IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 37, No. 3, June, 2020, pp. 503-509
Journal homepage: http://iieta.org/journals/ts

An Automatic Recognition Method for Students' Classroom Behaviors Based on Image Processing

Wei Huang[1], Ning Li[2], Zhijun Qiu[1*], Na Jiang[1,2], Bin Wu[1], Bo Liu[1]

[1] Yueyang Vocational and Technical College, Yueyang 414000, China
[2] SEHAN University, Mokpo 58447, South Korea

Corresponding Author Email: janezj2004@yahoo.com

https://doi.org/10.18280/ts.370318

Received: 17 December 2019
Accepted: 29 March 2020

Keywords:
classroom behavior analysis, head pose, facial expression, image processing

ABSTRACT

The classroom behaviors of students are the key objects in teaching analysis. It is very important to quantify such behaviors in an intuitive and dynamic manner. This paper summarizes the typical classroom behaviors of students, and specifies the steps to preprocess the collected sample images on these behaviors. Then, it is decided to discriminate students' classroom behaviors by head poses and facial expressions. Next, a positioning method for facial feature points was developed based on deep convolutional neural network (D-CNN) and cascading, and the head poses and facial expressions were analyzed and recognized. Our method was compared with other facial expression recognition algorithms. The results show that our method is more robust and accurate than the contrastive algorithms.

1. INTRODUCTION

The classroom behavior problems of students are increasingly prominent, owing to the proliferation of smart mobile terminals and computer networking technology. If the classroom behaviors of students are identified and analyzed accurately, it will be easy to effectively manage classroom teaching and objectively evaluate the learning engagement of students [1-6]. The students' classroom behaviors, as the main objects in teaching analysis, have been discussed by many scholars. In most cases, the data on these behaviors are collected from teachers' summary of classroom activities, questionnaire surveys on students, or interviews on students. There is little report that quantifies the classroom behaviors of students in an intuitive and dynamic manner [7-9]. Considering the maturity of face recognition technology, it is very necessary to integrate image processing into the recognition of students' classroom behaviors. The integration would be in line with the trend of information society [10-12].

As a hotspot and difficulty in image processing, human behavior recognition is widely used in many fields, namely, medical care, finance, and property management. However, it is immensely difficult to recognize the classroom behaviors of students [13-16]. Mollahosseini et al. [17] observed the behaviors of teachers and students in teaching videos, recorded and analyzed the student-teacher (S-T) interactions, and developed an S-T classroom behavior analysis method. Kim et al. [18] proposed a method that processes human motions in real time: Based on the information of human motions, the regions of interest (ROIs) were determined in specified parts, and the oriented gradient of each ROI was calculated; After that, human motions were classified by a two-dimensional (2D) convolutional neural network (CNN) algorithm. To recognize different human poses, Lopes et al. [19] created an unsupervised recognition method for static human behavior images, which calculates the image differences through global matching of human contours, and then optimizes the differences through minimization and clustering. Zavarez et al. [20] designed a Kinect pose estimation system capable of decomposing complex color images into single-color, single-frame images for analysis; the system can quickly detect the positions of major human joints in any single-frame image, and thus recognize and classify human behaviors.

Drawing on the relevant research, this paper puts forward a suitable recognition method for students' classroom behavior based on image processing. Firstly, the authors summarized the steps to preprocess the collected sample images on classroom behaviors, and decided to discriminate students' classroom behaviors by positioning facial feature points, estimating head pose angles, and recognizing facial expressions. Next, a positioning method for students' facial feature points was developed based on deep CNN (D-CNN) algorithm and cascading, and the head poses and facial expressions were analyzed and recognized. The proposed method was proved effective through experiments.

2. PREPROCESSING OF CLASSROOM BEHAVIOR IMAGES

To make students more attentive in classroom learning, this paper constructs an analysis model for complex learning activities in class based on four typical classroom behaviors of students, namely, raising head, lowering head, bending over desk, and looking around. The analysis results provide the reference for teachers to manage and evaluate informatized teaching, appraise classroom performance, etc.

Without considering the effects of objective factors like gender, body type, clothing, and teaching environment, the collected images were cropped and preprocessed as shown in Figure 1.

