Computer Interaction System using Hand Posture Identification

Ms. Sushma D. Ghode
ME (WCC)
Department of CSE,
G.H.Raisoni C.E.,
Nagpur University
Nagpur-10, Maharashtra
sushdg@gmail.com

Ms. S. A. Chhabria
Assist. Prof.
Department of CSE,
G.H.Raisoni C.E. ,
Nagpur University
Nagpur-10, Maharashtra
sachhabria@ghrce.edu.in

Dr. R. V. Dharaskar
Prof. and Head
Department of CSE,
G.H.Raisoni C.E.
Nagpur University
Nagpur-10, Maharashtra
rvdharaskar@rediffmail.com

doi: 10.4156/jdcta.vol4.issue5.1

Abstract

Natural and friendly interface is critical for the development of service robots. Gesture-based interface offers a way to enable untrained users to interact with robots more easily and efficiently. Robots are, or soon will be, used in such critical domains as search and rescue, military battle, mine and bomb detection, scientific exploration, law enforcement, and hospital care. Such robots must coordinate their behaviors with the requirements and expectations of human team members; they are more than mere tools but rather quasi-team members whose tasks have to be integrated with those of humans. In this sense, this paper goal is to present a method which is used for Posture recognition technique applied in ubiquitous computing and used the efficient bio inspired posture recognition algorithm for our proposed scheme. Here we present a scheme which reduces the size of the database which is used to store different postures of human beings which are used by robot as commands. The picture frame may divide into different scan lines and pixel color value under these scan lines are examined to guess the particular posture of user. The robot may use this as command and act accordingly.

Keywords: *Hand posture recognition, Robots, Hausdroff distance, Scan lines, Posture library*

1. Introduction

Human-robot symbiotic systems have been studied extensively in recent years, considering that robots will play an important role in the future welfare society [Ueno, 2001]. The use of intelligent robots encourages the view of the machine as a partner in communication rather than as a tool [1]. In the near future, robots will interact closely with a group of humans in their everyday environment in the field of entertainment, recreation, health-care, nursing, etc. In human-human interaction, multiple communication modals such as speech, gestures and body movements are frequently used. The standard input methods, such as text input via the keyboard and pointer/location information from a mouse, do not provide a natural, intuitive interaction between models for natural and intuitive communication between humans and robots. Furthermore, for intuitive gesture-based interaction between human and robot, the robot should understand the meaning of gesture with respect to society and culture. The ability to understand hand gestures will improve the naturalness and efficiency of human interaction with robot, and allow the user to communicate in complex tasks without using tedious sets of detailed instructions. This interactive system uses robot eye's cameras or cameras to identify humans and recognize their gestures based on face and hand poses.

Vision-based face recognition systems [10] have three major components: image processing or extracting important clues (hand pose and position), tracking the facial features (related position or motion of and hand poses), and posture recognition. Vision- based face recognition system varies along a number.

Human posture recognition is gaining increasing attention due to its promising application in the area of personal health care, environmental awareness, intelligent visual human machine interface (such as video game systems and human robot interaction), to name a few. Based on commercially available image sensors and powerful personal computers, impressive research work has been reported for a variety of applications [1]. Works on human gait recognition, standing posture recognition with different arm poses [2] and dynamic gesture such as sign language recognition were presented. In general, those approaches first detect moving objects by the analysis of video stream, then extract human silhouettes using background subtraction technique. Blob metrics are represented into multiple appearance models and finally posture profiling is conducted based on frame-by-frame posture classification algorithms. Due to the complexity, these algorithms are implemented on powerful computers, even when recognizing only a small subset of human body postures, such as standing, bending, sitting and lying [3]. This will limit the use of these algorithms in real life applications. On the other hand, small and lightweight wireless platforms, such as ultra-mobile PCs or smart cellular phones, are becoming an ubiquitous computation platform [9]. Unfortunately, these devices are still unable to perform power and computation hungry object recognition tasks. In fact, there is a growing gap between the latest computer-based vision algorithms and what is actually implementable in low-complexity hardware.

