International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:4, 2007

Intelligent Vision System for Human-Robot Interface

Al-Amin Bhuiyan, and Chang Hong Liu

***Abstract*—**This paper addresses the development of an intelligent vision system for human-robot interaction. The two novel contributions of this paper are 1) Detection of human faces and 2) Localizing the eye. The method is based on visual attributes of human skin colors and geometrical analysis of face skeleton. This paper introduces a spatial domain filtering method named 'Fuzzily skewed filter' which incorporates Fuzzy rules for deciding the gray level of pixels in the image in their neighborhoods and takes advantages of both the median and averaging filters. The effectiveness of the method has been justified over implementing the eye tracking commands to an entertainment robot, named "AIBO".

***Keywords*—**Fuzzily skewed filter, human-robot interface, rms contrast, skin color segmentation.

I. Introduction

WHEN robots are working cooperatively with human beings, it is necessary share and exchange their intentions and interests. Since the communication between humans and robots is established by means of visual perceptions, eye tracking and gaze directions, are important clues for this interaction. Development of an intelligent vision system is, therefore, a vital issue for robotics research.

A fair amount of techniques have been introduced recently in literature on face detection, eye localization, eye tracking and gaze detection. For example, geometric modeling [1], auto-correlation [2], neural networks [3], principal component analysis [4] and so on. Model based approaches assume that the initial location of the face is known. Color based approaches reduce the search space in face detection algorithm. The neural network-based approaches require a large number of face and non-face training examples, and are designed primarily to locate frontal faces in grayscale images. Traditional approaches of eye tracking use the controlled infrared lighting to illuminate the eye through geometric projections. These methods normally involve specialized high speed/high resolution camera, controlled lighting source, hardware equipments and are sometimes intrusive [5]. Sung and Poggio [6] have developed an example-based approach for locating frontal views of human face in complex scenes. Since this method has been developed for vertical frontal view faces, faces with other orientations cannot be detected. Rowley et al. [3] have developed a neural network-based frontal face detection system where a retinal connected neural network has been employed to justify the small windows of size 20×20 of an image whether it contains a face or not. Lam and Yan [7] have used snake model for detecting face boundary. Although the snake provides good results in boundary detection, but the main problem is to find the initial position. Moghaddam and Pentland [8] employed principal component analysis for describing the face pattern with lower-dimensional feature space.

This paper explores a face detection and eye tracking system which integrates the detection of human faces in different lighting conditions and localization of eyes on it. Face detection is established by a two step method: (i) skin color segmentation and (ii) morphological operations. The system is based on the detection of the face area and thresholding the image on color segmentation. The location of the face is then determined by employing morphological operations. The eye is then localized from the knowledge of the face geometry. Experimental results indicate that the system is capable of detecting face and locating the eye position from complex backgrounds with a high degree of variability in lighting conditions, pose and expression.

The remainder of this paper is organized as follows. Section II describes the vision system for human-robot interface. Face detection and eye localization methodologies are also illustrated. Section III presents experimental results and performance. Section IV draws the overall conclusion of this paper.

II. Vision System for Human-Robot Interface

The vision system for human-robot interaction involves an algorithm to detect human face and to localize the eye positions. The system uses video camera for data acquisition and is commenced with image pre-processing operations. Fig. 1 illustrates the architectural design of the human-robot interaction system.

The system is organized in two steps. The first step is dedicated to the detection of face in image sequences using skin color segmentation. Eye tracking is established in the second step to control the robot in accordance with the intentions of the user.

A. Image Pre-Processing

The video image is first subjected to image pre-processing operations to account for different lighting conditions and contrast. The image pre-processing phase includes contrast and illumination equalization, histogram equalization, and filtering.

Md. Al-Amin Bhuiyan is with University of Hull, HU6 7RX, Hull, UK, on study leave from the Department of Computer Science and Engineering, Jahangirnagar University, Bangladesh (corresponding author to provide phone: 1482-46-5026; e-mail: M.Bhuiyan@ hull.ac.uk).

Chang Hong Liu is with the Department of Psychology, University of Hull, HU6 7RX, Hull, UK (e-mail: c.h..liu@hull.ac.uk).

