Data Fusion of Laser Range Finder and Video Camera

by

Mohd Taufik bin Abd Jalil

Electrical and Electronic Engineering

11392

DISSERTATION

submitted to the Electrical & Electronics Engineering Programme

in Partial Fulfillment of the Requirements

for the Degree

Bachelor of Engineering (Hons)

(Electrical & Electronic Engineering)

SEPT 2013

UNIVERSITI TEKNOLOGI PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Data Fusion of Laser Range Finder and Video Camera

by

Mohd Taufik bin Abd Jalil

A project dissertation submitted to the

Electrical and Electronics Engineering Program

Universiti Teknologi PETRONAS

in partial fulfilment for the

Bachelor of Engineering (Hons)

(Electrical and Electronics Engineering)

Approved by,

____

(Mr Patrick Sebastian)

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or one by unspecified sources or persons.

____

(MOHD TAUFIK BIN ABD JALIL)

ABSTRACT

For this project, a technique of fusing the data from sensors are developed in order to detect, track and classify in a static background environment. The proposed method is to utilize a single video camera and a laser range finder to determine the range of a generally specified targets or objects and classification of those particular targets. The module aims to acquire or detect objects or obstacles and provide the distance from the module to the target in real-time application using real live video. The proposed method to achieve the objective is using MATLAB to perform data fusion of the data collected from laser range finder and video camera. Background subtraction is used in this project to perform object detection.

ACKNOWLEDGEMENT

Firstly, I would like to express my undying and eternal gratitude to Allah S.W.T for granting me guidance, assistance and strength in completing this Final Year Project successfully.

I extend my gratitude to my supervisor and mentor, Mr Patrick Sebastian for his support, advice and guidance have been a great help in order for me to complete this project successfully. His willingness to teach and share me his knowledge contributed greatly to my project. His exceptional attitude and helpful hand has been very crucial to motivate and keeps me moving in order to finish this project

Lastly, my appreciation goes to my parents and friends who are supporting me thoroughly towards the completion of my project. I would also like to express my gratitude to all people who have contributed directly or indirectly in accomplishment of this project.

Table of Content

List of figures

List of Abbreviation

LADAR/LIDAR	LIght Detection And Ranging
PDF	Probability Density Function
1-G	One Gaussian Background subtraction Algorithm
GMM	Gaussian Mixture Model
2D	2 Dimension

CHAPTER 1

1. INTRODUCTION

1.1 Background

There are many methods of determining distance and object classification using single type of sensor such as proposed by (Mendes, Bento, & Nunes, 2004) which is inefficient to be applied using only one laser range finder due to lack of certain property of environment required for the intended use such as object detection. This situation introduces the application of multi-sensor fusion. There are many applications that require elements of environment that can only be covered by multiple sensors such as vehicle and obstacle detection for traffic uses, multiple people detection in an open environment and many more. These applications prove to be inefficient to be executed using only single sensor due to the sensor's individual drawbacks or weaknesses. Current video cameras are unable to estimate the distance of objects or obstacles in front. The current methods available are by using two cameras for stereoscopic view or by using ultrasonic sensors and laser sensors alone.

1.2 Problem Statement

Current video cameras are unable to estimate the distance of objects or obstacles in front. The current methods available are by using two cameras for stereoscopic view or by using ultrasonic sensors and laser sensors alone. There are a number of methodologies in fusing data obtained from laser rangefinder and video cameras. There is a need to determine or develop a method in fusing the multiple methods of fusing the data from the different sensors for object or obstacles ranging purposes. The single method to be investigated or developed could be used in the application of a fixed background or a moving background. The appropriate targets can then be determined and the distance be estimated from the moving or fixed background environment.

1.3 Objective

To limit the usage of video camera for range detection purpose. The current technique to detect distance target using stereoscopic view which require at least 2 cameras.

To apply a detection system that employs fusion of sensors that capable of range detection of the particularly specified target. In this case, to specifically use a laser ranges finder and a video camera.

To employ a robust module that capable of object detection and range detection.

1.4 Significance of Project

This project is to apply and develop algorithm to fuse data from multiple sensors using MATLAB. This is needed due to drawbacks of a single sensor i.e. lack of capability to obtain a certain element of environment needed or inefficient in deriving the some information that are not specified for the sensor. For example, camera can be used to estimate range but inefficient as it will require more processing power.

1.5 Scope of Study

Fusion of data obtained from video camera and laser range finder. The fusion will be performed using any suitable fusion method available.

Data extraction and extrapolation from laser range finder using MATLAB.

Background subtraction with Gaussian mixture model for tracking and classification purposes.

1.6 Relevancy of the project

Since there is no technique of determining distance and object detection using a single sensor that is fast and efficient enough has been developed for individual sensor therefore create the need for multisensory fusion. This project is relevant as it will serve as stepping stone to other project that utilizes the fusion of these 2 sensors. That is one of the reason that this project is implemented in MATLAB which is for easy understanding of the nature of the data fusion from both sensors and processing the data from both sensors.

1.7 Feasibility of the Project

The project is carried out in two semesters, in which the research, development and testing has been conducted. Throughout these period, suitable method of fusion is determined then development of algorithm using MATLAB is carried out and testing are conducted to determine the accuracy and the feasibility of the chosen methodology. Given the time period, it is feasible for the project to be carried out in the provisioned time frame.

CHAPTER 2

2 Literature Review

The project revolves around the application of data fusion of data acquisition from laser range finder and background subtraction

2.1 Data Fusion

Multisensory fusion is to integrate the data or information obtained from multiple sources (in this case is multiple sensor) to provide better information about an entity. Linas et. al. (D. L. Hall & Llinas, 2001) has stated that multisensory data fusion has been widely used in many application, even for military purposes. The idea of fusing multiple sources of information into a singular decision is not a recent development. In fact, it is easy to trace a number of fusion methods to biological origins. Humans rely on many different senses to perceive the objects in their surroundings. Our eyesight, which provides vision which is an aspect of environment that can be obtained by a sensor, is actually fusing with other sensors in our body such as skin which capable of heat and pressure detection, ears that can catch the sound and many others. Using these concept, fusion methods has been developed by using computers as the central processing unit which serves the same function as our brain to fuse all the data obtain from our bodily sensors. There are also operations in our body that unnoticed to us such as body heat regulation, pupil dilation. All of these are govern by the natural biology sensors in our body. By applying data fusion, there are a lot of advantages generally such as robust operational performance, extended spatial/temporal coverage, reduced ambiguity, increased confidence, improved detection performance, enhanced resolution (spatial/temporal) and increase dimensionality.

The common way to distinguish velocity or position of a concerning target is to apply sequential estimation techniques, i.e. Kalman filter. In other hand, to recognize an entity or object, the common method is to apply pattern recognition techniques like

Bayesian Inference. The data fusion is not restricted at a certain level only. Fusion can be perform at observation level. At this level, fusion is implemented using raw information obtained from the sensors. However, in order for data fusion to be executed at this level, both the sensors must provide data commensurate to each other. Hence, this technique is only applicable for similar type of sensor only i.e. video camera and video camera. The fusion can also be implemented at feature level. Feature extraction is a process to reduce dimensionality and redundancy of information obtained by sensor. By performing feature extraction, the data obtained will be reduced to a smaller set of data which only consist of representative feature of the data. In feature level fusion, the feature extracted from different sensors are integrated into a vector which will be processed by pattern recognition method afterward. The multisensory fusion also can be performed at decision level in which each sensors already processed their own data individually. At this level, the sensors already obtained the element of environment involved such as location, attributes and identity of the target.

IDL Process	Processing Function	Techniques
Level 1: Object Refinement	Data alignment	• Coordinate transforms • Units adjustments
	Data/object correlation	• Gating techniques [52] • Multiple hypothesis association probabilistic data association [63],[64] • Nearest neighbor
	Position/kinematic and attribute estimation	• Sequential estimation [19], [73], [75] - Kalman filter - αβ filter - Multiple hypothesis [79] • Batch estimation [69], [70], [71] • Maximum likelihood [80] • Hybrid methods [76], [77], [78]
	Object identity estimation	• Physical models • Feature-based techniques - Neural networks - Cluster algorithms [56], [57] - Pattern recognition [87], [88], [89] • Syntactic models
Level 2: Situation Refinement	Object aggregation Event/activity interpretation Contextual interpretation	• Knowledge-based systems (KBS) - Rule-based expert systems - Fuzzy logic [60] - Frame-based (KBS) • Logical templating [114], [115] • Neural networks [96], [97], [98], [99] - Blackboard systems
Level 3: Threat Refinement	Aggregate force estimation Intent prediction Multi-perspective assessment	• Neural networks - Blackboard systems [122] • Fast-time engagement models
Level 4: Process Refinement	Performance evaluation	• Measure of evaluation [2] • Measures of performance [2] • Utility theory [59]
	Process control	• Multi-objective optimization [59] - Linear programming - Goal programming
	Source requirement determination	• Sensor models
	Mission management	• Knowledge-based systems

Figure 1 - Example of data fusion algorithm and techniques

In determining what kind of architecture to be used in fusion, 2 issues need to be addressed. First, whether the fusion to achieve locational information (i.e. range, azimuth or height) for position or velocity detection or is the fusion working with parametric data for object identification. Both of this fusion purposes works at the 3 level mentioned before which is raw data level or observational level, feature level and decision level. There are variations of level of data can work with in a fusion. For example, the feature level can work with decision level. For locational information, feature level data derived from the sensors is called state vector. These state vector is utilize in feature level fusion to integrate the data from another sensors. At raw data level, unlike fusion of parametric data which require the data to be commensurate, the fusion locational information does not required to be commensurate but the data from multiple sensors have to be synchronised. This is needed as to ensure that the data will represent the actual physical entity intended to be observed. In an environment with dense object, this method is not preferable as correlations between the sensors might be very complicated. The typically used method in this level of fusion is using Kalman filter which is a sequential estimation techniques. And finally, the proposed method used in this project, decision level fusion. At this level of fusion, each sensor has to made preliminary determination of the object prior to the fusion itself. The Bayesian inference is an example of decision level fusion.

