Transactions Papers

Face Segmentation Using Skin-Color Map in Videophone Applications

Douglas Chai, *Student Member, IEEE*, and King N. Ngan, *Senior Member, IEEE*

***Abstract*— This paper addresses our proposed method to automatically segment out a person's face from a given image that consists of a head-and-shoulders view of the person and a complex background scene. The method involves a fast, reliable, and effective algorithm that exploits the spatial distribution characteristics of human skin color. A universal skin-color map is derived and used on the chrominance component of the input image to detect pixels with skin-color appearance. Then, based on the spatial distribution of the detected skin-color pixels and their corresponding luminance values, the algorithm employs a set of novel regularization processes to reinforce regions of skin-color pixels that are more likely to belong to the facial regions and eliminate those that are not. The performance of the face-segmentation algorithm is illustrated by some simulation results carried out on various head-and-shoulders test images.**

The use of face segmentation for video coding in applications such as videotelephony is then presented. We explain how the face-segmentation results can be used to improve the perceptual quality of a videophone sequence encoded by the H.261-compliant coder.

***Index Terms*— Color image processing, face location, facial image analysis, H.261, image segmentation, quantization, video coding, videophone communication.**

I. Introduction

THE task of finding a person's face in a picture seems to be effortless for a human to perform. However, it is far from simple for a machine of current technology to do the same. In fact, development of such a machine or system has been widely and actively studied in the field of image understanding for the past few decades with applications such as machine vision and face recognition in mind. Moreover, in recent years, the research activities in this area have intensified as a result of its applications being extended toward video representation and coding purposes.

The main objective of this research is to design a system that can find a person's face from given image data. This problem is commonly referred to as face location, face extraction, or face segmentation. Regardless of the terminology, they all share the same objective. However, note that the problem usually deals with finding the position and contour of a person's face since its location is unknown, but given the knowledge of its existence. If this is not known, then there is also a need to discriminate between "images containing faces" and "images not containing faces." This is known as face detection. This paper, however, focuses on face segmentation.

The significance of this problem can be illustrated by its vast applications, as face segmentation holds an important key to future advances in human-to-human and human-to-machine communications. The segmentation of a facial region provides a content-based representation of the image where it can be used for encoding, manipulation, enhancement, indexing, modeling, pattern-recognition, and object-tracking purposes. Some major applications include the following.

- *Coding area of interest with better quality:* The subjective quality of a very low-bit-rate encoded videophone sequence can be improved by coding the facial image region that is of interest to viewers at higher quality [1], [2].
- *Content-based representation and MPEG-4:* Face segmentation is a useful tool for the MPEG-4 content-based functionality. It provides content-based representation of the image, which can subsequently be used for coding, editing, or other interactivity purposes.
- *Three-dimensional (3-D) human face model fitting:* The delimitation of the person's face is the fundamental requirement of 3-D human face model fitting used in model-based coding [3], computer animation, and morphing.
- *Image enhancement:* Face segmentation information can be used in a postprocessing task for enhancing images, such as the automatic adjustment of tint in the facial region.
- *Face recognition:* Finding the person's face is the first important step in the human face recognition, classification, and identification systems.
- *Face tracking:* Face location can be used to design a video camera system that tracks a person's face in a room. It can be used as part of an intelligent vision system or simply in video surveillance.

Manuscript received August 17, 1997; revised September 3, 1998. This paper was recommended by Associate Editor S. Panchanathan.

The authors are with the Visual Communications Research Group, Department of Electrical and Electronic Engineering, University of Western Australia, Nedlands, Perth 6907 Australia.

Publisher Item Identifier S 1051-8215(99)04160-9.

Although the research on face segmentation has been pursued at a feverish pace, there are still many problems yet to be fully and convincingly solved as the level of difficulty of the problem depends highly on the complexity level of the image content and its application. Many existing methods only work well on simple input images with a benign background and frontal view of the person's face. To cope with more complicated images and conditions, many more assumptions will then have to be made. Many of the approaches proposed over the years involved the combination of shape, motion, and statistical analysis [4]–[13]. In recent times, however, a new approach of using color information has been introduced.

In this paper, we will discuss the color analysis approach to face segmentation. The discussion includes the derivation of a universal model of human skin color, the use of appropriate color space, and the limitations of color segmentation. We then present a practical solution to the face-segmentation problem. This includes how to derive a robust skin-color reference map and how to overcome the limitations of color segmentation. In addition to face segmentation, one of its applications on video coding will be presented in further detail. It will explain how the face-segmentation results can be exploited by an existing video coder so that it encodes the area of interest (i.e., the facial region) with higher fidelity and hence produces images with better rendered facial features.

This paper is organized as follows. The color analysis approach to face segmentation is presented in Section II. In Section III, we present our contributions to this field of research, which include our proposed skin-color reference map and methodology to face segmentation. The simulation results of our proposed algorithm along with some discussion is provided in Section IV. This is followed by Section V, which describes a video coding technique that uses the face-segmentation results. The conclusions and further research directions are presented in Section VI.

II. Color Analysis

The use of color information has been introduced to the face-locating problem in recent years, and it has gained increasing attention since then. Some recent publications that have reported this study include [14]–[23]. They have all shown, in one way or another, that color is a powerful descriptor that has practical use in the extraction of face location.

The color information is typically used for region rather than edge segmentation. We classify the region segmentation into two general approaches, as illustrated in Fig. 1. One approach is to employ color as a feature for partitioning an image into a set of homogeneous regions. For instance, the color component of the image can be used in the region growing technique, as demonstrated in [24], or as a basis for a simple thresholding technique, as shown in [23]. The other approach, however, makes use of color as a feature for identifying a specific object in an image. In this case, the skin color can be used to identify the human face. This is feasible because human faces have a special color distribution that differs significantly (although not entirely) from those of the background objects. Hence this approach requires a color map that models the skin-color distribution characteristics.

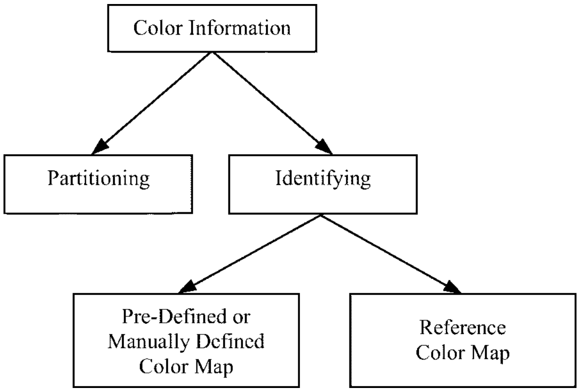


Fig. 1. The use of color information for region segmentation.

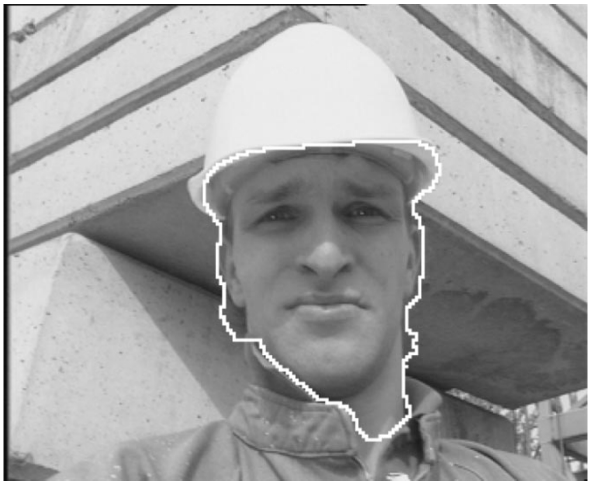


Fig. 2. *Foreman* image with a white contour highlighting the facial region.

The skin-color map can be derived in two ways on account of the fact not all faces have identical color features. One approach is to predefine or manually obtain the map such that it suits only an individual color feature. For example, here we obtain the skin-color feature of the subject in a standard head-and-shoulders test image called *Foreman*. Although this is a color image in YCrCb format, its gray-scale version is shown in Fig. 2. The figure also shows a white contour highlighting the facial region. The histograms of the color information (i.e., Cr and Cb values) bounded within this contour are obtained as shown in Fig. 3. The diagrams show that the chrominance values in the facial region are narrowly distributed, which implies that the skin color is fairly uniform. Therefore, this individual color feature can simply be defined by the presence of Cr values within, say, 136 and 156, and Cb values within 110 and 123. Using these ranges of values, we managed to locate the subject's face in another frame of *Foreman* and also in a different scene (a standard test image called *Carphone*), as can be seen in Fig. 4. This approach was suggested in the past by Li and Forchheimer in [14]; however, a detailed procedure on the modeling of individual color features and their choice of color space was not disclosed.

