Deep Neural Networks with Relativity Learning for Facial Expression Recognition

Yanan Guo[1], Dapeng Tao[1], Jun Yu[2], Hao Xiong[3], Yaotang Li[1], Dacheng Tao[3]

[1]Yunnan University, China, [2]Hangzhou Dianzi University, China, [3]University of Tchnology Sydney, Australia

Yananguo.YNU@qq.com, dapeng.tao@gmail.com, yujun@hdu.edu.cn, hao.xiong@student.uts.edu.au, liyaotang@ynu.edu.cn, dacheng.tao@uts.edu.au.

ABSTRACT

Facial expression recognition aims to classify facial expression as one of seven basic emotions including "neutral". This is a difficult problem due to the complexity and subtlety of human facial expressions, but the technique is needed in important applications such as social interaction research. Deep learning methods have achieved state-of-the-art performance in many tasks including face recognition and person re-identification. Here we present a deep learning method termed Deep Neural Networks with Relativity Learning (DNNRL), which directly learns a mapping from original images to a Euclidean space, where relative distances correspond to a measure of facial expression similarity. DNNRL updates the model parameters according to sample importance. This strategy results in an adjustable and robust model. Experiments on two representative facial expression datasets (FER-2013 and SFEW 2.0) are performed to demonstrate the robustness and effectiveness of DNNRL.

Index Terms— Facial expression, deep feature learning, convolutional neural network, social interaction.

1 INTRODUCTION

Human emotion recognition is attracting attention due to its many practical applications in, for example, social interactions, human-robot interactions, and call center systems. Facial expression recognition (FER), a basic part of motion analysis, can be used to recognize internal human emotions. FER methods attempt to classify facial expression in a given image or sequence of images as one of six basic emotions (anger, disgust, fear, happiness, sadness, and surprise) or as "neutral" [4]. Although much effort and some progress has been made in the field, accurately recognizing facial expression remains difficult due to the complexity and subtlety of facial expressions and how they relate to emotion.

Most FER systems involve three main steps: face image pre-processing, feature extraction, and classification. Deriving an effective and robust facial representation from raw face images is extremely important for FER. In general, methods used to extract facial representations can be grouped into two categories: action unit (AU)-based methods and appearance-based methods. AU-based methods, originally proposed by Ekman et al. [4], detect the presence of individual AUs, which are then extracted to form a feature based on their combinations. However, each AU detector usually requires careful hand engineering (such as through reliable facial detection and tracking) to ensure good performance, which is hard to accommodate in many situations. Appearance-based methods [15] represent an individual's emotion from their facial shape and texture. Appearance-based methods apply traditional handcrafted features such as Gabor wavelets [10], Local Binary Pattern (LBP) features [12], and Motion History Images (MHIs) [17] to either specific face regions or the whole face to extract the facial appearance changes; these methods have achieved impressive performance on several accepted FER benchmarks. However, most handcrafted features are limited by not being directly applicable to practical problems and lack generalizability when applied to novel data.

A number of well-established problems in computer vision have recently benefited from the rise in deep learning as appearance-based feature representations or classifiers. Deep learning methods have boosted performance in a variety of tasks including face recognition, speech recognition, and object detection. However, most deep learning methods minimize cross-entropy loss and employ the softmax activation function for prediction. Thus, when updating the model, they treat all samples equally and do not consider the fact that giving difficult samples more weight could learn a more robust model.

Here we present a scheme termed Deep Neural Networks with Relativity Learning (DNNRL) for FER. DNNRL directly learns a mapping from original images to a Euclidean space, where relative distances correspond to a measure of facial expression similarity. Furthermore, DNNRL updates the model according to sample importance, leading to a more

adjustable and robust model. We conduct extensive experiments on two well-known facial expression datasets (FER-2013 and SFEW 2.0) and obtain results that are significantly better than traditional deep learning or other state-of-the-art methods.

The remainder of the paper is organized as follows: in Section 2, we briefly review related works about feature representation using appearance-based methods for FER problems. We detail the newly proposed DNNRL in Section 3. The experimental results on the two representative datasets are presented in Section 4, and we conclude in Section 5.

2 RELATED WORK

In general, feature representation using appearance-based methods can be group into two main categories: handcrafted features and deep features.

