Finger Detection Based on Hand Contour and Colour Information.

Georgiana Simion*(1), Vasile Gui**(2) and Marius Otesteanu*
*Politehnica University of Timisoara / Department of Communication, Timisoara, Romania
**Politehnica University of Timisoara / Department of Communication, Timisoara, Romania
georgiana.simion@etc.upt.ro, vasile.gui@etc.upt.ro, marius.otesteanu@etc.upt.ro

Abstract—**In this paper an algorithm which is able to count the number of fingers from a hand posture is presented. The method is based on hand contour and skin colour information. The middle axes of the fingers are detected. The method is robust regard the hand edge detection.**

I. INTRODUCTION

The direct use of the hand as an input device is an attractive method for providing natural human–computer interaction (HCI).Gesture recognition is nowadays an active topic of vision research which has applications in diverse fields such as: surveillance, sign language translation, interactive games, performance analysis, monitoring, and remote control of home appliances, virtual reality, disability support, medical systems, and many others.

Since now, the only technology that satisfies the advanced requirements of hand-based input for HCI is glove-based sensing. Several drawbacks make this technology not so popular: first of all interaction with the computer-controlled environment loses naturalness and easiness and it also requires calibration and setup procedures. Computer vision has the potential to provide more natural and non-contact solutions, but has no lake of challenges including accuracy, processing speed, and general have to be overcome for the widespread use of this technology. Vision based models can be classified in two groups: 3D model based and appearance based models. The 3D hand models are articulated deformable objects with many degrees of freedom; a very large image database is required to cover all the characteristic shapes under different views. Another common problem with model based approaches is the problem of feature extraction and lack of capability to deal with singularities that arise from ambiguous views. By using appearance based methods, computer vision community tried to solve the hand gesture recognition problem by following the paradigm: hand localization, hand segmentation, feature extraction and next step hand classification, without any abstract representation in between the last two steps. View-based methods have been shown to be effective at discriminating between a certain numbers of hand poses, which is satisfactory for a number of applications in gesture recognition.

In this paper an algorithm which detects the number of fingers from a hand posture is shown. The paper is organized as follows: first related works are presented; the proposed algorithm is described in section III, and in the last sections the results and the conclusions are given.

II. RELATED WORK

The hand posture can be recognized using geometrical features such as finger tips, finger direction and hand contour. There are also non-geometric features such as colour, silhouette and texture. Silhouettes are easily extracted from local hand and arm images in the restrictive background setups. By silhouette, it is meant the outline of the hand provided by segmentation algorithms, or equivalently the partitioning of the input image into object and background pixels. In the case of complex backgrounds, techniques that employ colour histogram analyses can be used. In order to estimate the hand pose the position and orientation of the hand, fingertip locations, and finger orientation from the images is extracted. The center of gravity (or centroid) of the silhouette [1] is on choice, but it may not be very stable relative to the silhouette shape due to its dependence on the finger positions. The point having the maximum distance to the closest boundary edge, [2], [3], [4], as been argued to be more stable under changes in silhouette shape. A frequently used feature in gesture analysis is the fingertip. Fingertip detection can be handled by correlation techniques using a circular mask [5], [6], which provides rotation invariance, or fingertip templates extracted from real images [7]. Another common method to detect the fingertips and the palm-finger intersections [8],[9] is to use the curvature local maxima on the boundary of the silhouette. For the curvature-based methods, in case of noisy silhouette contours, the sensitivity to noise can be an issue. In [2], a more reliable algorithm based on the distance of the contour points to the hand position is utilized. The local maximum of the distance between the hand position and farthest boundary point at each direction gives the fingertip locations. The direction of the principal axis of the silhouettes [10, 11] can be used to estimate the finger or 2D hand orientation. All these features can be tracked across frames to increase computation speed and robustness using Kalman filters [2] or heuristics that determine search windows in the image [8], [12] based on previous feature locations or rough planar hand models. The low computational complexity of these methods enables real-time implementations using conventional hardware but their accuracy and robustness are arguable. These methods rely on high quality segmentation lowering their chance of being applied on highly cluttered backgrounds. Failures can be expected in some cases such as two fingers touching each other or out of plane hand rotations.

Jennings et al. [13] have used a more elaborate method by tracking the features directly in 3D using 3D models;

978-1-4244-9109-4/11/$26.00 ©2011 IEEE

their method has employed range images, colour, and edge features extracted from multiple cameras to track the index finger in a pointing gesture. Very robust tracking results over cluttered and moving.

