A Visual Interaction System Using Real-Time Face Tracking

Chil-Woo Lee and Akitoshi Tsukamoto

Laboratories of Image Information
Science and Technology
1-4-2 Shinsenri-Higashimachi
Toyonaka, Osaka, 565 Japan
cwlee@image-lab.or.jp

Kazuya Hirota and Saburo Tsuji

Faculty of Engineering Science
Osaka University
1-3 Machikaneyama-Cho
Toyonaka, Osaka, 560 Japan

Abstract

This paper describes a new method for pose estimation of human face moving abruptly in real world and its hardware implementation for real-time operation. The virtue of this method is to use a very simple calculation, correlation, among multiple model images, and not to use any facial features such as facial organs, so that it is very robust and can be easily practised with simple hardware.

1 Introduction

In developing a intelligent Human Interface system, visual approach is very useful since it does not need to attach any physical sensors on human body. However, visual data; image, is very unstable and has many problems to know about real world because it is nosy and a lot of three dimensional information while being formed. So, even though many computational algorithms are developed in computer vision field, more robust and practical methods should be developed for recognitizing real environment.

In this paper, we present a new method detecting human face and estimating pose of it, and a hardware system implementing the method in real-time. This method is based on simple calculation of correlation between input image and multiple model images which are artificially synthesized for various view direction. The merit of the method is that we do not use any features of face gestalt, so that complex calculations are not needed. Some portion of the algorithm have been already proposed and verified to be robust in preceding researches[1] [2].

To detect face from image, many methods have been developed, for example, methods employing principal component analysis[3], hierarchical symmetry[4], or matching with blocked template[5], however, it still has remained difficult to make a general model with flexibility for scale change and individuality. However, it can be said that the template matching method has some advantages in computational cost and robustness rather than geometric matching of facial features[6].

To bear with diversity occurring from the scale change and different person, we here employ a block matching technique as a model-based approach for detecting frontal face[1]. We call this face model "Qualitative Model for Face (QMF)", which contains statistical information about brightness according to small block. The information is gathered from a lot of sampled images of frontal face and refined as qualitative features according to each blocks. These features are used to calculate faceness of regions everywhere of input images, and a region which includes the highest faceness is taken as the initial model image.

After initial face region is extracted from input image, to track the face and estimate its pose, some model images are synthesized by texture mapping and reprojection technique. We have presented the whole of the algorithm as a face tracking and pose estimation algorithm using multiple model images[2], and in this paper, a modified one of the algorithm is adopted for real-time implementation.

As explained above, our approach is based on correlation between two images, and it is very easy to implement with simple architecture. Therefor, we have constructed a hardware system in which a commercial VLSI are used for block matching process, and the system is connected to other machines through Ethernet. The performance of the system is at least 15GIPS, and it corresponds to that we can calculate a motion vector and minimum correlation value of a template (16×16×8bit) from a image (512×512×8bit) twice in 33msec. And, since it is constructed on network system, we can easily apply the recognition results to control other machines.

The contents of the paper is as follows. We describe the detection process of a full face by using the qualitative model in section 2, and in section 3, the process of pose estimation is explained. A hardware system realizing our method in real-time is introduced in section 4. In latter part of the paper, we illustrate experimental results and conclude it with analysis of some features of the method.

2 Acquisition of Full Face

2.1 QMF: Qualitative Model for Face

Considering images of frontal face without beard, mustache, whisker and glasses, we become aware that

1058-6393/95 $4.00 © 1995 IEEE

the same portion of every image have similar features. For example, many images of frontal face have black hair(for oriental), light skin, and much edge components around eyes and mouth. To utilize such features, we divide facial images into N blocks like mosaic pattern, and identify each blocks using qualitative features, namely, "*Lightness*" which tends to be higher value in bright regions, and "*Edgeness*" which indicates relative amount of edge components. These two factors are used in our model and calculated from a lot of previously sampled image data.

