DOGO DOUGLAS SAVIOUR

TRAJECTORY MOTION CONTROL OF A MOVING VEHICLE USING IMAGE PROCESS ING

NEU 2017

TRAJECTORY MOTION CONTROL OF A MOVING VEHICLE USING IMAGE PROCESSING

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By
DOGO DOUGLAS SAVIOUR

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Mechatronic Engineering

NICOSIA, 2017

TRAJECTORY MOTION CONTROL OF A MOVING VEHICLE USING IMAGE PROCESSING

A THESIS SUBMITTED TO THE
GRADUATE SCHOOL OF APPLIED SCIENCES
OF
NEAR EAST UNIVERSITY

By
DOGO DOUGLAS SAVIOUR

In Partial Fulfillment of the Requirements for the
Degree of Master of Science
in
Mechatronic Engineering

NICOSIA, 2017

I hereby declare that the information in this thesis have been obtained and presented in accordance with academic rules and ethical conducts. I also declare that as required by the ethical rules and conduct, I have cited and referenced all materials and results that are not original to this work.

Name:

Signature:

Date:

ACKNOWLEDGEMENT

Firstly, my profound gratitude goes to the lord almighty God the giver and sustainer of all forms of knowledge.

My utmost gratitude and appreciations goes to my prompt and wonderful supervisor, Assist. Prof. Dr. Boran Şekeroğlu for his immense guidance in supervision as well as numerous advices given to me during the time of this research work. Indeed you are a kind, understanding and a knowledgeable supervisor. I appreciate your kind efforts.

Special thanks to my entire family, the family of Dogo Ajeye Jankasa. Indeed everyone is a contributing element in my success.

At this moment, I really want to acknowledge the government of Kaduna State, Nigeria for initiating this sponsor of which I am privileged to be a beneficiary. Forever Kaduna State have put a positive mark in my life, thank you.

Finally, my appreciation goes to the entire staff and members of the department of Mechatronic Engineering, Near East University Cyprus for grooming me into an engineer. Especially, the chairman Prof. Bülent Bilgehan, my good friend Mohamad AL-shahadat and other fellow mechatronic engineers. Everyone was a contributing factor to my success and I appreciate you all.

ABSTRACT

Successful control motion is possible if proper algorithms are followed. Trajectory motion control is one of the motion controls that require this proper algorithm. Although it appears harder but with aid of image processing algorithm, it has been brought closer to our understanding. Technology has create a great enthusiasm in the field of image processing with the advent of digital cameras which enables good procession of images into required ones with the proper algorithms. The fundamental processes involved in image processing depend on the task which has direct consequence on the choice of the appropriate technique. This provides the desirable quality of the image required since the objective of image processing is to pick out information that is trivial or to remove faults in an image.

The method adopted in this research work was guiding a moving car to a pre-defined trajectory. This involved getting snap shots of the prototype robot car with the colors of the RGB and transforming these colors to gray-scale images. The gray-scale are also transformed in to binary images using image subtraction according to threshold which enables the required intensity of the red and blue colors needed .The entire procedure of the control was successful using matrix laboratory(MATLAB) program.

Keywords: Image processing algorithm; digital video cameras; pre-defined trajectory; prototype robot car; gray-scale image; binary image; threshold

ÖZET

Yerinde algoritmalar kullanıldığı takdirde başarılı hareket kontrolü mümkün olabilmektedir. Yörünge hareket kontrolü de bu düzgün algoritmaların kullanımını gerektiren bir hareket kontrollerinden biridir. Zor gibi gözükse de, yörünge hareket kontrolü, görüntü işleme teknikleri kullanılarak insan algısına daha fazla yaklaşmıştır. Doğru algoritmaların kullanılmasıyla resimlerin daha iyi işlenmesini mümkün kılan dijital kameraların gelişimi, görüntü işleme alanında büyük bir ivme kazandırmıştır. Görüntü işleme ile ilgili temel işlemler doğru tekniğin seçilmesi ile sonuç vermektedir. Bu da, amacı görüntüden bilgi çıkarma veya daha iyi görüntüler elde etme olan görüntü işlemede, istenilen kaliteye ulaşılmasını sağlar.

Bu araştırmada kullanılan metotlar, hareket eden bir arabanın belirlenen noktaya gitmesine rehberlik etmektedirler. Bu da, prototip robot arabının Kırmızı Mavi Yeşil (KMY) renklerde anlık görüntüsünün alınmasını ve bu renkleri gri-ölçekli resme çevrilmesini içermektedir. Daha sonra gri ölçekli resimler de, eşik değerine göre görüntü çıkarma kullanılarak ikili resme dönüştürüldü. Tüm kontrol işlemleri MATLAB kullanılarak yapılmıştır.

Anahtar Kelimeler: Görüntü işleme algoritmaları; dijital video kameraları; tanımlanmış yörünge; prototip robot araba; gri-ölçekli resim; ikili görüntü

TABLE OF CONTENT

CHAPTER 4: SYSTEM COMPONENTS

CHAPTER 5: THE EXPERIMENTS AND RESULTS

CHAPTER 6: CONCLUSION

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Basically, robots navigation and motion vehicles require ideal environment for their dynamic and static state. It has been shown that there is tremendous improvements on our highways and traffic congestion have also improved dramatically (Betke et al, 2000). Conversely, areas that have proven to have a higher number of moving objects particularly urban centers, pose high challenges, therefore, movements in those type of environment posits largely unresolved problems (Ali and Shah, 2008). Amongst these problems, one tends to be the major challenge in environment full of dynamics and the possibility of predicting the state for decision making in the future when it comes to trajectory (path) planning. For an accurate motion trajectory and future location to be estimated, the object semantics such as whether the object is stationary or moving needs to be known (Ess et al, 2009 a).

In recently times, binocular stereo rigs and digital cameras have shown not to measure up in the geometrical accuracies in sensors ranges but they appear to give out a large advantage which is the delivery of information appearance in addition to the scene geometry (Ess et at, 2009 a). Moving objects following in video pictures have pulled in a lot of interest for PC vision. For protest acknowledgment, route frameworks and reconnaissance frameworks, question following is an irreplaceable initial step. Question following has centrally and continuously conditioned since it suffice a few essential applications, for example, Security and surveillance to understand individuals, to give better conviction that all is good utilizing visual data, In Restorative treatment to modify the personal satisfaction for non-intrusive treatment patients and debilitated individuals, In Retail space instrumentation to break down shopping conduct of clients to upgrade building and condition outline, Video deliberation to acquire programmed differentiation of recordings, to create protest based synopses, Activity administration to dissect stream, to distinguish mishaps, Video altering to dispense with awkward human administrator association, to plan advanced video impacts.

The Shading Picture Preparing and Protest Following Framework (Following Framework) was intended to beat these lacks. It was intended to be completely programmed, in this way expelling a portion of the mystery fi'om the investigation and also diminishing the repetitiveness of examining an extensive number of edges. Since the picture is changed over to computerized organize, it can be handled carefully, enhancing the picture and empowering less demanding discovery of edges. Now and again it might be ideal for the client to find the question physically with a mouse as opposed to have the PC do it. The Following Framework permits this. The greater part of the picture handling that can be performed in the programmed following mode can in any case be performed in the manual mode, at the client's carefulness. A few following strategies (or modes) hatchet executed. A layout coordinating strategy can be utilized as a substitute to the robotized limit based technique. The layout coordinating strategies, including connection and a subtractive method, track picture highlights as opposed to edges. A local parameters technique can track development of force based parameters, for example, power most extreme, least, and mean, in a chose stationary area. The last following strategy right now actualized is the territory estimation technique, which can be utilized to measure scaled or non scaled regions.

Various systems have been created in Picture Handling amid the final four to five contract. The greater part of the strategies are formed for upgrading pictures gotten from unamed projectile, storage tests and military supervision aviation. Picture Handling structure are obtained to be plainly prevalent in order to simplify the connectivity of dynamic work forcing the PCs, understanding the volume of store tools, representation propagation and so forth.

Current computerized invention has create it potential to lead multi-dimensional mark with frame that zone from basic advanced loop to harvest the border parallel of PCs. The goal of this observation can be separated in three classes: • Image Handling image in → picture out • Picture Examination image in → assessment out • Picture Understanding image in → abnormal state depiction out it will crowd on the principal ideas of image preparing. Area does not permit us to create more than a pair of early on comments about image inspection. Image understanding requires an approach that contrasts on a very basic level from the subject of this record.

CHAPTER 2

TRAJECTORY MOTION

2.1 A Trajectory

A trajectory or a flight path according to Saeed (2001), is the path that an object in motion follows through space as a function of time. In order to understand trajectory behavior, a motion pattern approach needs to be applied. A trajectory can be considered as an M-dimensional path which does not need to correspond to any visible object, but might correspond to a time evolution in the M-dimensional feature vector (Morris and Trivedi, 2009).

In object recognition tracking scenario, an object is anything of interest that can be used for further analysis (Yilmaz et al, 2006). Lot of objects can be tracked including cars moving on the road, planes in the sky, even people walking on the road. This tracking can be successful either by using the object shape, appearance, color, etc. the significance of this object detection cannot be over emphasized especially in this era of crimes and other form of human activities that pose threat to life in general. In real life, the tracking system enables us to perform a lot of tasks in areas like magnetic resonance imaging (MRI), Computer Tomography (CT), in shopping malls for customer surveillance. This also helps in the traffic control on highways as well as space surveillance (Yilmaz et al, 2006).

2.2 Trajectory versus Path

By definition, a path in a robotic configuration is a sequential order which does not necessarily consider time for this configuration (Saeed, 2001). Supposing an object moves from point 1 to point 2, then to another point say 3. The configurations form a trajectory.

However, trajectory concern is the timing of the attainment of this path; therefore, it specifies the timing of the path. Following this analogy irrespective of the timing, the pat from 1 to 2, and 2 to 3 remain constant. But the trajectory depends on the velocity and acceleration across these points. Hence, trajectories might vary depending on whether a moving object is on uniform velocity, or it is accelerating. Basically, moving objects in robotics are concerned with the path, than the velocities and accelerations (Saeed, 2001). A trajectory motion is a vector. The magnitude and direction of these vector is necessary in the analysis of the trajectory (Young and Freeman, 2006)

2.3 Trajectory and Vectors

Physical quantities can be described completely using a single number then a unit will be attached to it. Some of these quantities include mass, time, density, temperature etc. (Young and Freeman, 2006). Some quantities have proven to have both magnitude and direction. They are called vectors. Because they have magnitude as well as direction, they cannot be described using a single number. The motion of an airplane as an example. In the description of this motion, one must talk about the velocity and the direction of the plane. Motion of an airplane is a good example of a trajectory motion. In this trajectory motion, some physical quantities and vector quantities will be considered. The position vector in the trajectory, the length of the vector, as well as the magnitude of the vector in this trajectory motion.

The components of a vector $\vec{P}$ in the(x,y) plane is that vector which is the sum of the vector in the x direction $\vec{P}_x$ and the vector in the y direction $\vec{P}_y$ (Young and Freeman, 2006). The equation connecting this is given by:

$$\vec{P} = \vec{P}x + \vec{P}_y \tag{2.1}$$

Because all components of vectors lie within the x-coordinates, a unit number is needed to describe each of these vectors. The vector $\vec{P}_x$ is positive if and only if it points to a positive direction. The magnitude of $\vec{P}_x$ is still the vector $\vec{P}_x$ likewise the vector $\vec{P}_y$ is having a magnitude $\vec{P}_y$. we therefore say the vector $\vec{P}_x$ and $\vec{P}_y$ are the component of the vector$\vec{P}$. To determine the magnitude of these vectors, we require a resultant or modulus of the vectors to be calculated using the equation:

$$/\vec{P}/= \sqrt{(\vec{P}\mathrm{x}\,)^2 + (\vec{P}\mathrm{y})^2} \qquad (2.2)$$

The vector equation in 2.2 is valid with a positive root if x and y vectors are mutually perpendicular. The direction of this comes from taking the tangent of the angle between x and y. If the angle between these vectors is α taken from the positive x and y axis, then'

$$\mathrm{Tan}\alpha=\frac{P_y}{Px} \qquad (2.3)$$

Clearly, it is seen that α is

$$\alpha = \arctan\frac{P_y}{Px} \qquad (2.4)$$

The word arctan is used when describing the tan inverse ($\tan^{-1}$).

2.4 Direct Current (DC) Motors.

Trajectory motion is not possible without the use of electrical actuators (Saeed, 2006). The common actuator to be discussed in detail in this project is the direct current motor actuator. Direct currents (DC) are commonly found in industries. Their uses have been for a long period of time in the industrial sectors. DC motors have been proven to be reliable, sturdy and relatively powerful (Saeed, 2001).

