Real-Time Face Detection Using a Moving Camera

Deng-Yuan Huang
Department of Electrical Engineering
Da-Yeh University
Changhua, Taiwan
kevin@mail.dyu.edu.tw

Chao-Ho Chen, Tsong-Yi Chen, Jian-He Wu
Department of Electronic Engineering
National Kaohsiung University of Applied Sciences
Kaohsiung, Taiwan
thouho@cc.kuas.edu.tw;
chentso@cc.kuas.edu.tw; U100B235@gmail.com

Chien-Chuan Ko
Department of Computer Science and Information Engineering
National Chiayi University
Chiayi, Taiwan
kocc@mail.ncyu.edu.tw

Abstract—**This paper presents a real-time face detection system using a moving camera. The proposed system consists of three modules, including (1) detection of face candidates: Face candidates are generated using the information of skin color, edges, and face area, (2) verification of face candidates: HOG (Histogram of Oriented Gradient) features are generated from face candidates and a two-class C-SVM (Support Vector Machine) classifier with pretrained face samples is employed to determine whether face candidates are real faces or not, (3) face tracking: Overlapping area of two face targets in current and previous frames is estimated to determine whether the tracking will be continuous or not. By use of estimation of face size, the proposed method can avoid a huge amount of computation time that is required by a point-by-point scanning way in conventional methods. Moreover, the accuracy of the face detection can be improved greatly. The proposed system can successfully detect most faces of the crowds in open space, which is beneficial for quickly searching the specified persons to prevent the occurrence of possible criminal events.**

Keywords-face detection; skin color extraction; histogram of oriented gradient; support vector machine; face tracking.

I. INTRODUCTION

Traditional methods for face detection are commonly based on fixed surveillance cameras, which often capture face targets with similar sizes. But, however, the captured face targets are of greatly varied sizes in the crowds for moving cameras. In recent years, computer vision has been widely used in various types of smart devices such as wearable cameras, in-car cameras, and robot vision systems. Therefore, face detection through mobile wearable cameras becomes a very important issue.

The sizes of faces, captured by mobile wearable cameras, dramatically depend on the distance with captured targets. Therefore, how to overcome the problem of multi-scale target detection becomes a challenging task. Image pyramid is often used to detect multi-scale targets, which can be realized by multiplying original image with a Gaussian function at different scales. In [1], they employed skin color features, AdaBoost classifier and image pyramid for multi-scale target detection. By this method, they searched each layer in the image pyramid to find the face targets with different scales. The method of image pyramid can achieve high detection accuracies in static images, but computational intense prohibits real-time operations of image videos.

In order to reduce the huge amount of computational loads mentioned above, Beeck et al. [2] used the feature of histogram of oriented gradient (HOG) [3] as well as cascade classifiers for detecting faces of pedestrians, which was then applied to an image video with 640×480 pixels at 15 fps (frames per second) to achieve an average processing rate of about 9.0 fps. But, when the same image video was applied to the method proposed by Felzenszwalb et al. [4], only a processing time of 1.17 seconds for each frame can be achieved, indicating the infeasibility of real-time operation. In order to meet the requirements of real-time operation, this paper proposes a scheme using face candidate areas on which face targets are searched. By this method, the processing time can be greatly decreased by reducing the search space and without the use of image pyramid.

The rest of this paper is organized as follows: Section II describes the detailed framework of the proposed method on face detection in a crowd of people. Experimental results are given in Section III. Finally, concluding remarks are provided in Section IV.

II. THE PROPOSED METHOD OF FACE DETECTION

The flowchart of the proposed method is shown in Fig. 1. The proposed scheme is composed of three modules, including detection of face candidates, verification of face candidates and tracking of face targets. In the module of detection of face candidates, the information of skin color, edge and face area are used together to generate face candidates. The face candidates are further verified by the module of verification of face candidates, which extracts the feature of HOG from face candidates and then applies them to the two-class classifier of C-SVM to complete the verification procedure. In this phase, some false positives can be further removed and face targets are thus generated. Finally, the face targets can be tracked using the method of

978-1-5386-5395-1/18/$31.00 ©2018 IEEE
DOI 10.1109/WAINA.2018.00153

template matching of targets. The details of the three modules will be described in the following sections.