Contrast and Illumination Equalization

Contrast is a measure of the human visual system sensitivity. To achieve an efficient and psychologically-meaningful representation, and make the image illumination invariant in terms of sunny or cloudy environments, it is first processed with fixed rms contrast and illumination equalization.

The rms (root mean square) contrast which is equivalent to the standard deviation of luminance, is given by [9]:

$$\left.\begin{aligned} C_{r,rms} &= \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_{r,i}-\bar{x}_r)^2\right]^{1/2} \\ C_{g,rms} &= \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_{g,i}-\bar{x}_g)^2\right]^{1/2} \\ C_{b,rms} &= \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_{b,i}-\bar{x}_b)^2\right]^{1/2} \end{aligned}\right\}, \qquad (1)$$

where $x_{r,i}$, $x_{g,i}$, $x_{b,i}$ are the normalized illumination due to red, green and blue color components, respectively, such that $0 < x_{r,i} < 1$, $0 < x_{g,i} < 1$, $0 < x_{b,i} < 1$, and $\bar{x}_r$, $\bar{x}_g$, $\bar{x}_b$ are the mean normalized illuminations due to red, green and blue color components. With this definition, images captured at different lighting conditions will have the same contrast if their rms contrasts are equal. The rms contrast does not depend on spatial frequency contrast of the image or the spatial distribution of contrast in the image. All images are maintained with the same illumination and same rms contrast using the following equations:

$$\mathbf{g}_r = \alpha_r \mathbf{f}_r + \beta_r,\ \mathbf{g}_g = \alpha_g \mathbf{f}_g + \beta_g,\ \mathbf{g}_b = \alpha_b \mathbf{f}_b + \beta_b, \quad (2)$$

where α_r, α_g, α_b are the contrasts due to red, green and blue color components, respectively, and β_r, β_g, β_b are the brightness to be increased or decreased from the respective red, green and blue components $\mathbf{f}_r$, $\mathbf{f}_g$, $\mathbf{f}_b$ of the original image to $\mathbf{g}_r$, $\mathbf{g}_g$, $\mathbf{g}_b$ of the new image $\mathbf{g}$. The illumination and rms contrast equalization process is illustrated in Fig. 2.

Histogram Equalization

The face images may be of poor contrast because of the limitations of the lighting conditions. So histogram equalization is used to compensate for the lighting conditions and to improve the contrast of the image [10].

Let the histogram $h(r_i) = \frac{p_i}{n}$ of a digital face image consists of the color bins in the range $[0, C-1]$, where r_i is the *i*-th color bin, p_i is the number of pixels in the image with that color bin and *n* is the total number of pixels in the image. For any *r* in the interval [0,1], the cumulative sum of the bins provides with some scaling constant [11]. Histogram equalization is performed by transforming the function $s = T(r)$, which produces the mapping with the allowed range of pixel values, i.e., a level s for every pixel value r in the original image and $0 \le T(r) \le 1$ *for* $0 \le r \le 1$.

