Facial Expression Recognition and Generation using Sparse Autoencoder

Yunfan Liu, Xueshi Hou, Jiansheng Chen, Chang Yang, Guangda Su and Weibei Dou
Department of Electronic Engineering, Tsinghua University, Beijing, China
yf-liu11@mails.tsinghua.edu.cn
houxs11@mails.tsinghua.edu.cn
jschenthu@mail.tsinghua.edu.cn
yangchang0720@gmail.com
susu@mail.tsinghua.edu.cn
douwb@mail.tsinghua.edu.cn

***Abstract*—Facial expression recognition has important practical applications. In this paper, we propose a method based on the combination of optical flow and a deep neural network—stacked sparse autoencoder (SAE). This method classifies facial expressions into six categories (i.e. happiness, sadness, anger, fear, disgust and surprise). In order to extract the representation of facial expressions, we choose the optical flow method because it could analyze video image sequences effectively and reduce the influence of personal appearance difference on facial expression recognition. Then, we train the stacked SAE with the optical flow field as the input to extract high-level features. To achieve classification, we apply a softmax classifier on the top layer of the stacked SAE. This method is applied to the Extended Cohn-Kanade Dataset (CK+). The expression classification result shows that the SAE performances the classification effectively and successfully. Further experiments (transformation and purification) are carried out to illustrate the application of the feature extraction and input reconstruction ability of SAE.**

Keywords—facial expression recognition; deep learning; stacked sparse autoencoder; optical flow

I. INTRODUCTION

Facial expression conveys abundant and valuable information about emotion and thought of human beings. With an accurate comprehension of it, we are able to capture the state of people's inner world. The study of facial expression recognition has been developing for decades and the very early work on it could retrospect to the nineteenth century [1]. Nowadays, facial expression recognition has become an area of immense interest for its various applications in fields including human-computer interaction, user profiling, and image retrieval, etc. [2].

Computer-vision based facial expression recognition methods mainly discriminate six types of prototypical facial expressions (i.e. happiness, sadness, fear, anger, disgust and surprise) [3]. The researches on facial expression recognition can be generally divided into two categories [4]: the works based on static images and other studies who take dynamic video images as their research objects. As for the first category, most works make use of the appearance features (pixel information) or geometric features extracted from some selected points based on Facial Action Units (FAUs) [5]. Algorithms such as Gabor wavelet [6], Principal Component Analysis (PCA), Independent Component Analysis (ICA) are often used in this type of work. In terms of the second category, the feature-points tracking and optical flow are typical methods [7]. Since our research aims at facial expression recognition using image sequences, we consider using optical flow method with the expectation that dense motion information will be provided.

Optical flow method is a commonly used approach in video image sequence analysis and has a broad application in computer vision and image processing. The classical method like Horn-Schunck optical flow algorithm is based on the assumptions of gray-scale consistency [8]. Considering the non-rigid motion involved in facial expression, the traditional optical flow method will have a difficulty obtaining an accurate optical flow field. By applying modern optimization and implementation techniques, Deqing Sun, Stefan and Michael [9] improve the classical optical flow method, and we find their improved approach works well in obtaining optical flow field of facial muscle movement.

In this paper, a method based on optical flow analysis and stacked sparse autoencoder (SAE) for facial expression classification is presented. We extract optical flow field between certain frames in the video image sequence. Then, taking the optical flow vectors as the input of the first layer, we train every layer of the stacked SAE. By using this method, we classify the expression into six basic classes: happiness, sadness, anger, fear, disgust and surprise. Noticing the feature extraction and input reconstruction abilities of SAE, more experiments are carried out to additional applications of this method.

The rest of this paper is organized as follows: Section II and III give a brief introduction of optical flow and stacked SAE. Section IV describes our methodology in detail. Section V presents the experiments on Extended Cohn-

978-1-4799-5711-8/14/$31.00

Kanade Dataset (CK+). Finally, Section VI offers our conclusions.

II. OPTICAL FLOW

Considering the non-rigid motion involved in facial expression, we apply the improved optical flow method proposed in [9]. Compared with the traditional optical flow method, it performance better in obtaining an accurate optical flow field. In this section, we give a brief description of the improved optical flow approach. This advanced method improves the objective function of classical optical flow method by introducing non-local terms.