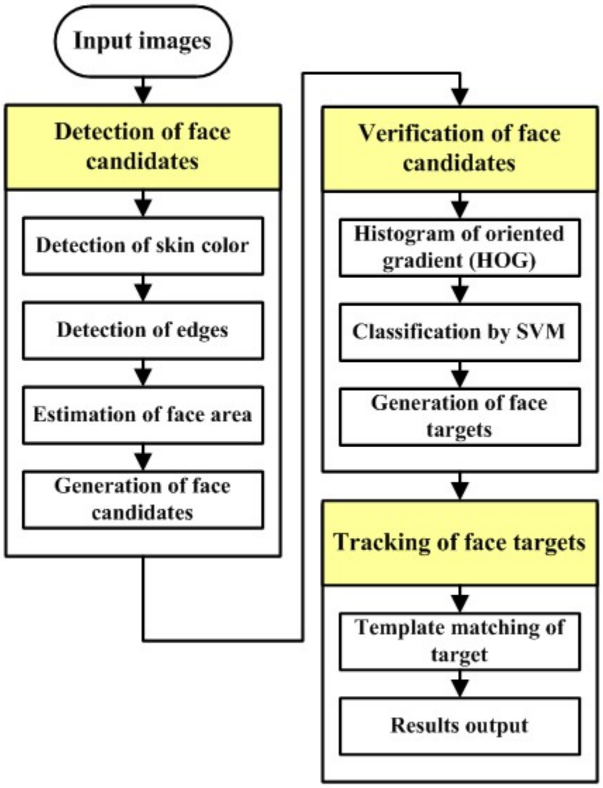

Input images
Detection of face candidates
Detection of skin color
Detection of edges
Estimation of face area
Generation of face candidates
Verification of face candidates
Histogram of oriented gradient (HOG)
Classification by SVM
Generation of face targets
Tracking of face targets
Template matching of target
Results output

Figure 1. Flowchart of the proposed method for face detection in a crowd of people.

2.1 Detection of face candidates

Traditionally, multi-scale face detection is carried out through the building of image pyramid. By this method, sliding window is used to search face targets at different scales of the layers in the image pyramid. This approach can work well in multi-scale target detection, but huge amounts of computation time are the primary disadvantage of this method. In this work, we propose an efficient algorithm that can effectively detect the hypothetical regions where face targets may appear. By our method, the computational complexity of multi-scale face detections in a crowd of people can be significantly reduced.

2.1.1 Detection of skin color

Color space conversion is used to eliminate the effect of illumination variation on the performance of face detection. For the RGB color space, the three components are of high correlation between channels; therefore, it is not a favorable choice for color analysis. In this paper, we utilize the skin color model using HSV color space [5] for detecting face regions. The formulas for extracting skin color are given as (1) and (2). If both conditions are satisfied, the point p(x, y) is labeled as a skin color pixel. Figure 2 shows the results of skin color detection in a crowd of people, where many skin color blobs are formed.

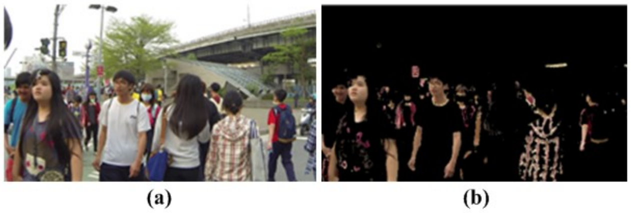


Figure 2. (a) Original image, and (b) results of skin color detection.