In Bayesian inference method, sensors are fused based on probability theory of Bayes. In essence, Bayes rule relates priori probability, conditional probability and posteriori probability of a hypothesis. Bayes rule mathematically is $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$ where A and B represent data from different sensor.

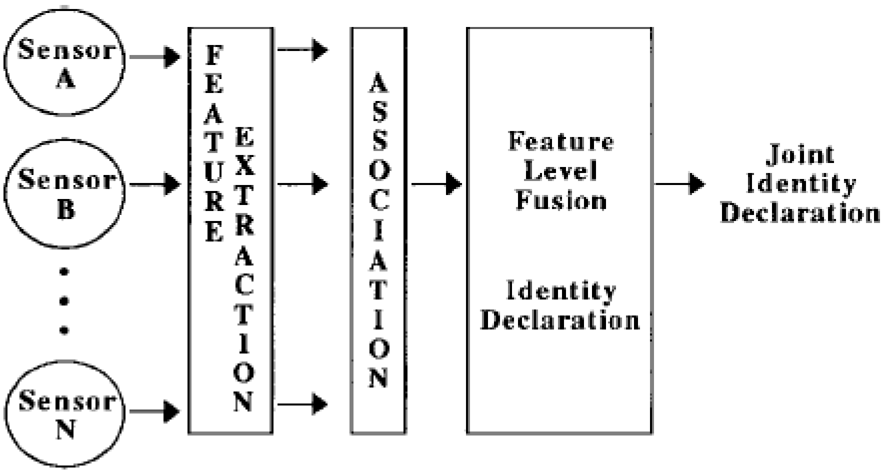

Figure 2 - Decision level fusion architecture

Monteiro (Monteiro, Premebida, Peixoto, & Nunes, 2006)has proposed almost similar system to this project. He stated that combination of these to sensors necessary prior to outdoor environment that are heavily affected by weather and surrounding environment. To ensure the robustness and full functionality of the module, he applied the same idea that being proposed in this project. In his paper, he has proposed to use laser range finder for detection and object tracking using 2D linear Kalman Filter as proposed by (Borges & Aldon, 2004) as their segmentation method. For this method, tracking and data association is associated to a laser referential system where the tracked object is said to be developed in time according to dynamic state equation driven by process noise. Raw data are being transformed into Cartesian coordinated the filtered using Kalman filter for object tracking(Bar-Shalom & Li, 1995).

Monteiro also stated that despite the measurement is in non-linear model, object motion is modelled by Cartesian coordinates in form of linear function. The proposed method for ladar based classifier was GMM classifier which is using finite GMM which parameter is determined during a supervised training. To extract information from the images, Monteiro uses *Haar-Like feature* to extract feature and run it through AdaBoost classifier to combine the feature extracted for more effective classification. The outputs of the classifiers are then fused using Bayesian framework to ultimately classify the object.

In this particular project, the main sensor is the video feed and the laser range finder is used to support main sensor. The system primarily must be capable of 'seeing' the object and the laser range finder is to provide range of the object from the sensors.

2.2 Background Subtraction

Background subtraction is very common in object detection for video capture from a static camera. In background subtraction, the algorith used should be able to model the pixel to be background or foreground. In this method, background will be ignored and the concerned area should be foreground. The foreground pixel then will be process for detection or tracking purposes (Toyama, Krumm, Brumitt, & Meyers, 1999). The process of recognising the background pixel is called background maintenance. However, in real life situation, change of illumination, fluctuation of small pixel due to enviromental changes like clouds movement, leaves movement will affect the performance of background maintenance. This is where Gaussian Mixture Model approach is applicable (Stauffer& Grimson, 1999). Each pixel is modelled as mixture of Gaussian distribution and each pixel will be classfied by the Gaussian distribution to be either background or foreground.

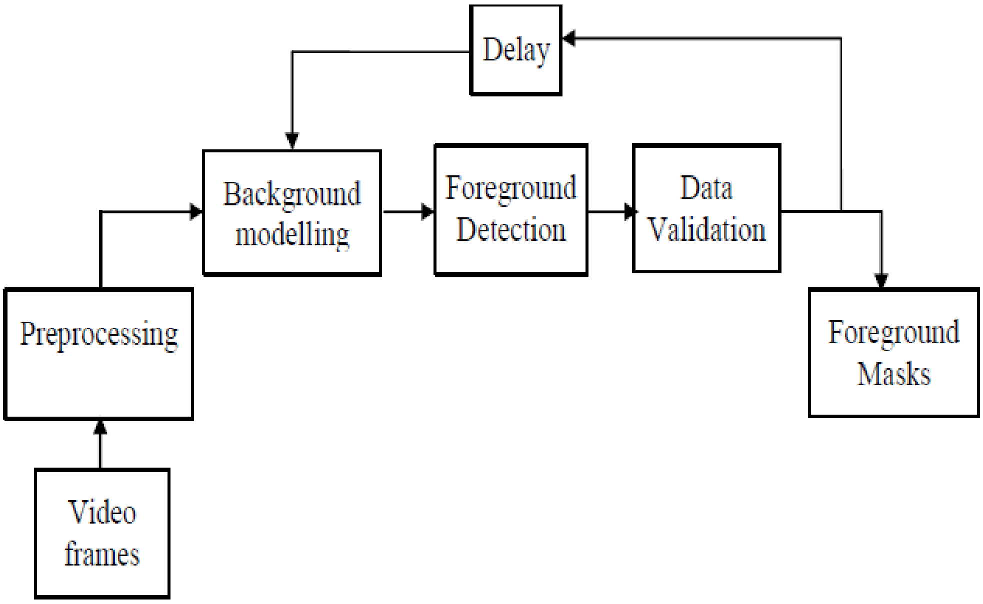

Figure 3 - Generic flow of background subtraction

Figure 3 shows the general flow of background subtraction method. Three main step in the sequence should be pre-processing, background modelling and foreground detection. In the pre-processing step, temporal and spatial smoothing of the video frame is performed to filter out the transient noise. Color or luminance usage for object identification will also be determined and executed at this stage. The next stage should be background modelling. In this stage, the generally, there are 2 category of background modelling which is recursive and non-recursive. In this stage the background maintenance requirement to be perform in the modelling is considered and the appropriate algorithm to meet the requirement is implemented. At the foreground detection stage, the initial video frame is compared to the modelled background in order to derive possible foreground pixel from the input frame.

.

Chapter 3

3 Research Methodology

The laser range finder and the video camera are set up and controlled by a single PC in which the whole system is running. Basically the system extract raw data obtained from laser range finder and provide data such as position, angle and distance of the target from the system. The live video provided by the camera then will be used to validate and confirms the data from laser range finder.

3.1 Project Activity

In order to fully execute the project, data from laser range finder has successfully acquired and extrapolated using MATLAB as interface. The raw data obtained from the laser range finder is actually reliable and accurate. However, to ensure reliability and data integrity, an algorithm need to be applied. The algorithm and encoding protocol of the URG scanner is applied to ensure data integrity and smooth communication between PC and the scanner.

Method for object detection which is specifically background subtraction is still being studied as there are many available methods with their own pros and cons. The most suitable method currently proposed for this project is Mixture of Adaptive Gaussian.

The reason that this method is chosen is due to its ability to copes with multimodal background distributions. Every values of the particular pixel is being modelled as a mixture of adaptive Gaussian which basically able discern multiple surface or aspect of background in a pixel and also able to works under changes of lighting condition. For this method, each iteration Gaussians are being evaluated using a simple heuristic to determine which ones are mostly likely to correspond to the background. Pixels that do not match with the “background Gaussians” are classified as foreground. Foreground pixel is classified using 2D connected component analysis.

The advantages of using this method are it changes the threshold for each and every pixel. It automatically selects different threshold for every pixel modelled. The threshold is actually changes according to the pixel time to time. Some objects are allowed to be in the background and still does not affect the modelled background. This method also has a fast recovery rate after the background model is ruined or disturbed.

One of the drawback of this method is it cannot perceive in sudden change of light. In this method, there are quite a number of parameters that should be weighted properly. To determine the appropriate value for the parameters is essential for this method to turns out well.

3.2 Data Extraction from Laser Range Finder

For this method, MATLAB will be used to obtain raw data from laser range finder. To obtain raw data, laser range finder will project the laser and based on time of flight calculation, the distance from appropriate target can be determine. (Mendes et al., 2004) stated in the DATMO (Detection and Tracking of Moving Object) system, the application of scan segmentation is translate the signal in segments obtained from laser range finder to represent each objects it detected. It will need to have several segment of scan which consists of clusters of data received from the laser range finder. In the DATMO system, scan segmentation needs the reading to be clustered based on proximity of detected points. Then, the clustered data will be able to determine whether the points detected belong to the same object or not. Consecutively, after applying clustering, line fitting is applied to obtain good object characteristic by approximating the polygonal shape in the surrounding environment. Then to ensure integrity and consistency of the system, joining of broken object is necessary as it may greatly influence the outcome and sustainability of the system. Leg detection and broken lines detection is among the proposed methods to cater to this problem(Cui, Zha, Zhao, & Shibasaki, 2008). In this particular project, scanning range finder specifically **URG-04LX-UG01** is used. The reason that this particular laser range finder is used is because of it is small, affordable and accurate laser scanner that is perfect for robotic applications. The URG-04LX-UG01 is able to report ranges from 20mm to 5600mm (1mm resolution) in a 240° arc (0.36° angular resolution).