A. Classical Model

Based on the assumption of gray-scale consistency, optical flow method estimates the motion of objects involved by computing a flow field that minimizes the difference between two images, which can be described as:

$$E(u,v) = E_{\text{constancy}}(u,v) + E_{\text{smoothness}}(u,v) \tag{1}$$

where $E_{\text{constancy}}(u,v)$ and $E_{\text{smoothness}}(u,v)$ are the constancy and smoothness term respectively. u and v are the horizontal and vertical components of the optical flow field to be estimated. The specific expression of $E_{\text{constancy}}(u,v)$ and $E_{\text{smoothness}}(u,v)$ can be found in [9].

B. Improved Model

Based on the idea of interleaved median filtering, (1) can be adjusted into:

$$\begin{aligned} E_A(u,v,\hat{u},\hat{v}) = E(u,v) + E_{\text{coupling}}(u,v,\hat{u},\hat{v}) \\ + E_{\text{non-local}}(u,v,\hat{u},\hat{v}) \end{aligned} \tag{2}$$

where $\hat{u}$ and $\hat{v}$ represent an additional auxiliary flow field. In this advanced objective formulation, the coupling term E_{coupling} and the non-local term $E_{\text{non-local}}$ are introduced. Their specific expression is:

$$E_{\text{coupling}} = \lambda_1 \left(\| u - \hat{u} \|^2 + \| v - \hat{v} \|^2 \right) \tag{3}$$

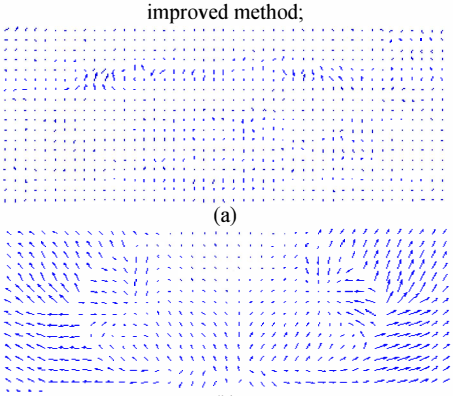
$$E_{\text{non-local}} = \sum_{i,j} \sum_{(i',j') \in N_{i,j}} \lambda_2 (| \hat{u}_{i,j} - \hat{u}_{i'j'} | + | \hat{v}_{i,j} - \hat{v}_{i'j'} |) \tag{4}$$

E_{coupling} is the function encouraging $\hat{u}$, $\hat{v}$ and u, v to be similar. $E_{\text{non-local}}$ imposes a particular smoothness assumption within a specified region, which represents the integrated information over a large spatial neighborhood. $N_{i,j}$ is the set of neighbors of pixel (i, j) and λ_1, λ_2 are the scalar weights.

Given non-rigid motions are involved, we propose to apply the improved model to compute optical flow of facial expressions. We make comparisons between the effect of the classical optical flow method and the improved method. In Fig.1, the optical flow fields computed by the classical model (Horn-Schunch method) and the improved method are warped back to the facial area of the first frame and the last frame respectively. The classical model fails to track the motion of the "key points" (such as the corners of mouth and eye brow) in facial expression while the advanced method does it more accurately. Specifically, as it is clearly

(a) (b) (c) (d)

Fig.1. (a) the first frame; (b) the peak frame; (c) the image warped with optical flow field computed by Horn-Schunch method; (d) the image warped with optical flow field computed by the improved method;



(a) (b)

Fig.2. (a) the mouth region of the optical flow field computed by Horn-Schunch method; (b) the mouth region of the optical flow field computed by the improved method;

shown in Fig.2, the flow vectors computed using the improved model regularly point to outward from the center, which better present the tendency of the movement of a grinning mouth.

