A Face-Detection Postprocessing Scheme Using a Geometric Analysis for Multimedia Applications

Kyounghoon Jang, Hosang Cho, Chang-Wan Kim, and Bongsoon Kang

Abstract—**Human faces have been broadly studied in digital image and video processing fields. An appearance-based method, the adaptive boosting learning algorithm using integral image representations has been successfully employed for face detection, taking advantage of the feature extraction's low computational complexity. In this paper, we propose a face-detection postprocessing method that equalizes instantaneous facial regions in an efficient hardware architecture for use in real-time multimedia applications. The proposed system requires low hardware resources and exhibits robust performance in terms of the movements, zooming, and classification of faces. A series of experimental results obtained using video sequences collected under dynamic conditions are discussed.**

Index Terms—**Face detection, adaptive boosting algorithm, face-region stabilization, face recognition, single-port line memory**

I. INTRODUCTION

Human faces have been broadly studied in digital image and video processing fields. As one of the most detailed attributes in the human-computer interface, the face conveys a great deal of the information including behavioral cues, emotional state, identity, human race, age, and so on. Face-based algorithms thus have been used in a wide range of multimedia applications [1-3]. In particular, most digital imaging systems have integrated face-based auto-focusing functions to provide users with a convenient interface from the perspective of visual attention. In addition, the face-recognition systems require a correct face-detection scheme to extract facial features fed into pre-trained classifiers [4]. Detection and feature extraction are concurrently performed while searching human faces in digital images under dynamic visual deformation conditions such as position, scale, in-plane rotation, orientation, pose, and illumination. Depending upon the specific applications, robust face segmentation has been employed to solve several problems including the non-rigid shape of faces, clever alignment, and occlusion. In order to successfully achieve face detection in tiny mobile multimedia platforms, the hardware architecture must be computationally optimized in terms of memory usage and operation time.

An appearance-based method, the adaptive boosting (AdaBoost) learning algorithm has been widely accepted in face detection using integral image representations. It is ideal for real-time hardware systems such as digital cameras and mobile phones [5] and requires low computational complexity for feature extraction. The AdaBoost employs local binary pattern (LBP) images to avoid the effects of illumination and to express detailed textures. The pyramid representation is assigned for multifarious sizes of human faces. To segment face regions according to the in-plane rotations of the human face, pre-defined feature factors are used. Although AdaBoost exhibits acceptable performance, instantaneous face detection in the spatial domain caused by cascaded structures limits its performance. For example, the pyramid representation is used to compare

Manuscript received Apr. 30, 2012; revised Dec. 26, 2012.
Dept. of Electronic Engineering, Dong-A University, Busan, Korea.
E-mail: bongsoon@dau.ac.kr

the confidence factor between the reference and the candidate feature images to deal with an unknown number of various faces in the images. Only a single candidate feature that is close to the reference is chosen to determine the face region. While the reference is being retained, the mask window searching face region remains fixed across the pyramid feature images. Accordingly, both the trained reference and fixed size of the mask window yield instantaneous face detection in consecutive images due to the pyramid features' varying sizes. Therefore, the detected regions are unsteady, even though the real human faces are unmoving. This may degrade the auto-focusing function and disrupt visual perception. In particular, when applied to face recognition, the accuracy will be critically dropped due to heterogeneous face segmentation. In order to achieve robust face segmentation, sophisticated algorithms using facial landmark information have been studied to segment and align faces from images [6].

From the perspective of hardware implementation, high-resolution images and video processing require significant resources for detecting faces in real time according to hardware structures. [7] makes use of dual-port line memory for both the LBP image generator and the input image data. This scheme requires few additional registers to control individual line memory. A mapping reference table is integrated for the pyramid scaling process, requiring additional memory. [8] assigns two frame memories to handle the input image. This consumes huge memory resources. [9] allocates dual-port line memory to segment the facial images. However, the integral image generation used in face segmentation takes additional computation time.

In this paper, we propose a computationally efficient face-detection algorithm that is not influenced by external environments and also stabilizes the variability in face detection. Furthermore, it performs in the spatial and temporal domains. More specifically, the developed algorithm equalizes varying geometric positions of the face regions acquired from AdaBoost. In order to address the problem of unsteady face detection of AdaBoost, digital filters are selectively applied in the spatial domain. By doing so, the noisy movements of the regions containing fixed human faces are coarsely filtered out. The geometrical differences between the filtered/ unfiltered facial region and the region of the previous image frame are analyzed in terms of vertices, areas, and locations. Facial marginalization is carried out to refine the facial region. Our design is aimed at optimizing hardware processing in terms of memory usage and the computation time. We use single-port memory and compute two bytes in a parallel fashion. One single-port memory and one frame memory are used to generate the LBP image. An additional single-port memory is used to segment the facial regions. Since the line memory processes two bytes in parallel, the number of read/write clocks and control signals is reduced when compared to the dual-port line memory. This method can help to reduce the chip area.