In another approach, the skin-color map can be designed by adopting histograming technique on a given set of training

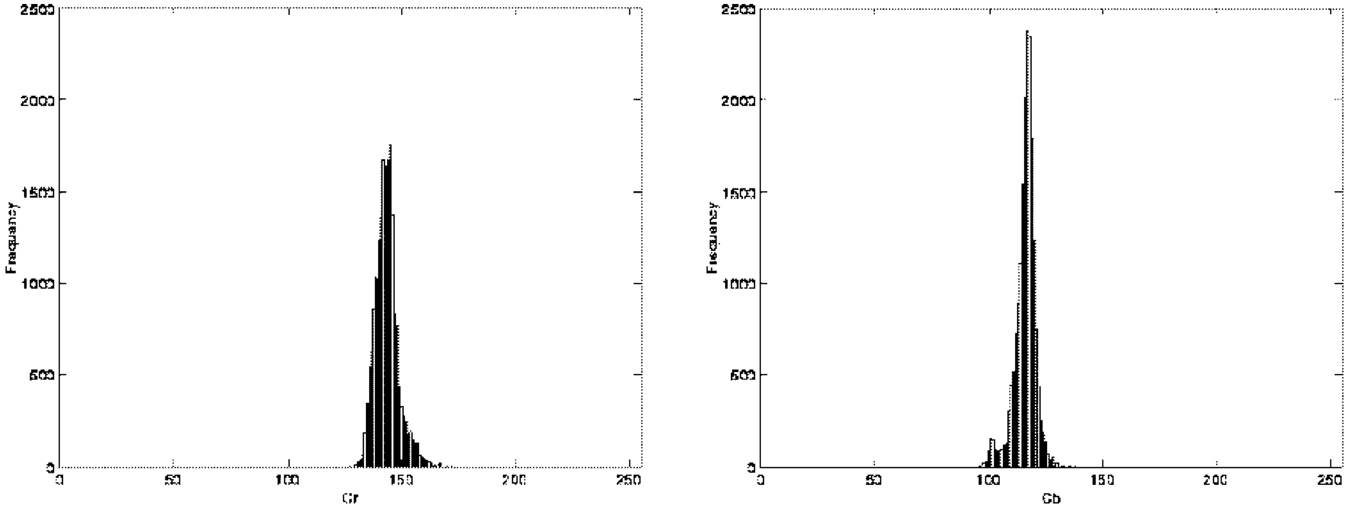


Fig. 3. Histograms of Cr and Cb components in the facial region.

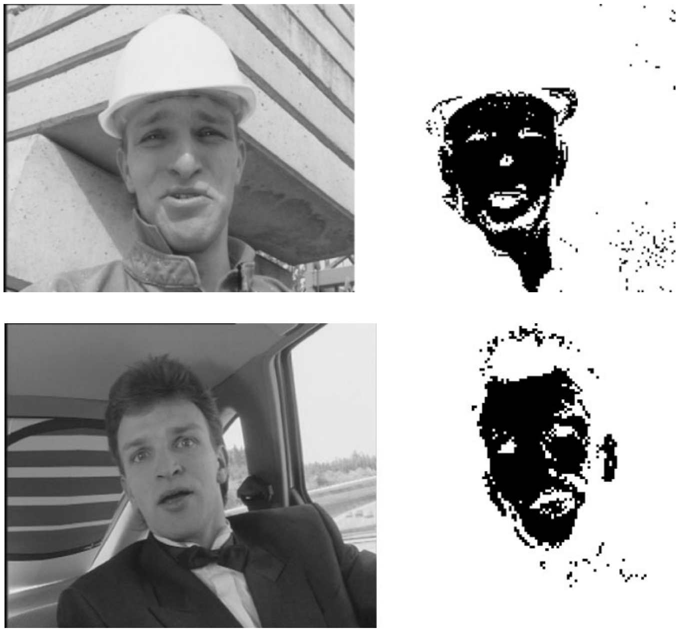


Fig. 4. *Foreman* and *Carphone* images, and their color segmentation results, obtained by using the same predefined skin-color map.

data and subsequently used as a reference for any human face. Such a method was successfully adopted by the authors [21], [25], Sobottka and Pitas [18], and Cornall and Pang [22].

Among the two approaches, the first is likely to produce better segmentation results in terms of reliability and accuracy by virtue of using a precise map. However, it is realized at the expense of having a face-segmentation process either that is too restrictive because it uses a predefined map or requires human interaction to manually define the necessary map. Therefore, the second approach is more practical and appealing, as it attempts to cater to all personal color features in an automatic manner, albeit in a less precise way. This, however, raises a very important issue regarding the coverage of all human races with one reference map. In addition, the general use of a skin-color model for region segmentation prompts two other questions, namely, which color space to use and how to distinguish other parts of the body and background objects with skin-color appearance from the actual facial region.

A. Color Space

An image can be presented in a number of different color space models.

- *RGB:* This stands for the three primary colors: red, green, and blue. It is a hardware-oriented model and is well known for its color-monitor display purpose.
- *HSV:* An acronym for hue-saturation-value. *Hue* is a color attribute that describes a pure color, while *saturation* defines the relative purity or the amount of white light mixed with a *hue; value* refers to the brightness of the image. This model is commonly used for image analysis.
- *YCrCb:* This is yet another hardware-oriented model. However, unlike the RGB space, here the luminance is separated from the chrominance data. The Y value represents the luminance (or brightness) component, while the Cr and Cb values, also known as the color difference signals, represent the chrominance component of the image.

These are some, but certainly not all, of the color space models available in image processing. Therefore, it is important to choose the appropriate color space for modeling human skin color. The factors that need to be considered are *application* and *effectiveness*. The intended purpose of the face segmentation will usually determine which color space to use; at the same time, it is essential that an effective and robust skin-color model can be derived from the given color space. For instance, in this paper, we propose the use of the YCrCb color space, and the reason is twofold. First, an effective use of the chrominance information for modeling human skin color can be achieved in this color space. Second, this format is typically used in video coding, and therefore the use of the same, instead of another, format for segmentation will avoid the extra computation required in conversion. On the other hand, both Sobottka and Pitas [18] and Saxe and Foulds [19] have opted for the HSV color space, as it is compatible with human color perception, and the *hue* and *saturation* components have been reported also to be sufficient for discriminating color information for modeling skin color. However, this color space is not suitable for video coding. Hunke and Waibel [15] and Graf *et al.* [26] used a normalized RGB color space. The normalization was employed to minimize the dependence on the luminance values.

On this note, it is interesting to point out that unlike the YCrCb and HSV color spaces, whereby the brightness component is decoupled from the color information of the image, in the RGB color space it is not. Therefore, Graf *et al.* have suggested preprocessing calibration in order to cope with unknown lighting conditions. From this point of view, the skin-color model derived from the RGB color space will be inferior to those obtained from the YCrCb or HSV color spaces. Based on the same reasoning, we hypothesize that a skin-color model can remain effective regardless of the variation of skin color (e.g., black, white, or yellow) if the derivation of the model is independent of the brightness information of the image. This will be discussed in later sections.

B. Limitations of Color Segmentation

A simple region segmentation based on the skin-color map can provide accurate and reliable results if there is a good contrast between skin color and those of the background objects. However, if the color characteristic of the background is similar to that of the skin, then pinpointing the exact face location is more difficult, as there will be more falsely detected background regions with skin-color appearance. Note that in the context of face segmentation, other parts of the body are also considered as background objects. There are a number of methods to discriminate between the face and the background objects, including the use of other cues such as motion and shape.

Provided that the temporal information is available and there is *a priori* knowledge of a stationary background and no camera motion, motion analysis can be incorporated into the face-localization system to identify nonmoving skin-color regions as background objects. Alternatively, shape analysis involving ellipse fitting can also be employed to identify the facial region from among the detected skin-color regions. It is a common observation that the appearance of a human face resembles an oval shape, and therefore it can be approximated by an ellipse [2]. In this paper, however, we propose a set of regularization processes that are based on the spatial distribution and the corresponding luminance values of the detected skin-color pixels. This approach overcomes the restriction of motion analysis and avoids the extensive computation of the ellipse-fitting method. The details will be discussed in the next section along with our proposed method for face segmentation.

In addition to poor color contrast, there are other limitations of color segmentation when an input image is taken in some particular lighting conditions. The color process will encounter some difficulty when the input image has:

- a "bright spot" on the subject's face due to reflection of intense lighting;
- a dark shadow on the face as a result of the use of strong directional lighting that has partially blackened the facial region;
- been captured with the use of color filters.

Note that these types of images (particularly in cases 1 and 2) are posing great technical challenges not only to the color segmentation approach but also to a wide range of other face-segmentation approaches, especially those that utilize edge image, intensity image, or facial feature-points extraction.

However, we have found that the color analysis approach is immune to moderate illumination changes and shading resulting from a slightly unbalanced light source, as these conditions do not alter the chrominance characteristics of the skin-color model.

III. Face-Segmentation Algorithm

In this section, we present our methodology to perform face segmentation. Our proposed approach is automatic in the sense that it uses an unsupervised segmentation algorithm, and hence no manual adjustment of any design parameter is needed in order to suit any particular input image. Moreover, the algorithm can be implemented in real time, and its underlying assumptions are minimal. In fact, the only principal assumption is that the person's face must be present in the given image, since we are locating and not detecting whether there is a face. Thus, the input information required by the algorithm is a single color image that consists of a head-and-shoulders view of the person and a background scene, and the facial region can be as small as only a 32×32 pixels window (or 1%) of a CIF-size (352×288) input image. The format of the input image is to follow the YCrCb color space, based on the reason given in the previous section. The spatial sampling frequency ratio of Y, Cr, and Cb is $4:1:1$. So, for a CIF-size image, Y has 288 lines and 352 pixels per line, while both Cr and Cb have 144 lines and 176 pixels per line each.