Skin colour has proven to be a useful cue for hand/face detection, localization and tracking. Numerous colourspaces for skin modeling have been proposed. The most popular colour spaces are: RGB, HIS, HSV, HSL, TLS, YCrCb [14]. RGB is a colourspace originated from CRT display applications, when it was convenient to describe colour as a combination of three coloured rays (red, green and blue). It is one of the most widely used colourspaces for processing and storing of digital image data.

Hue-saturation based colourspaces were introduced when there was a need for the user to specify colour properties numerically. They describe colour with intuitive values, based on the artist's idea of tint, saturation and tone. Several interesting properties of Hue were noted: it is invariant to highlights at white light sources, and also, for matte surfaces, to ambient light and surface orientation relative to the light source and hue is also less sensitive to different skin colour. RGB, HS [15] and Hue [16] colour spaces have shown the best results for colour skin segmentation.

III. THE PROPOSED ALGORITM FOR FINGER DETECTION

Our work is based on hand edge detection. The framework diagram can be seen in figure 1.

In order to detect the hand contours, in this work John Canny's approach to optimal edge detection was fallowed. In his work [17], [18], Canny specifies three performance criteria that an edge detector should fulfill. The criteria Canny uses to detect step edges can be summarized as follows:

Good detection: Edge detection should be robust to noise; it means that should be a high probability to mark real edge points and a low probability to falsely mark non edge points. This criterion can be formalized [17], [18] by requiring the edge detector to maximize the output signal-to-noise ratio (SNR) for a given input signal-to-noise ratio.

Good localization: The points considered to be edge should be as close as possible to the center of the true edge. This criterion is formalized in [178] by requiring that the location of the maximal detector response to an edge exhibits low variance.

Uniqueness of response: The edge detector should produce only one response to a single edge. This criterion can be formalized by requiring the filter which performs edge detection to have a small spatial width [18].

In order to detect the hand contour using the Canny edge detector, the RGB image is converted in a gray scale image. To detect the contours of the hand, a high sensibility threshold is needed. Hand is a textured object, and by using a high sensibility threshold many details from the hand are also detected. These details are not useful for the next processing steps. In order to discard them, a function which counts the number of pixels from a connected object is used.

The result can be seen in figure 2 d).

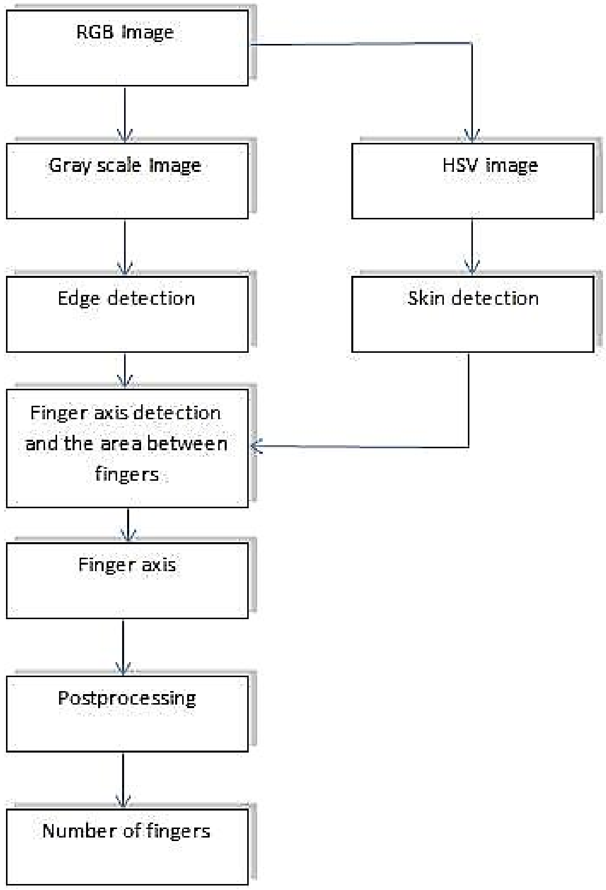
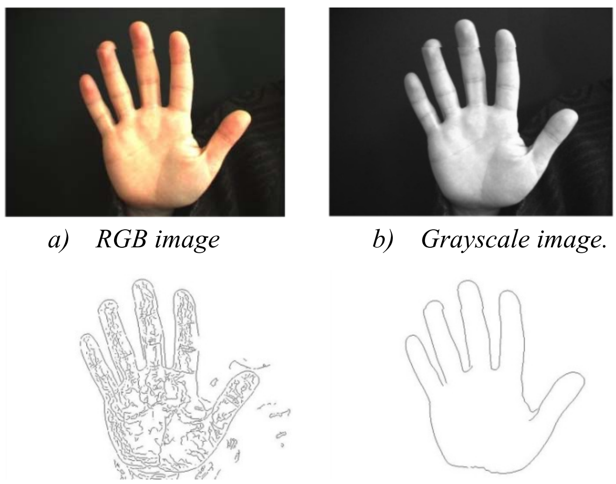


Figure 1. Framework diagram.



a) RGB image *b) Grayscale image.*

c) Canny=0.05 edge *d) Processed contour.*

Figure 2. Good hand contour.

