# Smart Wheelchair Using Mobile Accelerometer And Gesture Control

**Saurabh Chavan[1], Hemant Nikam[2], Akash Wawage[3], Madhuri Toshniwal[4]**

[1]*Student, Computer Department, Trinity Academy Of Engineering, Maharashtra, India*
[2]*Student, Computer Department, Trinity Academy Of Engineering, Maharashtra, India*
[3]*Student, Computer Department, Trinity Academy Of Engineering, Maharashtra, India*
[4]*Student, Computer Department, Trinity Academy Of Engineering, Maharashtra, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

## Abstract –

***This work a natural interaction framework for programming a mobile robot with gestures is developed using 2 cheap human interface devices accessible in the market. The use of natural motion has been growing among user interface researches and developers for contributing cozy ways of interacting with the devices around us. Some examples can be seen in how touch screens, accelerometers and image processing have influenced our interactions with cell phones, gaming devices and computers, thus, granting us to take advantage of our body ergonomics. The objective of this work is to blend an perceptive tool for tele-operating a mobile robot through gestures as an alternative input for supporting non-experts to state with robotics and ease navigation tasks. For accomplishing the objective of this work, the mobile robots programming architecture is studied and integrated with those of the user interfaces for allowing the teleoperation of the robotics device. Using this technology we will implement a wheel chair for handicapped which can be operated using accelerometer from android phones and human gestures.***

***Key Words***: Touch screen, Android, Accelerometer, Wheelchair, Gesture control.

## 1. INTRODUCTION

Inability or limitation in performing tasks, activities and roles to levels habitual within physical and social background are solved at the maximum with the help of an evergreen field called Rehabilitation Technology. A wide range of supportive devices and modern equipment has been developed to help improve their quality of life. This technology is an integration of an individual with disability into the society. Paralysis is failure of muscle operation for one or more muscles. Paralysis can be accompanied by sensory loss in the affected area if there is sensory damage as well as motor damage. Paraplegia is impairment in motor or sensory function of the lower terminus. It is commonly induced by spinal cord damage or a congenital condition such as spina bifida that affects the neural aspects of the spinal canal. The field of the spinal canal that is affected in paraplegia is either the thoracic, lumbar, or sacral regions. The proposed system helps people to overcome these defects easily by assisting with the smart wheel chair system. In the suggested dual input system, accelerometer from android phone used which gives direction based on its motion to the controller on the wheelchair. Also the camera from same android phone is used to give input as a gesture. To achieve the movement of all direction in the wheel chair, microcontroller is coded with a range of digital values.

The Android Mobile Phone Platform by Google becomes more and more popular among software developers, due to its dominant capacity and open architecture. This system uses built in accelerometer in mobile phone to control robots and games. Gesture control can also be used to control robots.

## 2. METHODOLOGY AND WORKING

### 2.1. Architecture

In the system, shown in fig.1, 3-D co-ordinates are taken as input from the accelerometer and computation is done on these co-ordinates, the result is transmitted to the controller mounted on the wheelchair through bluetooth. While using gesture control the image of gesture is captured and various algorithms like Gaussian Blur, RGB to HSV conversion, HSV thresholding and Blob Detection are applied on these images. After processing the particular movement is found and transmitted to the controller. The controller receives input from either from accelerometer or gestures, and maps the input to the H-Bridge driver and the movement is done accordingly.