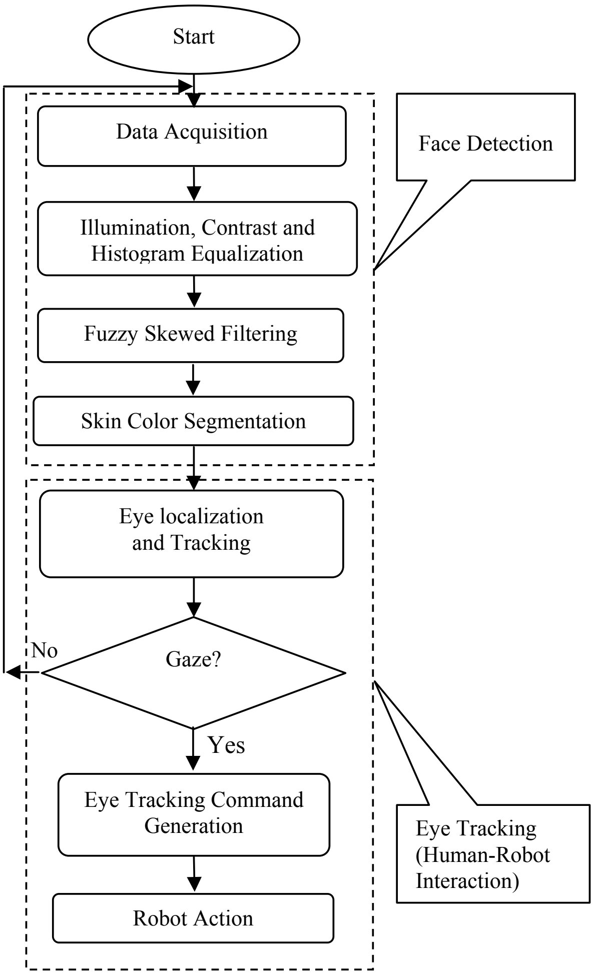
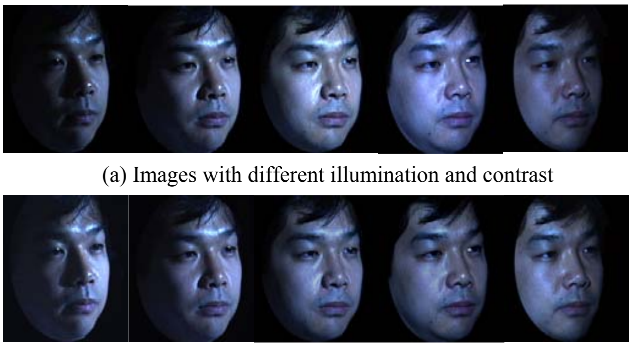

Start
Data Acquisition
Illumination, Contrast and Histogram Equalization
Fuzzy Skewed Filtering
Skin Color Segmentation
Face Detection
Eye localization and Tracking
Gaze?
No
Yes
Eye Tracking Command Generation
Robot Action
Eye Tracking (Human-Robot Interaction)

Fig. 1 Architectural design of human-robot interaction system



(a) Images with different illumination and contrast

(b) Images with same illumination and same rms contrast

Fig. 2 Illumination and rms contrast equalization. Images were captured at illumination angles of -38.4º, -21.6º, -0.2º, 20.6º, 37º, respectively, all images at a pose of 31.08º. After rms contrast

scaling, the average luminances are: $\overline{x}_r$ =35.99, $\overline{x}_g$ =41.21, $\overline{x}_b$ =58.71, respectively and rms contrasts are: $C_{r,rms}$ =39.80, $C_{g,rms}$ =45.98, $C_{b,rms}$ =63.58, respectively

Fuzzily Skewed Filter

Various sources of noise may exist in the input image. The fine details of the image represent high frequencies which mix up with those of noise. So low-pass filters are used to obliterate some details in the image. This paper introduces a new filter named "fuzzily skewed" filter to suppress the noise.

In the fuzzily skewed filter, fuzzy rules are applied for deciding the brightness of a pixel in the image from the neighborhood of that pixel. This is a modification of the median filter and neighborhood averaging filter. The decision process includes the following steps:

1. The brightness of the neighborhood pixels (n×n neighborhood) are stored and then sorted in ascending or descending order.
2. Fuzzy membership value is assigned for each neighbor pixels with the following notions:
    i. A Π-shaped membership function is defined.
    ii. The highest and lowest brightness get the membership value 0.
    iii. Membership value 1 is assigned to the mean value of the neighborhood.
3. Consider only 2×k+1 pixels (k<=n^2/2), where k is the number of participant pixels in the skewing process, in the sorted pixels list (these are the median brightness value and k previous and forward values in sorted list. The value of k is chosen empirically.
4. Select the value that has the highest membership value and place it as the output brightness of the corresponding pixel.

Example: Let us consider a 3x3 neighborhood with pixels as follows:

91	114	175
92	116	176
95	111	182

Here,
Original value: 116;
Mean value: 128;
Median value: 114;
Let range value, k=2;

Sorted list: [91, 92, 95, 111, 114, 116, 175, 176, 182];
Membership value: [0, 0.0018, 0.0286, 0.5635, 0.6864, 0.7622, 0.0409, 0.0302, 0];
Therefore, considering the list of pixels (centering the median value) with brightness levels {95, 111, 114, 116, 175}, the pixel with value 116 will be selected (corresponding to highest membership value of 0.7622).

