Vision Based Floor-Pattern Tracking Method Using Elliptical Geometry Approach in Robotic Routing

Wai Kit Wong, Wei Qing Ong, Wei Yi Ong, Kenny Ang, Thu Soe Min, Eng Kiong Wong, Yee Kit Chan

***Abstract*— This paper presents a new vision based routing method using elliptical geometry approach for floor-pattern tracking on robot navigation. Path tracking is important in vision based mobile robot navigation and it acts as the sense of seeing, recognizing and directing the mobile robot to its destination. Geometry for floor pattern is designed and vision based path tracking algorithm is formulated and implemented into a mobile robot framework for automated navigation purposes. The performance of the developed floor patterns and algorithm are tested on robotic path finding, path tracking and location aware intelligences. Experimental results shown that the proposed vision based floor pattern tracking algorithm supplies solutions to single path, multiple paths robotic routing under various floor background and light intensity conditions, which appears sufficient to be utilized in most practical mobile robot logistics cases.**

***Index Terms*— Control of Intelligent Systems, Image Processing, Pattern Recognition, Real Time Vision.**

I. INTRODUCTION

Mobile robot has significant applications potential in areas such as navigation, logistic/delivery task, surveillance and etc. Logistic mobile robots can assist human workers with tedious and tiring manual labor. Many researchers looking into warehouses and factories logistic mobile robot development, such as service robotics in the pharmaceutical industry [1], fuzzy logic path planners' mobile robot navigation in warehouses for materials handling [2], positioning and motion control for mobile robot using odometer system in warehouse [3] and etc. One of the main components in logistic mobile robots is the routing system. It is for path finding, path planning and location-aware. In general, there are three routing methods apply worldwide for logistic mobile robots: 1) Railway routing method, 2) line tracking sensory routing method and 3) Virtual map routing method. Below are some of the examples.

Railway routing method was once applied in Ohio State University Medical center since 1980s in their preliminary ATS (Automated Transportation System) [4]. It was a system consisted of carts and rails in its initial stage operate in a 1.6 million square feet medical center compound. Overhead rail system was built on the ceiling for carts to transport and required workers to follow. However, for some region without rail coverage, it required workers to push for a small distance to reach destination. No dedicated elevators for transporting carts, and needed to wait for busy elevators that share with patients, doctors, nurses and etc. In the year 2005, ATS emerge into the use of virtual map routing method with components consist of 46 robotic transporters, 9 dedicated elevators, 8 cart types, 2 automatic cart washers, some battery charging systems, a master control system and 55 touch screen operator terminals. The working operation is: the transporter recalculates its exact position several times per second based on the laser signal returned from wall mounted reflectors. This is how it stays on the guide-path with a virtual map. The guide-path is a path where the robot has been programmed to travel follows the virtual map. Impact of the ATS on operations: improved access to patient and public elevators, increased staff productive time, increased time for staff to spend with patients and improved work structure in some departments.

As in line tracking sensory routing method, Mobile-Robot-Enabled smart warehouses system [5] is one of the success stories in logistic mobile robot created by Kiva Systems (ranked number 6 (number 1 in logistics) in the Inc. 500 ranking of the fastest-growing private companies in the United States, 2011). It uses a new approach on order fulfillment, in which operators stand still while the products come to them. Pallets, cases and orders are stored on inventory pods that are picked up and moved by hundreds of mobile robotic drive units. As a result, any product can be moved to any operator, no need for the workers pushing carts around in warehouses and manually filling orders. Hence it is more cost–effective. Other samples of line tracking sensory routing method are RFID warehouse robot [6] and mini forklift robot [7]. They are logistic robots that can store and pick up object to/ from specified storages lot from/to a base using line follower (path finding/tracking) and RFID (to find the coordinates of targeted storage slot and goods identification). The difference between research works in [6] and [7] is: [6] only can pick up and store the item at the appropriate place and location in one dimension only whereas the robot in [7] can put stuff up to four levels cabinet.

As in virtual map routing method many research works had been done on developing localization techniques and creating better maps. Research work in [8] presented the dynamic Markov localization technique as a uniform approach to position estimation, which is able (1) to globally estimate the position of the robot, (2) to efficiently track its position whenever the robot's certainty is high, and (3) to detect and recover from localization failures. This supersede by researchers in [9] which introduced the concept of "virtual pheromones" for use in controlling autonomous mobile robots. Virtual pheromones are stored in a map of the

Wai Kit Wong, Wei Qing Ong, Wei Yi Ong, Kenny Ang, Thu Soe Min, Eng Kiong Wong, Yee Kit Chan, Faculty of Engineering and Technology, FET Building Multimedia University(MMU), Jln Ayer Keroh Lama, 75450 Melaka, Malaysia

environment maintained by a "pheromone server", which acts like a shared memory for all the agents. Autonomous agents communicate with the pheromone server by means of a radio communication link. The pheromone server can be embodied by a regular computer, a handheld device, or an embedded controller carried by a robot. It is emphasized that the whole system, comprising the pheromone server, the communication system, and the robotic agents act like a neural network. This technique leads to significant simplification of the task of controlling individual robots and robot swarms. Researchers in [10] further studied on the map fusion problem in the context of a multi-robot map building approach. In their studies, a set of several robots performs map building tasks without the notion of other robots' existence. Each robot builds its own local map using its observations and estimates its path independently. As a result, there will be a set of local maps that can be fused into a global one. This is the case when the map fusion takes importance. Particularly, they focus their experiments on landmark-based maps constructed using visual information and by means of a particle filter. When fusing two maps, they considered the uncertainty of the landmarks integrated by each different robot to its map.

