Visual Tracking for Multimodal Human Computer Interaction

Jie Yang, Rainer Stiefelhagen, Uwe Meier, Alex Waibel
Interactive Systems Laboratory
Carnegie Mellon University
Pittsburgh, PA 15213 USA
{yang+, stiefel, uwem, waibel}@cs.cmu.edu

ABSTRACT

In this paper, we present visual tracking techniques for multimodal human computer interaction. First, we discuss techniques for tracking human faces in which human skin-color is used as a major feature. An adaptive stochastic model has been developed to characterize the skin-color distributions. Based on the maximum likelihood method, the model parameters can be adapted for different people and different lighting conditions. The feasibility of the model has been demonstrated by the development of a real-time face tracker. The system has achieved a rate of 30+ frames/second using a low-end workstation with a framegrabber and a camera. We also present a top-down approach for tracking facial features such as eyes, nostrils, and lip corners. These real-time visual tracking techniques have been successfully applied to many applications such as gaze tracking, and lip-reading. The face tracker has been combined with a microphone array for extracting speech signal from a specific person. The gaze tracker has been combined with a speech recognizer in a multimodal interface for controlling a panoramic image viewer.

Keywords

Visual tracking, multimodal human computer interaction, skin-color modeling, face tracking, gaze tracking, lip-reading, sound localization

INTRODUCTION

While multimodal interfaces offer greater flexibility and robustness than traditional mouse/keyboard interfaces, they have been largely pen/voice-based, user activated, and operated in settings where some constraining devices are required. For truly effective and unobtrusive multimodal human-computer interaction, we envision systems that allow for freedom of movement in a possibly noisy room without the need for intrusive devices such as headsets and close-talking microphones. In order to make this goal a reality, we require not only efficient ways to integrate multiple modalities but also a better model of the human user based on a mixture of verbal and non-verbal, acoustic and visual cues. A visual tracking system can provide much useful information about users for computer systems. Using visual information and combining it with other information, it is possible to identify the message source, message target, and extract the message content. For example, a system can locate a user by merging visual face tracking algorithms and acoustic sound source localization, identify who is talking to whom by extracting head orientation and eye gaze, and extract message content by visual and acoustic speech recognition.

In this paper, we present visual tracking techniques for multimodal human computer interaction. First, we discuss techniques of tracking human faces. A human face provides a variety of different communicative functions such as identification, the perception of emotional expressions, and lip-reading. Many applications in human computer interaction require tracking a human face. Human skin-colors can be used as a major feature for tracking human faces. An adaptive stochastic model has been developed to characterize the skin-color distributions. Based on the maximum likelihood method, the model parameters can be adapted for different people and different lighting conditions. The feasibility of the model has been demonstrated by the development of a real-time face tracker. The system has achieved a rate of 30+ frames/second using a low-end workstation (e.g., HP9000) with a framegrabber and a camera. Once a face is located, it is much easier to locate the facial features such as eyes, nostrils, and lips. This top-down approach works very well for many applications such as gaze tracking, and lip-reading. We describe some applications of the visual tracking techniques to multimodal human computer interaction. The face tracker has been combined with a microphone array for extracting speech signal from a specific person. The gaze tracker has been combined with a speech recognizer in a multimodal interface to control a panoramic image viewer.

TRACKING FACES IN REAL-TIME

Locating and tracking human faces is a prerequisite for face recognition and/or facial expressions analysis, although it is often assumed that a normalized face image is available. In order to locate a human face, the system needs to capture an image using a camera and a framegrabber, process the image, search for important features in the image, and then use these features to determine the location of the face. In order to track a human face, the system not only needs to locate a face, but also needs to find the same face in a sequence of images.

Facial features, such as the eyes, nose and mouth, are natural candidates for locating human faces. These features, however, may change from time to time. Occlusion and non-rigidity are basic problems with these features. Four basic techniques are commonly used for dealing with feature variations: correlation templates (Brunelli and Poggio, 1993; Pentland et al., 1994), deformable templates (Yuille et al. 1992), spatial image invariants (Sinha, 1994), and neural networks (Sung and Poggio, 1994; Rowley et al., 1995). These methods are computation expensive and hardly achieve real-time performance. Several systems of locating the human face have been reported. Eigenfaces, obtained by performing a principal component analysis on a set of faces, have been used to identify faces (Turk and Pentland, 1991). Sung and Poggio (1994) reported a face detection system based on clustering techniques. The system passes a small window over all portions of the image, and determines whether a face exists in each window. A similar system with better results has been claimed by Rowley et al. (1995).