Gabor-wavelet representations have been extensively used in face image analysis and show promising performance. However, Gabor-wavelet representations are computationally costly. Bartlett et al. [2] showed that Gabor-wavelet representations derived from every 48×48 face image have the high dimensionality of $O(10^5)$. LBP describes local texture variation and is often used with a holistic representation. Shan et al. [14] observed that LBP features perform robustly and stably over a range of low-resolution images, and the method showed promising performance in low-resolution video sequences captured in practice.

Deep learning has also recently obtained state-of-the-art performance for FER problems as an appearance-based feature representation or classifier. Khorrami *et al.* [7] showed that convolutional neural networks (CNNs) are effective, and they introduced a method to decipher which part of the face image influences the CNN's predictions. Mollahosseini *et al.* [11] proposed a deep learning architecture consisting of two convolutional layers, each followed by max pooling layers and four Inception layers, to address the FER problem across multiple representative face datasets. By minimizing a margin-based loss rather than the cross-entropy loss, Tang *et al.* [16] demonstrated that switching from softmax to SVM is simple and beneficial for classification.

3. DEEP NEURAL NETWORKS WITH RELATIVITY LEARNING

Deep learning methods have, therefore, achieved excellent performance in many applications including face recognition and person re-identification. However, when updating the deep learning model, they treat all samples equally and do not consider that giving difficult samples more weight learns a more robust model. In this section, we present a novel deep feature learning method, DNNRL, which can update the model according to sample importance.

3.1 The Network Architecture

The overview of the network architecture is shown in Figure 1. The network consists of three elements. First, the network contains three convolutional layers followed by one max pooling layer. Following these layers, three inception modules are added consisting of 1×1, 3×3, and 5×5 convolution layers in parallel. Each inception module is followed by a pooling layer. The final layer is a fully connected layer with normalization [8] and dropout [6]. The Inception layer allows for improved recognition of local facial features, since smaller convolutions are applied to local features while larger convolutions approximate global features; looking at local features including the mouth and eyes allows humans to easily distinguish most emotions [1]. Normalization ensures that the relative distance derived from two images has an upper bound. Dropout is a statistical randomness that reduces the risk of network overfitting. The non-linear activation functions for all convolutional layers are set as rectified linear units (ReLUs) [8], and using the ReLU activation function avoids the vanishing gradient problem caused by some other activation functions. Finally, Batch normalizations [9] are used for all convolutional layers to be less careful about initialization and use much higher learning rates. The network configuration is shown in Table 1.

To train a more robust model, data are pre-processed as follows. The images are first resized to 96×96. Next, we randomly scale the resized images, where the range of scaling is $[-10,10]$. We then pad 0 round scaled images to the size of 120×120. Finally, we rotate the obtained images randomly in the range of $[-15^{o},15^{o}]$ and crop these images randomly to 108×108. The random rotation generates additional unseen samples and, therefore, makes the network even more robust to deviated and rotated faces. The number of images is enlarged by a factor of 10. Thus, the images are normalized to a standard size of 108×108, and this manipulation causes shape distortion that can train a more accurate model. The top and the third rows in Figure 2 give 14 examples of raw faces, while the second and the fourth rows show their corresponding pre-processed faces. Each column represents the same expression: from left to right, angry, disgust, fear, happy, sad, surprise, and neutral.

Figure 2. Examples of pre-processed faces. The first and the third rows show the original faces, while the second and the fourth rows show their corresponding pre-processed faces. Each column represents the same expression. From left to right: angry, disgust, fear, happy, sad, surprise, and neutral.