Vision based floor-pattern tracking approach is chosen to study in compare to existing approaches (Railway approach, Sensory line tracking approach, Virtual Map approach), with justification:

· *Railway Approach*: need to build railway, railway might cause walk path uneven, might stumble people, railway might conceal dirt and hygiene, railway need maintenance and it required more sensors to be placed on railway track. However, in vision based floor-pattern tracking approach, it only required impressive direction pattern printed at ceramic tiles on floor as track.

· *Sensory Line Tracking Approach*: guided center approach, once the line on floor break or block, the line tracing sensors unable to trace back or continue the movement. However, in floor-pattern tracking, once the direction pattern in the captured image is missing/block, the algorithm will rotate camera to seek for nearby direction pattern to get back to its track.

· *Virtual Map approach*: required memorable floor plan by importing CAD drawings of the building to create on-board map. It also required plenty of RFID tags/reflectors to be placed around the facility to provide fixed reference points for the mobile robots to maintain location accuracy. However, in vision based floor- pattern tracking approach, complex map is replaced by simple destination graph in which the robot needs only to store home, destination and direction pattern. The only sensor to track path is the webcam. No RFID tags or some other sensors/reflectors required for path tracking.

This paper introduced a vision based floor pattern tracking method on elliptical geometry patterns in robotic routing. The following results are presented:

1. Designed of effective path guided symbols (home, destination and directed path) using elliptical geometry approach for floor pattern tracking robot routing.
2. Development of a novel algorithm for visual based floor pattern tracking and routing through direction, distance and angle calculation of the path guided symbols in 1. Auto-searching for path guided symbols are also embedded and proved working well as shown in details in Section VI.
3. Designed and fabricated robotic structural to carried out physical experiments to evaluate the developed path guided symbols, floor pattern tracking and routing algorithm in a real world scenario.

The outline of the paper is as follows. In section II, a review of representative works related to vision based robot routing methods is done. In section III, the proposed Vision Based Floor Pattern Tracking (VBFPT) Robot Routing System Model will be presented followed by section IV for an overview of the variant-size circular floor pattern design and selection. Next, in section V, the developed Visual Based Floor Pattern Tracking (VBFPT) Algorithm will be presented. The simulation as well as experimental results will be presented in section VI. The overall accuracy and performance evaluation are highlighted. Finally, the paper ends with a conclusion and future work discussion in section VII.

II. Literature Review and Justification

This section provides a review of representative works related to vision based robot routing methods. The particular focus is on reviewing techniques on vision wise routing, namely virtual map routing and symbolic routing.

A. Virtual Map Routing

Virtual Map Navigation is a method which uses vision tracking system to recognize the environment of the mobile robot and known landmark or local map obtained is compared with the global map that programmed in the memory system in earlier stage. If the local map is matched, the robot could localize its position and advanced to the destination by following the information stored in advanced. The global map stored could be a model pre-constructed according to the actual environment or it can be built by using sensory data. Fig 2.1 shows the process of virtual map routing. There are three recent related studies reviewed namely: Light fixtures on ceiling approach [11], Cyber Map [12] and SURF algorithm [13].

Light Fixtures on Ceiling Approach

Robot localization is achieved by using two type of filter, namely particle filter and Kalman filter. Both methods utilize the light fixtures and estimation of pose is done with information about robot kinematics. The entire working space is established in the form of occupation grid. Path planning is resolved by a search algorithm also known as Wavefront algorithm [11]. The captured image is processed and light fixtures position and orientation on reference frame is

detected based on the following steps: 1. Feature extraction, 2. Characteristics validation, and 3. Pose calculation. The particle filter is used when the starting point of robot is not known. The robot localizes itself by considering group of particles with circular boundary which has radius less than the minimum distance between two light fixtures. By doing so, different group of particles which belongs to different light fixture can be distinguished.

The advantages of this type of routing is light fixtures are commonly used in any office-type environment, which allow the implementation in indoor service robots. However, this type of routing having disadvantages of it can only be applied in indoor environment with lit up light fixtures. Also, the error may be generated in differentiating the real and corresponding light fixture because all fixtures are very similar and also the symmetry characteristics in most of the widely used light fixtures.

Cyber Map

This routing method mounted the laser range scanner on a mobile vehicle to construct a global map for the environment. Localization is done by scan matching method which projecting point and line features on horizontal plane and the vertical structure in the environment is detected. The algorithm will finds best pose in the GlobalMap by scanning around location in CurrentPose.