A different approach for locating and tracking faces using skin-colors is described in (Hunke and Waibel, 1994; Chang et al., 1994; Yang and Waibel, 1995; Oliver et al. 1997). Color has been long used for recognition and segmentation (Ohta et al., 1980;Swain and Ballard, 1991). Using skin-color as a feature for tracking a face has several advantages. Processing color is much faster than processing other facial features. Under certain lighting conditions, color is orientation invariant. This property makes motion estimation much easier because only a translation model is needed for motion estimation. However, color is not a physical phenomenon. It is a perceptual phenomenon that is related to the spectral characteristics of electro-magnetic radiation in the visible wavelengths striking the retina (Wyszecki and Styles, 1982). Thus, tracking human faces using color as a feature has several problems. First, different cameras may generate different colors even for the same person under the same lighting condition. Second, different people have different color appearances. Finally, the color appearance of the same person may differ under different environmental conditions. In order to use color as a feature for face tracking, we have to deal with these problems.

Skin Color Modeling

Color is the perceptual result of light in the visible region of the spectrum incident upon the retina. Physical power (or radiance) is expressed in a spectral power distribution. Much research has been directed to understanding and making use of color information. The human retina has three different types of color photoreceptor cone cells, which respond to incident radiation with somewhat different spectral response curves. Based on the human color perceptual system, three numerical components are necessary and sufficient to describe a color, provided that appropriate spectral weighting functions are used.

In order to use skin color as a feature, we first have to characterize skin colors. Color can be characterized by a non-parametric model such as a color map, or a parametric model such as a distribution model. We are interested in developing a distribution model for representing human skin color distributions. The general procedure for developing a distribution model includes finding clusters, extracting features (dimensionality reduction), and determining a distribution. In order to investigate all these problems, we need a large amount of data. We have built up a database which contains about 1000 face images downloaded from the Internet and taken from our laboratory. This database covers different races (Caucasian, Asian, African American) and different lighting conditions.

Skin Color Cluster

A color histogram is a distribution of colors in the color space and has long been used by the computer vision community in image understanding. For example, analysis of color histograms has been a key tool in applying physics-based models to computer vision. In the mid-1980s, it was recognized that the color histogram for a single inhomogeneous surface with highlights will have a planar distribution in color space (Shafer, 1984). It has since been shown that the colors do not fall randomly in a plane, but form clusters at specific points (Klinker et al., 1987). The histograms of human skin color coincide with these observations. Figure 1 shows a face image and the skin-color occurrences in the RGB color space (256x256x256). The skin-colors are clustered in a small area in the RGB color space, i.e., only a few of all possible colors actually occur in a human face.

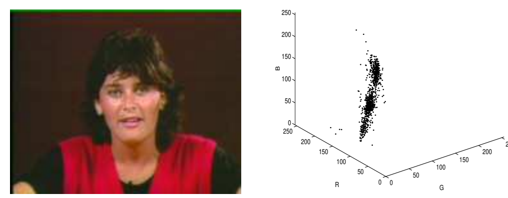


Figure 1. An example of a face image and the skin-color occurrences in the RGB space

Dimensionality Reduction

It is well known that different people have different skin-color appearances. Even for the same person, his/her skin-color appearance will be different in a different environment. In other words, many factors contribute to human skin-color appearance. In order to characterize skin-color, we hope to find a color space in which skin-colors are less variant.

For human color perception, a 3D color space such as an RGB space, is essential for describing a true color. However, a 3D space is not necessarily essential for all other problems. In the problem of tracking human faces, brightness is not important. Therefore we can remove it from the original information by normalization. Our experiments reveal that human color appearances differ more in brightness than in color itself. If we can remove the brightness from the color representation, the difference among human skin-colors can be greatly reduced. In fact, a triple $[r, g, b]$ in the RGB space represents not only color but also brightness. If the corresponding elements in two points, $[r_1, g_1, b_1]$ and $[r_2, g_2, b_2]$, are proportional, i.e.,

$$\frac{r_1}{r_2} = \frac{g_1}{g_2} = \frac{b_1}{b_2}. \quad (1)$$

They have the same color but different brightness. They can be mapped onto the same point through the color normalization (Wyszecki and Styles, 1982):

$$r = R / (R + G + B), \quad (2)$$

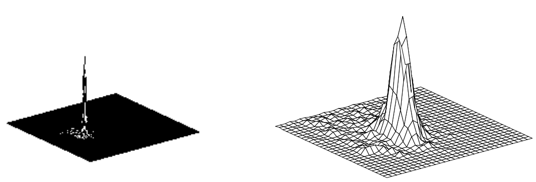
$$g = G / (R + G + B). \quad (3)$$

In fact, (2) and (3) define a $\mathbf{R}^3 \rightarrow \mathbf{R}^2$ mapping. the color blue is redundant after the normalization because $r+g+b=1$.

