Android Based Wireless Gesture Controlled Robot

Altaf Ganihar*, Shreyas Joshi*, Rahul G†, Rohini Hongal‡ and Uma Mudenagudi§
Department of Electronics and Communication
B. V. Bhoomaraddi College of Engineering and Technology, Hubli
Email: *altafganihar@gmail.com, *shreyasvj25@gmail.com, †rg619rahul@gmail.com, ‡rohini_sh@bvb.edu, §uma@bvb.edu

***Abstract*—In this paper we propose a framework for the implementation of an android based wireless gesture controlled robot. There is great demand for robotic systems in regions of operation where a direct human intervention is not possible either because of redundant nature of the task or due to the harmful nature of the region of operation. There is a constant need in the robotic industry for an improved efficiency, accessibility and accuracy in the operation of the robotic systems. In this context we propose a color segmentation based gesture controlled robot along with an ultrasonic sensor to monitor the 3D spatial co-ordinates in region of operation. The gesture controlled robot is implemented using Bluetooth communication protocol and tested under various conditions of lighting.**

I. Introduction

In this paper we propose a framework for the implementation of an android based gesture controlled robot. Recent trends in robotic industry have set high standards in the device operation of the robotic systems. There is a constant need in the robotic industry for an improved efficiency, accessibility and accuracy of the systems. There is great demand for robotic systems in regions of operation where a direct human intervention is not possible either because of redundant nature of the task or due to the harmful nature of the region of operation. Roots are either defined to be autonomous which take decisions on their own and semi-autonomous where part of the decision making is in the user's hand. The human control in the semi-autonomous robotic systems can be incorporated either through a wired protocol or a wireless protocol. Wireless protocols are more prominent in robotic systems which are to be employed in regions which are at a substantial distance from the controller. The controlling of the robotic system is carried out through physical devices. But recent trends in image processing have enabled the controlling systems to be optimized. The controlling methods involving gestures establish a natural and intuitive way of controlling the robotic system. The controlling systems for the implementation of the gesture recognition algorithms have long been personal computers. The advent of smart-phones has enabled us to build custom application for various control purposes. The smart-phones with sophisticated operating systems like Android enable us to implement state of the art algorithms for the control of the robotic systems.

There have been many methods on gesture recognition and control in the literature. The existing methods involve hand tracking and template matching for the identification of gestures. These methods do not provide robust solution for the gesture recognition as they involve a predefined template to which the existing gesture is compared. Another system utilizes the machine interface to provide real-time gestures to the robot [1]. Analog flex sensors are employed in [2] for the measurement in the deviations of the finger positions from the equilibrium. In [3] the ultrasonic sensors are employed for the gesture recognition and control. In [4] Microsoft's Kinect sensor is used for capturing user gestures to control the robotic movements. Another exiting technique is motion tracking method which involves consecutive comparison of frames or background subtraction to recognize the pattern of motion. The above methods in the literature are either computationally complex to be implemented on a portable system like a smart-phone or do not provide the efficient results required for the optimum functioning of the robotic system.

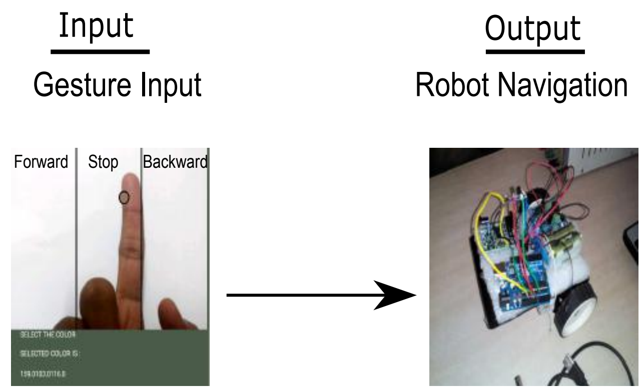


Fig. 1: Gesture based navigation system with the android application and robotic system

We propose a system to implement a gesture controlled robotic navigation system with a portable control station like a smart-phone. The main objective of the proposed system is to port the gesture control algorithm on a low end portable device like an android operating system based smart-phone. The navigation system is controlled through the gesture inputs provided by the user on a smart-phone's camera and the range sensing input provided by the ultrasonic system interfaced to the micrcontroller. The utilization of the gesture inputs from the camera co-ordinates and the range sensing enables the robotic system to analyze the region of operation in 3D dimensional co-ordinates. Hence the spatial information of the region of operation is efficiently acquired by the robotic system using the proposed set of techniques.