In particular, the proposed algorithm increases end-user preference in digital imaging systems. The stabilized face detection enhances the seamless visual attention of human faces during the zooming in/out process. We show improved face detection results in a series of experiments. Furthermore, the developed system prevents accuracy loss caused by AdaBoost when applied to face recognition systems. Steady face segmentation must be secured as much the test inputs as for the reference, because the recognition systems' performance critically hinges on the data-correlation pattern between the trained reference and the test inputs from a qualitative perspective.

This paper is organized as follows. Section II presents the proposed face detection system that segments and refines the facial region with hardware architecture. The stabilization of face detection is described focusing on a geometric analysis for use in streaming videos. Section III presents simulation results that show the performance of face detection and improved accuracy of face recognition. Section IV compares the hardware complexity and the operation time. Section V provides concluding remarks.

II. ROBUST FACE-DETECTION SCHEME

Fig. 1 illustrates a block diagram of the proposed face-detection system. The proposed hardware system consists of four consecutive steps: gray image conversion, preprocessing, face segmentation, and postprocessing. These steps are described in more detail in the following sections.

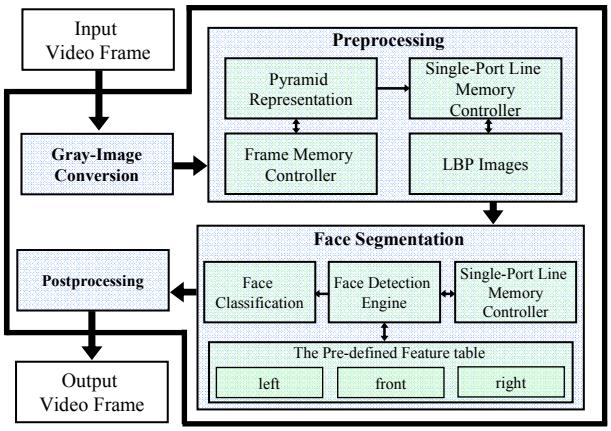


Fig. 1. Block diagram of the proposed face-detection system.

1. Gray-Image Conversion

The first step is gray-image conversion. A 24-bit RGB image is averaged to generate an 8-bit gray image Y as follows:

$$Y = \frac{(R+G+B)}{3} \tag{1}$$

where R is red, G is green, and B is blue. The converted image is then stored in the frame memory for preprocessing.

2. Preprocessing

The converted image is then preprocessed via the frame memory controller. A mask window of (3×3) is used to generate the LBP image in (2) with (3) using the pyramid representation in [7]:

$$LBP(x_m, y_m) = \sum_{n=0}^{7} 2^n f(g_n - g_m) \tag{2}$$

$$f(g) = \begin{cases} 1, if\ \ g \geq 0 \\ 0, if\ \ g < 0 \end{cases} \tag{3}$$

where x_m and y_m are pixel locations, g_n surrounds eight pixels in the window excluding the center pixel of g_m, and $f(g)$ is the binary LBP image. Fig. 2 illustrates the single-port line memory controller having a half image width for the memory address depth, two bytes for the memory width, and a flip-flop (FF) for data delay. A sequence of 8-bit input/output data is processed for 16-bit operation for the memory using two FFs with each individual clock. In the first clock, the odd incoming byte reaches the left FF while the even outcoming byte reaches the right FF. In the second clock, the even and odd incoming bytes are simultaneously loaded at the single-port line memory while the even outcoming byte at the right FF is released. Therefore, no clock latency is used to perform the simultaneous reading/writing for the line memory. This architecture helps to reduce hardware complexity.

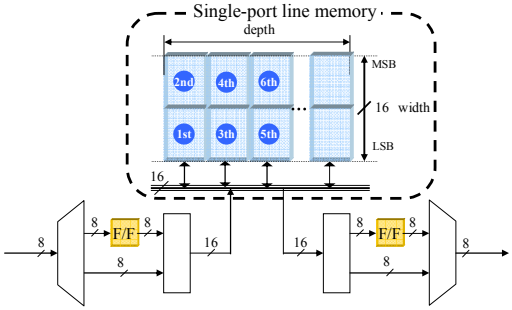


Fig. 2. Used line-memory architecture.

