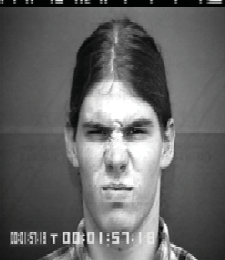
Facial Expression

Recognition Using Deep Neural Networks

Junnan Li and Edmund Y. Lam

Department of Electrical and Electronic Engineering The University of Hong Kong, Pokfulam,

Hong Kong

***Abstract*—We develop a technique using deep neural network for human facial expression recognition. Images of human faces are**

**preprocessed with photometric normalization and histogram**

**manipulation to remove illumination variance. Facial features are then extracted by convolving each preprocessed image with 40 Gabor filters. Kernel PCA is applied to features before feeding them into the deep neural network that consists of 1 input layer, 2 hidden layers and a softmax classifier. The deep network is trained**

Angry Disgust

Fear

**using greedy layer-wise strategy. We use the Extended Cohn-**

**Kanade Dataset for training and testing. Recognition tests are performed on six basic expressions (i.e. surprise, fear, disgust, anger, happiness, sadness). To test the robustness of the classification system further, and for benchmark comparison, we add a seventh emotion, namely “contempt”, for additional**

Happiness Sadness

Surprise

**recognition tests. We construct confusion matrix to evaluate the performance of the deep network. It is demonstrated that the network generalizes to new images fairly successfully with an average recognition rate of 96.8% for six emotions and 91.7% for seven emotions. In comparison with shallower neural networks and SVM methods, the proposed deep network method can provide better recognition performance.**

***Keywords—Multi-layer neural network; emotion recognition; Gabor filters; Kernel principal component analysis.***

Fig.1. Sample face images of six basic emotions from the CK+ database [2]

recognition systems also find uses in other fields such as education software, animations, automobile safety and behavioral science.

Facial expression recognition falls within the research framework of pattern recognition. A recognition system would consist of three stages: face detection, feature extraction and expression classification. A large amount of research has been

carried out in these three central issues of facial erepression

1. INTRODUCTION

In computer vision, the automatic recognition of facial expressions has been an active research area for a long time. Facial expression recognition refers to detecting human emotions based on expressions. Since the early 1970s, studies performed by Ekman have shown that there are six “universal categories of emotional expressions” that are easiest to be recognized by humans. Those prototypical facial expressions are: surprise, fear, disgust, anger, happiness and sadness [1]. Since facial expression recognition is characterized as a pattern recognition problem in human cognition, computers with pattern recognition abilities have the potential to perform as well as, or even better than human.

recognition. A variety of facial expression recognition systems have been developed using different feature extraction and classification methods. Tian et al. studied the combination of permanent features and transient features with artificial neural networks (ANN) as the classifier [3]. Wang and Yin applied topographic context (TC) expression descriptors and a combined classifier of quadratic discriminant classifier (QDC), linear discriminant analysis (LDA) and naive Bayesian network classifier (NBC) [4]. More recent work by Sánchez et al. applied optical flow-based methods for feature extraction and Support Vector Machine (SVM) as classification method [5].

Among those published work, neural network-based method has shown promising results. It has obtained over 85% accuracy for six basic emotion classificati ns. Our study continues to

The automatic expression recognition has significant meaning

explore the potential of neural network to recognize facial

expressions. In order to represent better mapping from the feature

to many applications. With the advances in robotics, the requirement of a robust real-time facial expression recognition system is urgent. It could improve the performance of human- computer interaction and help to construct more intelligent robots with the ability to understand human emotions. Apart from robotics and human-computer interaction, facial expression

space to the facial expression space, we investigate feedforward deep neural networks, which have the ability to model more complex nonlinear functions.

We have developed a real-time facial expression recognition system with high recognition rates. We design a multistep image

978-1-4799-8633-0/15/$31.00 ©2015 IEEE

preprocessing and feature extraction procedure that improves the classification performance. In particular, histogram remapping to

Viola-Jones algorithm constructs an intermediate image representation called the internal image, which can be computed

a normal distribution is used after photometric normalization to

rapidly. It builds a simple and

efficient classifier using the

remove illumination variance. Then we use Gabor wavelets and Kernel Principal Component Analysis to extract representative features of the preprocessed image. The nonlinear features extracted by Kernel PCA provide better recognition rates compared with linear PCA. The features are fed into the deep neural network for classification. The deep network applies greedy layer-wise training strategy that can learn from both labeled and unlabeled data. The use of unlabeled data helps the system to learn better features when the amount of labeled data is limited.