III. SPARSE AUTOENCODER (SAE)

In this section we describe the architecture of sparse autoencoder, whose typical structure is depicted in Fig.3. An Autoencoder is a neural network that attempts to get the output $\hat{x}$ that approximate the input vector $\boldsymbol{x}$ with constraints of the total number and the entire activation of the hidden units imposed on during the training process. Let $\boldsymbol{a}_\mathbf{j}$ denotes the weighted sum at the **j**-th neuron $\boldsymbol{n}_\mathbf{j}$ (in the hidden layer and output layer), the output of this neuron is:

$$f(a_j) = f\left(\sum_{i=1}^{n} w_{ij} x_i + b_j \right) \tag{5}$$

where $\boldsymbol{x}_\mathbf{i}$ denotes the **i**-th component of the output of the preceding layer, $\boldsymbol{w}_\mathbf{ij}$ means the weight associating $\boldsymbol{x}_\mathbf{i}$ and $\boldsymbol{n}_\mathbf{j}$ and $\boldsymbol{b}_\mathbf{j}$ is the bias term. In order to introduce non-linear mapping into our system, we use sigmoid function ($\boldsymbol{f}$ **(*)**) as the activation function of neural units [10].

By minimizing the mean square error (MSE) between the input and the output, SAE generates an approximation of the input vector. The final objective of our task is described as:

$$\arg\min_{W_{1,2}, b_{1,2}} \sum_{i=1}^{n} \left\| f(W_2 \cdot f(W_1 \cdot x + b_1) + b_2) - x \right\|_2 \tag{6}$$

where $\boldsymbol{W}_\mathbf{1}$ is the weight matrix between the input layer and the hidden layer ("encoder"), $\boldsymbol{W}_\mathbf{2}$ is the weight matrix between the hidden layer and the output layer ("decoder"), and $\boldsymbol{b}_\mathbf{1}$, $\boldsymbol{b}_\mathbf{2}$ are the bias vectors respectively.

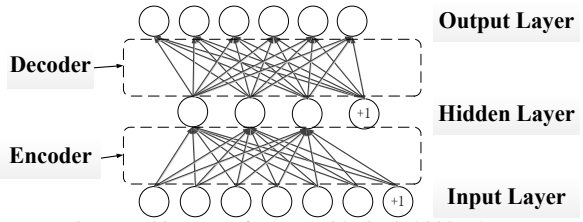


Fig.3. Architecture of a SAE with single hidden layer

The optimization problem shown in (6) can be solved by neural activation forward passing and error back propagation. The gradient of the parameters in SAE can be computed accurately so that we could use advanced optimization methods, such as L-BFGS [11], to handle the problem. By solving (6), we get the "encoder" and the "decoder" networks and their bias terms, thus we could do the reconstruction. Fig.4 is an example of reconstruction on the handwritten figures dataset MNIST [12].

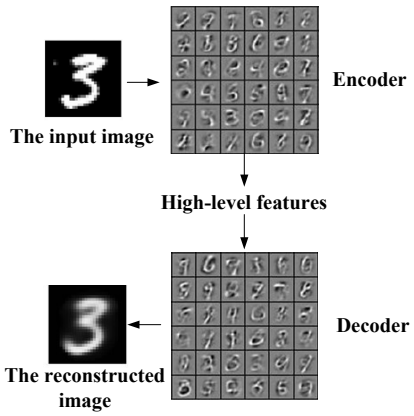


Fig.4. An example of reconstruction

IV. METHODOLOGY

Based on the improved optical flow method and the stacked SAE, our method is proposed to achieve two tasks of facial expression recognition. First of all, we complete facial expression classification. Then, considering the feature extraction and input reconstruction ability of SAE, we achieve facial expression transformation and purification using our method.

A. Facial Expression Classification

We compute the optical flow field between the first frame $\boldsymbol{I}_\mathbf{f}$ and the last frame $\boldsymbol{I}_\mathbf{l}$ in each sequence. Then, the horizontal and vertical components $\boldsymbol{v}_\mathbf{x}$, $\boldsymbol{v}_\mathbf{y}$ are reshaped and combined into a vector and is treated as the input of the stacked SAE.

As it is shown in Fig.5, the stacked SAE in our experiment has two hidden layers and each of them has 200 hidden units. We employ a softmax classifier on the final layer of the system to achieve classification. The output of the softmax classifier is the "activation label vector" whose components can be comprehended as the probability of the corresponding class the input belongs to, and we take the class who has the largest output (activation) as the predict result of the input.