3. Face Segmentation

A mask window of (22×22) is used to detect the facial region on the LBP image in the form of the cascaded structure in [8]. The confidence factors obtained from facial features are then compared to the pre-defined feature table. Since the size and the rotation of the human face are unpredictable, the confidence factors must be satisfied at each cascade stage. Among the different face regions obtained through the pyramid representation, the region containing the highest factor is chosen for segmentation.

4. Postprocessing

Fig. 3 is a flowchart of the postprocessing. Given the segmented facial region of F_i at the ith frame obtained from AdaBoost, digital filtering is performed as follows:

$$F_i = \{x_i, y_i, w_i, h_i\} \tag{4}$$

$$\hat{F}_i = \begin{cases} \alpha F_i + (1-\alpha) F_{i-1}, & if\ \ S_i \cap S_{i-1} \geq \dfrac{S_{i-1}}{4} \\ F_i, & otherwise \end{cases} \tag{5}$$

where x_i and y_i represent the left-top locations of the

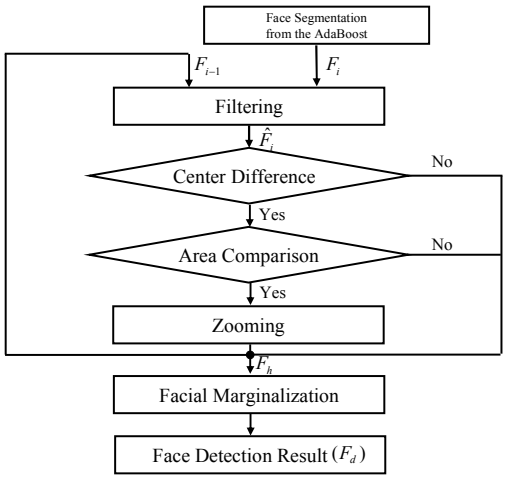


Fig. 3. Flowchart of the postprocessing.

facial region, w_i and h_i are the width and height for the region, α is the adaptive rate, S_i is the facial area of F_i, and $\cap$ denotes the overlapped area. The regions between the consecutive image frames shake when the overlapped face area is greater than previous area divided by four. Hence, the unstable regions obtained from AdaBoost need to be lowpass filtered in the temporal domain to retain stable facial regions. α can be changed to fall within a range of 0 to 1.

The geometric properties of consecutive facial regions are analyzed to stabilize the region. For this, the center difference is measured to compare the movement of consecutive regions. The center difference between $\hat{F}_i$ and F_{i-1} can be calculated as follows:

$$D_{center} = \left| C_{\hat{F}_i} - C_{F_{i-1}} \right| \tag{6}$$

where $C_{\hat{F}_i}$ is the center point of $\hat{F}_i$ and $C_{F_{i-1}}$ is the center point of F_{i-1}. We assume that there is a large movement in the facial regions when D_{center} is greater than a threshold value of Th_{center}. We then choose $\hat{F}_i$ as the candidate region (F_h) for facial marginalization. Otherwise, their regions are used to determine whether the previous region is either fixed or zoomed in/out in the current facial region:

$$F_h = \begin{cases} \hat{F}_i & if\ D_{center} > Th_{center}\ (movement) \\ F_{i-1} & if\ \Delta_{area} \geq Th_{area}\quad (fixed) \\ Zw_{i-1} + F_{i-1} & otherwise\qquad (zoom) \end{cases} \tag{7}$$

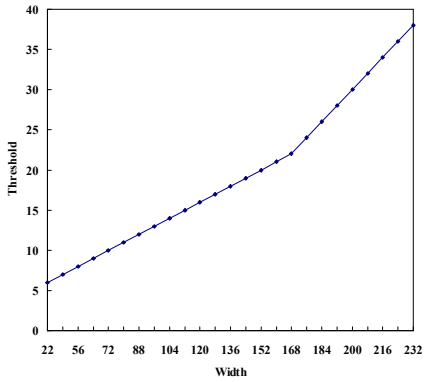


Fig. 4. Threshold value for the center difference.

where Z is the zooming parameter of {-1/10, -1/10, 1/5, 1/5}, and w_{i-1} is the width of F_{i-1}. As the facial region increases, Th_{center} must be increased according to the increasing difference of D_{center}. We compute Th_{center} proportional to w_i based on a 320×240 input image, as shown in Fig. 4. We consider the consecutive facial regions to be zoomed in/out when Δ_{area} is smaller than Th_{area}. In this case, the areas of $\hat{S}_i$ and S_{i-1} are compared to derive Δ_{area} as follows:

$$\Delta_{area}(\%) = \frac{\hat{S}_i \cap S_{i-1}}{I} \times 100,\ I = \begin{cases} \hat{S}_i & if\ \hat{S}_i \geq S_{i-1} \\ S_{i-1} & otherwise \end{cases} \tag{8}$$

