ICIUS 2010
Nov 3-5, 2010
Bali, Indonesia

ICIUS-2010-0147

Vision-based Virtual Input Using Gesture Recognition for Robotic Platform Control

Nadira Nordin[1] and M.R.Arshad[2]

*Underwater Robotics Research Group,
School of Electrical and Electronic Engineering
Universiti Sains Malaysia, Malaysia
e-mail: n_adira@yahoo.com[1] , rizal@eng.usm.my[2]*

Abstract

The aim of this research is to build a robust image-based virtual input for robotic platform control. In this paper, we propose robust real-time hand gesture recognition as the input devices to control the robotic platform. Basically, the hand gesture algorithm is built on two main steps: hand segmentation and gesture recognition. The hand segmentation is performed using skin colour as a detection cue. From the segmentation process, we extract the hand features that are required for the gesture recognition process. The fundamental model of the system is to control the robotic platform based on the number of active fingers counted from the gesture recognition process. A set of six instructions including stop, start, forward, backward, left and right is presented.

Index Terms – Gesture recognition, HCI, skin colour, virtual input.

1 Introduction

Human-computer interaction (HCI) is a communication system whereby the user interacts with a computer as part of their working, learning, communicating and recreational lives. It is also increasingly important in robotic system where HCI can give the instructions to the robot to work in a cooperative and efficient manner. Traditionally, most of the input devices are based on mechanical devices such as joystick, keyboard and mouse. But today, hand gesture has shown a lot of potential as the input for HCI. Gloved-based system for example is one of the methods for hand gesture recognition where the system requires the user to wear a glove or attach sensors on the skins to give the instruction to the computer. This glove-based analysis can give very accurate results, however the cost for the hardware is expensive, invasive and requires complex calibration. Hence, vision-based approach is a promising alternative to glove-based gesture recognition where it only needs a camera to capture the input from user. It is a most natural way of constructing a human-computer gestural interface.

R. Cipolla and N. J. Holinghurst [1] proposed a pointing-based interface for robot guidance. The system used uncalibrated stereo vision with active contours to track the position and pointing direction of a hand in real-time. Haruhisa Kawasaki [2] used the combination of HIRO control system with a finger-force control and an arm position control to maximize a hand manipulability measure to utilize the redundancy of the mechanism. M. K. Bhuyan et. al. [3] described a system which enables a robot to recognize and respond to hand gestures from a human operator. Each hand gesture represents an individual control command for the motion control of the remote robot.

The major motivation for this current research is the potential to give control instructions to the robotics platform to move based on the the number of active fingers counted from the gesture recognition process. Since the input will come from hand gestures, additional parameters are required to improve the performance. Initially, skin color segmentation is used as the detection cues since it is fast, robust and can minimize the processing time. Then gesture recognition process is performed to the segmented image to classify the gesture.

2 Methodology

2.1 Video capturing

Since this application is a vision-based system, a camera is been used to capture the hand gesture. Phillips SPC 900NC webcam with USB interface has been used as the input device. The image that captured from the webcam will send to the computer for further image processing. Figure 1 demonstrates the system setup for this system.

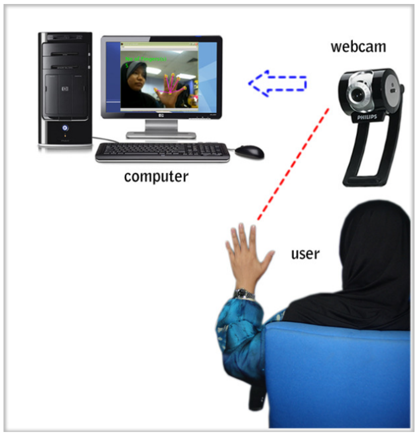

Figure 1: System Setup

2.2 Hand Segmentation

Hand segmentation can be defined as the process to extract the hand gesture from the image. There are varied proposed methods to segment the hand gesture using pattern recognition. But from reviews, detection using skin color cue is fast to robust and can minimize the processing time [4]. Therefore, in this application, skin color segmentation approaches are used as the detection instrument.

2.2.1 Color spaces

The main feature of skin color segmentation is to choose the suitable color space. Red, Green, Blue (RGB) color space is the most common color space used to represent images. However, RGB is an additive color space and it has a high correlation, non-uniformity and mixing of chrominance and luminance data [5]. Therefore RGB is not suitable for color analysis and color based recognition. As for skin color segmentation, researchers have proposed the used of YCbCr color space which contains luminance (Y) and chromaticity (CbCr) information. The separation of brightness information from the chrominance and chromaticity in YCbCr color space can reduces the effect of uneven illumination in an image [6]. Therefore, YCbCr are typically used in video tracking and surveillance.

