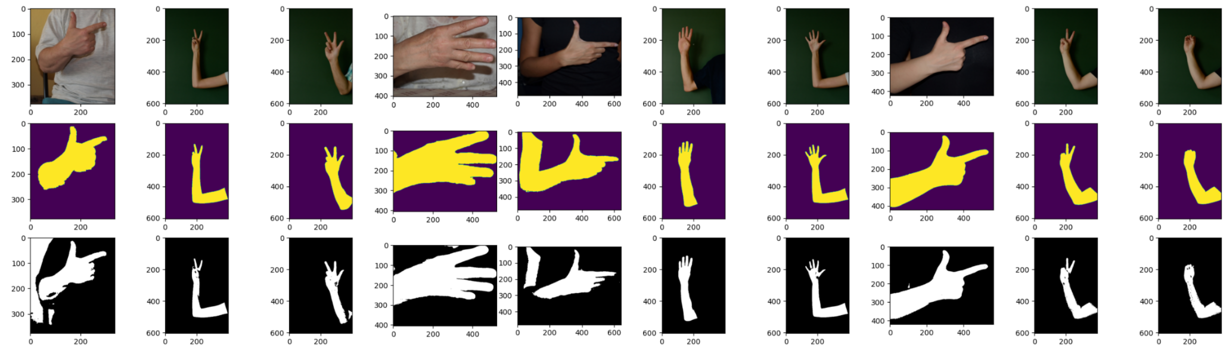
IMAGE SEGMENTATION BASED ON SKIN COLOUR FILTERING AND HAND CONTOUR

Ivan Romero and Ignacio Sancho
Universitat politécnica de Catalunya



***Abstract* — In this paper an algorithm which is able generate a mask of a hand from a given image. The method is based on hand contour and skin colour information. The method is robust regarding the hand edge detection.**

I. INTRODUCTION

It is appealing to utilize the hand directly as an input device to enable natural human-computer interaction (HCI). Gesture recognition is a current research area in vision research with applications in a wide range of areas, including surveillance, sign language translation, interactive games, performance analysis, monitoring, and remote control of home appliances, virtual reality, disability support, medical systems, and many others.

Glove-based sensing is currently the only technology that meets the sophisticated criteria of hand-based input for HCI. This technology is less common due to a number of disadvantages, including the fact that using the computer-controlled environment loses its naturalness and ease and necessitates calibration and setup processes. In order for computer vision to be used widely, a number of obstacles, such as accuracy, processing speed, and generality, must be removed. This technology has the ability to offer more natural and non-contact solutions.

In this paper how to generate a binary mask of a hand from an image is shown. The paper is organized as follows: first the general description of the algorithm explanation, afterwards the skin colour detection is explained; the contour hand detection is described in section IV, and in the last sections the results and the conclusions are given.

II. PROPOSED ALGORITHM FOR HAND DETECTION

We propose to combine skin colour detection with hand contour detection to get better results. The skin colour detector will be our main algorithm but it is not sufficient in many cases where we can find other objects or backgrounds with similar colour and it would be impossible to get a perfect masking only from this detector. So

We further improve the hand detection by simultaneously combining the skin colour filtering with edge detection to separate the different elements on the input image and get only the hand mask. The process algorithm is shown in the following figure.

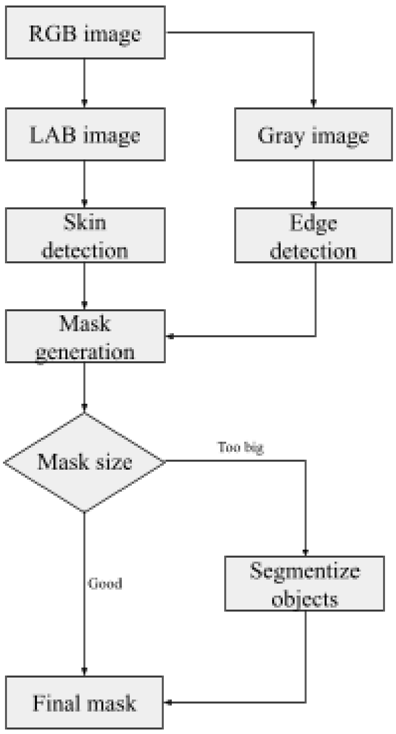


figure 1: Pipeline diagram

III. SKIN COLOUR DETECTION

SKIN SPACE OF COLOUR SELECTION

There are two reliable space colours for skin selection YCbCr and Lab space. The most frequent colour space is YCbCr since with the conversion to the Lab space there is a small loss. But after several tests we found that in the Lab space much less dispersion in the chrominance levels linked to the skin and then much easier to search for thresholds to delimitate these ones with better precision.

In the following figure is shown an example of booth colour spaces.