Next step was to find the fingers middle axis. Let's consider one row of pixels from the processed contour image. The distance between two contour pixels is computed. If this distance has a value which in a specific range, than the mean value of the distance is also computed and marked. The distance represents the finger width, determined from experiments. This process is repeated for each row from the image. The purpose is to find the fingers middle axis. If we look only to the image rows, then only the fingers middle axis which are in a vertical position can be detected. In order to detect the axis of the horizontal fingers, the same procedure is followed for the image columns.

Beside the fingers middle axis also the geometrical locus of the regions between the fingers may be detected. This happens when the fingers are not too spread. Colour cue is used to decide which the fingers middle axes are. In order to define the skin model, several samples of skin were used to get a skin collage. The RGB image is converted in a HSV image. The colour space that we used was HS. The skin region is defined, by using a preset range for H (hue) and S (saturation).The H range found was [0÷0.11; 0.9÷0.99] and S range found was [0.2÷ 0.6]. The V values depended a lot on the illumination level. Because we wanted a model independent of the illumination level, the V value was eliminated.

The result is presented in figure 3.

Figure 3. Fingers axes after the image was scan horizontal and vertical.

The resulted fingers middle axes (after the colour information was used) are analyzed. As it can be seen in figure 4 a), not only the fingers middle axes are detected. In order to discard the false detections, the lines with a smaller number of pixels than α are deleted. The result can be seen in figure 4 b).

a) Finger middle axis before being computed.

b) Finger middle axis after being computed

Figure 4. Fingers middle axes after the image was scan horizontal and vertical.

Dilation with a linear structural element is used. A function which traces the region boundaries from a binary image gives the number of objects found in an image.

This work is related to our work reported in [19] and [20]. However, in [19] and [20], the fingers are detected from video sequences, the hand is segmented using background estimation and no skin colour information is used. In [20], skin detection is added to background estimation, in order to enhance the robustness of hand detection. In this work we deal only with static hand postures. The hand contour is detected with a Canny edge detector from static frames, as background estimation cannot be used. We detect pairs of Canny edges belonging to a same finger by distance and colour information. The geometrical locus of the regions between the fingers is also detected. The colour information is used to discard these false fingers axis.

IV RESULTS AND CONCLUSIONS

The algorithm was tested on a set of 50 pictures from Massey database.

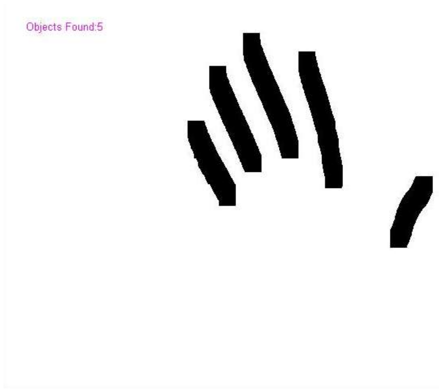


Figure 5. Number of fingers detected from the hand posture.

These pictures consist in hand postures indoor taken, while artificial light was added. In these images one, two, three, four or five fingers are shown. The pictures resolution is 640×480.

The Canny edge detector sensibility threshold, for all images, was set to 0.1.

The numbers of pixels of each connected object from the image must be greater than 150. This parameter is adapted to image resolution and hand scale. For variable hand scale, this parameter should be chosen adaptively, from scale estimation.

The width of the finger was set in the range 20 to 70 pixels. The value for H is in the range [0 0.11] respectively [0.9 0.99] and the value for S is in range [0.2 0.7]. The α value was set to 60 pixels.

The results are presented in table below. The total recognition rate is 96%. The number of rise fingers was not correctly detected for that hand posture in which the other fingers were shown. For 2 finger hand postures the other 3 fingers are visible, and when the hand contour is detected, also the contour for these three fingers might be detected. The length of these fingers is similar with the rise ones. The method is robust even when parts of the hand contour are not detected. The gap between fingers middle axes is filled using morphological operations such as dilatation.

TABLE I. THE ERROR RATE

Number of fingers	Error rate
5 fingers	0
4 fingers	0
3 fingers	0

2 fingers	0.2
1 finger	0

The images from the database can be seen in figure 6.

Figure 6. Hand postures: first row 5 fingers, second row 4 fingers, third row three fingers, fourth row 2 fingers, fifth row 1 finger.