Table 1: Comparison of mean and variance

	RGB Space	rg Space
Mean	m_R = 234.29 m_G = 185.72 M_B = 151.11	m_r = 94.22 m_g = 81.59
Variance	σ_R = 26.77 σ_G = 30.41 σ_B = 25.68	σ_r = 4.93 σ_g = 3.89

Another advantage of the color normalization is, we found out, that the color variance can be greatly reduced after the normalization. The same skin color cluster has a smaller variance in the normalized color space than that in an RGB space. Skin-colors of different people are less variant in the normalized color space. This result is significant because it provides evidence of the possibility of modeling human faces with different color appearances in the chromatic color space. Table 1 shows mean values and variances of the same skin color cluster in different color spaces. Obviously, the variances are much smaller in the normalized color space.



(a) Global view (b) Local view

Figure 2. Skin-color distribution of the image in Figure 1 in the normalized color space

Skin Color Distribution

We have so far revealed that human skin-colors cluster in the color space and are less variant in the chromatic color space. We are further interested in the representation of the skin-color distributions. Since we are investigating the skin-color distributions in a bivariate normalized color space, it is convenient to examine them graphically. Figure 2 shows the skin color distribution of the image in Figure 1. We have found that the shape of the skin-color distribution of a person remains similar although there is a shift in the distribution under changing lighting conditions. By closely investigating the face color cluster, we have discovered that the distribution has a regular shape. By comparing the shape of skin-color distributions with a bivariate normal distribution, we conclude that it is possible to use a bivariate normal distribution to characterize the skin-color distributions.

Goodness-of-fit Tests

Unlike most of the methods used in engineering statistics which assume a normal distribution of the measured data, we have examined whether the measured data of a sample do indeed have a normal distribution by goodness-of-fit techniques (Yang et al., 1997). Goodness-of-fit techniques examine how well a set of sample data matches with a given distribution as its population. The methods of performing a goodness-of-fit test can be an analytic or graphic approach. In the graphic approach, the most common method is Q-Q plot. We use this method to test our skin-color distributions.

The basic idea of the Q-Q plot is to use the cumulative probability of the sampling data against that of the tested distribution. A straight line indicates that we cannot reject the null hypothesis. We have tested marginal distributions and bivariate distribution. When we do marginal test, we test each variable separately against the normal distribution. When we test the bivariate distribution, we test the transformed variable against Chi-square distribution. We have built up a database which contains about 1000 face images down-loaded from the Internet and taken from our laboratory. This database covers face images of people in different races (Caucasian, African American, and Asian), genders, and the lighting conditions. Using this database, we tested the following NULL hypothesis: *human skin-color is normally distributed in a normalized bivariate space*. The results indicate that we cannot reject the null hypothesis (Yang et al., 1997).

Skin Color Adaptation

Although under a certain environment the skin-color distribution of each individual is a multivariate normal distribution, the parameters of the distribution for different people and different lighting conditions are different. A number of viewing factors, such as light sources, background colors, luminance levels, and media, impact greatly on the change in color appearance of an image. Most color-based systems are sensitive to changes in viewing environment. Even under the same lighting conditions, background colors such as colored clothes may influence skin-color appearance. Furthermore, if a person is moving, the apparent skin-colors change as the person's position relative to camera or light changes. Therefore, the ability of handling lighting changes is the key to success for a skin-color model.

There are two schools of philosophy to handle environment changes: tolerating and adapting. The most common approach for tolerating lighting changes is Color constancy. Color constancy refers to the ability to identify a surface as having the same color under considerably different viewing conditions. Although human beings have such ability, the underlying mechanism is still unclear. A few color constancy theories have demonstrated success on real images (Forsyth, 1990). On the other hand, the adaptive approach is to transform the previous developed color model into the new environment, Because the Gaussian model only has a few parameters, it is possible to update them in real-time. One way to adapt these parameters is to use a linear combination of the known parameters to predict the new parameters. The underlying theory is that a linear combination of Gaussian distributions is still a Gaussian distribution.

$$\hat{\mu} = \sum_{i=0}^{r} \alpha_i m_i , \tag{4}$$

$$\hat{\Sigma} = \sum_{i=0}^{r} \beta_i S_i , \tag{5}$$

Where $\hat{\mu}$ and $\hat{\Sigma}$ are updated mean and covariance, m and S are the previous mean and covariance, α and β are coefficients.