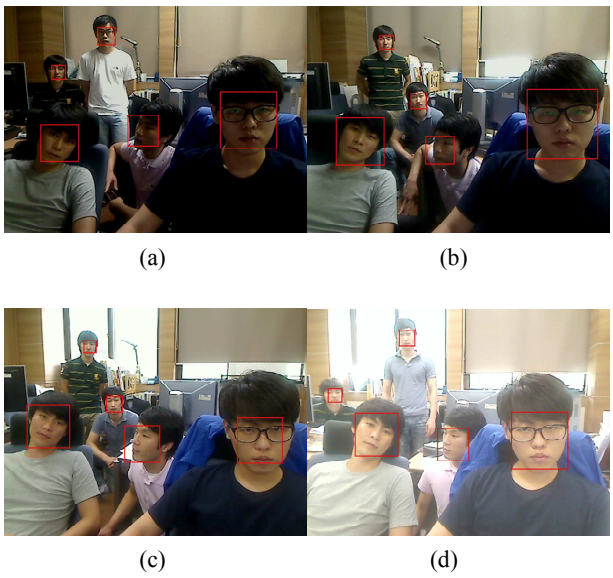
where $\hat{S}_i$ is the area of $\hat{F}_i$ and I is the larger area of the two areas. We now move to the facial marginalization step. F_h is used for the facial marginalization as follows:

$$F_d = Mw_h + F_h \tag{9}$$

where F_d is the detected facial region. w_h is the width of F_h, and M is the marginal parameter of {-1/16, -1/16, 1/8, 1/8}.

III. SIMULATION RESULTS

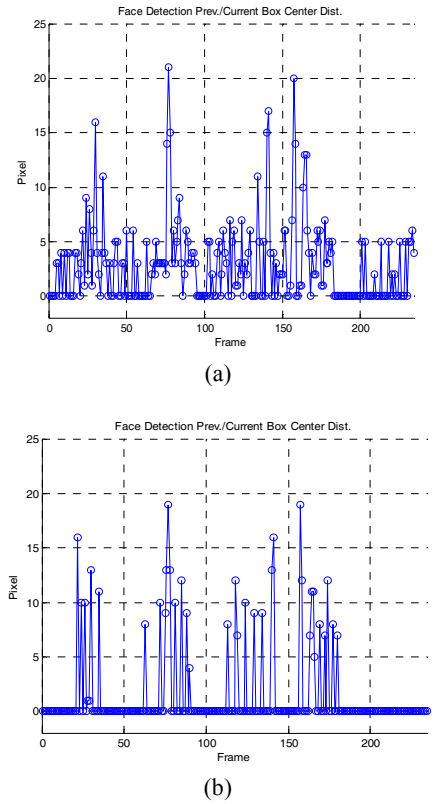
The proposed system was evaluated using a series of video sequences recorded with a fixed camera (320×240, 30 frames per second (fps)) under dynamic conditions. Fig. 5 shows the performance of face detection in different luminance levels. Fig. 5(a) and (b) depict a dim condition with average luminance of 62.3 square meters per candela (cd/m^2) and 74.8 cd/m^2, respectively. Fig. 5(c) is a normal condition with 105.9 cd/m^2 average



(a) (b)

(c) (d)

Fig. 5. Performance comparison of face detection at different luminance levels (a) dim condition (62.3 cd/m^2), (b) dim condition (74.8 cd/m^2), (c) ordinary condition (105.9 cd/m^2), (d) bright condition (176.1 cd/m^2).



(a)