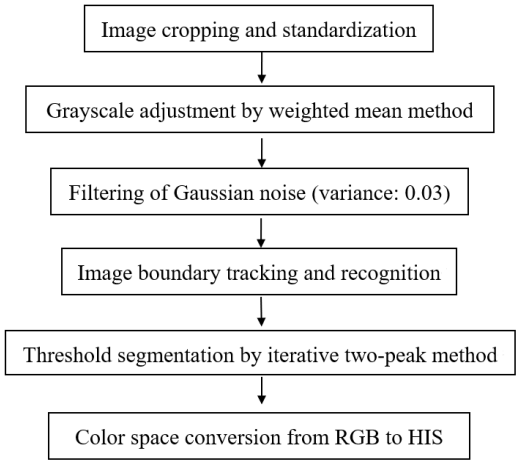

Figure 1. Steps of image preprocessing

The sample images, captured by cellphones, are color images with a large amount of information and a high degree of complexity. The grayscale of each image should be adjusted to remove the redundant information brought by colors and speed up computer processing. During gray processing, each pixel of the image was normalized by:

$$a' = \frac{a - \overline{w}_{neig}}{\sigma_{neig}} \tag{1}$$

By the above formula, the grayscale a of a pixel is subtracted by the weighted mean grayscale w_{neig} of its neighbors; then, the difference is divided by the mean squared error (MSE) of the grayscales of its neighbors, producing the new grayscale a' of that pixel.

Then, the images went through a necessary denoising process to prevent false recognition induced by objective factors like transmission interference, light intensity interference, and equipment errors. The denoising process reduced the noise variance proportionally, without changing the mean level of noise, that is, effectively suppressed noise.

To acquire effective image features, the region boundary tracking method was adopted to find the boundaries between pixels with grayscale of zero and those with grayscale of one, thus identifying the boundaries of the cropped image.

After the boundaries have been defined, each student should be separated from other personnel or the background, laying the basis for accurate feature recognition of students' classroom behaviors. Here, the iterative two-peak method is employed to delineate and segment the target student and background from the grayscale image with a suitable threshold.

Finally, the essential features of image colors were clustered to create a complete color space with high diversity. Through the change of bases, the RGB (red-green-blue) color space was converted into the HSV (hue-saturation-value) color space.

3. AUTOMATIC RECOGNITION OF CLASSROOM BEHAVIORS

The existing recognition models for classroom behaviors mostly target head poses of students, failing to process the original images on students' facial expressions. The recognition of facial expressions directly bears on the recognition rate of students' classroom behaviors. Therefore, this paper proposes a classroom behavior recognition method based on the recognition of head poses and facial expressions. The proposed method discriminates classroom behaviors by positioning facial feature points, estimating head pose angles, and recognizing facial expressions. The entire process covers four steps shown in Figure 2.

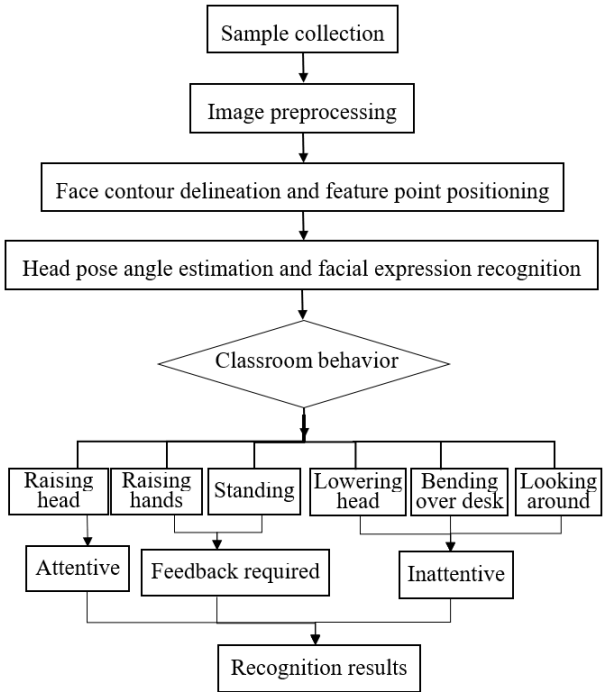

Figure 2. Steps of classroom behavior recognition

3.1 Positioning of facial feature points

Before positioning facial feature points, the collected images should be compensated for illumination, making the illumination evenly distributed. Let r, g and b be the values of the three primary colors R, G and B, respectively. Then, the mean grayscale of the image and the weights of the three primary colors can be respectively expressed as:

