World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:6, 2008

Face Tracking using a Polling Strategy

Rodrigo Montufar-Chaveznava

***Abstract*—**The colors of the human skin represent a special category of colors, because they are distinctive from the colors of other natural objects. This category is found as a cluster in color spaces, and the skin color variations between people are mostly due to differences in the intensity. Besides, the face detection based on skin color detection is a faster method as compared to other techniques. In this work, we present a system to track faces by carrying out skin color detection in four different color spaces: HSI, YCbCr, YES and RGB. Once some skin color regions have been detected for each color space, we label each and get some characteristics such as size and position. We are supposing that a face is located in one the detected regions. Next, we compare and employ a polling strategy between labeled regions to determine the final region where the face effectively has been detected and located.

***Keywords*—**Tracking, face detection, image processing, color spaces.

I. INTRODUCTION

FACE detection is actually a very active research area in the computer vision community. The quickly expanding research in this area is based on premise that information about a person can be obtained from images.

The detection of faces has a wide range of applications such as video surveillance, face recognition, human computer interface, image database management and tracking. Detecting faces is crucial in these identification applications.

A first action in the face processing system is detecting the locations where faces are present. However, face detection from a single image is a challenging task due to variability in scale, location, orientation and pose. Facial expression, occlusion, and lighting conditions also change the overall appearance of faces [1]. It is important to consider the following: most face recognition algorithms assume that the face location is known and, in analogy, face-tracking algorithms often assume the initial face location is known [2].

According to [1], *face detection* can be defined as follows: Given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face. Human skin colors are a special category of colors, distinctive from colors of most other natural objects. This category has been found as a cluster in color space, and skin color variations between people are mostly due to differences in intensity.

Manuscript received October 11th, 2006.

Rodrigo Montufar-Chaveznava is with the ITESM Campus Santa Fe, Av. Carlos Lazo 100, Col. Santa Fe, Del. Álvaro Obregón, 01389 México DF. México (phone: +52 55 91778000 ext 7815; e-mail: rmontufar@itesm.mx).

The use of skin color as a characteristic for detecting faces has several advantages. For example, color processing is faster than processing other face characteristics; and under certain lighting conditions, color is orientation invariant. This property makes easy the development of some applications.

In this work, we use four color spaces to detect faces: HSI, YCbCr, YES and RGB. We detect skin color clusters in every color space, and supposing the face will be located in certain region and that it will not exceed certain area we determine a face region, one for every color space. Next, we apply a polling strategy in obtained regions to determine the final face position in image. It means, if the face region was detected in almost the same position in two or more color spaces, the region is valid, else the process is repeated.

The process describe above is performed for every frame in a video sequence obtained from a PT camera. When a face region position change is detected, the camera will move pretending always the face remains located in the horizontal image center. In order to avoid abrupt movements when a person moves, we make discrete the horizontal camera axis in five positions and employ some delays and memory in processing.

II. COLOR SPACES

The study of skin color classification has gained a lot of attention in recent years due to the active research in content-based image representation. For instance, the ability to represent a face as an image object can be exploited for image coding, editing, indexing or other user interactivity objectives [3].

The most popular algorithm for face detection is based on color skin information. Then, skin color classification has become an important task. The principal research in skin color based face detection is founded on RGB, YCbCr and HIS color spaces, but the presented work is also based on YES color space.

We have selected these color spaces because they are very robust, it means, they are reliable to light variations. Next we will describe them.

A. RGB

The RGB color model is an additive model where red, green, and blue lights are combined in different ways to create other colors. The abbreviation "RGB" comes from these three primary colors. The RGB color model does not define what exactly is meant by "red", "green" and "blue", in consequence same RGB values can describe noticeably different colors on different devices employing this color model. It means, while they share a common *color model*, their actual color spaces can vary considerably.

A three-dimensional cube represents the RGB model where red, green and blue are located at the corners of each axis (Fig. 1). The black is at the origin and the white is at the opposite end of the cube. The gray scale trails the line that goes from black to white. In a 24 bits color graphics system where each channel has 8 bits per color, the red is (255, 0, 0), and on the color cube, it is (1, 0, 0). The RGB model simplifies the design of computer graphics systems but is not the best for all applications. The red, green and blue color components are highly correlated. Then, it is difficult to carry out some image processing algorithms. Many processing techniques, such as histogram equalization, only work on the intensity component of an image.