The remainder of the paper is structured as follows. Section 2 we present an analysis of the most relevant identified proposal. Section 3 discusses proposed plan for system and, finally, Section 4 presents concluding remarks and future directions.

2. Analysis of selected proposals

In Shoushun Chen paper[4] the architecture of the proposed system is illustrated in Fig.1. It includes an image sensor working at temporal difference mode, a hierarchical edge feature extraction unit and a classifier with a set of library postures.

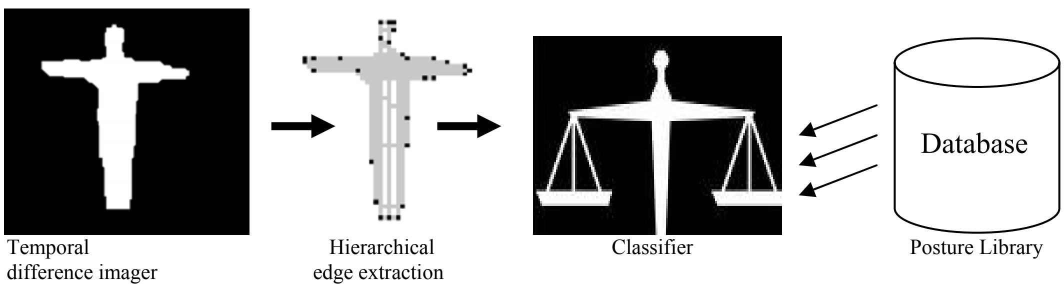

Figure 1. Block diagram of the system

The temporal difference image sensor compares two continuous image frames and only outputs addresses of those pixels with illumination variance larger than certain threshold. If the scene illumination and object reflectance are constant then the changes in scene reflectance only result from object or viewer movement. Therefore the background information is naturally filtered since the received pixels only come from the active object of interest. This shows great computational efficiency as compared to conventional image sensors used in other systems. With the address of the events, an edge feature extraction unit will reorganize the contour of the objects into vectorial line segments. The extracted line segments are fed to a modified Hausdorff distance scheme to measure the similarity of the input line segments with those of a set of library objects. The proposed classifier is able to perform size and position invariance recognition from object or viewer movement. Therefore the background information is naturally filtered since the received pixels only come from the active object of interest. This shows great computational efficiency as compared to conventional image sensors used in other systems. With the address of the events, an edge feature extraction unit will reorganize the contour of the objects into vectorial line segments. The extracted line segments are fed to posture library.

3. Proposed plan of the system

In above case we have to save each and every posture in database. So instead of managing a huge database, we are going to divide the frame into different scan lines and just examine the pixels which fall under a particular scan line. This reduces the size of database to some extent because we are going to store the coordinates of the scan lines in the form of database. Initially we are going to examine whether this plan work for the color of arm(i.e. the color of shirt) and if it is successful then we are going to implement this for entire hand posture.

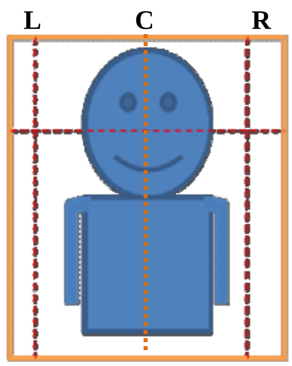

Figure 2. Initial Position

As shown in fig.3, the system scan the pixel color which fall under lines indicated as L for left hand movements and R for right hand movements respectively and accordingly takes the respective action.

3.1. Processing of Scan Lines

Following steps are used for scan lines

- **Capturing Camera View:**

First of all we find the list of driver in the project to start the capturing of the camera view in the picture box. We need to pass the driver detail to this function that is name of the driver and the ID of driver. This function will return a window object on successfully capturing the camera window.

- **Getting Current Frame out of it:**

Now we are having the camera view but the problem is that this view is handled by the O.S. and the camera driver can't work on it for processing. So we need to convert the live video in to processing format. So to get the current view as image we used a function which will convert the view in to image format which be processed by our project. Next we used a variable which can hold entire image in memory After this step we have image in variable which we can used to extract the pixel.

