Development of Real-Time Face Detection Architecture for Household Robot Applications

Dongil Han, Hyunjong Cho, Jaekwang Song, Hyeon-Joon Moon,
and Seong Joon Yoo

Department of Computer Engineering, Sejong University
98 Gunja, Kwangjin, Seoul, Korea
{dihan,hmoon,sjyoo}@sejong.ac.kr, Hyeonjong@live.co.kr,
song0814@hotmail.com

Abstract. This paper describes the structure of real-time face detection hardware architecture for household robot applications. The proposed architecture is robust against illumination changes and operates at no less than 60 frames per second. It uses Modified Census Transform to obtain face characteristics robust against illumination changes. And the AdaBoost algorithm is adopted to learn and generate the characteristics of the face data, and finally detected the face using this data. This paper describes the hardware structure composed of Memory Interface, Image Scaler, MCT Generator, Candidate Detector, Confidence Mapper, Position Resizer, Data Grouper, and Overlay Processor, and then verified it using Virtex5 LX330 FPGA of Xilinx. Verification using the images from a camera showed that maximum 16 faces can be detected at the speed of maximum 30.

Keywords: multiple face detection, MCT (Modified Census Transform), real-time FPGA implementation, hardware design.

1 Introduction

With the development of biometric technologies using the information of human bodies, conventional security methods using keys and keypads are being replaced by identity certification methods using face, iris, fingerprint, retina, etc. The face detection method, among them, is currently used with other biometric methods for cases where enhanced security is required, such as some companies, banks, and governmental agencies, due to its low cost and convenience resulting from contactless operation, despite its drawback that relatively large variability is caused by plastic surgery, changes in facial expression, etc. [1] The method is gradually expanding its scope of application as users are certified by the face for personal laptops. Reliable face recognition rate of the method necessarily requires the face detection technology that can accurately extract faces from images.

Face detection algorithms have so far been developed mainly for PC-based environments. The technologies could not efficiently detect a face, however, when applied in an embedded system due to the relatively insufficient resources and performance of the system. The need is far more increasing, however, for high-performance real-time

C. Stephanidis (Ed.): Universal Access in HCI, Part II, HCII 2009, LNCS 5615, pp. 57–66, 2009.

face detection technologies for embedded systems as human face information becomes useful in more fields for mobile devices such as cellular phones and digital cameras and as the household robot market, covering cleaning, entertainment, toys, etc.

It is known that the face detection performance is most affected by illumination changes, face rotation, and facial expression changes. Illumination changes, among them, can result not from the user's intention but from time, lighting, and the changes in the indoor and outdoor environment though face rotation and expression changes can be controlled in their influence if the user is cooperative with the face detecting system. It is still a challenging task, therefore, to clearly detect a face regardless of illumination changes.

This paper extracted only the structural information of an object and transformed the images by MCT(Modified Census Transform), which reduces the influence of illumination changes, to reduce the cost required for illumination compensation, and then presented a hardware structure that instantly detects a face using thus transformed images against illumination changes in an embedded system. It also verified the performance by implementing the structure with Virtex5 LX330 FPGA and testing it in the daily life environment with various illumination changes.

This paper consists of five chapters. Chapter 2 describes MCT used in this paper and AdaBoost that generates learned data using transformed images. Chapter 3 describes the details of the whole hardware structure proposed in this paper. Chapter 4 analyzed the results of operating the structure in the PC environment and of testing it in the daily life environment after implementing it by FPGA. Chapter 5 finally discusses the conclusion and the future research plan.