Based on the identification of the skin-color distribution at each sampling point, we can obtain its mean vector and covariance matrix. Then the problem becomes an optimization problem. We can use the maximum likelihood criterion to obtain the best set of coefficients for the prediction. We have investigated adapting the mean only, and adapting both the mean and covariance matrix (Yang et al., 1997).

Adapting Mean

In this case, the covariance matrix is assumed to be a constant and the mean vector μ is assumed to be a linear combination of the previous mean vectors:

$$\hat{\mu} = \sum_{i=0}^{r} \alpha_i m_i , \qquad \hat{\Sigma} = \Sigma \tag{6}$$

By setting the derivatives of the likelihood function with respect to α to 0, we can obtain linear equations for solving α:

$$\sum_{k=1}^{r} m_j' \Sigma^{-1} m_k \hat{\alpha}_k = m_j' \Sigma^{-1} \bar{x} , \ \text{j = 1... r} \tag{7}$$

Adapting mean and Covariance

In this case, the both mean vector covariance matrix are assumed to be a linear combination:

$$\hat{\mu} = \sum_{i=0}^{r} \alpha_i m_i \quad , \qquad \hat{\Sigma} = \sum_{i=0}^{r} \beta_i S_i \tag{8}$$

In this case, there is no simple analytic solution available. An EM algorithm has been used to iteratively estimate parameters (Yang et al., 1997):

Algorithm

1. Initialization

$$\sum_{k=1}^{r} m_j' m_k \hat{\alpha}_k^{(0)} = m_j' \bar{x} \quad , \hat{\mu}_j^{(0)} = \sum_{k=0}^{r} \alpha_k^{(0)} m_k$$

$$C^{(0)} = \frac{1}{N} \sum_{k=1}^{N} (x_k - \bar{x})(x_k - \bar{x})' + \left(x_k - \hat{\mu}^{(0)}\right)\left(x_k - \hat{\mu}^{(0)}\right)'$$

$$\sum_{k=1}^{r} trS_j S_k \hat{\beta}_k^{(0)} = trS_j C^{(0)} , \ \hat{\Sigma}^{(0)} = \sum_{i=0}^{r} \hat{\beta}_k^{(0)} S_k$$

2. Iteration

$$\sum_{k=1}^{r} m_j' \Sigma^{-1} m_k \hat{\alpha}_k^{(i)} = m_j' \Sigma^{-1} \bar{x} , \ \hat{\mu}_j^{(i)} = \sum_{k=0}^{r} \alpha_k^{(i)} m_k$$

$$C^{(i)} = \frac{1}{N} \sum_{k=1}^{N} (x_k - \bar{x})(x_k - \bar{x})' + \left(x_k - \hat{\mu}^{(i)}\right)\left(x_k - \hat{\mu}^{(i)}\right)'$$

$$\sum_{k=1}^{r} trS_j S_k \hat{\beta}_k^{(i)} = trS_j C^{(i)} , \ \hat{\Sigma}^{(i)} = \sum_{i=0}^{r} \hat{\beta}_k^{(i)} S_k$$

3. If max ($|\beta_j(i) - \beta_j(i-1)|$) < ε for a small number $\varepsilon > 0$, stop; otherwise continue

Tracking Human Face in Real-time

A direct application of the skin-color model is to locate a face in an image. A straightforward way to locate a face is to match the model with the input image to find the face color clusters. Each pixel of the original image is converted into the chromatic color space and then compared with the distribution of the skin-color model. Since the skin colors occur in a small area of the chromatic color space, the matching process is very fast. This is useful for real-time face tracking. By combining the adaptive skin color model with the motion model and the camera model, we have developed a real-time face tracker (Yang and Waibel, 1995). The system has achieved a rate of 30+ frames/second with 305 x 229 input sequences of images on both HP and Alpha workstations. The system can track a person's face while the person walks, jumps, sits and rises. The QuickTime movies of demo sequences in different situations and on different subjects can be found on the web site http://www.is.cs.cmu.edu/ISL.multimodal.face.html.

Application to Tele-conference

An immediate application of the face tracker is to use it to automatically follow the speaker in a tele-conference. We describe a more interesting application in this subsection.