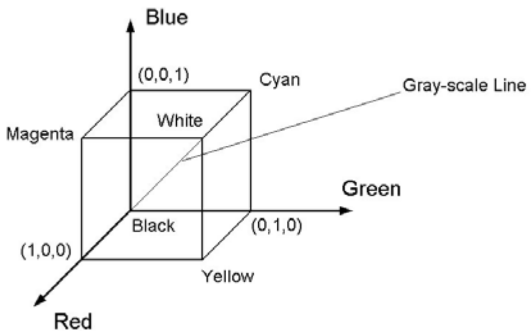

Fig. 1 The RGB Color Cube

B. YCbCr

YCbCr belongs to a family of color spaces used in video systems. Y is the luminance component and Cb and Cr the chrominance components. YCbCr is sometimes abbreviated to YCC. The family includes others spaces such as YUV and YIQ. YCbCr is the digital color system, while YUV and YIQ are the analog spaces for the respective PAL and NTSC systems. These color spaces separate RGB into luminance and chrominance information and they are useful in compression applications. However, the specification of colors is fairly inaccurate. YCbCr signals are created from the corresponding gamma-adjusted RGB source using two defined constants Kb and Kr as follows:

$$Y' = Kr \times R' + (1 - Kr - Kb) \times G' + Kb \times B'$$
$$Cb = 0.5 \times (B' - Y') / (1 - Kb)$$
$$Cr = 0.5 \times (R' - Y') / (1 - Kr)$$

where Kb and Kr are ordinarily derived from the definition of the corresponding RGB space. R', G' and B' are assumed to be nonlinear and to nominally range from 0 to 1, with 0 representing the minimum intensity (e.g., for display of the color black) and 1 the maximum (e.g., for display of the color white). The prime symbols denote the use of gamma adjustment. The resulting luminance (Y) value will then have a nominal range from 0 to 1, and the chrominance color-difference (Cb and Cr) values will have a nominal range from -0.5 to +0.5. Inverting the above equations can readily derive the reverse conversion process.

Equations defining YCbCr are formed in a way that rotates and scales the entire nominal RGB color cube to fit within a larger YCbCr color cube. There are some points within the YCbCr color cube that cannot be represented in the corresponding RGB domain. This causes some difficulty in determining how to correctly interpret and display some YCbCr signals. To figure out the relationship between the RGB and YCbCr color space cubes, Fig. 2 shows the RGB color space contained within the YCbCr color space where the color outside the RGB cube are not valid as mentioned.

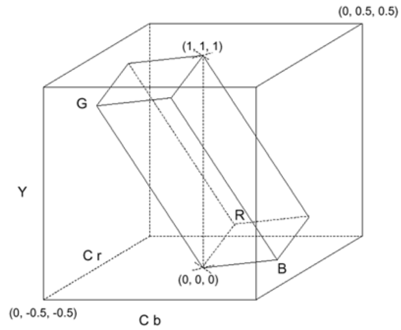

Fig. 2 The RGB color space within the YCbCr color space. Colors outside the RGB cube are not valid

C. HSI

Hue, saturation and intensity are three properties used to describe color, then there is a corresponding color model: HSI. When using the HSI color space, it is not necessary to know what percentage of blue or green is required to produce a color. We adjust the hue to obtain the color. To change a deep red to pink, we adjust the saturation. And to make the color darker or lighter, we alter the intensity.

Many applications use the HSI color model. Machine vision uses HIS color space to identifying the color of different objects. Image processing applications such as histogram operations, intensity transformations and convolution operate only on an intensity image. These operations are performed with much ease on an image in the HIS color space.

HIS can be modeled in cylindrical coordinates (Fig. 3). The hue (*H*) is represented as the angle 0, varying form 0º to 360º. Saturation (*S*) corresponds to radius, varying from 0 to 1. Intensity (*I*) varies along the *z* axis with 0 being black and 1 being white. When *S* = 0, color is a gray value of intensity 1. When *S* = 1, color is on the boundary of tope cone base. The grater the saturation, farther the color is from white/gray/black (depending on the intensity). Adjusting the hue will vary the color from red at 0º, through green at 120º, blue at 240º and back to red at 360º. When *I* = 0, the color is black and therefore *H* is undefined. When *S* = 0, the color is grayscale. *H* is also undefined in this case. By adjusting *I*, a color can be made darker or lighter. By maintaining *S* = 1 and adjusting *I*, shades of that color are created.