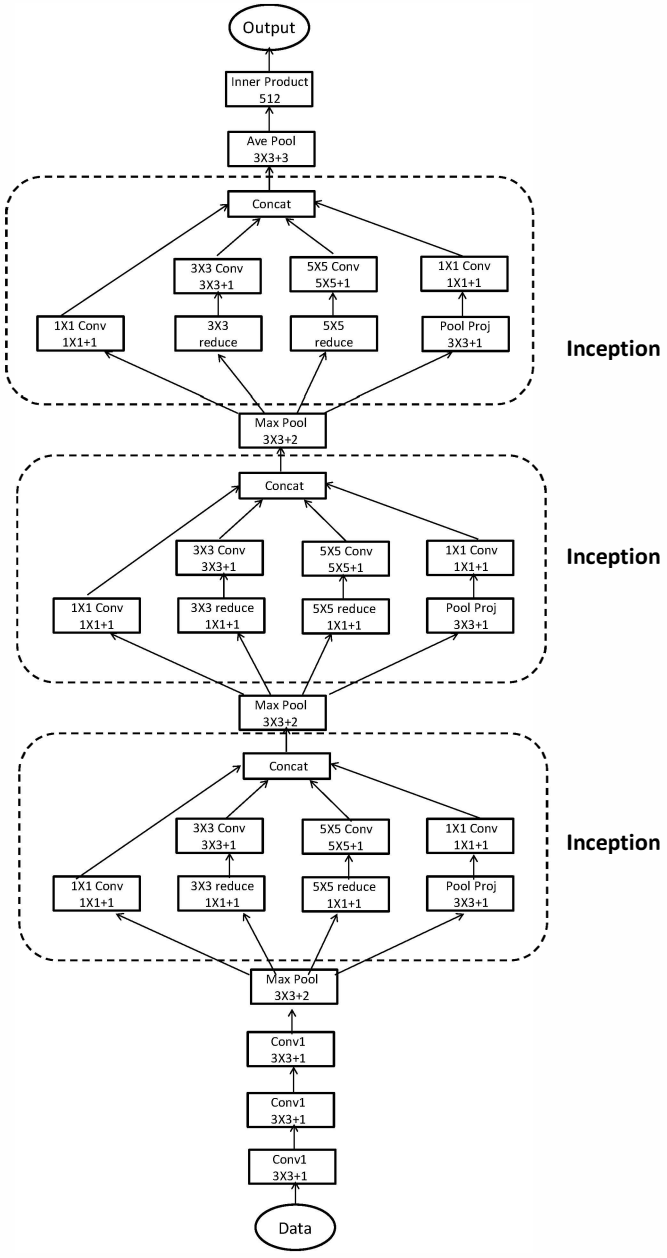

Figure 1. The network architecture.

Table 1. Network configuration

Layer type	Patch Size/Stride	Output	Params	Ops
convolution	3×3 / 2	54×54×32	288	839K
convolution	3×3 / 1	54×54×32	9K	26M
convolution	3×3 / 1	54×54×64	18K	53M
max pooling	3×3 / 2	27×27×64	-	-
inception(3a)	-	27×27×256	203K	148M
max pooling	3×3 / 2	13×13×256	-	-
inception(4a)	-	13×13×288	823K	139M
max pooling	3×3 / 2	6×6×288	-	-
inception(5a)	-	6×6×288	926K	33M
ave pooling	3×3 / 3	2×2×288	-	-
dropout(20%)	-	2×2×288	-	-
linear	-	1×1×512	589K	589K

3.2 Loss Function and Optimization

Given a network and treating it as an end-to-end system, triplet loss [13] can be directly applied to the practical problem of expression recognition; that is, triplet loss training aims to learn an optimal feature representation that maximizes the relative distance. However, when updating the model, all samples are treated equally and giving difficult samples more weight to learn a more robust model is not considered. Here we utilize the exponential function to address this problem.

Take a set of triplets $p = \{p_i\}$, $p_i = \{(p_i, p_i^+, p_i^-)\}$, where p_i and p_i^+ are images with the same expression and p_i and p_i^- are images with different expressions. Let $f(x)$ denote the network output of image x; that is, $f(x)$ is the embedded feature representation of image x. We want to learn effective features $f(p_i)$, $f(p_i^+)$, and $f(p_i^-)$ to ensure that $f(p_i)$ is closer to all other features $f(p_i^+)$ of the same expression than to any feature $f(p_i^-)$ of any other expression; we are, therefore, prone to giving difficult samples more weight when updating the model. Thus, the loss function is

$$L = \sum_{(p_i, p_i^+, p_i^-) \in p} L_i = \sum_{(p_i, p_i^+, p_i^-) \in p} e^{-\alpha \max\left\{\left\|f(p_i)-f(p_i^-)\right\|^2 - \left\|f(p_i)-f(p_i^+)\right\|^2 - \gamma, 0\right\}} \quad (1)$$

where the fixed scalar $\gamma \geq 0$ is a learning margin to prevent learning very difficult samples (the relative distance between $\left\|f(p_i) - f(p_i^+)\right\|^2$ and $\left\|f(p_i) - f(p_i^-)\right\|^2$ is smaller than γ), and α is a tradeoff parameter.