The algorithm consists of five operating stages, as outlined in Fig. 5. It begins by employing a low-level process like color segmentation in the first stage, then uses higher level operations that involve some heuristic knowledge about the local connectivity of the skin-color pixels in the later stages. Thus, each stage makes full use of the result yielded by its preceding stage in order to refine the output result. Consequently, all the stages must be carried out progressively according to the given sequence.

A detailed description of each stage is presented below. For illustration purposes, we will use a studio-based head-and-shoulders image called *Miss America* to present the intermediate results obtained from each stage of the algorithm. This input image is shown in Fig. 6.

A. Stage One—Color Segmentation

The first stage of the algorithm involves the use of color information in a fast, low-level region segmentation process. The aim is to classify pixels of the input image into skin color and non-skin color. To do so, we have devised a skin-color reference map in YCrCb color space.

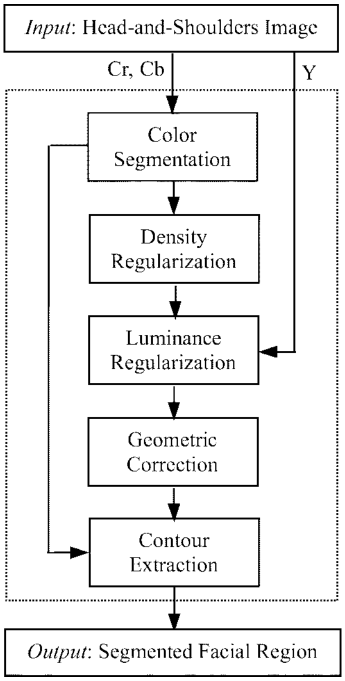


Fig. 5. Outline of face-segmentation algorithm.

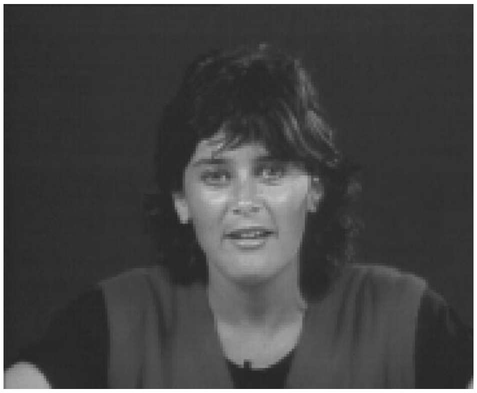


Fig. 6. Input image of *Miss America.*

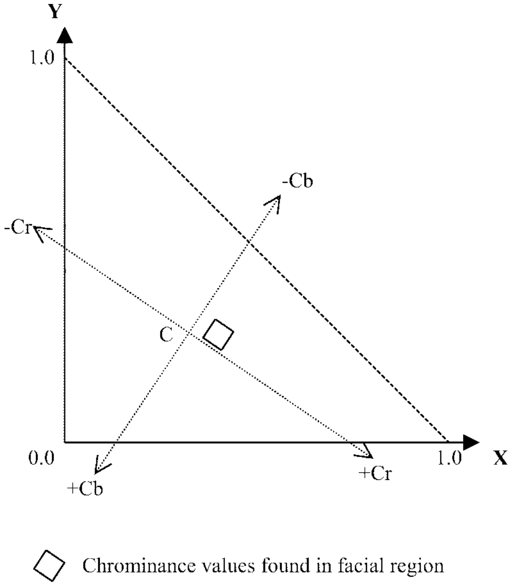


Fig. 7. Skin-color region in CIE chromaticity diagram.

We have found that a skin-color region can be identified by the presence of a certain set of chrominance (i.e., Cr and Cb) values narrowly and consistently distributed in the YCrCb color space. The location of these chrominance values has been found and can be illustrated using the CIE chromaticity diagram as shown in Fig. 7. We denote $R_{\rm Cr}$ and $R_{\rm Cb}$ as the respective ranges of Cr and Cb values that correspond to skin color, which subsequently define our skin-color reference map. The ranges that we found to be the most suitable for all the input images that we have tested are $R_{\rm Cr} = [133\,173]$ and $R_{\rm Cb} = [77\,127]$. This map has been proven, in our experiments, to be very robust against different types of skin color. Our conjecture is that the different skin color that we perceived from the video image cannot be differentiated from the chrominance information of that image region. So, a map that is derived from Cr and Cb chrominance values will remain effective regardless of skin-color variation (see Section IV for the experimental results). Moreover, our intuitive justification for the manifestation of similar Cr and Cb distributions of skin color of all races is that the apparent difference in skin color that viewers perceived is mainly due to the darkness or fairness of the skin; these features are characterized by the difference in the brightness of the color, which is governed by Y but not Cr and Cb.

With this skin-color reference map, the color segmentation can now begin. Since we are utilizing only the color information, the segmentation requires only the chrominance component of the input image. Consider an input image of $M \times N$ pixels, for which the dimension of Cr and Cb therefore is $M/2 \times N/2$. The output of the color segmentation, and hence stage one of the algorithm, is a bitmap of $M/2 \times N/2$ size, described as

$$O_1(x,y) = \begin{cases} 1, & \text{if } [\mathrm{Cr}(x,y) \in R_{\rm Cr}] \bigcap [\mathrm{Cb}(x,y) \in R_{\rm Cb}] \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $x = 0, \cdots, M/2 - 1$ and $y = 0, \cdots, N/2 - 1$. The output pixel at point (x, y) is classified as skin color and set to one if both the Cr and Cb values at that point fall inside their respective ranges $R_{\rm Cr}$ and $R_{\rm Cb}$. Otherwise, the pixel is classified as non-skin color and set to zero. To illustrate this, we perform color segmentation on the input image of *Miss America,* and the bitmap produced can be seen in Fig. 8. The output value of one is shown in black, while the value of zero is shown in white (this convention will be used throughout this paper).

Among all the stages, this first stage is the most vital. Based on our model of human skin color, the color segmentation has to remove as many pixels as possible that are unlikely to belong to the facial region while catering for a wide variety of skin color. However, if it falsely removes too many pixels that belong to the facial region, then the error will propagate down the remaining stages of the algorithm, consequently causing a failure to the entire algorithm.

Nevertheless, the result of color segmentation is the detection of pixels in a facial area and may also include other areas

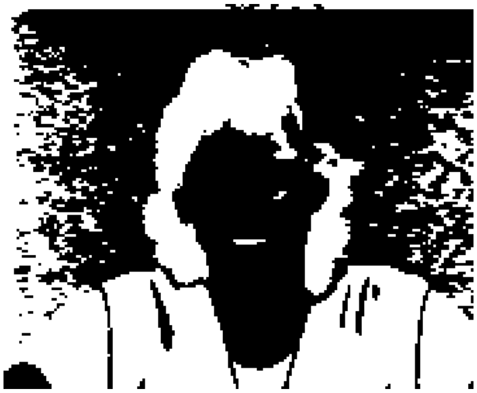


Fig. 8. Bitmap produced by stage one.

where the chrominance values coincide with those of the skin color (as is the case in Fig. 8). Hence the successive operating stages of the algorithm are used to remove these unwanted areas.

B. Stage Two—Density Regularization

This stage considers the bitmap produced by the previous stage to contain the facial region that is corrupted by noise. The noise may appear as small holes on the facial region due to undetected facial features such as eyes and mouth, or it may also appear as objects with skin-color appearance in the background scene. Therefore, this stage performs simple morphological operations such as *dilation* to fill in any small hole in the facial area and *erosion* to remove any small object in the background area. The intention is not necessarily to remove the noise entirely but to reduce its amount and size.

To distinguish between these two areas, we first need to identify regions of the bitmap that have higher probability of being the facial region. The probability measure that we used is derived from our observation that the facial color is very uniform, and therefore the skin-color pixels belonging to the facial region will appear in a large cluster, while the skin-color pixels belonging to the background may appear as large clusters or small isolated objects. Thus, we study the density distribution of the skin-color pixels detected in stage one. An $M/8 \times N/8$ array of density values, called density map $D(x,y)$, is computed as

$$D(x,y) = \sum_{i=0}^{3}\sum_{j=0}^{3} O_1(4x+i, 4y+j) \tag{2}$$

where $x = 0, \cdots, M/8-1$ and $y = 0, \cdots, N/8-1$. It first partitions the output bitmap of stage one $O_1(x,y)$ into nonoverlapping groups of 4 × 4 pixels, then counts the number of skin-color pixels within each group and assigns this value to the corresponding point of the density map.

According to the density value, we classify each point into three types, namely, zero ($D = 0$), intermediate ($0 < D < 16$), and full ($D = 16$). A group of points with zero density value will represent a nonfacial region, while a group of full-density points will signify a cluster of skin-color pixels and a high probability of belonging to a facial region. Any point of intermediate density value will indicate the presence of noise. The density map of *Miss America* with the three density classifications is depicted in Fig. 9. The point of zero density is shown in white, intermediate density in gray, and full density in black.

