different conditions such as lighting, subject independence, and acted/spontaneous facial expressions. This is particularly useful for "in the wild" scenarios where AU detection is performed on real-world camera captures without any prior knowledge of the environment or settings in which the data is acquired. A suitable example would be YouTube videos, where no additional information about the video stream is available.

Our contributions in this paper are novel in two aspects. Firstly, we explore the use of convolutional neural networks (CNNs) for the task of AU detection. We are motivated by the widespread success of deep learning (and particularly CNNs) for large-scale image recognition [7]. We pose the problem of AU recognition as a multi-label classification problem, and instead of training a separate CNN for each AU, jointly train all the AUs through a multi-label softmax classification loss approach. We obtain competitive results on various datasets, including CK+ (Cohn-Kanade extended) [4], DISFA (Denver Intensity of Spontaneous Facial Actions) [8], and BP4D [9], which is part of the FERA 2015 (Facial Expression and Analysis Challenge). Secondly, we explore the effect of domain mismatch in the context of cross-dataset training and testing to obtain an insight into the robustness of these networks. Our results, along with visualization experiments indicate that CNNs obtain good performance even under different training and testing conditions.

## II. RELATED WORK

Since the introduction of FACS (Facial Action Coding System) by Ekman et al. in [1], there have been a number of efforts in the literature for the automatic recognition of facial AUs. Tian et al. [10] proposed an AFA (Automatic Face Analysis) system for

978-1-4799-9953-8/15/$31.00 ©2015 IEEE

facial feature modeling and evaluated their approach on the Cohn-Kanade dataset. Bartlett et al. [11] compared several learning algorithms, such as AdaBoost, SVMs (Support Vector Machines), and Linear Discriminant Analysis to classify 17 AUs on the Cohn-Kanade dataset. Littlewort et al [12] introduced CERT (Computer Expression Recognition Toolbox) for estimating AU intensities using SVM classifiers.

Wu et al. [13] explored multilayer architectures using GEFs (Gabor Energy Filters) for AU recognition. Zhang et al. [14] modeled AU recognition using the Lp norm MTMKL (Multi-Task Multiple Kernel Learning) framework. Li et al. [15] used DBNs (Dynamic Bayesian Networks) for modeling correlations among AUs and estimating their intensities. Song et al. [16] modeled AU sparsity and co-occurrence using a Bayesian compressed sensing model, reporting 86% accuracy on the DISFA dataset, and 94% on the Cohn-Kanade extended (CK+) dataset. Jiang et al. [17] explore the problem of which face regions features should be extracted from and propose a decision-level fusion with LBP (Local Binary Pattern) and LPQ (Local Phase Quantization) to obtain a best performance of 0.81 2AFC (Two-alternative Forced Choice) score on the DISFA dataset.

However, there has been very little work in the literature on neural networks for AU recognition. Tian et al. in [35] employed a three-layer neural network with one hidden layer for identifying lower face AUs. Handcrafted facial features such as lip height, width and corners; transient features such as left nasio-labal furrow angle, and presence of nose wrinkles were used as inputs to the neural network. Fasel [18] explored convolutional neural networks (CNN) for multi-scale facial expression recognition. These previous methods use a combination of hand-engineered feature extractors along with standard off-the-shelf machine learning models ; to the best of our knowledge, our approach is the first one in which the facial image is used directly as an input to a deep network for learning a shared representation of AUs for a subsequent classification task.

CNNs were applied to document recognition by LeCun et al. [19], and have enjoyed a resurgence with access to ever-increasing computational resources, and access to web-scale image and video data from media websites such as YouTube and Flickr. Krizhevsky et al. [7] used CNNs for classification of images from LSVRC-2010 ImageNet dataset. Data augmentation through random distortions, ReLU (Rectified Linear Units) and dropout layers were proposed [20] to improve generalization capabilities of these networks. In the domain of affect recognition, the FER (Facial Expression Recognition) 2013 challenge [21] was launched on Kaggle for identifying facial expressions from images, where the winning entry used a CNN with an L-1 SVM as a post-classifier using features extracted from the top layer, obtaining an accuracy of 69% on the test dataset. Deep CNNs have also been applied recently to video classification [35]. This recent success has motivated us to explore the use of CNNs for learning representations relevant to AU classification.