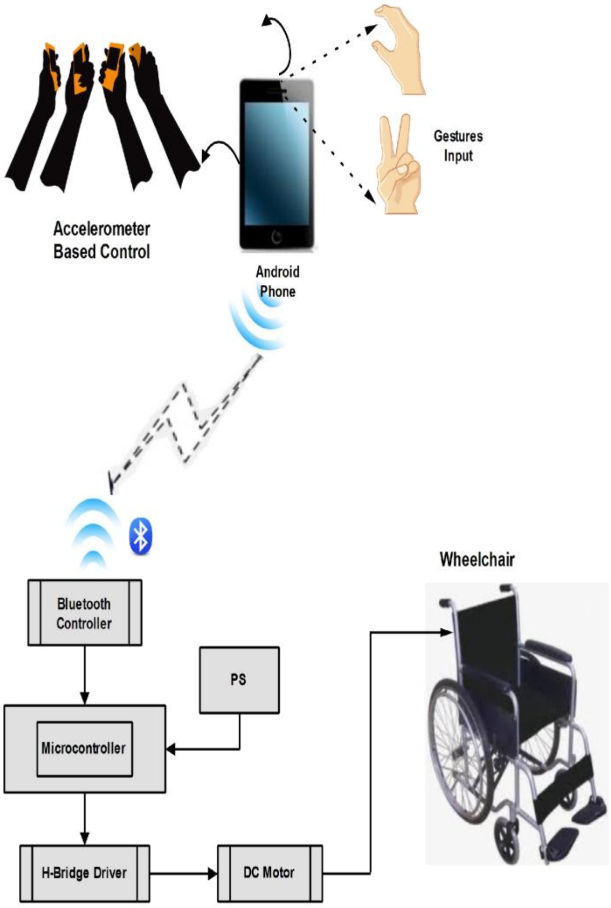

**Fig -1:** Architecture

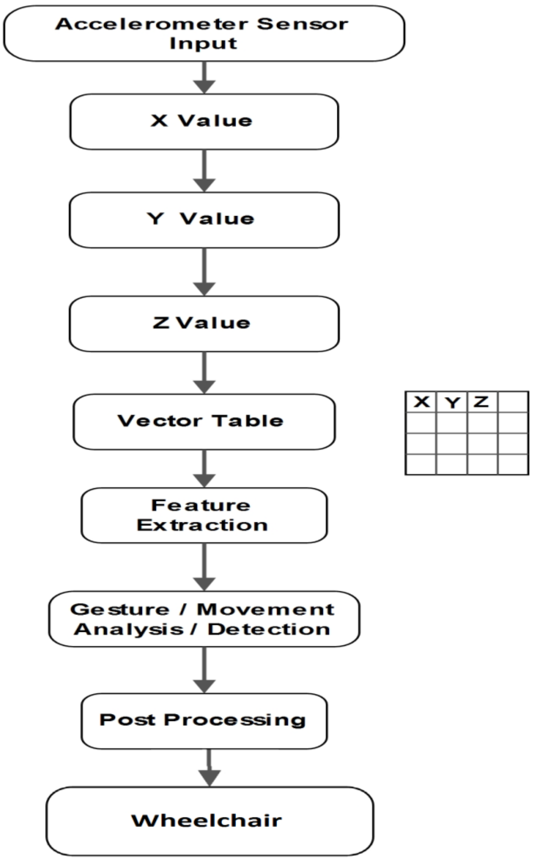

**Fig -2:** Accelerometer values workflow

## 2.2. Working of wheelchair with accelerometer

As shown in Fig 2, the accelerometer values are taken from its sensor from android phone as 3 dimensional coordinates. These coordinates are then mapped with the values in pre stored vector table. Mapping these coordinates will give the difference between actual values and vector table values. Using this difference feature extraction is done. In feature extraction depending on difference in coordinates the values from which particular direction can be detected are calculated. These calculated values are used to detect actual direction or movement of the wheelchair. Then finally in post processing the particular direction is sent to controller through Bluetooth. Controller then gives instructions to the motor to perform desired movement of the wheelchair.

## 2.3. Working of wheelchair with gesture recognition

As shown in figure 3 the accelerometer values are taken from its sensor from android phone as 3 dimensional coordinates. These coordinates are then mapped with the values in pre stored vector table. Mapping these coordinates will give the difference between actual values and vector table values. Using this difference feature extraction is done. In feature extraction depending on difference in coordinates the values from which particular direction can be detected are calculated. These calculated values are used to find actual direction or movement of the wheelchair. Then finally in post processing the particular direction is sent to controller through Bluetooth. Controller then gives instructions to the motor to perform desired movement of the wheelchair.