The equation for RGB conversion to YCbCr can be seen in equation (1).

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ C_r &= R - Y \\ C_b &= B - Y \end{aligned} \quad (1)$$

2.2.2 Skin Modeling

Skin modeling is used to model the distribution of skin and non-skin color pixels. In our approach, parametric method Single Gaussian Model is used where it modeled the skin color using mean and covariance of chrominant color with a bivariate Gaussian distribution.

A set of 100 skin sample images recorded with a single Philips SPC900NC USB camera. Eight users with slightly different skin color participated in this skin modeling. The data of each sample are appended together and the mean vector and covariance matrix of the total data from the skin samples are calculated.

The skin colour distribution can be modelled by an elliptical Gaussian joint probability density function (pdf) as follows:

$$p[c/W_s] = (2\pi)^{-1} |\Sigma_s|^{-\frac{1}{2}} exp^{(c-\mu_s)^T \Sigma_s^{-1} (c-\mu_s)} \quad (2)$$

Where *c* is a colour vector that represents the random measured values of chrominance (*x,y*) of a pixel with coordinates (*i,j*) in an image. *Ws* is the class describing the skin.

$$c = \left[\underline{x}\,(i,j)\ \underline{y}(i,j)\right]^T \quad (3)$$

μ_s is the mean vector and Σ_s as in equation (5) is the covariance matrix for skin chrominance.

$$\mu_s = \frac{1}{n} \sum_{j=1}^{n} c_j \quad (4)$$

$$\Sigma_s = \frac{1}{n-1} \cdot \sum_{j=1}^{n} (c_j - \mu_s)(c_j - \mu_s)^T \quad (5)$$

The Mahalanobis distance can be used to measure the distance between *c* colour vectors to the mean vector μ_s.

$$\lambda_s(c) = (c - \mu_s)^T \Sigma_s^{-1} (c - \mu_s) \quad (6)$$

Figure 2 demonstrates the results from the skin colour segmentation model. We can see that the skin colour and non-skin colour regions are separated well in segmented image. Literally the system will also consider the face as a part of the skin colour. But since our region of interest is only the hand, so the face which has less area detected compared to the hand, is eliminated from the segmentation process. The segmented image

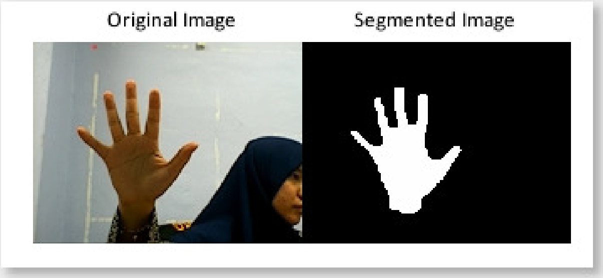

Figure 2: Result of hand segmentation

2.3 Gesture Recognition

Gesture recognition pertains to recognize meaningful expressions by humans. There are many proposed method for gesture recognition including learning based and model based methods. The learning based method uses a classifier or detector with machine learning from training data which is constructed with multi-cue features and with plenty of sample images [7].

As for this system, our intension is to calculate the number of active fingers in a particular gesture. This was inspired by Malima, Lee C. K. et al. [8] and Xiaoming Yin et al. [9] works. At this point, finding interesting points as the active fingers became the key point of this proposed method.

2.3.1 Edge detection

One of the proposed ways of selecting points of interest is corner detection. Corner is used to indicate any image feature that is useful for establishing point correspondence between images [10]. So in order to perform corner detection, the edge points of fingers are the most useful features to extract [9]. In our approach, Canny edge detection is applied to extract the edge from the segmented image. Figure 3 illustrates a hand contour which has been extracted by the Canny operator.