## III. DATASETS

We perform our experiments on three datasets: Extended Cohn-Kanade dataset (CK+), DISFA (Denver Intensity of Spontaneous Facial Expressions), and the BP4D-Spontaneous dataset, which is a part of the FERA 2015 Challenge on AU detection. We choose CK+, since it is a standard dataset for AU detection, DISFA since it is a video dataset, and BP4D since it is a very new dataset with accurate annotations. While performing cross dataset training, we only consider 10 AUs which all three datasets share, which are: 1 (Inner brow raiser), 2 (Outer brow raiser), 4 (Brow lowerer), 5 (Upper Lid raiser), 6 (Cheek raiser), 9 (Nose wrinkler), 12 (Lip corner puller), 15 (Lip corner depressor), 17 (Chin raiser) and 20 (Lip stretcher).

**CK+ Dataset [4]**: The CK+ dataset consists of 582 fully FACS coded image sequences of both spontaneous and nonspontaneous facial expressions from 123 subjects. Each frame containing an AU was either given an intensity degree on a 7-point ordinal scale or an "unspecified intensity" label. The lack of an AU label indicates absence. Since we pose AU detection as a multi-label binary classification problem, we mapped any frame containing an intensity of 3 or higher to a positive presence label. The CK+ dataset was substantially smaller than DISFA or BP4D. Hence we used it strictly for the testing phase in cross dataset experimentation.

**DISFA Dataset [8]**: The DISFA dataset contains stereo videos of 27 subjects spontaneously generating facial expressions when watching an emotive video stimulus, with a total of 54 videos. Each video consists of 4845 FACS coded frames for 12 AUs: 25 (Lip part) and 26 (Jaw Drop) in addition to the above, with presence, absence, and intensity labels. For this experiment, we only used presence and absence labels.

**FERA 2015 BP4D-Spontaneous Dataset [9]**: This partition of BP4D was part of the 2015 FERA challenge dataset. It consists of 41 subjects and 34 AUs, and the subjects were young adults who spontaneously generated emotional responses to stimulus tasks. For this experiment, we only used the 10 common AUs shared between the CK+ and DISFA datasets. Binary presence and absence labels were available, so these were straight-forwardly parsed without thresholding the AU intensities for every frame.

## IV. REPRESENTATION LEARNING OF ACTION UNITS

Convolutional neural networks (CNNs) are very effective at learning image representations with shift-invariant features, directly from a raw input image without any extraction of handcrafted features. This is also referred to as "representation learning" and it also generalizes well to unseen data. These properties motivate us to explore CNNs for AU recognition. We use a multi-label loss function since each facial image could have more than one AU associated with it, and train the CNN with this loss function.

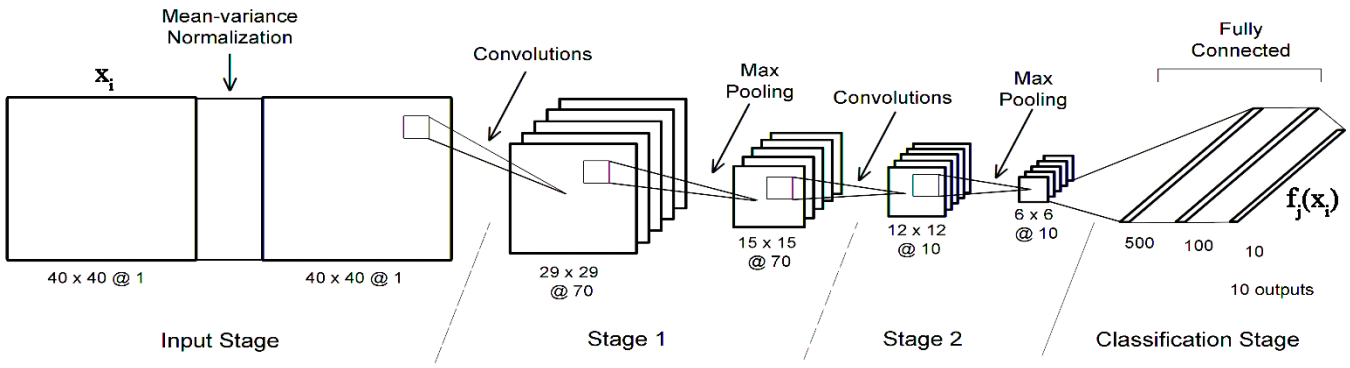



Fig 1. Architecture of the CNN trained on the BP4D dataset. There are two convolutional stages and two fully-connected layers interspersed by max-pooling layers, with the size of the output layer equal to the number of AUs.