The advantages of this type of routing are it can be used in outdoor environment surrounded by heavy vegetation and it does not require specially designed floor pattern for loacalization and navigation. However, this type of routing having disadvantages of it requires additional work to reconstruct the global map when any changes are made to the environment. Also, it tracking will fail when scanning is carried out in incomplete section of map. Incomplete section of map means that minimal scan coverage is taken in that specific area.The authors of the accepted manuscripts will be given a copyright form and Registration form at final submission.

SURF Algorithm

This method consists of two parts, map creation and pose determination. SURF algorithm [13] is used to extract the feature points from the environment by using a camera placed on the robot. Global map is created by calculating the global coordinates of the extracted feature points and then combine feature points and global position into map. After global map is created, robot can be navigated by comparing captured image with the map. From matched points between the image and map, series of possible position and orientations of robot are obtained.

The advantage of this type of routing method is it does not require specially designed pattern. However, this type of routing method having disadvantages of it requires map creation. It also requires memory space to store global map and the recreation of global map is required when changes is made to the environment.

B. Symbolic Routing

Symbolic routing applies similar techniques used for virtual map. Both virtual map and symbolic recognition methods apply vision based routing which relies on visual devices instead of dead-reckoning and inertial sensors. However, storage of global map is not required for symbolic routing. Symbolic routing uses symbol on routing path to guide direction for moving forward, backward, turn left, right and stop. Pattern recognition aims to classify these symbol based on the pre-known data extracted from the symbol. Comparison between virtual map and global map is not available and this comparison is replaced by comparison of the captured image with prior knowledge of the symbol such as measurement of points in multidimensional space. Since there is no global map, pre-stored information such as distance and turning angle are not available. The measurements of this information need to be conducted once a symbol is recognized. The distance of the symbol can be known by calculating the symbol's pixels while angle is calculated by using trigonometry theorem. Fig. 2.2 shows the process of symbolic recognition and routing. There are two recent related studies reviewed, namely: Geometric structure robust measurement validation [14], and Visual Pattern Recognition by Moment Invariant [15].

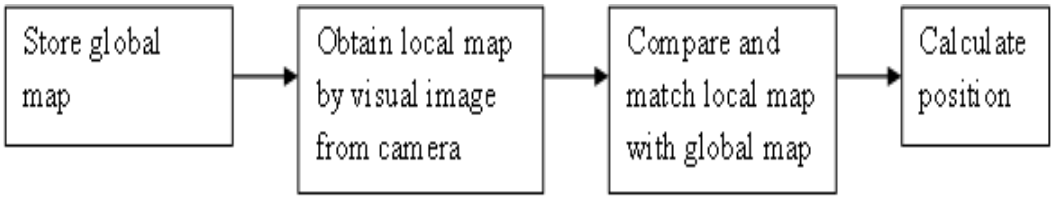


Fig 2.1: Block Diagram for Virtual Map Routing Process

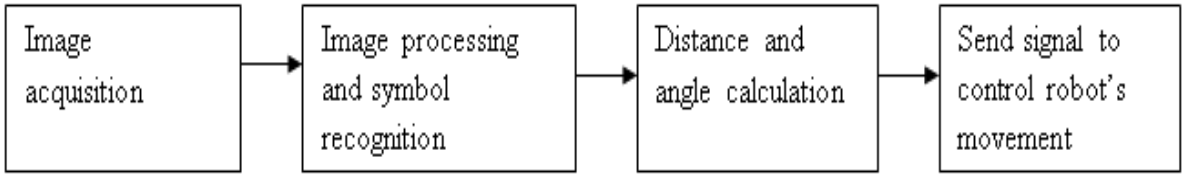


Fig. 2.2: Block diagram for Symbolic Recognition and Routing

Geometric Structure Robust Measurement Validation

Geometric structure is used to validate robust measurement in target tracking. A gating process is required to minimize number of potential measurements which can be used as distance measure to separate signals from noise (false alert) in target tracking. In the method propose by Yoon and Roberts [14], an algorithm which combines the typical gate selection scheme and geometric distance measurement is used to find target and its related measurements by using the Voronoi diagram. The first step in creation of Voronoi diagram is to partition the area by grouping every point in the area to the nearest site. The example of points grouping is as shown in the Fig. 2.3.

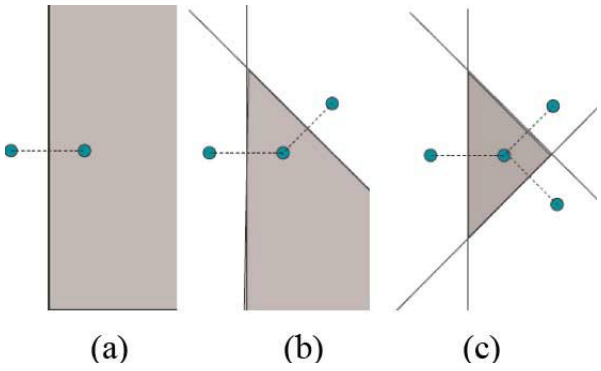


Fig. 2.3: Points Grouping in Voronoi Diagram

Step (a) Forming of perpendicular bisector between 2 points and divide the 2 planes. Plane closer to the right points is highlighted.
Step (b) Collection of points which are close to the right point in (a) is found and intersection of both is taken.
Step (c) The algorithm operates recursively for all points.