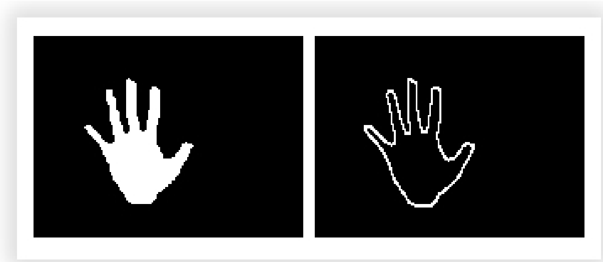

Figure 3: Segmented image extracted with Canny edge detection

However, from our observations sometimes the extraction is not smooth and there are gaps between the contours. One way to fill the gaps in the contours is by dilation. Dilation adds pixels to boundaries of the objects in an image. See Figure 4. The dilation equation is given below:

$$A \ominus B = \{ z | (B)_Z \subseteq A \} \qquad (7)$$

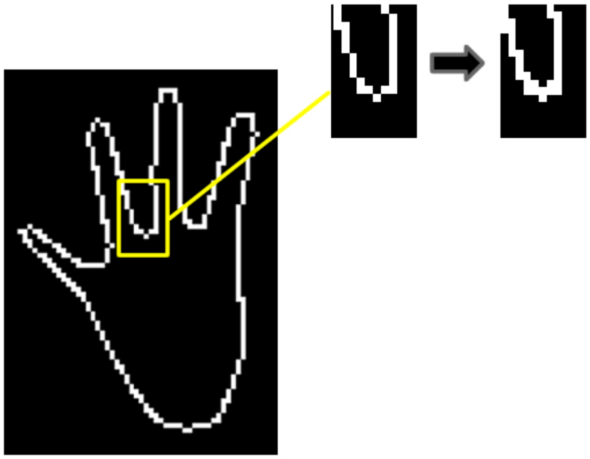

Figure 4: Dilation

2.3.2 Corner detection

Corner detection is used as the feature extraction to classify the hand. As for corner detection, the key criteria that require are, all the true corners should be detected and corner points should be well localized.

Harris corner detector is a popular interest point detector due to its strong invariance to rotation, scale, illumination variation and image noise. The Harris detector [11] is based on the following equations.

The intensity variation measure can be stated as:

$$M = \begin{bmatrix} \sum I_x{}^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y{}^2 \end{bmatrix} \qquad (8)$$

Where Σ is over a small region around a corner.

I_x is the gradient with respect to *x*.

I_y is the gradient with respect to *y*.

M is symmetric matrix that contains all the differential operators describing the geometry of the image surface at a given point *(x,y)*. The eigenvalues of *M* will be proportional to the principle curvatures of the image surface and form a rotationally invariant description of *M*. In this case, λ_1 and λ_2 are denoted as the eigenvalues of *M*.

Corner strength is specify if at a certain point the two eigenvalues of the matrix *M* are large. To classify the corner strength, Harris and Stephen have proposed the corner response function *R(x,y)* for each pixel *(x,y)* as follow:

$$R = det(M(x,y)) - \; k \, trace^2(M(x,y)) \qquad (9)$$

where, $\det M = \lambda_1 \lambda_2$

$traceM = \lambda_1 + \lambda_2$

$k = empirical\ constant$

Where *k* is denotes to remove sensitivity to strong edges.

Figure 5(a) demonstrates the corner detection extracted from an image of a hand. The Harris corner detection is marked in purple boxes. As we can see, the markers are marked at the peaks and valleys of each fingertip. The peak markers are defined as the outermost vertex points of the extracted image of a hand while the valley markers are the inner most vertex points of the extracted image of a hand.

Since our interest is to calculate only the active fingers, so we only consider the peak markers. The numbers of peak markers detected are to be used in setting the rules in hand gesture classification. In other words, the peak markers correspond to the number of active fingers in a particular gesture. So the peak markers can be discriminated from the valley markers by comparing the distance from the centroid. See Figure 5 (b)

The distance calculation is given as:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (10)$$

The peak markers are determined when the distance value is more than the threshold value. The threshold value is determined by heuristic method. Figure 5(c) illustrates the valley markers have been eliminated from the extracted image.

As the system is a real-time system, it might suffer illumination variation and random noise which cause incorrect detection of the corner detection markers. So in order to identify the correct peak markers from the extracted image, local maxima and minima approach is used as follow:

Local maximum point

$$a, \text{if } f(a) \geq f(x) \text{ when } |x-a| < \in \qquad (11)$$

Local minimum point

$$a, \text{if } f(a) \leq f(x) \text{ when } |x-a| < \in \qquad (12)$$

As a result, only local maxima markers are consider as the peak markers whilst the local minima markers will be eliminated. Figure 5(d) demonstrates that the peak markers are correctly detected and the errors in detection that occur from the illumination noise have been eliminated.