*Multi-label softmax classification-:* We pose the problem of detecting AUs as a multi-label binary classification task. Let us assume that there are $C$ AU categories, $N$ data points $\{\boldsymbol{x_i}\}$, where each $\boldsymbol{x_i}$ denotes the $i$-th image and $y_{ij} \in \{0,1\}$ is a label denoting presence or absence of the *j-th* AU. Let $f(\boldsymbol{x_i})$ be the transformation learnt by the neural network at the hidden layer just prior to computation of the loss function. Similar to [22], we define the multi-label softmax classification loss as:

$$J = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{p}_{ij}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C_+}\log(\hat{p}_{ij}) \qquad (1)$$

Where $C_+$is the number of AUs actually present in image $\boldsymbol{x_i}$, $\hat{p}_{ij}$ are the prediction probabilities from the last softmax layer for the $i$-th input image and the $j$-th AU, obtained as:

$$\hat{p}_{ij} = \frac{\exp\left(f_j(\boldsymbol{x_i})\right)}{\sum \exp\left(f_j(\boldsymbol{x_i})\right)} \qquad (2)$$

Our experimental results, as described in Section VI show that the joint representations learnt by the network are effective at learning correlations among the AUs.

*CNN architecture-:* We use the open-source Caffe toolbox [23] for our experiments. Since the multi-label softmax regression is not built in by default, we created a new layer for this task. The CNN contains a large number of parameters, and since we have limited participant data, we used three techniques to avoid over-fitting the dataset: (1) two dropout layers between fully connected ones (2) data augmentation by random mirroring and cropping when creating data batches for the CNN, and (3) usage of L2 regularization with a suitable validated learning rate. Fig.1 shows the architecture of the CNN trained on the BP4D dataset. Our preliminary experiments also indicate that a Mean-Variance Normalization layer improves detection performance. After the CNN is trained, it still remains an interesting problem to map the prediction probabilities $\{p_{ij}\}$ into a set of binary labels.

*AU classification-:* We use the learnt top layer representations for each data point as training data for a QDA (Quadratic Discriminant Analysis) [24] classifier to predict each AU. QDA is a probabilistic classifier with a quadratic decision boundary. Its usage was motivated by two design considerations. Firstly, QDA requires the classifier to have minimal tuning, since it is probabilistic in nature, without any hyper-parameters, and thus additional validation is not required (unlike a hyper-parameter based classifier such as an SVM or a Maximum Entropy Classifier). Secondly, the usage of a simple classifier helps us obtain an insight into the efficacy of the representations learnt by the CNN.

Assuming $\boldsymbol{h_i}$ is the feature vector (representing the $i$-th test example) obtained from the last fully-connected layer of the network and that $y_{ij}$ has a value of 0/1 depending on the respective absence or presence of the j-th AU for the image example, each classifier estimates the posterior probability $P(y_{ij}|\boldsymbol{h_i})$ and predicts the class with the higher likelihood.

*Data imbalance-:* The sparsity of the AU occurrences also poses an additional challenge. Previous work [25] in the literature reports that performance improvements are obtained by balancing the dataset prior to training. However, unlike in a single label setting, balancing one AU may result in another AU being unbalanced, since we consider all AUs jointly in the multi-label loss function. We solved this issue by trying out different approaches to multi-label balancing, so that the fraction of positive/negative classes across all AUs is as balanced as possible. While leading to some improvement overall, this did not result in any significant performance improvements.

## V. EXPERIMENTAL METHODOLOGY

*Preprocessing the datasets-:* The datasets were preprocessed, with an OpenCV face detector (frontal) being applied on every frame to obtain the bounding box of the face. Subsequently, the face was cropped out. We assume that multiple facial images of each participant are available to us for training and testing, where the mean facial image of each participant can be computed for mean normalization. This technique is similar to subtraction of a person's face from his "neutral" face to obtain a subject-independent analysis of expressions. It has been widely used in the literature [26], and we assume that the mean face averaged over all frames is very similar to the neutral face. This also removes the need for a separate neutral facial image to aid the classification. After subject specific normalization,