$$\overline{GV} = \frac{\overline{r} + \overline{g} + \overline{b}}{3} \tag{2}$$

$$W_r = \frac{\overline{GV}}{r}, W_g = \frac{\overline{GV}}{g}, W_b = \frac{\overline{GV}}{b} \tag{3}$$

Then, the three primary color components of each pixel in the image can be adjusted to:

$$r' = r \times W_r, g' = g \times W_g, b' = b \times W_b \tag{4}$$

To ensure the positioning speed and robustness, the D-CNN algorithm and cascading were adopted to position the facial feature points in four steps:

Step 1. Define the boundaries of the facial organs of the student on the image after illumination compensation.

Step 2. Determine the preset positions of 50 facial feature points to form a dot matrix.

Step 3. Position the eyes accurately according to the layout of human face.

Step 4. Output the positions of 80 key points, including the facial boundary points and facial feature points.

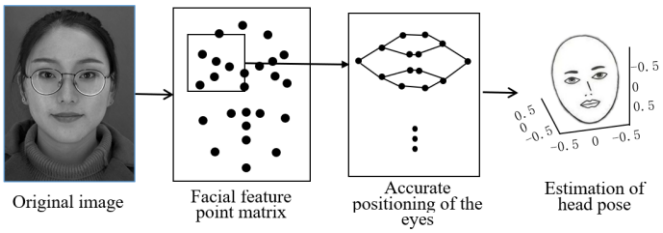

Figure 3. Steps of facial feature point positioning

The cascaded network, consisting of 3 convolution layers and 3 pooling layers. First, the face image of each student was convoluted by the following formula and constraint:

$$V: f(F, s_{ij}) \to \Delta_{sij}, \quad \left\| s_i + \Delta_{si} - s_i' \right\|_2^2 = 0 \tag{5}$$

where, V is the update process of feature points; F is the face image of a student; s_{i0}, s_{ij}, and s_{ij}' are the initial, regressed, and actual positions of the i-th facial feature point, respectively; f is the relationship between F and s_{ij}; Δ_{sij} is the position change of the i-th facial feature point.

The convoluted image was processed by max pooling with the main regressor, which is formed by a series of m sub-regressors:

$$C = c_1 \cdot c_2 \cdots c_i \cdots c_m \tag{6}$$

where, c_i depends on the symmetric matrix A_m obtained from the training set of facial feature points and the relative change Δr of the i-th sub-regressor. Once the main regressor C is trained, the initial position of facial feature point s_{i0} was updated by c_1 to s_{i1}. Then, s_{i1} was inputted to c_2 for further update. The position was updated iteratively until the final position s_{im} was outputted:

$$s_{im} = s_{i(m-1)} + A_m \cdot f\left(F, s_{i(m-1)}\right) + \Delta_r \tag{7}$$

The D-CNN algorithm alone may lead to errors, omissions and false recognitions of facial feature points. To solve these problems, this paper further applies D-CNN algorithm and cascading to check the presence of eyes in the facial area. If eyes are detected, it is more suitable to simulate the colors of the eyes with HSV color space than RGB color space. The HSV color space can characterize the brightness of colors. Suppose $u_{\max}=\max(r, g, b)$ and $u_{\min}=\min(r, g, b)$, the RGB can be converted into HSV by:

$$H = \begin{cases} \left(\dfrac{g-b}{u_{\max} - u_{\min}} + 0 \right) \times 80, r = u_{\max} \\ \left(\dfrac{b-r}{u_{\max} - u_{\min}} + 3 \right) \times 80, g = u_{\max} \\ \left(\dfrac{r-g}{u_{\max} - u_{\min}} + 5 \right) \times 80, b = u_{\max} \end{cases} \tag{8}$$

Then, the Hough transform algorithm was employed to process the black circles in the image. After acquiring radius and center coordinates, all the black circles that are not eyes were filtered out. Firstly, the area around 3-5 black circles was scanned. The black circles in the area were removed, except the two black circles with the largest number of black pixels. If the area was found to contain no eyes, the image would be further confirmed by the skin tone model. Figure 4 provides examples of eye region processing.