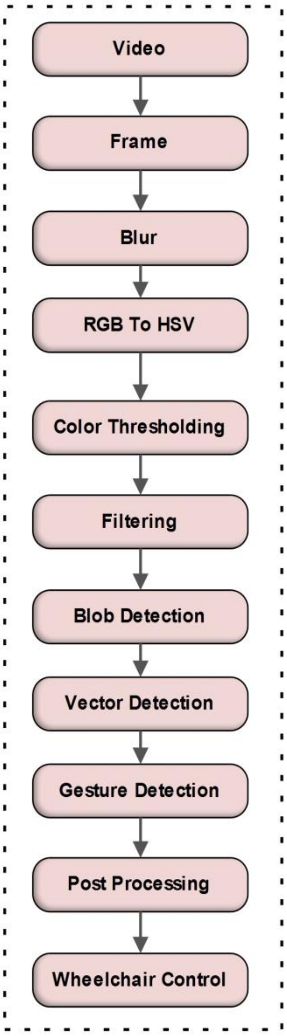

**Fig -3:** Gesture Recognition

As shown in above figure the video is captured and frames are separated from this video. Then the Gaussian Blur algorithm is applied on these frames. A Gaussian blur is the result of blurring an image by a Gaussian function. It is extensively used issue in graphics software, commonly to reduce image noise and reduce detail. Gaussian blur function is used to reduce sharpness of the image. The blurred image is in RGB format. It is the basic color model used in computers and for web graphics not for print production. Therefore, we convert the RGB image into the HSV. HSV is strong model compared to RGB as a result it offers a more perceptive representation of the relationship between colors. Thresholding is the easiest way of image segmentation. Thresholding can be used to create binary images, from a grayscale image. Midst the thresholding case, each pixels in an image are marked as "object" pixels if their value is greater than any threshold value. Images are often corrupted by random fluctuations in intensity, illumination, or have low contrast and can't be used directly. Thus filtering transforms pixel intensity values to affirm certain image characteristics like enhancement, smoothing and template matching. Blob detection refers to visual modules that are aimed at detecting points and regions in the image that vary in behavior like brightness or color compared to the surrounding. Informally, a blob is a region of a digital image in which some properties are constant or vary within a prescribed range of values; all the points in a blob can be treated in some sense to be similar to each other. After applying these algorithms, the result is then mapped with the pre-stored values and gesture detection is done.

## 3. ALGORITHMS USED IN GESTURE RECOGNITION

### 3.1. Gaussian Blur

Blurring is a very powerful operation used in image processing and procedural texture generation. Blurs involve calculating weighted averages of areas of pixels in a source image for each pixel of the final blurred image. Computing these weighted averages can be very expensive. For example, to create a blurry image you may need to touch hundreds of pixels for every pixel that you output. In this text, I'll show you some tricks for performing blurs very quickly. These tricks are employed in the blur operator in the texture generator.

Following is a simple blurring function:

Following is a simple blurring function:

```
function Blur (source, dest, radius) {
    for (y = 0; y < height; ++y) {
        for (x = 0; x < width; ++x) {
            total = 0;
            for (ky = -radius; ky <= radius; ++ky)
                for (kx = -radius; kx <= radius; ++kx)
                    total += source(x + kx, y + ky);
            dest(x, y) = total / (radius * 2 + 1) ^ 2;
        }
    }
}
```

This function blurs a source image and places the result in the destination image. A parameter of the function is the blur radius to use. This determines the size of the area of source pixels that is averaged for each blurred destination pixel.

Instead of just computing the average, many blurs use a weight for each pixel. By using weights you can give more emphasis to the pixels near the center of the summed area and achieve a more attractive blur. Since we're focused on size and speed we'll stick with just averaging and use multiple passes to achieve different quality levels.

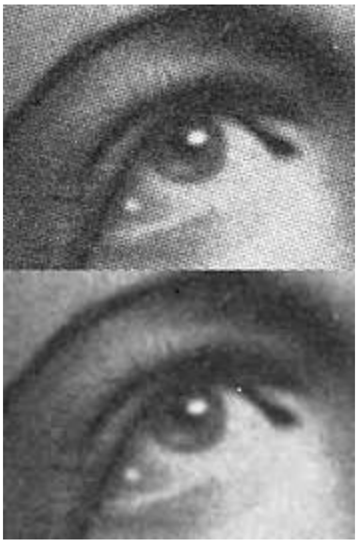

**Fig -4:** Gaussian Blur

As shown in above figure the Gaussian blur is applied on the original image (top image). After applying Gaussian blur the image looks like bottom image.