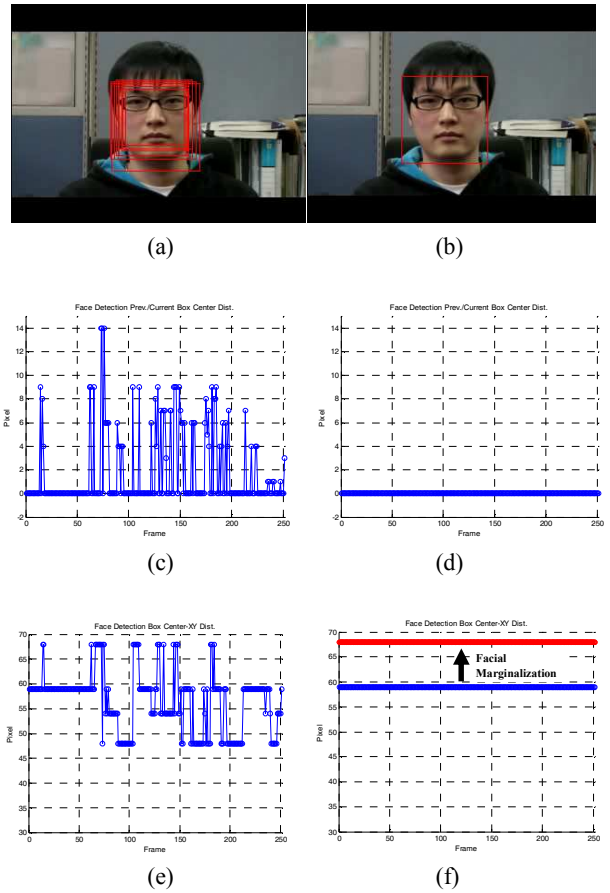
(b)

Fig. 6. Performance comparison of face movements (a) AdaBoost, (b) the proposed system.

luminance and Fig. 5(d) shows a bright condition with 176.1 cd/m^2. We see that the proposed system effectively detected the human facial region regardless of the lighting conditions.

Fig. 6 shows the level of face detection with a naturally moving face. Fig. 6(a) shows D_{center} of the video sequence from AdaBoost. It can be seen that the detected face regions are unstable below a pixel difference of 5. The peaks with a pixel difference of more than 5 indicate the natural movements of the face. Fig. 6(b) depicts the results obtained from the proposed system. When compared to Fig. 6(a), most of the unstable movements are filtered out effectively while the natural movements remain.

Fig. 7(a) shows the facial regions detected by AdaBoost, marked in red. Although the volunteer's face is not moving, the marked boxes shake in consecutive frames in the temporal domain. Fig. 7(c) and 7(e) depict D_{center} and the pixel length from the center to the left top of the facial region, respectively. Fig. 7(b) shows the

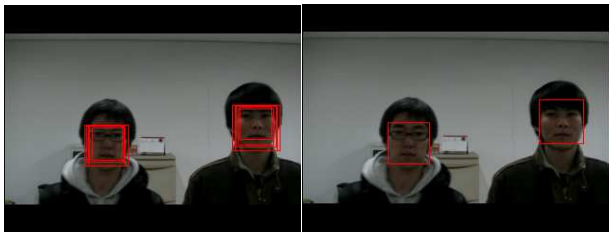


(a) (b)

(c) (d)

(e) (f)

Fig. 7. Performance comparison of one-face detection (a) AdaBoost, (b) the proposed system, (c) the center difference of (a), (d) the center difference of (b), (e) the pixel length from the center to the left top of (a), and (f) the pixel length from the center to the left top of (b).



(a) (b)

Fig. 8. Performance comparison of two-face detection (a) AdaBoost, (b) the proposed system.

facial region detected by the proposed system. The marked region consists of a single red box containing both eyebrows and the jawline in a stabilized manner from the marginalization step. Fig. 7(d) and 7(f) depict its D_{center} and its pixel length, respectively. Since the human face is not moving, all the peaks in Fig. 7(c) should be removed, as shown in Fig. 7(d). The fluctuating pixel length in Fig. 7(e) should also be fixed at a pixel difference of 59, as shown in Fig. 7(f). The length is then moved to 68 by facial marginalization.

Fig. 8 shows a performance comparison with two faces in consecutive images. Once again, we see that the regions in Fig. 8(b) are stabilized by the proposed system.

Fig. 9 shows the a performance comparison of zooming of the AdaBoost and the proposed system. Video sequences are recorded with zoomed zooming-in from 1 to 130 frames, fixed from 131 to 189 frames, and zoomed zooming-out from 190 to 298 frames. The facial regions marked in green are detected during zooming, and the regions marked in red are detected in the last image frame. As depicted in Fig. 9(a) and 9(c), the detected facial regions are noncentrically increased and decreased due to unstable face detections. It is shown that the proposed system helps the AdaBoost to detect face regions centrically, as shown in Fig. 9(b) and 9(d). We see that the proposed system effectively removes the noisy movements, as shown in Fig. 9(f) and 9(h), when compared to Fig. 9(e) and 9(g). Fig. 9 demonstrates that the proposed system can stabilize face detection during zooming as well.