In the remainder of the paper, we will first discuss the preprocessing techniques in Section 2. In Section 3, the feature extraction methods are presented. Training procedure of the deep network is proposed in Section 4. Experimental results for recognition tests on the Extended Cohn-Kanade Dataset are demonstrated and analyzed in Section 5. Conclusions are stated in Section 6.

1. PREPROCESSING OF IMAGES

Image preprocessing represents an essential part of a facial expression recognition system. It has significant impact on the

AdaBoost learning algorithm. The face detection is achieved by combining classifiers in a cascade structure that is capable of increasing the detection performance while reducing computational complexity.

*B. Photometric normalization*

The variation of illumination conditions of a facial image can introduce large changes to the image, hence impairing the performance of the facial expression classification. Many photometric normalization algorithms have been proposed that can remove illumination variance at the preprocessing level, including multiscale retinex method [9], isotropic diffusion based normalization method, and anisotropic diffusion based normalization method [10].

Among those photometric no malization algorithms, it is shown by Short et al. that homomorphic filtering based normalization yields the most consistent results compared with other techniques [11]. Hence, we apply homomorphic filtering based normalization in our recognition system.

An image is the product of two components, illumination and reflectance. A homomorphic filter decomposes them by taking the logarithm. Then it applies Fourier transform to transform the

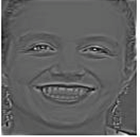
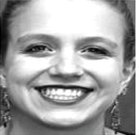
robustness and performance of the system. The object of preprocessing is to reduce the influence of noise on feature extraction and enhance the discriminative information contained in images.

*A. Face detection*

In our facial expression recognition system, we need images of frontal faces that are normalized in scale. Hence, it is important that we can localize and extract the face region from an image. The exclusion of background is crucial for reliable expression classification.

two components into 𝐹𝐿 and 𝐹𝑅 . 𝐹𝐿 mainly comprises of high

frequency components, while 𝐹𝑅 mainly comprises of low frequency components. Convolving them with a homomorphic filter will emphasize high frequency components and reduce low frequency components. Therefore, the contrast of the image is improved and the dynamic range is compressed. The image is transformed back into spatial domain by applying inverse Fourier transform.

Different face detection algorithms have been proposed,

including shape-information-based approach [6] and skin-color- based approach [7]. Shape-information-based approach is often not fast enough for real-time detection, while skin-color-based approach only works for colored images. In our system, we use the Viola-Jones face detection framework, which is a robust algorithm capable of processing images extremely rapidly for

Fig.3. Sample images of homomorphic filtering based normalization

*C. Histogram remapping to normal distribution*

Histogram remapping is a common pre- or post-processing step for photometric normalization. The most common histogram

real-time situations [8]. It is most effective on images of frontal

faces, which is exactly the type of images we use. Tests show that Viola-Jones algorithm achieves a detection rate of 100% for

remapping approach is histogram equalization, where pixel values are mapped to a uniform distribution so as to improve contrast and compensate for the illumination variance. Study by

the images used in this study.

Ranawade has shown that using histogram equalization in

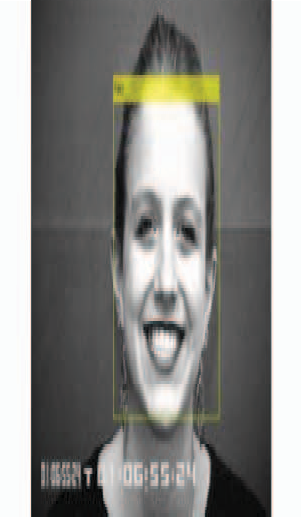


Fig.2. Viola-Jones algorithm applied to sample image of human face

conjunction with photometric normalization leads to a better classification performance than using photometric normalization on its own [12]. However, the usefulness of histogram equalization is determined empirically rather than theoretically. Histogram equalization is only a special case for a more general concept, which is altering pixel intensity values in a way that the distribution fits a predefined function. Rather than fitting a

uniform distribution to the histogram of the images, as it is done

ƒ𝑢 = ƒ𝑚𝑎𝑥/2𝑢/2 𝜃 = 𝜋𝑣/8 

in histogram equalization, we propose to use a normal distribution. The experimental results in Section 5 will show that normal distribution mapping leads to better recognition rates than histogram equalization.