- **Finding Pixel RGB value:**

As we are having image in variable we have pixels not pixel RGB information. So we declare one variable which is of type Byte and by using some function we can get the pixel RGB value detail which in range of 0 to 255. Here we provide an ImageData( ) 3D array to hold pixel detail. After execution of this function we get the array filled with pixel detail where to get the R value of any pixel we can use R= ImageData (2,x,y) where X, Y is the coordinates of the pixel. This is same for all other that is G and B

- **Comparing Pixel color:**

After extracting the pixel value we compaire the value and find out which color it represent.
Eg. If value for (R,G,B) is (255,0,0) then it is Red color. Similarly (0,255,0) is Green and (0,0,255) is Blue.

- **Decision making:**

Depending upon the color value of pixel we extract the value of pixel under specific scan lines by using getnearbyvalue() function. This function take the value for red color in initR variable and imagedata() array as argument. The function returns a Boolean value which shows the pixel in current red, green and blue region and if it is found then set red color line in that region. The function picks a pixel and compares other pixel with current pixel, if value of new pixel is near to current pixel then it ignore that pixel otherwise save this value in array which is used to draw scan lines

The overall procedure is done in screenshot below. As shown in fig.3 the image, here the live image capture by camera is divided into scan lines and then area near each scan line is examined for specific commands. For example, when left hand is moved, area under left scan line is cut and hence it is detected that left hand is moved and same for other movements also. This is treated as commands for robot or any device which then work accordingly depending on the commands given to them.

Figure 3. Screenshot of processing image

Any device, one as shown in fig. 4 can used these hand posture as different commands to move forward, backward, left, right etc. this device performs various movements depending on the hand posture of the user.

Figure 4. Model which moves on hand posture

4. Conclusion

This paper reports a size and position invariant human posture recognition algorithm. The image is first acquired using an address event temporal difference image sensor and followed by a bio-inspired hierarchical line segment extraction unit. A simplified scan line algorithm is used to get the command from user. The proposed algorithm achieves 88% average recognition rate while features 10−100× computational saving as compared to conventional approach.

So in our propose system we are going to use same algorithm which may be further improved by using dynamic images using CCTV camera, instead of storing all posture images into database and then perform the same action on real-time images as explained in this algorithm.

5. References

[1] Naouki Kubota, Yu Tomioka, Toru Yamaguchi (2008) "Gesture Recognition for Partner Robot based on Computational Intelligence", *IEEE 2008*.

[2] Shoshun Chen, Fopefolu Folowosele, Dondsoo Kim (2008) "A size and position Invarient Event- based Human Posture Recognition Algorithm", *IEEE 2008*

[3] R.S.A. Brinkworth, D.C. O'Carroll "Applications for Bio-Inspired Visual Processing Algorithms" *IEEE 2008.*

[4] Shoushun Chen, Berin Martini and Eugenio Culurciello, "A Bio-inspired Event-based Size and Position Invariant Human Posture Recognition Algorithm", *IEEE 2009.*

[5] LIU Zhe LI Xiao-jiu," Image Abnormal Region Recognition with Fuzzy Clustering Based on Multi-characteristic Variable Window", *First International Workshop on Education Technology and Computer Science 2009.*

[6] A.H. Muhamad Amin, R.A. Raja Mahmood, and A.I. Khan "Analysis of Pattern Recognition Algorithms using Associative Memory Approach: A Comparative Study between the Hopfield Network and Distributed Hierarchical Graph Neuron (DHGN)", *IEEE 8th International Conference on Computer and Information Technology Workshops.*

[7] Tong wen Wang, Lin gaun, Yao Zhang "a Modified Pattern Recognition and its application in Power system Transient stability assessment*", IEEE 2008*.

[8] B.A. Kitchenham, "Procedures for Performing Systematic Reviews" *Technical Report TR/SE- 0401, Keele University, and Technical Report 0400011T.1, NICTA, 2004.*

[9] Gyu Myoung Lee, Jun Kyun Choi, Noel Crespi "Object Identification for Ubiquitous Networking", *ICACT 2009*.

[10] Tutorials on "Human Robot Interaction" from Wikipedia free encyclopedia