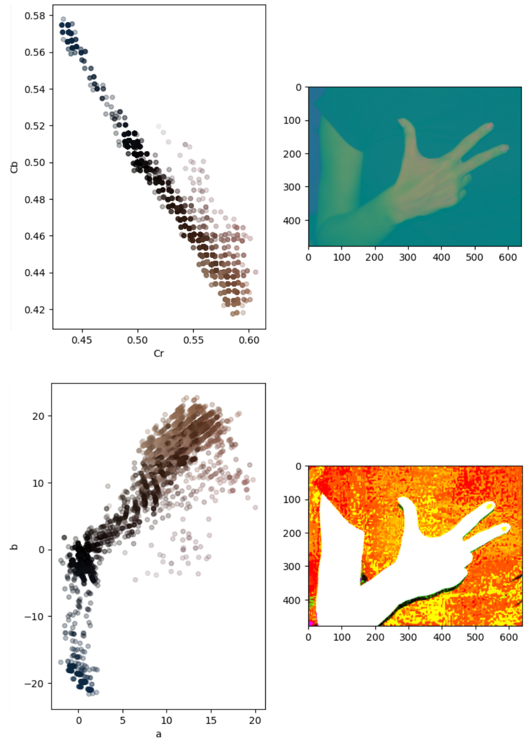


figure 2: Chrominance levels in YCrCb vs Lab

The chrominance histogram of the whole database of hand images is shown in figure 3. Using this histogram we can get the idea of which are the skin chrominance levels and then set thresholds from that information.

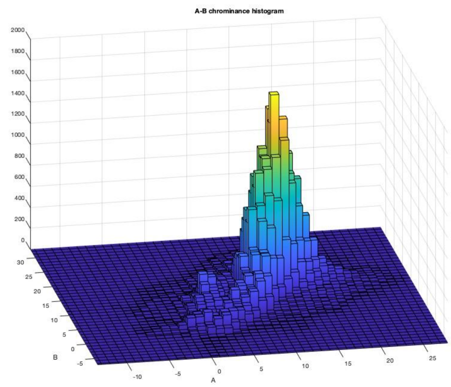


figure 3: Histogram of chrominance in YCrCb space

SEGMENTATION OF THE REGION OF THE HAND

To get a better classification of the hand in our algorithm we implement a phase segmentation where we first run a Support Vector Machine algorithm to classify the chrominance levels related to the hand and afterwards we use a threshold classifier to further reduce none interesting chrominance levels from our mask.

figure 4: Original image, Transformation to lab colour space, Predicted mask

Support Vector Machine Classification

The idea is to use a binary classifier that classifies whether it is skin or not based on the chrominance components of the training images. To do so, we have tried to implement a supervised Machine Learning model based on Support Vector Machines.

This model consists of representing each chrominance component (A and B) as a point in an n-dimensional space (where n is the number of features it has). Then, the model performs the classification by finding the hyperplane that best differentiates the two classes.

figure 5: Comparison of ideal mask with estimated mask

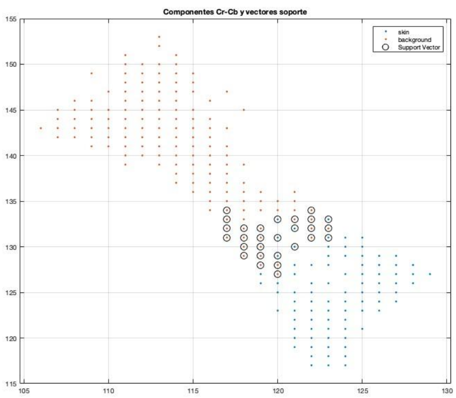


figure 6: Chrominance components

In the figure above we see the representation of the chrominance components of the image and whether it is skin or not. The decision boundaries taken by the model are also represented.

Histogram threshold classifier

After the first classification via the Support Vector Machine algorithm we can get a more accurate mask if we use a threshold classifier in order to only get the chrominance levels we are interested in from the second masked image.

IV. SYSTEM ENHANCEMENT WITH EDGE DETECTION

Simultaneously we use an edge detection algorithm which is based on the canny edge detector with a really low threshold to get a second mask from textures with lots of edges and plain backgrounds. Using this second mask allows us to discriminate possible textures with the same chrominance levels as the skin from actual skin.

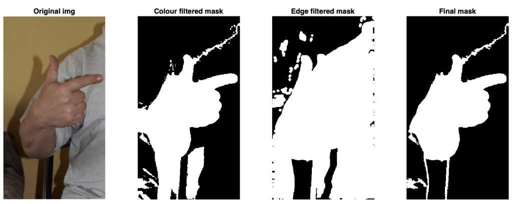


figure 7: Edge detection mask combined with colour mask

This filter can also alter perfect masks if the hand skin texture has a very flat texture but that is not the case in many pictures since the hand skin usually has lots of them from hair, fingerprints, veins, stains and more. And overall we get a better score and much better masks with less noise.