The expression for the normal distribution curve is given by

ƒ𝑚𝑎𝑥 denotes the maximal frequency. 𝛾 and 𝛼 determine the sharpness along a and b axis. u and v define the number of orientations and scales of the filter bank.

In this system, we construct a Gabor filter bank consisting of 40 filters. We choose eight orientations to capture subtle features

f(𝑝) = 1

𝜎√2𝜋

exp (−(𝑝−𝜇)2)

2𝜎

2



of the facial expression, and five scales to efficiently represent features of a 128×128 image. The other parameters selected are 𝛾

where 𝜇 represents the mean value, and 𝜎 denotes the standard deviation.

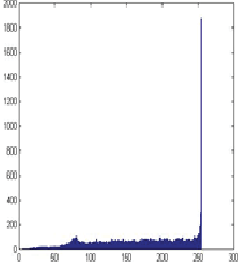
In our system, we set 𝜇 to be 0 and 𝜎 to be 1. However, due to the nature of neural networks, 𝜇 has no influence on the

= 𝛼 = √2 and ƒ𝑚𝑎𝑥 = 0.25, which are also appropriate for the image size.

We extract the Gabor features of a grey-scale image by convolving the image 𝐼(𝑎, 𝑏) with the Gabor filter bank

classification results. Figure 4 gives a visual example of the

histogram remapping. Here the mapped pixel values are rescaled back to the 8-bit interval for visualization purposes.

G𝑢,𝑣(𝑎, 𝑏), i.e.,

𝐹(𝑎, 𝑏) = 𝐼(𝑎, 𝑏) \* G𝑢,𝑣(𝑎, 𝑏). 

The magnitude responses of a sample image filtered by two of the Gabor filters are shown in Fig.5.

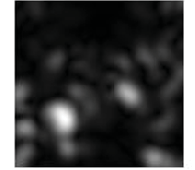
  

Fig. 5. Magnitude responses of a preprocessed sample image

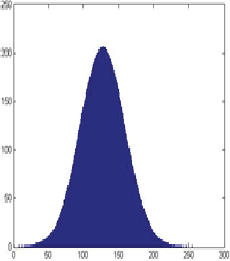
Since we have resized the images to 128×128 before

Fig.4. A sample image and its histogram before and after mapping the histogram to the normal distribution

1. FEATURE EXTRACTION

Feature extraction is an essential component of the recognition system. It aims to identify the most appropriate and meaningful representation of the face images for recognition. In our system, Gabor filtering and Kernel Principal Component Analysis are used to extract features.

1. *Gabor Wavelets*

The Gabor filter is a useful tool to extract meaningful facial features. It is similar to the receptive field profile of cortical simple cells, which is characterized as localized, frequency selective and orientation selective [13]. Study by Zhang et al.

convolving with Gabor filters, the Gabor features reside within a space of dimension 655360 (128×128×40). The features are of too high a dimension to efficiently process and store. In order to reduce the dimension, we apply downsampling by a factor of 64 to all feature vectors. However, the features still have a very large dimensionality, which is computationally expensive and contains redundant information. Therefore, Kernel Principal Component Analysis is applied.

1. *Kernel Principal Component Analysis*

PCA (Principal Component Analysis) and KPCA (Kernel Principal Component Analysis) are important techniques used to reduce dimensionality of features. Dimensionality reduction helps create uncorrelated features and reduces computation cost [15]. Features of lower dimension can also provide a better representation of the face images.