Figure 4. Examples of eye region processing

In addition, the D-CNN algorithm cannot detect facial feature points, if the student's head is deflected, i.e. the student is lowering head or bending over desk. In this case, the image was further detected by the skin tone model. The color space of the image was converted into YCbCr, which can separate brightness from chromaticity:

$$\begin{bmatrix} y \\ c_b \\ c_r \end{bmatrix} = \begin{bmatrix} 18 \\ 144 \\ 144 \end{bmatrix} + \begin{pmatrix} 63.4 & 133.6 & 18.2 \\ -18.9 & -72.3 & 89 \\ 109 & -101.3 & -22.5 \end{pmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix} \tag{9}$$

To position the facial contours accurately, the kernel size in the last convolution layer was set to 8×8, 4×4, and 3×3. Moreover, the step size was set to 1 to maintain the integrity of local feature points. Then, the position of each facial feature point can be computed by:

$$y_{ij} = \Phi\left(\sum_{i \in P_j} y_{(i-1)j} \times k_{ij} + \delta_{ij}\right) \tag{10}$$

where, P_j is the set of initial positions of facial feature points; k_{ij} is the preset weight coefficient; δ_{ij} is bias; Φ is the activation function. For stable convergence, the rectified linear unit (ReLU), capable of solving vanishing gradient, was selected as the activation function:

$$\Phi(x) = \begin{cases} x, x > 0 \\ 0, x \le 0 \end{cases} \tag{11}$$

3.2 Solving head poses

As shown in Figure 5, our recognition method can estimate a total of nine head poses by comparing the relative positions of eyes in the image (obtained in the previous subsection) with the relative positions in the forward-looking faces of Figure 4.

The left eye, right eye, and nostrils were selected as inputs for head pose discrimination. The head pose angle was solved by:

$$H_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \tag{12}$$

$$H_y(\beta)=\begin{bmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{bmatrix} \tag{13}$$

$$H_z(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix} \tag{14}$$

where, α, β and θ are the rotation angles about axes x, y and z, respectively.

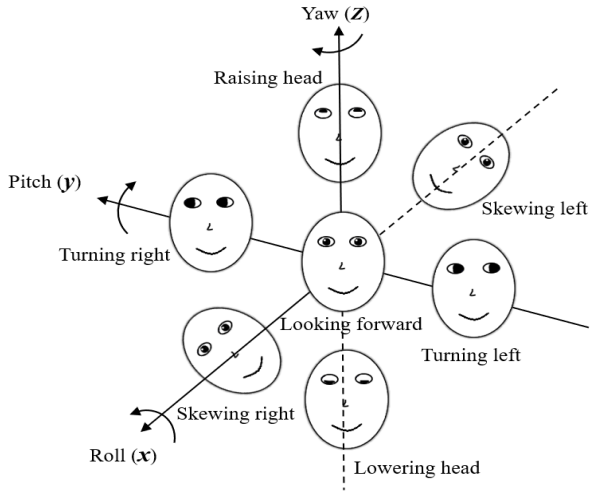

Figure 5. Different head poses defined in coordinate system

As the student changed his/her head poses, the rotation matrix and translation matrix can be respectively expressed as:

$$H=\begin{bmatrix}h_1 & h_2 & h_3\\ h_4 & h_5 & h_6\\ h_7 & h_8 & h_9\end{bmatrix},\ T=\begin{bmatrix}t_1\\ t_2\\ t_3\end{bmatrix} \tag{15}$$

The distance between left and right eyes on the y axis can be derived from the angles in the three-dimensional (3D) space:

$$\begin{aligned}\theta_Y &= \alpha\tan 2(h_8,h_9)\\ \theta_p &= \beta\tan 2(-h_7,\sqrt{h_9^2+R_9^2})\\ \theta_R &= \theta\tan 2(h_4,h_1)\end{aligned} \tag{16}$$

where, θ_Y, θ_P, and θ_R are the yaw angle, pitch angle, and rotation angle of the student's head, respectively.