The "*Lightness*" $L_{face}(i)$ and "*Edgeness*" $E_{face}(i)$ for the ith block $B_{face}(i)$ in sampled facial region are defined as

$$\begin{aligned} L_{face}(i) &= Nl_{face}(i)/N_{face}(i) \\ E_{face}(i) &= Ne_{face}(i)/N_{face}(i) \end{aligned} \quad (1)$$

where $N_{face}(i)$ is the number of pixels in a block $B_{face}(i)$, $Nl_{face}(i)$ is the number of pixels whose image intensity is higher than the average value μ_{face} of a facial region in a block. Because μ_{face} is observed for each facial region of sample images, $L_{face}(i)$ is calculated adaptively for different illumination condition. $Ne_{face}(i)$ is the number of pixels in $B_{face}(i)$, whose edge intensity is higher than a settled threshold value T_e. The edge intensity is obtained by Sobel operator, and because it is not severely changed by light conditions, we can define T_e previously as a certain constant.

The difference of our model from others is that QMF employs the ratio of the number of pixels which satisfy qualitative conditions, the *Lightness* and *Edgeness*. Consequently, it is not necessary to consider the difference comming from location change of each pixel within the same block. If a facial deformation is restricted in the same block, the matching result is not influenced by the deformation.

2.2 Detection of Full Face with QMF

In this section, we describe details on the method detecting full face with QMF. Because QMF has qualitative information of frontal face to each block, by comparing the features of input image with those of QMF, we can detect a face from input image. As a criterion, we herein define "*Faceness*" of a region R of input image as follows;

$$F_R = \sum_i Ml_R(i) + \sum_i Me_R(i) \quad (2)$$

Where $Ml_R(i)$ and $Me_R(i)$ indicates the matching result adopting qualitative feature $L_R(i)$ and $E_R(i)$ respectively for the ith block $B_R(i)$, and those are defined as

$$Ml_R(i) = \begin{cases} w_l(i) & \text{if } (L_R(i) - \overline{L_{face}(i)})^2 < \sigma^2(L_{face}(i)) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$Me_R(i) = \begin{cases} w_e(i) & \text{if } (E_R(i) - \overline{E_{face}(i)})^2 < \sigma^2(E_{face}(i)) \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

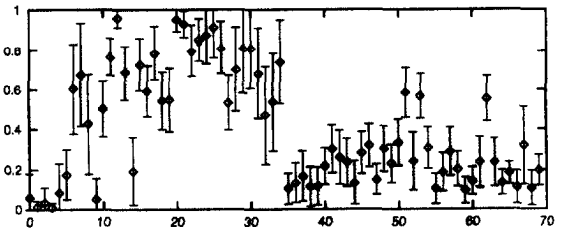

Figure 1: The distribution of *Lightness* (No.0..34) and *Edgeness* (No.35..69). Each mark shows $\overline{L_{face}(i)}$ and $\overline{E_{face}(i)}$, and each line shows $\overline{L_{face}(i)} \pm \sigma(L_{face}(i))$ and $\overline{E_{face}(i)} \pm \sigma(E_{face}(i))$.

In the definitions, $\overline{L_{face}(i)}$ and $\sigma^2(L_{face}(i))$ is the average and variance of $L_{face}(i)$ respectively, and those are estimated from a lot of sampled facial images.

$w_l(i)$ and $w_e(i)$ are weight parameters of *Lightness* and *Edgeness* for $B_{face}(i)$, and they control how $L_R(i)$ and $E_R(i)$ are significant in matching. Consequently, it is very important to determined the values, and hereafter we explain a method to induce them from a lot of sampled facial images.

First of all, we should notice that every block of a facial image has a specific value of *Lightness* and *Edgeness* as shown in Fig.1, and the value of *Lightness* is more fluctuating than that of *Edgeness*. And also in Eq.(2), the *Faceness* is calculated with very simple summation of the two factors which have different properties in fluctuation. From the facts, if we try to determine the weight parameters from the difference of each values in other blocks, namely if we want to improve the defect of unbalance in amount of fluctuations of both parameters, we should fairly treat *Lightness* and *Edgeness* with the same meaning as a principal factor in calculation of *Faceness*. Considering it, we temporally define *peculiarity* of two qualitative factors as follows:

$$\begin{aligned} p_l(i) &= \sum_{j \neq i} (\overline{L_{face}}(j) - \overline{L_{face}}(i))^2 + \\ &\quad \sum_j (\overline{E_{face}}(j) - \overline{L_{face}}(i))^2 \end{aligned} \quad (5)$$

$$\begin{aligned} p_e(i) &= \sum_{j \neq i} (\overline{E_{face}}(j) - \overline{E_{face}}(i))^2 + \\ &\quad \sum_j (\overline{L_{face}}(j) - \overline{E_{face}}(i))^2 \end{aligned} \quad (6)$$

In the equations, the right two terms are introduced to eliminate the defect of unbalance between two features as described above. As the result, the peculiarity is calculated as the total difference of each values from others without distinction of properties; *Lightness* and *Edgeness*.