The membership function employed for the fuzzily skewed filter, as shown in Fig. 3, is a π - shaped curve, implemented as a combination of s- curve and z-curve, respectively given by [12]:

$$s(x_l, x_r, x) = \begin{cases} 0, & x < x_l \\ \dfrac{1}{2} + \dfrac{1}{2}\cos\left(\dfrac{x - x_r}{x_r - x_l}\pi\right), & x_l \le x \le x_r \\ 1, & x > x_r \end{cases} \tag{3}$$

$$z(x_l, x_r, x) = \begin{cases} 1, & x < x_l \\ \dfrac{1}{2} + \dfrac{1}{2}\cos\left(\dfrac{x - x_l}{x_r - x_l}\pi\right), & x_l \le x \le x_r \\ 0, & x > x_r \end{cases} \tag{4}$$

where x_l and x_r are the left and right breakpoints, and x is brightness of the corresponding pixel, respectively.

This method incorporates the advantages of both the median filter and averaging filter. For k=0, it acts like median filter and for k<=n^2/2, it acts like neighborhood averaging filter. This method can successfully reduce noise that results due to sharp transitions in the brightness, and it does not make the resultant image as blur as the neighborhood averaging filter.

B. Face Detection

Face detection is achieved by employing skin color segmentation. The skin color segmentation process is based on visual information of the human skin colors from the image sequences. For this, the dominant and perceptually relevant skin colors are extracted from image sequences. There are two ways of segmenting the image based on skin color: converting the RGB picture to YCbCr space or to HSV space. The YCbCr space segments the image into a luminosity component and color components, whereas an HSV space divides the image into the three components of hue, saturation and color value. The main advantage of converting the image to the YCbCr domain is that influence of luminosity can be removed during image processing. In the RGB domain, each component of the picture (red, green and blue) has a different brightness. However, in the YCbCr domain all information about the brightness is given by the Y-component, since the Cb (blue) and Cr (red) components are independent from the luminosity.

Images are being searched in YCbCr space depending on the amount of color content of image, that is, whether the skin color value is present in an image or not. Conversion from RGB to YCbCr is given by the following equation [13]:

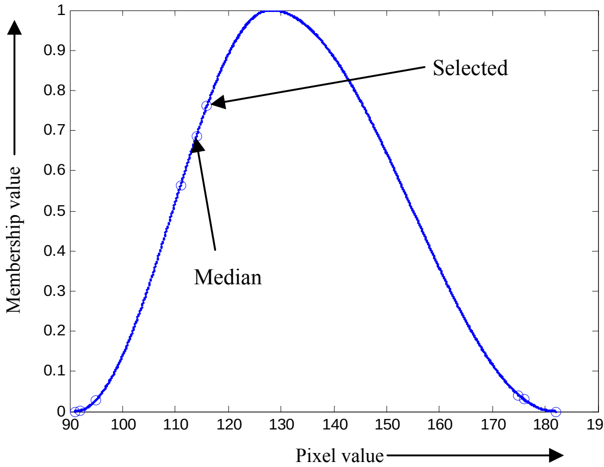
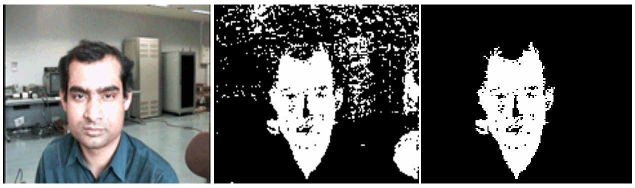


Fig. 3 Membership function for the fuzzily skewed filter



(a) Original image (b) Skin color segmented (c) Face area detected
Fig. 4 Skin color segmentation

$$\left.\begin{aligned} Y &= 0.257\times R + 0.504\times G + 0.098\times B + 16 \\ C_b &= 0.148\times R - 0.291\times G + 0.439\times B + 128 \\ C_r &= 0.439\times R - 0.368\times G - 0.071\times B + 128 \end{aligned}\right\} \quad (5)$$

where R,G, B are the red, green, and blue component values which exist in the range [0,255].