Visual Pattern Recognition by Moment Invariant

The work in [15] presented a theory of extracting 2-D moment invariants for planar alphabetical characters patterns and geometrical patterns in visual images. The complete pattern recognition system of moment invariants under translation, orthogonal transformations and similitude were derived in [15]. The results shown that recognition of alphabetical characters patterns and geometrical patterns independently of position, size and orientation can be well recognized.

As selection for guided symbol to be place in the surrounding environment for robot navigation, alphabetical characters patterns is not suitable due to lacking information of distance and turning angle. Therefore, geometrical patterns which can provide better information extraction in terms of distance and turning angle will be selected as guided symbol.

As for the Geometrical pattern, three geometric structures were under consideration. They were rectangle, triangle and circle/ellipse. The rectangle shaped pattern as guided symbol was designed by using combination of two rectangles, one with bigger size and one with smaller size, as shown in Fig. 2.4. By using such combination as path guided symbol, distance and turning angle information can be calculated by obtaining the centroids of both rectangles. The direction can also be indicated by recognizing the orientation of both rectangular. As example indicates in Fig. 2.4, the orientation with smaller rectangle lies before the larger rectangle indicates a forward direction. The rectangle shaped pattern as guided symbol has advantage of simple design and the pattern can be easily recognized. However, the disadvantages of such pattern are the shape might vary at different point of view. Since the shape is not constant, this designed pattern cannot be well recognized by searching robot within its capture range.

Fig. 2.4: Rectangular shaped pattern

The triangle shaped pattern as guided symbol was designed by using a non-equilateral triangle as shown in Fig. 2.5. The direction, distance and turning angle information can be easily calculated. The recognition process is stated as below:

i. Calculate a, b and c.
ii. Compare the length of a, b and c.
iii. Determine the vertice that is connecting to two longest edges as T.
iv. Calculate distance of the triangle by using the centroid.
v. Calculate the angle to new direction by using the centroid and vertice T.

The triangular shaped pattern as guided symbol has advantages of simple design and the pattern can be easily recognized. It also does not require the combination of two structures. However, the disadvantages of such pattern are the shape might vary at different point of view. Since the shape is not constant, this designed pattern cannot be well recognized by searching robot within its capture range.

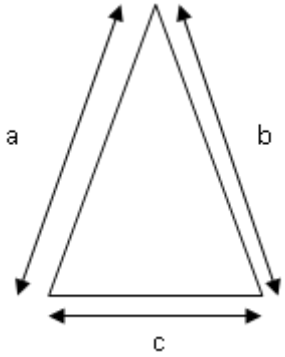


Fig. 2.5: Triangular shaped pattern

The circle shaped pattern as guided symbol was designed by using combination of two circles, one with bigger size and one with smaller size, as shown in Fig. 2.6. This design applies the same technique as rectangle shaped pattern to calculate direction, distance and turning angle. The observed

advantages are simple design and the pattern can be easily recognized. The most important is it possesses same shape from all points of view, therefore its centroid and size normalization can be easily exploited. Circle shape symbol also can achieve orientation independence, unlike square, rectangle, triangle or others shape. The detail discussion for circular shaped pattern and the method of extracting its direction, distance and turning angle information is covered in Section IV.

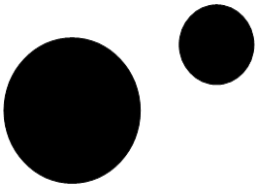


Fig. 2.6: Circle shape pattern

As for the location selection for placing path guided symbol, it can be either on ceiling, on wall or on floor. Placing path guided symbol on ceiling has the advantage of path guided symbol not easily block by moving surrounding obstacles. However, it has the disadvantages that for minimizing blocking view of path searching and tracking, imaging tool has to be mounting on top of the robot, inconvenient design for delivery robot with top-side as delivery platform. Imaging tool facing upward easily collect dust on optical lenses which required frequent cleaning. If the ceiling is too high, path guided symbol placed on ceiling might not clearly view, it required bigger size of path guided symbols.

Placing path guided symbols on wall has the advantage of imaging tool can be freely mounted at any height of the robot parallel to the horizontal viewing plane. However, it has the disadvantages that the path guided symbols easily block by moving obstacles. It also required frequent replacing of path guided symbols on track since adding furniture, framed paintings, decoration or repainting wall might blocks certain path guided symbols on initial track. If the site is too broad, distance to reach wall, path guided symbols placed on wall might not clearly view, it required bigger size of symbols.

Placing path guided symbols on floor has the advantages of no issue on the height and the broad of the operating site. Since path guided symbols are placing on ground level, imaging tool on the robot can be located close to ground level with better viewing angle and distance. Placing path guided symbol on floor following human walking track, imitating human viewing capability to follow track on navigation. The imaging tool tends to facing downward, uneasy to collect dust on lenses. With such advantages listed above, floor pattern tracking is selected in compare to ceiling and wall pattern tracking for studying robotic routing in this paper.