Using the peculiarities, the values of the weight parameters are obtained by normalizing the peculiarities with total sum of the two factors in the block as shown

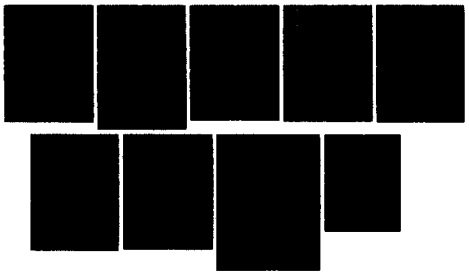

Figure 2: Examples of model images synthesized by texture mapping.

in Eq.(7) and Eq.(8).

$$w_l(i) = p_l(i)/(\sum_j p_l(j) + \sum_j p_e(j)) \quad (7)$$

$$w_e(i) = p_e(i)/(\sum_j p_l(j) + \sum_j p_e(j)). \quad (8)$$

3 Facial Pose Estimation with Synthesized Images

3.1 Synthesis of Model Images

Since face image is apt to be so changeable, simple template matching methods do not work well in face tracking. So we adopt a algorithm using multiple images synthesized by texture mapping and reprojection technique. The procedure of image synthesis is as follows.

- The two dimensional size of the 3-D graphic model in image plane is adjusted to that of the image model M_i. At the time, depth value is also abruptly calculated in proportion to the two dimensional variation.
- Facial region extracted from the initial image is mapped onto the 3-D graphic model. Because of simplicity of calculation, we take the orthographic projection accompanying hidden surface removal.
- Several images are synthesized by reprojecting 3-D image model into image plane while view direction is changed. In this step, to make different model images; there must be a minimum difference among them, minimum disparity S is checked between each other.

Figure 2 shows a set of the model images.

3.2 Pose Estimation

The disparity $D(R(x,y), M_i)$ between ith model image of face M_i and a region $R(x,y)$ in input image I, is defined as the sum of absolute difference of image intensity between $R(x,y)$ and M_i for each pixels.

$$D(R(x,y), M_i) = \sum_{k,l} |I(x+k, y+l) - M_i(k,l)| \quad (9)$$

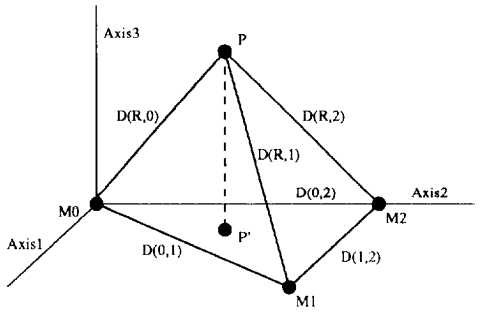

Figure 3: Pose estimation using three model images. See text.

In general, this disparity between two image increase monotonously within certain range, and we call the circumscription "*Permissible Motion Range.*" To keep the reliability of matching result, disparity must be calculated within the range and by introducing a threshold $T(M_i)$, we can ascertain it.

In image sequence, we can track facial region by iterative calculation of disparity between model images M_i and input imagery in neighboring area of the previous facial region. But if the facial motion exceeds the *Permissible Motion Range* for M_i, the facial region of moving face will not be detected from the image with M_i itself. In general, detecting such region is very difficult because there is no absolute way to forecast facial motion. From the reason, we have proposed a algorithm using synthesized multiple images[2]. However, to implement it in real-time, here we newly introduce a modified one in which nearly one hundred model images are adopted.

In the case, recognition results become very unstable near boundary of two model images, therefore, we take a idea to use three model images which have three lowest disparity in template matching procedure, as shown in Fig.3.

Assume a three dimensional space which can be defined by disparity between model images. We call it *Disparity Space*. In the space, three model images are separated by distances determined in model synthesis procedure. So, using the three position of model images as reference points, we can estimate facial pose robustly as follows.