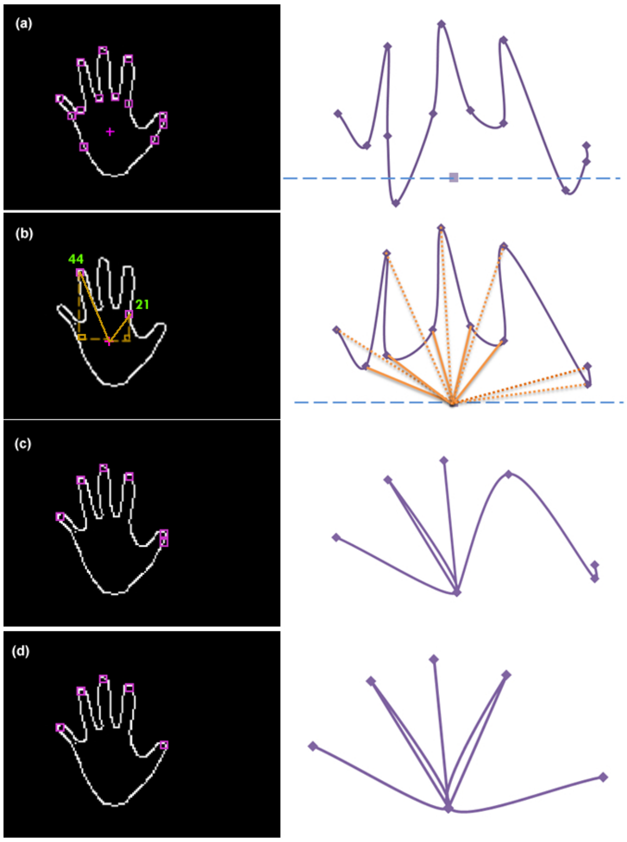

Figure 5: (a) corner detection of extracted image. (b) calculated the markers distance from the centroid to discriminate peak and valley markers (c) apply local minima and maxima to eliminate the wrong detection marker. (d) peak and valley of detected markers are correctly discriminated.

3.0 Result and discussion

As we discuss earlier, the number of peak markers detected are correspond to the number of active fingers in particular hand gesture. In this application, there is a set of six hand gesture that needs to classify and each gesture will represent the control navigation of robotic platform. See Table 1.

Table 1: The control navigation for each hand gesture based on the number of active fingers.

Number of active fingers	Direction
0	Start
1	Left
2	Right
3	Backward
4	Forward
5	Stop

This proposed technique is been tested in lab environment with standard fluorescent ceiling lighting. Hand gesture is shown in front of the webcam and six different gestures were used to control the navigation model. From the experiment, it gives 93.19% of the number of active fingers is correctly detected. Only 6.8% is false or missed recognition. But the performance will slightly decreases with poor lighting arrangement. Figure 6 illustrates the six gestures including *zero, one, two, three, four* and *five* with correspond control navigations.

4.0 Conclusion and future work

In this paper, a robust image-based virtual input for robotic platform control is presented. This marker-less system can be easily setup compared to other systems which are invasive, such as data glove. The hand gesture recognition can be initiated by skin color segmentation. In this application, the skin color and non-skin color regions are separated well using Single Gaussian Model. In gesture recognition process, corner detection plays the important role in selecting points of interest to classify the hand posture. The number of peak markers detected is used in setting the rules for hand gesture classification. . A set of six instructions can be virtually input using this hand gesture classification. Further improvement could be focused on the user variation and background robustness to make this application more efficient and robust.