The Cb and Cr components provide a significant indication on whether a pixel belongs to the skin or not. There is a strong correlation between the Cb and Cr values of skin pixels. Since the human skin colors are clustered in color space and differ from person to person and of races, so in order to detect the face parts in an image, the skin pixels are thresholded empirically [14]. In this experiment, the threshold value is chosen by the following equation:

$$(140 < C_r < 160) \text{ and } (105 < C_b < 140) \quad (6)$$

The detection of face region boundaries by such a YCbCr segmentation process is illustrated in Fig. 4.

The exact location of the face is then determined from the image with largest connected region of skin-colored pixels. The connected components are being determined by applying a region-growing algorithm at a coarse resolution of the segmented image. In this experiment, 8-pixel neighborhood connectivity is employed. In order to remove the false regions from the isolated blocks, smaller connected regions are assigned by the values of the background pixels. After thresholding the image may be encountered by some holes in the face skin region. In order to remove the false regions, the face region is subjected to morphological dilation operation with a 3×3 structuring element several times followed by the same number of erosion operations using the same structuring element. The dilation operation is used to fill the holes and the erosion operations are performed on the dilation results to restore the shape of the face [14].

C. *Eye Localization*

Once the face is found in an image, the eyes are then searched around the restricted area inside the face. The face image block is then divided into four parts and the left and right eyes are then searched in the upper left and upper right sub-blocks of the facial image, respectively. Assuming a frontal view of the face, eyes are then extracted from this face area, assigning a unique tag to each isolated candidate block by labeling in the binarized image. These tag points are the representatives of each feature block.

Let $\mathbf{P}_j = (x_i, y_i), (j \in [1,2])$ be the two tag points of the image blocks of the left eye and right eye, respectively. Since in the binary image, all of the pixels have got the intensity value of 0 and 255 only, these tag points are assigned with their respective representative numbers. The first tag point, i.e., the tag point for the left eye, is determined by searching for the first white pixel (intensity value of 255) from bottom to up and right to left in the top left quadrant of the image window and assigning it with the value of 128. The algorithm for this searching method is as follows:

[Step 1] Compute the center point of the face:

[Step 2] Starting from the center point, search the first black pixel and assign it with the value of '128' (first tag value) as follows:

```
$For\ \ i \to x_{\max}/2$ to 1 step $-1$
    $For\ \ j \to y_{\max}/2$ to 1 step $-1$
        $If\ \ \mathbf{Pixel}[i][j] == 0\ \ then\ \ \mathbf{P}_i = 128$
return
```

[Step 3] Search the entire window for the tag value '128' and assign all connected pixels with '128'.

Similarly, the 2nd tag point and its associated connected pixels are determined from the top right quadrant of the searching window and assigned with its representative number. These tag points and representative numbers are starting positions of two eyes.

D. *Calculating Gravity Center of the Eye*

Since the exact location of the pupil is very confusing, so instead of searching the pupil point, we are determining the position of the gravity center of the eyes from the black portions of the irises of the segmented image. If the gray level at each point (x, y) of the given area E of an eye is considered as the "mass" of (x, y), we can define the center of gravity of that component, as well as the moment of inertia about specified points or lines. The pq-th moment of area E about the origin (0,0) is given by:

$$m_{pq} = \sum_{(x,y)\in E}\sum x^p y^q \quad (7)$$

where (x, y) are the coordinates of the pixel included in area E. The 0-th moment m_{00} represents the area E and the center of gravity of E is the point $(\bar{x}, \bar{y})$ whose coordinates are given by [11]:

$$\bar{x} = \frac{m_{10}}{m_{00}} = \frac{\sum\limits_{(x,y)\in E}\sum xy^0}{\sum\limits_{(x,y)\in E}\sum x^0 y^0}, \ \bar{y} = \frac{m_{01}}{m_{00}} = \frac{\sum\limits_{(x,y)\in E}\sum x^0 y}{\sum\limits_{(x,y)\in E}\sum x^0 y^0} \qquad (8)$$