While conventional PCA aims to extract a subspace where

suggests that Gabor wavelet extracted from face images can achieve a much better performance than geometric positions of a set of fiducial points [14].

the variance of the features is maximized, some undesired variations might be retained. The linear projection of PCA may be suboptimal to represent information based on higher order

A Gabor filter bank in 2D spatial domain (𝑎, 𝑏) is defined by

dependencies in an image, such a

nonlinear relations of pixel

where

G𝑢,𝑣

f 2 −((𝑎

𝑎, 𝑏 = e

( ) u

𝜋𝛾𝛼

𝘍2

/𝛾2)+(𝑏

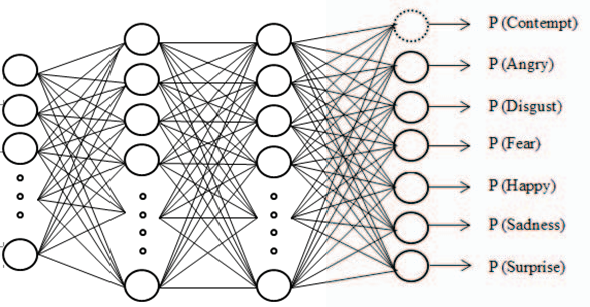
𝘍2

/𝛼2)) ej2𝜋fu𝑎𝘍2



values [16]. Schölkopf et al. have extended the linear PCA to Kernel PCA, where inputs are mapped from their original space to a space of higher dimension. KPCA can extract nonlinear features and thus provides better recognition performance [17].

𝑎′ = 𝑎𝑐o𝑠𝜃 + 𝑏𝑠i𝑛𝜃 𝑏′ = −𝑎𝑠i𝑛𝜃 + 𝑏𝑐o𝑠𝜃 

Given a set of samples {𝑥1 , 𝑥2 …… 𝑥k } ∈ 𝑅𝑛 with zero mean and unit variance, linear PCA finds the projection directions by finding the eigenvalues 𝜆 and eigenvectors u of the covariance matrix C, such that 𝜆u = Cu.

In KPCA, each vector 𝑥i in 𝑅𝑛 is projected to a feature space

𝑅f of higher dimension, using a nonlinear ma**p**ping function ɸ.

Hence, the eigenvalue problem in feature space becomes

𝜆𝑢ɸ = 𝐶ɸ𝑢ɸ

where 𝐶ɸ is the new covariance matrix,



and 𝑢ɸ is the

eigenvector with 𝑢ɸ = ∑𝑚 𝛼i ɸ(𝑥i).

i=1

Fig.6. Architecture of the Deep Network

Then, we can project ɸ(𝑥i) in feature space 𝑅f to a low dimensional space spanned by eigenvectors 𝑢ɸ . For a input vector 𝑥j , those projections are the nonlinear principal components corresponding to ɸ .

The activation function of hidden units is the logistic sigmoid function:

g(𝑧) = 1 .



1+e𝑥𝑝(−𝑧)

𝑢ɸ ∙ ɸ(𝑥j) = ∑𝑚

i=1

Denote the kernel function by

𝛼i (ɸ(𝑥i) ∙ ɸ(𝑥j)). 

In the function, z = Wx, where x is the input vector and W is the weight parameter.

The output layer is a softmax classifier. Each output unit will

𝑘(𝑥i, 𝑥j) = ɸ(𝑥i) ∙ ɸ(𝑥j) .



output the probability of the input image being its corresponding expression. The output h(x) of the softmax classifier for an input

Hence, the nonlinear principal components can be extracted

implicitly using the kernel function without the explicit projection of input vectors to high dimensiona**l** space 𝑅f. This

1

vector 𝑥i is given by

𝑝( 𝑦i = 1| 𝑥i; W)

𝖥e𝖶𝑇𝑥i 1

2

makes the Kernel PCA have a similar computational complexity

ℎ (𝑥 ) = [ 𝑝( 𝑦i = 2| 𝑥i; W) ] = 1 Ie𝖶𝑇𝑥i I



compared with linear PCA.