The LIDAR scanner consume 5V 500ma of power at a time. It is capable of performing 10 scan per second using Laser class I which is 785nm wavelength. The sensor is actually a laser diode mounted on top of a brushless motor. It is capable of scanning in 240° angle as the motor rotates during the scans and the encoder will synchronize the laser pulses with the position of the laser receiver in the sensor. Scan area is 240º semicircle with maximum radius 4000mm. Pitch angle is 0.36º and sensor outputs the distance measured at every point (683 steps). Laser beam diameter is less than 20mm at 2000mm with maximum divergence 40mm at 4000mm.

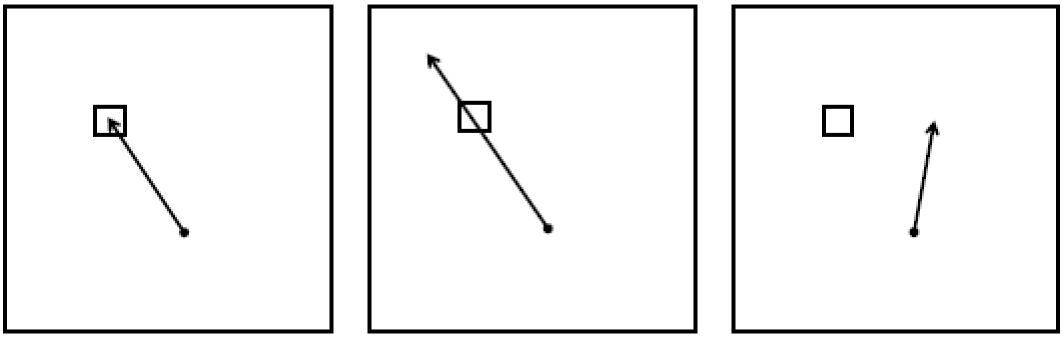

Figure 4 (Left) - The beam is blocked by opaque object or surface thus limiting the range. (Middle) The beam passes through as the cell is empty. (Right) The beam is projected out of working range of the systems

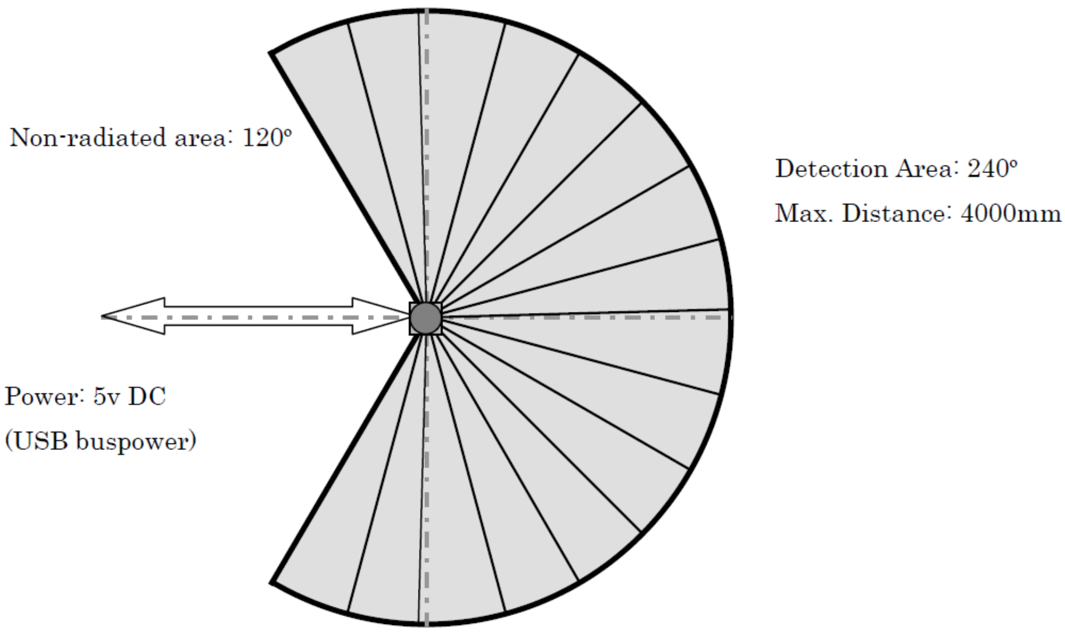

Figure 5 - Detectable area for URG-04LX-UG01

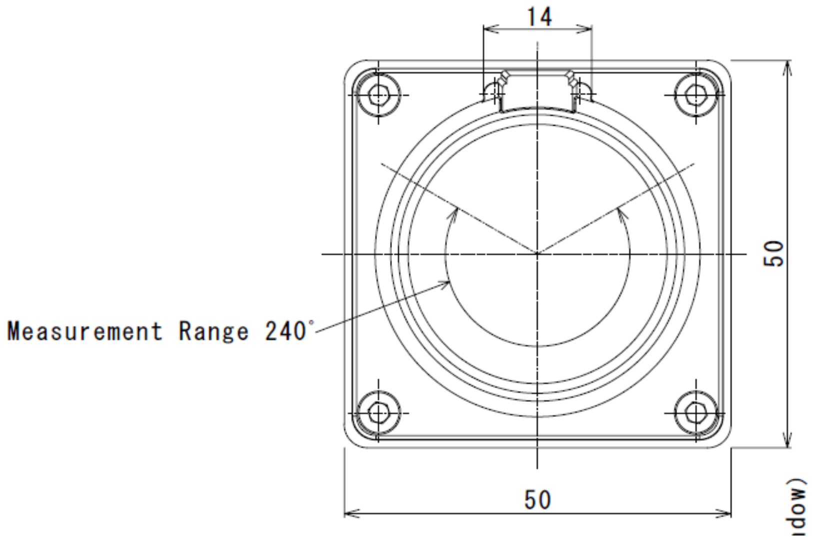

Figure 6 - Top view of the laser range finder and its scanning range

Hypothesis generation of the data from laser range finder consists of 2 step which is to generate cluster and to determine whether the cluster is object or not. In this step, clustering algorithm applied is

- The scan is initialize and a set of 682 data is obtained
- Read data from left to right $d[n] = (R_0 : R_{682})$
- Filter out the region that are not included in intersection of both sensor $d[n] = (R_{left} : R_{right})$
- For $n = 1; R_{left} < n < R_{right}$, $n{+}{+}$
- If $d[n] > \max\_range$ or $d[n] < \min\_range$, $d[n] = 0$
- If $d[n] \neq 0$, $d[n]$ will be assign to a cluster, this will be loop
- If $d[n] = 0$ again, end the cluster, start a new cluster
- If $d[n] \neq 0$, $d[n]$ will be assign to a cluster, this will be loop

Figure 7 - The algorithm to cluster the laser range finder data

3.3 Data Extraction from Video Camera

To extract data from the video camera, MATLAB software are used. MATLAB is capable of displaying real-time live video feed and capable to perform image processing needed to analyse the video feed. To specifically obtain video feed from video camera, a MATLAB tool called Image processing toolbox is utilized. The background subtraction will also be performed by MATLAB.

3.3.1 Background Subtraction

Laser range finder is assigned to specifically for object detection and range estimation, and to classify and position the object will be handled by the video camera. For this project, the method of background subtraction is applied to achieve the purpose of classifying and positioning the object. In a nutshell, background subtraction is a process of determining “foreign” or unfitting object in a sequence of statistically well described manner from a model. Basically, background subtraction is to describe the behavioural change of a set of background statistically(Zivkovic, 2004).

In the background subtraction method used, which is Gaussian Mixture Model, the training set is updated to adapt to changes, such as illumination change. For background maintenance, in the given time frame, each sample frame within the given time is updated into the training set and re-estimate. However, in the sample frame, there are possible pixel that belong to foreground pixel. Using means and variance of the possible foreground component in the Gaussian distribution to investigate the possibility of suspected pixel to be foreground or background. This is also to find out the amount of possible foreground pixel portion that does not influence the background model.

For adaptive Gaussian Mixture Model algorithm, in order to obtain full adaptation to the scene, the algorithm automatically selects the needed number of component pixel.

3.4 Data Fusion of Laser Range Finder and Video Camera

Modelling the laser range data the individual pixel are not suitable for this method as the camera resolution are comparatively high to the resolution of the laser range finder. In this project the resolution of the camera is 640x480 pixel covering only estimated of 56° of the angular view whereas the laser range finder resolution is 682 reading in 240° angular coverage. The laser range finder and the camera are mounted together to overlay the viewing angle of the camera to the detected range of the laser range finder. By overlaying the effective range of both sensor, the fusion at decision level are easier to be made.