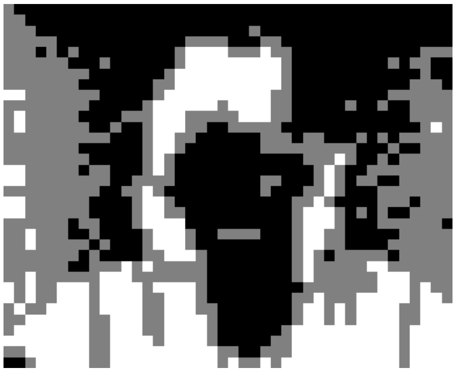


Fig. 9. Density map after classification.

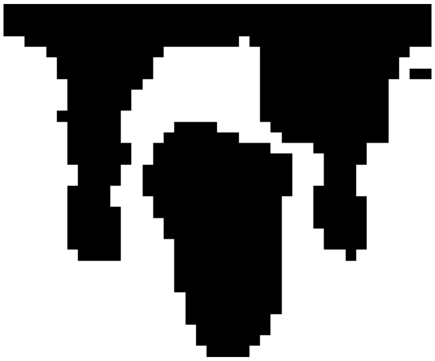


Fig. 10. Bitmap produced by stage two.

Once the density map is derived, we can then begin the process that we termed as density regularization. This involves the following three steps.

1) Discard all points at the edge of the density map, i.e., set $D(0,y) = D(M/8-1,y) = D(x,0) = D(x,N/8-1) = 0$ for all $x = 0,\cdots,M/8-1$ and $y = 0,\cdots,N/8-1$.
2) Erode any full-density point (i.e., set to zero) if it is surrounded by less than five other full-density points in its local 3 × 3 neighborhood.
3) Dilate any point of either zero or intermediate density (i.e., set to 16) if there are more than two full-density points in its local 3 × 3 neighborhood.

After this process, the density map is converted to the output bitmap of stage two as

$$O_2(x,y) = \begin{cases} 1, & \text{if } D(x,y) = 16 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

for all $x = 0,\cdots,M/8-1$ and $y = 0,\cdots,N/8-1$.

The result of stage two for the *Miss America* image is displayed in Fig. 10. Note that this bitmap is now four times lower in spatial resolution than that of the output bitmap in stage one.

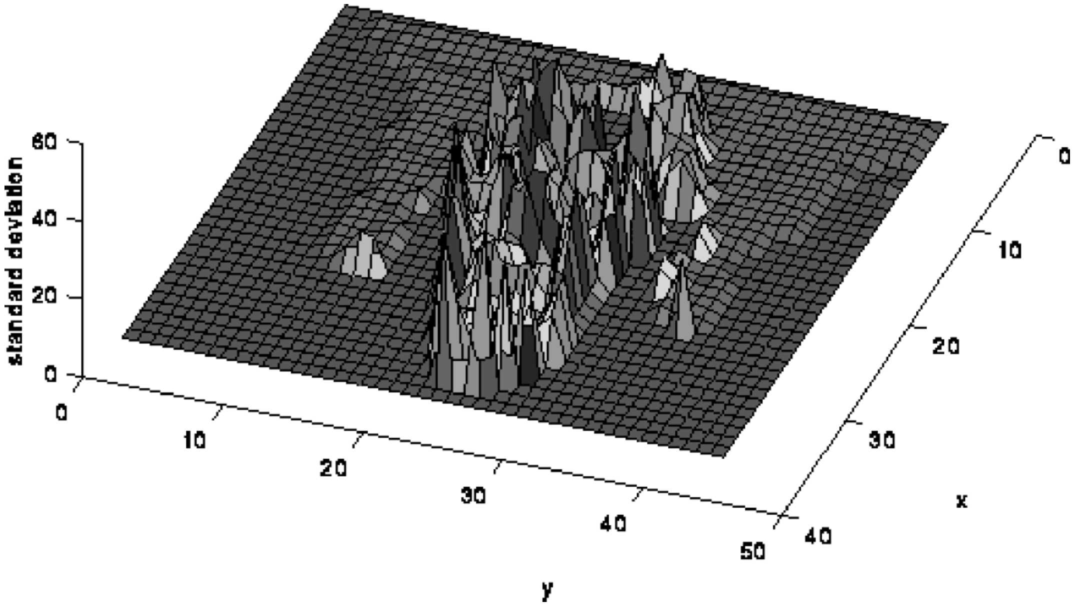


Fig. 11. Standard deviation values of the detected pixels in $O_2(x, y)$.

C. Stage Three—Luminance Regularization

We have found that in a typical videophone image, the brightness is nonuniform throughout the facial region, while the background region tends to have a more even distribution of brightness. Hence, based on this characteristic, background region that was previously detected due to its skin-color appearance can be further eliminated.

The analysis employed in this stage involves the spatial distribution characteristic of the luminance values since they define the brightness of the image. We use standard deviation as the statistical measure of the distribution. Note that the size of the previously obtained bitmap $O_2(x,y)$ is $M/8 \times N/8$; hence each point corresponds to a group of 8 × 8 luminance values, denoted by W, in the original input image. For every skin-color pixel in $O_2(x,y)$, we calculate the standard deviation, denoted as $\sigma(x,y)$, of its corresponding group of luminance values, using

$$\sigma(x,y) = \sqrt{E[W^2] - (E[W])^2}. \tag{4}$$

Fig. 11 depicts the standard deviation values calculated for the *Miss America* image.

If the standard deviation is below a value of two, then the corresponding 8 × 8 pixels region is considered too uniform and therefore unlikely to be part of the facial region. As a result, the output bitmap of stage three, denoted as $O_3(x,y)$, is derived as

$$O_3(x,y) = \begin{cases} 1, & \text{if } O_2(x,y) = 1 \quad \text{and} \quad \sigma(x,y) \geq 2 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

for all $x = 0, \cdots, M/8-1$ and $y = 0, \cdots, N/8-1$. The output bitmap of this stage for the *Miss America* image is presented in Fig. 12. The figure shows that a significant portion of the unwanted background region was eliminated at this stage.

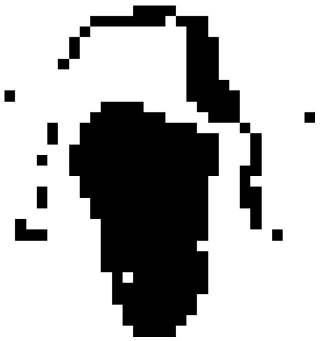


Fig. 12. Bitmap produced by stage three.

D. Stage Four—Geometric Correction

We performed a horizontal and vertical scanning process to identify the presence of any odd structure in the previously obtained bitmap, $O_3(x,y)$, and subsequently removed it. This is to ensure that a correct geometric shape of the facial region is obtained. However, prior to the scanning process, we will attempt to further remove any more noise by using a technique similar to that initially introduced in stage two. Therefore, a pixel in $O_3(x,y)$ with the value of one will remain as a detected pixel if there are more than three other pixels, in its local 3 × 3 neighborhood, with the same value. At the same time, a pixel in $O_3(x,y)$ with a value of zero will be reconverted to a value of one (i.e., as a potential pixel of the facial region) if it is surrounded by more than five pixels, in its local 3 × 3 neighborhood, with a value of one. These simple procedures will ensure that noise appearing on the facial region is filled in and that isolated noise objects on the background are removed.

We then commence the horizontal scanning process on the "filtered" bitmap. We search for any short continuous run of pixels that are assigned with the value of one. For a CIF-size image, the threshold for a group of connected pixels to

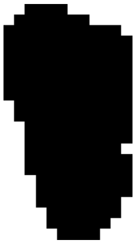


Fig. 13. Bitmap produced by stage four.

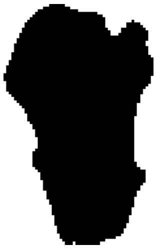


Fig. 14. Bitmap produced by stage five.

belong to the facial region is four. Therefore, any group of less than four horizontally connected pixels with the value of one will be eliminated and assigned to zero. A similar process is then performed in the vertical direction. The rationale behind this method is that, based on our observation, any such short horizontal or vertical run of pixels with the value of one is unlikely to be part of a reasonable-size and well-detected facial region. As a result, the output bitmap of this stage should contain the facial region with minimal or no noise, as demonstrated in Fig. 13.

E. Stage Five—Contour Extraction

In this final stage, we convert the $M/8 \times N/8$ output bitmap of stage four back to the dimension of $M/2 \times N/2$. To achieve the increase in spatial resolution, we utilize the edge information that is already made available by the color segmentation in stage one. Therefore, all the boundary points in the previous bitmap will be mapped into the corresponding group of 4× 4 pixels with the value of each pixel as defined in the output bitmap of stage one. The representative output bitmap of this final stage of the algorithm is shown in Fig. 14.