TABLE I. LEAVE-ONE-OUT CLASSIFICATION PERFORMANCE OF MULTI-LABEL CNN ON DISFA DATASET (12 ACTION UNITS)

| Action Unit | Classification Performance | |
|---|---|---|
| | *Accuracy* | *2AFC Score* |
| 1 | 87.5 | 0.702 |
| 2 | 88.6 | 0.715 |
| 4 | 78.5 | 0.704 |
| 5 | 90.6 | 0.773 |
| 6 | 83.9 | 0.855 |
| 9 | 90.2 | 0.829 |
| 12 | 86.1 | 0.913 |
| 15 | 86.7 | 0.683 |
| 17 | 81.4 | 0.679 |
| 20 | 88.4 | 0.639 |
| 25 | 78.5 | 0.838 |
| 26 | 74.9 | 0.757 |
| **Average** | 84.6 | 0.757 |
| **Baselines** [16][29] | 86.8 | 0.76 |

we also normalize each facial image using the entire dataset, and divide each pixel's intensity by 255 to keep it between 0 and 1. It is worth noting that this is a simple normalization method, and usage of advanced methods such as facial landmarks [10] would definitely improve performance.

*Train/validation set splits-:* We employed a leave-one-subject-out testing scheme, and split the DISFA and BP4D datasets into training, validation and testing sets, where. For DISFA, in each testing fold, 75% of the subjects were used for training, and the remaining 25% for validation according to a random seed. All experiments were performed in a subject independent manner for all datasets. For BP4D a similar scheme was used, except that the train-validation split was done according to that predefined in the FERA 2015 challenge guidelines [9]. With the CK+ dataset, we performed only testing, since the number of images in the dataset are inadequate to train the large number of weights in the network (~100,000) without risk of over-fitting. It also helps us measure the generalization ability of the network, and compare with baseline results reported in existing literature.

*Hyper-parameter Validation-*: The relevant hyper-parameters to tune for the CNN, along with search range were (1) Optimal training iterations – the number of iterations beyond which over-fitting occurs (5000 to 10000) (2) Base learning rate of the network (0.0001 to 0.01) (3) Weight decay parameter (5e-3 to 5e-6) (4a) Kernel size for the convolutional layers (5 to 15) (4b) Kernel size for the pooling layers (2 to 4) (5) Learning momentum (0.5 to 0.9). A detailed description of the hyper-parameters can be found in [27]. Due to the large number of hyper-parameters involved, we have resorted to a random search over the hyper-parameter space in place of a grid search. It has also been reported in the literature, that random search is more beneficial in this setting [28]. We defined two *mega layers*, each consisting of a convolutional, ReLU (Rectified Linear Unit), dropout layer and max-pooling layer, with two fully-connected layers on top. After validation is complete, we retrain the network with a combination of training and validation data prior to leave-one-out testing. Early stopping is used for training, with the optimal model being selected 1000 iterations after the validation error starts increasing. We extracted the top layer representation $\{\boldsymbol{h_i}\}$ from the trained CNN, and for each AU, we trained a QDA (Quadratic Discriminant Analysis) classifier to predict presence/absence of the AU in the frame of interest.

TABLE II. LEAVE-ONE-OUT TEST CLASSIFICATION AND HELD-OUT DEVELOPMENT PERFORMANCE OF MULTI-LABEL CNN ON BP4D DATASET (10 ACTION UNITS)

| Action Unit | Classification Performance | |
|---|---|---|
| | *Accuracy* | *2AFC Score* |
| 1 | 70.85 | 0.643 |
| 2 | 76.11 | 0.675 |
| 4 | 73.53 | 0.720 |
| 5 | 85.49 | 0.619 |
| 6 | 73.82 | 0.839 |
| 9 | 81.36 | 0.763 |
| 12 | 79.49 | 0.878 |
| 15 | 65.97 | 0.642 |
| 17 | 65.25 | 0.665 |
| 20 | 86.08 | 0.728 |
| **Average** | 75.80 | 0.717 |
| **7-AU subset average on dev** | 75.69 | 0.751 |
| **FERA 2015 Baseline [9]** | 72.00 | 0.427 |

*Performance Metrics-:* To measure the performance of the CNN classifier on the task of AU detection, we use two measures – (1) Accuracy and (2) 2AFC (Two-alternative Forced Choice) score. We were motivated to use these metrics due to the ease of comparison with baselines in existing literature, and our concern that the choice of evaluation metrics should be insensitive to the amount of skew in the testing set, as reported by Jeni et al. in [30]. The accuracy is equal to the percentage of testing examples correctly classified, while the 2AFC score is the fraction of correctly classified examples in a 2AFC trial experiment, which has been shown to be a good approximation to the AUC score (Area under ROC curve) [9].