### 3.2 RGB To HSV Conversion

Color vision can be processed using RGB color space or HSV color space. RGB color space describes colors in terms of the amount of red, green, and blue present. HSV color space describes colors in terms of the Hue, Saturation, and Value. In situations where color description plays an integral role, the HSV color model is often preferred over the RGB model. The HSV model describes colors similarly to how the human eye tends to perceive color. RGB defines color in terms of a combination of primary colors, whereas, HSV describes color using more familiar comparisons such as color, vibrancy and brightness. The basketball robot uses HSV color space to process color vision.

Hue represents the color type. It can be described in terms of an angle on the above circle. Although a circle contains 360 degrees of rotation, the hue value is normalized to a range from 0 to 255, with 0 being red. Saturation represents the vibrancy of the color. Its value ranges from 0 to 255. The lower the saturation value, the more gray is present in the color, causing it to appear faded. Value represents the brightness of the color. It ranges from 0 to 255, with 0 being completely dark and 255 being fully bright. White has an HSV value of 0-255, 0-255, 255. Black has an HSV value of 0-255, 0-255, 0. The dominant description for black and white is the term, value. The hue and saturation level do not make a difference when value is at max or min intensity level. The color camera, on the android phone, uses the RGB model to determine color. Once the camera has read these values, they are converted to HSV values. The HSV values are then used in the code to determine the location of a specific object/color. The pixels are individually checked to determine if they match a predetermined color threshold.

### 3.3 Color Thresholding

The method of selecting a limit in the range of feasible colors, and assigning values above that limit to one color, and values beneath that restrict to another color, is thresholding. When an optimization program attempts to clarify a blurry area of high-contrast, it will openly differ the blurry shading into portions that are part of portions that are part of the lighter area. In order to do this, the algorithm compares the color value of individual pixel to an existing threshold. The color value of the pixel is on the 'dark' side of the threshold, pixel is changed to the full color value of the darkest color in the range. If the color value is on the 'light' side of the threshold, pixel's color value differs to the lightest value in the range.

How is Thresholding convinient?

Changing values to display contrast is extremely useful when optimizing text or other images that have clearly lay out corners. For applying the algorithm to a page with background color, or other images behind the text, it is vital to drag the text layer off the page first. Tell-tale signs of a document that has been selectively enhanced in this fashion contains:

Extremely dark, monochromatic text: no shading or other coloration in the lines of the words signifies text that has passed by the thresholding process. A "halo" of light-colored pixels around the dark lines of text: this can show that the various shades, including noise, that naturally surrounds an image has been detached using a thresholding algorithm. Both the above signs, with very dark text and very light surrounding pixels, is a good indication of the use of optimization techniques to brighten or clarify the lines or text in a document.

### 3.4. Blob detection

Blobs appear in different ways depending on their scale and can be detected using local operations in a multi-scale representation of the image Blobs occur in many shapes and places like in an image of sunflowers or in an image of a hand.

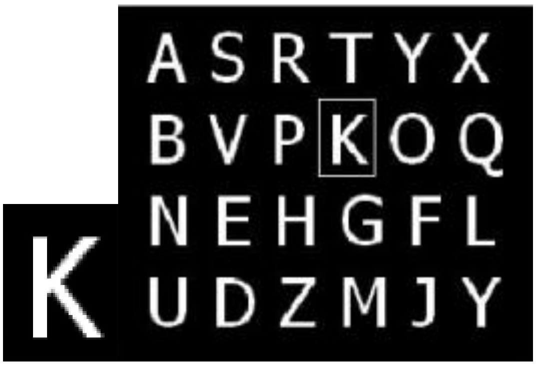

**Fig -5:** Blob detection

As shown in above figure the blob to be matched is moved over the search image and blobs are detected where the template matches a part of the image.
The following steps are performed:
1. Overlay the template on the initial image position (0,0).
2. Calculate the sum of squared differences (SSD) or the sum of absolute differences (SAD) for the overlaid area and store it in a correlation matrix.
3. Move on to the next image position and repeat step 2 until the final image position is reached.

