Field-Programmable Custom Computing Machines
Napa, April 23-25, 2007

Hand-based Interface for Augmented Reality

F. Javier Toledo, J. Javier Martínez, J. Manuel Ferrández

Dpto. Electrónica, Tecnología de Computadoras y Proyectos
Universidad Politécnica de Cartagena
e-mail: javier.toledo@upct.es

Introduction

Augmented reality (AR) is a highly interdisciplinary field which has received increasing attention since late 90s. Basically, it consists of a combination of the real scene viewed by a user and a computer generated image, running in real time. So, AR allows the user to see the real world supplemented, in general, with some information considered as useful, enhancing the user's perception and knowledge of the environment.

In this paper, a hand-based interface for mobile AR applications is described. It detects the user hand with a pointing gesture in images from a camera placed on a head-mounted display worn by the user, and it returns the position in the image where the tip of the index finger is pointing at. Our approach is based on skin color, without the need of glove or colored marks.

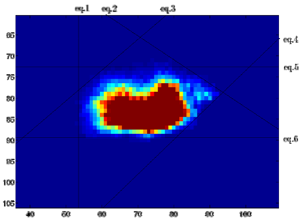
Skin recognition

Human skin color has proven to be a useful cue in applications related to face and hands detection and tracking, and skin color segmentation has become the first step in several processing tasks. The color feature is pixel based and therefore it allows fast processing. Besides, its orientation and size invariance confer high robustness on geometric variations of the skin-colored pattern.

When building a skin color classifier, two main problems must be addressed: the choice of the most suitable colorspace and the modelling of the skin color distribution.

The transformation of the image data into another colorspace is aimed at achieving invariance to skin tones and lighting conditions. In this work, the following colorspaces have been evaluated: RGB, normalized RGB, YCbCr (601 standard), HSV, YUV, YIQ and TSL.

To model skin color, two different statistical solutions have been adopted: one based on explicitly defined rules (Fig. 1) and the other on a look-up table derived from the histograms



eq1: $x > 54 + c_1$
eq2: $y > 0.4x + 42 + c_2$
eq3: $y > -0.8x + 112 + c_3$
eq4: $y < -0.8x + 156 + c_4$
eq5: $y > 71 + c_5$
eq6: $y < 90 + c_6$

Fig. 1. Rules-based classifier. The three 2D histograms of each colorspace have been analyzed and the boundaries of the skin cluster have been defined through a number of line equations, which define a closed area in the 2D histogram. The bias c_i allows making wider or narrower this area.

Receiver Operating Characteristic (ROC) curves (Fig. 2) have been used to evaluate the performance of both solution in each colorspace. The ROC curve shows the relationship between correct detections (pixels belonging to skin correctly classified, SC) and false detections (pixels not belonging to skin erroneously classified, NSF), providing a measure for the classifier performance that can be used to compare classifier designs.

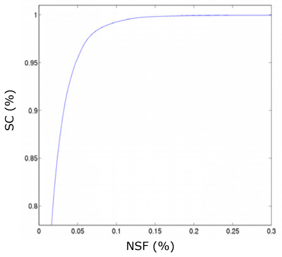


Fig. 2. ROC curve.

To merge the output of each classifier, logic AND and OR functions have been evaluated. The AND of the outputs yields a better NSF percentage at the expense of the SC percentage, whereas the OR leads to the contrary effect. ROC curves have also been used to determine the values of parameters, thresholds and logic combinations that imply the optimum set of SC/NSF ratios.

Hand gesture recognition

Once the image has been segmented the next processing task is to look for the pointing gesture, shown in Fig. 3. The solution adopted in this work consists of convoluting the binary image from the skin classifier (Fig. 3b) with three different templates: one representing the forefinger, another the thumb and the third the palm. This modularity makes easier the addition of new functionality to the system through the recognition of more gestures.

Each convolutional module sends to MicroBlaze its maximum value and its coordinates on the image (marks in Fig. 3c). A software algorithm running in MicroBlaze decides that a hand with the wanted gesture is present when the maximum of each convolution reaches a threshold and their relative positions satisfy some constraints derived from training data. Then, the algorithm returns the position of the forefinger (Fig. 3d). Otherwise, it reports that no pointing hand is detected.

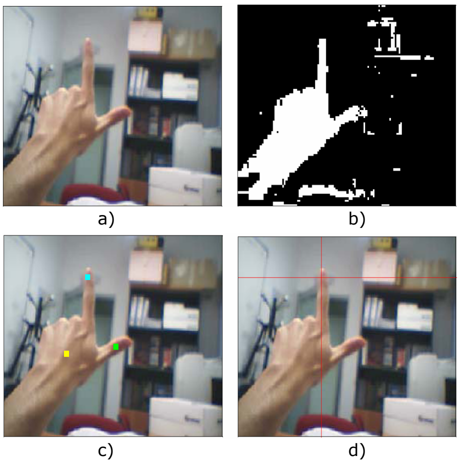


Fig. 3. Debugging output. a) original image from the camera; b) skin segmented image; c) coordinates of the maximum of each convolution; d) place where the hand is pointing.

The goodness of the gesture recognition relies upon the skin classification: if it classifies correctly the pixels the hand pointing pose is easily detected when it is present. The classifier achieves good performance ratios around 90% on SC and 10% on NSF. However, results get worse on either highly saturated or shadowed skin, where its color changes dramatically. To improve the results in these situations, an algorithm for dynamically adapting the skin classification has been developed to be executed on MicroBlaze. It tunes the biases and the thresholds of each skin classifier and the merging of their binary output images to their suitable values in order to achieve the optimum SC/NSF ratio.

The Fig. 4 depicts the block diagram of the overall system. It has been implemented on a XC2V4000 Xilinx FPGA and it can process 640×480 pixel images at more than 190 frames per second with a latency of one frame.

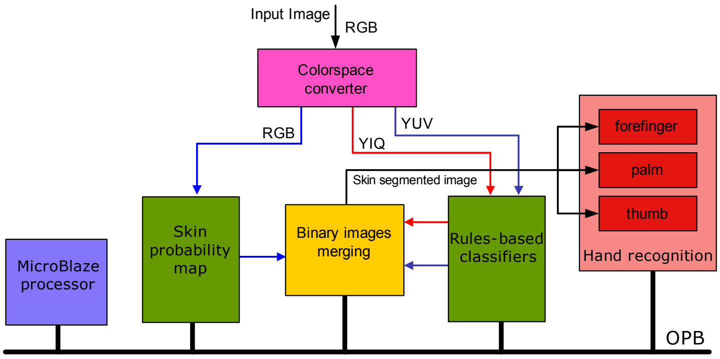


Fig. 4. Block diagram of the hardware architecture proposed.

This research has been funded by Ministerio de Trabajo y Asuntos Sociales of Spain IMSERSO RETVIS 150/06.