## VI. RESULTS

In this section, we present the results of experiments for two paradigms: dataset specific and cross-dataset evaluation. We also perform visualization experiments to obtain an insight into the representations learnt by the CNN, and its ability to generalize across datasets.

*Dataset specific evaluation-:* The methodology of these experiments are explained in Section V and in this section, we present results in a subject independent manner. In Table I, we show the accuracy and 2AFC scores for each of the 12 AUs in

TABLE III. CROSS-DATASET GENERALIZATION PERFORMANCE OF MULTI-LABEL CNN

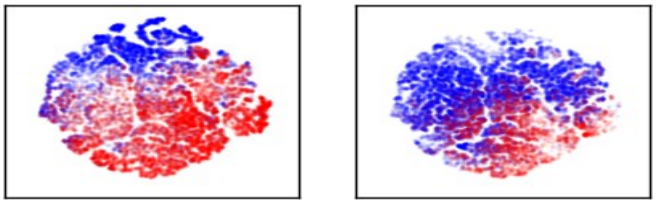
| Action Unit | Classification Performance | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | *BP4D to CK+* | | *BP4D to DISFA* | | *DISFA to CK+* | | *DISFA to BP4D* | |
| | *Accuracy* | *2AFC score* | *Accuracy* | *2AFC score* | *Accuracy* | *2AFC score* | *Accuracy* | *2AFC score* |
| 1 | 86.05 | 0.791 | 83.81 | 0.66 | 85.21 | 0.739 | 75.91 | 0.676 |
| 2 | 85.11 | 0.860 | 81.17 | 0.717 | 86.88 | 0.786 | 79.69 | 0.637 |
| 4 | 79.94 | 0.730 | 83.26 | 0.740 | 75.58 | 0.669 | 70.95 | 0.678 |
| 5 | 93.11 | 0.723 | 92.22 | 0.762 | 93.73 | 0.790 | 92.57 | 0.706 |
| 6 | 61.87 | 0.723 | 70.33 | 0.870 | 85.64 | 0.729 | 69.78 | 0.818 |
| 9 | 85.37 | 0.949 | 90.45 | 0.841 | 93.42 | 0.974 | 83.87 | 0.794 |
| 12 | 55.42 | 0.889 | 62.36 | 0.873 | 91.07 | 0.887 | 63.63 | 0.838 |
| 15 | 75.59 | 0.493 | 75.17 | 0.617 | 85.13 | 0.618 | 76.67 | 0.597 |
| 17 | 63.08 | 0.636 | 68.85 | 0.585 | 88.14 | 0.803 | 66.92 | 0.634 |
| 20 | 92.16 | 0.798 | 95.79 | 0.635 | 94.59 | 0.807 | 93.35 | 0.525 |
| **Average** | 77.77 | 0.759 | 80.34 | 0.730 | 88.14 | 0.780 | 77.34 | 0.690 |

the DISFA dataset. We compare our results to papers [16] and [29], where we have ensured that the same experimental methodology and choice of metrics are followed for a fair comparison. From an examination of Table I, we find that our approach performs best on AUs 12 (Lip corner puller) and 20 (Lip stretcher) in terms of 2AFC scores. The average accuracy over all the twelve AUs is very close to the state-of-the-art performance as reported in [16]. Further, the average 2AFC score (0.757) is comparable to the performance reported in [17], where the authors use region-specific features, along with prior knowledge of AU muscle contractions to obtain average 2AFC scores of 0.77 for feature level fusion, and 0.805 for decision-level-fusion. It is worth noting that prior work uses carefully hand-engineered features, whereas our approach learns relevant features directly from the facial image with minimal preprocessing (simple face detection and normalization) to achieve similar performance.

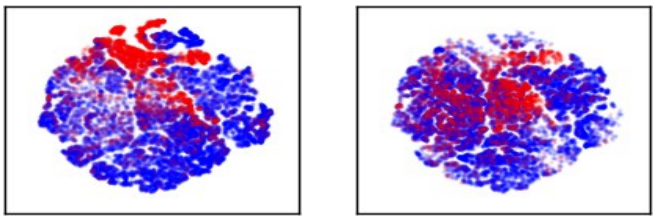
In Table II, we present leave-one-out testing accuracies and 2AFC scores for a subset of 10 AUs (1,2,4,5,6,9,12,15,17,20) from the BP4D dataset obtained as training and development data for the FERA 2015 challenge [9]. For evaluation, we compare our development set results with the results reported in the baseline paper [9], for a subset of common AUs (1, 2, 4, 6, 12, 15 and 17), since the organizers have not yet released the full test dataset for the challenge. We obtain an average development set accuracy of 75.69%, and an average 2AFC score of 0.7518, which respectively outperform the baseline scores of 72% (Geometric features), and 63.2% (Appearance features) for the accuracy; and 0.357 (Geometric features) and 0.427 (Appearance features) for the 2AFC score.