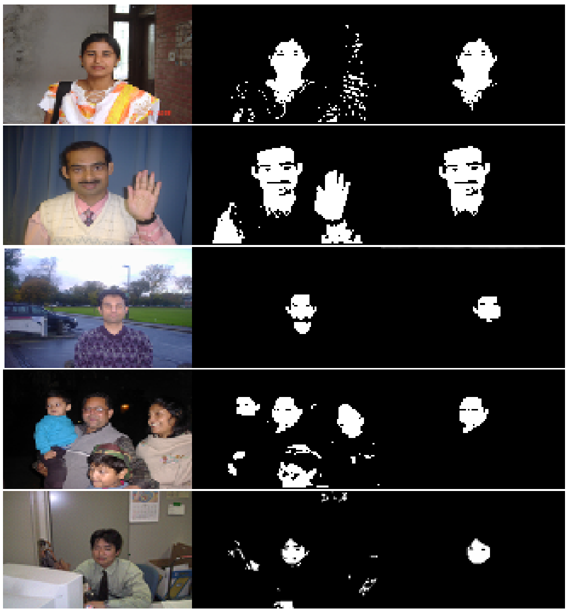
On detection and localization of eyes, the gaze positions are estimated. Robots are then instructed depending on the movement of the eyes.

III. Experimental Results and Performance

The effectiveness of this approach has been justified using different images with various kinds of lighting conditions. Experiments were carried out on a Pentium IV 2.2MHz PC with 256 MB RAM, and the SONY VISCA camera. The algorithm has been implemented using Visual C++. When a complex image is subjected in the input, the face detection result highlights the facial part in the image, as shown in Fig. 5. Most of the images are captured using a digital camera, but some are from scanner, and some from video tapes recorded from different television channels. The algorithm is capable of detecting single face in an image. For multiple faces, the system finds the largest face only, i.e. the face containing more number of skin colored pixels. A total of 410 images, including 82 different persons, were used to investigate the capacity of the proposed algorithm. Among them only 6 faces were found false. Experimental results demonstrate that the success rate of approximately 98.5% is achieved. The main reason behind the failure of those images in finding face regions is the substantially presence of pink, reddish or yellowish background regions in the image which are much larger than the true skin regions.

The response of the proposed fuzzily skewed filter has been analyzed. A graphical comparison between the ‘Fuzzily skewed’ filter and averaging and Median filter is shown in Fig. 6. The impact of mask size on the output of the ‘Fuzzily skewed filter’ is illustrated graphically in Fig. 7. The graphical analysis imply that the fuzzily skewed filter snatches the advantages of both the median filter and averaging filter. This method can successfully reduce noise, and on the contrary, it does not make the resultant image as blur as the neighborhood averaging filter.

Eye localization has been performed successfully with images of different persons at various kinds of lighting conditions and environments both in shiny and cloudy weather. Fig. 8 shows the eye detection results at different environments.



(a) (b) (c)

Fig. 5 Face detection results of some images. (a) original images (b) skin color segmentations (c) Detected faces

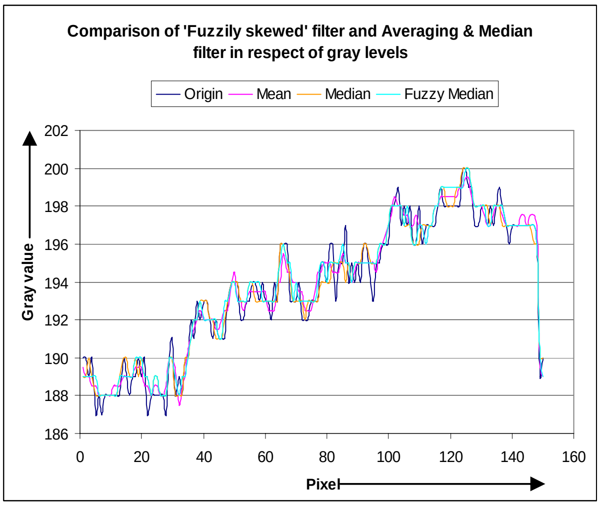


Fig. 6 Comparison between ‘Fuzzily skewed’ filter and Averaging and Median Filter