$$\begin{cases} H \le -0.4V + 75 \\ V \ge 40 \\ 10 \le S \le -H - 0.1V + 110 \end{cases} \tag{1}$$

$$\begin{cases} S \le 0.08(100 - V)H + 0.5V, \ H \ge 0 \\ S \le (0.5H + 35), \ H < 0 \end{cases} \tag{2}$$

2.1.2 Detection of edges

Sobel operators [6] aim to implement first derivatives in image processing using the magnitude of the gradient for detecting edges. The first derivative of an image can be expressed as $\nabla f = \left[G_x,\ G_y\right] = \left[\partial f / \partial x,\ \partial f / \partial y\right]$, and the magnitude as well as angle of the gradient can be formulized as $M(x, y) = \sqrt{G_x^2 + G_y^2}$ and $\theta(x, y) = \tan^{-1}\left(G_y(x, y)/G_x(x, y)\right)$, respectively. Therefore, horizontal and vertical edges can be effectively detected to form a binary image, i.e., $Edge_H(x, y)$ and $Edge_V(x, y)$, by the proposed formula (3), where θ_M, θ_H and θ_V are thresholds that are determined empirically.

$$\begin{gathered} \Omega = \left\{(x, y) \mid M(x, y) > \theta_M\right\} \\ Edge_H(x, y) = \begin{cases} 1, & \left(\theta(x, y) > \theta_H\right) \wedge \left((x, y) \in \Omega\right) \\ 0, & otherwise \end{cases} \\ Edge_V(x, y) = \begin{cases} 1, & \left(\theta(x, y) < \theta_V\right) \wedge \left((x, y) \in \Omega\right) \\ 0, & otherwise \end{cases} \end{gathered} \tag{3}$$

2.1.3 Estimation of face area

Estimation of face areas is required to generate resulting face candidates. In this work, the information of skin color is used to determine face area (i.e., both face width and height). First, the upper edge of each skin color blob is used to estimate the width of face candidate by which the outer enclosing rectangle of skin color blob can be further decided. Then, morphological operations are performed on these rectangular regions. By this method, a more complete skin color blob within the rectangular region can be formed, as shown in Fig. 3(a).

In Fig. 3(a), we can find many selected rectangular regions that are not human faces, which should be further filtered out. First, we filter out the rectangular regions where the width is too small to be a human face, i.e., $W_{SR} < W_{\min}$, where W_{SR} and $W_{\min}$ denote the width of selected rectangular regions and the threshold of minimum face width, respectively. Next, we calculate the aspect ratio for selected rectangular regions as $\rho = H_{SR} / W_{SR}$, where H_{SR} represents the height of selected rectangular regions, as shown in Fig. 4 (a). In this work, we consider that if two faces are too close together, they may be regarded as the same skin color region, so we only consider the height of the selected rectangular region as the criterion to filter out the rectangular regions that are not human faces.

According to the size information of the head skull, reported by the Ministry of Labor of Taiwan in 2016 [7], the average face width (i.e., distance between the ears) of the Chinese people is about 15.09 cm and the face height (i.e., head to chin) is about 24.28 cm. The skin color blobs may contain face and neck parts, therefore we consider that the face height should not exceed twice the width of the face, i.e., $\rho \le 2$. The result after applying the constraint $\rho \le 2$ is shown in Fig. 3 (b). By this method, the face area can be further determined using (4). Figure 4(b) shows the schematics of the face area of a human.

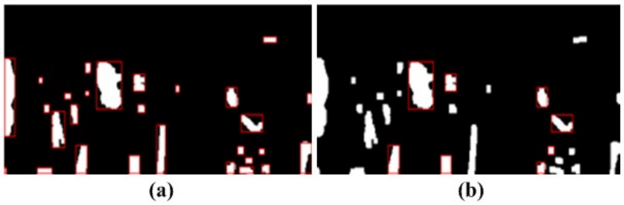


Figure 3. Detection of face candidate. (a) Result after morphlogical operation, and (b) results after applying the constraint of face height not exceeding twice the width.