IV. Segmentation Results

The proposed skin-color reference map is intended to work on a wide range of skin color, including that of people of European, Asian, and African decent. Therefore, to show that it works on subject with skin color other than white (as is the case with the *Miss America* image), we have used the same map to perform the color-segmentation process on subjects with black and yellow skin color. The results obtained were very good, as can be seen in Fig. 15. The skin-color pixels were correctly identified, in both input images, with only a small amount of noise appearing, as expected, in the facial regions and background scenes, which can be removed by the remaining stages of the algorithm.

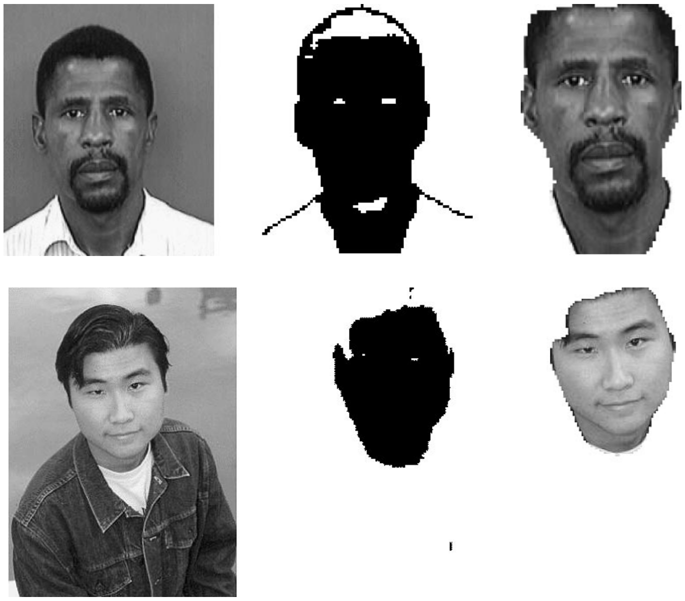
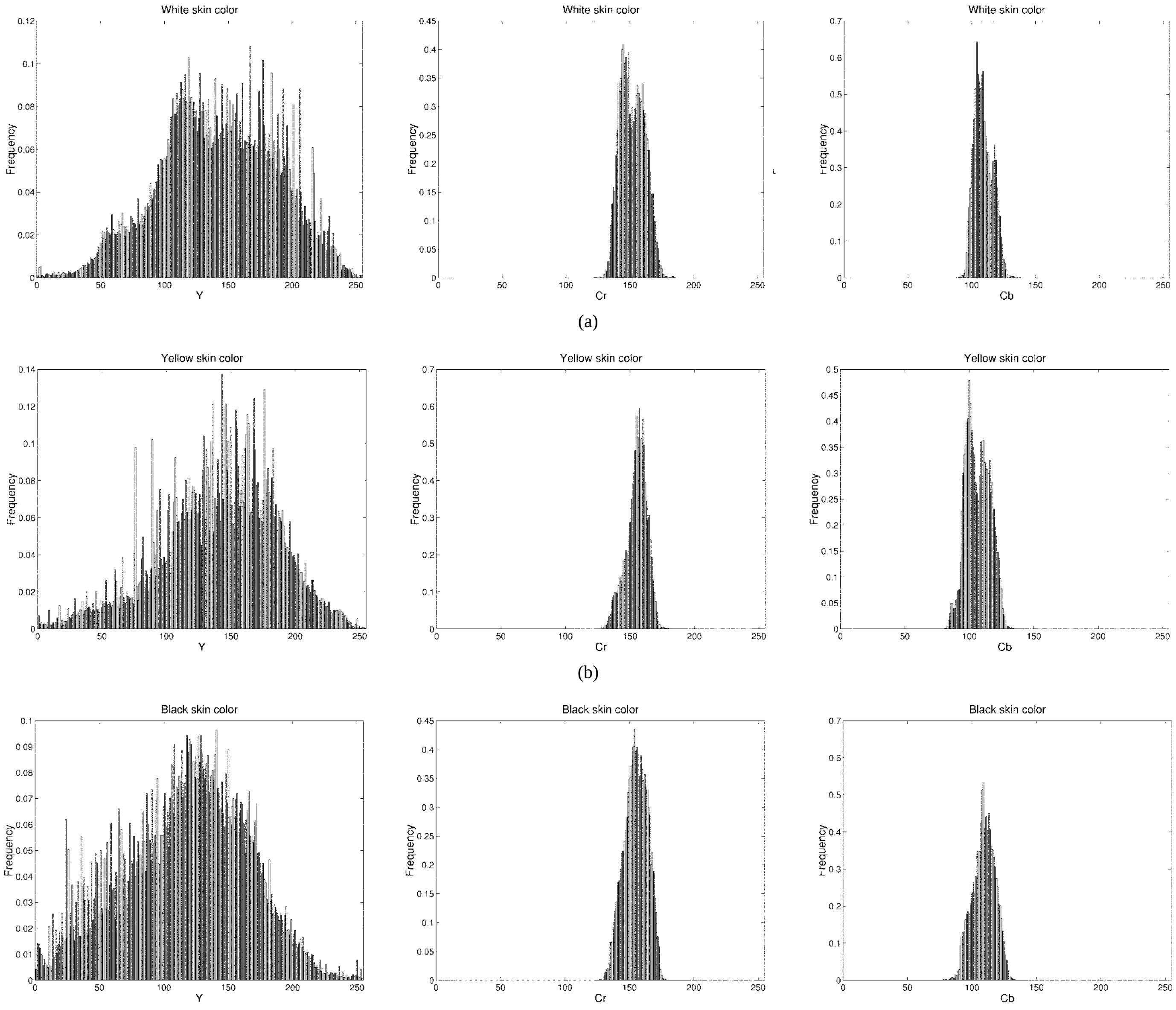


Fig. 15. Results produced by the color-segmentation process in stage one and the final output of the face segmentation algorithm.

We have further tested the skin-color map with 30 samples of images. Skin colors were grouped into three classes: white, yellow, and black. Ten samples, each of which contained the facial region of a different subject captured in a different lighting condition, were taken from each class to form the test set. We have constructed three normalized histograms for each sample in the separate Y, Cr, and Cb components. The normalization process was used to account for the variation of facial-region size in each sample. We have then taken the average results from the ten samples of each class. These average normalized histogram results are presented in Fig. 16. Since all samples were taken from different and unknown lighting conditions, the histograms of the Y component for all three classes cannot be used to verify whether the variations of luminance values in these image samples were caused by the different skin color or by the different lighting conditions. However, the use of such samples illustrated that the variation in illumination does not seem to affect the skin-color distribution in the Cr and Cb components. On the other hand, the histograms of Cr and Cb components for all three classes clearly showed that the chrominance values are indeed narrowly distributed, and more important, that the distributions are consistent across different classes. This demonstrated that an effective skin-color reference map could be achieved based on the Cr and Cb components of the input image.

The face-segmentation algorithm with this universal skin-color reference map was tested on many head-and-shoulders images. Here we emphasize that the face-segmentation process was designed to be completely automatic, and therefore the same design parameters and rules (including the reference skin-color map and the heuristic) as described in the previous section were applied to all the test images. The test set now contained 20 images from each class of skin color. Therefore, a total of 60 images of different subjects, background complexities, and lighting conditions from the three classes were



(c)

Fig. 16. Histograms of Y, Cr, and Cb values of different facial skin colors: (a) white, (b) yellow, and (c) black.

used. Using this test set, a success rate of 82% was achieved. The algorithm has performed successful segmentation of 49 out of 60 faces. Out of the 11 unsuccessful cases, seven cases have incorrect localization, two have partial localization, and two have both incorrect and partial localization.

The representative results shown in Fig. 17 illustrated the successful face segmentation achieved by the algorithm on two images with different background complexities. The edges of the facial regions were accurately obtained with no noise's appearing on either the facial region or the background. Moreover, the results were obtained in real time, as it took a SunSPARC 20 computer less than 1 μs to perform all computations required on a CIF-size input image.

In all seven incorrect localization cases, the segmentation results did contain the complete facial regions but also included some background regions. In four out of seven, the subject's hair, which is considered as a background region, was falsely identified as a facial region. Partial localization occurred in two cases and resulted in the localization of an incomplete facial region. These cases were caused by thick facial hair, i.e., mustache and beard. The two cases with both incorrect and partial localization have facial regions partially localized, and the results also contained some background regions.

Note that in all cases, the facial regions were always located, whether completely or partially.