III. Vision Based Floor Pattern Tracking (VBFPT) Robot Routing System Model

This section illustrates the robotic framework designed to test run the developed vision based floor pattern tracking (VBFPT) robot routing method. Fig. 3.1 shows the proposed framework and the VBFPT robot routing system, including the main systems, the subsystems and their supporting hardware and software.

A. Robotic Framework

This subsection describes the process flow of the VBFPT robot, hardware and software selection for constructing the robot and the completed hardware model.

Process Flow

Fig. 3.2 describes the process flow designed for the VBFPT robot routing system. Firstly, the interfacing tool will control image capturement from imaging tool (Path A and B). After that, the captured image will pass to decision making tool to perform image processing for extracting path direction, home or destination information (Path C). After the information is extracted, decision making tool will return the subsequence commands back to interfacing tool (Path D). The interfacing tool will then queues up the commands in a list and send them one-by-one to Robot Control System through wireless communication tool (Path E, F, G). A simple flow control algorithm is used between interfacing tool and Revision Control System (RCS) embedded in robot hardware control system to ensure no command is lost or overwritten throughout this process. RCS is a software implementation of revision control that automates the storing, retrieval, logging, identification, and merging of their revisions. When the last command is completed, the RCS will let the Interfacing tool know and the interfacing tool will repeat the whole process (Path H, I, J).

Hardware and Software Selection

By referring to the designed process flow in Fig. 3.2, the components (Hardware and software) required to build VBFPT robot routing system includes imaging tool, wireless communication tool, image processing software and interfacing software.

Imaging Tool

Wireless webcam is preferred over conventional wired camera due to its portability and energy saving. Within the category of wireless webcam, IP camera is favourable due to it uses Wi-Fi connection, which has superior data rate compared to Bluetooth webcam. In this paper, DCS-930L was chosen due to two reasons:

- Cheaper in hardware price compare to some other IP cameras with fine captured image resolution (640 x 480).
- Offers ad-hoc connection, which removes the need of a router between computer and camera.

Project	Vision-based Floor Pattern Robot Routing System				
Systems	Vision System	Intel System		Robot Routing System	
Subsystems	Image Capturing	Decision Making	Interfacing	Communication	Motor Control
Supporting hardware and software	IP Webcam (DCS-930L)	PC/Laptop		Wireless Communication (ZigBee)	Main Control Board and motors (G15)
		Image Processing Software (Scilab)	Graphical User Interface (Visual Basic)		