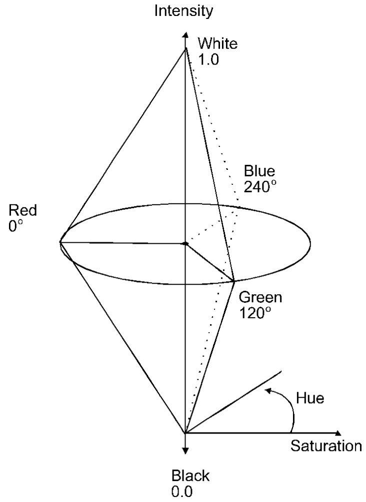

Fig. 3 The HIS color space

D. YES

The YES color space is defined as follows:

$$\begin{bmatrix} Y \\ E \\ S \end{bmatrix} = \begin{bmatrix} 0.253 & 0.684 & 0.063 \\ 0.500 & -0.500 & 0.000 \\ 0.250 & 0.250 & -0.500 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where the Y channel is the luminance component, and E, S channels are the chrominance. The YES color space constitutes a linear transformation from RGB, free of singularities and provides some computational efficiency. The E and S channels can be computed from RGB by shifting bits rather than multiplying. However, it is generally agreed that there does not exist a single color space that is good for all images [4]. The E and S channels provide a suitable space for recognition of the classes under consideration based on color in classification applications. The Y channel will be mainly utilized to model the texture information.

III. Face Detection

In this section we describe the algorithm employed to classify color skin regions. As mentioned, we consider four color spaces: RGB, YCbCr, YES and HSI. The same process is applied to every color space to obtain at most four skin regions. Next, we apply a polling strategy to determine the final region where the face is located. It is important to mention that the algorithm is considered for frames where only one face is present. The face detection process is shown in Fig. 4.

A. Segmentation

An important task in several vision applications is the image pixel classification in a discrete number of classes. The objective of segmentation is providing an effective and real-time classification.

The first step for segmentation is obtaining a set of pixel values, which corresponds to the color skin in the images. The set is obtained manually, selecting a small region directly from a frame where the face is located. Next, we employ this set to define the maximum and minimum skin pixel values for each channel of the color space. It means we will have a vector with six values as shown in Table I. This vector will be employed in subsequent classifications due the skin color constitutes a regular cluster in the color space.

Next, we classify the pixels applying a thresholding operation to the frame. Thresholding involves the use of the vector values in the corresponding three-dimensional color space. Every vector values can be considered as a class (interest object defined by a color, it means, the characteristic pixel). The classification operation evaluates whether a pixel belongs to a class or not.

TABLE I
Vector Composition for Color Skin

Channel 1	**Channel 2**	**Channel 3**
Max value 1	Max value 2	Max value 3
Min value 1	Min value 2	Min value 3

B. Regions

After segmentation is performed, we need to connect all pixels to produce regular regions or blobs. This process is exhaustive and can affect the real time performance.

We employ the blob coloring algorithm [5]. This algorithm usually expects to find a certain number of small objects or even only one. The algorithm is very simple to label all regions in a binary image, but it can be extended to color images applying it to every channel by separate and using threshold values within the comparisons. The threshold values for the region growing determine the color difference between neighboring pixels.

The threshold values have a large influence on the segmentation results. A small threshold value leads to a large number of small regions while with a large threshold value few large regions are calculated.

C. Localization

In this stage we detect and locate the face in the frame. We search in all labeled regions for the one that satisfies a specific area (an approximately number of pixels to form a face) and size (high and width considering the region is rectangular). After the region of interest is located, we obtain the position of the region center. Dimension and center position of the face region is computed for each color model. In this way, we will have as much regions as color spaces where the detection was successful for the polling process.

D. Polling

Finally, we validate the face detection and position in the frame by polling. Simply, we examine all the regions, and if there is a region common to at least three colors spaces, we considered such region as the detected face, in the other way, if the mentioned condition is not satisfied, the region is discarded and the process is performed to the next region or the next frame.