Fig. 10 shows the reference faces for use in face-recognition experiments. We employed a principle component analysis (PCA) in [10] to perform face recognition. The PCA involves weighted comparisons of the eigenface features to recognize individual faces. Twenty people volunteered to prepare the reference and

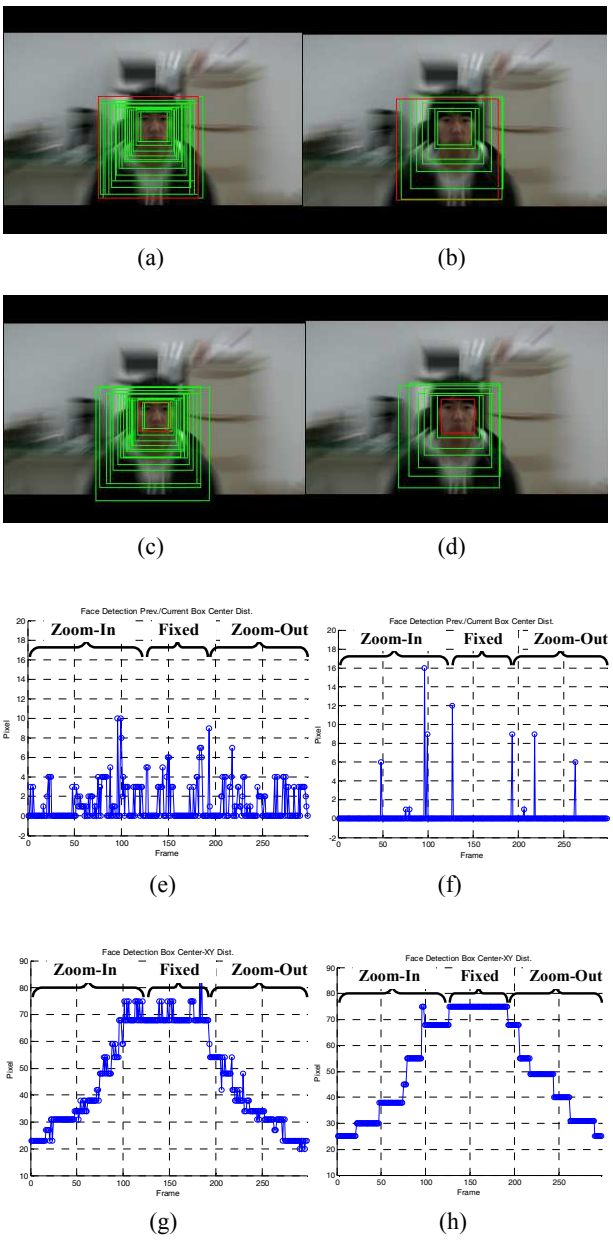


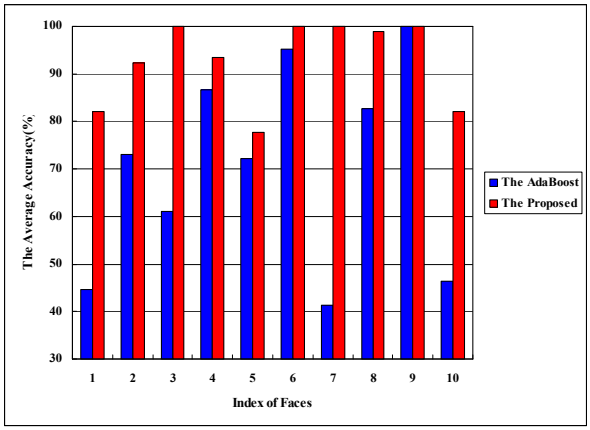
Fig. 9. Performance comparison of zooming (a) zoom-in of AdaBoost, (b) zoom-in of the proposed system, (c) zoom-out of AdaBoost, (d) zoom-out of the proposed system, (e) center difference of the AdaBoost, (f) center difference of the proposed system, (g) pixel length from the center to the left-top of the AdaBoost, (h) pixel length from the center to the left-top of the proposed system.

the test video sequences. Ten people in the first and second rows were moving naturally during the recording. The remaining ten people did not move. Both AdaBoost and the proposed system were tested to segment the facial regions. The classification performance was evaluated and compared by the Leave-N-Out procedure in [11]. We selected the best face detections obtained