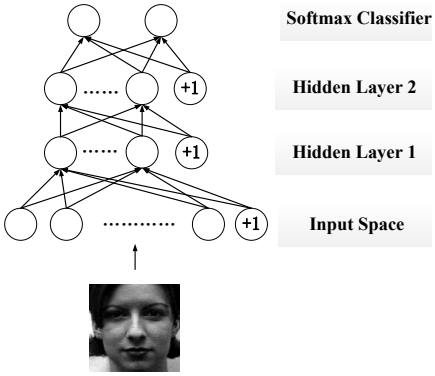


Fig.5. The stacked SAE in classification experiment

B. Facial Expression Transformation and Purification

1) Facial Expression Transformation

With the feature extraction ability of SAE, we could map the input image to a more compressed feature space. Since we have the "decoder" network, we are able to reconstruct the corresponding vector in the input space $\mathcal{D}$ once a vector in the feature space $\mathcal{F}$ is given (actually it is an approximation of the input vector in the output space, we will not distinguish the term "input space" and "output space" in the rest of the methodology section).

If two input images $\boldsymbol{I}_1$, $\boldsymbol{I}_2$ under different labels $\boldsymbol{L}_1$, $\boldsymbol{L}_2$ are known, we could get the features $\boldsymbol{f}_1$, $\boldsymbol{f}_2$ of each of them by applying the "encoder" network of a well-trained SAE. Suppose we have a feature $\boldsymbol{f}_\mathbf{c}$ in the feature space $\mathcal{F}$ which is on the "straight line" between two given features $\boldsymbol{f}_1$, $\boldsymbol{f}_2$, and then $\boldsymbol{f}_\mathbf{c}$ can be computed by solving

$$\min_{f_c} \lambda \left\| f_c - f_1 \right\|_2 + (1-\lambda) \left\| f_c - f_2 \right\|_2 \qquad (7)$$

By adjusting the parameter $\boldsymbol{\lambda}$, we could have the point $\boldsymbol{f}_\mathbf{c}$ travel from $\boldsymbol{f}_1$ to $\boldsymbol{f}_2$ linearly. Then we could use the "decoder" network to reconstruct the relating vector $\boldsymbol{I}_\mathbf{c}$ in the input space at any position of $\boldsymbol{f}_\mathbf{c}$. In this way, we achieve "transformation" from two given input in $\mathcal{D}$ through the feature space $\mathcal{F}$. The transformation process is illustrated in Fig.6.

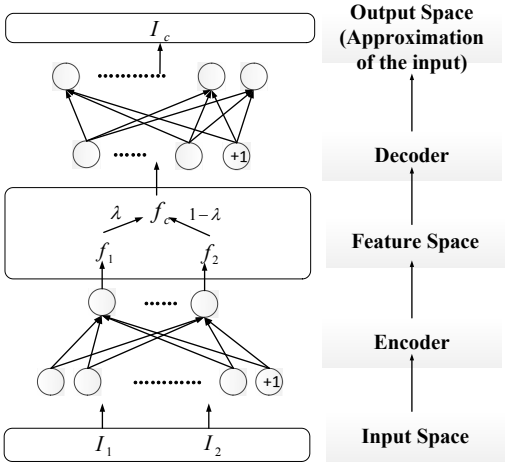


Fig.6.The transformation process

2) Facial Expression Purification

Each component in the output vector of the softmax classifier could be explained as the probability of the each corresponding class the input data belongs to. Inspired by this idea, we could revise some terms in (7) to:

$$\min_{f_p} \left\| L\left(f_p\right) - L \right\|_2 + \left\| f_p - f \right\|_2 \tag{8}$$

$\boldsymbol{L}$**(*)** is the operation we compute the output of the input feature $\boldsymbol{f}$. $\boldsymbol{L}$ is the expected output label we set according to the input, e.g. if the input $\boldsymbol{f}$ belongs to class k, thus we set the component of $\boldsymbol{L}\triangleq(l_1,l_2,\ldots,l_n)$ to be

$$l_i = \begin{cases} 0 & \left(i \neq k\right) \\ 1 & \left(i = k\right) \end{cases} \tag{9}$$

which we consider the most purest label of the input. The first term in the revised object function "purifies" the label of $\boldsymbol{f}$ and the second term encourages $\boldsymbol{f}_p$ to keep the style of $\boldsymbol{f}$. By solving (8), we could get the "purified" feature $\boldsymbol{f}_p$ which does not only keep the characteristic of the input $\boldsymbol{f}$ but also get a purer output label from the network. After that, we could use the "decoder" network to reconstruct the relating vector of $\mathbf{I}_p$ in the input space.