Fig. 3.1: Framework of Robot System

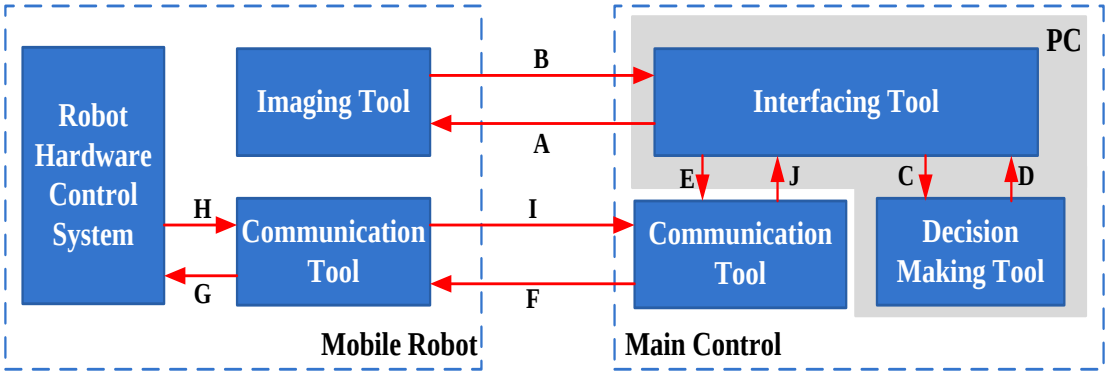


Fig. 3.2: The Designed Block Diagram for VBFPT Process Flow

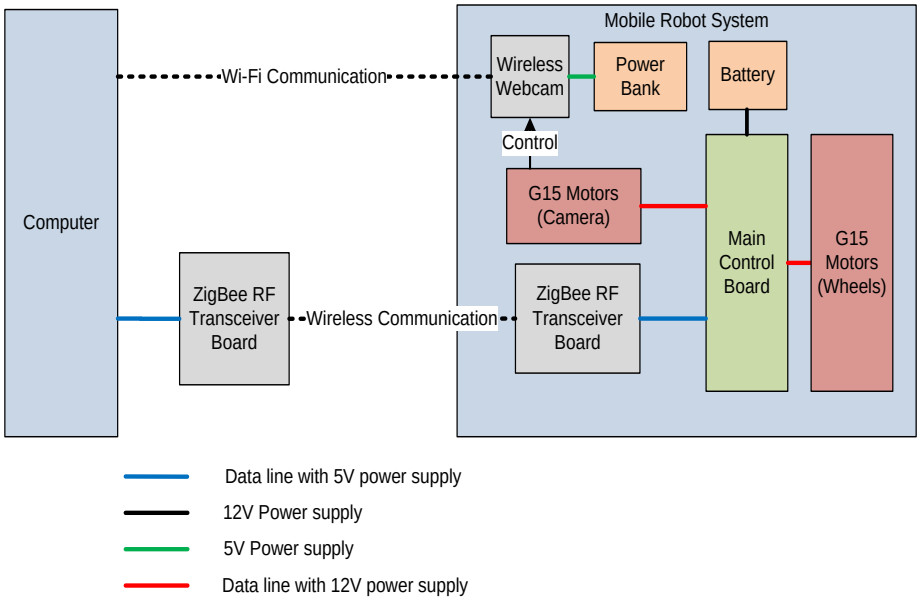


Fig. 3.3: Completed Hardware model

Wireless Communication Tool

Five choices were shortlisted for wireless communication tool selection. The cheapest choice is the infrared transmitter and receiver set. However, it is not a good choice as the communication range will be very short [17] and it operates line-of-sight coverage only [18]. The second choice is the Radio Frequency (RF) modules. They have good speed and distance coverage, and not limited to line of sight coverage only. It has very fundamental design which consists of a transmitter and receiver. Nonetheless, its reliability depends on the system designed by the designer, which means designer has to do more work to ensure the consistency of the communication link [19].

Bluetooth is a popular choice and it is listed as the third option in this paper. It supports very good bit rates but its communication range is short, which usually expand up to 10 meters only [20]. The fourth choice is the Wi-Fi modules. It has the highest speed of all with very good coverage area, but the price is very expensive.

The fifth choice is the ZigBee modules. ZigBee is designed for small, low-power uses such as robots. It has long distance coverage with very low power consumption, and the

built-in protocols allowing it to form reliable communication networks with other ZigBee [21]. For these reason ZigBee was chosen as wireless communication modules in this paper.

Image Processing Software

Image processing software is the focus point in this robot system as it is the subsystem that will drive the robot by processing the image capture and then locates and decides the next destination to go. As Scilab is freeware, for research purpose, Scilab is chosen as the image processing software at current stage. It can be further implement on Matlab or Mathematica in future.

Interfacing software

Interfacing software acts as the intermediate unit between image processing software and wireless communication device. It allows the operators to control the robot manually or initiate the robot to operate by itself through the assistance of image processing software.

The choice of interface software largely depends on the preferences of the programmer. In this paper, Visual Basic by Microsoft is chosen since it offers a quick solution to program a reliable and multifunctional GUI. Other choices such as Java program, C# or C++ can be used as well; with C++ has the reputation of capable to produce the program of the higher speed but takes longer time to write a complete program. The decision to choose Visual Basic over the other languages is further motivated by the fact that Visual Basic has the built-in function to access authorized IP camera such as DCS-930L.

Completed Hardware Model

Fig. 3.3 pictures the completed hardware model for the VBFPT robot routing system based on the designed block diagram. The computer performs most of the processing operation for the robot routing system, including the image processing and interfacing function. It also developed with GUI function for human operator to operate the robot manually. ZigBee RF Transceiver Board contains the ZigBee modules mounted on the ZigBee board. The two ZigBee modules communicate with each other through low power RF signals. Main Control Board controls the actions of the robot such as rotating the wireless webcam or travel robot around. It receive commands from computer via ZigBee and controls the motors based on the commands. G15 motors are selected due to the reason that they are high precision servo motors which can receive commands from the main board in the form of half-duplex serial data. One G15 motor applies to control the rotation of the wireless webcam and another two G15 motors are applied to control the wheels' movement. Power bank is the power supply prepared solely for power up the wireless webcam. It can directly supply 5V and 1.2A of power to the wireless webcam without voltage conversion. This might overcome power surge damaging webcam problem if sharing power supplies were practiced. A separated 12V battery is used to supplies power to the rest of the robot component.

B. Robot Structural Design and fabrication

A robot structure has to be designed and fabricated in order to run-test the effectiveness of the VBFPT method. The design requirements of the robot are as listed below:

a) A machine capable to move forward, backward and rotate left or right at any angle.
b) A platform for placing all the required components.
c) A holder with adjustable height to hold the imaging tool, and capable to rotate the imaging tool at specific angle and to allow the vision of the camera free from obstacles.
d) Actuators capable to support the weight of the whole structure.
e) Imaging tool capable to stays in the pivot of rotation while the whole structure is rotating.