Next, the y axis distance between left and right eyes under each head pose was compared with that of the forward-looking face. After that, the distance between the right eye and the right boundary of face under each head pose was compared with that of the forward-looking face. If the difference is smaller than 30 pixels, then the student must be looking forward to the podium; otherwise, the head pose should be determined against Figure 5.

3.3 Facial expression recognition

Based on the determination of head poses, the facial expressions were recognized through training on facial feature points extracted from key parts. The number and positions of expression feature points in the face image of a student were determined, using head pose estimation and facial feature extraction network. The initial positions of expression feature points were estimated by:

$$y_{ij}=\begin{cases}\rho_Y H(\rho_P)P\cdot\text{Model}+\rho_R, & i=0 \text{ and } j>0\\ F(\rho_Y,\Delta_Y)H(F(\rho_P,\Delta_p)P\cdot\text{Model}+F(\rho_R,\Delta_R), & i>0 \text{ and } j>0\end{cases} \tag{17}$$

where, ρ_Y, ρ_p and ρ_R are the yaw, pitch and rotation parameters computed by the head pose estimation network, respectively; Δ_Y, Δ_p and Δ_R are the yaw, pitch and rotation disturbances, respectively; P is the orthogonal projection matrix; F is the uniform probability density function; Model is the existing face parameter model.

After estimating the initial positions of expression feature points, the feature information at these positions was extracted on the expression feature extraction layer of the CNN. Then, the expression feature descriptor was generated and imported, together with the feature information, to the nonlinear regression layer of the cascade network. Let τ_i be the weight coefficient of the i-th sub-regressor. Then, the difference between the estimated and actual positions of a facial feature point can be minimized by:

$$E=\min_{\tau_j}\sum(DNN(D_i,s_{ij})\tau_j-s'_{ij})^2 \tag{18}$$

where, DNN is the expression feature descriptor; D_i is the i-th sub-regressor in the expression feature extraction layer. A supervised facial expression training model was adopted to assign an expression label to the face image of each student, e.g. happy, sad, surprised, disgusted, and angry. The workflow of facial expression estimation is summed up in Figure 6.

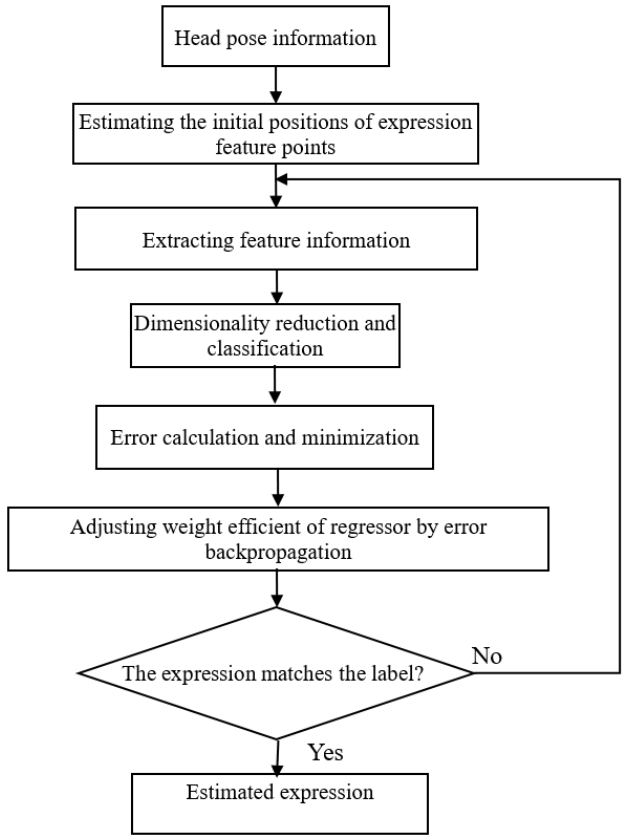

Figure 6. Steps of expression estimation

4. EXPERIMENTS AND RESULTS ANALYSIS

This section mainly tests the accuracy of the proposed automatic recognition algorithm for students' classroom behaviors, and compares it with popular methods like principal component analysis (PCA), Adaptive Boosting (AdaBoost) algorithm, and local binary pattern (LBP) algorithm.