Direct current machines are made up of four main parts. These include the stators, rotors, air-gap, and commutators and brushes

2.4.1 Stators

The stator is a set of fixed permanent magnets in a DC motor. These magnets are responsible for the creation of fixed magnetic fields. The output torque T_m is proportional to the magnetic flux $\emptyset$ and the current in the rotor windings I_{rotor} if permanent magnets are used to generate the magnetic field. Usually, the stators are made up of solid cast steel frame mounted to which a set important poles from the inner layer. Well-built iron lamination makes up the poles. The field windings (coil) are mounted on these poles. DC excitation current is carried by the field coil when connected to an external DC source. This excitation current is responsible for generating the magnetic flux in the machine. In some construction, the stator may be constructed from permanent magnet instead of the field windings. This is referred to as permanent magnet DC motor.

2.4.2 Rotor

Normally, the rotor is made of slotted, insulated iron lamination cylinder. The rotor carries the current. It carries the armature winding that is usually arranged in slots. The armature is insulated either with paper or mica layers. Through brushes and commutators, the direction of current is changed continuously, causing the rotor to rotate continuously. In converse, if the rotation of the rotor is within the magnetic field, a DC will develop and the motor will act as a generator.

2.4.3 Air Gap

Between the stator and the rotor is the air gap. Within the air gap, the energy conversion phenomenon occurs.

2.4.4 Commutators and Brushes

The commutator is made of tapered copper segments mounted on the rotting haft. These segments are insulated using either paper or mica sheets and connected to the armature.

Brushes on the other hand, are made of carbon. Carbon brushes are a mixture of carbon and a little amount of copper which is seldom used to increase the conductivity. In order to provide suitable pressure, springs are used to make proper contact with the commutator. See figure of brushed dc motor below.

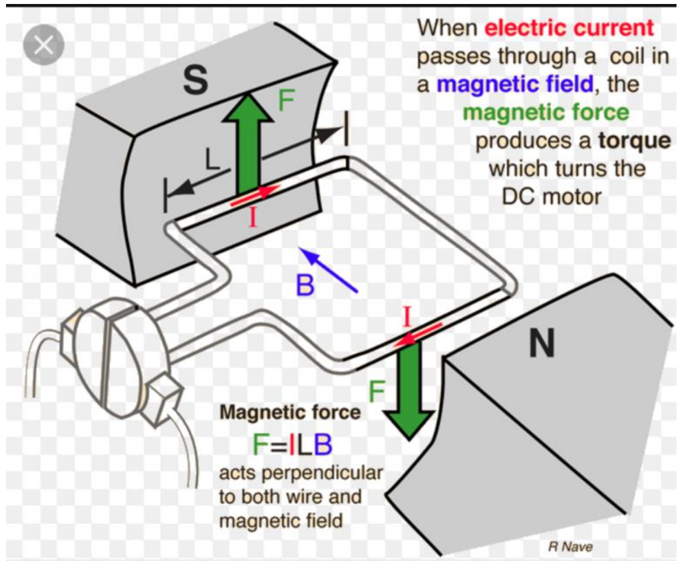


Figure 2.1: DC Motor

2.5 DC Motors Mode of Operation

In a direct current motor (DC motor) the magnetic field is generated in the air gap either by the field windings or the permanent magnets. A direct current is forced to flow through the armature windings from an external DC source passing through the brushes and the commutator. As a result of the interaction between the field circuit magnetic field and the armature current, a magnetic force is produced. This force acts on the side conductor of the same loop in an opposite direction due to an opposite direction of the current in order to create a torque that start the rotation of the DC motor's armature. The motion of the armature creates a back electromotive force (emf) which is induced in the armature windings. This back emf opposes the armature voltage and therefore reduces the armature current. When the motor rotates, each tune reaches a point in between the pole of the field. This point is called the neutral point because the conductors are no longer cutting the field's magnetic field. Hence, the force acting on the conductors at this moment is equal to 0. Because of the motor's inertia and also due to the armature coils cut off at the field at the same time, the rotor continues its clockwise rotation. Based on design, the commutator and brushes set reverse current in the armature coil after passing the neutral point to maintain the clockwise direction of rotation.

Therefore the current in the armature winding is an alternating current (AC) while the current in the armature external circuit (before the brushes) maintain its direction as direct current.

2.6 Types of DC Motors

Generally, there are four main categories of DC motors. They are categorized based on their fields winding connection. The following are the four categories:

2.6.1 Separately Excited DC Motors The separately excited DC motors have the armature and the field windings separated from each other. Their connection is equally from different DC sources. See Figure 2.2. below

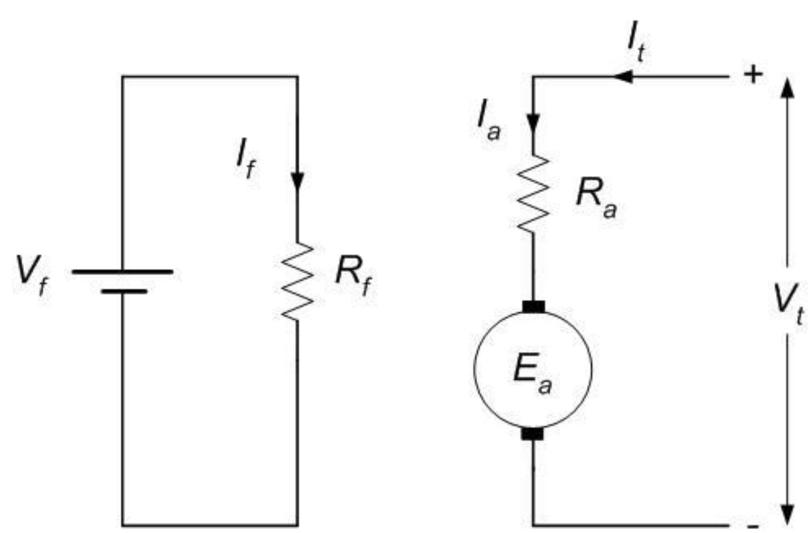


Figure 2.2: Separately Excited DC Motor Equivalent Circuit.

2.6.2 Shunt DC Motors

The shunt DC motors have the armature and the field windings connected in parallel and they are connected to same DC source. The application requires high speed and low torque. See Figure 2.3 below

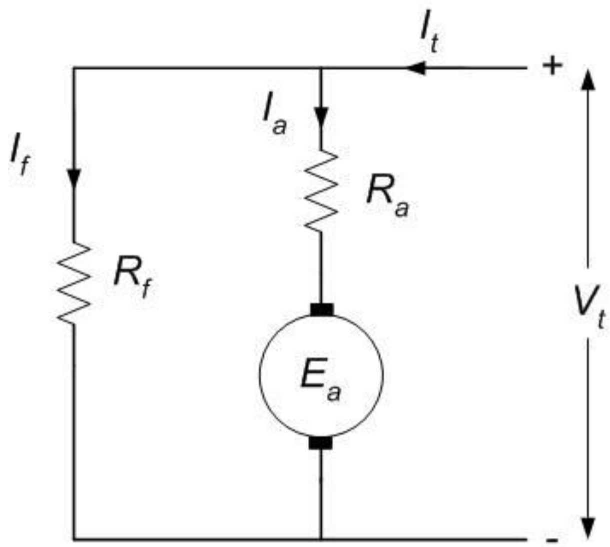


Figure 2.3: Shunt DC Motor equivalent circuit

2.6.3 Series DC Motors

The series DC motors have the armature and the field windings connected in series. Bothe the armature and the coils are connected to the same Dc source. See Figure 2.4 below

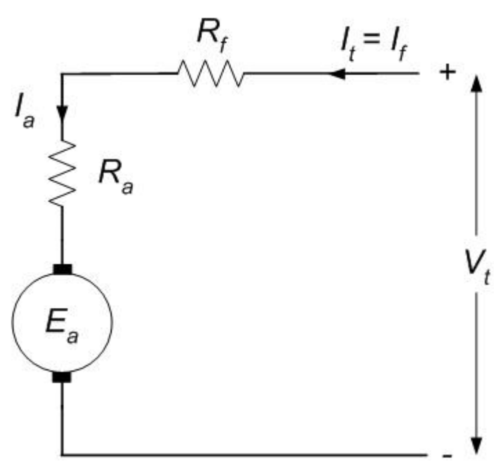


Figure 2.4: Series DC Motor equivalent circuit

2.6.4 Compound DC Motors

The compound DC motors have the field windings divided into two parts. These parts include a series part that is connected in series to armature winding and the parallel part connected to DC source. Compound DC motors are classified into two main categories. These are the commutative compound and the differential compound DC motors

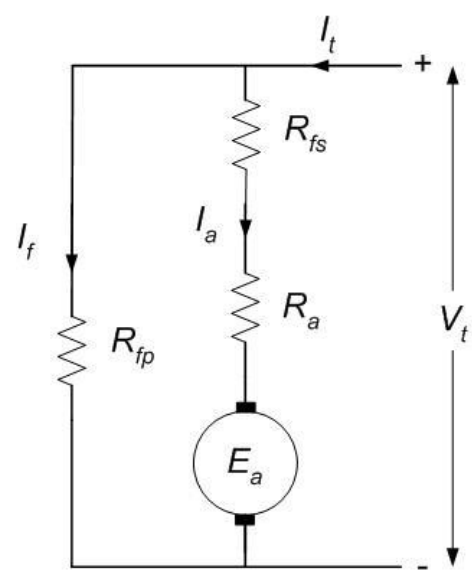


Figure 2.5: Compound DC motor equivalent circuit.

The work of actuators in a trajectory motion cannot be over emphasis as the entire drive depends on the types of actuators used in the set up.

CHAPTER 3

IMAGE PROCESSING

3.1 Introduction to Image Processing

Information about scientific research around the world has no other aim than to share with all the discoveries and innovations achieved in several areas. However, there is at least one problem in the images required for the publication of a complete and reliable work has documented; the images published in most areas are not the original images used in the research (Dias, 2008).

The digital imaging began in the early 20's of last century, when images were digitized and sent by a submarine cable between London and New York to be published in American newspapers. This initiative reduced the transport time of images through the Atlantic Ocean from more than a week to less than three hours. However, only in1964 during the early stages of space scientific exploration that emerges automatic analysis of digital images with methods to improve the visual information for human analysis and interpretation rapidly and with greater accuracy. At this time other areas such as medicine, physics, and geography e.t c started the increasing use of processing and analyzing digital images (Zhou et al., 2010).

According to (Saeed, 2001), Image processing relate to the preparation of an image for later analysis and use. He stressed that images captured by a camera or a similar technique (e.g. . by a scanner) are not necessarily in a form that can be used by image analysis routine. Some may need improvement to reduce noise, others may need to be made in a simpler form, and others may be enhanced, changed divided or be filtered etc. Therefore by virtue of definition, image processing is a collection of routines and techniques that improve, enhance, simplify or alter an image (Saeed, 2001). He further stressed that, an image is a representation of a real life scene, either in black and white or in color, and either in print or in a digital form.

3.2 Digital Image Processing.

Digital images are developed using an analogue-to-digital converter (ADC). These images are then stored in the computer memory in an image format. Some of these formats are either Tag Image File Format(TIFF), Joint Photographic Expert Group(JPEG), Bitmap, Portable network Graphics(PNP), etc. or can be displayed on the monitor (Saeed,2001). After the digitization, information, stored in a series collection of 0's and 1's which are used in the representation of the intensities of light at each cell (pixel). A digital image is a computer file which contain the 0's and 1's collection that are stored sequentially to represent high intensity at each cell (pixel).

A gray image is an image that have a ununiformed gray level at each pixel location .Usually gray values are digitized using a digitizer producing strings of 0's and 1's which are subsequently stored or displayed . On the other hand, color image is produced or developed using superimposition of three images of red, green and blue hues, both having a varying intensity and each intensity is equivalent to a gray image but in a color form. Binary images are images pixels either fully dark or fully light. Binary images are usually obtained by converting a gray scale image using histogram the image. Cut off values are called Threshold. According to Zhou (2010), preparation of a computerized picture involved application of several innovations using the computer in the handling of advance pictures. The procedure involved generates either a pictures or pixel content the original picture.

In electrical designs, a technique called flag handling is used in examination and preparation of ordinary and simple computerized program or signal. It puts away as well as separate different operation on signal .The signs take along with them sound or you call it voice signal, picture signal, transmission signals etc. In each of these signs, all forms of signals for which the information of the particular arrangement is either a picture or contains the properties of the picture.

In digital image processing, a transformation of image in different dimension is always indicated on the digital pc. In digital image processing, there are there are features that are generally acceptable. These are conservation of the original data, accuracy, repeatability and the variation. Image processing have basic techniques in its procession, some of these will be discuss as needed in this research.

3.3 Image Processing Techniques

Lots of techniques have been employed or used to increase the quality of images. Some of these techniques improve, enhance or change the image in totality so that it will be appropriate for further analysis. According to Saeed (2001), information is not removed from an image during the process of image processing .The objective of image processing is to pick out information that are trivial or remove faults. Sometimes important information that are not useful might be removed in order to have an improved image. For the purpose of this research work, some of these techniques will be discuss and they are as viz :

I. Histogram of images
II. Image segmentation
III. Image enhancement
IV. Edge detection.

3.3.1 Histogram of Images

The entire pixel number at each gray represents the histogram of an image (Saeed, 2001). It is also observed that different images might have histogram that is same to both. For instant, figure 3.1 shows three pictures with same histogram.