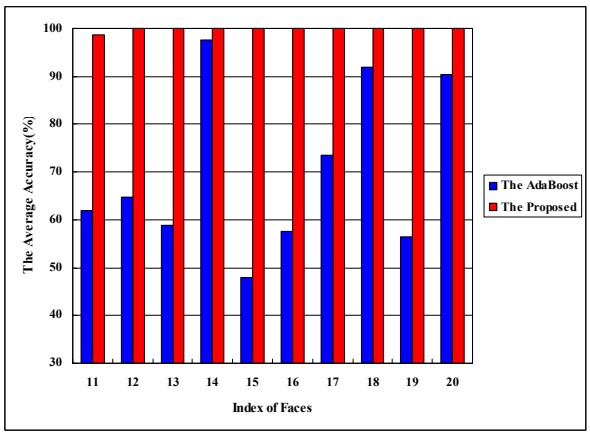
(a)

(b)

Fig. 10. Reference faces for face recognition (a) AdaBoost, (b) the proposed system.



(a)



(b)

Fig. 11. Accuracy of face recognition (a) the naturally moving people, (b) the unmoving people.

from AdaBoost and the proposed system as the reference face images.

Fig. 11 shows the accuracy of face recognition. First, Fig. 11(a) shows the accuracy for naturally moving people. An average accuracy of 92.7% was obtained by the proposed system, whereas an average of 70.3% was obtained by the AdaBoost. The proposed system thus improved recognition by about 22%. Fig. 11(b) shows the accuracy for unmoving people. An average accuracy of 99.9% was obtained by the proposed system, whereas an average accuracy of only 70.1% was obtained by AdaBoost alone. Once again, the proposed system was about 30% more accurate. The simulation results show that the proposed system achieves an average of 26% accuracy improvement in face recognition. Thus, the proposed system improves the accuracy of face recognition when combined with AdaBoost.

Table 1. Comparison of hardware complexity

	The proposed system	[7]	[8]	[9] Single classifier	[9] Triple classifier
Slice registers	23,271	75,766	–	19,066	21,163
Slice LUTs	66,156	135,041	48,600	64,143	79,537
BRAMs	37	285	–	41	41
Memory usage (Kb)	1,296	–	9,504	–	–

IV. HARDWARE COMPLEXITY AND OPERATION TIME

The proposed face-detection system was designed using Verilog and implemented in Xilinx Virtex-5 LX330 FPGA to verify the operation.

Table 1 provides a comparison of the proposed and existing systems' hardware complexity. The slice

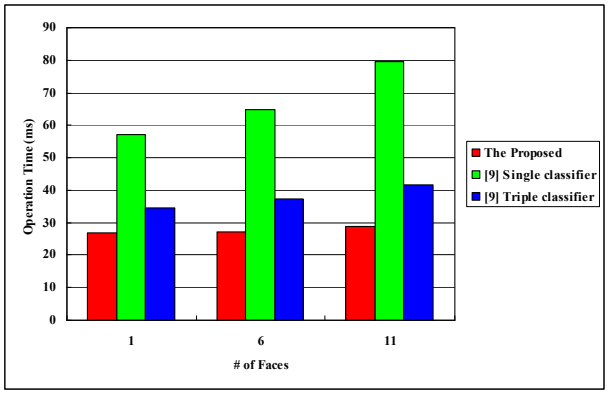


Fig. 12. Comparison of operation times.

registers and slice look-up-tables (LUTs) represent the hardware area. The block RAMs (BRAMs) and the memory indicate the memory usage. The proposed system requires 23,271 slice registers, 66,156 LUTs, 37 BRAMs, and 1,296 kilo-bit (Kb) of memory usage. The maximum operating frequency is 120 MHz. [7] requires 75,766 slice registers, 135,041 LUTs, and 285 BRAMs. [8] requires 48,600 LUTs and 9,504 Kb of memory usage. [9] with a single classifier requires 19,066 slice registers, 64,143 LUTs, and 41 BRAMs. [9] with a triple classifier requires 21,163 slice registers, 79,537 LUTs, and 41 BRAMs. From the table, it is clear that the proposed system requires the least amount of hardware resources to achieve the same level of overall complexity.

Fig. 12 shows the operation times in the hardware. Given the input image (320×240 at 60 fps), different numbers of human faces with (1, 6, 11) were tested to measure the time. The proposed system requires 26.86 milliseconds (ms) for detecting one face, 27.29 ms for six faces, and 28.80 ms for eleven faces, respectively. The average time is 27.65 ms, which is equal to 36.17 fps. [9] with a single classifier takes a considerable amount of operation time as the number of human faces increases. The measured frame rate is as low as 15.14 fps. [9] with a triple classifier decreases the rate to 26.51 fps. Once again, it is demonstrated that the proposed system provides the best performance in terms of hardware complexity and operation time.