In a tele-conference, the quality of the conference greatly depends on image transmission. The bottle neck of the traffic is in the network. People have been working very hard on data compression techniques to reduce data transmission. However, there is a limitation on compression. In such a case, if we want to continue the conference, we have to discard some data. One way to do this is to skip some frames, which may result in losing important information. We want to keep the important information and discard relative unimportant data. To achieve this goal, we can add a selective function on the top of original codec to select the important information. In a tele-conference, the speaker is the center, We would like to keep updating speaker's information. Then, we could use the face tracker to track the speaker and select the region surrounding the facial area by a window. The window size is adjustable based on network bandwidth. When network traffic is good, the window is the entire image. When the network bandwidth is not enough, the window size is shrunk, and even the image is converted to grey scale. We have developed a system (Yang et al., 1996) by adding the face tracker on the top of vic, a public domain available tele-conference software. The system can provide several filtering schemes such pseudo-cropping, slicing, and blurring. Figure 3 shows how these filtering schemes work.

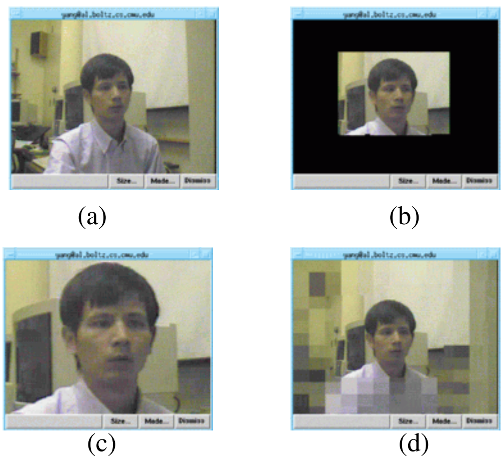


Figure 3. Different filtering schemes: (a) original; (b) pseudo-cropping; (c) slicing; (d) blurring

SEARCHING AND TRACKING FACIAL FEATURES

The face tracker can also be used as a base for other applications. Once a face is located, it is much easier to locate other features such as pupils, lips and nostrils. This top-down approach works very well for many applications. In this section, we show how to track these facial features in real-time and use them to estimate human gaze direction (Stiefelhagen et al., 1996).

Locating Facial Features

We first describe methods to locate and track the pupils, the lip corners and the nostrils within a found face.

Searching For Pupils

Assuming a frontal view of the face initially, we can search for the pupils by looking for two dark regions that satisfy certain geometric constraints and lie within a certain area of the face. For a given situation, these dark regions can be located by applying a fixed threshold to the gray-scale image. However, the threshold value may change for different people and lighting conditions. To use the thresholding method under changing lighting conditions, we developed an iterative thresholding algorithm. The algorithm iteratively thresholds the image until a pair of regions that satisfies the geometric constraints can be found.

Figure 4 shows the iterative thresholding of the search window for the eyes with thresholds k_i. After three iterations, both pupils are found.

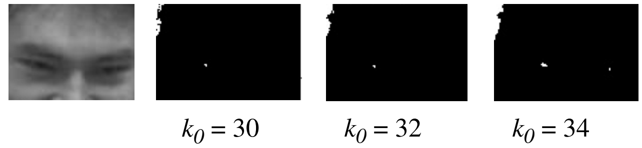


Figure 4. Iterative thresholding of the eye region

Searching For Lip Corners

First, the approximate positions of the lip corners are predicted, using the positions of the eyes, the face-model and the assumption, that we have a near-frontal view. A generously big area around those points is extracted and used for further search.

Finding the vertical position of the line between the lips is done by using a horizontal integral projection P_h of the grey-scale-image in the search-region. Because the lip line is the darkest horizontally extended structure in the search area, its vertical position can be located where P_h has its global minimum. The horizontal boundaries of the lips can be found by applying a horizontal edge detector to the refined search area and regarding the vertical integral projection of this horizontal edge image. The positions of the lip corners can be found by looking for the darkest pixel along the two columns in the search area located at the horizontal boundaries.

Searching For Nostrils

Similarly to searching for the eyes, the nostrils can be found by searching for two dark regions, that satisfy certain geometric constraints. Here the search-region is restricted to an area below the eyes and above the lips. Again, iterative thresholding is used to find a pair of legal dark regions, that are considered as the nostrils.

Tracking Facial Features

Once the facial features are located, the problems become tracking those features.

Tracking Eyes

To track the eyes, simple darkest pixel finding in the predicted search-windows around the last positions is used.

Tracking Lip Corners

Our approach to track the lip-corners consists of the following steps:

1. Search for the darkest pixel in a search-region right of the predicted position of the left corner and left of the predicted position of the right corner. The found points will lie on the line between the lips.

2. Search for the darkest path along the lip-line for a certain distance *d* to the left and right respectively, and choose positions with maximum contrast along the search-path as lip-corners.