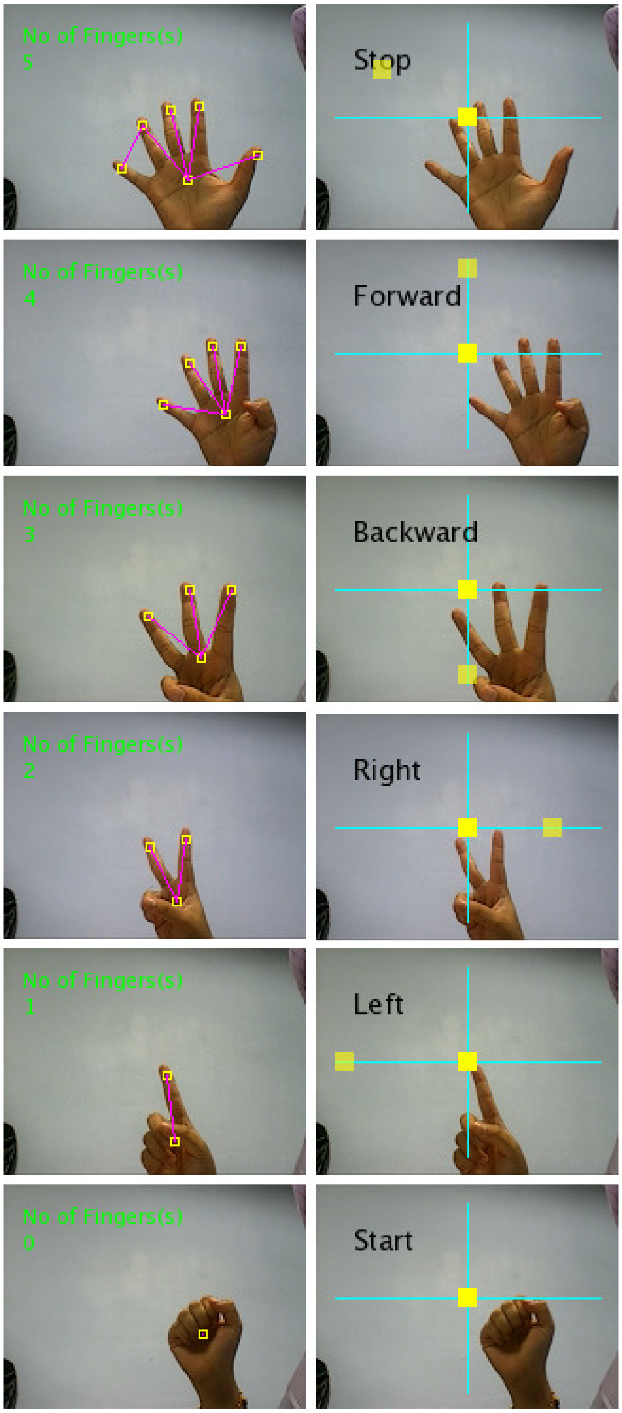

Figure 6: A set of six gestures including zero, one, two, three, four and five that correspond to the control navigations.

Acknowledgements

This part of research is funded by Malaysian government, National Oceanography Dictorate (NOD), Ministry Of Science, Technology and Innovation (MOSTI) Grant. (Grant No: 6050124)

References

1. Cipolla, R. and N.J. Hollinghurst, *Human-robot interface by pointing with uncalibrated stereo vision.* Image and Vision Computing, 1996. **14**(3): p. 171-178.
2. Kawasaki, H. and T. Mouri, *Design and Control of Five-Fingered Haptic Interface Opposite to Human Hand.* Robotics, IEEE Transactions on, 2007. **23**(5): p. 909-918.
3. Bhuyan, M.K., D. Ghosh, and P.K. Bora. *Designing of Human Computer Interactive Platform for Robotic Applications*. in *TENCON 2005 2005 IEEE Region 10*. 2005.
4. Vladimir Vezhnevets Vassili , V.S., Alla Andreeva *A Survey on Pixel-Based Skin Color Detection Techniques*. in *Proc. Graphicon*. 2003.
5. Manresa, C., Varona, J., Mas, R., Perales, F.J.,, *Real-time hand tracking and gesture recognition for human-computer interaction.* ELCVIA(5) 2005(3): p. 96-104.
6. Sebastian, P., V. Yap Vooi, and R. Comley. *The effect of colour space on tracking robustness*. in *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on*. 2008.
7. Yikai, F., et al. *Hand Gesture Recognition Using Fast Multi-scale Analysis*. in *Image and Graphics, 2007. ICIG 2007. Fourth International Conference on*. 2007.
8. Malima, A., E. Ozgur, and M. Cetin. *A Fast Algorithm for Vision-Based Hand Gesture Recognition for Robot Control*. in *Signal Processing and Communications Applications, 2006 IEEE 14th*. 2006.
9. Yin, X. and M. Xie, *Finger identification and hand posture recognition for human-robot interaction.* Image Vision Comput., 2007. **25**(8): p. 1291-1300.
10. Kenney, C.S., M. Zuliani, and B.S. Manjunath. *An axiomatic approach to corner detection*. in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. 2005.
11. Harris, C.G. *Determination of ego-motion from matched points*. in *Proc. Alvey Vision Conf.* . 1987.