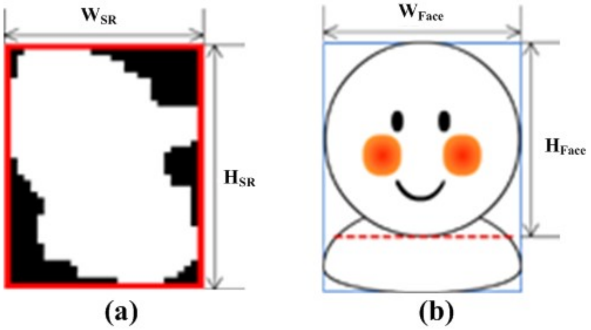


Figure 4. Schematics of (a) selected rectangular region, and (b) face region that contains no neck part.

$$\begin{aligned} W_{Face} &= W_{SR} \\ H_{Face} &= W_{SR} \times (24.28 / 15.09) \end{aligned} \quad (4)$$

2.1.4 Generation of face candidates

Some false positives still can be found in Fig. 3(b) and they should be further filtered out. To generate face candidates and filter out the false positives, three criteria are proposed based on the information of skin color, edge, and face area described earlier. The first criterion (see (6)) is used to determine whether the total number of horizontal edge pixels in the upper region (i.e., $NH_{upperEdge}$) that is shown as a red box (see Fig. 5(a)) is greater than half number of that box, where $NH_{upperEdge}$ is calculated from the integral image of the horizontal edge binary image of that red box, denoted as I_U^*, where $(x + W_{face} / 4,\ y)$ is the left-top corner of the red box, and $W_{face} / 2$ as well as $H_{face} / 8$ represent the width and height of the red box, respectively.

$$NH_{upperEdge} = Sum\left(I_U^*,\ x + \frac{W_{face}}{4},\ y,\ \frac{W_{face}}{2},\ \frac{H_{face}}{8}\right) \quad (5)$$

$$NH_{upperEdge} > (\frac{W_{face}}{2} \times \frac{H_{face}}{8}) \times 0.5 \quad (6)$$

The second criterion (see (9)) is used to determine whether the number of vertical edge pixels in the two outer regions (i.e., $NV_{outerEdge}$) is greater than that of the one inner region (i.e., $NV_{innerEdge}$). In (7) and (8), the integral images of I_{LL}^*, I_{LR}^* and I_{LM}^* are calculated from the vertical edge binary images of the lower left quarter region, lower right quarter region, and lower middle half region, respectively, as shown in Fig. 5(b). The parameters defined in the Sum function are similar with (5) described earlier.

$$\begin{aligned} NV_{outerEdge} = {} & Sum\left(I_{LL}^*,\ x,\ y + \frac{H_{face}}{2},\ \frac{W_{face}}{4},\ \frac{H_{face}}{2}\right) + \\ & Sum\left(I_{LR}^*,\ x + 3 \times \frac{W_{face}}{4},\ y + \frac{H_{face}}{2},\ \frac{W_{face}}{4},\ \frac{H_{face}}{2}\right) \end{aligned} \quad (7)$$

$$NV_{innerEdge} = Sum\left(I_{LM}^*,\ x + \frac{W_{face}}{4},\ y + \frac{H_{face}}{2},\ \frac{W_{face}}{2},\ \frac{H_{face}}{2}\right) \quad (8)$$

$$NV_{outerEdge} > NV_{innerEdge} \quad (9)$$

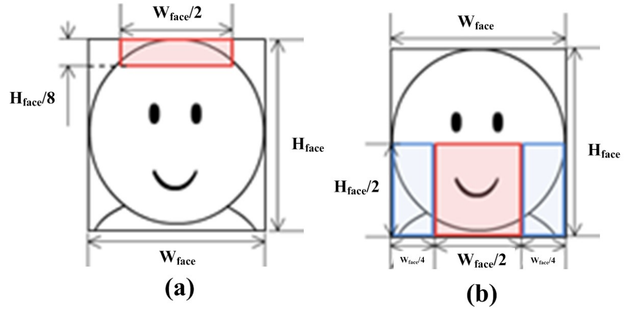


Figure 5. Schematics of (a) upper horizontal edge region, and (b) lower half vertical edge region with two outer and one inner regions.