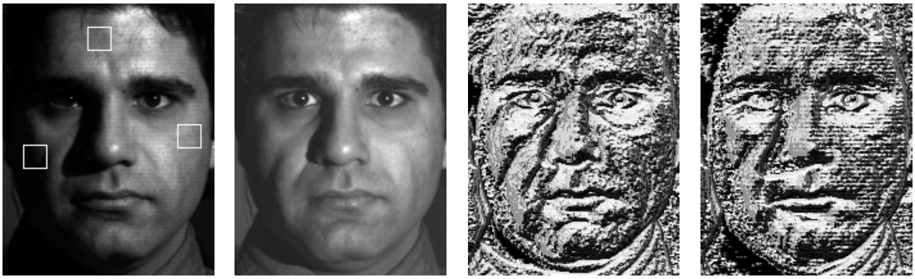
2 MCT and AdaBoost Learning Algorithm

This chapter describes MCT that extracts the characteristics required for recognizing a face in an image while minimizing the cost required for illumination compensation and the AdaBoost learning algorithm that effectively processes obtained face data characteristics. This chapter also describes the performance and speed change of the single-layer structure used, instead of the cascade structure, to minimize the memory space used to save learned data in an embedded system.

2.1 MCT

Ideal operation of the face detection or recognition system requires to extract and use only the structural characteristics of a face from images. MCT [2] expresses the structural information of the window with the binary pattern {0, 1} moving the 3x3 window in an image as small as to be able to assume that illumination of small region is almost constant, though the value is actually variable, where the pattern contains the information on the edges, contours, intersections, etc.

Thus connected patterns are transformed into decimal numbers and then the values of the pixels in MCT-transformed images. The figure 1 shows MCT transformed images in different illumination level.



a. Test image 1 b. Test image 2 c. MCT result of Test image 1 d. MCT result of Test image 2

Fig. 1. Gray level and MCT images in different illumination level

2.2 AdaBoost Learning Algorithm

Viola and Jones [3] extracted characteristics to effectively identify a face using the AdaBoost learning algorithm proposed by Freund and Schapire [4], and composed them in a cascade structure of 38 phases. Fröba and Ernst also introduced a face detector composed of a 4-phase cascade structure using MCT-transformed images and AdaBoost learning algorithm [1]. Such a cascade structure removes an image of the parts that is clearly out of a face in the early phase using a few characteristics with high discrimination capability, and uses more characteristics with relatively weak discrimination capability to concentrate on screening the parts difficult to identify. The 4-phase cascade structure proposed by Fröba and Ernst highly raised the speed by passing all the facial parts and removing at least 99% of the non-facial parts in the first phase.

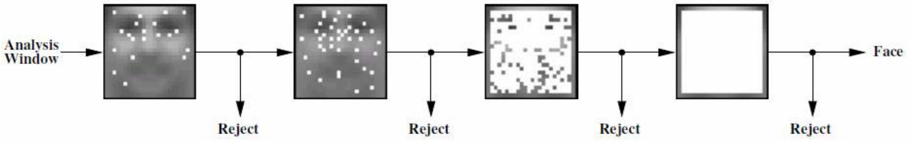


Fig. 2. The four classifiers of Cascade Detector [1] (the white dots mark positions of elementary classifiers in the analysis window)

The cascade structure thus needs to save, in the form of a lookup table, the location information for required characteristics and the face reliability values corresponding to the characteristic values. The memory is usually not enough to store all the information in an embedded system where the resource is limited compared with the PC environment. The hardware is advantageous, however, in that it is faster than the software operation because it supports multi-tasking.

This paper, therefore, composed the face detector with a single-layer structure using only the last fourth phase of the cascade structure proposed by Fröba and Ernst. The test result shows that the single-layer structure can also effectively and instantly detect a face by processing 30 frames per second. The details of the detection performance are discussed in Chapter 4.

3 Proposed Hardware Structure

Figure 3 shows the whole hardware structure proposed by this paper. Memory Interface block controls the streams of the image data and used to save images. Image Scaler down-scales images by phases, MCT Generator transforms images by MCT, and Candidate Detector delivers to Confidence Mapper the MCT values for the 400 pixels of the 20x20 area currently processed in the window. Confidence Mapper determines the reliability value of the possibility that a pixel is out of the facial area and cumulatively sums the 400 values, using the reliability values learned offline in advance and saved in the ROM table. Thus summed total reliability value is compared with the preset threshold and used to find candidate facial areas. Position Resizer calculates the location, in the original 320x240 image of the candidate facial area identified by Candidate Detector and Confidence Mapper. Data Grouper finds and removes non-facial background area out of the facial areas identified by Confidence Mapper and Position Resizer. Finally, Overlay Processor covers, with rectangular shape such as □, the location in the original image identified as a facial area and outputs the image on the display.