Figure 3 shows L_i. The smaller the relative distance, the larger the gradient of L_i; that is, we are biased to updating the model from difficult samples, and when the relative distance is large (the relative distance between $\left\|f(p_i) - f(p_i^+)\right\|^2$ and $\left\|f(p_i) - f(p_i^-)\right\|^2$ is large), the gradient of L_i is close to 0 and the model is only updated slightly when learning simple samples.

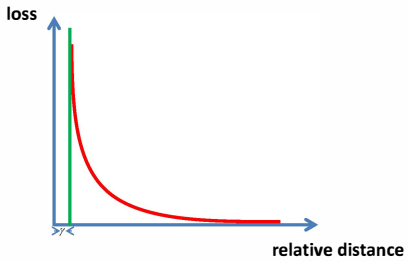

Figure 3. The exponential triplet-based loss function. The x coordinate is the relative distance and the y coordinate is the corresponding loss. The absolute value of the gradient of loss decreases as the relative distance increases.

The gradients with respect to $f\left(p_i\right)$, $f\left(p_i^+\right)$, or $f\left(p_i^-\right)$ are

$$\frac{\partial L_i}{\partial f\left(p_i\right)} = \begin{cases} -2\alpha\Delta\left[\left(f\left(p_i\right)-f\left(p_i^-\right)\right)-\left(f\left(p_i\right)-f\left(p_i^+\right)\right)\right] \\ , \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 > \gamma \\ 0, \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 \le \gamma \end{cases}$$

$$= \begin{cases} -2\alpha\Delta\left(f\left(p_i^+\right)-f\left(p_i^-\right)\right), \\ , \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 > \gamma \\ 0, \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 \le \gamma \end{cases} \quad (2)$$

$$\frac{\partial L_i}{\partial f\left(p_i^+\right)} = \begin{cases} -2\alpha\Delta\left(f\left(p_i\right)-f\left(p_i^+\right)\right), \\ , \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 > \gamma \\ 0, \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 \le \gamma \end{cases} \quad (3)$$

$$\frac{\partial L_i}{\partial f\left(p_i^-\right)} = \begin{cases} -2\alpha\Delta\left(f\left(p_i^-\right)-f\left(p_i\right)\right), \\ , \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 > \gamma \\ 0, \left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2 \le \gamma \end{cases} \quad (4)$$

where Δ denotes $e^{-\alpha\left\|f\left(p_i\right)-f\left(p_i^-\right)\right\|^2-\left\|f\left(p_i\right)-f\left(p_i^+\right)\right\|^2-\gamma}$. Hence, the loss function in (1) can be easily integrated in back propagation in neural networks.

4. EXPERIMENTAL RESULTS

We conducted facial expression recognition experiments on two widely used datasets, the FER-2013 dataset [5] and the SFEW 2.0 dataset [3], to demonstrate the effectiveness of the proposed method. The FER-2013 dataset contains 27,809 training images, 3,589 validation images, and 3,589 test images. Faces are labeled with any of the six basic expressions or neutral: the number of images representing the six basic expressions (anger, disgust, fear, happiness, sadness, and surprise) and neutral are 4953, 547, 5121, 8989, 6077, 4002, and 6189, respectively. This dataset was created using the Google image search API and contains large variations reflecting real-world conditions. The images are 48×48 pixels.

The SFEW 2.0 dataset was created from Acted Facial Expressions in the Wild (AFEW) [3] using the key-frame extraction method. It contains 880 training samples, 383 validation samples, and 372 test samples. Since SFEW 2.0 is a competition dataset in the Wild Challenge (EmotiW) 2015, test sample labeling is private and, therefore, unavailable. Furthermore, SFEW 2.0 is relatively small, rendering it prone to overfitting when trained on the DNNRL. Hence, we pre-trained the model on the FER-2013 training set and fine tuned on the SFEW 2.0 and FER-2013 training sets; the test samples are FER-2013 test images. The SFEW 2.0 dataset includes different head poses, occlusions, backgrounds, and close to real-world illuminations; thus, face alignment is necessary to extract face-related information and for unifying all face images to the same domain. We used SDM [18] for face alignment, transformed the aligned faces to grayscale, and resized to 48×48, which is the same as FER-2013 data. Figure 4 shows examples of face alignment by SDM, where the top row shows original images and the bottom row shows their corresponding aligned faces.

Figure 4. Examples of face alignment by SDM, where the top row shows original images and the bottom row shows their corresponding aligned faces.