*Cross-dataset Evaluation-:* Our hypothesis is that the CNN will learn features which are discriminative between presence/absence of AUs not only on the training dataset, but also on unseen datasets. To evaluate the generalization ability of our network we perform training and validation on DISFA and BP4D datasets, and test them on a different dataset. The results are presented in Table III. We observe that since the training dataset differs from the evaluation dataset, the accuracy and 2AFC scores are generally less than those obtained with same-dataset testing. The network trained on DISFA generalizes well to BP4D, where a relative accuracy improvement of 1.54% is obtained over the BP4D-only baseline network with a slight decrease of 0.02 in the 2AFC score. The performance obtained on the CK+ dataset (accuracy 88.14% and 2AFC 0.78) is similar to that reported in [5] (accuracy 80.56% and 2AFC 0.80), which is also in a cross dataset setting (training on MMI database [32] and testing on CK+) showing that our approach is robust across datasets. When trained on BP4D and tested on CK+, the network does not generalize as well, obtaining an average accuracy of 77.7%, and a 2AFC score of 0.759. However, the testing results on DISFA are close to those obtained when trained entirely on it. Thus, we can conclude that our network performs well even when trained and tested across different datasets.

*Visualization of CNN features:* To obtain an insight into the joint representation which the CNN learns from the input images and its generalization ability on unseen data, we present scatter plots of the representation $\{\boldsymbol{h_i}\}$ extracted from the last fully connected layer of the network. We reduce the dimensionality of $\{\boldsymbol{h_i}\}$ using t-SNE [33] (t-Stochastic Neighbor Embedding). We train the CNN on the BP4D dataset, and extract $\{\boldsymbol{h_i}\}$, both for the BP4D development set and the DISFA test set for AUs 12 (Lip corner puller) and 15 (Lip corner depressor). The scatter plots are shown in Figure 2. Data points corresponding to presence of an AU are marked with red and absences are marked blue respectively. Figure 3 shows scatter plots corresponding to a network trained and $\{\boldsymbol{h_i}\}$ extracted for the DISFA development set and the BP4D test set. We observe that the representations are discriminative of the AU presence, including those from the test set, of which the network has no knowledge during training phase.

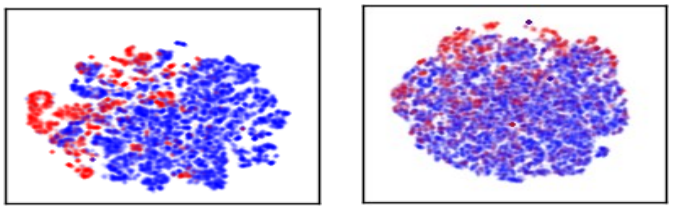


AU 12: (a) BP4D development (b) DISFA testing set

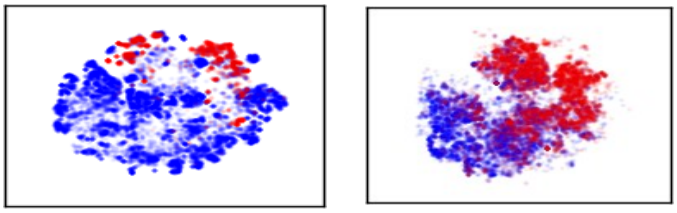


AU 15: (a) BP4D development (b) DISFA testing set

Fig. 2. Scatter plots representing BP4D development set and DISFA testing set when the CNN is trained on BP4D and tested on DISFA. The learned representations are discriminative of AU presence/absence



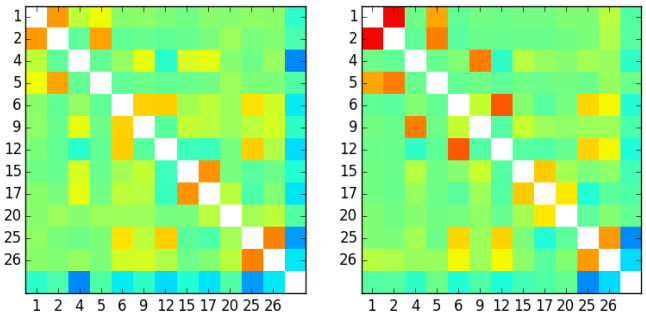
AU 6: (a) DISFA development (b) BP4D testing set



AU 12: (a) DISFA development (b) BP4D testing set

Fig. 3. Scatter plots representing DISFA development set and BP4D testing set when the CNN is trained on DISFA and tested on BP4D.