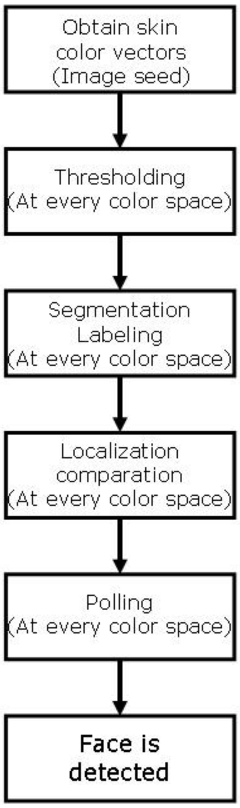

Fig. 4 The face detection algorithm

IV. Face Tracking

We employ a PT camera for face tracking. The horizontal axis is divided in five positions to determine the final face region location and avoid abrupt movements when camera is following people. Note that five horizontal positions imply the frame can be divided in six zones as show in Fig. 5.

Region center positions for every color space are located in a certain frame zone. The polling strategy determines the final face region by majority, counting the regions located in the same zone. In case of draw, none action is executed, the face detection process is repeated, while the previous results are preserved. Some times, a color space can fail when detecting faces, then none region is detected.

Fig. 6 shows when all color spaces have selected the same region and it is located at the same zone. Meanwhile, we note in Fig. 7 that it was not possible to detect the face in YES color space, but other three color spaces did it, then the final region is selected correctly.

The system always try to locate the face region at the frame center, then, depending the face position, different commands are sent to the PT camera to follow the face. The camera moves slowly and discretely from position to position in the horizontal axis avoiding abrupt movements and allowing fast corrections when person is moving around.

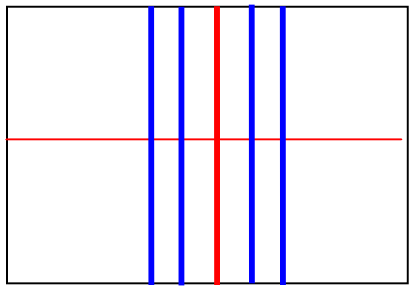

Fig. 5 Horizontal axis division in five positions. Frame division in six zones

V. Conclusion

We have developed a strategy to detect and track faces in video from the color skin characteristics. We employ four different color spaces considering every space has its own virtues and defects. The idea is to account the positive face detections by polling. In this way we are sure that a face in image has been detected correctly and can be tracked.

The presented work is part of a teleconference system where the speaker is followed by the camera during the speech. An advantage of detecting objects using the proposed method is the processing time, which is considered innocuous. This property allows real time face detection, considered adequate for tracking systems. Actually, the system has presented an acceptable performance and calibration is possible when light conditions change.

References

[1] M. Yang, D. J. Kriegman and N. Ahuja, “Detecting Faces in Images: A Survey”, *IEEE Trans. on PAMI*, vol. 24, no. 1, pp. 34-58, Jan 2002.

[2] R. Hsu and M. Abdel-Mottaleb, “Face Detection in Color Images,” *IEEE Trans. on PAMI*, vol. 24, no. 5, pp. 696-706, May 2002.

[3] S. K. Singh, D. S. Chauhan, M. Vatsa and R. Singh, “A Robust Skin Color Based Face Detection Algorithm”, *Tamkang Journal of Science and Engineering,* vol. 6, no. 4, pp. 227-234, 2003.

[4] J. Liu and Y. H. Yang, “Multiresolution color image segmentation”, *IEEE Trans. on PAMI*, vol. 16, pp. 689-700, July 1994.

[5] D. H. Ballard and C. Brown. *Computer Vision*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1982.