Bright spots in the correlation image conform to likely blob locations. By defining a threshold, an exact number of blobs and definite areas can be utilized as result. At the time when the template covers pixels outside the image, those values could be calculated by mirroring or extrapolation. Otherwise, the template positions could be restricted to positions with template analysis in the image. Mini templates can be used to detect primitive blobs while large templates can discover individually shaped blobs. For getting more flexible blob detection, multiple templates could be designed.

## 4. CONCLUSION

We have designed a smart wheelchair equipped with various controllers and driven by intelligent control algorithms and a vision-based robot control interface that allows the rider to interact with the system at various levels of abstraction. This design provides a successful and fast way for handicapped people to interact with the wheelchair. The main objective of a wheelchair is to make easy life for handicapped people.

## ACKNOWLEDGEMENT

The authors would like to thank all the reviewers and advisors for their helpful suggestions to improve this paper. We would also like to give special thanks to our Prof. Sonali Kale for her valuable guidance and support.

## REFERENCES

[1] Design of wheelchair using finger detection with image processing algorithms, Chhaya.G.Patil, Sayali.K.Gharge, Sonal.V.Modhave, Y.S.Angal, International Journal of Research in Engineering and Technology ,Feb 2014.

[2] J.S. Han, Z. Zenn Bien, D.J. Kim, H.E. Lee, and J.S. Kim. Human- machine interface for wheelchair control with EMG and its evaluation. In Proc. of the 25th Annual Int. Conf. of the IEEE Engineering in Medicine & Biology Society, volume 2, pages 1602–1605. IEEE, 2003.

[3] P. Jia, H.H. Hu, T. Lu, and K. Yuan. Head gesture recognition for hands-free control of an intelligent wheelchair. Industrial Robot: An International Journal, 34(1):60–68,2007.

[4] Gonzalez, Rafael C. & Woods, Richard E. 2002. Digital Image Processing, second edition Prentice-Hall Pearson[8]. R. C. Simpson, "Smart wheelchairs: A literature review," J. of Re- habilitation Research & Development, Vol.42, No.4, pp. 423-436, 2005.

[5] "An assistive navigation system for wheelchairs based upon mobile robot obstacle avoidance," in Proc. IEEE Int. Conf. Robot. Automation, New York: IEEE Press, 1994, pp. 2018–2022.

[6] Ruize Xu, Shengli Zhou, and Wen J. Li, Fellow, IEEE, "MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognition", IEEE SENSORS JOURNAL, VOL. 12, NO. 5, MAY 2012.

[7] Prof. Vishal V. Pande, Nikita S.Ubale et.al "Hand Gesture Based Wheelchair Movement Control for Disabled Person Using MEMS.", IJERA, Vol. 4, Issue 4, April 2014.

[8] Sakshi Goyal, Ishita Sharma, Shanu Sharma, "Sign Language Recognition System For Deaf And Dumb People", IJERT, Vol. 2 Issue 4, April – 2013.

[9] Amarjot Singh, Devinder Kumar et.al "An Intelligent Multi-Gesture Spotting Robot to Assist Persons with Disabilities".

## BIOGRAPHIES

**Saurabh Chavan,** Pursuing degree in bachelor of Computer Science and Engineering at TAE from Savitribai Phule Pune University, Maharashtra, India.

**Hemant Nikam,** pursuing degree in bachelor of Computer Science and Engineering at TAE from Savitribai Phule Pune University, Maharashtra, India.

**Akash Wawage,** Pursuing degree in bachelor of Computer Science and Engineering at TAE from Savitribai Phule Pune University, Maharashtra, India.

**Madhuri Toshniwal,** Pursuing degree in bachelor of Computer Science and Engineering at TAE from Savitribai Phule Pune University, Maharashtra, India.