V. Coding

Here, we describe a video coding technique, termed a foreground/background (FB) coding scheme, that uses the face-segmentation results to code the area of interest with better quality. In applications such as videotelephony, the face of the speaker is typically the most important image region for the viewer. Therefore, the face-segmentation algorithm is used to separate the facial area from its background scene to become

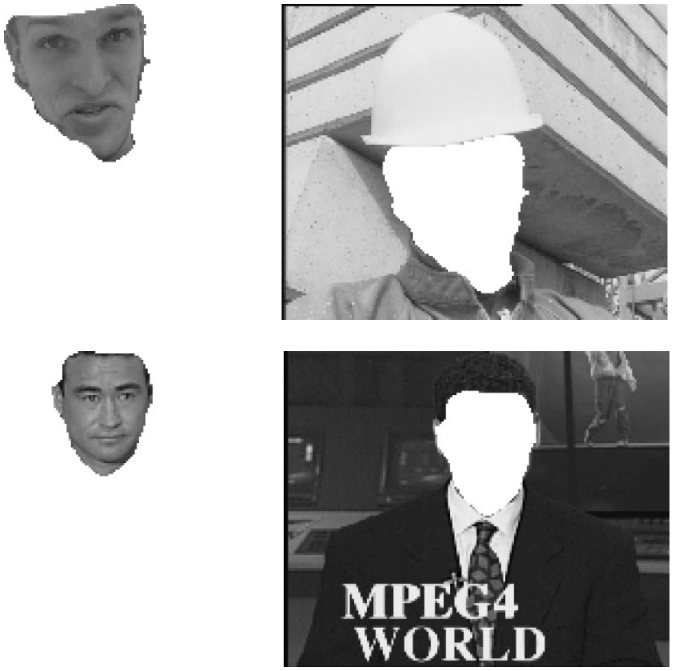


Fig. 17. Segmented facial regions and remaining background scenes.

the foreground region. Here, we propose to use the classical block-based video coding system. To be consistent with many of the video coding standards [27]–[30], the foreground and background regions will only need to be identified at the macroblock (MB) level.

In the FB encoding process, we allocate fewer bits for encoding the background MB's by using a higher quantization level. In doing so, we free up more bits that can then be used for encoding the foreground MB's. This bit transfer leads to a better quality encoded area of interest at the expense of having a lower quality background image. This is based on the premise that the background is usually of less significance to the viewer's perception, so the overall subjective quality of the image is perceptively improved and more pleasing to the viewer.

This concept was initially proposed by us in [1], where we introduced the FB coding scheme and its implementation as an additional encoding option for the H.263 codec [30]. In this paper, however, we will use the H.261 codec.

A. H.261FB

We have integrated the FB coding scheme into the well-known H.261 video coding system [29]. Hereafter, we term this approach H.261FB. The H.261FB coder utilizes the information obtained from the face-segmentation algorithm, as described in Section III, to enable bit transfer between the foreground and background MB's. This redistribution of bit allocation is simply attained by controlling the quantization level in a discriminatory manner. In addition, a new rate-control strategy is devised in order to regulate the bitstream produced by this discriminatory quantization process.

This approach will still produce a bitstream that conforms to the H.261 standard. The reason is that the new quantization process does not involve any modification to the bitstream syntax; it merely assigns two different values to two different regions. As for the rate control, there is no standardized technique. Hence the manufacturers of the encoder have the freedom to devise their own strategy. Moreover, we do not need to transmit the segmentation information to the decoder, as it is used in the encoder only. Therefore, the integration is supported by the syntax, and a full H.261 decoder compatibility is maintained.

B. Discriminatory Quantization Process

Two quantizers, instead of one, are used in the H.261FB approach. We assigned Q_f and Q_b to be the quantizers for the foreground (FG) and background (BG) MB's, respectively. Among the two, Q_f is a finer quantizer, while Q_b is a coarser one. H.261FB uses the *MQUANT* header to switch between these two quantizers, as shown in (6). The *MQUANT* header is a fixed-length code word of five bits that indicates the quantization level to be used for the current MB. Hence this 5-bit code word represents a range of quantization levels from 1 to 31

$$\mathit{MQUANT} = \begin{cases} Q_f, & \text{if current MB belongs to FG} \\ Q_b, & \text{if current MB belongs to BG.} \end{cases} \tag{6}$$

It is not necessary, however, for the encoder to send this header for every MB. The transmission of the *MQUANT* header is only required in one of the following cases:

1) when the current MB is in a different region from the previously encoded MB, i.e., a change from foreground to background MB or vice versa;
2) when the rate-control algorithm updates the quantization level in order to maintain a constant bit rate.

Naturally, this approach has to sustain a slight increase in the transmission of an *MQUANT* header. However, the benefit easily outweighs this overhead cost, as will be demonstrated in the simulation results.

C. Rate-Control Function

A new rate-control strategy is needed to adjust not one but now two quantizers periodically in order to regulate the bit rate. To do so, the quantizer can be adjusted as follows. The quantization parameter (or level) assigned to the quantizer can be defined as a simple function of buffer contents. Mathematically, the quantization parameter *QP* can be expressed as

$$\mathit{QP} = \frac{\mathit{BufferContents}}{Q_{\text{division}}} + Q_{\text{offset}} \tag{7}$$

where Q_{division} is the quantization division factor of the buffer and Q_{offset} is the offset factor. The *BufferContents* variable indicates how much data (in unit of bits) is currently stored in the buffer.

According to the RM8 coder [31] (a reference implementation of the H.261 coder, developed by the standardization study group), Q_{offset} is set to one to avoid zero quantization, while Q_{division} is equal to the target *Bitrate* divided by a constant value of 320, i.e.,

$$Q_{\mathrm{division}} = \frac{\mathit{Bitrate}}{320} = \frac{64\,000 \times p}{320} = 200p \tag{8}$$

where $\mathit{Bitrate} = p \times 64$ kbits/s, $p = 1, \cdots, 30$. Hence for the RM8 coder, the next quantization parameter is determined by the function described as

$$\mathit{QP} = \min\left\{ \left[\frac{\mathit{BufferContents}}{200p} + 1 \right], 31 \right\}. \tag{9}$$

The value of *QP* is clipped at 31 because the *MQUANT* header is a fixed-length code word of five bits. As the *BufferContents* increases, *QP* also increases in order to offset any rise in bit rate. The value of *QP* will remain at the maximum of 31 until the buffer is full, which takes place when the *BufferContents* variable reaches the maximum capacity of the buffer. When the *BufferContents* variable exceeds the buffer size, buffer overflow is said to occur. In such an event, the macroblock is skipped (i.e., not transmitted), and as a result, quantization is no longer needed.

In the H.261FB approach, two similar rate-control functions as mentioned above are used—one for the foreground region and another for the background. Each function will have different values of Q_{division} and Q_{offset}. For instance, we can set Q_{offset} to a higher value such that the function forces the quantizer to always adopt a coarser quantization parameter. Therefore, the amount of bit transfer between foreground and background MB's is mainly determined by the value of Q_{offset} being assigned to their respective rate-control functions. On the other hand, the offset factor Q_{division} governs how the bits are distributed within the same region.

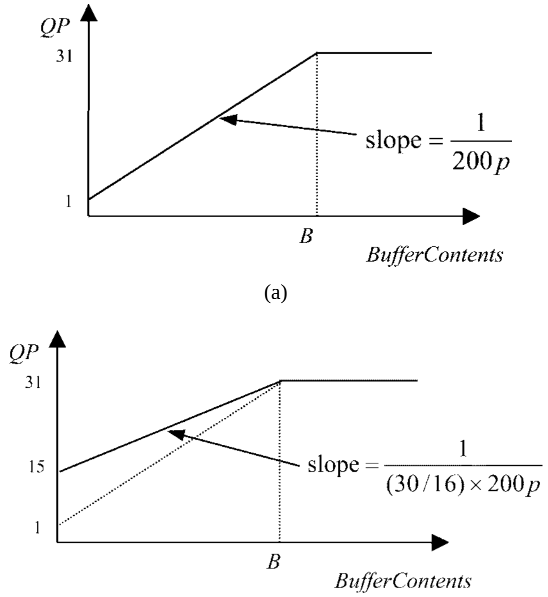
Here, we choose (9), the function defined in RM8, for the foreground region [see Fig. 18(a)]. As for the background region, we shift Q_{offset} to 15 and set Q_{division} to (30/16) $\times$ 200 p [see Fig. 18(b)]. This constrains the quantizer to a minimum value of 15, while the clipping of the quantization level to its maximum value will occur at the same level of buffer occupancy B as in the case of RM8.

D. Coding Results

The FB coding scheme is demonstrated on the CIF *Foreman* video sequence. First, we used our proposed face-segmentation algorithm to separate each frame of the input sequence into foreground and background MB's. The results for the first frame of the sequence are shown in Fig. 19(a) and (b).

We then encoded the sequence with both the RM8 and H.261FB coders. Note that, other than the use of the discriminatory quantization process and the new rate-control function as described in the previous section, the rest of the implementation of the H.261FB coder is the same as for RM8.

To evaluate the discriminatory quantization process, we performed intraframe coding on the first frame. To provide a fair comparison of image quality, the quantization parameters were manually obtained so that both approaches consume a similar amount of bits. Therefore, the quantizer for the RM8 coder was fixed at 22 throughout the entire encoding processing. For the H.261FB coder, the foreground quantizer Q_f and the background quantizer Q_b were set at 11 and 31 respectively. Overall, the RM8 coder spent an average of 105.81 bits per MB. Furthermore, we have identified that it spent an average of 89.01 bits per MB in the foreground region and 109.54 bits per MB in the background region. The quality of the encoded image is shown in Fig. 19(c). This is compared with the H.261FB-encoded image shown in Fig. 19(d), whereby the coder spent an average of 134.72 bits per foreground MB and 90.70 bits per background MB, while its overall average bit per MB was 98.70. This overall amount of bits used is about 7.11 bits per MB fewer than that of RM8, and yet the figures clearly show that the area of interest is much improved in the H.261FB-encoded image as a result of the bit transfer from the background to foreground region, while its degradation in the background region was hardly noticeable. The improvement can be further illustrated by magnifying the face region of the images as shown in Fig. 19(e) and (f).