This mask is obtained by applying a

closing to the edge detected image, then closing the possible holes and afterwards applying an opening to delete small artifacts.

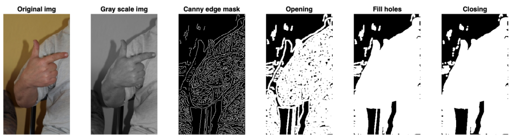


figure 8: Steps on edge detection mask

The algorithm presented has also the ability to use edge detection to separate various objects from the same mask and decide which elements are not reliable for our hand detection. This algorithm is only used in masks that are big relative to the original image.

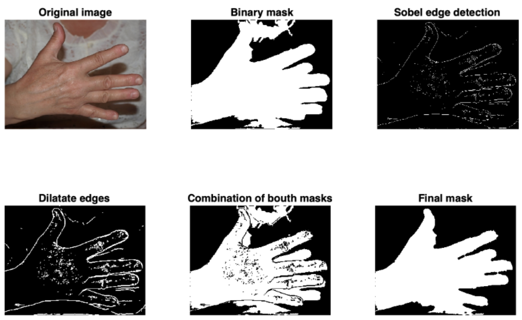


figure 9: Remove none desirable objects

This final step is based on an edge detection filter using Sobel filter with an adaptive threshold. Then it dilates the contours and subtract them to the previous mask. After we get only the hand removing the other objects by getting only the biggest object and close the holes created in the subtraction of the elements.

This post process of the mask could segmentize parts from the image like fingers so it must only be used in circumstances where the final mask can not be recognised as hand because of noise from other elements in the mask.

V. EVALUATION OF THE QUALITY OF ALGORITHMS

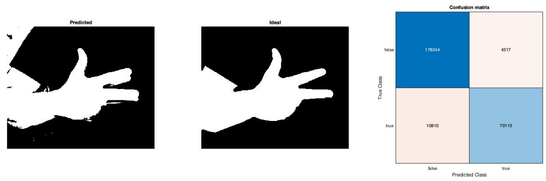


figure 10: Evaluation via precision and recall

We measure the final quality of the system with the harmonic mean of Precision and Recall, the F1 score. To calculate the total F1 of the whole database, we calculate the confusion matrix of each of the images. This matrix allows us to extract the value of the TP, FP, TN and FP.

For each image in the validation database we add up all the true positives, and the same with the true negatives, false positives and false negatives.

$$Precision = \frac{tp}{tp + fp}$$

$$Recall = \frac{tp}{tp + fn}$$

Finally, we calculated F1 for both the training database and the validation database.

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

With the provided databases we get the following score:

F1_Training = 0.9373

F1_Validation = 0.8865

The algorithm takes 2.22sec to run on a single image and 19.88 sec for the whole database.

VI. CONCLUSIONS

The segmentation works very well in images where we have clear backgrounds and good contrast with it.

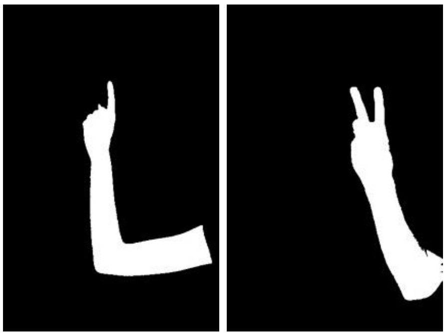


figure 11: Ideal masks

Images with different elements in the background with similar chrominance as the skin, are more difficult to get ideal masks but thanks to the edge detection we can get better results even though they are not ideal.

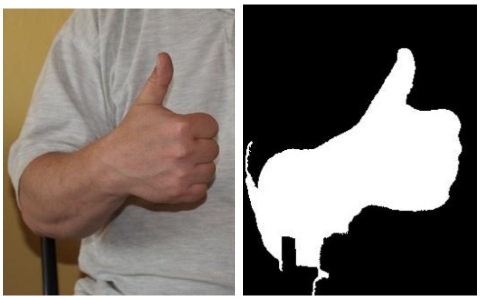


figure 12: Artifacts on masks

Another known issue as commented before is that the edge filter can eliminate some parts of the mask if the skin texture is flat but this is not a big issue since our final goal will be to count fingers from the masks and fingers are not affected by this.

figure 13: Artifacts on masks

Accurate masking of hands is a challenging task in terms of the difficulties to get an accurate binary image. This paper presented a binary masking algorithm based on skin colour and edge detection. The algorithm consists of the following steps. First, it uses a Support Vector Machine classifier model to detect skin colour of human hand in Lab colour spaces and afterwards filter again using Histogram threshold classifier. In parallel the algorithm calculates a mask of high texture regions from flat ones and combines both masks to get a more accurate prediction. Last, if the resulting mask has an exceptional big size, might be because there are other objects which have been masked and are of no interest, and the algorithm discriminates the different objects with a sobel contour filter and deletes the non desirable ones.

Github Repository