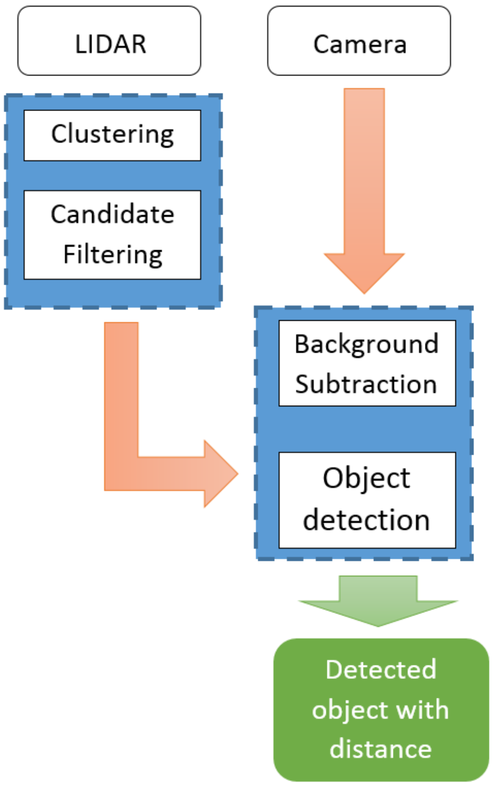

Figure 8 - The overall of the system

To fuse both of the data obtain, Bayesian approach will be used in this system. The best suggestion would be to use direct approach of Bayesian Law. The direct method of applying fusion is to match the segmented image with the data from the laser range finder. The data from the laser range finder can be clustered or averaged to obtain the most appropriate value of range from the designated object. The detected object which should be isolated from the background can be matched directly through algorithm in MATLAB. The segmentation of the data from laser range finder is to determine the location and range of the particular object. In order to detect the object using laser range finder, the data from laser range finder are observed from left to right. A cell that deviate from maximum value will be noted and the cell next to it will be observe. If it reads a value close to previous reading, it will be clustered together. This will go on until the column yield maximum value or big difference to the previous cell. A cluster or a segment of cell will determine the location of the object in accordance to the angular position of the cell. To obtain the distance, the values from the columns will be averaged and the outcome will be the average distance of the object.

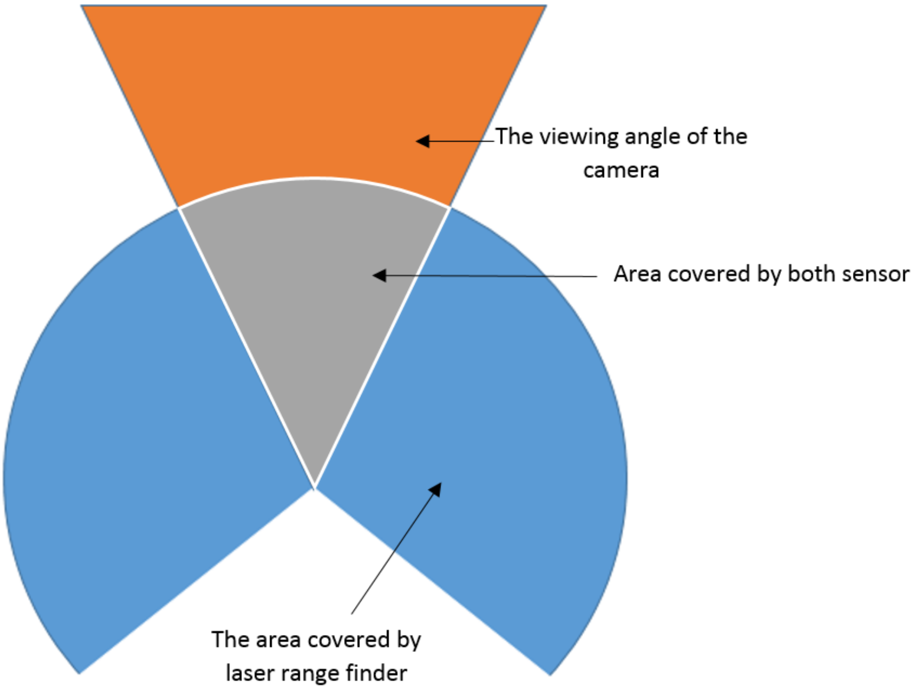

Figure 9 - Viewing range of both sensors

The hypothesis generated from the data obtained from the laser range finder will be verified by the video camera in this stage. Data from laser range finder are acquired simultaneously with a frame of picture for a part of simulated scenario. In order to simulate the system, a set of data from laser range finder measuring an object at a given time has to be collected and simultaneously a picture is taken. Then, the object is moved and the data of both sensor will again be taken, this will continue until the object goes out of the sight of the camera. The frames of pictures are then connected to form a video file which is a requirement for background subtraction. In the background subtraction, the system will be able to discern background and object. In order to this in an immaculate manner, the webcam is attached to the URG scanner.

Figure 10 - The camera mounted to the laser range finder

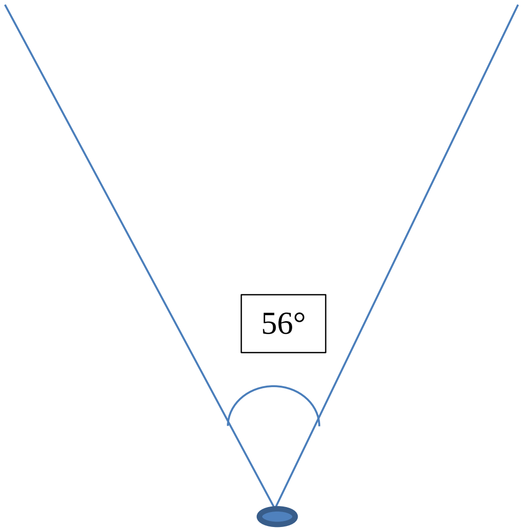

Figure 11 - The viewing angle of the camera

However the angular coverage of the laser range finder is from -240° - 240°. So, in order to efficiently execute the system, the laser range finder angular range must be limited to the match the whole angular view accommodated by the camera. Nonetheless, the mounting of the camera is not actually accurately on the centre of the URG scanner but a few centimetres in front of the scanner. Somehow, that makes the area covered by the laser range finder is wider. Since the main sensor in this case is camera so the offset does not really bother the whole outcome of the experiment.

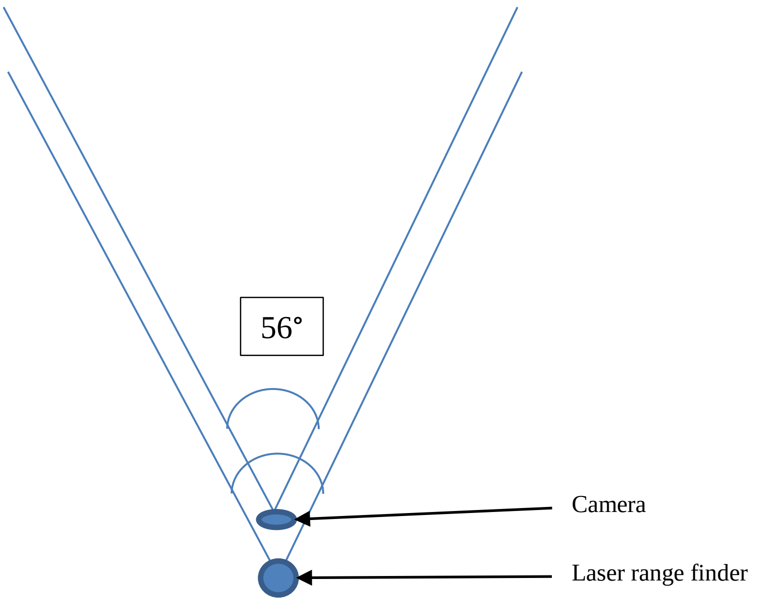

Figure 12 - Angular view of the camera overlapped with angular range of the finder