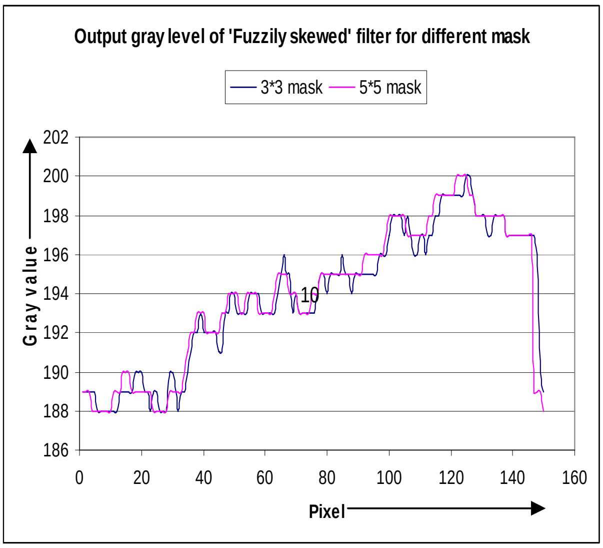
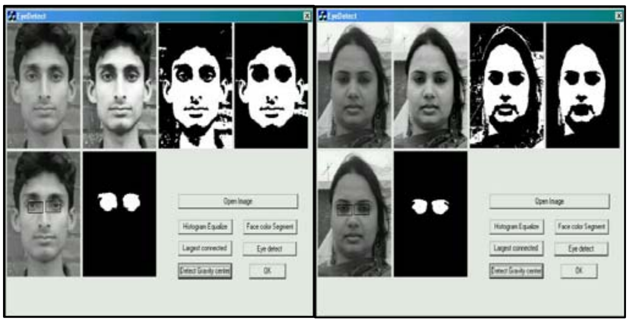
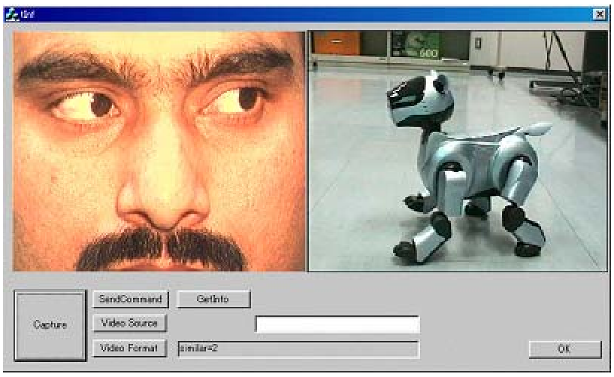


Fig. 7 Comparison of gray values of output image for use of different mask using 'Fuzzily skewed' filter



(a) Dark environment (b) Cloudy environment

Fig. 8 Eye detection at different environments



similar=2

Fig. 9 Eye tracking command to motivate Aibo to start walk

The system has been implemented by an eye tracking program running in the PC. The PC has been considered as the server and AIBO as a client. The communication link has been established through TCP-IP protocol. The human-robot interaction results are: (i) turn light off, (ii) move forward, (iii) turn right, and so on, according to the eye positions at left top, center top, and right top, respectively. In order to demonstrate different user's desires, this system defines, interprets and recognizes 10 commands depending on the eye movements. These eye tracking commands and their corresponding actions are furnished in Table I. It is possible to recognize more eye movements including new gazes. Finally, a real time human-robot interaction system has been implemented on eye tracking commands. One of the actions of the robot as a result of eye movement is shown in Fig. 9.

TABLE I
EYE TRACKING COMMANDS TO CONTROL AIBO

Eye Positions	Actions
Left Top	Turn Light Off
Center Top	Move Forward
Right Top	Turn Right
Left Middle	Turn Left
Center Middle	Start Walking
Right Middle	Dance
Left Down	Turn Light On
Center Down	Move Backward
Right Down	Sing a Song
Eye Close	Stop Walking

IV. CONCLUSION

This paper presents an intelligent vision system to control the robot depending on the movement of the eyes. A fast face detection and eye localization method has been established depending on color segmentation and morphological analysis.