V. EXPERIMENTS

A. Database and Preprocessing

1) Database

Experiments concerning facial expressions are carried out on the Extended Cohn-Kanade Dataset (CK+), which have 593 sequences across 123 subjects with landmarks given [13]. Each of the sequences contains images from the onset expression (first frame) to the peak expression (last frame). All emotion labels (i.e. happiness, sadness, anger, fear, disgust and surprise) have been revised and validated according to the FACS Investigators Guide [14] on our own. To enlarge the scale of the training samples, we computed the optical flow field between the first frame and the last three frames in every single sequence.

2) Preprocessing

Most images in CK+ database have the resolution of 640*490 or 640*480, which is too large to put in a deep neural network. Since only the information in the facial area is needed, we cut out the facial optical flow field based on the landmarks given. Taking the midpoint of the inner corner of the left and right eyes as the base point, we cut out the facial area and down-sampled it to 37×36 pixels. The process is shown in Fig.7.

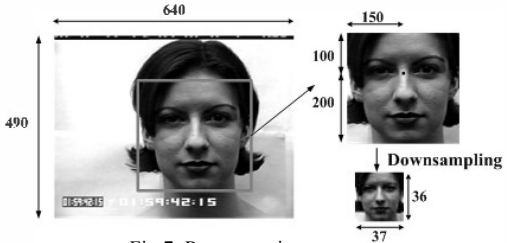


Fig.7. Preprocessing

TABLE I EXPRESSION CLASSIFICATION ACCURACY ON CK+

	Angry	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**
Angry	24	0	0	0	3	0
Disgust	1	30	0	1	0	0
Fear	1	0	15	0	0	0
Happy	1	0	0	23	0	0
Sad	3	0	0	0	21	3
Surprise	0	0	0	0	0	24
Accuracy	80.0%	100.0%	100.0%	95.8%	87.5%	88.9%

B. Facial expression classification

Generally, we classify the facial expressions into six basic classes: happiness, sadness, anger, fear, disgust and surprise. After revising the labels of each sequences provided in CK+, we pick out 50 sequences whose facial muscle motion is little even in the last frame (peak face), therefore it is hard to classify them into any of the six classes mentioned above. Some examples of these sequences are presented in Fig.8, please note that all of them are the last frame (peak face) in the image sequence. We take the rest 543 sequences in the database as the training samples and test samples in our classification experiments.

In the training process, we randomly select 1392(493*3) sequences to train SAE and 150(50*3) images as test samples. The classification results are presented in Table 1, where the average recognition accuracy rate is 91.3%. This proves that our method performs the classification effectively. (Note: the terms in first line are test labels while the terms in first column are predicted labels.)

Fig.8. Examples of peak faces with little changes

C. Transformation and Purification

In order to verify our method of transformation and purification clearly, we first carry out the related experiments on MNIST. Then, we apply the experiments of transformation and purification to the facial expression on CK+.

1) On MNIST

a) Transformation

In order to achieve a good reconstruction effect, we use all 60000 samples in MNIST to train our SAE. The input layer and the output layer both have the length of 784 (28*28) which is same to the size of the input data. Our SAE has only one hidden layer which contains 200 neurons. The input images are the figure "3" and "8" which are shown in Fig.9.

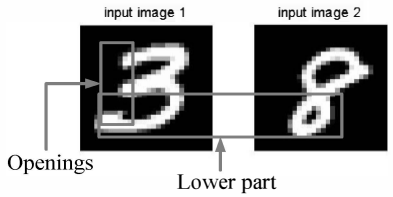


Fig.9. The input images of transformation experiment on MNIST

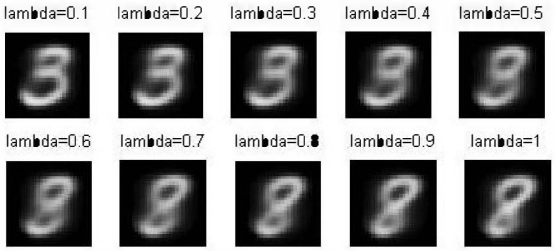


Fig.10. The transformation process from "3" to "8"