3.5 Key Milestone

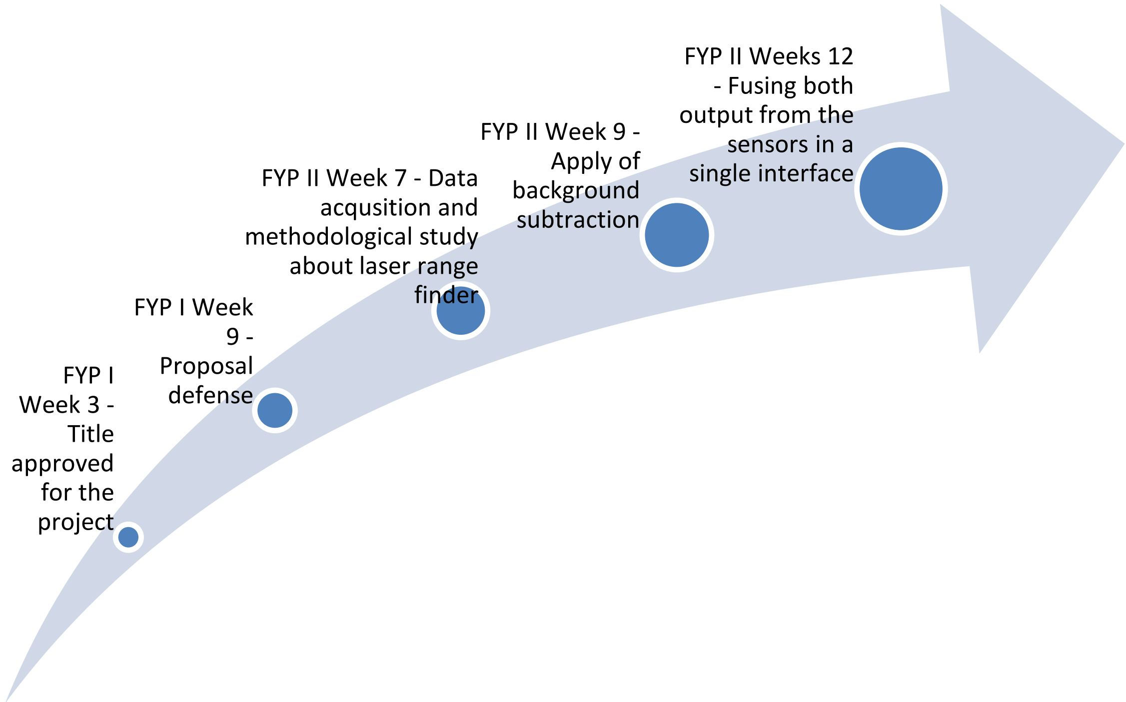

Figure 13 - Key milestone for the project

3.6 Study Plan

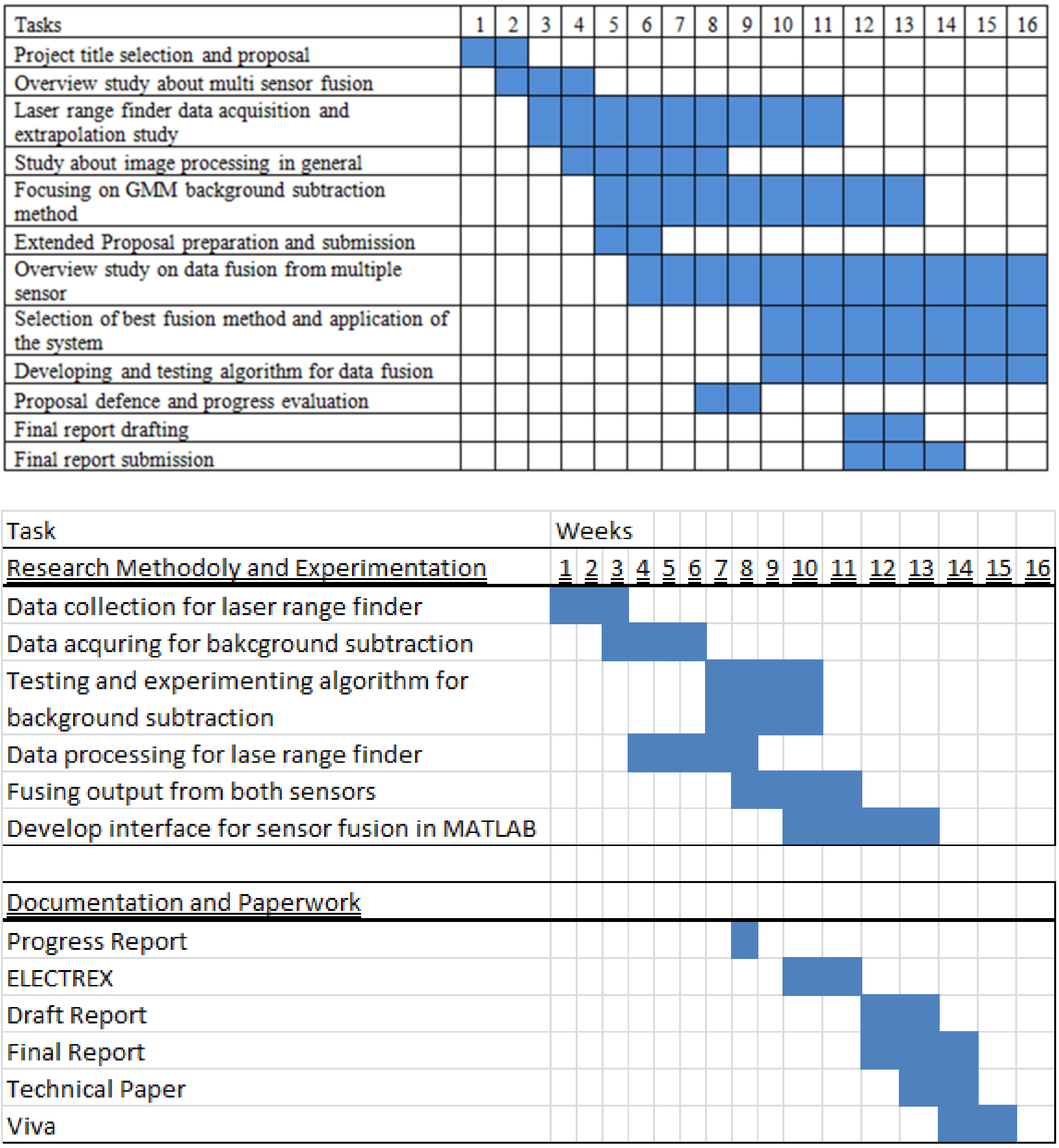

Figure 14 - Gant Chart for the project for FYP I (top) and FYP II (bottom)

3.7 Tools required

In this project, tools utilized are:

3.7.1 MATLAB

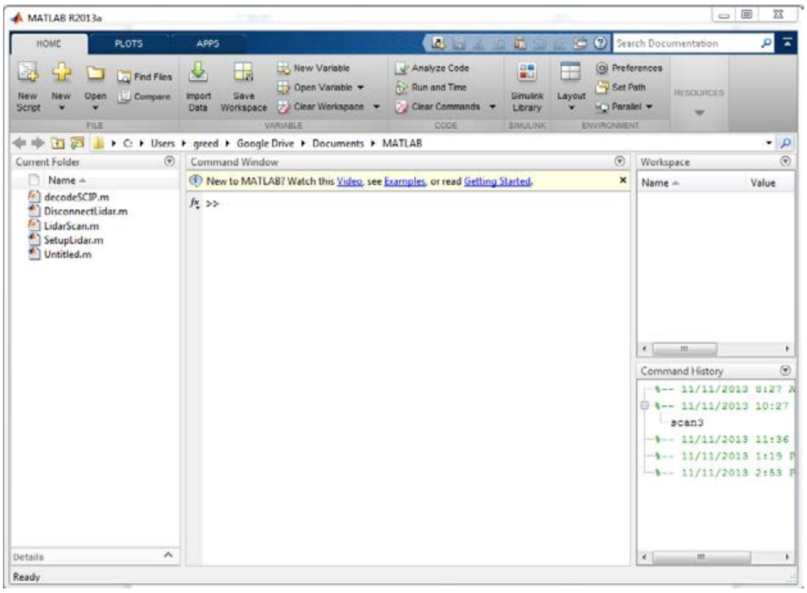

Figure 15 - MATLAB interface

3.7.2 URG-04LX-UG01 scanner type Laser Range Finder

Figure 16 - URG-04LX-UG01

3.7.3 KWORLD V1000 Webcam

Figure 17 - KWORLD V1000 Webcam

CHAPTER 4

4 RESULT AND DISCUSSION

4.1 Data Gathering and Analysis

4.1.1 Laser Range Finder

In order to view the data obtain from the laser range finder, the software provided by the manufacturer of the laser range can be used. The software is capable of efficiently and consistently provide readings of 240° arc and distance from 20mm to 5600mm in the angular range.

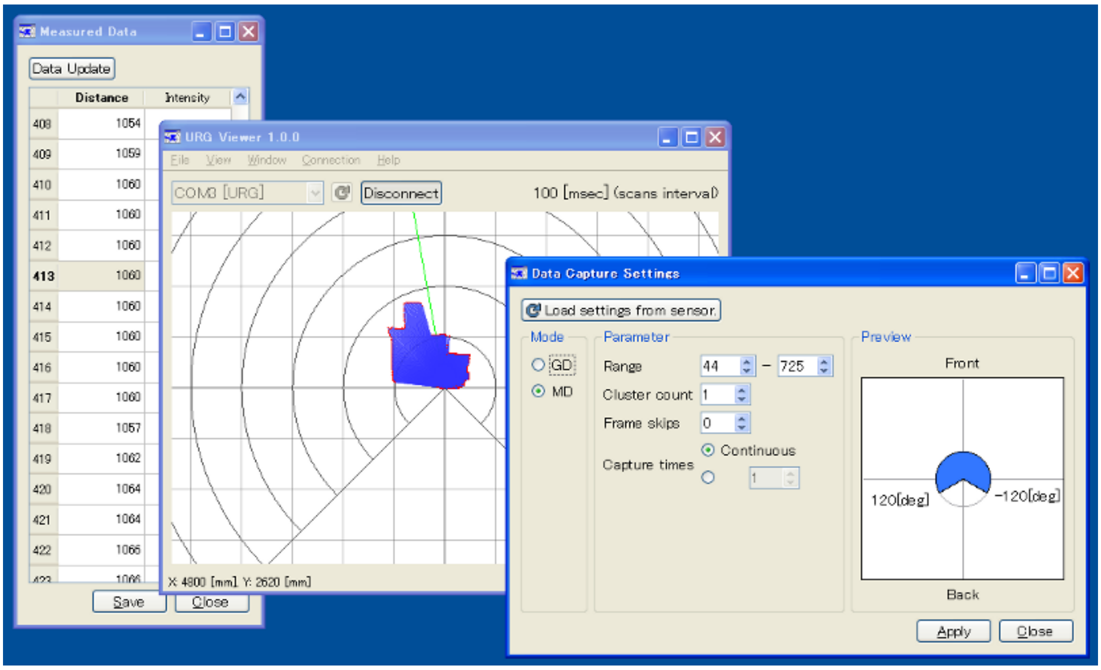

Figure 18 - URG Viewer

However, in this project the data needed to be acquired using MATLAB as interface. Thanks to Shikhar Shrestha from IIT Bhubaneswar for developing the MATLAB code for the laser range finder interfacing in MATLAB(Shrestha, 2012). The MATLAB code capable of initiating a LIDAR Scan and Returns a range vector of 682 elements after a Lidar Scan from minimum step to maximum step. Range of the scan corresponds from -120 to +120 degrees. Communication of data from laser range finder is made via SCIP 2.0 protocol. Commands of SCIP protocols are used to interact with the laser range finder. SCIP commands are classified as "Sensor information command", "Distance acquisition command", and "Mode setting command". The laser range finder must receive certain order or command in order for it to receive appropriate range value in return. These are the commands required to be received