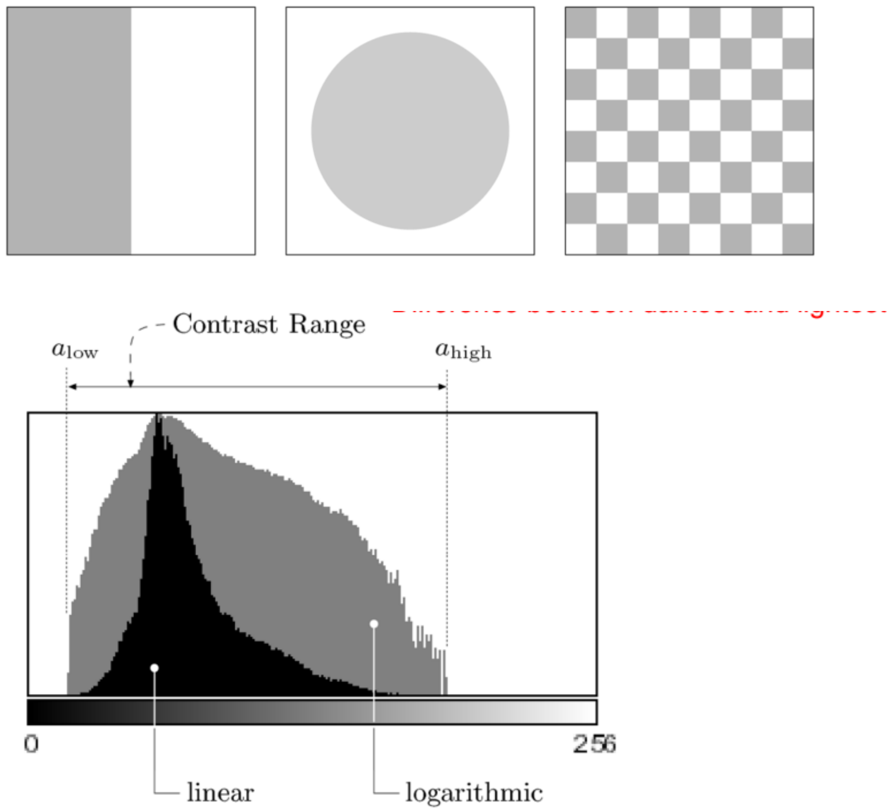


Figure 3.1: Histogram of image

In the figure above, half of their pixel values are gray and the other half is white pixel. Although they have same histogram implies equal statistic but distribution of the intensities might be different. It is worthy of note that histogram cannot be reconstructed, its pixel value remain but its intensities value might vary. During image acquisition, histogram might be of good help in image detection. Some problems with images might be identified on histogram. These problems include:

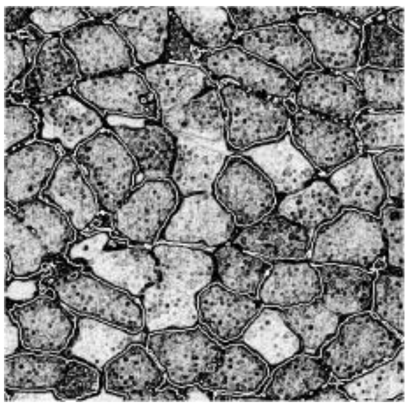
i. Contrast
ii. Brightness
iii. Dynamic range, and
iv. Over and under exposure (Wilhelm and Mark, 2008).

3.3.2 Image segmentation

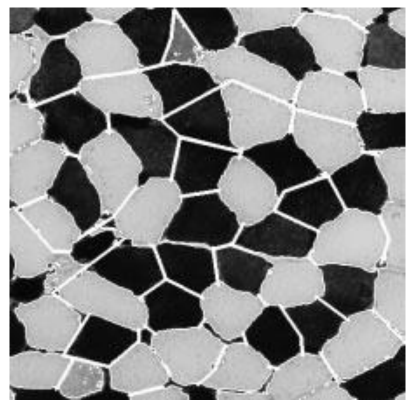
Different techniques have shown different ways of dividing an image into different segments in its constituents. Segmentation is aimed at separating the content of image information into smaller units for the purpose of other uses (Saeed, 2001). In the early study of segmentation according to Williams (1996), routines of segmentation were based on the principles of edge detection using same model like polyhedrons. Objects of three dimensional analyses such as cones, spheres, cylinders, cubes etc. were used also. Though the shapes of these figures might not fit into those of real objects but, they were sources of early work which led to the development of segmentation today (Williams,1996). Another technique which gives a clear view of segmentation is (Rattan et al.; 1999). Thresholding, deal with the process of dividing the whole image into different segments or levels where certain levels of grayness are picked as threshold, and making comparison of each pixel value with its threshold (Saeed, 2001). This is done by assigning the different portion to the pixels (or levels) . Saeed (2001), noted that thresholding can be done either at a multiple level or at a single level where the image is processed using and breaking the image into smaller "layers" and each layer having a selected threshold. For the best threshold to be chosen, many techniques have been suggested with a range for easier routines for binary images to complex routines for sophisticated images. Initial routines for binary images used the light within the object and taking the background to be dark completely (Solomon and Breckon, 2010). For this condition to be achieved, it has to be in a controlled lighting situation. In a binary image, the pixel is either off or on , this give the selection of thresholds to be easier . Some images may have multiple gray level in particular and a bionomical of the histogram will be exhibited. Advanced techniques use distributional characteristics and statistical information in developing the pixel value. Preferably, the best edge uses the protest pixel then chat them to dark. Consider the equation below:

$$F(x,y) = \sum \begin{pmatrix} 1\ if\ g(x,y) \geq T(x,y) \\ 0\ if\ g(x,y) < T(X,Y) \end{pmatrix} \tag{3.1}$$

Where F(x,y) is the estimated portion of the picture, g (x,y) is the dim scale of the pixel (x,y). T(x,y) is the incentive limit in the direction of (x,y). From the equation 3.1, the lowest harder state T(x,y) promote free and regular area for the dim scale histogram of the entire image. Solitary limits of the entire picture are chosen at a point where the histogram has pairs of articulate maximum and this propose the dark scale of the theme.



Before Thresholding



After Thresholding

Figure 3.2: Segmentation by Thresholding

3.3.3 Image Enhancement

Image enhancement deals with the modification of an image in order to suite the sweet ability of the viewer. In general, enhancement causes distortion of the original image values of the digital image; hence it is better to do image restoration before image enhancement (Solomon and Breckon, 2010).

In some cases conventional cameras and digital cameras do not have enough brightness and contrast due to the sub-system limitation of digital imaging. An image can contain a lot of noise hence needs enhancement. The main objective of image enhancement is to accentuate particular features of the image which can be used for further analysis (Solomon and Breckon, 2010). Some of these features include contrast, pseudo-coloring, edge enhancement noise filtering, magnification and sharpening. The process of image enhancement is important in image analysis, feature extraction and image display. During image enhancement process, inherent information contents of the required data does not increase, it only specifies particular image characteristics. Some of the enhancement will be considered.

3.3.3.1 Contrast Enhancement

The resolution capability power and the capability of an image detection is strongly influenced by the ratio of the contrast .There have been several techniques in the process of improving contrast of an image ;it has been shown that enhancement amongst them is the most widely used(Saeed, 2001). To Solomon and Breckon (2010), the range of sensitivity of any detector in remote sensing is designed in such a way as to record a large of area brightness from a dark basalt rock to a bright sea bed within a long rang of light condition. Some ranges of that utilize full sensitivity uses this detector. For an image to have a full contrast ratio, it is expedient to use the whole brightness range of the medium which displays the generality of the film. Contrast enhancement methods are many but most widely used are: linear - contrast stretch, nonlinear contrast stretch and Gaussian stretch. For the purpose of this research, theses three methods will be considered.

3.3.3.2 Linear Contrast Stretch

According to Swain and Davis (1978), linear contrast stretch improves contrast greatly in the original value of brightness, but also, there is a contrast loss at the extreme peaks and at the lower ends of Digital Numbers (DN) value. A digital number (DN) is a variable which is assigned to a pixel. Usually, it is in the form of a binary integer with a range of 0-255 byte. In remote sensing, the range of these energies under examination is broken into 256 bins (encyclopedia.com). Therefore comparing the overall contrast improves the loss of brightness at the extremes of the contrast is acceptable except there is interesting the cell of the scene. Due to the flexibility in the digital methods, the investigator might cause the digital numbers that fall less than 106 to appear as black(0) allowing linear stretch of the remaining digital numbers above 105 to be fall in the range of 1 to 255.This extensive stretch enhances the contrast within the bright pixels at the detriment of the other scenes. See figure 3.3 below

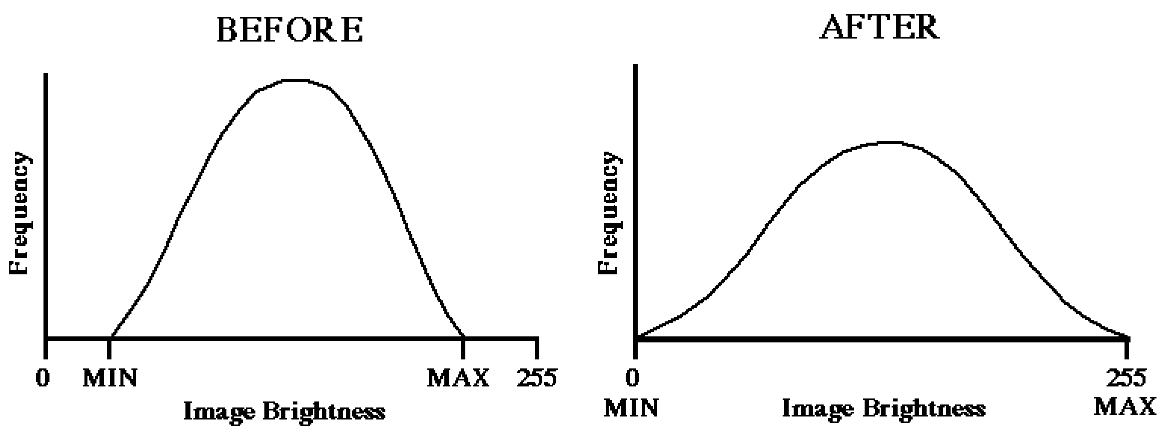


Figure 3.3: Maximum and Minimum linear Contrast Stretch

3.3.3.3 Nonlinear Contrast Stretch.

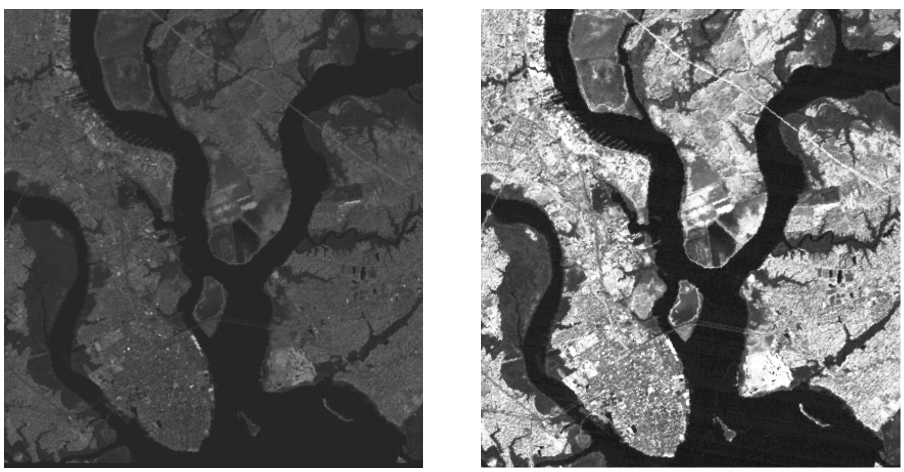
Nonlinear contrast enhancement has been shown in different ways. One of which is the histogram equalization which takes the form of a uniform stretch of distribution .This enhancement involved the use of certain algorithm. Nonlinear contrast stretch has shown one major disadvantage. It has been shown that this method that each value in the input image might generate a lot of values in the output image, such that an object in the original scene loses their real relative brightness value (Jain, 2007). During histogram equalization, the entire pixel values of the image are redistributed leading to a uniform number of pixels in the user output gray scale class. The contrast is increased around the mostly populated range of brightness value of the peak. This automatically lowers the contrast in either the dark or the light part of the image. Histogram equalization also separates pixels into unique groups. If the output values are few within the wide range. Histogram equalization becomes effective if the original image has poor contrast in the beginning or else equalization might reduce the quality of the image. To enhance this, adaptive histogram equalization is applied in order to improve the quality of the image.

3.3.3.4 Gaussian Stretch

Gaussian stretch is another nonlinear contrast enhancement method which takes its bases from the histogram of the pixel digital number. It involved fitting the observed histogram to a Gaussian or a normal distribution .The normal distribution that indicates the observable value x of the probability with a mean ∩ is given by:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-(x-\cap)^2}{2\sigma^2}\right] \tag{3.2}$$

Where σ is the standard deviation which defines the range of variables which the function p(x) lowers by the factor 0.607 or $e^{-0.5}$ of its maximum value.