𝖶 i

𝑝( 𝑦

j

⁝

= 𝑚| 𝑥 ; W)

𝑚 j=1

∑ e

W𝑇𝑥i I

⁝ I

𝖶𝑇𝑥

In this system, we use the fractional power polynomial kernel,

i i Le 𝑚 iے

which is defined by

k(𝑥 , 𝑥 ) = sgn(𝑥 ′𝑥 ) ∙ |𝑥 ′𝑥 |0.8. 

where m equals to the number of emotions, and W is the weight parameter.

i j i j i j

In Section 5, we will show that the nonlinear principal components extracted by KPCA achieve better recognition rates than linear principal components extracted using PCA.

1. FEEDFORWARD DEEP NEURAL NETWORKS

Deep neural networks are ones in which there are multiple hidden layers. Since each hidden layer computes a nonlinear

*B. Greedy layer-wise training*

Even though the significant power of deep networks has been proved theoretically and appreciated for decades, researchers found it difficult to train deep networks. Traditional gradient- based optimization algorithms are not effective when the gradient is propagated across multiple la ers of non-linear functions [19][20]. Reasons include insufficiency of labeled data, converging to local optima and diffusion of gradients.

In order to address those problems, Hinton et al. has proposed

transform of the previous layer, multiple hidden layers have the power to generate much more complex features of the input. As a result, a deep network can learn significantly more complex functions than a shallow network. It has been shown that a k- layer network can represent functions that a (k – 1) layer network can only represent with an exponentially large number of hidden units [18].

*A. The network architecture*

The deep network we design contains one input layer, two hidden layers and one output layer as shown in Fig.6. The inputs are feature vectors obtained after KPCA. Each hidden layer contains 200 units.

a greedy layer-wise unsupervised training strategy based on restricted Boltzmann machines (RBM) [21]. Bengio et al. further improved the greedy layer-wise procedure with autoassociator networks. The main idea of the method is to train different layers of the deep network one at a time [18]. It is the training strategy that we apply.

In our training process, the two hidden layers of the network are firstly trained using unlabeled images. They will try to learn an identity function where the desired output is the same as the input. This process is unsupervised feature learning. The use of unlabeled images helps the network learn good feature representations prior to supervised learning. Then we feed the labeled image data into the two hidden layers that are pre-trained

and perform forward propagation to obtain feature vectors. Those feature vectors are used to train the output layer, which is a softmax classifier. Supervised training is applied here, where the target value of a output unit is 1 if the labeled emotion is the same as the one it represents, and 0 otherwise. We apply fine- tuning of the whole network as the final step. We treat all layers as one single model and use back propagation algorithm to improve upon all the weights in one iteration.

1. EXPERIMENTAL RESULTS

We use the Extended Cohn-Kanade Dataset for training and testing the deep neural network. It is a very popular database used to evaluate the performance of facial expression recognition systems. The number of images contained in the dataset is shown in TABLE I.

TABLE I. Number of images of the seven emotions in CK+ Dataset

|  |  |
| --- | --- |
| **Emotion** | **Number of Images** |
| Angry | 45 |
| Disgust | 59 |
| Fear | 25 |
| Happiness | 69 |
| Sadness | 28 |
| Surprise | 82 |
| Contempt | 19 |
| Total | 327 |

1. *Recognition of six basic facial expressions*

Recognition tests are firstly performed on six basic expressions where effects of different preprocessing techniques on classification performance are studied. We conduct six sets of test. Each set uses the same 327 images, but the images are preprocessed with different techniques. We use leave-one-out subject cross-validation strategy, which is an exhaustive cross- validation method. We use one image as the validation set and the remaining as the training set. For each set of test, the cross- validation is repeated 327 times. And the correct recognition rate is defined by dividing the number of correctly recognized image

with the total number of images.

For each test set, we have recorded the recognition rate for each emotion as well as the total recognition rate. TABLE II shows the recognition rate using original/uniform/normal histogram distribution and PCA/KPCA.

Based on the results, we can see that histogram remapped images significantly outperform images with no histogram manipulation. Of the two histogram manipulation techniques, fitting a normal distribution leads to better recognition rates than histogram equalization. Moreover, Kernel PCA is more powerful over PCA in terms of improving recognition rates. Overall, fitting a normal histogram distribution combined with KPCA yields the best performance.