In order to show the details in the transformation process, we set the increasing step of parameter λ to be 0.1. The result is shown in Fig.10. We could clearly see that the entire transformation process proceeds smoothly. Specifically, the openings on the left side of the figure "3" have become closed at the end of the transformation process and the lower part of the figure "3" becomes thinner to fit the lower part of the figure "8" gradually.

b) Purification

In the experiment of purification, we proposed a stacked SAE since we will make use of the output of the softmax classifier. The input original image of figure "3" is shown in Fig.11 (a), and its label is shown in the first row of Table 2. We could clearly see that the largest component of the output is 0.2798 under class 3, which represents the probability of the figure in the input image is the figure "3". Therefore we finally predict the figure in the input image to be "3", which is the correct answer. However, the component of the output under class 2 (0.1200), class 5 (0.1879) and class 8 (0.1526) is also big compared to class 3 (0.2798), and this could be explained as the figure in the input image also has a high probability, though not the highest, to be the figure "2", "5", or "8".

We set the objective label to be [0, 0, 1, 0, 0, 0, 0, 0, 0, 0] , which is the "purest" label of the input image, and then we solve the optimization function (10). The result is shown in Fig.11 (b) and its purified label is shown in the second row of Table 2. The component under class 3 is now 0.6309, which is much larger than any other components. This means that the figure in the input image is much more "likely" to be 3 than any other figures. Besides, in Fig.11 (b), the figure "3" is much more "upright" than before, meanwhile lean to left side to keep its previous "style".

2) On CK+

a) Expression Transformation

We further apply our experiment to image sequences of facial expressions in CK+. As it is exhibited in Fig.12, the

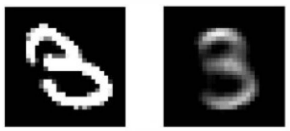


Fig.11. (a) the original input image of figure "3";
(b) the purified figure "3"

two inputs are the optical flow field $\boldsymbol{I}_1$ (between the normal face and the surprise face) and $\boldsymbol{I}_2$ (between the normal face and the sad face). After getting the "combined" optical flow field $\boldsymbol{I}_c$ from $\boldsymbol{I}_1$ and $\boldsymbol{I}_2$, we warp it back to the normal face and get the final result (Fig. 13). It can be easily seen that the facial expression of the woman turns from a surprise face to a sad face smoothly. To be specific, the woman firstly "closes" her mouth and then "bends" it down coherently. She squints her eyes gradually during the whole process as well.

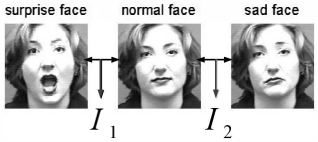


Fig.12. The optical flow fields I_1 , I_2 are taken as input

Fig.13. The transformation from surprise to sad

b) Expression Purification

The input of the purification experiment on CK+ is the optical flow between the normal face and the peak face. In Fig.14, we present 3 examples of the result. The optical flow fields of image pair (a1, b1), (a2, b2) and (a3, b3) are treated as the input and their original labels are show in Table 3. After setting the objective labels according to the classes the facial expressions belong to, we do the purification and the results are shown as (c1), (c2) and (c3) in Fig. 14, and their labels are shown in Table 3.

It could be seen that the underlined components (components related to the activation of the corresponding class) become larger after the purification. This means that the purified facial expressions have a greater probability to be classified into the corresponding class. To be specific, the corners of mouth in (c1) and (c2) are raised up than that in (a1) and (b1) respectively, which make the facial expression more like smiling. In (b3), the woman bends her mouth downward, so it may be considered as a sad face. However, the label shows that it should be an anger face. From (b3) to (c3), the corners of the mouth are raised up and the lips are compressed, which makes it look more like an angry face.