The physical gesture's by the users are acquires on the camera interfaced to an android based smart-phone. The physical gestures are continuously acquired for every frame and processed for recognition using the device's camera. The acquired frame is preprocessed for the gesture identification and the gesture commands for the navigation system are generated. The communication to guide the navigation system

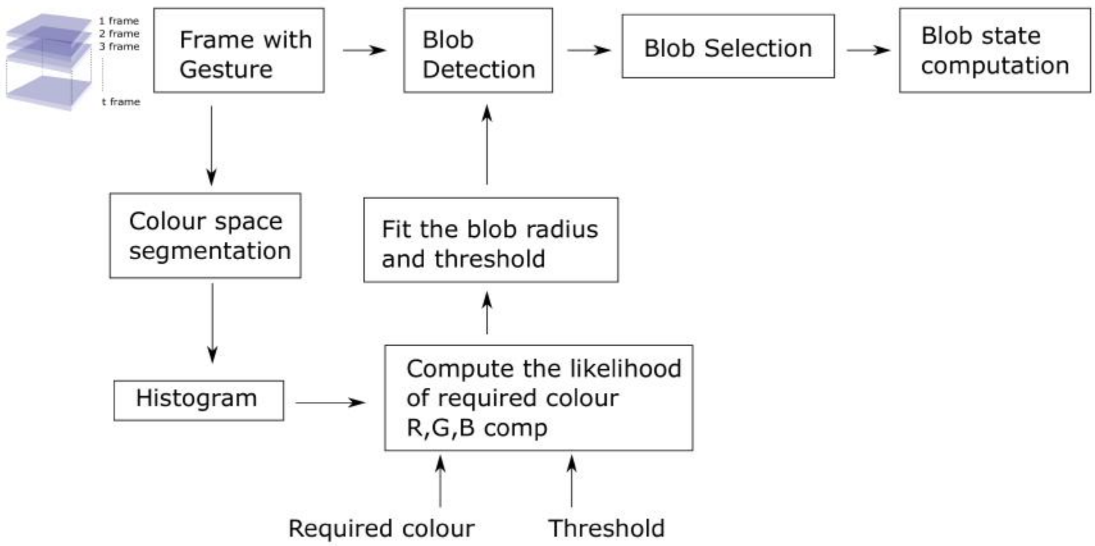


Fig. 2: Overview of the proposed Gesture Recognition System

on the robot is carried out using IEEE 802.15 standard, commonly referred as Bluetooth. The Bluetooth communication is established between the smart-phone as the master device and the bluetooth module interfaced to the microcontroller as the slave device. The navigation of the robotic system is controlled by the gesture commands received by the microcontroller via the bluetooth slave device.

II. Gesture recognition

The overview of the gesture recognition technique employed is shown in Fig 2. The physical gesture to be recognized is continuously monitored through the camera on the android smart-phone. The frame to be processed for the gesture recognition is passed through filters for preprocessing of the image. The acquire image or frame is segmented into three color spaces viz., red, green and blue color space. The segmentation is carried out in order to independently analyze the color distribution on the pixel of the image. The histogram $\mathcal{A}$ of the segmented image is computed and is modeled as a Gaussian Normal distribution [5] as given below,

$$\mathcal{A}_j(t|\theta) = \mathcal{N}(t|\mu_j, \Sigma_j) \tag{1}$$

and

$$\mathcal{N}(t|\mu, \Sigma) = \frac{1}{\sqrt{2\pi\Sigma}} e^{(-\frac{(x-\mu)^2}{2\Sigma})} \tag{2}$$

where μ gives the mean of the distribution and Σ gives the variance of the distribution and $j = 1, 2, 3$ for red, green and blue component respectively.

The likelihood of the color to be tracked is computed from the Normal distribution $\mathcal{N}_j$ and based on the input tolerance level ϵ the region of interest $\mathcal{M}$ on the distribution is established as shown in Fig 3. Based on the region of interest $\mathcal{M}$ the blob radius R_{blob} are computed. The blob radius R_{blob} is modeled as being proportional to the tolerance ϵ and n the

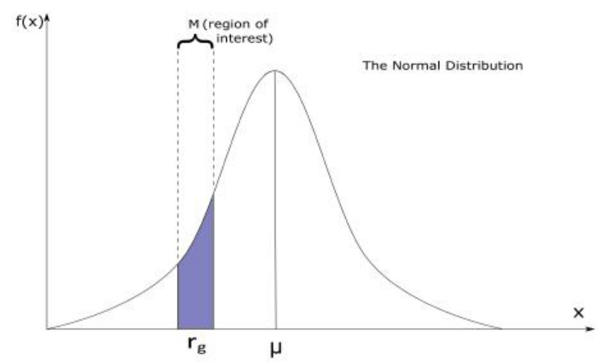


Fig. 3: The Gaussian Normal Distribution for the histogram of the image. The $\mathcal{M}$ represents the region of interest calculated based on the input color r_g to be matched and the tolerance level ϵ

size of the region of interest $\mathcal{M}$ and is given as,

$$R_{blob} = \alpha\epsilon n \tag{3}$$

where α is a constant of proportionality.