1. *Recognition of seven facial expressions*

In order to further test the robustness of the system and do reliable benchmark comparison, we perform recognition tests on seven emotions, including six basic emotions and contempt. A confusion matrix of the test results is presented in TABLE III.

TABLE III. Confusion matrix of seven emotion recognition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **An** | **Di** | **Fe** | **Ha** | **Sa** | **Su** | **Co** |
| **An** | **84.4** | 3.0 | 2.2 | 0 | 10.4 | 0 | 0 |
| **Di** | 5.2 | **94.7** | 0 | 0 | 0 | 0 | 0 |
| **Fe** | 6.9 | 4.2 | **81.9** | 0 | 4.2 | 2.8 | 0 |
| **Ha** | 0 | 0 | 0 | **100** | 0 | 0 | 0 |
| **Sa** | 19.8 | 0 | 4.9 | 0 | **66.7** | 7.4 | 0 |
| **Su** | 0 | 0 | 0 | 0 | 0 | **100** | 0 |
| **Co** | 0 | 0 | 1.9 | 0 | 18.5 | 0 | **79.6** |

The overall recognition rate for seven emotions is 91.7%, which is lower than the recognition rate for six emotions (96.8%). An explanation for the drop in recognition rate is that adding one more possibility in the output of the network will dilute the possibility of the correct emotion, since the total possibility of seven emotions has to be equal to 1. Another reason may be due to the fact that contempt is a very subtle emotion and can be easily confused with other stronger emotions.

TABLE II. Recognition rates of six emotions with different histogram distribution and PCA/KPCA

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Emotion** | **PCA** | **KPCA** | **Histogram equalization**  **+ PCA** | **Histogram equalization**  **+ KPCA** | **Normal distribution**  **+ PCA** | **Normal distribution**  **+ KPCA** |
| Angry | 80% | 84.4% | 91.1% | 91.1% | 91.1% | 95.6% |
| Disgust | 94.9% | 94.9% | 98.3% | 98.3% | 98.3% | 100% |
| Fear | 64% | 68% | 72% | 72% | 76% | 84% |
| Happiness | 100% | 100% | 100% | 100% | 100% | 100% |
| Sadness | 64.3% | 67.9% | 78.6% | 78.6% | 85.7% | 85.7% |
| Surprise | 100% | 100% | 100% | 100% | 100% | 100% |
| Average | 89.9% | 91.2% | 94.2% | 94.2% | 95.1% | 96.8% |

The results given in TABLE III are in line with the study by Lucey et al., where Support Vector Machine is used for recognizing seven emotions on the Extended Cohn-Kanade Dataset [2]. In their study, Lucey et al. applied two SVMs with one using SPTS (similarity-normalized shape) features and the other using CAPP (canonical appearance) features. The same leave-one-out subject cross-validation is applied. It is worth mentioning that Lucey et al. were also involved in the creation of the Extended Cohn-Kanade Dataset.

TABLE IV shows a comparison of the performance between our deep neural network and the SVM method. Generally speaking, the deep network performs better with a +3.7% higher average recognition rate. Explicitly speaking, the deep network performs better for Angry, Fear and Surprise. For Disgust and Sadness, both systems have the same performance while the SVM performs better for Sadness and Contempt.

TABLE IV. Recognition rates of 7 emotions for the Deep Network and SVM

|  |  |  |  |
| --- | --- | --- | --- |
| **Emotion** | **Deep Network** | **SVM** | **Difference** |
| Angry | 84.4% | 75% | + 9.4% |
| Disgust | 94.7% | 94.7% | 0 |
| Fear | 81.9% | 65.2% | + 16.7% |
| Happiness | 100% | 100% | 0 |
| Sadness | 66.7% | 68% | - 1.3% |
| Surprise | 100% | 96% | + 4% |
| Contempt | 79.6% | 84.4% | - 4.8% |
| Average | 91.7% | 88.3% | + 3.4% |

1. CONCLUSION

In this paper, a facial expression recognition system based on feedforward deep neural networks is built. The system consists of three major stages, which are image preprocessing, feature extraction and expression classification. Recognition tests were performed on the Extended Cohn-Kanade Dataset. We have shown that fitting normal distribution to the histogram of images combined with Kernel PCA yields an improved recognition rate compared with conventional histogram equalization and linear PCA. In the experimental results presented, the deep network provides better performance on seven emotion recognition compared with the SVM method proposed in [2]. However, since the recognition tests were performed only on one dataset, future work is to improve the system so that it can adapt to a variety of datasets. Moreover, the application of the system to real-life engineering problems will be studied.