Because the shadow between the upper and lower lip is the darkest region in the lip-area, the search for the darkest pixel in the search windows near the predicted lip corners ensures that even with a bad prediction of the lip corners, a point on the line between the lips is found. Then the true positions of the lip corners can be found in the next step. Figure 5 shows the two search windows for the points on the line between the lips. The two white lines mark the search paths along the darkest paths, starting from where the darkest pixel in the search windows have been found. The found corners are marked with small boxes.

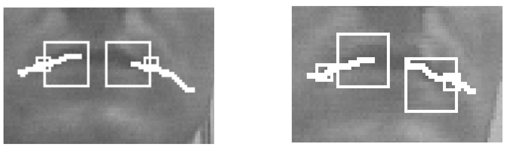


Figure 5. Search for lip corners along the line between the lips

Tracking Nostrils

Tracking the nostrils is also done by iteratively thresholding the search-region and looking for 'legal' blobs. But whereas we have to search for a relatively large area in the initial search, during tracking, the search-window can be positioned around the previous positions of the nostrils, and can be much smaller. Furthermore, the initial threshold can be initialized with a value that is a little lower than the intensity of the nostrils in the previous frame. This limits the number of necessary iterations to a small value.

However, both nostrils are not always visible in the image. For example, when the head is rotated strongly to the right, the right nostril will disappear, and only the left one will remain visible. To deal with this problem, the search for two nostrils is done only for a certain number of iterations. If no nostril-pair is found, then only one nostril is searched for by looking for the darkest pixel in the search window for the nostrils.

To decide which of the two nostrils was found, we choose the nostril, that leads to the pose which implies smoother motion of the head compared to the pose obtained choosing the other nostril.

Gaze Tracking

The locations of facial features can be used to estimate the person's gaze direction, or head pose, using a 3D model (Stiefelhagen et al., 1996).

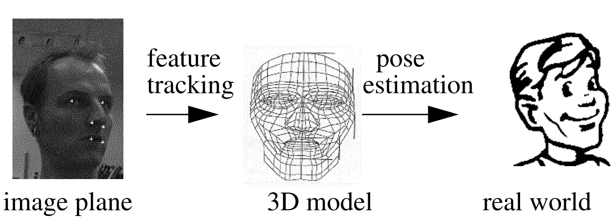


Figure 6. Gaze tracking as feature tracking + pose estimation

The basic idea is to estimate the head pose by finding correspondences between a number of head model points and their locations in the camera image as shown in Figure 6. DeMenthon & Davis (DeMenthon and Davis, 1992) developed an algorithm for estimating 3D pose of an object using as few as 4 correspondences of non-coplanar points. Since we can locate and track six non-coplanar facial features (eyes, lips and nostrils) we can compute the head pose. The current system has achieved a rate of 15 frames/second on a low-end work station (Stiefelhagen et al., 1996).

APPLICATIONS TO MULTIMODAL HCI

Human-human communication takes advantage of many communication channels. We use verbal and non-verbal channels. We speak, point, gesture, write, use facial expressions, head motion, and eye contact. However, most of current multimodal human computer interfaces have been focused on integration of speech, handwriting and pen gestures. In fact, visual information can play an important role in multimodal human computer interaction. We present three examples of multimodal interfaces that include visual information as a modality in this section.

Lip-reading

It is well known that hearing impaired listeners and listening in adverse acoustic environments rely heavily on visual input to disambiguate among acoustically confusable speech elements. It has been demonstrated that visual information can enhance the accuracy of speech recognition. However, many current lip-reading systems require users to keep still or put special marks on their faces. We have developed a lip-reading system based on the face tracker. The system first locates the face and then extracts the lip regions as shown in Figure 7.

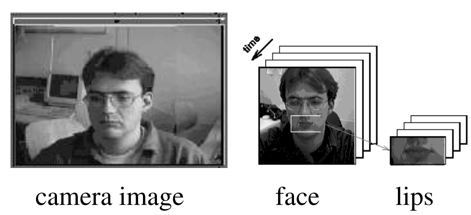


Figure 7. Basic idea of lip tracking

System Description

The system is based on a modular MS-TDNN (Multi-State Time Delay Neural Network) structure (Meier et al. 1996). The visual and acoustic TDNNs are trained separately, and visual and acoustic information are combined at the phonetic level. The system has been applied to the task of speaker-dependent continuous spelling German letters. Letter sequences of arbitrary length and content are spelled without pauses. Words in our database are 8 letters long on average. The task is thus equivalent to continuous recognition with small but highly confusable vocabulary.