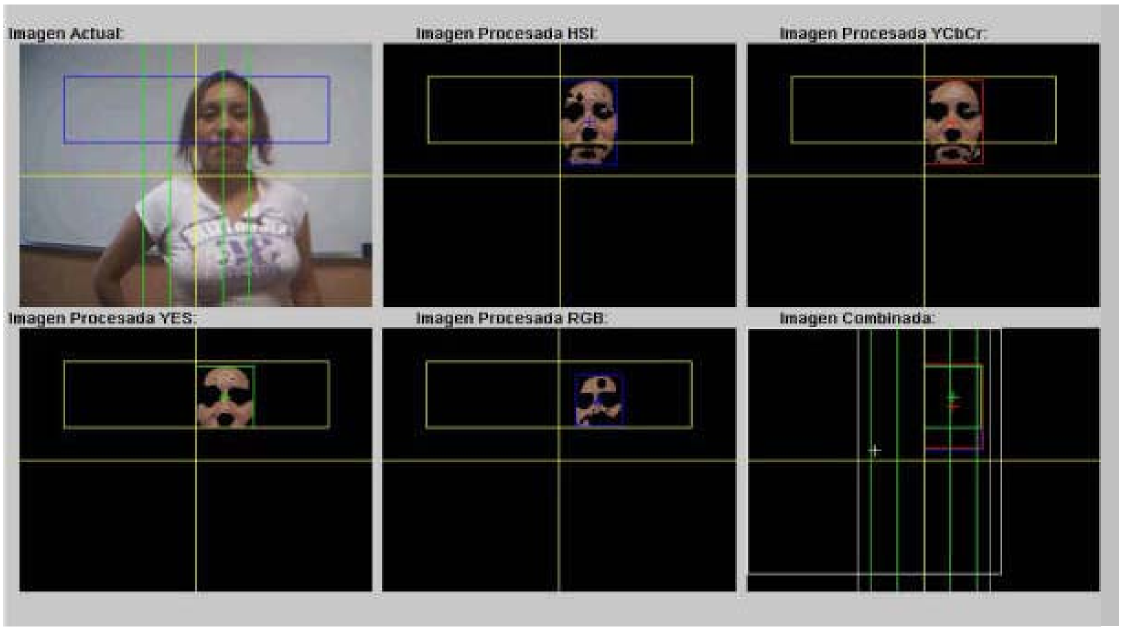

Fig. 6 Face color skin detection in HIS, YCbCr, YES and RGB color spaces

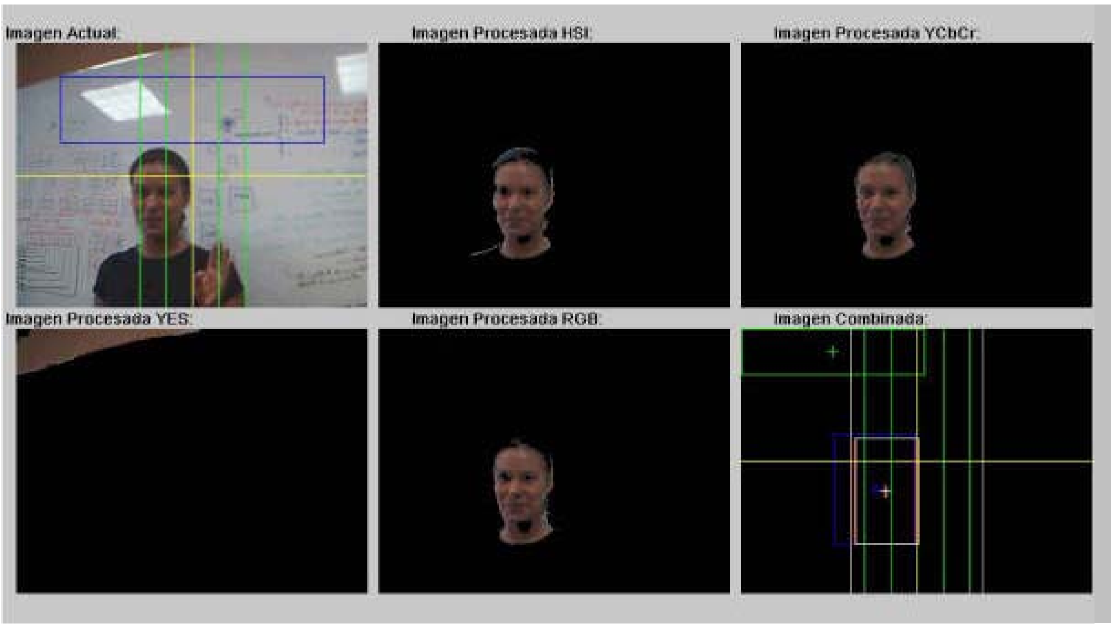

Fig. 7 Face color skin detection fail for YES, it detects the wall, but by polling, the face is detected correctly due HIS, YCbCr and RGB did not fail