(a)

(b)

Fig. 18. (a) Rate-control function used in the RM8 coder and (b) proposed rate-control function for the background MB's in the H.261FB coder.

To demonstrate the performance of our proposed rate-control functions for the FB coding scheme, both the RM8 and H.261FB coders were used to encode 100 frames of the *Foreman* sequence at a target bit rate of 192 kbits/s and frame rate of 10 f/s. A plot displaying the bit rates achieved by both coders is provided in Fig. 20. The simulation revealed that the subjective quality of the H.261FB-coded images was much better than that the RM8-coded images, and yet their bit rates were slightly lower. We illustrate the improvement by showing a representative frame 72 of the encoded images in Fig. 21. It can be clearly observed that the H.261FB-coded image in Fig. 21(b) has a better perceived quality and rendition of facial features than the RM8-coded image shown in Fig. 21(a).

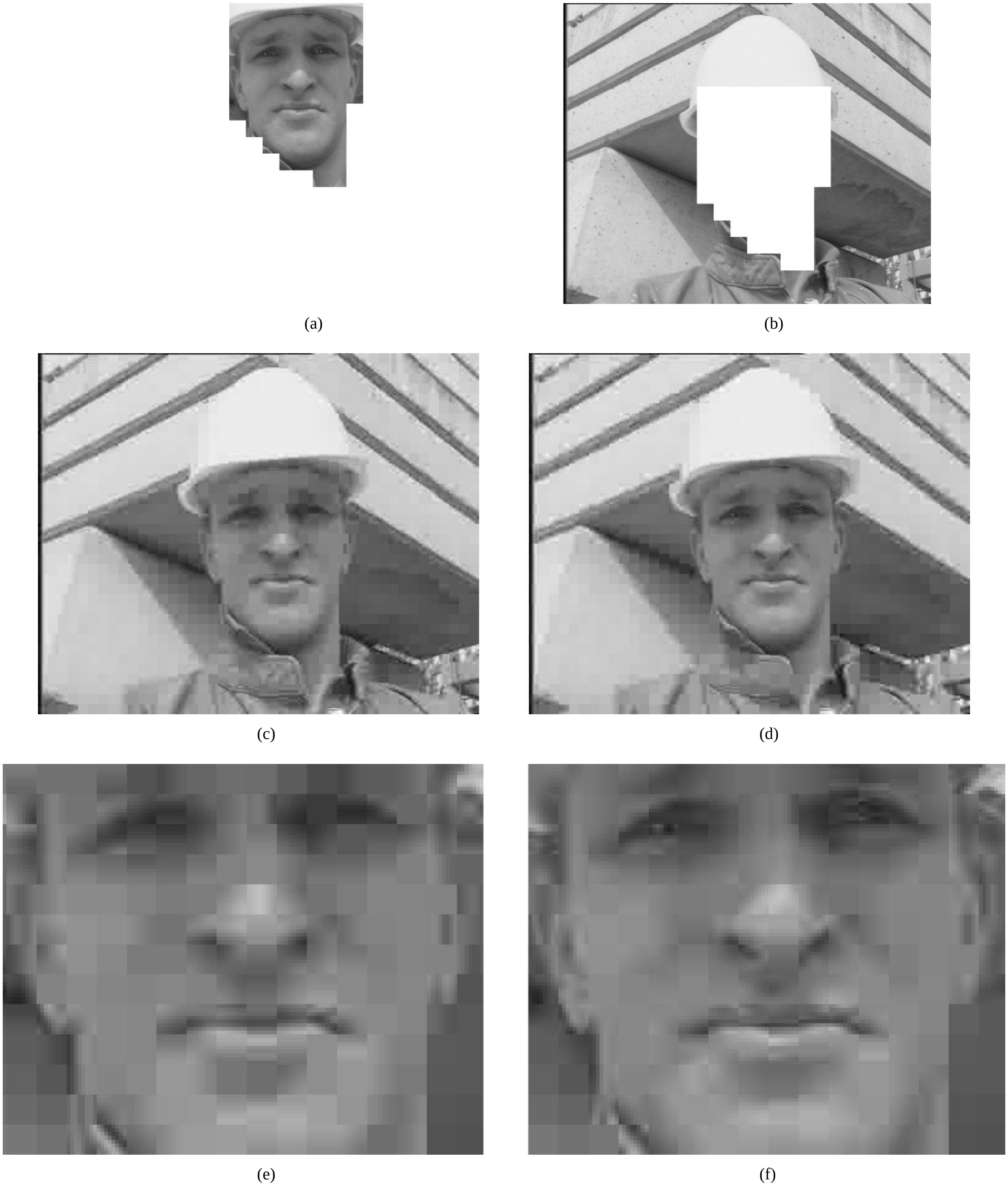


Fig. 19. (a) Foreground MB’s and (b) background MB’s (c) coded by RM8 and (d) coded by H.261FB. (e) Magnified image of (c). (f) Magnified image of (d).

VI. Concluding Remarks

The color analysis approach to face segmentation was discussed. In this approach, the face location can be identified by performing region segmentation with the use of a skin-color map. This is feasible because human faces have a special color distribution characteristic that differs significantly from those of the background objects. We have found that pixels belonging to the facial region, of the image in YCrCb color space, exhibit similar chrominance values. Furthermore, a consistent range of chrominance values was also discovered from many different facial images, which include people of European, Asian, and African descent. This led us to the derivation of a skin-color map that models the facial color of all human races.

With this universal skin-color map, we classified pixels of the input image into skin color and non-skin color.

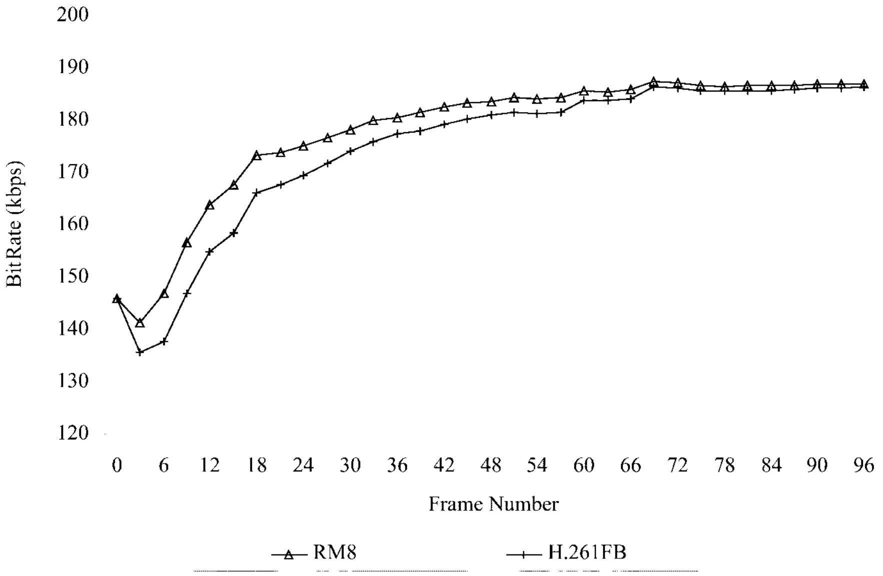


Fig. 20. Bit rates achieved by RM8 and H.261FB coders at a target bit rate of 192 kbits/s.

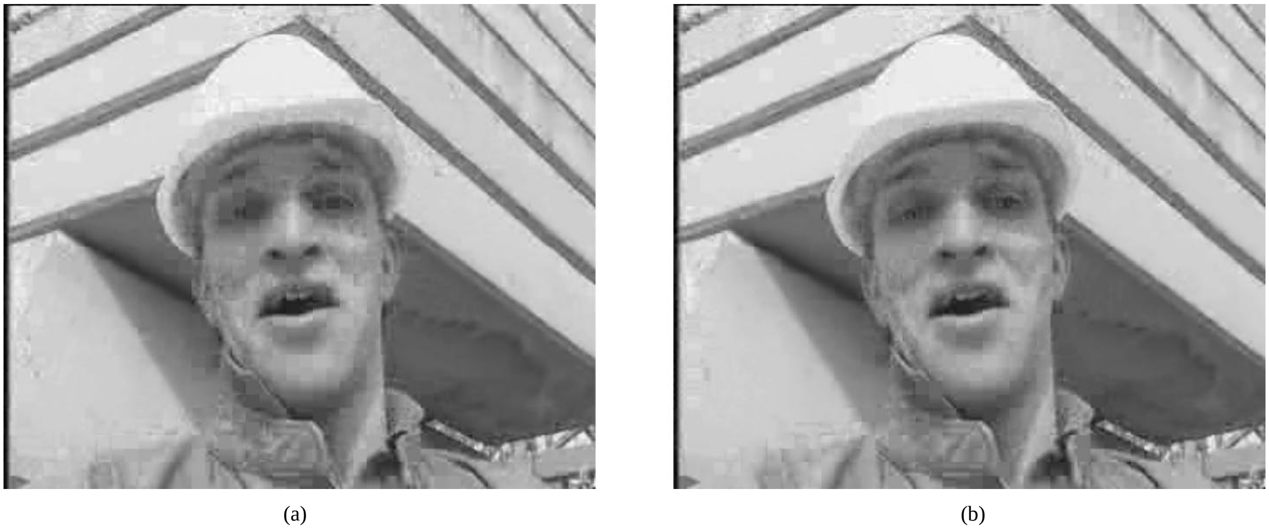


Fig. 21. Frame 72 of the coded results in Fig. 20: (a) RM8 and (b) H.261FB.