The blob detection is carried out on the original unsegmented image and blobs with the radius nearer to the blob radius R_{blob} are selected. The position of the detected blob is determined to compute the region in which the blob occurs and hence the gesture command is determined based on the position of the required color in the image frame.

III. Navigation System

The overview of the proposed robotic navigation system is as shown in Fig 5. The gesture commands once computed are then communicated from the android device to the robotic

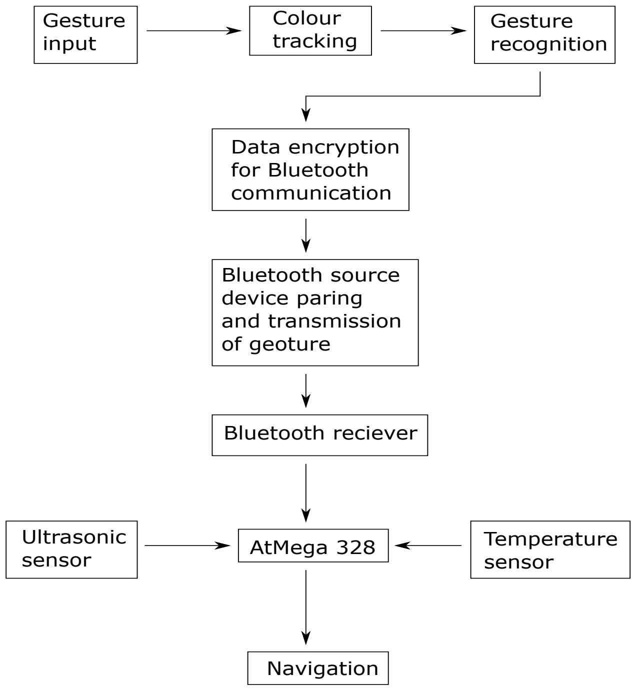


Fig. 4: Overview of the proposed Robotic Navigation System

system using Bluetooth communication system. The android device is considered as the master device for the Bluetooth communication and the bluetooth module HC-06 interfaced to the microcontroller on the robot is considered as the slave device. The Bluetooth communication between the android device and the HC-06 module is established with the help of a four digit pairing code which results in the assigning of a 43-bit unique code to the Bluetooth slave device. The gesture commands are broadcast only to the slave device based on the device ID so as to avoid redundant communication between other devices connected to the android device via Bluetooth communication.

The gesture commands received by the HC-06 Bluetooth module is in the byte array format and serially communicated to the Atmega 328 microcontroller based on an interrupt. The gesture commands in the byte array format are then converted into the ASCII format of characters by the microcontroller for decision making. The ultrasonic sensor HC-SR04 is interfaced to the microcontroller to obtain the depth measurement in the region of operation of the robotic system. The ultrasonic sesor works on the principle of SONAR which evaluate depth of a target by interpreting the echoes from ultrasonic sound waves transmitted by the sensor. Based on the gesture command and the depth measurement from the ultrasonic sensor the 3D spatial information is acquired by the robotic system for navigation. For the purpose of navigation two 12V DC motors are used along with a motor driver circuit for their control. The motor driver circuit comprises of the L293D IC for the optimum driving of the driver circuit using 12V DC supply and the control inputs from the microcontroller. LM-35 temperature sensor is also interfaced to the microcontroller to monitor the temperature of the region of operation. The temperature in the region of operation is computed using the voltage reading provided by the temperature sensor. The temperature value is communicated back to the android device via Bluetooth communication using the HC-06 module. The temperature value is converted too byte array format for transmission via Bluetooth and subsequently decoded to the floating point value and displayed in the android application on the android based smart-phone.

IV. Results and Discussion

Arduino program for controlling the movements of the robot is written in the Arduino-1.0.5 IDE on desktop having i3 processor, windows 7 professional, 64-bit and 4.00 GB of RAM memory. The arduino code is dumped on the 8-bit Atmega 328p based microcontroller having a flash programmable memory of 32 kilobytes.The android application is written using processing-2.0.1 IDE. Initially the code is compiled in Java mode of Processing IDE and then implemented in Android-mode and run on the in-built emulator supported by Processing IDE. The proposed gesture controlled automated robot is implemented on varied conditions. The timing response of the Bluetooth module HC-06 set at 9600 baud-rate is less than 300ms. The range for controlling the robot is between 0-85 meters within the coverage of the Bluetooth range. The android application is run on the following Android based phones:

1) Micromax Canvas 4 with a rear camera of 8.0 MP, Quad-core 1.2 GHz Cortex-A7 processor and 1GB of RAM.
2) SONY XPERIA M with a rear camera of 5.0 MP, Dual-core 1 GHz Krait processor and 1GB of RAM.
3) SAMSUNG GALAXY S4 with a rear camera of 13.0 MP, Quad-core 1.6 GHz Cortex-A15 processor and 2 GB of RAM.