*Visualization of correlations between AUs:* In Figure 4, we present a heat map of pairwise correlations among the 12 AUs for the development partition of the DISFA dataset, constructed using a 75%-25% split. As described in Section III, we augment our label set with a 13$^{th}$ label to indicate neutral facial expressions, before performing training and validation of the network. Since label presence/absence is binary, we use the Phi-coefficient [34] as a measure of AU correlation. Figures 4(a) and 4(b) respectively show the correlations measured from the ground truth labels, and the prediction labels generated by our approach. The heat-maps are very similar, showing that the CNN is able to learn a good approximation of the ground truth correlations. For example, AUs 1 (Inner brow raiser) and 2 (Outer brow raiser) are highly correlated in Fig. 4(b). Further, the 13$^{th}$ label (neutral) is negatively correlated with the other AUs, which is evident in Figure 4(b).




(a) Ground truth correlations (b) Predicted correlations

Fig 4. Pairwise correlations among AUs for ground truth and predicted labels (blue indicates -1 phi-index, red indicates +1). The similarity between (a) and (b) shows that the CNN learns correlations among AUs.

## VII. Conclusion and Future Work

In this paper, we propose a multi-label convolutional neural network based approach to the task of AU detection from facial images. Our approach is inherently "multi-task", where a network learns a shared representation among different AUs, and each AU-specific classifier can utilize the learnt features for effective prediction. Our experiments indicate that convolutional neural networks are not only good at learning discriminative features for the same dataset, but that the learnt features are also robust when tested on different datasets. Further, the network also learns correlations among the AUs, thus obviating the need to design and train multiple networks for each AU. As future work, we plan to extend our approach to temporal modeling of AU occurrences in a multi-label setting, and regression of AU intensities from facial images.

## Acknowledgment

The work depicted here is sponsored by the U.S. Army Research Laboratory (ARL) under contract number W911NF-14-D-0005 and the National Science Foundation under Grant No. IIS-1118018. Statements and opinions expressed and content included do not necessarily reflect the position or the policy of the National Science Foundation or the Government and no official endorsement should be inferred. Sayan Ghosh also acknowledges the USC Viterbi Graduate School Fellowship for funding his graduate studies. The Tesla K40 used for this research was donated by the NVIDIA Corporation.

## REFERENCES


[1] P. Ekman, and W. V. Friesen. "Facial action coding system." 1977.

[2] A. Vinciarelli, M Pantic, and H Bourlard. "Social signal processing: Survey of an emerging domain." *Image and Vision Computing,* 27.12 (2009)

[3] Z. Zeng, M. Pantic, G. I. Roisman, & T. S Huang. "A survey of affect recognition methods: Audio, visual, and spontaneous expressions." *IEEE Transactions on Pattern Analysis and Machine Intelligence*. pp (39-58.) 2009.

[4] J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, & I. Matthews. "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression." *2010 IEEE Computer Society Conference on*. IEEE *Computer Vision and Pattern Recognition Workshops (CVPRW)*. June 2010.

[5] B. Jiang, B, M. Valstar, B. Martinez, and M. Pantic. "A dynamic appearance descriptor approach to facial actions temporal modeling." *IEEE Transactions on Cybernetics*. pp(161-174.) 2014.

[6] T. Wu, N. J. Butko, P. Ruvolo, J. Whitehill, M. S. Bartlett, and J. R. Movellan. "Action unit recognition transfer across datasets." *2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011)*. (pp. 889-896). IEEE. March 2011.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105). 2012.

[8] 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*. (pp. 151-160). 2013.

[9] M. Valstar, J. Girard, T. Almaev, G. McKeown, M. Mehu, L. Yin, ... & J. F. Cohn. "Fera 2015-second facial expression recognition and analysis challenge." *Proc. IEEE ICFG*. 2015.

[10] Y. Tian, T. Kanade, and J. F. Cohn. "Recognizing action units for facial expression analysis." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* (pp. 97-115.) 2001.

[11] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. Recognizing facial expression: machine learning and application to spontaneous behavior. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2005.* (Vol. 2, pp. 568-573). June 2005.