Through the first three layers (input-hidden-phoneme/viseme) the acoustic and visual inputs are processed separately. The third layer produces activations for 62 phoneme or 42 viseme states for acoustic and visual data, respectively. A viseme, the rough visual correlate of a phoneme, is the smallest visually distinguishable unit of speech. Weighted sums of the phoneme and corresponding viseme activations are entered in the combined layer and a one stage DTW algorithm finds the optimal path through the combined states that decodes the recognized letter sequence. The weights in the parallel networks are trained by backpropagation. There are 15 hidden units in both subnets. The combination weights are computed dynamically during recognition to reflect the estimated reliability of each modality:

$$hyp_{AV} = \lambda_V hyp_V + \lambda_A hyp_A \tag{9}$$

$$\lambda_A = b + \frac{S_V - S_A}{\Delta S_{MaxOverData}} \tag{10}$$

$$\lambda_V = 1 - \lambda_A \tag{11}$$

The entropy quantities S_A and S_V are computed for the acoustic and visual activations by normalizing these to sum to one (over all phonemes or visemes, respectively) and treating them as probability mass functions. High entropy is found when activations are evenly spread over the units which indicates high ambiguity of the decision from that particular modality. The bias *b* pre-skews the weight in favor of one of the modalities. This bias is set depending on the signal-to-noise ratio (SNR). The quality of the speech data is generally well described by the SNR. Higher SNR means higher quality of the acoustic data and therefore the consideration of the acoustic side should increase for higher SNR-values and decrease for smaller ones. We used a piecewise-linear mapping to adjust bias b as a function of the SNR (Meier et al. 1996).

The system uses the gray-scale images of the lip region as inputs. Adaptive gray value modification is used to eliminate different lightning conditions (Meier et al. 1997). For acoustic preprocessing 16 melscale coefficients are used.

Table 2: Speaker Dependent Results

Test Set	visual only	acoustic only	combined
clean	55%	98.4%	99.5%
16dB SNR	55%	56.9%	73.4%
8 dB SNR	55%	36.2%	66.5%

Experimental Results

We have trained a speaker dependent recognizer on 170 sequences of acoustic/visual data, and tested on 30 sequences. For testing we also added white noise to the test set. The results are shown in table 2, as performance measure word accuracy is used (where a spelled letter is considered a word).

With our system we get an error reduction up to 50% compared with the acoustic recognition rate.

Panoramic Image Viewer

A panoramic image provides a wide angle view of a scene. In order to view a 360 degree panoramic image, we need to use a special viewer. In a panoramic image viewer, we need to control three parameters: pan, tilt, and zoom. The current interface uses keyboard and mouse to control them. Alternatively, we could control them by changing the positions of the viewing point., e.g., moving towards left and right, forward and backward, up and down. But it is more natural to control the panning and tilting with the gaze, and the zooming with the voice. We have developed an interface to control a panoramic image viewer by combining the gaze tracker with a speech recognizer (Stiefelhagen and Yang, 1997). With such an interface, a user can fully control the panoramic image viewer without using his/her hands. The user can scroll through the panoramic images by looking to the left and right or up and down, and he can control the zoom by speaking the commands “zoom in,” “zoom in two times,” “zoom out” “zoom out five times,” etc. Figure 8 shows how the system works.

Figure 8. Gaze-voice controlled panoramic image viewer

Beamforming by Face Tracking

A one-dimensional microphone array allows the speech signal to be received in the half plane in front of the array. If the array is steered towards a given spot the differences of sound arrival time between the microphones are compensated for waves originating exactly from this location. By summing these aligned signals one achieves an enhancement of the desired signal while sound coming from other locations is not in the same phase and thus its audibility is deteriorated. On the other hand, if the system knows the speaker’s location from visual tracking, it is possible to form a beam to select the desired sound source to enhance the quality of speech signal for speech recognition. We have demonstrated that a more accurate localization in space can be delivered visually than acoustically. Given a reliable fix, beamforming substantially improves recognition accuracy (Bub et al., 1995). Figure 9 shows the setup of the system.

Figure 9. Setup of microphone array and face tracker

CONCLUSION

In this paper we have described real-time visual tracking techniques and their applications to multimodal human computer interaction. We described how to track human faces and features in real-time. We demonstrated that systems that combine visual information other communication modalities into interfaces that are robust, flexible, and intuitive to use.

ACKNOWLEDGEMENTS

Support for this work has come from the NSF, under contract CDA-9726363, and the DARPA, under contracts N00014-93-1-0806 and N6601-97-C8553.