This paper presents a method for finger detection. The hand contour is first extracted and in the next stage is computed, in order to discard the hand texture. The image is scanned horizontally and vertically; the distance between two contour points is computed and if its value is in a specific range (a range which represents the finger width), the mean distance between points is also computed and that point is marked. The skin colour information is used to check if the marked point is a hand pixel or it belongs to the background. The finger middle axis detected is computed and finally the number of fingers presented in an image is displayed.

ACKNOWLEDGMENT

(1) This paper was supported by the project "Develop and support multidisciplinary postdoctoral programs in primordial technical areas of national strategy of the research - development - innovation" 4D-POSTDOC, contract nr. POSDRU/89/1.5/S/52603, project co-funded from European Social Fund through Sectorial Operational Program Human Resources 2007-2013.

(2) This work was supported by the national grant ID 931, contr. 651/19.01.2009.

REFERENCES

[1] S.M. Sato Y. , Koike H., "Real-time input of 3D pose and gestures of a user's hand and its applications for HCI," *IEEE Virtual Reality Conference, IEEE Computer Society,*, 2001.

[2] L.J.P. Mo Z., Neumann U. , "Smartcanvas: a gesture-driven intelligent drawing desk system," *In 10th International Conference on Intelligent User Interfaces, ACM Press*, 2005, pp. 239-243.

[3] S.Y. Koike H., Kobayashi Y. , "Integrating paper and digital information on enhanceddesk: a method for realtime finger tracking on an augmented desk system," *ACM Transactions on Computer-Human Interaction* vol. 8, no. 4, 2001, pp. 307–322.

[4] S.H. Abe K., Ozawa S. , "3D drawing system via hand motion recognition from two cameras," *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, 2000, pp. 840–845.

[5] S.Y. Oka K., Koike H., "Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems," *Fifth IEEE International Conference on Automatic Face and Gesture Recognition, IEEE Computer Society*, 2002.

[6] B.F. Letessier J., "Visual tracking of bare fingers for interactive surfaces," *17th Annual ACM symposium on User Interface Software and Technology*, 2004, pp. 119–122.

[7] Z.A. O'Hagan R., "Finger track—a robust and real-time gesture interface," *10th Australian Joint Conference on Artificial Intelligence*, 1997, pp. 475–484.

[8] Z.A. O'Hagan R.G. , . Rougeaux S, "Visual gesture interfaces for virtual environments," *Interacting with Computers* vol. 14, 2002, pp. 231–250.

[9] L.J. Malik S., "Visual touchpad: a two-handed gestural input device," *6th International Conference on Multimodal Interfaces*, 2004, pp. 289–296.

[10] K.S. Segen J., " Gesture VR: Vision-based 3D hand interface for spatial interaction," *Sixth ACM International Conference on Multimedia*, 1998, pp. 455–464.

[11] S.R. Pavlovic V.I. , Huang T.S., "Invited speech: gestural interface to a visual computing environment for molecular biologists," *2nd International Conference on Automatic Face and Gesture Recognition, IEEE Computer Society*, 1996.

[12] K.B. Maggioni C., "Gesturecomputer—history, design and applications," *in: R. Cipolla, A. Pentland (Eds.), Computer Vision for Human-Machine Interaction, Cambridge*, 1998, pp. 312–325.

[13] Jennings C., "Robust finger tracking with multiple cameras," International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 1999.

[14] M.S. Kakumanu P., Bourbakis N., "A survey of skin-colour modeling and detection methods," *Pattern Recognition* 2007, pp. 1106 – 1122.

[15] M.-y.Z. Qiu-yu Zhang, Jian-qiang Hu, "Hand Gesture Contour Tracking Based on Skin Colour Probability and State Estimation Model," *JOURNAL OF MULTIMEDIA*, vol. 4, no. 6, 2009.

[16] A.S. Farhad Dadgostar "An adaptive real-time skin detector based on Hue thresholding: A comparison on two motion tracking methods," *Pattern Recognition Letters* no. 27, 2006, pp. 1342–1352.

[17] Canny J., "Finding edges and lines in images," Tehnical Reoprt AITR-720, Massachusetts Institute of Technologie, 1983.

[18] Canny J., "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, 1986, pp. 679–698.

[19] V. Gui, G. Popa, P. Nisula, and V. Korhonen, "Finger Detection in Video Sequences Using a New Sparse Representation," *Acta Technica Napocensis* – submitted December 2010.

[20] C. David,V. Gui, P. Nisula, and V. Korhonen, "Dynamic Hand Gesture Recognition for Human-Computer Interactions," *SACII 2011*