- Start step and End step to receive distance data
- Scan interval
- Cluster Count,
- Number of scans

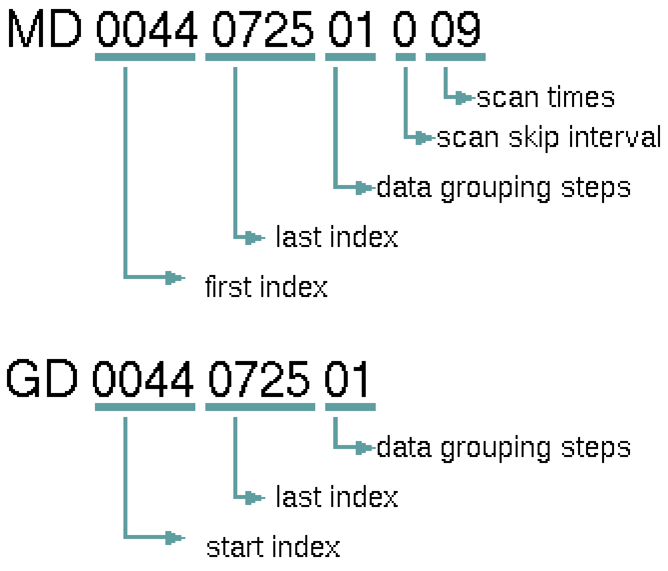

Figure 19 - Parameters of commands to get distance data

When the data transmitted to the laser range finder, it will send back this type of raw data.

Send data

```
GD0044072501
```

Array of received data


```
GD0044072501
00P
0DKO>
00i00i00i00i00k00k00n01101101101101101101100o00m00o00o0130130140]1401201201401501701
7017016017017016016015015015014014014014015010501801<01<01?01D01D01D01F01F01L01O01R0
1T01V01W01X01X01X01Z01Z01Ze01\01b01j02;02`09H09H09Z09Z09_0:90:90:@0:@0:@0:;0:@0:;0:;
0:90:90]9Z08X08408408408608608608408408408408908908908908808608308008008V007m07m07j0
7h07h07h07d06E04D04>04=04=04>04C04H04H04I04J04K04U04Ue04X04X04X04W04W04W04W04[04]04_
04`04`04h04l04l04n05005005305;05>0N5D05F05J05M05Q05T05W05[05]05^05`05f05f05m05n06506
5065068065065060906:06:06;06<06>06A06L06L06N06S06T06d07S07[09D0hH0hH0hH0gO0fk0fDV0eg
0eU0e@0db0db0db0000000000000000000000000a40`N0_o0_`0_G0_=0^a0^^I0^<0]h0]W0]@0]00\X0\
L0[l0[f0[S0[?0[00Zi0ZJ0ZC0Z70Z70Z70Z90Z90Z290Z40Z00XR0XR0XR0XR0XR0XO0XD0W]0VT0VT0VM0
V;0Um0Uc0U]0UQ0UJ0UC0U9Y0Te0Tc0T^0TK0T=0T70T60Sm0Sf0Sf0SR0SO0SD0S?0S70Rn0Rh0Rh0Rd0R]
0RK0YRD0RD0R70R60R20Qo0Qb0Q^0Q\0QV0QL0QI0QH0QC0Q50Q40Q30Po0Pk0Pi0Pg0PDa0P[0PR0PR0PQ0
PI0PI0PG0PB0PB0P@0P?0P:0P90P00P00Oh0Od0Oc0Oc0O`0O_[0O]0O]0OZ0OZ0OZ0OZ0OZ0O[0O[0Og0PO
0PO0PO0PL0PL0PI0P90P90O_0OP0OP0kOP0Od0P50P50P>0PG0PG0PC0PC0PC0Oa0OH0OH0OH0OJ0OK0OL0O
K0OK0OL0OP0O1Q0OQ0OQ0OR0OR0OT0OT0OU0OZ0OZ0O[0O[0O\0O]0Oc0Oc0Oc0Od0On0On0Oo0OoY0P40P4
0P80P=0PC0PE0PE0PE0PN0PN0PP0PX0P`0Pb0Pg0Ph0Pm0Q90Q90Q90Q?0?QC0QF0QI0QM0Q[0Qa0Qc0Qi0R
20R20R=0RA0RG0RO0RR0RX0R]0Rj0S10S20S90ST@0SJ0SP0SS0Sa0Sk0T80T:0T>0TI0TN0T]0Ta0Tl0U40
U;0UN0UR0UV0Ul0V20V?50VC0VQ0Va0W30W50WH0Xg0Xn0Xn0Xm0Xm0Xm0Xm0Z30Z<0Zb0Zb0Zb0ZW0ZW0ZW
0EZX0[20[50[S0[a0\;0\G0\V0\c0]=0]T0]a0^00^E0^[0^k0_J0_Y0`30`E0`Y0`2g0aE0aW0al0bK0b\0
c10cH0ck0d;0dS0dg0eF0ek0fE0f_0g?0g]0h;0iV0j`0jaW0jc0lY0l]0la0le0m>0mn0n[0oQ10110i115
12W000007000000000000000000000L000000071?d1?d1Af1Af1B800000000000000000000000000000000
000000000j00000000000000000000000000000000000000000000000000000000000000000000000000
00000005@05905905905805304m04N03P03F03@02n03202i02b02Y0U2:02101h01h01h01d01m01n01o02
002002001i01d01d01d01l01l01l01l01ml01o01o01o01o02102102102101k01k01k01k01k01h01_0
1S01Q01P01P01P01PW01O01Q01O01N01N01N01N01M01M01G01I01H01G01H01H01G01E01A01>01=01=0J1
=01:0180170170160170180180180190190190190170140140140140140139
```


First 3 return lines are echo back, status and timestamp respectively.

```
GD0044072501
00P
0DKO>
```

- 1st Line = Echo back of sent command
- 2nd Line = Status code that represent normal(00) + Checksum(P)
- 3rd Line = Time stamp + Check sum

682 individual distance data are conveyed simultaneously after first 3 lines.

Distance data and time stamp obtained from the URG are in decoded form composed of ASCII characters.

Each line of received distance data consist of 64 bytes of distance data and 65th byte is a check sum of that line.


```
00i00i00i00i00k00k00n01101101101101101100o00m00o00o0130130140]1401201201401501701
70170160170170160160150150150140140140140150 10
```


In addition, if obtained distance is lesser than first step of the measurement range value, it means measurement error. Measurement error occurs when obtained data is out of range. Another cause for error is during measurement of lustrous metals or reflected signal is very weak for calculation.

In MATLAB, after decoding the raw information received from the URG scanner, the data received in an array of number arranged in the order of minimum step to maximum step.

```
Columns 301 through 306

    4246    4233    4230    4216    4213    4210

Columns 307 through 312

    4210    4210    5283    5283    5318    5285

Columns 313 through 318

    5283    5277    5272    5269    5269    5266

Columns 319 through 324

    5246    5239    5234    5233    5233    5233

Columns 325 through 330

    5233    5233    5242    5242    5233    5233

Columns 331 through 336

    5233    5233    5240    5240    5240    5236
```

Figure 20 - A small portions of data extrapolated by MATLAB after decoded.

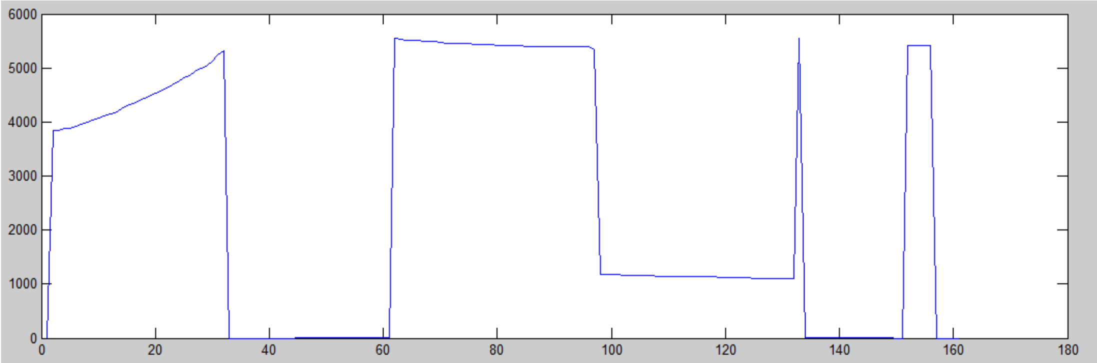

Figure 21 - The initial data obtained from a scan

As we can see that the range data is nothing but sequence of numbers. However there is more pleasing way to convey the data. Using URG Benri, software provided by Hokuyo, we can see the in a more realistic and understandable way. The important thing is to analyse and locate the location of the particular object of interest in the reading. In a sequence of number, it is quite a trouble to pinpoint the location of the object that assigned for detection. So, the software is an added value to enable us to better understand and analyse the data retrieved in MATLAB. After obtaining the data of range in form of sequence of number, the data will be segmented to determine the range and location of the object.