Detection of faces and facial features using machine vision techniques has many useful applications. Though human beings accomplish these tasks countless times a day, they are still very challenging for machine vision. In this paper, face detection is established by skin color segmentation and morphological operations. Here, eye localization is done by calculating largest connected area from the skin color segmented image and then searching the tag positions of the two eyes. Finally, the gravity centers of eyes are calculated and robot is being instructed depending on the movement of the eyes. Our next approach is to extend the algorithm for multi-face detection and overlapping faces in images and images with obstacles in front of eye (e.g. spectacles) with different orientation and pose and images with more complex backgrounds.

A fuzzy rule based spatial domain filter is also proposed in this project. Experimental results justify that it incorporates the advantages of median and averaging filter with minimizing their limitations. Implication of the rms contrast scaling and fuzzily skewed filtering provide momentum in spatial domain image enhancement for the development of intelligent vision system. Our main target is to instruct operations to robots and make them understand the human intensions and interests over facial expressions and gaze detection so that they are capable of grasping with more intelligence while working cooperatively with human beings.

ACKNOWLEDGMENT

This research is supported by Economic & Social Research Council RES-000-22-1759. The authors would like to express their deep sense of gratitude to Mr. Ezharul Islam and Ms. Nasima Begum for their valuable supports.

REFERENCES

[1] H. Shinn, and H. Hui-Ling, "Facial modeling from an uncalibrated face image using a coarse-to-fine genetic algorithm", Pattern Recognition, Vol. 34, No. 8, 2001, pp. 1015-1031.

[2] F. Goudail, E. Lange, T. Iwamoto, K. Kazuo, and N. Otsu, "Face recognition system using loacal autocorrelations and multiscale integration", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 10, 1996, pp. 1024-1028.

[3] H. Rowley, B. Shumeet, and T. Kanade, "Neural network-based face detection", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, 1998, pp. 23-37.

[4] M. Turk, and A. Pentland, "Eigenfaces for recognition", Journal of cognitive neuroscience, Vol. 3, No. 1, 1991, pp. 71-86.

[5] D. Tampe, "Heuristic filtering and reliable calibration methods for video based pupil-tracking systems, *Instruments and Computers*, *25*(2), 1990, 137-142.

[6] K. Sung, and T. Poggio, "Example-based learning for view-based human face detection", IEEE Transactions on Pattern Analysis and Machine Intelligence ,Vol. 20, No. 1, 1998, pp. 39-50.

[7] K. Lam, and Y. Hong, "Locating and extracting the eye in human face images", Pattern Recognition, Vol. 29, No. 5, 1996, pp. 771-779.

[8] B. Moghaddam, and A. Pentland, "Face recognition using View-Based Modular Eigenspaces", Proc. of Automatic Systems for the Identification and Inspection of Humans, SPIE Vol. 2277, 1994.

[9] E. Peli, "Contrast in Complex Images", Journal of Optical Society, Vol. 7, No. 10, 1990, pp. 2032-2040.

[10] Y. Tae-Woong, O. Il-Seok, "A fast algorithm for tracking human faces based on chromatic histograms", Pattern Recognition Letters, Vol. 20, No. 10, 1999, pp. 967-978.

[11] R.C. Gonzalez, R.E. Woods, Digital image processing, Prentice Hall, Inc., 2nd Edition, London, 2002, pp. 88-93.

[12] J. Jantzen, "Tutorial on Fuzzy Logic", www.iau.dtu.dk/~jj/pubs/logic.pdf

[13] D. Marius, S. Pennathur, and Klint Rose, "Face detection using color thresholding and eigenimage template matching", ww.stanford.edu/class/ee368/Project_03/Project/reports/

[14] M.A. Bhuiyan, V. Ampornaramveth, S. Muto, and H. Ueno, "On Tracking of Eye for Human-Robot Interface", International Journal of Robotics and Automation, Vol. 19, No. 1, 2004, pp. 42-54.