[12] G. Littlewort, J. Whitehill, T. Wu, I. Fasel, M. Frank, J. Movellan, and M. Bartlett. The computer expression recognition toolbox (CERT). *2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops.* (pp. 298-305). March 2011.

[13] T. Wu, M. S. Bartlett, and J. R. Movellan. Facial expression recognition using gabor motion energy filters. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).* (pp. 42-47). IEEE. June 2010.

[14] X. Zhang, M. H. Mahoor, S. M. Mavadati, and J. F. Cohn. A l p-norm MTMKL framework for simultaneous detection of multiple facial action units. *2014 IEEE Winter Conference on Applications of Computer Vision (WACV).* (pp. 1104-1111). March 2014.

[15] X. Li and Q. Ji. "Active affective state detection and user assistance with dynamic Bayesian networks." *IEEE Transactions on Systems, Man and Cybernetics Part A: Systems and Humans,* 2005.

[16] Y. Song, D. McDuff, D. Vasisht, and A. Kapoor. "Exploiting Sparsity and Co-occurrence Structure for Action Unit Recognition." *International Conference on Automatic Face and Gesture Recognition*. May 2015.

[17] B. Jiang, B. Martinez, M. F. Valstar, and M. Pantic. "Decision Level Fusion of Domain Specific Regions for Facial Action Recognition." 2014 IEEE *International Conference on Pattern Recognition (ICPR). (pp. 1776-1781).* August, 2014

[18] B. Fasel. "Head-pose invariant facial expression recognition using convolutional neural networks." *2002 IEEE International Conference on Multimodal Interfaces*. 2002.

[19] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, (pp. 541-551.) 1989.

[20] V. Nair and G.E. Hinton. "Rectified linear units improve restricted boltzmann machines." *2010 International Conference on Machine Learning*. 2010.

[21] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, and Y. Bengio. Challenges in representation learning: A report on three machine learning contests. *Neural Networks*. 2014.

[22] Y. Gong, Y. Jia, T. Leung, A. Toshev, & S. Ioffe. "Deep convolutional ranking for multilabel image annotation." *arXiv preprint arXiv:1312.4894*. (unpublished). 2013.

[23] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, and T. Darrell. "Caffe: Convolutional architecture for fast feature embedding." ACM International Conference on Multimedia(pp. 675-678). November 2014

[24] R. O. Duda, P. E. Hart, and D. G. Stork. "Pattern classification." John Wiley & Sons. 2012.

[25] G. Sandbach, S. Zafeiriou, and M. Pantic. "Local normal binary patterns for 3D facial action unit detection." IEEE International Conference on *Image Processing (ICIP)*. (pp. 1813-1816). IEEE. September 2012.

[26] Y. L. Tian, T. Kanade, and J. F. Cohn. "Facial expression analysis." In *Handbook of face recognition* (pp. 247-275). Springer New York. 2005.

[27] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. "Algorithms for hyper-parameter optimization." In *Advances in Neural Information Processing Systems.* (pp. 2546-2554). 2011.

[28] J. Bergstra, and Y. Bengio. "Random search for hyper-parameter optimization." *Journal of Machine Learning Research*. (pp. 281-305). 2012.

[29] B. Jiang, B. Martinez, M. F. Valstar, and M. Pantic. "Decision Level Fusion of Domain Specific Regions for Facial Action Recognition." 2014 IEEE International Conference on *Pattern Recognition (ICPR).* (pp. 1776-1781). IEEE. August 2014.

[30] L. A. Jeni, J. F. Cohn, and F. De La Torre. "Facing Imbalanced Data--Recommendations for the Use of Performance Metrics." *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII).* (pp. 245-251). IEEE. September.2013.

[31] "FERA 2015 - Second Facial Expression Recognition and Analysis Challenge." *The University of Nottinghom*. Web. 2015.

[32] M.F. Valstar, LM Maat, RC Sondak MF, and M. Pantic P. Gaia. "MMI Face Database." 2005.

[33] L.J.P. van der Maaten. "Accelerating t-SNE using Tree-Based Algorithms." *Journal of Machine Learning Research*. October 2014.

[34] Cramér, Harald. "Mathematical methods of statistics." Vol. 9. Princeton University Press. 1999.

[35] Y. L. Tian. "Evaluation of face resolution for expression analysis." 2004 IEEE International Conference on *Computer Vision and Pattern Recognition Workshop. CVPRW*. 2004