Since some collected images are occluded, the areas in each image that cover facial contours and facial feature points were compared with artificially labeled feature areas. The collected images were removed unless the overlap between the said two kinds of areas is greater than 60%. The remaining images were divided into a training set and a test set.

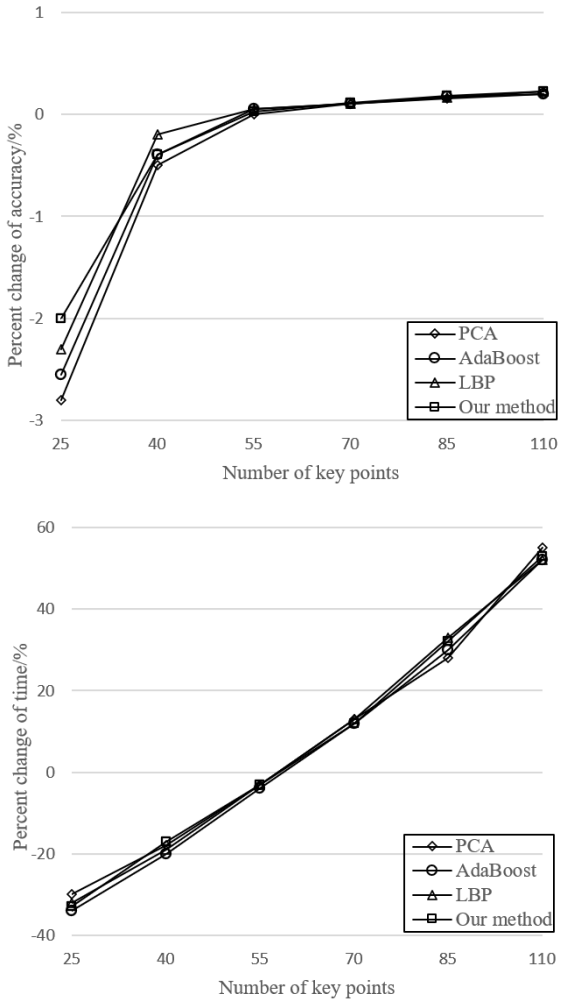

Figure 7. Effects of the number of key points on accuracy and training time

Figure 7 illustrates the variation in recognition accuracy and training time with the number of key points. It can be seen that, with the growing number of key points, the recognition became increasingly accurate, because the head pose and facial expression can be described more precisely with more facial feature points. However, when the number of key points reached and surpassed 80, the recognition accuracy changed slightly, while the training time exhibited a linear increase. Hence, the number of key points should be controlled at 80.

Next, our method and the contrastive methods were separately applied to estimate the yaw, pitch and rotation angles of the heads in the test images. The test results (Table 1) show that our method achieved higher accuracy than the other methods. The superiority is attributed to the fact that our method preprocesses the images and compensates for the error induced by brightness before head pose estimation.

Figure 8 compares the facial expression recognition accuracies of different methods. Obviously, our method generated the most stable accuracy curve among all methods, revealing that our method is more robust than other methods; the facial expressions estimated by our method are more realistic than those of other methods.

In our method, the D-CNN algorithm and cascading are fully combined to extract rich and diverse head poses and facial expressions from student images. Figure 9 shows the recognition accuracies of our method for different classroom behaviors. It can be seen that the mean accuracy was greater than 93%. This is because our model does not rely on the positioning accuracy of facial feature points. Instead, the facial expressions are recognized through training on estimated head poses; the estimated poses and facial expressions are both considered to recognize the classroom behaviors. Despite relatively high complexity, our method boasts better effect than other methods.