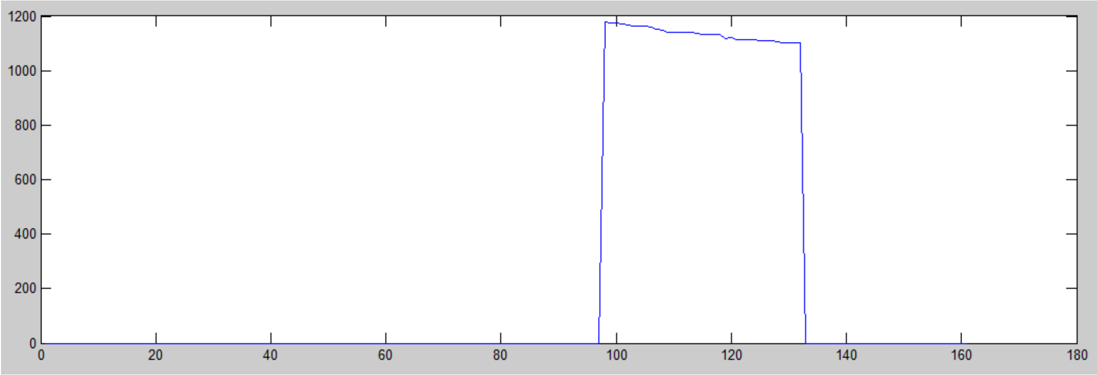

Figure 22 - The data filtered and clustered

4.1.2 Video Camera

The main element in the project is the video feed as it will provide visual insight to the targeted environment and the target itself. In this project a webcam with about 56° arc of viewing angle with a resolution of about 5 Mega pixels is being used to capture images. In this particular part of the project, sequences of pictures are captured in a same background view. This is enabling us to subtract the background using background subtraction algorithm in MATLAB.

Figure 23 - Sequence of images capture

As we can see the images are captured in a same background which will be filtered out in the background subtraction process leaving on the box which is the target specified in this particular scenario.

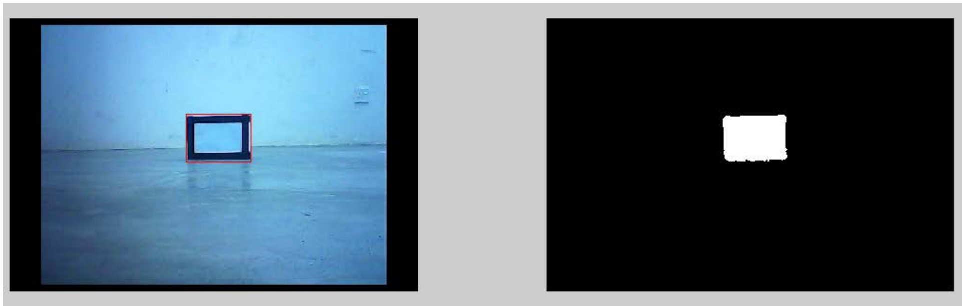

Figure 24 - The target is isolated from background and boxed

4.1.3 Data Fusion

The applied sensor fusion for this project is Bayesian Inference. Using Bayesian Inference, the range data from laser range finder can be translate into a range of the object from the laser range finder and combined with the masked area in the data obtained from camera.

Figure 25 - The data from the laser range finder and the video frame fused

4.2 Experimentation and Prototype

In order to test the functionality and the accuracy of the fusion, a series of testing has been conducted. In the testing, the measurement of distance from the object to the sensor are manipulated in each of the frame taken. In one of the testing, a single object is set to run across the viewing angle of the camera and the distance of the object to the sensor for the testing are 1.5 meter at all time.

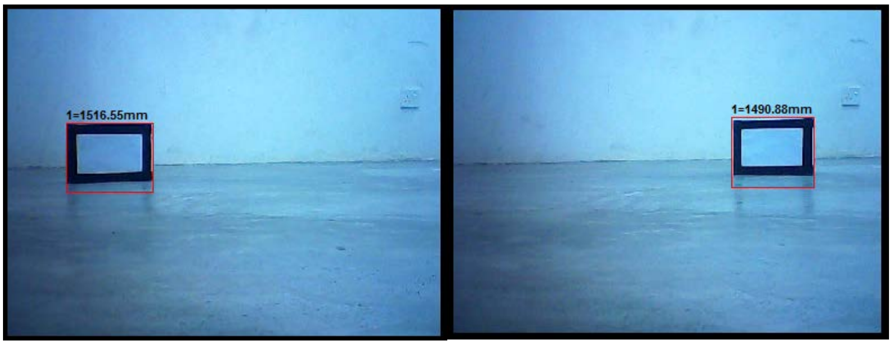

Figure 26 - The first testing conducted to observed the accuracy

The first testing yielded the data in the table in Appendix A. From the observation, the system yielded error less than $\pm 3cm$. In this testing the error occur may be due to orientation of the targeted object or human error during measuring the position of the particular target.

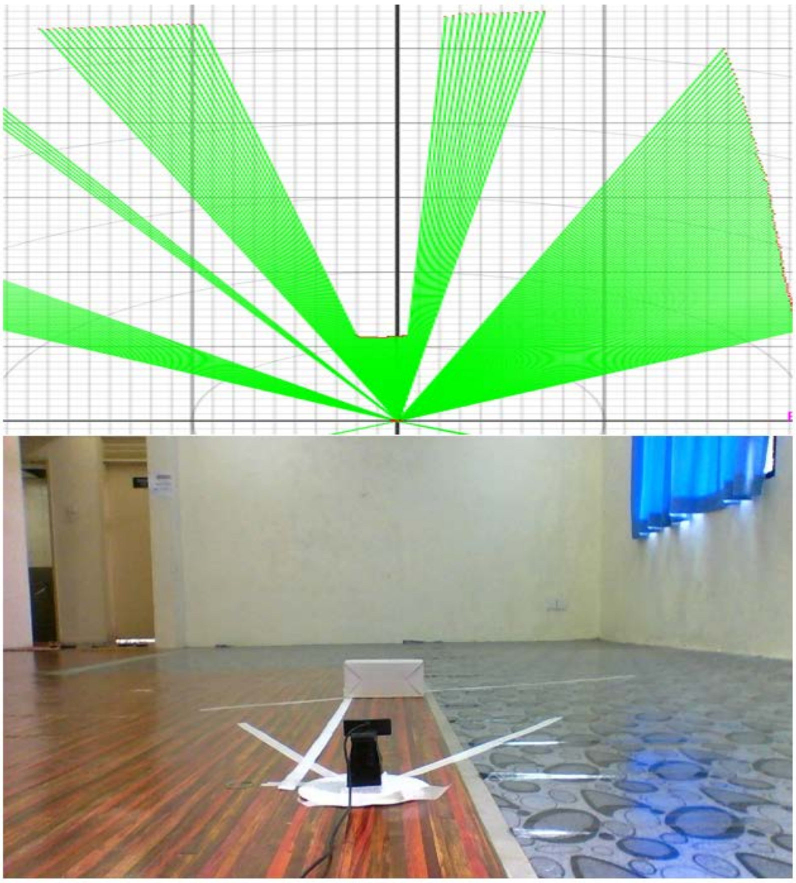

Figure 27 - The layout during one of the testing. The figure on the top is actually the data from laser range finder conveyed by the software

In second set of testing, the object is measure in a straight line from 0.5m from the sensors up until 1.4 metre range from the sensors. The distance of object between 2 frames is 10 cm apart. Based on data from Appendix A, the error is $\pm 3cm$ which may occur due to placement orientation of the box or human error in measuring the distance or placing the object. Many test has been conducted but due to lighting condition which drastically changes at times and poor camera lighting adaptability, the background subtraction could not identify the object thus failed the fusion.

Figure 28 - The among the frames recorded from the second testing

CHAPTER 5

5 CONCLUSION

As this project has been progressing, some of the objectives of this project are being accomplished. This project focuses on using 1 camera and 1 laser range finder for object detection and ranging purposes. The objective of the project have been accomplished. This project actually very simple and it will serve as initiation for other projects that requires laser range finder and video camera as one of their sensors. For example, this project could be developed in to obstacle detecting system for vehicle, people counting in a large area such as mall or hall and traffic counting.

5.1 Relevancy to objective

So far this project has been going through the right path. This project has been going on at the pace relevant to the timeline and the milestone decided. With proper guidance and support from supervisor, the project is going as intended as stated in the initial objective which is to establish a module that combines the usage of laser range finder and data from video camera. It is crucial that the project need to utilize the laser finder and video camera and also using MATLAB as interface software to fuse the data as it will serve as reference for future project.

References

Bar-Shalom, Yaakov, & Li, Xiao-Rong. (1995). Multitarget-multisensor tracking: principles and techniques. *Storrs, CT: University of Connecticut, 1995.*

Benezeth, Yannick, Jodoin, Pierre-Marc, Emile, Bruno, Laurent, Hélène, & Rosenberger, Christophe. (2010). Comparative study of background subtraction algorithms. *Journal of Electronic Imaging, 19*(3), 033003-033003-033012.

Borges, Geovany Araujo, & Aldon, Marie-José. (2004). Line extraction in 2D range images for mobile robotics. *Journal of Intelligent and Robotic Systems, 40*(3), 267-297.

Bracio, Boris R, Horn, Wolfgang, & Moller, Dietmar PF. (1997). *Sensor fusion in biomedical systems.* Paper presented at the Engineering in Medicine and Biology Society, 1997. Proceedings of the 19th Annual International Conference of the IEEE.

Cui, Jinshi, Zha, Hongbin, Zhao, Huijing, & Shibasaki, Ryosuke. (2008). Multi-modal tracking of people using laser scanners and video camera. *Image and vision Computing, 26*(2), 240-252.

Friedman, Nir, & Russell, Stuart. (1997). *Image segmentation in video sequences: A probabilistic approach.* Paper presented at the Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence.

Hackett, Jay K, & Shah, Mubarak. (1990). *Multi-sensor fusion: a perspective.* Paper presented at the Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on.

Hall, David David Lee, & McMullen, Sonya Anne Hall. (2004). *Mathematical techniques in multisensor data fusion*: Artech House.

Hall, David L, & Llinas, James. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE, 85*(1), 6-23.