Raw Image Gaussian Stretch

Figure 3.4: Gaussian Stretch (histogram equalization)

3.3.4 Edge Detection

Edge detection is one of the techniques used in image processing. By definition, an edge is a significant local change of intensity in an image (Jain et al, 2007). Edges occur typically between boundaries of two regions of an image (Jain et al, 2007). These intensities occur as a result of geometrical events also some due to non-geometrical events. Some of the geometrical events include:

- Object boundary; this shows the discontinuity in the depth and texture or surface color.
- Surface boundary; surface boundary shows discontinuity on the surface orientation, texture and/or surface color.

The non-geometrical events include:

- Peculiarity which gives direct reflection of light.
- Shadows are shown from the save object or from other objects.

Edge detection is a major taken when it comes to considering and placing sharp discontinuity in an image. Some of these discontinuities are show in the figure below

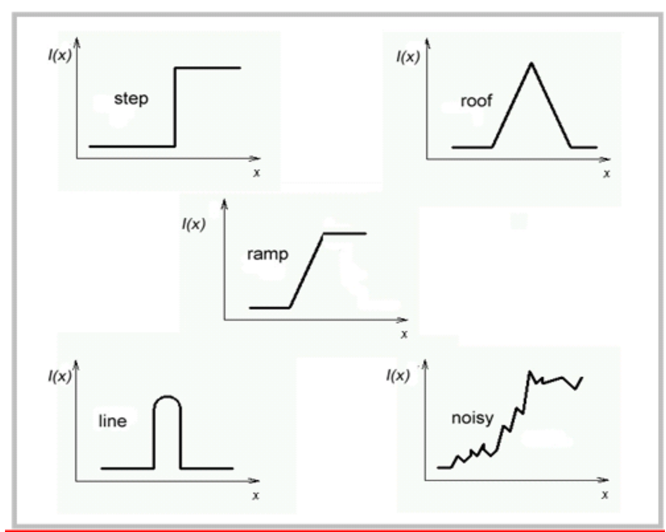


Figure 3.5: Types of edges

According to Muthukrishnan and Radha (2011), the major to be considered when it comes to considering and placing sharp discontinuity in an image can be described as edge detection. The immediate changes in pixel concentration that differentiate objects boundaries in a scene define the discontinuity. Several techniques have been used in edge detection but a few will be considered for the purpose of this research .These techniques include:

3.3.4.1. Robert Edge Detection

In 1965, Robert Edge detection was introduced by Lawrence Robert. The computation was simplified, it was also quick to be computed, 2-D spatial inclination of measurement was done on an image. The major emphases of this method were on the region of high spatial frequencies that normally correspond to edges. The operator used as input is a gray scale image which was same to that of the output which is the mostly used in the technique. The pixel values at all point in the output corresponding to the estimated full magnitude of the spatial slope of the input image at the point. Consider the illustration of the Robert mask in this table below:

Table 3.1: Robert Mask

-1	0
0	+1

Mx

0	-1
+1	0

My

3.3.4.2 Sobel Edge Detection

In Sobel edge detection, the sobel operator is applicable within the algorithm of edge detection. The use of discrete differential is applicable in the approximation and computation of the gradient of the intensities function of the image (Debosmit, 2013). In every point within the image of the sobel operator results is either a normalized vector or a corresponding gradient vector of the image. The operator based on the convolution of the image with a smaller integer and separable value of the filter in a horizontal direction as well as in a vertical direction thereby making it easier when it comes to computation. Conversely, the produced gradient approximation is almost crude, particularly for higher variation of frequencies I the image. It precedes all the edges at points where the gradient quantity of an image highlights the region with greater spatial frequency which correspond to the edge. The operator consists of a pair of 3 by 3 kernels as shown in the table below:

Table 3.2: A 3×3 Sobel Kernel

-1	-2	-1
0	0	0
+1	+2	+1

Mx

-1	0	-1
-2	0	0
-1	O	+1

my

3.3.4.2.1 Sobel Filter Design.

Several edge detection techniques work on the assumption of the occurrences of edges where discontinuity in intensity function is. It also takes the assumption of the image intensity gradient. Based on these assumptions , suppose we assumed the deviation of the intensity values in the image and distinguished areas where the deviation is at its peak(maximum), the edge could be located theoretically . We can say that the gradient is a vector whose component +shows how fast values are changing with respect to the x and y direction hence, the gradient components can be conveniently found base on these assumptions:

$$\frac{\delta F}{\delta Y}(x, y) = \Delta y = \frac{f(x,y+dy)-f(x,y)}{dy} \tag{3.3}$$

Where dx and dy indicates the distances along x and y directions. For a discrete image, dx and dy can be considered in terms of the pixel numbers between two points under consideration .This implies that dx=dy =1.which shows the pixel spacing at the coordinate (i,j).Therefore

$$\Delta x = f(i+1, j) - f(i,j) \tag{3.5}$$

$$\Delta y = f(i, j+1) - f(I,j) \tag{3.6}$$

For the gradient discontinuity presence to be deleted, the change in the gradient at (I,j) needs to be calculated . The magnitude thus is given by:

$$M = \sqrt{(\Delta x)^2 + (\Delta y)^2} \tag{3.7}$$

The direction of the angle α is determine as

$$\alpha = \arctan \frac{\Delta y}{\Delta x} \tag{3.8}$$

We can therefore see that Sobel operator is a clear example of gradient method used in filter design.

3.3.4.3 Canny Edge Detection.

The canny edge detection was developed by John Canny in the year 1983 during his master's degree thesis. This method has been proven to be the most outstanding amongst all other methods including those developed with modern algorithms (Muthukrishnan and Rodha, 2011). It detects edges by separating noise from the image then finds the edge of the image. This method does not disturb the features of the image edge but focuses on the thresholds in finding the edges of the image. Canny edge detection has the following aims

i. Good detection: In this method, it is aimed at minimizing the low probability of any failure in marking of the real edges. Also, areas of falsely marked edge points are detected. It is seen that all these decreasing functions are monotonically, hence the decrease function of output to noise ratio serves as a criteria which correspond to maximum signal to noise ratio.

ii. Minimal response: It is aimed at pairing one response to one edge at a time. The whole idea is based on the ground that, an edge needs to be marked only ones the image noise should not be allow to create a false edge.
iii. Good localization: This is to emphasis the closeness of marked points of edges to the center as true edges. The marked edges therefore very close to the edge of the real possible edge.

For the above requirements to be satisfied, Canny used/applied calculus of variation as a technique to determine the function that is capable of optimizing given functions. Canny detector uses the optimal function to describe the sum of four exponential terms but approximate it using the Gaussian first derivative.

3.3.4.4 Prewitt Edge Detection

The prewitt edge detection was developed by Judith M.S.Prewitt in the year 1970. The prewitt edge detection is accurate when it comes to estimation of the magnitude and direction of the edges (Muthukrishnan and Rodha, 2011). It is shown that it is time consuming in computation and estimation in the direction of x and y of different gradient edges. The compass edge detector obtain the direction from the kernel with the highest response .The direction is limited to 8 possible points but it has been proven that direct direction estimate are never perfect. Therefore, the gradient based edge detector is assumed to be a 3×3 neighborhood in 8 directions. These 8 convolutional masks can then be computed as shown in the table below:

Table 3.3: An 8 Convolution Prewitt Mask

-1	-1	-1
0	0	0
+1	+1	+1

Mx

-1	-1	-1
0	0	0
+1	+1	+1

My

The Prewitt method of computation appears quite simpler but seldom produces noisier results.

CHAPTER 4

SYSTEM COMPONENTS

4.1 Arduino (UNO/atmel328)

This is another form of microcontroller board that is based on ATmega328, a data sheet. It is made up of 14 digital input or output terminals and 6 can be used as pulse-width modulation (PWM) output. It also has six analogue inputs and sixteen crystalline oscillators. These are connected to a universal serial bus (USB), a reset button and an ICSP head. The entire composition of the arduino has all the components needed to support the work of microcontroller. Its connection can be either with the use of an AC to DC adaptor, a battery or a using a computer with USB terminal in order to start it.

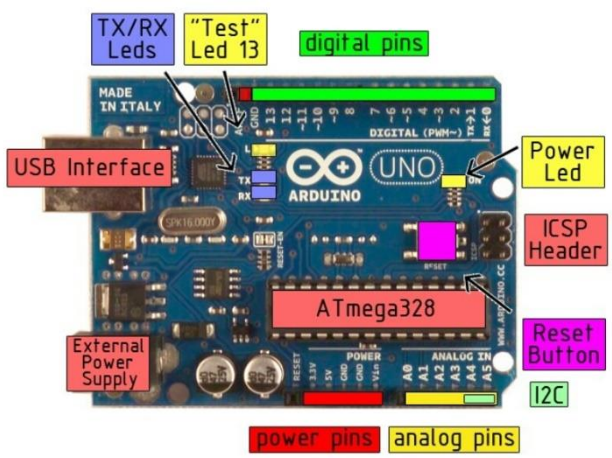


Figure 4.1: Arduino (UNO).

4.2 DC Motor

The Direct Current motor is classified amongst the common actuators in use today. Basically, direct current motors produce continuous motion and its rotational speed can be controlled easily. For this reason, they are ideal in speed control application. The direct current motor is made up of two basic parts: the stators, the part that consist of the stationary components and the Rotor the part that consist of the rotating components.

There are basically three types of direct current motors. For the purpose of this research, work, the Brushed Direct Current Motor will be used.

4.2.1 Brushed Direct Current Motor

The Brushed DC motor produces tremendous magnetic field in its sealed rotor. It is basically used in many applications such as push buttons, toys, and adjustable seats of cars. Brushed Direct Current (BDC) Motors are made up of stators, rotors commutator and brushes. In the stators, a stationary magnetic field is generated by the wounded rotor. Either the permanent magnet or the windings generate the field. The rotor on the other hand consists of a single or multiple windings. It is also called the armature. When the windings are powered, a magnetic field is produced. The brushed DC motor has advantages which include:

i. They are inexpensive
ii. They are highly reliable
iii. Readily available in different shapes and sizes
iv. They are easily control

They also have disadvantage in the area of low life span when highly intensity use is applicable as well as high cost of maintenance.

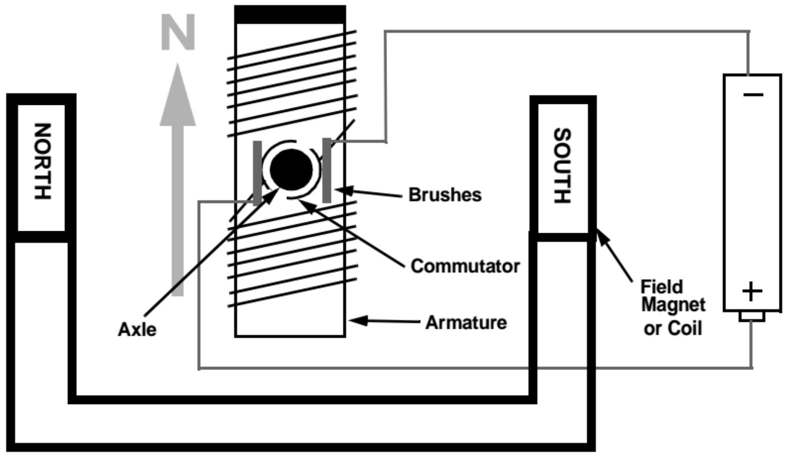


Figure 4.2: Simple Two-Pole Brushed DC Motor.

4.3 Servo Motor

4.3.1 Pulse Width Modulation (PWM)

The basis of power control in electronics is based on the pulse width modulation. In theory the zero rise and fall in time for an ideal pulse width modulation waveform is a perfect representation of modern semi-conductors devices when it comes to power drive. Many electronic circuits that are in vast use today uses the pulse width resonance control except that some of the resonance converters have proven otherwise. The enormous rise and fall in the edges shows the on/off turns of the semiconductor power device which minimizes the switching form of transistors rate with respect to losses in switching. The heartbeat changes in width within the area of 0in a period T. The basic rule controls the energy when the circle obligation is shifted. The speed of a direct current motor rotation is proportional to the supply voltage average value. The motor experience fast rotation when the average voltage supply is high and conversely. The rotation of a DC motor can be varied by varying the duty ratio when it is turn on/off.

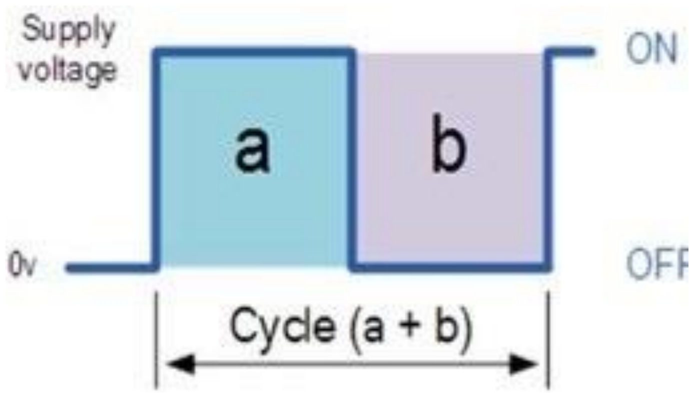


Figure 4.3: PWM (Duty Cycle)

For a good understanding of pulse-width modulation, the illustrations in Fig 4.4, 4.5 and 4.6 shows. These figures give the required waveform at different voltages required. A 90% signal indicates rapid signal (Fig 4.3), 50% indicates a signal with half voltage and 10% indicates a signal with low voltage.