Consequently, a bitmap is produced, containing the facial region that is corrupted by noise. The noise may appear as small holes on the facial region due to undetected facial features, or it may also appear as objects with skin-color appearance in the background scene. To cope with this noise and, at the same time, refine the facial-region detection, we have proposed a set of novel region-based regularization processes that are based on the spatial distribution study of the detected skin-color pixels and their corresponding luminance values. All the operations are unsupervised and low in computational complexity.

Our proposed face-segmentation methodology was implemented and tested on many input images, each of which contains the head-and-shoulders view of a person and a complex background scene. A set of representative results from our simulations was shown in this paper. The results demonstrated that our algorithm can accurately segment out the facial regions from a diverse range of images that includes subjects with different skin colors and the presence of various background complexities. Furthermore, the face segmentation was done automatically and in real time.

The use of face segmentation for video coding in applications such as videotelephony was then presented. We described a foreground/background video coding scheme that uses the face-segmentation results to improve the perceptual quality of the encoded image with better rendition of the facial features. This technique involves bit transfer between the facial region and the background. The redistribution of bit allocation is controlled by a discriminatory quantization process. Then the bitstream generated from this process is regularized by a new rate-control strategy. We have integrated this approach into the H.261 framework with success. Improved image quality was obtained as shown by the simulation results in the paper.

Our future research will involve the use of temporal information to assist in face localization and also for tracking. For coding, a further study of the rate-control strategy, the use of segmentation-assisted motion estimation, and the proposal of coding the foreground and background regions at different frame rates will be investigated.

REFERENCES

[1] D. Chai and K. N. Ngan, "Foreground/background video coding scheme," in *Proc. IEEE Int. Symp. Circuits Syst.*, Hong Kong, June 1997, vol. II, pp. 1448–1451.

[2] A. Eleftheriadis and A. Jacquin, "Model-assisted coding of video teleconferencing sequences at low bit rates," in *Proc. IEEE Int. Symp. Circuits Syst.*, London, U.K., June 1994, vol. 3, pp. 177–180.

[3] K. Aizawa and T. Huang, "Model-based image coding: Advanced video coding techniques for very low-rate applications," *Proc. IEEE*, vol. 83, p. 259–271, Feb. 1995.

[4] V. Govindaraju, D. B. Sher, R. K. Srihari, and S. N. Srihari, "Locating human faces in newspaper photographs," in *Proc. IEEE Computer Vision Pattern Recognition Conf.*, San Diego, CA, June 1989, pp. 549–554.

[5] G. Sexton, "Automatic face detection for videoconferencing," in *Proc. Inst. Elect. Eng. Colloquium Low Bit Rate Image Coding*, May 1990, pp. 9/1–9/3.

[6] V. Govindaraju, S. N. Srihari, and D. B. Sher, "A computational model for face location," in *Proc. Int. Conf. Computer Vision*, Dec. 1990, pp. 718–721.

[7] H. Li, "Segmentation of the facial area for videophone applications," *Electron. Lett.*, vol. 28, pp. 1915–1916, Sept. 1992.

[8] S. Shimada, "Extraction of scenes containing a specific person from image sequences of a real-world scene," in *Proc. IEEE TENCON'92*, Melbourne, Australia, Nov. 1992, pp. 568–572.

[9] M. Menezes de Sequeira and F. Pereira, "Knowledge-based videotelephone sequence segmentation," in *Proc. SPIE Visual Commun. and Image Processing*, vol. 2094, Nov. 1993, pp. 858–869.

[10] G. Yang and T. S. Huang, "Human face detection in a complex background," *Pattern Recognit.*, vol. 27, no. 1, pp. 53–63, Jan. 1994.

[11] A. Eleftheriadis and A. Jacquin, "Automatic face location detection and tracking for model-assisted coding of video teleconferencing sequences at low-rates," *Signal Process. Image Commun.*, vol. 7, nos. 4–6, pp. 231–248, Nov. 1995.

[12] J. Luo, C. W. Chen, and K. J. Parker, "Face location in wavelet-based video compression for high perceptual quality videoconferencing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 411–414, Aug. 1996.

[13] T. F. Cootes and C. J. Taylor, "Locating faces using statistical feature detectors," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Killington, VT, Oct. 1996, pp. 204–209.

[14] H. Li and R. Forchheimer, "Location of face using color cues," in *Proc. Picture Coding Symp.*, Lausanne, Switzerland, Mar. 1993, paper 2.4.

[15] M. Hunke and A. Waibel, "Face locating and tracking for human-computer interaction," in *Proc. Conf. Signals, Syst. and Computers*, Nov. 1994, vol. 2, pp. 1277–1281.

[16] S. Matsuhashi, O. Nakamura, and T. Minami, "Human-face extraction using modified HSV color system and personal identification through facial image based on isodensity maps," in *Proc. Conf. Electrical and Computer Engineering*, Montreal, P.Q., Canada, 1995, vol. 2, pp. 909–912.

[17] Q. Chen, H. Wu, and M. Yachida, "Face detection by fuzzy pattern matching," in *Proc. Int. Conf. Computer Vision*, Cambridge, MA, June 1996, pp. 591–596.

[18] K. Sobottka and I. Pitas, "Face localization and facial feature extraction based on shape and color information," in *Proc. IEEE Int. Conf. Image Processing*, Sept. 1996, vol. III, pp. 483–486.

[19] D. Saxe and R. Foulds, "Toward robust skin identification in video images," in *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Killington, VT, Oct. 1996, pp. 379–384.

[20] R. Kjeldsen and J. Kender, "Finding skin in color images," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Vermont, Oct. 1996, pp. 312–317.

[21] D. Chai and K. N. Ngan, "Automatic face location for videophone images," in *Proc. IEEE TENCON'96*, Perth, Australia, Nov. 1996, vol. 1, pp. 137–140.

[22] T. Cornall and K. Pang, "The use of facial color in image segmentation," in *Proc. Australia Telecommun. Networks and Applications Conf.*, Melbourne, Australia, Dec. 1996, pp. 351–356.

[23] Y. J. Zhang, Y. R. Yao, and Y. He, "Automatic face segmentation using color cues for coding typical videophone scenes," in *Proc. SPIE Visual Commun. and Image Processing*, San Jose, CA, Feb. 1997, vol. 3024, pp. 468–479.

[24] M. J. T. Reinders, P. J. L. van Beek, B. Sankur, and J. C. A. van der Lubbe, "Facial feature localization and adaptation of a generic face model for model-based coding," *Signal Process. Image Commun.*, vol. 7, no. 1, pp. 57–74, Mar. 1995.

[25] D. Chai and K. N. Ngan, "Extraction of VOP from videophone scene," in *Proc. VLBV'97 Conf.*, Linköping, Sweden, July 1997, pp. 45–48.

[26] H. P. Graf, E. Cosatoo, D. Gibbon, M. Kocheisen, and E. Petajan, "Multi-modal system for locating heads and faces," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Killington, VT, Oct. 1996, pp. 88–93.

[27] "Information technology—Coding of moving pictures and associated audio—For digital storage media up to about 1.5 Mbits/s—CD 11172," ISO/IEC MPEG, Dec. 1991.

[28] "Information technology—General coding of moving pictures and associated audio information: Video," Draft Int. Standard, ISO/IEC 13818-2, ITU-T Rec. H.262, Nov. 1994.

[29] "Video coder for audiovisual services at $p\times$ 64 kbit/s," ITU-T Rec. H.261, Mar. 1993.

[30] "Video coding for low bitrate communication," ITU-T Rec. H.263, May 1996.

[31] CCITT Study Group XV, "Document 525, description of reference model (RM8)," June 9, 1989.

Douglas Chai (S'91) was born in Kuching, Malaysia, in 1973. He received the first class honors degree in electrical and electronic engineering from the University of Western Australia, Australia, in 1994, where he currently is pursuing the Ph.D. degree with the visual communications research group.

His research interests are in image compression, video coding, image segmentation, and facial image analysis.

Mr. Chai received the Australian Postgraduate Award and the Telstra Research Laboratories Postgraduate Fellowship Award.

King N. Ngan (M'79–SM'91), for a photograph and biography, see p. 3 of the February 1999 issue of this TRANSACTIONS.