REFERENCES

Brunelli, R., and Poggio, T. (1993), "Face recognition: features versus templates," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, No. 10, pp. 1042-1052.

Bub, U., Hunke, M., and Waibel, A. (1995), "Knowing who to listen to in speech recognition: visually guided beamforming," Proceedings of ICASSP'95.

Chang, T.C., Huang, T.S., and Novak, C. (1994), "Facial feature extraction from color images," Proc. the 12th IAPR International Conference on Pattern Recognition, Vol. 2, pp. 39-43.

DeMenthon, Daniel. F. and Davis, Larry S. (1992), "Model based object pose in 25 lines of code", Proceedings of Second European Conference on Computer Vision, Santa Margherita Ligure, pp. 335 - 343.

Forsyth, D. (1990), "A novel algorithm for color constancy," International Journal of Computer Vision. Vol. 5, No. 1, pp.5-36.

Hunke, M., and Waibel, A. (1994), "Face locating and tracking for human-computer interaction," Proc. Twenty-Eight Asilomar Conference on Signals, Systems & Computers, Monterey, CA, USA.

Klinker, G.J., Shafer, S.A., and Kanade, T. (1987), "Using a color reflection model to separate highlights from object color," Proc. ICCV, pp. 145-150.

Meier U., Hürst W., and Duchnowski P. (1996), "Adaptive Bimodal Sensor Fusion for Automatic Speechreading" Proc. Intern. Conference on Acoustics, Speech and Signal Processing, ICASSP 1996

Meier U., Stiefelhagen R., Yang J. (1997), "Preprocessing of Visual Speech under Real World Conditions" European Tutorial & Research Workshop on Audio-Visual Speech Processing: Computational & Cognitive Science Approaches (AVSP 97)

Ohta, Y., Kanade, T., and Sakai, T. (1980), "Color information for region segmentation," Computer Graphics and Image Processing, Vol. 13, No. 3, pp.222-241.

Oliver, N., Pentland, A., and Berard, F. (1997), "LAFTER: lips and face realtime tracker," Proceedings of CVPR'97, pp. 123-129.

Pentland, A., Moghaddam, B., and Starner, T. (1994), "View-based and modular eigenspace for face recognition," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 84-91, Seattle, WA, USA.

Rowley, H.A., Baluja, S., and Kanade, T. (1995), "Human face detection in visual scenes," Technical Report CMU-CS-95-158, CS department, CMU, 1995.

Shafer, S.A. (1984), "Optical phenomena in computer vision," Proc. Canadian Soc. Computational Studies of Intelligence, pp. 572-577.

Sinha, P. (1994), "Object recognition via image invariants: a case study. Investigative ophthalmology and visual science," Vol. 35, pp. 1735-1740.

Stiefelhagen, R., Yang, J., and Waibel, A., (1996) "A model-based gaze tracking system," Proceedings of Joint Symposia on Intelligence and Systems (Washington DC).

Stiefelhagen, R. and Yang, J. (1997), "Gaze tracking for multimodal human-computer interaction," Proceedings of 1997 ICASSP (Munich, Germany).

Sung, K., and Poggio, T. (1994), "Example-based learning for view-based human face detection," Technical Report 1521, MIT AI Lab.

Swain, M.J., and Ballard, D.H. (1991), "Color indexing," International Journal of Computer Vision. Vol. 7, No.1, pp. 11-32.

Turk, M.A., and Pentland, A. (1991), "Face recognition using eigenfaces," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 586-591, Maui, HI, USA.

Wyszecki, G., and Styles, W.S. (1982), "Color Science: Concepts and Methods, Quantitative Data and Formulae," Second Edition, John Wiley & Sons, New York.

Yang, J., and Waibel, A. (1995), "A real-time face tracker," Proceedings of the Third IEEE Workshop on Applications of Computer Vision (Sarasota, Florida, 1996), pp. 142-147 (Technical Report CMU-CS-95-210, CS department, CMU, 1995).

Yang, J., Wu, L., and Waibel, A. (1996), "Focus of attention: towards low bitrate video tele-conferencing," Proceedings of 1996 IEEE International Conference on Image Processing (Lausanne, Switzerland), Vol. 2, pp. 97-100.

Yang, J., Lu, W., and Waibel, A. (1997), "Skin-color modeling and adaptation," Proceedings of ACCV'98 (Technical Report CMU-CS-97-146, CS department, CMU, 1997).

Yuille, A., Hallinan, P., and Cohen, D. (1992), "Feature extraction from faces using deformable templates," Int. J. Computer Vision, Vol. 8, No. 2, pp. 99-111.