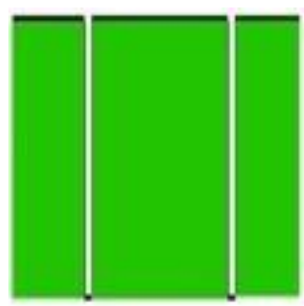


Figure 4.4: Signal with High voltage (90%)

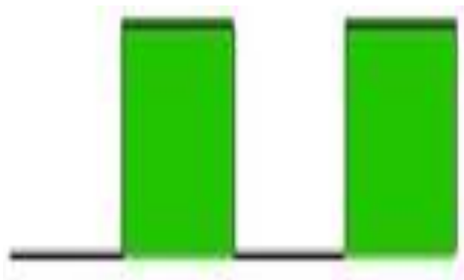


Figure 4.5: Signal with Half voltage (50%)

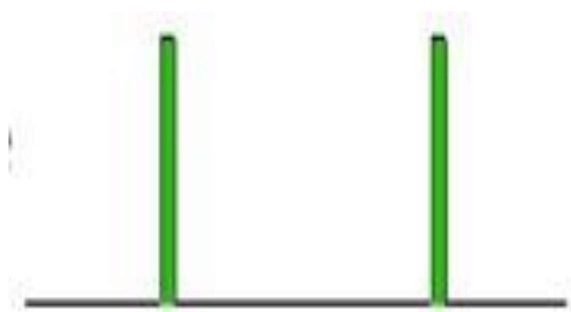


Figure 4.6: Signal with Low voltage (10%)

3.3.2 Servo motor Principle:

Its position is defined by the width of duty cycle of the PWM pulses arriving from Arduino as shown in Figure 4.5

This will redirect the mechanical structure towards the target.

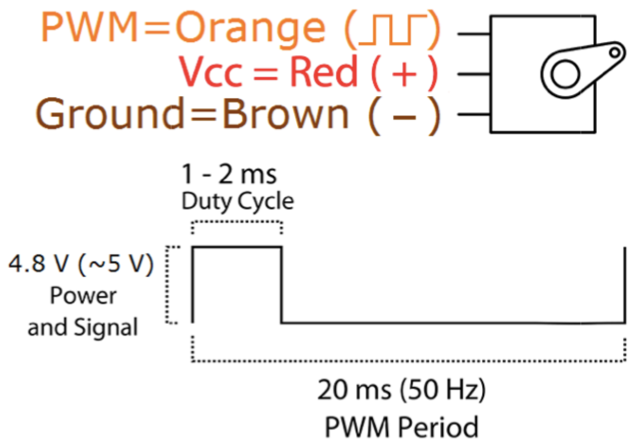


Figure 4.7: PWM Period in Servo Motor

Figure 4.8: Servo Motor (SG90)

Table 4.1: Specifications

Weight	9 g
Stall torque	1.8 kgf cm
Speed	60 degree/0.1s
Voltage	4.8 V (~5V)

4.4 Camera

A true HD-quality video camera capable of recording at frame rate up to (30 fps) is connected to pc via USB. See figure 4.9

Figure 4.9: Camera brand (A4TECH)

We choose this camera because it has ability to recording at 30 fps

4.4.1 Camera Processing Time

Table 4.2: Execution time of Red and blue object detection with different image resolution

Image type	Execution time
250X250	0.11 second
350X350	0.18 second
500X500	2.0 seconds

CHAPTER 5

THE EXPERIMENTS AND RESULTS

5.1 Aim

The aim of this subsystem is to guide the moving car to determined location along the pre defined trajectory.

The pre defined trajectory might be calculated by the Main control PC according to some algorithm. See figure 5.1

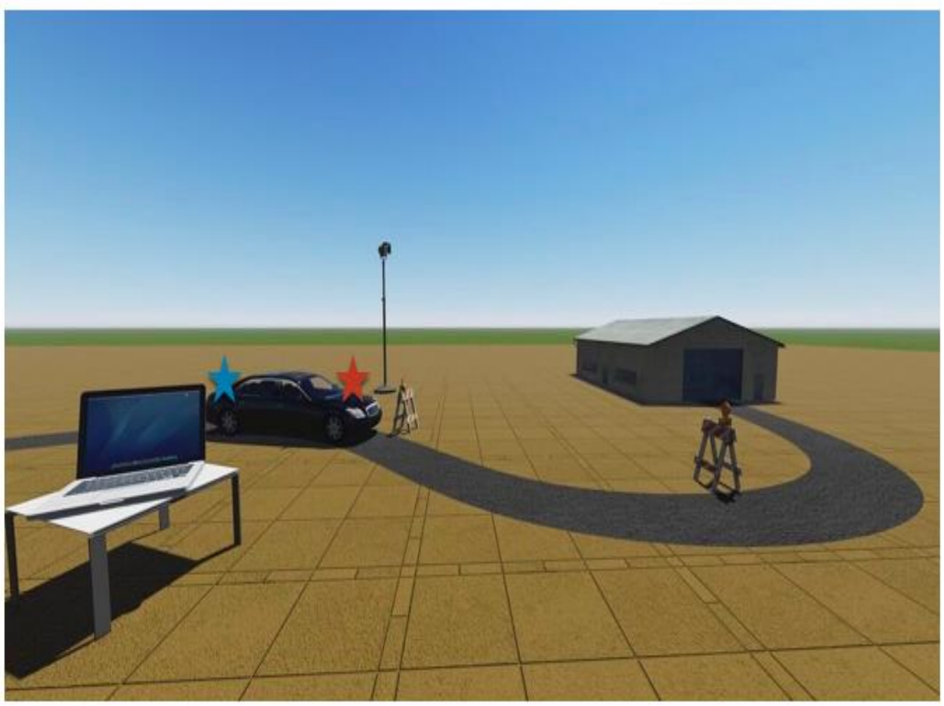


Figure 5.1: Moving the Car toward Specific Area (Lumion3D photo)

5.2 Images of Platform

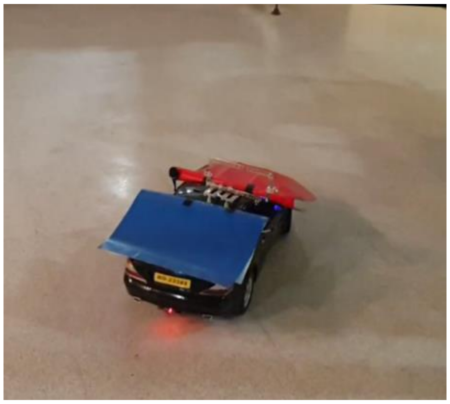


Figure 5.2: Car out of Target

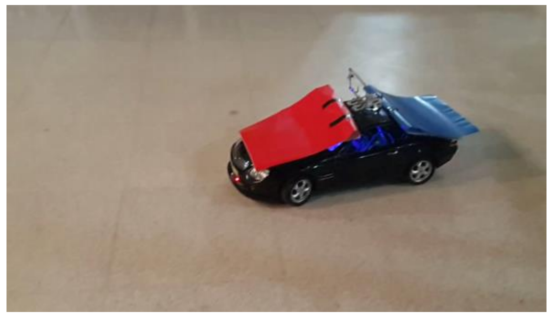


Figure 5.3: Car moving towards Target

Figure 5.4: Car on Target

Table 5.1: Calibration of the thresholding value for red pixel detection; Optimum values are highlighted.

Threshold	Result Accuracy
1%	30
5%	40
12%	95
20%	59

Table 5.2: Calibration of the thresholding value for red pixel detection; Highlighted row is the optimal result.

Threshold	Result Accuracy
1%	25
5%	55
15%	95
20%	60

5.3 System Overview

As shown in Figure 5.2, the controlling pc gets images from the camera (cam), processes this images to get current position of the car and calculate the vector between front and rear of the car and vector between the rear of the car and the target . The bluetooth as a wireless device transmit the signal from the pc to the microcontroller, then the microcontroller set the values to be sent to servo motor (controlling right-left) and dc motor (controlling forward-backward) via the Arduino.

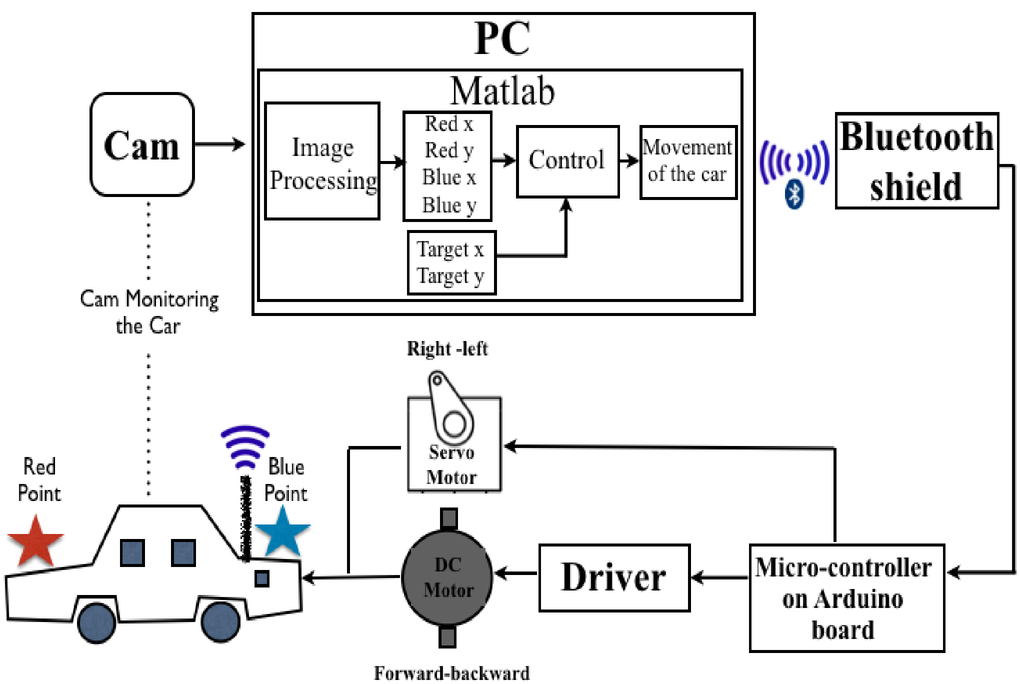


Figure 5.5: Block Diagram of moving the car toward specific area

5.4 Flow Chart

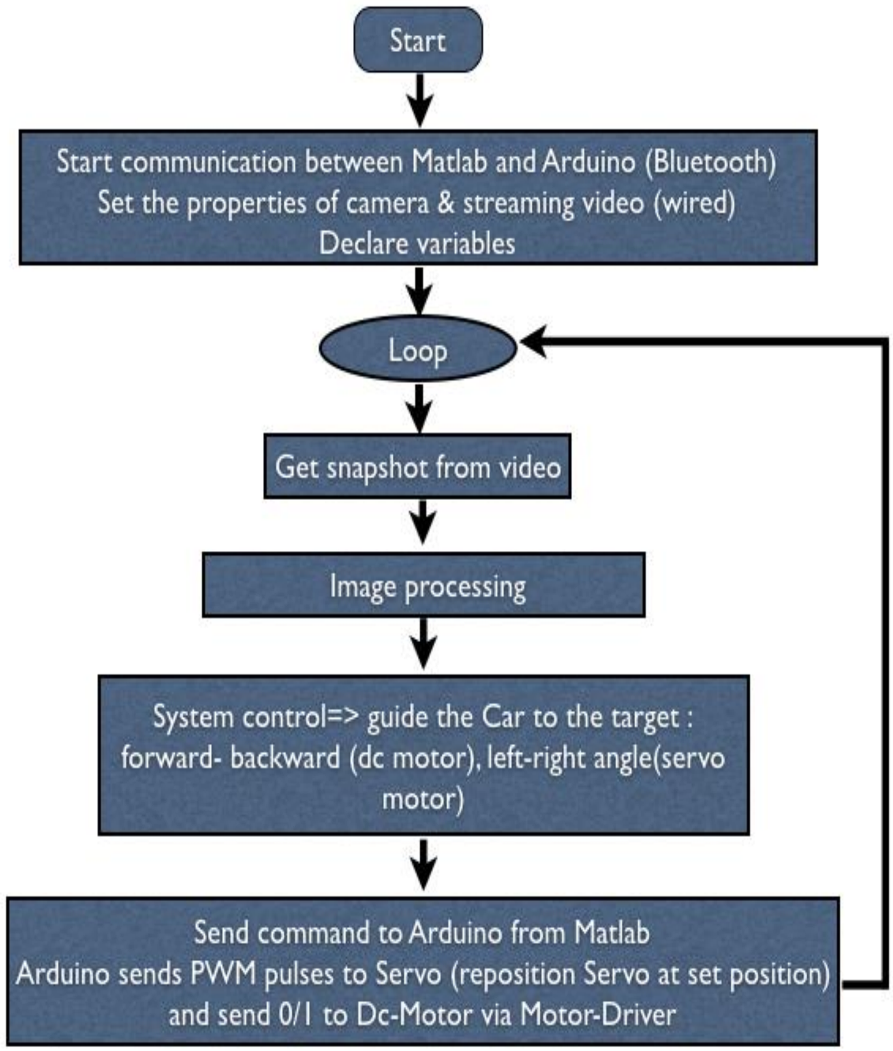


Figure 5.6: Flow chart of Moving the Car toward Specific Area

As shown in Figure 5.3, the communication start between Matlab and Arduino (Bluetooth), set properties of camera and streaming video (wired).