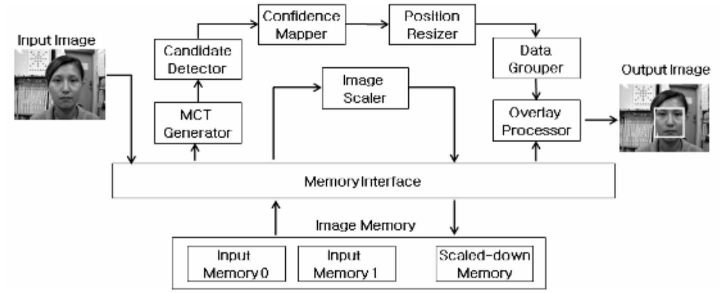


Fig. 3. Proposed Face Detection Hardware Structure

3.1 Memory Interface

Memory Interface receives original images from a camera and stores these images into two input memory block. This block also distributes the received images to the image scaler block and the MCT generator block. And this memory interface block receives scaled down images from the image scaler and stores them into to scale-down memory block. This block also generates synchronizing information and image size information required for processing images, and transmits them to other blocks.

3.2 Image Scaler

Image Scaler down-scales the images input from a camera by phases during the 13 steps of scaling. Such down-scaling is to find a face with windows for each phase while down-scaling the original image by phases and identifying faces of various sizes because the size of the window seeking a face is fixed to 20x20 in the image. It eventually enables to detect a face regardless of the size occupied in the image.

Image Scaler uses bilinear interpolation to down-scale an image. Table 1 below shows the changes in size of the down-scaled images and the duration required for down-scaling.

Table 1. Size changes and processing time with 54 MHz operating speed

Scaling Step	Image Size	Processing Time	Accumulated Time
0 (Original size)	320 x 240	1. 464 ms	1.464 ms
1	284 x 213	1. 152 ms	2. 616 ms
2	252 x 189	0. 910 ms	3. 526 ms
3	224 x 168	0.718 ms	4. 244 ms
4	199 x 149	0.561 ms	4. 805 ms
5	176 x 132	0.440 ms	5. 245 ms
6	156 x 117	0. 344 ms	5. 589 ms
7	138 x 104	0. 268 ms	5. 857 ms
8	122 x 92	0. 210 ms	6. 067 ms
9	108 x 81	0. 166 ms	6. 233 ms
10	96 x 72	0.129 ms	6. 363 ms
11	85 x 64	0.101 ms	6. 464 ms
12	75 x 56	0.078 ms	6. 542 ms
13	66 x 49	0.060 ms	6. 602 ms

3.3 MCT Generator

This block internally consists of window Interface block and MCT calculator block. The window interface block transmits the nine pixels in a 3x3 window to MCT calculator at the same time while moving the window in an image. For the 3x3 window extracted from the window interface block, the MCT calculator performs MCT operation.

3.4 Candidate Detector and Confidence Mapper

For these MCT-transformed images, 20x20 search windows are used to detect candidate facial areas. The search window moves by one pixel in the horizontal direction first from the left top end. Every pixel in the window has unique non-facial reliability value to the MCT value. Each of the 400 pixels goes through the process that determines it is out of the face if the total exceeds the threshold. This process is designed with the divided modules of Candidate Detector and Confidence Mapper.

Candidate Detector collects the successively input MCT images while providing 20x20 window images, using total 19 Line Memories and a separate delaying logic.

As shown in Table 2, each pixel of the 20x20 window has a unique reliability value to the MCT value for each location based on the data learned by AdaBoost. Confidence Mapper sums the 400 non-facial reliability values and outputs the coordinate and reliability when the total is within the threshold.

Table 2. Confidence Rule Table using fixed point transform.