Hall, David L, & Llinas, James. (2001). *Multisensor data fusion*: CRC press.

Hashemipour, Hamid R, Roy, Sumit, & Laub, Alan J. (1988). Decentralized structures for parallel Kalman filtering. *Automatic Control, IEEE Transactions on, 33*(1), 88-94.

Luo, Ren C, & Kay, Michael G. (1989). Multisensor integration and fusion in intelligent systems. *Systems, Man and Cybernetics, IEEE Transactions on, 19*(5), 901-931.

Luo, Ren C, & Kay, Michael G. (1990). *A tutorial on multisensor integration and fusion.* Paper presented at the Industrial Electronics Society, 1990. IECON'90., 16th Annual Conference of IEEE.

Mendes, Abel, Bento, Luis Conde, & Nunes, Urbano. (2004). *Multi-target detection and tracking with a laser scanner.* Paper presented at the Intelligent Vehicles Symposium, 2004 IEEE.

Monteiro, Gonçalo, Premebida, Cristiano, Peixoto, Paulo, & Nunes, Urbano. (2006). *Tracking and classification of dynamic obstacles using laser range finder and vision.* Paper presented at the Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

Shrestha, Shikhar. (2012, 17 May 2012). Hokuyo URG-04LX LIDAR Driver for MATLAB. Retrieved 25 July 2013, 2013, from http://www.mathworks.com/matlabcentral/fileexchange/36700-hokuyo-urg-04lx-lidar-driver-for-matlab

Thomopoulos, Stelios CA. (1990). *Sensor integration and data fusion.* Paper presented at the 1989 Advances in Intelligent Robotics Systems Conference.

Toyama, Kentaro, Krumm, John, Brumitt, Barry, & Meyers, Brian. (1999). *Wallflower: Principles and practice of background maintenance.* Paper presented at the Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on.

Wren, Christopher Richard, Azarbayejani, Ali, Darrell, Trevor, & Pentland, Alex Paul. (1997). Pfinder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 19*(7), 780-785.

Zivkovic, Zoran. (2004). *Improved adaptive Gaussian mixture model for background subtraction.* Paper presented at the Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on.

Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* (Vol. 2). IEEE.

APPENDICES

APPENDIX A

The result of the first testing of the fusion. The distance measured at all-time should be 1500mm.

No. of frame	Distance from sensor (in mm)
1	0
2	1.4933e+03
3	1.5066e+03
4	1.5056e+03
5	1.5166e+03
6	1.5145e+03
7	1.5166e+03
8	1.5152e+03
9	1.5171e+03
10	1.5140e+03
11	1.5155e+03
12	1.5110e+03
13	1.5234e+03
14	1528
15	1.4960e+03
16	1.4909e+03
17	1.4900e+03
18	1.4909e+03
19	1.4774e+03
20	1.4921e+03
21	1.4889e+03
22	1.4901e+03
23	1.4951e+03
24	1.4901e+03
25	1.4950e+03

The second testing where the distance between 2 iteration should be 10cm apart.

No. of frame	Distance from sensor (in mm)
1	NaN
2	524.3571
3	623.5714
4	720.1429
5	814.7857
6	917.7857
7	1000
8	1.1181e+03
9	1.2144e+03
10	1.3119e+03
11	1.4135e+03

APPENDIX B

MATLAB code for background subtraction using (Zivkovic, 2004).

```
% This is a simple program to get the initial work of getting the vectors

clear;

position=[];                                %variable to store track
coordinates

nr_groups=6;

track=[];

reg=[];line_c=[];line_s=[];

testy=[];tester=[];testers=[];

display=2; %variable to control display onscreen

vector=zeros(10,1); % line to store calculated vector

angles=zeros(10,1); % line to store angle of vector

vec_m=[];ang_m=[];vector_m=[];

vecs=[]; % variable for determining the number of vectors

stored=[];lined=[];

%display=nr_groups+1;

sName='test.avi';                           % video file read

fInfo=aviinfo(sName);                       % get information from video
file

d=aviread(sName,1);                         % get video first frame

h=mexCvBSLib(d.cdata);                 %Initialize or first background image

initial_f=d.cdata;

mexCvBSLib(d.cdata,h,[0.01 14*14 1 0.1]);   %set parameters

figure(1);

[m,n]=size(initial_f(:,:,1));

for loop=1:nr_groups

mask1(:,:,loop)=zeros(m,n);

%  mask2(:,:,loop)=zeros(m,n);

end

blank=zeros(m,n);                       %blank template for empty frames
```

```
msk1=zeros(m,n);

msk2=zeros(m,n);

msk3=zeros(m,n);

msk4=zeros(m,n);

msk5=zeros(m,n);

msk6=zeros(m,n);

se=strel('diamond',4);

se2=strel('square',4);

bands=[0:1:255];

time=0;                                  %variable for running correlation

result1=[0:.01:1]';

for i=1:fInfo.NumFrames

d=aviread(sName,i);                     %read in from video file

snap=d.cdata;                           % snapshot of video file

imMask=mexCvBSLib(snap,h);              %to get masked image

snap1=rgb2ycbcr(snap);

%% section on getting a masked image

imMask=im2bw(imMask,graythresh(imMask));

imMask=imopen(imMask,se);

imMask=bwareaopen(imMask,100);          % remove speckles from image

%imMask(end-1:end,:)=0;                  %add line at bottom of image

%imMask(1:20,:)=0;                   %add line at bottom of image

imMask=imclearborder(imMask,4);%to get image not connected to left and right
edges

imMask=imfill(imMask,'holes');          %to fill holes in images

LMask=imMask;                           % logical image LMask and imMask

LMask1=cat(3,imMask,imMask,imMask);

ycbcr_masked=immultiply(LMask1,snap1);

labelled=bwlabel(LMask);                % label the area in a mask

stats=regionprops(labelled,'basic');    % getting basic information

identity=find([stats.Area]>3500);       % getting area of identity of area
larger than 1000 pixels

identity=identity';

%% section to display input inmage

subplot(1,display,1);

subplot(1,2,1);
```

```
imshow(snap);

drawnow;

hold on;

for blocks = 1:size(identity,1)

rectangle('Position',stats(blocks).BoundingBox,'EdgeColor',[1 0 0],
'LineWidth',1.0);%need to change to reg

drawnow;

end

drawnow;

hold off;

videoOut = insertObjectAnnotation(videoFrame,'rectangle',bbox,'Face');

subplot(1,display,2);

imshow(imMask);

drawnow;

end;

mexCvBSLib(h);                                      %Release memory
```

APPENDIX C

MATLAB code to obtain and encode data from URG-04LX-UG01.

To initiate function of laser range finder in MATLAB

```
% Setup Lidar

% % Configures Serial Communication and Updates Sensor Communication
to

% SCIP2.0 Protocol.

% % Checks Version Information and switches on Laser.

lidar=serial('COM3','baudrate',115200);

set(lidar,'Timeout',0.1);

set(lidar,'InputBufferSize',40000);

set(lidar,'Terminator','CR');

fopen(lidar);

pause(0.1);

fprintf(lidar,'SCIP2.0');

pause(0.1);

fscanf(lidar);

fprintf(lidar,'VV');

pause(0.1);

fscanf(lidar)

fprintf(lidar,'BM');

pause(0.1);

fscanf(lidar)
```

To initiate scan using the laser range finder

```
% Get a LIDAR Scan

% Returns a range vector of 682 elements after a Lidar Scan from min step

% to max step.

% Range Values correspond from -120 to +120 degrees.

function [rangescan]=LidarScan(lidar)

proceed=0;

while (proceed==0)

fprintf(lidar,'GD0044072500');

pause(0.01);

data=fscanf(lidar);

if numel(data)==2134

    proceed=1;end

end

i=find(data==data(13));

rangedata=data(i(3)+1:end-1);

for j=0:31

onlyrangedata((64*j)+1:(64*j)+64)=rangedata(1+(66*j):64+(66*j));

end

j=0;

for i=1:floor(numel(onlyrangedata)/3)

encodeddist(i,:)=[onlyrangedata((3*j)+1) onlyrangedata((3*j)+2) onlyrangedata((3*j)+3)];

j=j+1;

end

for k=1:size(encodeddist,1)

    rangescan(k)=decodeSCIP(encodeddist(k,:));end
```

```
end
```

To decode the SCIP 2.0 protocol data obtained from the laser range finder

```
% Function to decode range information transmitted using SCIP2.0 protocol.

% Works for only two and three bit encoding.

function rangeval=decodeSCIP(rangeenc)

% Check for 2 or 3 Character Encoding

if rangeenc(1)=='0' && rangeenc(2)=='0' && rangeenc(3)=='0'

    rangeval=0;

    return;end

if rangeenc(1)=='0'

    dig1=((rangeenc(2)-'!')+33);

    dig2=((rangeenc(3)-'!')+33);

    dig1sub=dig1-48;

    dig2sub=dig2-48;

    dig1bin=dec2bin(dig1sub,6);

    dig2bin=dec2bin(dig2sub,6);

    rangeval=bin2dec([dig1bin dig2bin]);

    return;

else

    dig1=((rangeenc(1)-'!')+33);

    dig2=((rangeenc(2)-'!')+33);

    dig3=((rangeenc(3)-'!')+33);

    dig1sub=dig1-48;

    dig2sub=dig2-48;

    dig3sub=dig3-48;

    dig1bin=dec2bin(dig1sub,6);

    dig2bin=dec2bin(dig2sub,6);

    dig3bin=dec2bin(dig3sub,6);
```

```
rangeval=bin2dec([dig1bin dig2bin dig3bin]);

return; end end
```