The third criterion is evaluated using the information of skin color. As shown in Fig. 6, the region labeled as red box is used to calculate the number of skin color pixels. The third criterion (see (11)) is used to determine whether the number of skin color pixels (i.e., N_{skin}) is greater than one quarter area of that red box. In (10), the integral image of I^*_{skin} is calculated from the skin color image described in Section 2.1.1.

Base on the criteria of (6), (9) and (11), if all conditions are satisfied, the selected rectangular regions are detected as face candidates that will be further verified by the classifier of SVM using the HOG features that will be described in Section 2.2.

$$N_{skin} = Sum\left(I^*_{skin},\ x+\frac{W_{face}}{4},\ y+\frac{H_{face}}{4},\ \frac{W_{face}}{2},\ \frac{3\times H_{face}}{4}\right) \quad (10)$$

$$N_{skin} > \frac{W_{face}\times H_{face}}{4} \quad (11)$$

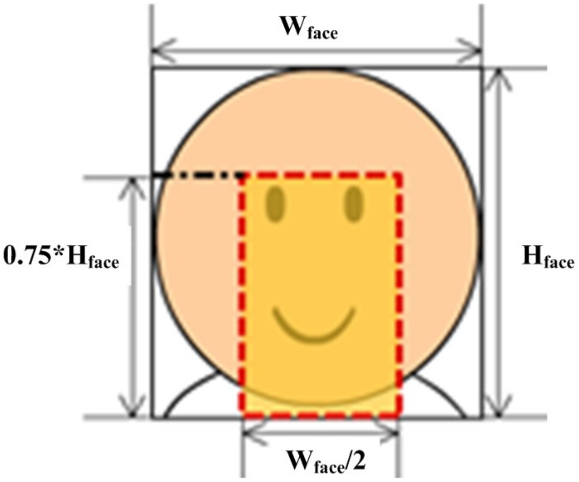


Figure 6. Schematics of skin color region of face with two outer and one inner regions.

2.2 Verification of face candidates

After face candidates in a crowd of people are detected, the face candidates will be further verified by a two-class SVM classifier using the HOG features. In this work, the size of face candidate is resized to 64×64 pixels and the HOG feature is used. To extract HOG feature, face candidate is first partitioned into 4×4 blocks, which is further split into 4 cells with a size of 8×8 pixels. Moreover, a sliding window with the same size of the block is applied to scan the face candidate region from top to bottom and left to right. The strides of the sliding window in both directions are set to 8 pixels and thus 49 blocks are generated. To estimate the gradient at pixel (x, y), both masks [-1 0 1] and $[-1\ 0\ 1]^T$ are used to extract the horizontal gradient (i.e., $G_x(x,y)$) and the vertical gradient (i.e., $G_y(x,y)$), respectively. Thus, the angle $\alpha(x,y)=\tan^{-1}(G_y(x,y)/G_x(x,y))$ for every pixel can be calculated. For each cell, nine intervals in both angle ranges of [0°, 180°] and [-180°, 0°] are equally partitioned. As a result, the dimension of the HOG feature vector is 49(blocks)×4(cells/block)×9(features/cell) = 1,764 for each face candidate.