V. CONCLUSIONS

A face-detection postprocessing method that makes use of single-port line memory was presented. The line memory developed in this paper efficiently minimizes clock cycles and cell areas in a parallel mode. Stabilization of facial regions was achieved using a geometric analysis. The unsteady facial regions obtained from the AdaBoost were processed further for stabilization using the proposed method. The stabilized region made it possible to perform robust face detection in instantaneous facial regions. When fed into the PCA for face recognition, recognition accuracy increased by about 22% for a sequence of naturally moving people and by about 30% for unmoving people. The required hardware resources were also highly efficient when compared to other systems in terms of memory usage and operation time. Therefore, we see that the proposed method facilitates accurate face detection when combined with the conventional AdaBoost.

ACKNOWLEDGMENTS

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0004551).

REFERENCES

[1] K.H. An and M. Chung, “Cognitive face analysis system for future interactive TV,” *IEEE Trans. on Consumer Electronics*, Vol.55, No.4, pp.2271-2279, Nov., 2009.

[2] H.C. Lee, D.T. Luong, C.W. Cho, E.C. Lee, and K.R. Park, “Gaze tracking system at a distance for controlling IPTV,” *IEEE Trans. on Consumer Electronics*, Vol.56, No.4, pp.2577-2583, Nov., 2010.

[3] D.J. Kim, K.W. Chung, and K.S. Hong, “Person authentication using face, teeth, and voice modalities for mobile device security,” *IEEE Trans. on Consumer Electronics*, Vol.56, No.4, pp.2678-2685, Nov., 2010.

[4] R. Atta and M. Ghanbari, “Low-memory requirement and efficient face recognition system based on DCT pyramid,” *IEEE Trans. on Consumer Electronics*, Vol.56, No.3, pp.1542-1548, Aug., 2010.

[5] B. Froba and A. Ernst, “Face detection with the

modified census transform," *in Proc. 6th IEEE Int. Conf. Automatic Face and Gesture Recognition*, pp.91-96, May, 2004.

[6] Y.S. Huang, T.C. Hsu, and F.H. Cheng, "Facial landmark detection by combining object detection and active shape model," *in the 3rd Intl. Symp. Electronic Commerce and Security (ISECS)*, pp.381-386, July, 2010.

[7] S. Jin, D. Kim, T.T. Nguyen, B. Jun, D. Kim, and J.W. Jeon, "An FPGA-based parallel hardware architecture for real-time face detection using a face certainty Map," *IEEE Int. Conf. ASAP*, pp.61-66, July, 2009.

[8] D. Han and J. Choi, "High performance real-time face detection architecture for HCI applications," *International Symposium on Ubiquitous Virtual Reality*, pp.48 – 51, July, 2010.

[9] J. Cho, S. Mirzaei, J. Oberg, and R. Kastner, "FPGA-based face detection system using haar classifiers," *ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp.103-112, 2009.

[10] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, Vol.3, No.1, pp.71-86, 1991.

[11] B.D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, 1996.

Kyounghoon Jang received a B.S in Electronic Engineering from Dong-A University, Busan, Korea in 2010. He is currently working toward his Ph.D. degree at the university. His research interests include digital processing systems, image/ video processing, and VLSI architecture design.

Hosang Cho received a B.S in Electronic Engineering from Dong-A University, Busan, Korea in 2010. He is currently working toward his Ph.D. degree at the university. His research interests include digital signal processing, VLSI architecture design, and image processing.

Chang-Wan Kim received a B.S. degree from the School of Electrical Engineering and Computer Science, Kyungpook National University, Korea, in 1997, and M.S. and Ph.D. degrees in Engineering from the Information and Communications University (ICU), Daejeon, Korea in 2003 and 2006, respectively. From 2006 to 2007, he worked for Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea. Since 2007, he has been with the Department of Electronic Engineering, Dong-A University, Busan, Korea, where he is now an assistant professor. His main research interests are UWB RF transceiver design and system-level integration of transceivers.

Bongsoon Kang received a B.S. in Electronic Engineering from Yonsei University, Korea in 1985, and an M.S. degree in Electrical Engineering from the University of Pennsylvania in 1987, and a Ph.D. in Electrical and Computer Engineering from Drexel University in 1990. From Dec. 1989 to Feb. 1999, he worked as a senior staff researcher at Samsung Electronics Co. Ltd., Korea. Since March 1999 he has been with the Department of Electronics Engineering, Dong-A University, Busan, Korea. He is currently a professor of the department. His research interests include image processing, hardware architecture designs, and wireless communications.