REFERENCES

1. P. Ekman and W. Friesen, “Constants across cultures in the face and emotion. Journal of personality and social psychology,” *Journal of personality and social psychology*, vol. 17, no. 2, pp. 124-129, 1971.
2. 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 *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops(CVPRW)*, June 2010, pp. 94-101.

1. Y. Tian, T. Kanade, and J. F. Cohn, “Recognizing action units for facial expression analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 97-115, 2001.
2. J. Wang and L. Yin, “Static topographic modeling for facial expression recognition and analysis,” *Computer Vision and Image Understanding*, vol. 108, no. 1, pp. 19-34, 2007.
3. A. Sánchez, J. V. Ruiz, A. B. Moreno, A. S. Montemayor, J. Hernández and J. J. Pantrigo, “Differential optical flow applied to automatic facial expression recognition,” *Neurocomputing*, vol. 74, no. 8, pp. 1272-1282, 2011.
4. J. Wang and T. Tan, “A new face detection method based on shape information,” *Pattern Recognition Letters*, vol.21, no. 6, pp. 463-471, 2000.
5. J. Kovac, P. Peer and F. Solina, “Human skin clor clustering for face detection”, in *EUROCON 2003, Computer as a Tool. The IEEE Region8*, September 2003, pp. 144-148.
6. P. Viola and J. M. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137-154, 2004.
7. D. J. Jobson, Z. Rahman and G. A. Woodell, “A multiscale retinex for bridging the gap between color images and the human observations of scenes.” *IEEE Transations on Imagae Processing*, vol. 6, no. 7, pp. 965- 976, 1997.
8. G. Ralph and B. Vladimir, “An image preprocessing algorithm for illumination invariant face recognition,” in *Proccedings of the 4th International Conference on Audio- and Video-Based Biometric Personal Authentication*, June 2003, pp. 10-18.
9. J. Short, J. Kittler and K. Messer, “A comparison of photometric normalisation algorithms for face verification,” in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, May 2004, pp. 254-159.
10. S. S. Ranawade, “Face recognition and verification using artificial neural network,” *International Journal of Computer Applications*, vol. 1, no. 14, pp. 21-25, 2010.
11. S. R. Zhou, P. J. Yin and M. J. Zhang, “Local binary pattern (LBP) and local phase quantization (LBQ) based on Gabor filter for face representation,” *Neurocomputing*, vol. 116, pp. 260-264, 2013.
12. Z. Zhang, M. Lyons, M. Schuster and S. Akamatsu, “Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron,” in *Proceedings Third IEEE International Conference on Automatic face and gesture recognition,* April 1998, pp. 454-459.
13. L. Cao, K. Chua, W. Chong, H. Lee and Q. Gu, “A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine,” *Neurocomputing*, vol. 55, no. 1-2, pp. 321-336, 2003.
14. H. M. Yang, “Face Recognition Using Kernel Methods,” in *Advances in Neural Information Processing Systems (NIPS)*, vol. 14, 2001.
15. B. Schölkopf, A. Smola and R. K. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Computation*, vol. 10, no. 5, pp. 1299-1319, 1998.
16. Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy layer- wise training of deep networks,” *Advances in neural information processing systems*, vol. 19, pp. 153-160, 2007.
17. H. Larochelle, Y. Bengio, J. Louradour and P. Lamblin, “Exploring strategies for training deep neural networks,” *The Journal of Machine Learning Research*, vol. 10, pp. 1-40, 2009.
18. H. Larochelle, D. Erhan, A. Courville, J. Bergstra and Y. Bengio, “An empirical evaluation of deep architectures on problems with many factors of variation,” in *Proceedings of the 24th international conference on Machine learning*, June 2007, pp. 473-480.

G. Hinton, S. Osindero and Y. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp. 1527-1554, 2006.