- At first, disparities between a region $R(M_i)$ and every model image $M(i)$ are calculated.
- Select three model images which have the three least disparities and located them in disparity space as in Fig.3.
- Determine the position p of input facial image by referencing the three disparities between the three model images.
- Calculate a position p' by projecting the position p onto model image plane.
- The model which is closest to the position p' is selected as a matched one with $R(M_i)$.

- If the position is within *Permissible Motion Region* of any model, the pose of the model is estimate as correct one, else the pose estimated in previous sequence is adopted as recognition result.

4 Hardware System

4.1 ReMOT: Real-Time Moving Object Tracker

As described above, we have adopted a very simple algorithm calculating correlation between two images for pose estimation. For the algorithm, we have developed a special hardware, ReMOT, which can calculate the disparity defined in Eq.(9) very fast. Figure 4 shows the simple block diagram of the ReMOT.

ReMOT emploies a special VLSI, MEP(Motion Estimation Processor:STI3220), which can calculate motion vector of a small block image in a search image[7]. However, it is not convenient to manage large image at once, so we have constructed more flexible and general system to calculate disparity from large images. Also, this system equips three real-time DSPs(Digital Signal Processors), so that many image processing algorithm can be implemented on this system. ReMOT consists of five blocks: image I/O, VME interface, frame buffering, early processing and switching, and motion detection blocks.

In image I/O block, two video camera and one monitor can be connected. Video signal is grabbed here, and visual output is converted into NTSC composite signal for display. Also image overlapping facility is available to superimpose output image with some appropriate demands.

To have much variety in processing ability, ReMOT is designed based on standard VME-Bus interface. All devices of the system is mapped into 1Mbyte of VME-Bus space and the window is switched freely in a 16Mbyte memory space. At most, 256 boards can be connected to one VME system. As the system has standard VME-Bus interface, any type of computer is available as host computer even if it has a VME interface board.

In frame buffering block, dual port VRAMs are equipped and those store four pieces of image (the size is 512×512×8bit). This memory space is used as temporary memory for image processing or as sequential stacks for pipe line processes.

In early processing block, we can use three DSPs for early image processing such as edge extraction, binarization, thinning, and convolution of a image with small size , and etc. The three DSPs are dynamically connected to each other through switching circuit to perform a pipe line processing. Every blocks are connected to switching circuit which is realized programable logic devices. So that, processing stream can be changed even in operation without any loss of real-time performance.

In the motion detection blocks, four MEPs are equipped for parallel processing. And image memories for search and reference blocks are implemented with very high speed SRAMs. And also, to follow scale change of moving objects, a flexible switching circuit, which can change the sampling rate of search and reference blocks, is prepared. The recognition results; motion vector of reference block and minimum disparity of the block, are outputd to FIFO memory and then host computer, SUN workstation in our case, pick up the data through a VME interface.

4.2 Realization of Network System

To implement a real-time interaction system utilizing our recognition algorithm described above with ReMOT, we have connected several computers through Ethernet, and the system synthesize some sounds and images synchronized to facial motion. In this section, we describe the system briefly according to functional groups.

- Image Recognition Group:
  Real-time template matching results are obtained from this block. The block is composed by a video camera, a VME-Bus system with two boards of ReMOT and a host computer.

- Image Synthesis Group:
  In making model images, we have used graphic workstation IRIS Crimson VGX which has real-time texture mapping board. With this computer, a initial face image is also extracted with QMF, then appropriate number of model images are synthesized and exported to the host computer of image recognition block. And also artificial facial images are synthesized by using recognition results as output of the graphic workstation.

- Audio Synthesis Group:
  In this block, the recognition result, the pose and position of a face is applied to control a electronic synthesizer. In fact, this block is fundamentally not related to any of recognition system, however, as one of experimental applications, it have been included. A personal computer PC9801FA is involved as server of audio synthesis group, and it generate standard MIDI(Musical Instrument Digital Interface) signals and voice of human. A electronic musical synthesizer convert the MIDI signal into some interesting sounds.

5 Experiment

Making QMF, we have utilized 152 sampled images of 22 different persons, in which face appears with various facial expressions, light conditions, and arbitrary size. To exclude the scale difference among the images, we moderate the size of 5 × 7 blocks in QMF to each facial region of the images, which is inclusive of whole face, and take correspondence of eyes and mouth between QMF and a face. And then, the *Lightness* and *Edgeness* are calculate as defined in Eq.(9) for each block in every image. In this experiment, we take 128 as T_e. And finally, we obtained the matching parameters of qualitative features and weight parameters in Eq.(4), which are shown in Fig.1.