Start the loop and get snapshot from video then process the snapshot to get current position of the moving car.

Compute the angle between two vector (vector#1 between the front and the rear of the car, vector#2 between the rear of the car and the target) using cross product.

Then calculate the driving parameters of the servo motor and dc motor according to the control system.

5.5 Methodology

We developed a program under Matlab that do two tasks: image processing and controlling.

5.5.1 *Image processing (detecting the car)*

This algorithm designed to detect the red and blue pixels of image according to value of intensity of pixels. Each image has three components of color Red, Green and Blue (RGB) with different intensity from 0 – 255. The algorithm will stream the images then convert each RGB image to Gray scale image using (1).

$$\text{Gray scale index} = 0.2988 \times R + 0.5870 \times G + 0.1141 \times B \qquad (5.1)$$

The idea is developed with the aim of producing a binary image which have the Red or Blue to be “1” and any other color to be represented by “0” owing the fact that the original image have three colors in its components. These colors are red, green and blue (RGB). In order to enhance these components, we introduce a technique of open morphology which prevents regional overlap amongst the different color intensities. This enables filtering of the image then allows the removal of unnecessary pixels which are created as result of noise without changing the pixels from the original image. In order to get the separate regions, a complete structural value which is symmetrically applied in the entire direction is done with an opening operation with the spatial direction implied a 5×5 matrix of unitary elements in each cell. The choice of these structural elements depends on the image resolutions.

The detection components center can be determine by labeling all the connected components of the binary image, then find the center of its label. This technique identifies disconnected components of the binary image. The technique is defined using two methods in the connectivity; a 4-connectivity and an 8-connectivity. This connectivity is responsible for the choice of the pixel to be included in the image. This algorithm assigned all pixels found in the same region a level of 8-connectivity. First connected components pixels are equally label as one, and second connected are also label as two and the series continue indefinitely.

5.5.2. Detection method in details:

1-Get the snapshot from camera that streaming video to Matlab

First input the video with properties from camera to matlab :

```
v = videoinput('winvideo',1, 'YUY2_640x480' );
```

then start the video :

```
start(v);
```

Read snapshot from camera that streaming video to matlab :

```
c = getsnapshot(v);
```

Then do some command to get red components:

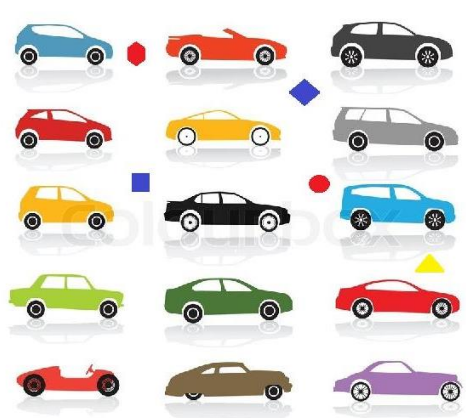


Figure 5.7: Original image

2-subtract a grayscale image from red component image using appropraite command

e = imsubtract(c(:,:,3), rgb2gray(c));

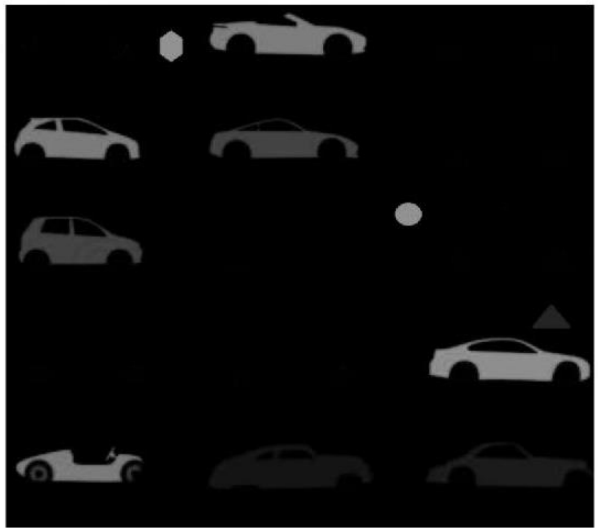


Figure 5.8: Subtraction Result

3- Convert the result into a binary image, according to a threshold to detect the specific intensity of red color

e = im2bw(e,0.19);

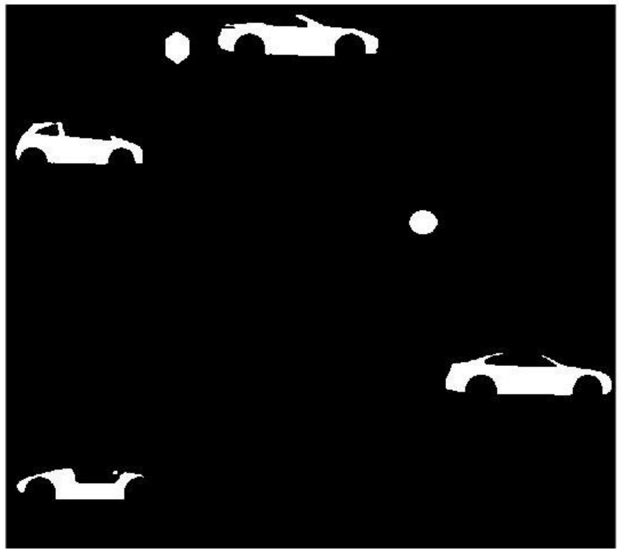


Figure 5.9: Convert the gray scale image to Binary Image

4- Remove small objects from binary image to focus on a big objects

```
e = bwareaopen(e,500);
```

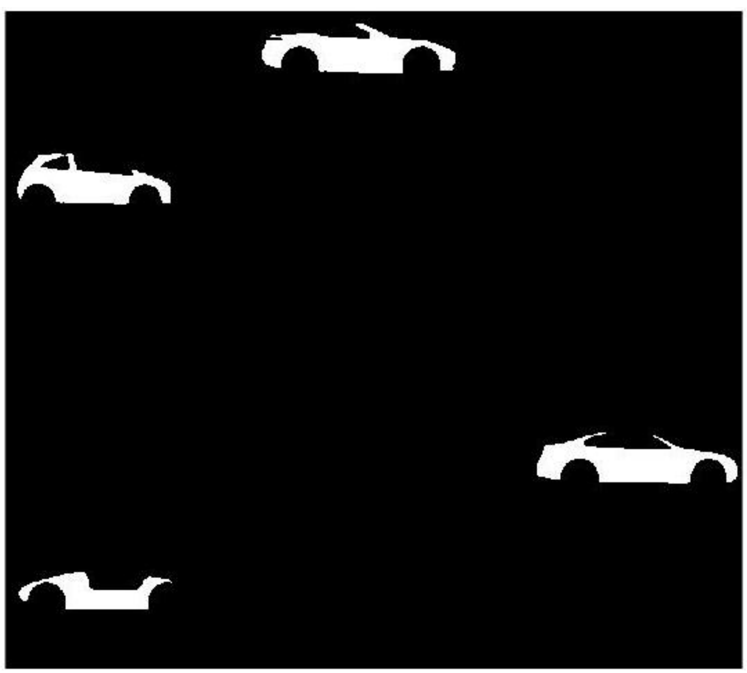


Figure 5.10: Remove small objects

cc = bwlabel(e, 8);

5-Then connected components in the image are label and the properties are set for each labeled region and display calculated properties on the original image:

5- Display calculated properties on the original image:

```
   stats = regionprops(cc, 'BoundingBox', 'Centroid');


   figure(4);imshow(c)
 hold on
   %This is a loop to bound the red objects in a rectangular box.
   for object = 1:length(stats)
       flag=1;
       bb = stats(object).BoundingBox;
       bc = stats(object).Centroid;
       rectangle('Position',bb,'EdgeColor','b','LineWidth',5)
        plot(bc(1),bc(2), '*')
       x= num2str(round(bc(1)));
       y= num2str(round(bc(2)));
         m=text(bc(1)+50,bc(2), strcat('X: ', x, '    Y: ', y));
      set(m, 'FontName', 'Arial', 'FontWeight', 'bold', 'FontSize', 14, 'Color', 'red');
   x=str2num(x);
   y=str2num(y);
   end
   hold off
```

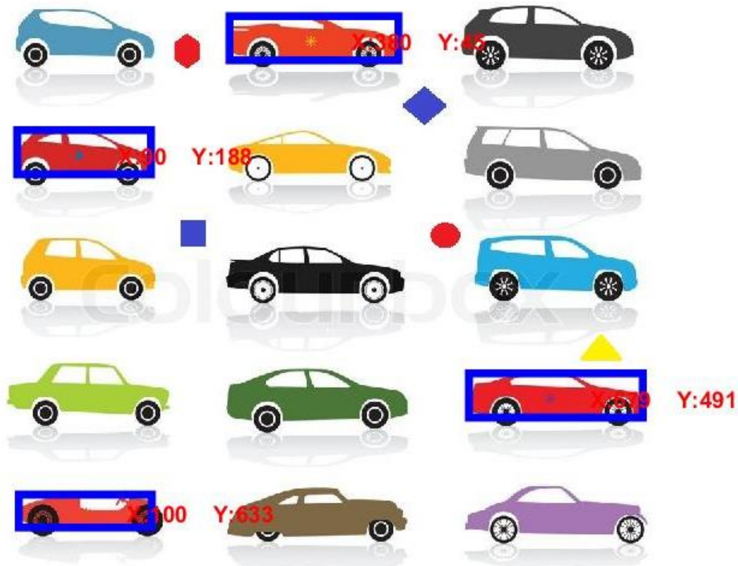


Figure 5.11: Detect the target

(At the end of image processing we get the centroid [X,Y] of the target.

5.6. *Controller*

A- Compute the angle between two vector (vector#1 between the front and the rear of the car, vactor#2 between the rear of the car and the target) using cross product then, calculate the driving parameters of the servo motor and DC motor according to the control system.

The microcontroller comprises of the DC motor and the Servo motor. The two are responsible for the motion of the car. The DC motor is responsible for the backward and forward movement of the car. The servo motor also does the responsibility of right and left movement of the car.

To determine the angle between the vectors in the plane, we let the vector along the red and blue be #1 and the vector between the blue and the target be #2. Let the coordinates of the vectors in the plane be (x,y) .

For the x-components of #1, we have;

$$X_{red} - X_{blue} \quad \rightarrow \vec{X} = \vec{x}_r - \vec{x}_b \tag{5.1}$$

For the y-components of #1, we have

$$\text{Y}_{\text{red}} - \text{Y}_{\text{blue}} \rightarrow \vec{Y} = \vec{y}_{\text{red}} - \vec{y}_{\text{blue}} \tag{5.2}$$

Therefore, the length of the vector $\vec{R}_1$ is given by;

$$\vec{R}_1 = \sqrt{(\vec{X})^2 + (\overrightarrow{Y})^2} \tag{5.3}$$

The normalized vector along the x is

$$\vec{N}_{1x} = \frac{\overrightarrow{X}}{\vec{R}1} \tag{5.4}$$

Also the normalized vector along the y is

$$\vec{N}_{1y} = \frac{\vec{Y}}{\vec{R}1} \tag{5.5}$$

The vector along the blue and the target can be determined as viz:

The position vector along the blue and the target #2 along the x-coordinates $\vec{X}_p$ is given by

$$\vec{X}_p = \vec{X}_b - \vec{X}_t \tag{5.6}$$

Also position vector along the y coordinates of #2

$$\vec{Y}_p = \vec{Y}_b - \vec{Y}_t \tag{5.7}$$

$$\vec{R}_2 = \sqrt{(\vec{Y}\text{p})^2 + (\vec{Y}\text{t})^2} \tag{5.8}$$

The normal position vector x coordinates is therefore

$$\vec{N}_{2x} = \frac{\vec{X}\text{p}}{\vec{R}2} \tag{5.9}$$

Also normal vector along the y coordinates is therefore

$$\vec{N}_{2y} = \frac{\vec{Y}p}{\vec{R}2} \tag{5.10}$$

The resultant of the entire car vector is therefore.