For two-class (i.e., face and non-face) classification, the C-SVM model is employed to remove non-face candidates, where an optimal hyperplane can be derived as a decision surface as follows.

$$f(\mathbf{x})=\mathrm{sgn}\left[\sum_i y_i\alpha_i K(\mathbf{x_i},\mathbf{x})+b\right] \quad (12)$$

where $\mathbf{x}_i$, $i=1,\ldots,n$, is a training vector, $y_i\in\{+1,-1\}$ is a target label, n is the number of total training samples, $\mathrm{sgn}[\cdot]$ represents the sign function, and $K(\mathbf{x}_i,\mathbf{x})$ is a predefined kernel function. In this work, linear kernel $K(\mathbf{x_i},\mathbf{x})=\langle\mathbf{x_i},\mathbf{x}\rangle$ is adopted, where $\langle\cdot\rangle$ denotes the inner product of two vectors. The coefficients α_i and b in (12) can be determined from the dual form of the quadratic optimization problem as:

$$\max\left[\sum_i\alpha_i-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\mathbf{x_i},\mathbf{x_j})\right]\ s.t.\sum_i\alpha_i y_i=0 \quad (13)$$
$$0<\alpha<C,\ \forall i,$$

where α_i is a Lagrangian multiplier and the training sample is called support vector if $\alpha_i\neq 0$. The parameter C is a penalty that represents the tradeoff between minimizing the training set error and maximizing the margin. In the training phase, 8,754 positive images and 10,000 negative images (i.e., $n=18,754$), as shown in Fig. 7, are utilized to train the two-class C-SVM classifier. Therefore, after the verification of face candidates, face targets can be produced. In Section 2.3, the method of face target tracking will be described.

Figure 7. Typical training samples. (a) Positive images, and (b) negative images.

2.3 Tracking of face targets

Once face targets are determined, subsequent tracking for each face target will be carried out. The face targets i at current frame t is chosen to be matched with face targets j at its previous frame $t-1$ one-by-one using the metrics of Euclidean distance (see (14)) and the minimum distance of face target j is further selected as the best matching of face target i (see (15)), where (x_i^t, y_i^t) and (x_j^{t-1}, y_j^{t-1}) denote the centers of face targets i and j, respectively. As shown in Fig. 8, if face target j is located within the red circle, we consider face targets i and j being matched and the tracking record is updated accordingly, otherwise the matching is fail, which may be due to the tracking target that has left the search space.

$$D(i,j)=\sqrt{\left(x_i^t - x_j^{t-1}\right)^2 + \left(y_i^t - y_j^{t-1}\right)^2} \tag{14}$$

$$Matched(i,j) = \arg\min_j D(i,j) \tag{15}$$

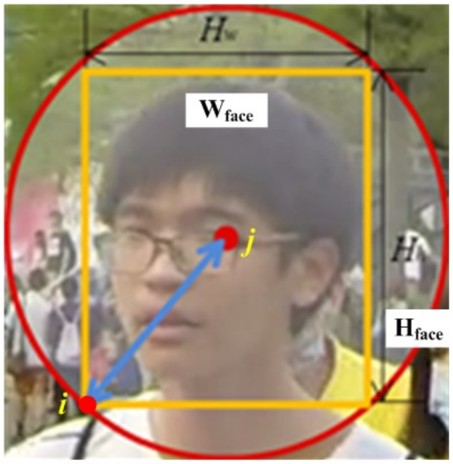
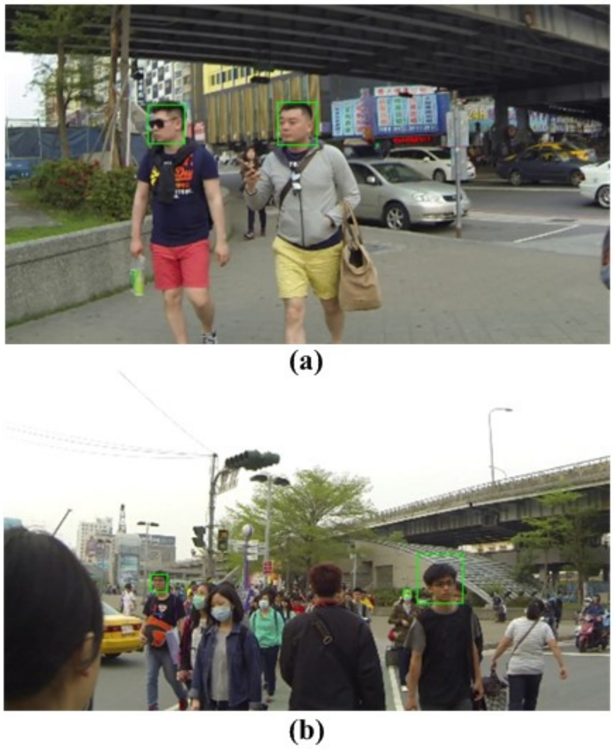