Several autonomous visual navigation mobile robots designed by past researchers were studied in order to improve the robot design in this paper. The first one is the four wheeled robot in Fig. 3.4 [22], where it designed as the rear wheels are driven and front wheels perform the steering. This type of four-wheeled robot design is not suitable for the proposed navigation robot in this paper as it is unable to perform rotation tasks. The second approach is a patrol service robot [23] as shown in Fig. 3.5, where the mobility of the robot is controlled using 3 driven omni-directional wheels. The 3 omni-direction wheel design in this robot allows rotation and movement at all directions. However, the design is costly and the requirement of the robot in this paper to be tested only requires few simple movements for accuracy in positioning. Hence, it is not adopted here. The third approach is a Pioneer 3DX as shown in Fig. 3.6 [24]. This robot consists of 2 driven wheels and a castor wheel at the rear. It serves to be a better design meeting the requirement of the robot structure to be used for VBFPT. Therefore, the proposed robot in this paper is modified with two differentially driven wheels at the middle of a square base together, with two ball wheels in front and behind the base as shown in Fig. 3.7. Fig. 3.8 and Fig. 3.9 shows the orthographic view and robot movement view for the proposed robot design respectively.

Unlike the Pioneer 3DX, the proposed robot adopted two supporting wheels to increase the size of the base for the placement of the components. The proposed robot will be able to move forward/backward when both wheels move at the same direction and same speed. To perform rotation, both wheels will turn at opposite direction and at the same speed. Aluminum material is selected to build the robot structural due to its high strength and light weighted characteristics [25]. Square hollow aluminum bars were used to build a square base and a tower at the middle of the base to hold the wireless webcam. The wireless webcam is elevated and placed at the middle of the base so that it will turn on the pivot of rotation and its view will not obstructed by the components. The wireless webcam is designed to be able to rotate up and down manually and also left and right by a servo motor. The same

types of servo motors were also selected to drive the moving robot's wheels.