$$Z = (\vec{N}_{1x} - \vec{N}_{1y}) - (\vec{N}_{2x} - \vec{N}_{2y}) \tag{5.11}$$

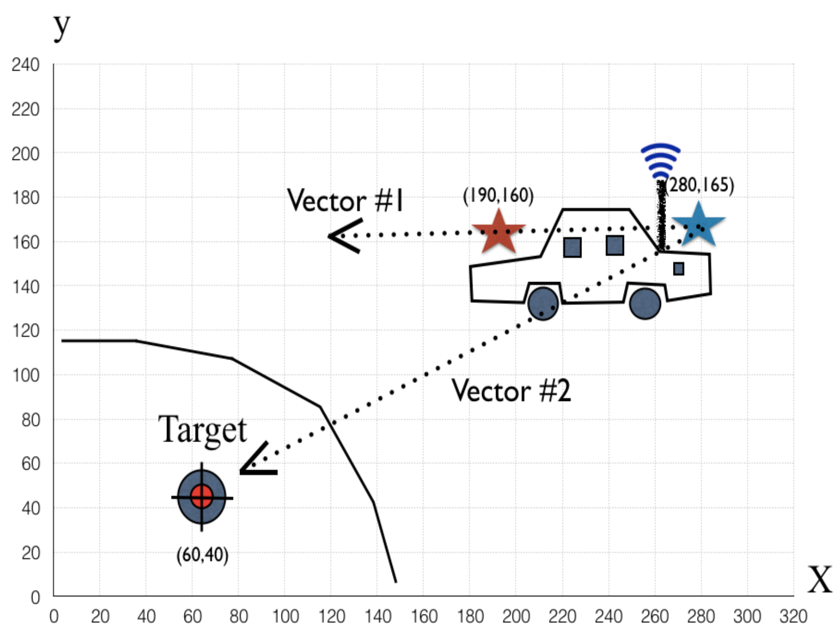


Figure 5.12: Define the angle between v1&v2

B- If the car gets out of the camera range it will go back in the opposite direction (if it moves out forward should move back backward , if it moves out backward should move back forward). See Figure 5.9

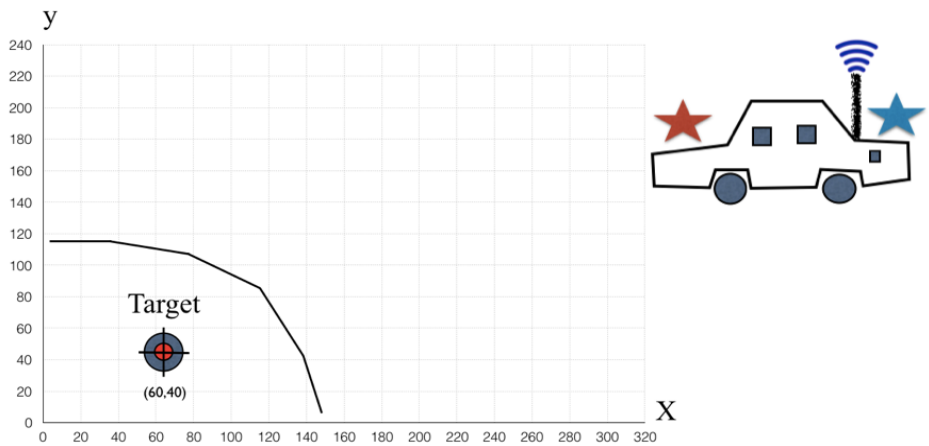


Figure 5.13: Car is out of camera range

C- Send appropriate command to Arduino (Bluetooth communication).

CHAPTER 6

CONCLUSION

6.1 Conclusion

Successful trajectory can be achieved if proper algorithms are followed. Environments for a proper trajectory motion pose many challenges as urban areas tend to be crowded. A good consideration of some factors that could guarantee a successful trajectory motion was found to include both the velocity and the direction in the plane which are basic elements of a vector. Trajectory motion was successfully carried out with the aid of some peculiar actuators i.e. the DC motor and the servo motor which are primarily responsible for the forward and backward motion in the trajectory as well as the left and right motion of the entire system respectively. Other components used include the Arduino Uno, Bluetooth shied, digital camera and the robotic car which was used for the color detection.

Digital image processing (DIP) algorithm does the motion control using the computer PC. The entire system is a series of changes from analogue-to-digital using an analogue-to-digital converter (ADC). The algorithm was fashion to detect red and blue pixels images according to the value of intensity of the pixel inputted. Color images were converted to gray-scale images then, gray-scale images to binary images according to threshold which results into the required intensity of the red and blue needed.

REFERENCES

Ali, S., and Shah, M (2008). Floor field for tracking in high density crowd scenes. *European Conference on Computer Vision*, 1-14.

Canny, J. F. (1983). *Finding edges and lines in images*, Master's thesis, MIT. AI Lab. TR-720.

Canny, J. F. (1986). A computational approach to edge detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence, 8, 679-714.*

Chapman, S. J. (2005). Electrical machines fundamentals. New York, NY: MC Graw Hill.

Ray, D. (2015). Edge detection in digital image processing. *International Journal of Biomedical Imaging,* doi:10.1155/2015/687810

Dorf, C. R.,and Bishop, R.H. (2014). *Modern Control System.*Upper Saddle River, NJ: Pearson (Prentice Hall)

Ess, A., liebe, B., Schindler, K., and Gool van. L. (2009). Moving obstacles in highly dynamic scenes. *In Proceedings of IEEE Conference on Robotics and Animation.* Japan, JP

Gonzalez, C. R, Woods, E. R, and Eddins, L. S. (2004). *Digital image processing using MATLAB.*, Upper Saddle River, NJ: Prentice Hall.

Hinz, S., Bamler, R., and Stilla, U. (2006). Theme issue: “Airborne and space-borne traffic monitoring”. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61, (3–4), 135–280

Hinz, S., Weihing, D., Suchandt, S., and Bamler, R. (2008). Detection and velocity estimation of moving vehicles in high-resolution spaceborne synthetic aperture radar data. *In IEEE Conference of Computer Vision and Pattern Recognition.* Anchorage, AK, USA.

Kerekes, J., Muldowney, M., Strackerhan, K., Smith, L., and Leahy, B. (2006).“Vehicle tracking with multi-temporal hyper spectral imagery,” in Proc. SPIE, vol. 6233, 62330C.

Ladya, da. (2014). *Arduino Tips, Tricks, and Techniques*.New York. NY: Adafruit Industries.

Lee, J. G., Han, J., and Whang, K. Y.(2007). "Trajectory Clustering: a partition-and-group framework," *In proceeding of the ACM SIGMOD international conference on of data,* Pp 593-604 .

Liu, A., Peng, C., and Chang, S. (1997, January). "Wavelet Analysis of Satellite Images". *IEEE Journal of Oceanic Engineering,* 22, 9-17

Morris, B. and Trivedi, M. (2009). Learning Trajectory Patterns by Clustering, Experimental Studies and Comparative Evaluation in Computer Vision and Pattern Recognition. *IEEE Computer Society Conference*. Miami, FL, USA.

Morris, B. and Trivedi, M. (2008). An Adaptive Scene Description for Activity Analysis in Surveillance Video. *In Proceeding of the 19th ICPR International conference on Pattern Recognition.* Tampa, FL, USA.

Muthukrishna,R. and Rodha,M (2011)." Edge Detection Techniques for Image Segmentation", *International Journal of Computer Science and Information Technology,* Vol 3, Issue 6 pp.259,

Saeed, N. B. (2001). *Introduction to Robotics. Analysis, Systems, Applications.* California, CA: Prentice hall.

Solomon, C.J., and Breckon, T.P.(2010). Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab. Wiley-Blackwell. Doi:10.1002/9780470689776.ISBN 0470844736.

Stark, J.A. (2000). Adaptive Image Contrast Enhancement Using Generalization of Histogram Equalization. *IEEE Transaction on Image Processing,* 12(5), 889-896.

Wang, C. and Ye, Z. (2005). Brightness Preserving Histogram Equalization with Maximum Entropy: A Variation Perspective, *IEEE Transaction on Consumer Electronics, 51(4),* 1326-1334.

Wildi, T. (2006). *Electrical Machines Drive and Power System*. Ohio, OH: prentice hall.

Yilmaz, A., Javed, O., and Shah, M. (2016, December). "Object tracking: A survey," ACM Computer. Survey., vol. 38.

Zhou, H.,Wu,J., and Zhang, J. (2010). *Digital Image Processing part 2.* Denmark: Ventus Publishing ApS.

Zhou,H.,Wu,J., and Zhang, J. (2010). *Digital Image Processing part 1.* Denmark: Ventus Publishing ApS.

APPENDICES

APPENDIX 1

MATLAB Code

```
a = imaqhwinfo;

% input the video

v = videoinput('winvideo',2, 'YUY2_352x288' );

% Set the properties of the video

set(v, 'FramesPerTrigger', Inf);

set(v, 'ReturnedColorspace', 'rgb')

flag_dir_l_r=0; %% direction of car in x axis if 1 left ___ if 2 right

flag_dir_f_b=0; %% direction of car in y axis if 1 forward ___ if 2 backward

flag1=0;

flag2_blue=0;

flag2_red=0;

flag_return_f_b=1;

x_parking=270;

y_parking=70;

x_red=0;

x_blue=0;

y_red=0;

y_blue=0;
```

```
teta1=0.08;

teta2=0.1;

range_of_target=40;

distance_x=0;

distance_y=0;

message=0;

c = getsnapshot(v);

m=111

b=Bluetooth('Robotic_car',1);

fopen(b); % open bluetooth stream between matlab and arduino

m=222

while(1)

   flag2_blue=1;

   flag2_red=1;

distance_x=1000;

distance_y=1000;

% image processing to find blue component (rear of the car) and red component(front of the car)

   % Get the snapshot of the current frame

   c = getsnapshot(v);
```

```
% we have to subtract the blue component

e = imsubtract(c(:,:,3), rgb2gray(c));

% Convert the resulting grayscale image into a binary image.

e = im2bw(e,0.17);

% Remove all those pixels less than 250px

e = bwareaopen(e,300);

% Label all the connected components in the image.

cc = bwlabel(e, 8);

% We get a set of properties for each labeled region.

stats = regionprops(cc, 'BoundingBox', 'Centroid');

% we have to subtract the red component

e1 = imsubtract(c(:,:,1), rgb2gray(c));

% Convert the resulting grayscale image into a binary image.

e1 = im2bw(e1,0.17);

% Remove all those pixels less than 250px

e1 = bwareaopen(e1,300);

% Label all the connected components in the image.

cc = bwlabel(e1, 8);

% We get a set of properties for each labeled region.

stats2 = regionprops(cc, 'BoundingBox', 'Centroid');
```

```
% Display the image
   figure(1);imshow(c)
   hold on
 for object = 1:length(stats)
   flag2_blue=0;
   bb = stats(object).BoundingBox;
   bc = stats(object).Centroid;
   rectangle('Position',bb,'EdgeColor','b','LineWidth',5)
   plot(bc(1),bc(2), '*')
   x= num2str(round(bc(1)));
   y= num2str(round(bc(2)));
   m=text(bc(1)+50,bc(2), strcat('X: ', x, '    Y: ', y));
   set(m, 'FontName', 'Arial', 'FontWeight', 'bold', 'FontSize', 14, 'Color', 'yellow');
   x_blue=str2num(x);
   y_blue=str2num(y);
 end
 for object = 1:length(stats2)
   flag2_red=0;
   bb = stats2(object).BoundingBox;
   bc = stats2(object).Centroid;
```

```
        rectangle('Position',bb,'EdgeColor','r','LineWidth',5)

        plot(bc(1),bc(2), '*')

        x= num2str(round(bc(1)));

        y= num2str(round(bc(2)));

        m=text(bc(1)+50,bc(2), strcat('X: ', x, '    Y: ', y));

        set(m, 'FontName', 'Arial', 'FontWeight', 'bold', 'FontSize', 14, 'Color', 'yellow');

        x_red=str2num(x);

        y_red=str2num(y);

    end


% find teta between two vectors

vector_x_car=x_red-x_blue;

vector_y_car=y_red-y_blue;

vector_car_length=sqrt(vector_x_car.^2+vector_y_car.^2);

normal_vector_x_car=vector_x_car/vector_car_length;

normal_vector_y_car=vector_y_car/vector_car_length;

vector_x_pos=x_parking-x_blue;

vector_y_pos=y_parking-y_blue;

vector_pos_length=sqrt(vector_x_pos.^2+vector_y_pos.^2);

normal_vector_x_pos=vector_x_pos/vector_pos_length;
```

```
normal_vector_y_pos=vector_y_pos/vector_pos_length;

z=(normal_vector_x_car*normal_vector_y_pos)-
(normal_vector_y_car*normal_vector_x_pos)

if(flag2_red==0&&flag2_blue==0)

if(flag1==0)

if(z<-teta1)

   if(z>-teta2)

      degree=1;

   else degree=2;

   end

   switch(flag_return_f_b)

      case (1)

      message=10+degree;

      fwrite(b,message);% move the car forward and left

      flag_dir_l_r=1; % flag left

      flag_dir_f_b=1; % flag forward

      case(2)

      message=60+degree;

      fwrite(b,message); % move the car backward and right

      flag_dir_l_r=2; % flag right
```

```
    flag_dir_f_b=2; % flag backward
    end
else if (z>teta1)
        if(z<teta2)
            degree=1;
        else degree=2;
        end
    switch(flag_return_f_b)
        case(1)
    message=20+degree;
    fwrite(b,message);% move the car forward and right
    flag_dir_l_r=2; % flag right
    flag_dir_f_b=1; % flag forward
        case(2)
    message=50+degree;
    fwrite(b,message);% move the car backward and left
    flag_dir_l_r=1;% flag left
    flag_dir_f_b=2;% flag backward
    end
    else
```

```
        fwrite(b,3); % move the car forward straight
        flag_dir_f_b=1;
        sprintf('%s','forward ')
        end
end
end
end
distance_x=abs(x_red-x_parking);
distance_y=abs(y_red-y_parking);
flag1=0;
if(distance_x<range_of_target&&distance_y<range_of_target)
    fwrite(b,90); % stop the car
    flag1=1;
    flag_dir_f_b=0;
    flag_dir_l_r=0;
    flag_return_f_b=1;
end
if(flag2_red==1||flag2_blue==1)
     switch(flag_dir_f_b)
          case(1)
```

```
        fwrite(b,40);

        flag_return_f_b=2;

          case(2)

        fwrite(b,30);

        flag_return_f_b=1;

    end

end

       hold off

end

% Stop the video aquisition.

stop(v);


% Flush all the image data stored in the memory buffer.

flushdata(v);


% Clear all variables

clear all
```