Confidence Value Q 8.8 Format (16bits) Transform example. (at coordinate (1, 1))			
MCT Value	Real number value	Integer value	Float value
1	1.302622	00000001	01001101
2	1.77252	00000001	11000101
3	0.487639	00000000	01111100
4	0.937634	00000000	11110000
5	0.15517	00000000	00100111
6	0.316182	00000000	01010000
7	0.308015	00000000	01001110
8	1.77252	00000001	11000101
9	1.154538	00000001	00100111
10	0.358852	00000000	01011011
11	0	00000000	00000000
511	0.153845	00000000	00100111

Table 2 shows the example non-facial reliability values of each MCT value for locations (1, 1) in a window. For example, the non-facial reliability value is 0.487639 when the MCT value is 3 at the location (1, 1) in the window. This value is 0 when the MCT value is 11 and it means the location is a part of a face with the high probability. The 400 reliability values are summed in the window to select candidate areas. To implement the hardware, as shown in Table 2, the probability value within the range of real numbers are converted into fixed points Q 8.8 (16 bits) to produce 16-bit Confidence LUT. Confidence Mapper was thus implemented by creating total 400 Confidence ROMs (511x16bit).

3.5 Position Resizer

This block calculates the corresponding location in the original image for any candidate facial area found in a down-scaled image. To reduce the complex real number calculation and time for real-time processing, this paper first calculates the corresponding coordinates in the original image for all the coordinates, converts them into integers, saves the values in the ROM table. Thus original position can be calculated by simple LUT read operation.

3.6 Data Grouper

A process is required to group the duplicate areas determined as the same face prior to determining the final face detection areas: the candidate areas detected by Candidate Detector and Confidence Mapper are detected four to five times around the final detection area.

(a) Before grouping (b) After grouping

Fig. 4. Before and after image of grouping

As shown in Figure 4, duplicate areas can be detected not only in the same image size but in different down-scaled sizes. Overlap is determined when an existing area is overlapped with at least 1/4 of a newly detected area and then the smaller value - i.e. more possibly facial area - is selected out of the two non-facial reliability values. This paper determined final detection areas only when the overlap frequency is at least three to lower the wrong detection rate and raise the detection reliability.

3.7 Overlay Processor

This block finally displays the face detection result after adding the final detection areas provided by Data Grouper. This block enables to instantly view the face detection results without the help of the processor.

4 Design Verification and Performance Analysis

The verification environment was realized with the system by Virtex5 lx330 FPGA where the result was verified by converting the 320x240 QVGA camera images at the gray level. The proposed architecture operates at the clock speed of 13.5 MHz using a memory of 53.01 KB (31.6 KB RAM + 21.4 KB ROM) and 74,632 (35%) LUTs.

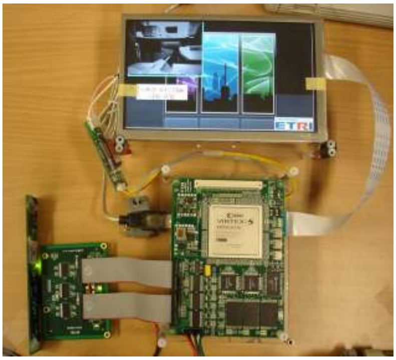


Fig. 5. Verification environment

The face detection performance was verified using the frontal face test set [9] and Yale Test set[11], each face with different expressions(angry, surprised, etc., under various lighting conditions) provided by MIT, CMU and Yale University. The MIT+CMU three tests A, B, and C include total 130 frontal images with 506 faces and 50 rotated images with 223 faces for the rotated test set. For the detection performance, as shown in Table 3, the detection rate was as high as 80.48%.