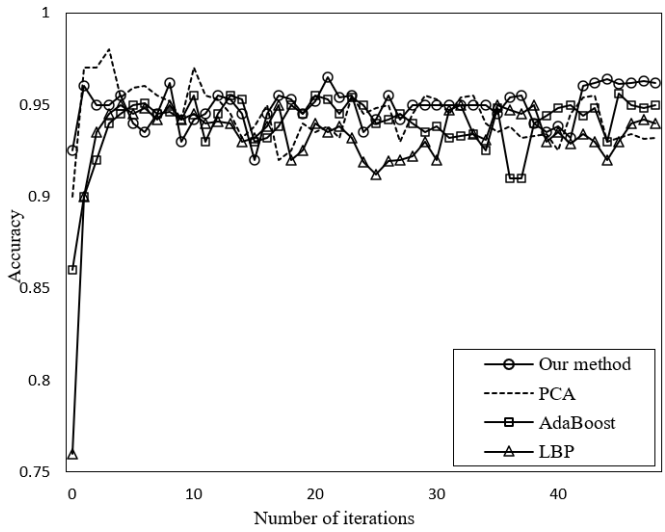

Figure 8. Facial expression recognition accuracies of different methods

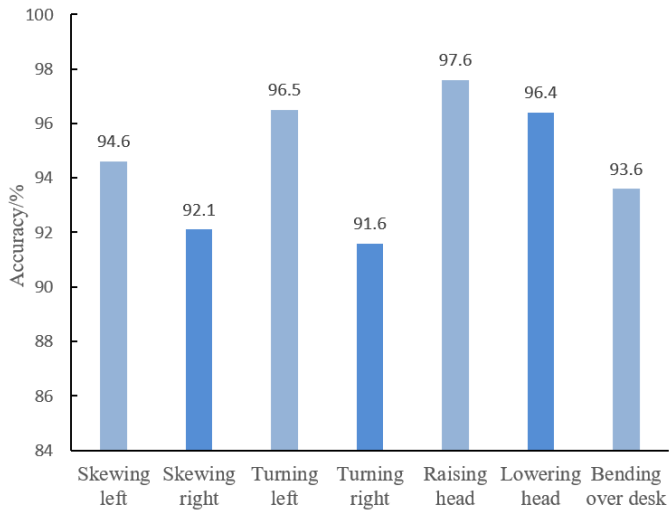

Figure 9. Recognition accuracies of different classroom behaviors of students

Table 1. Head pose test results of different methods

Method	$\Delta_{\theta Y}$/°	$\Delta_{\theta P}$/°	$\Delta_{\theta R}$/°	False alarm rate/%	Missed alarm rate/%	Accuracy/%	Mean recognition time/s
Our method	4.0	4.1	4.8	6.25	0	90.5	2.12
PCA	13.5	12.7	1.64	9.44	0	89.7	6.57
AdaBoost	6.8	9.2	5.9	4.50	7	86.4	5.60
LBP	10.2	11.9	12.5	24.5	0	92.5	3.75

5.CONCLUSIONS

This paper summarizes the typical classroom behaviors of students, and specifies the steps to preprocess the collected sample images on these behaviors. On this basis, the authors decided to discriminate students' classroom behaviors by head poses and facial expressions, developed a positioning method for facial feature points based on D-CNN algorithm and cascading, and applied the method to solve head poses and recognize facial expressions. Through experimental verification, the proposed method was found more robust and accurate than other popular facial expression recognition algorithms. The experimental results show that our method can correctly recognized 93.7%, 97.6%, 94.6%, and 93.6% of four typical classroom behaviors of students, with a mean recognition rate above 93%. Despite relatively high complexity, our method boasts better effect than other methods.

REFERENCES

[1] Pramerdorfer, C., Kampel, M. (2016). Facial expression recognition using convolutional neural networks: State of the art. arXiv preprint arXiv:1612.02903.

[2] Khan, K., Mauro, M., Migliorati, P., Leonardi, R. (2017). Head pose estimation through multi-class face segmentation. 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, pp. 175-180. https://doi.org/10.1109/ICME.2017.8019521

[3] Alioua, N., Amine, A., Rogozan, A., Bensrhair, A., Rziza, M. (2016). Driver head pose estimation using efficient descriptor fusion. EURASIP Journal on Image and Video Processing, 2016(1): 2. https://doi.org/10.1186/s13640-016-0103-z

[4] Kang, M.J., Lee, J.K., Kang, J.W. (2017). Combining random forest with multi-block local binary pattern feature selection for multiclass head pose estimation. Plos One, 12(7): e0180792. https://doi.org/10.1371/journal.pone.0180792