APPENDIX 2

Arduino Code

```
#include <Servo.h>

#include <SoftwareSerial.h>

Servo myservo;  // create servo object to control a servo

#define RxD 6 // This is the pin that the Bluetooth (BT_TX) will transmit to the Arduino (RxD)

#define TxD 7 // This is the pin that the Bluetooth (BT_RX) will receive from the

int pos =50;    // variable to store the servo position

SoftwareSerial bluetooth(RxD,TxD);

  int x;

  int x1;

  int x2;

  int left_max=20; // left max

  int left_min=30; // left min

  int right_max=95; //right max

  int right_min=65; // right min

  int straight=55; // center

void setup() {

 Serial.begin(9600);
```

```
  bluetooth.begin(9600);
   myservo.attach(11);  // attaches the servo on pin 11 to the servo object
        pinMode(8,'Output'); // pin number 8 as digital output
        pinMode(9,'Output');  // pin number 9 as digital output
          myservo.write(55); //send pulse to servo to make it at center
}
void loop()
{
  analogWrite(5,255);
  Serial.flush(); // delete every thing from bluetooth
  x=-1;
  if (bluetooth.available() < 1)
  {
  return; // if serial empty, return to loop().
  }
  x = bluetooth.read(); //read the bluetooth signal and save it in x
  x1=x/10;
  x1=round(x1);
  x2=x-x1*10;
switch(x1)
```

```
{
case 1 :
{
 if(x2==2)
 pos=left_max;
 else if(x2==1)
 pos=left_min;
 myservo.write(pos); //send command to servo motor
 digitalWrite(8,LOW);
 digitalWrite(9,HIGH); //move car forward
 return;
}
case 2 :
{
 if(x2==2)
 pos=right_max;
 else if(x2==1)
 pos=right_min;
 myservo.write(pos);
 digitalWrite(8,LOW);
```

```
 digitalWrite(9,HIGH);
 return;
}
case 3 :
{
 pos=straight;
 myservo.write(pos);
 digitalWrite(8,LOW);
 digitalWrite(9,HIGH);
 return;
}
case 4 :
{
 pos=straight;
 myservo.write(pos);
 digitalWrite(9,LOW);
 digitalWrite(8,HIGH);
 return;
}
case 5 :
```

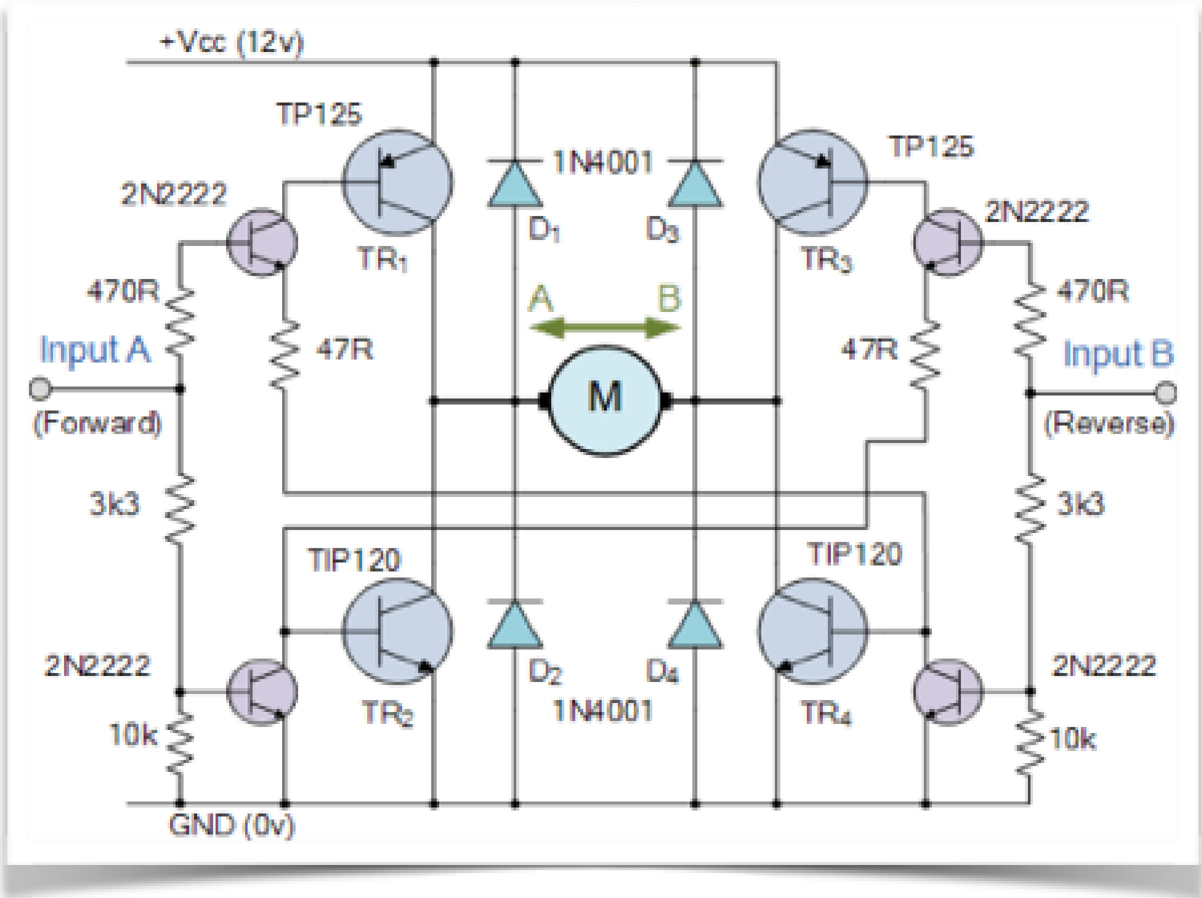
```
{
 if(x2==2)
 pos=left_max;
 else if(x2==1)
 pos=left_min;
 myservo.write(pos);
 digitalWrite(9,LOW);
 digitalWrite(8,HIGH);
 return;
}
case 6 :
{
 if(x2==2)
 pos=right_max;
 else if(x2==1)
 pos=right_min;
 myservo.write(pos);
 digitalWrite(9,LOW);
 digitalWrite(8,HIGH);
 return;
```

```
}
case 9 :
{
  pos=straight;
  myservo.write(pos);
    digitalWrite(9,LOW);
    digitalWrite(8,LOW);
  return;
}
}
  }
```

APPENDIX 3

Motor Driver H-bridge :

The H-bridge circuit, is named because the basic configuration of the four switches, resembles that of the letter "H" with the motor positioned on the centre bar. The MOSFET H-bridge is one of the most commonly used type of bi-directional DC motor control circuits. It uses "complementary transistor pairs" both NPN and PNP in each branch with the transistors being switched together in pairs to control the motor.

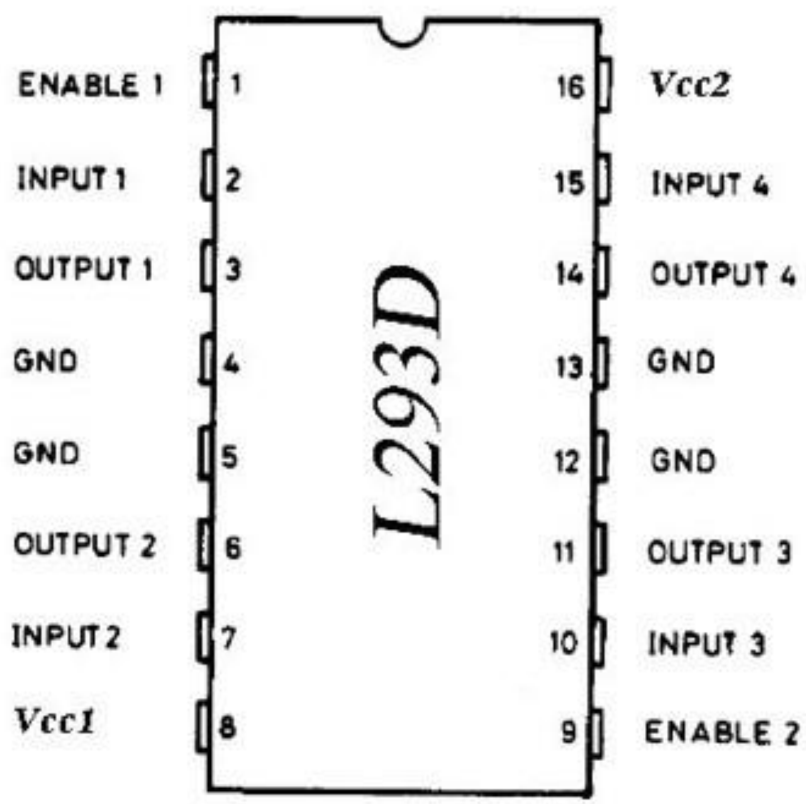


Control input A operates the motor in one direction i.e, Forward rotation while input B operates the motor in the other direction i.e, Reverse rotation. Then by switching the transistors "ON" or "OFF" in their "diagonal pairs" results in directional control of the motor. For example, when transistor TR1 is "ON" and transistor TR2 is

"OFF", point A is connected to the supply voltage (+Vcc) and if transistor TR3 is "OFF" and transistor TR4 is "ON" point B is connected to 0 volts (GND). Then the motor will rotate in one direction corresponding to motor terminal A being positive and motor terminal B being negative. If the switching states are reversed so that TR1 is "OFF", TR2 is "ON", TR3 is "ON" and TR4 is "OFF", the motor current will now flow in the opposite direction causing the motor to rotate in the opposite direction.

Then, by applying opposite logic levels "1" or "0" to the inputs A and B the motors rotational direction can be controlled as follows.

The L293D which has 2 H-Bridges are available with all the necessary control and safety logic built which are specially designed for H-bridge bidirectional motor control circuits. See Figure 6.1



Featuring Unitrode L293 and L293D Products Now From Texas Instruments

-Wide Supply-Voltage Range: 4.5 V to 36 V D Separate Input-Logic Supply

-Internal ESD Protection

-Thermal Shutdown

-High-Noise-Immunity Inputs

-Functionally Similar to SGS L293 and

SGS L293D

-Output Current 1 A Per Channel (600 mA for L293D)

-Peak Output Current 2 A Per Channel (1.2 A for L293D)

-Output Clamp Diodes for Inductive 1,2EN Transient Suppression (L293D)

The L293 and L293D are quadruple high-current half-H

drivers. The L293 is designed to provide

bidirectional drive currents of up to 1 A at voltages HEAT SINK AND from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other highcurrent/high-voltage loads in positive-supply applications.

GROUND

All inputs are TTL compatible. Each output is a complete

totem-pole drive circuit, with a

Darlington transistor sink and a pseudo-

Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. When an enable input is high, the associated drivers are enabled, and their outputs are active and in phase with their inputs. When the enable input is low, those drivers are disabled, and their outputs are off and in the highimpedance state.

With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

Dogo Douglas Saviour

INBOX | NOW VIEWING: NEW PAPERS ▾

Submit File

Edit assignment settings | Email non-submitters

AUTHOR	TITLE	SIMILARITY	FILE	PAPER ID	DATE
Dogo Douglas Saviour	Abstract	0%		891895659	07-Dec-2017
Dogo Douglas Saviour	Chapter 1	0%		891894051	07-Dec-2017
Dogo Douglas Saviour	Chapter 6	0%		891894862	07-Dec-2017
Dogo Douglas Saviour	Chapter 4	1%		891894483	07-Dec-2017
Dogo Douglas Saviour	Chapter 5	1%		891894626	07-Dec-2017
Dogo Douglas Saviour	Chapter 2	8%		891896204	07-Dec-2017
Dogo Douglas Saviour	Thesis	9%		891894763	07-Dec-2017
Dogo Douglas Saviour	Chapter 3	14%		891894268	07-Dec-2017