The color tracking in the android phone is tested in varying lighting conditions. During the dull or less bright conditions, when the camera is unable to detect/distinguish colors, the flash light of camera comes handy. The color tracking is also tested using different distinguishable color bands viz. red, dark green, dark blue, violet, etc.The sensors mounted on the system constantly feed the live environmental conditions on the mobile application through the Bluetooth communication from the controller to the android phone.

Advantages of the proposed technique over the existing systems

1) The proposed method makes use of color gesture recognition to control an automated robot which makes the method more flexible for the users. The gesture control reflects an easier and a natural way of controlling unlike the old school hard/soft keys.
2) The system is reactive and real time in action. The color gesture recognition, processing the recognized color, generating command signals and Bluetooth communication with the controller all happens just within one second.
3) Being an automated robot, there is no sophisticated hardware involved other than the use of an android phone with a decent camera and the sensors. Thus the system is cost effective.

4) The system is portable as any wireless communication protocol like wifi can be used for the communication between the phone and the controller. Also the application can be ported to Windows platform, iOS platform or Java platform.

V. Conclusion

Color gesture controlled automated robot system is a new and innovative way of controlling a robot. We have proposed an efficient and a natural way of controlling the robot through the user based color gesture movements. The system is automated to provide the real time environment conditions. The temperature of the atmosphere around the robot system is constantly displayed on the android application. The system can be controlled from a remote place. This can be accomplished by setting up a server with the robot and sending gesture recognized command signals from far off places to the server. The server establishes a wireless communication with the robot and thus controls the automated robot. Instead of color gesture recognitions, speech signals can also be used as commands to control the system.

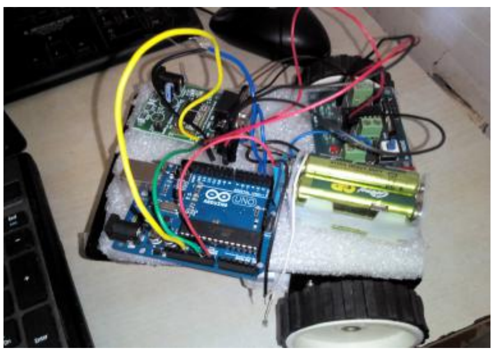
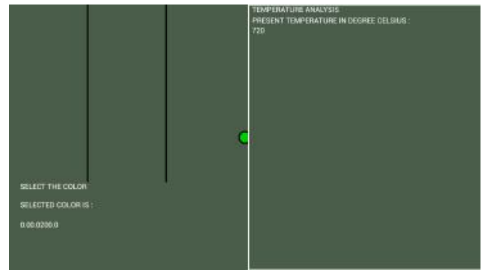
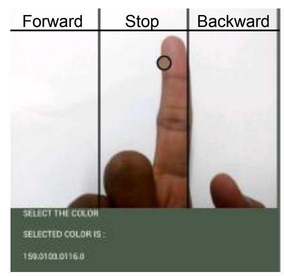


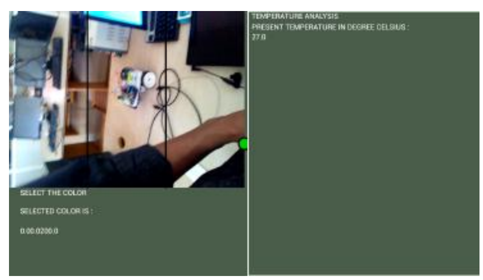
Fig. 5: Robot designed to implement the proposed technique



(a) Screen shot showing the initialization screen of the android application



(b) Screen shot showing the blob tracking algorithm to track the human skin color along with the RGB values of the tracked color



(c) Screen shot showing the android application along with the tracked color and the temperature of the surroundings of the robot transmitted via Bluetooth communication

Fig. 6: Screen shots of the developed android application for gesture control

References

[1] A. Malima, 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*, April 2006, pp. 1–4.

[2] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill, "A hand gesture interface device," in *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*, ser. CHI '87. New York, NY, USA: ACM, 1987, pp. 189–192. [Online]. Available: http://doi.acm.org/10.1145/29933.275628

[3] J. L. Raheja, R. Shyam, U. Kumar, and P. B. Prasad, "Real-time robotic hand control using hand gestures," in *Proceedings of the 2010 Second International Conference on Machine Learning and Computing*, ser. ICMLC '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 12–16. [Online]. Available: http://dx.doi.org/10.1109/ICMLC.2010.12

[4] B. MA, W. XU, and S. WANG, "A robot control system based on gesture recognition using kinect," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 11, no. 5, pp. 2605–2611, 2013. [Online]. Available: http://iaesjournal.com/online/index.php/TELKOMNIKA/article/view/2493

[5] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2001.