TABLE II OUTPUT LABELS ON MNIST

Classes	1	2	3	4	5	6	7	8	9	10
Original image	0.0168	0.1200	**0.2798**	0.0306	0.1879	0.0839	0.0294	0.1526	0.0320	0.0664
Purified image	0.0099	0.0788	**0.6309**	0.0129	0.0984	0.0249	0.0210	0.0757	0.0181	0.0289

TABLE III OUTPUT LABELS ON CK+

Classes		1	2	3	4	5	6
Facial expression		Angry	Disgust	Fear	Happy	Sad	Surprise
1	(a1,b1)	0.0979	0.2758	0.2067	**0.3038**	0.1154	0.0966
	(a1,c1)	0.0341	0.1167	0.1498	**0.7287**	0.0554	0.0444
2	(a2,b2)	0.0191	0.0212	0.1626	**0.7355**	0.1084	0.0031
	(a2,c2)	0.0102	0.0131	0.1124	**0.8961**	0.0581	0.0017
3	(a3,b3)	**0.6026**	0.2945	0.0511	0.0033	0.2007	0.0077
	(a3,c3)	**0.7551**	0.2243	0.0374	0.0019	0.1366	0.0055

Fig.14. (a1), (a2), (a3): the onset face;
(b1), (b2), (b3): the peak face without purification;
(c1), (c2), (c3): the purified peak face

VI. CONCLUSION

In this paper, a method based on the combination of optical flow and a deep neural network—stacked sparse autoencoder (SAE) is presented. This method could achieve facial expression classification, transformation and purification. We employed the improved optical flow method to compute the optical flow of the facial expression where non-rigid motion is involved. Then, the stacked SAE are trained with the optical flow field as the input to extract high-level features.

As for the experiments, the result of the classification experiment shows that our method does the classification work effectively and achieves a high accuracy. Further experiments on transformation and purification on MNIST and CK+ reveals the feature extraction and input reconstruction capacity of SAE. These experiments could be expected to have a broad application, e.g. image sequence analysis, motion tendency prediction and modification of specific image patterns.

ACKNOWLEDGEMENT:

This work was supported by the Beijing Higher Education Young Elite Teacher Project (YETP0104), the Tsinghua University Initiative Scientific Research Program (20131089382), and the National Natural Science Foundation of China (61101152).

REFERENCES

[1] B. Fasel and J. Luettin, "Automatic facial expressin analysis: a survey," Pattern Recognition, vol. 36, pp.259-275, 2003.

[2] M. Pantic, A. Pentland, A. Nijholt, T.S. Huang, "Human Computing and Machine Understanding of Human Behavior: A Survey," Computer Science, vol. 4451, pp. 47-71,2007.

[3] I. Cohen, N. Sebe, A. Garg, L.S. Chen, T.S. Huang, "Facial expression recognition from video sequences: temporal and static modeling," Computer Vision and Image Understanding, vol. 91, pp.160-187, July-August 2003.

[4] P.Ekman, W.Friesen, Facial Action Coding System: A Technique for the Measurement of Facial movements, Consulting Psychologists Press, California, 1978.

[5] P. Ekman and W.V. Friesen, "Facial Action Coding System," Consulting Psychologist Press, Palo Alto, CA,1978.

[6] Liu, Wei Feng, and ZengFu Wang. "Facial expression recognition based on fusion of multiple Gabor features." ICPR 2006. 18th International Conference on. vol.3. IEEE, 2006.

[7] C. Shana, S. Gongb, P.W. McOwanb, "Facial expression recognition based on Local Binary Patterns: A comprehensive study," Image and Vision Computing, vol. 27, pp. 803-816, 4 May 2009.

[8] A. Bruhn, J. Weickert, C. Schnörr, "Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods," International Journal of Computer Vision, vol.61, pp.211-231, February 2005.

[9] Sun, Deqing, Stefan Roth, and Michael J. Black. "Secrets of optical flow estimation and their principles." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.

[10] http://deeplearning.stanford.edu/wiki/index.php/Autoencoders_and_Sparsity

[11] C. Zhu , R. H. Byrd and J. Nocedal, "L-BFGS-B: Algorithm 778: L-BFGS-B: Fortran routines for large-scale bound-constrained optimization."ACM Trans. Math. Softw., vol. 23, no. 4, pp. 550-560, 1997.

[12] LeCun, Yann, and Corinna Cortes. "The MNIST database of handwritten digits, 1998." URL: http://yann. lecun. com/exdb/mnist.

[13] Lucey, Patrick, et al. "The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression." Computer Vision and Pattern Recognition Workshops, 2010.

[14] Ekman, Paul. Facial action coding system. Salt Lake City: A Human Face, 2002.