Figure 8. Searching range for face targets during tracking phase.

III. Experimental Results

The proposed method of pedestrian face detection is evaluated on real-world videos captured at public places. The integrated development environment is Microsoft Visual Studio 2013, and the programming language used is Visual C++ 2013 with Intel OpenCV 2.4.9 image processing library. The system is running on Intel i7-4770 processor with 16GB memory. The moving camera used is the GoPro Hero4 Black with a wide angle of 155°. The recording speed supports 4K motion pictures at 30 fps, while for 1080p motion pictures, it can support up to 120 fps.

As shown in Fig. 9, which are captured in the nearby of a train station, our proposed method can successfully detect most of multi-scale faces for a crowd of people in a clustered environment. As shown in this figure, some too small faces cannot be detected due to the limitation of our system with a detection resolution from 16×16 to 64×64 pixels. However, if human faces are too close our moving camera, the faces will not be detected by our system. For test videos, the average detection rate (=TP/(TP+FN)) of about 64.0% can be achieved with a false detection rate (=FP/(TP+FP)) of 13.6%, where TP, FP, and FN denote true positive, false positive, and false negative, respectively.



(a)

(b)

Figure 9. Results of face detection of a crowd of people in a clustered environment.

IV. Conclusion

In this paper, we have proposed a real-time face detection system using a moving camera in an open space. To overcome the challenge of multi-scale face detection using image pyramid, the proposed method restricts the search space of face targets in a limited area, which can be achieved using the information of skin color, edge, and face area estimation. By this method, face candidates are generated, which are further verified by the two-class C-SVM classifier with the HOG features to form face targets. The face targets are then matched and tracked using the metrics of Euclidean distance. Experimental results show our proposed method can successfully detect most of human faces, indicating the feasibility of the proposed method.

Acknowledgment

This research was partly supported by a grant from Ministry of Science and Technology, Taiwan, under contracts MOST 106-2221-E-212-013 and MOST 106-2221-E-151-061.

REFERENCES

[1] S. Ji, X. Lu, and Q. Xu, "A fast face detection method combining skin color feature and adaboost," in: Proc. of International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), pp. 1-5, 2014.

[2] V. B. Kristof, G. Toon, and T. Tinne, "Towards an automatic blind spot camera: robust real-time pedestrian tracking from a moving camera," in: Proc. of 12th IAPR Conference on Machine Vision Applications, Nara, Japan, 2011.

[3] N. Dalal, and B. Triggs, "Histograms of oriented gradients for human detection," in: Proc. of Computer Vision and Pattern Recognition, pp. 886-893, 2005.

[4] P. Felzenszwalb, R. Girschick, and D. McAllester, "Discriminatively trained deformable part models, release 4," in: http://people.cs.uchicago.edu/~pff/latent-release4/.

[5] G. H. Joblove and D. Greenberg, "Color spaces for computer graphics," ACM SIGGRAPH Computer Graphics, vol. 12, pp. 20-25, 1978.

[6] I. Sobel and G. Feldman, "A 3x3 isotropic gradient operator for image processing," *a talk at the Stanford Artificial Project,* pp. 271-272, 1968.

[7] Online available: https://data.gov.tw/dataset/40584.