Table 3. Face Detection rate. (MIT+CMU and Yale Test set)

Class of test set	Detection rate	False-positives
Test Set A (CMU)	70.41% (119/169)	1
Test Set B (MIT)	64.94% (100/154)	0
Test Set C (CMU)	85.25% (156/183)	0
Sum of A, B, C	74.11% (375/506)	0
Yale Test set	100% (165 / 165)	0
Average	80.48% (540/671)	-

Fig. 6. Results from the Yale test set and MIT + CMU test set

Fig. 7. Rotated face case: correctly detected faces and missed faces

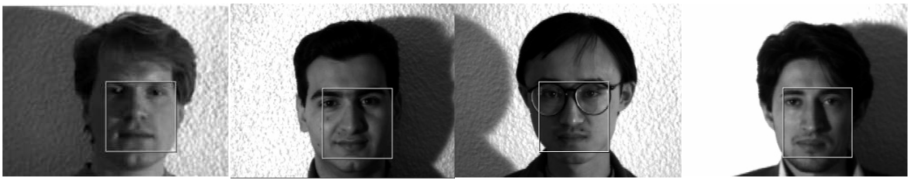


Fig. 8. Output result in various intensity level

Figure 6 shows a successful detection for a test image. For rotated faces, as shown in Figure 7, the system could detect not only a frontal face but also a rotated face within 12° left and right and within 15° up and down.

It is noteworthy that it showed reliable detection performance not only with normal illumination but also for a face blocked with illumination or for a bright face without any difference in brightness.

5 Conclusion and Future Research Plan

There are already various researches reported for face detection algorithms. This paper applied a robust detection algorithm to overcome the deterioration of the detection rate due to illumination by MCT. And it implemented a high-performance real-time detection engine operated at 30 to 60 fps. The result of this paper is very meaningful compared with so far reported face detection hardware engines because they have not been for real-time processing due to low processing speed and limited performance.

Accurate acquisition of the face location with a high-performance face detector can contribute to the improvement of the face recognition technology. The detection engine introduced in this paper needs to solve some tasks, despite its excellent detection rate, for complete application. For the images from an indoor robot, unlike those from security systems, the face may not be frontal due to limited cooperation to the system.

For the future research plan, therefore, complementary research is under way to establish a model that can detect a face rotated in the lateral or vertical direction, and it is expected to contribute to development of a dedicated visual processing chip for robots with the addition of a recognition function.

Acknowledgments. This work is supported by ETRI and Seoul Research & Business Development Program (CR070048). The hardware verification tools are supported by IC Design Education Center.

References

1. Samir, N., Michael, T., Raj, N.: Biometrics, pp. 63–75. Wiley Computer Publishing, Chichester (2002)
2. Bernhard, F., Andreas, E.: Face detection with the Modified Census Transform. In: 6th IEEE International Conference on Automatic Face and Gesture Recognition, pp. 91–96 (2004)

3. Paul, V., Michael, J.J.: Robust real-time face detection. International Journal of Computer Vision, 137–154 (2004)
4. Yoav, F., Robert, E., Schapire: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 119–139 (1997)
5. Ming-Hsuan, Y., Dan, R., Narendra, A.: A snow-based face detector. In: 12th Advances in Neural Information Processing Systems, pp. 855–861 (2000)
6. Najwa, A., Srivaths, R., Anand, R.M., Niraj, K.J.: Hybrid architectures for efficient and secure face authentication in embedded systems. In: 7th IEEE Transaction on VLSI Systems, vol. 15(3), pp. 296–308 (2007)
7. Su-Hyun, L., Yong-Jin, J.: A design and implementation of Face Detection hardware. In: 44th IEEK Transaction on System Design, vol. 44(4), pp. 43–54 (2007)
8. Duy, N., David, H., Parham, A., Ali, S.: Real time Face detection and Lip feature extraction using Field-Programmable Gate Arrays. IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics 36(4), 902–912 (2006)
9. Rowley, H., Baluja, S., Kanade, T.: Neural network-based face detection. IEEE patt. Anal. Mach. Intell., 22–38 (1998)
10. CMU/VASC Image Database, `http://vasc.ri.cmu.edu/idb/html/face/index.html/`
11. Georghiades, A.: Yale Face Database, Center for Computational Vision and Control at Yale University, `http://cvc.yale.edu.proje/yalefaces/yales.html`