We have used 36 test images in the face detection experiment. These test images include faces of 27 persons, and they are different from the sample images

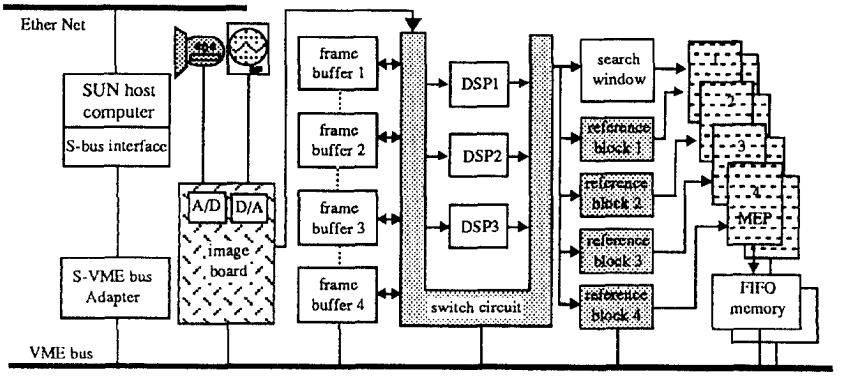

Figure 4: The simplified block diagram of ReMOT(see text).

used for creating QMF. In the experiment, the *Faceness* defined by Eq.(2) is calculated everywhere in each test image, with changing block size of the QMF from 3×3 pixels to 16×16 pixels. Twenty regions which have the highest *Faceness* is detected from each test image as candidates, and we checked each candidate region whether it really includes a face.

From 24 test images (66.7%), we have obtained correct facial regions with only the highest *Faceness* ,and from 35 test images (97.2%), we could find a facial region considering 20 candidate regions. We have failed in detecting face from only one test images and it has a face of bald-headed man. These type of facial patterns are not adequate to the QMF because blocks placed on hair regions have higher peculiarity, and those contribute much in calculation of faceness. We think that it is needed to prepare specific models for such delicate faces.

In the real-time tracking and pose estimation experiment, the system works very well even though subject person and illumination condition are changed. At first, the initial image is sent to image synthesis group, then a facial region is extracted by using QMF. And then, about one hundred model images are synthesized and exported to the host computer of ReMOT. And then, the system starts to track facial motion and output recognition results by synthesized sounds. To enter the tracking loop, it takes about 2 seconds including communication time on Ethernet and we think that is sufficient as initial cost.

6 Conclusion

In this paper, we have proposed a new algorithm estimating pose with multiple model images, qualitative model used for detecting the initial image model, and a hardware system implementing the algorithm in real-time. The qualitative model is defined by statistical data of brightness of facial images. Using this model, we can detect frontal face of various individuals.

This method has been presented as one approach of image-based vision, using multiple model images synthesized by texture mapping and reprojection technique. It contrasts well with conventional vision approach since very simple estimation method, correlation, is adopted to a lot of model images rather than very difficult and complicated theory based on feature extraction.

References

[1] A.Tsukamoto, C.W.Lee and S.Tsuji, "Detection and Tracking of Human Face with Synthesized Templates", *ACCV'93*, pp.183-186, 1993.

[2] A.Tsukamoto, C.W.Lee and S.Tsuji, "Motion Estimation of Human Face with Multiple Model Images", *The Transactions of IEICE*, vol.j77-D-II, No.8 pp.1582-1590, 1994.8 (Japanese)

[3] M.A.Turk and A.Pentland, "Eigenfaces for Recognition", *J. Cognitive Neurosci.*, vol.3, no.1, 1991.

[4] H.Zabrodsky, S.Peleg and D.Avnir, "Hierarchical Symmetry", *ICPR'92*, vol.III, pp.9-12, 1992.

[5] G.Yang and T.S.Huang, "Human Face Detection in a Scene", *CVPR'93*, pp.453-458, 1993.

[6] R.Brunelli and T.Poggio, "Face Recognition: Features versus Templates", *IEEE Trans. Pattern Anal. & Mach. Intell.*, vol.15, no.10, 1993.

[7] Image Processing Data Book(2nd Ed.), SGS-THOMSON Microelectronics, 1992, Italy