We first normalized the images in the above two datasets to 96×96 and then pre-processed these images using the manipulation described in Section 3.1. After obtaining deep features from the DNNRL model, to make a class choice, we used a k-NN classifier to obtain the recognition accuracy, where any test sample was classified by a majority vote of its k training samples, with the test sample being assigned to the class most common among its k nearest training samples. Use of the k-NN classifier is reasonable, because the DNNRL loss is also based on Euclidean distances. We determined the value of k for sample testing when the validation samples obtained the highest accuracy on the value of k.

The initial learning rate was set to 0.01, while the minimum learning rate was set to 0.0001. Each training epoch had $[N/128]$ batches, with the training samples randomly

selected from the training set. The learning margin γ was set to 0, and the tradeoff parameter α was set to 0.2. The trained network parameters and loss at each epoch were recorded. The validation loss was assessed after each round of training; if the validation loss increased by more than 10%, the learning rate was reduced by one-tenth and the previous network with the best validation loss was reloaded. The termination criterion for the training model was judged by the new lowest value for the validation loss in 20 iterations.

4.1 Performance on FER-2013

The performance of DNNRL and baselines on FER-2013 are shown in Table 2. "DNN" refers to the accuracy of the proposed network described in Section 3.1 with softmax, and "T-DNN" refers to the accuracy of the proposed DNNRL with the k-NN classifier, where the loss function of DNNRL is replaced by the triplet-based loss. "DNNRL" refers to the accuracy of the proposed DNNRL with the k-NN classifier.

AlexNet [8] and FER-2013 Champion [16] results are also listed. From Table 2, it can be seen that DNNRL obtains the highest recognition rate of the tested methods, demonstrating its effectiveness and robustness for FER. DNN outperforms AlexNet because the Inception layer improves local feature recognition. T-DNN outperforms DNN because it can directly learn a mapping from original images to a Euclidean space, where relative distances correspond to a measure of facial expression similarity. DNNRL outperforms T-DNN because it updates the model more or less according to the sample difficulty.

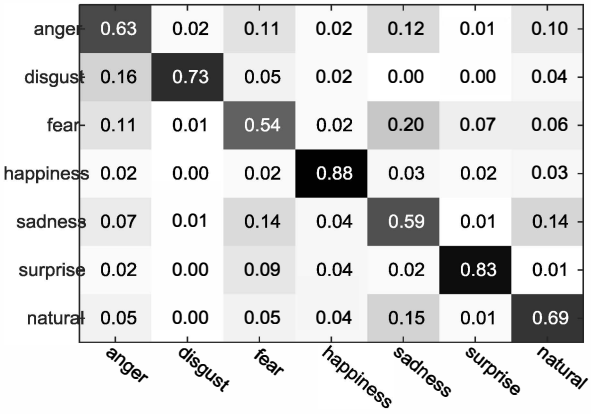

Figure 5. DNNRL classification confusion matrices on FER-2013 test set (trained on FER-2013 training set).

The confusion matrix of DNNRL on the FER-2013 dataset (trained on FER-2013 training set) is shown in Figure 5. The confusion matrix between the ground-truth class label and the most likely inferred class label information provide a better understanding of DNNRL's limitations. As expected, confusion frequently occurs between "anger", "fear", and "sadness" because they create similar motions. The confusion matrices also show that "natural" is easily misclassified as "sadness".

Table 2. Classification accuracy of the proposed DNNRL with other methods on the FER-2013 test set (trained on FER-2013 training set).

Accu	DNNRL	T-DNN	DNN	AlexNet	FERwin [16]
Test	0.7060	0.7013	0.6922	0.6482	0.693

4.2 Performance on SFEW 2.0 and FER-2013 datasets

The performance of DNNRL and baselines on FER-2013 (trained on SFEW 2.0 and FER-2013 training sets) are shown in Table 3. The baselines include AlexNet [8], DNN, and T-DNN. DNNRL obtains the highest recognition rate of the tested methods and outperforms the highest recognition rate of Table 2, demonstrating its robustness for FER.

Figure 6 shows the confusion matrices of DNNRL on the FER-2013 (trained on SFEW 2.0 and FER-2013 training sets). Confusion also frequently occurs between "anger", "fear", and "sadness", and "natural" is easily misclassified as "sadness".