[5] Patacchiola, M., Cangelosi, A. (2017). Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. Pattern Recognition, 71: 132-143. https://doi.org/10.1016/j.patcog.2017.06.009

[6] Ruiz, N., Chong, E., Rehg, J.M. (2018). Fine-grained head pose estimation without keypoints. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, pp. 2155-215509. https://doi.org/10.1109/CVPRW.2018.00281

[7] Ahn, B., Choi, D.G., Park, J., Kweon, I.S. (2018). Real-time head pose estimation using multi-task deep neural network. Robotics and Autonomous Systems, 103: 1-12. https://doi.org/10.1016/j.robot.2018.01.005

[8] Diaz-Chito, K., Del Rincon, J. M., Hernández-Sabaté, A., Gil, D. (2018). Continuous head pose estimation using manifold subspace embedding and multivariate regression. IEEE Access, 6: 18325-18334. https://doi.org/10.1109/ACCESS.2018.2817252

[9] Chen, K., Jia, K., Huttunen, H., Matas, J., Kämäräinen, J. K. (2019). Cumulative attribute space regression for head pose estimation and color constancy. Pattern Recognition, 87: 29-37. https://doi.org/10.1016/j.patcog.2018.10.015

[10] Hariri, W., Tabia, H., Farah, N., Benouareth, A., Declercq, D. (2017). 3D facial expression recognition using kernel methods on Riemannian manifold. Engineering Applications of Artificial Intelligence, 64: 25-32. https://doi.org/10.1016/j.engappai.2017.05.009

[11] Nugroho, M.A., Kusumoputro, B. (2017). Fuzzy vector implementation on manifold embedding for head pose estimation with degraded images using fuzzy nearest distance. 2017 the 3rd International Conference on Communication and Information Processing, Tokyo, Japan, pp. 454-457. https://doi.org/10.1145/3162957.3163020

[12] Gupta, A., Thakkar, K., Gandhi, V., Narayanan, P.J. (2019). Nose, eyes and ears: Head pose estimation by locating facial keypoints. ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, United Kingdom, pp. 1977-1981. https://doi.org/10.1109/ICASSP.2019.8683503

[13] Borghi, G., Fabbri, M., Vezzani, R., Cucchiara, R. (2018). Face-from-depth for head pose estimation on depth images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(3): 596-609. https://doi.org/10.1109/TPAMI.2018.2885472

[14] Abate, A.F., Barra, P., Bisogni, C., Nappi, M., Ricciardi, S. (2019). Near real-time three axis head pose estimation without training. IEEE Access, 7: 64256-64265. https://doi.org/10.1109/ACCESS.2019.2917451

[15] Talegaonkar, I., Joshi, K., Valunj, S., Kohok, R., Kulkarni, A. (2019). Real time facial expression recognition using deep learning. Proceedings of International Conference on Communication and Information Processing (ICCIP) 2019, Pune, India. http://dx.doi.org/10.2139/ssrn.3421486

[16] Alshamsi, H., Meng, H., & Li, M. (2016). Real time facial expression recognition app development on mobile phones. 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, pp. 1750-1755. https://doi.org/10.1109/FSKD.2016.7603442

[17] Mollahosseini, A., Chan, D., Mahoor, M.H. (2016). Going deeper in facial expression recognition using deep neural networks. 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, pp. 1-10. https://doi.org/10.1109/WACV.2016.7477450

[18] Kim, D.H., Baddar, W.J., Jang, J., Ro, Y.M. (2017). Multi-objective based spatio-temporal feature representation learning robust to expression intensity

variations for facial expression recognition. IEEE Transactions on Affective Computing, 10(2): 223-236. https://doi.org/10.1109/TAFFC.2017.2695999

[19] Lopes, A.T., de Aguiar, E., De Souza, A.F., Oliveira-Santos, T. (2017). Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. Pattern Recognition, 61: 610-628. https://doi.org/10.1016/j.patcog.2016.07.026

[20] Zavarez, M.V., Berriel, R.F., Oliveira-Santos, T. (2017). Cross-database facial expression recognition based on fine-tuned deep convolutional network. 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Niteroi, pp. 405-412. https://doi.org/10.1109/SIBGRAPI.2017.60