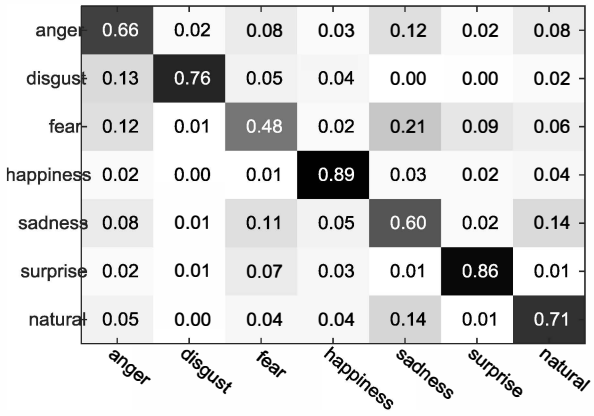

Figure 6. DNNRL classification confusion matrices on FER-2013 validation and test sets.

Table 3. Classification accuracy of the proposed DNNRL with other methods on FER-2013 test set (trained on SFEW 2.0 and FER-2013 training sets).

Accu	DNNRL	T-DNN	DNN	AlexNet
Test	0.7133	0.7097	0.6994	0.6509

5. CONCLUSION

This work introduces DNNRL, a new deep learning method for FER. DNNRL consists of three convolutional layers, four pooling layers, three Inception layers, and one fully connected layer. The Inception layers increase the width and depth of the network while not increasing computational cost, furthermore, they improves local feature recognition. By using the exponential triplet loss, DNNRL can directly learn a mapping from original images to a Euclidean space, where relative distances correspond to a measure of facial expression

similarity. Furthermore, the model is updated to a greater or lesser degree according to the sample difficulty, which leads to a more adjustable and robust model.

Compared to traditional deep neural networks such as AlexNet, DNNRL is competitive and achieves excellent performance for FER.

REFERENCES

[1] E. Bal, E. Harden, D. Lamb, A. Van Hecke, J. Denver, and S. Porges, "Emotion recognition in children with autism spectrum disorders: Relations to eye gaze and autonomic state," *Journal of Autism and Developmental Disorders*, vol. 40, no. 3, pp. 358-370, 2010.

[2] 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 *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 568-573.

[3] A. Dhall, O. V. R Murthy, R. Goecke, J. Joshi, and T. Gedeon, "Video and image based emotion recognition challenges in the wild: Emotiw 2015," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, 2015 ,pp. 423-426.

[4] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *Journal of personality and social psychology*, vol. 17, no. 2, pp. 124-129, 1971.

[5] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D. H. Lee, Y. Zhou, C. Ramaiah, F. Feng, R. Li, X. Wang, D. Athanasakis, J. Shawe-Taylor, M. Milakov, J. Park, R. Lonescu, M. Popescu, C. Grozea, J. Bergstra, J. Xie, L. Rpmaszko, B. Xu, Z. Chuang, and Y. Bengio, "Challenges in representation learning: A report on three machine learning contests," *Neural Networks*, vol. 64, pp. 59-63, 2015.

[6] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing coadaptation of feature detectors," *arXiv preprint*, arXiv:1207.0580, 2012.

[7] 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 (ICCV),* 2015, pp. 19-27.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Conference on Neural Information Processing Systems(NIPS)*, 2012, pp.1097–1105.

[9] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift, " *arXiv preprint* arXiv:1502.03167, 2015.

[10] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 21, no. 12, pp. 1357-1362, 1999.

[11] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going Deeper in Facial Expression Recognition using Deep Neural Networks," *arXiv preprint*, arXiv: 1511.04110, 2015.

[12] T. Ojala, M. Pietikanien, and T. Maenpa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.

[13] F. Schroff, K. Dmitry, and P. James, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2015, pp: 815-823.

[14] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Image and Vision Computing*, vol. 27, no. 6, pp. 803-816, 2009.

[15] M. Soleymani, S. Asghari-Esfeden, M. Pantic, and Y. Fu, "Continuous emotion detection using eeg signals and facial expressions" in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2014, pp. 1-6.

[16] Y. Tang, "Deep learning using linear support vector machines," *arXiv preprint*, arXiv:1306.0239, 2013.

[17] M. Valstar, M. Pantic, and I. Patras, "Motion history for facial action detection in video," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2004, pp. 635-640.

[18] X. Xiong and F. Torre, "Supervised descent method and its applications to face alignment," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 532-539.