Fig. 3.4: Four-wheeled robot

Fig. 3.5: Patrol Service robot

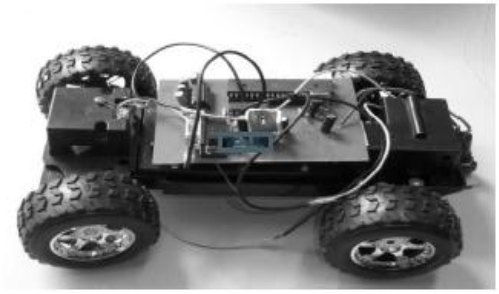


Fig. 3.6: Pioneer 3DX

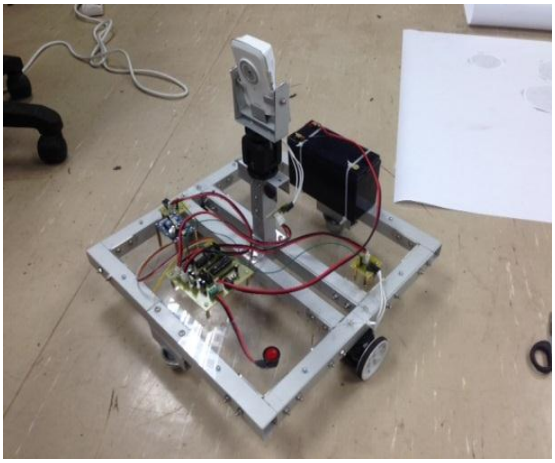


Fig. 3.7: The Proposed Robot Design

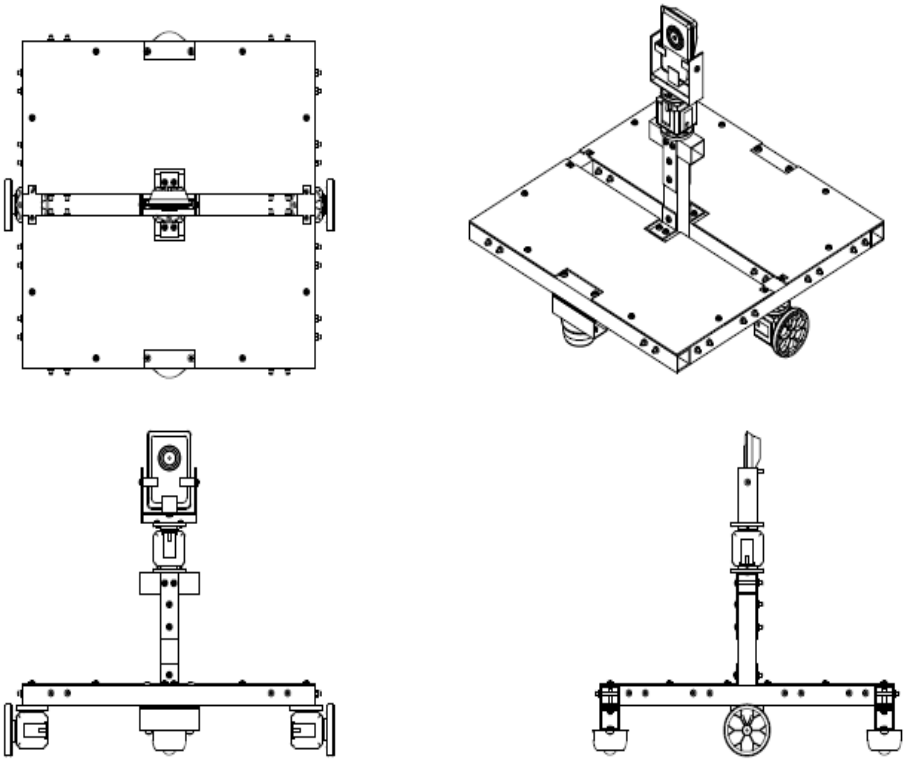


Fig. 3.8: Orthographic view of robot

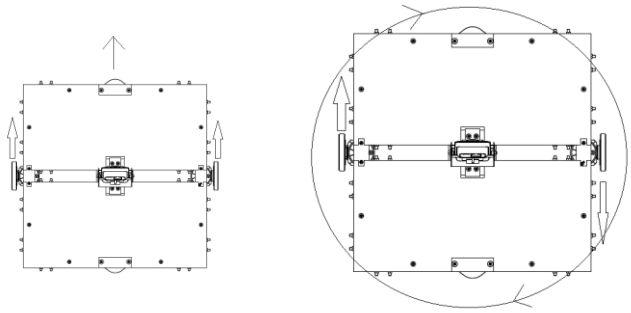


Fig. 3.9: Robot moving forward and rotate

IV. VARIANT-SIZE CIRCULAR FLOOR PATTERN DESIGN AND SELECTION

In this section, floor pattern design for path navigation, home and destination are discussed. The placement of floor pattern on-site is also determined and formulas are developed to establish the relationship between certain factors that determines the placement of floor pattern on-site.

A. Floor pattern design for path navigation

The design of floor pattern is crucial for an accurate detection and calculation of distance and direction for mobile robot navigation. The floor pattern design as shown in Fig. 4.1 is in black colour. This grayscale design forms a big contrast that allows the floor pattern to be easily detected. This floor pattern is designed to navigate the mobile robot along the travelling path to the desired destination. Circular shape is used instead of other geometric shape because it is invariance under geometrical transformations. The floor pattern is designed such that it is constant in shape from different viewing angle. Information that can be abstract from this floor pattern design includes direction, distance and angle. The floor pattern is designed with combination of two different size of circles to indicate forward direction and preventing the mobile robot in moving to opposite direction. When the mobile robot is moving from home position to destination, the designed floor pattern will be seen as smaller circle lying below the larger circle. In contrast, the floor pattern will be seen as reversed when the robot moves back from destination to home.

In practical environment, the circles of floor pattern are detected as ellipse instead of circular shaped. As of the lateral view from imaging tool mounted on mobile robot, the circles are flatted and manifest themselves in ellipses. Therefore, ellipse will be used to describe the floor pattern design in the following passage. Fig. 4.1 shows the top view of the designed floor pattern and Fig. 4.2 show the lateral view of the designed floor pattern. Calculation of distance and angle of floor pattern from mobile robot's position is based on the information extracted from centroid of both ellipses. The method used to detect object in images is Thresholding and Blob Analysis as described in [15]. The objects are detected in the image by searching connected areas of true pixels. The detailed algorithm for detection of connected areas is described in [16]. By using the blob analysis algorithm which is implemented in Scilab, the centroid of detected object is abstracted.

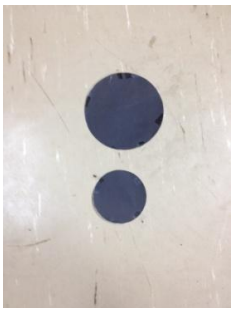


Fig. 4.1: Top view of floor pattern design

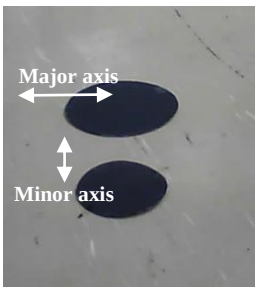


Fig. 4.2: Lateral view of floor pattern design

Fig. 4.3 shows the real image captured by the wireless webcam (DCS-930L) mounted on VBFPT mobile robot as constructed in Section III. The image capture by the wireless webcam has a resolution of 640 x 480, where 640 is the number of pixel in horizontal x-axis and 480 is the number of pixel in vertical y-axis. The designed floor pattern which indicates forward direction has the following characteristics,

1. Consists of two ellipses with different sizes,
2. Both ellipse should not lies too far from each other ($<D_{max}$),
3. When robot moving from home to destination, small ellipse of designed floor pattern lies nearer to robot,
4. When robot moving from destination to home, large ellipse of designed floor pattern lies nearer to robot.

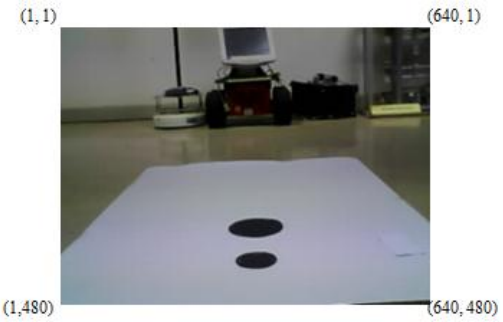


Fig. 4.3: Two dimensional image captured by wireless webcam.

The colour of designed floor pattern can be manipulated according to the demand of application. When the number of destination increases, different colour of floor pattern is designed to represents each destination and their respective paths. In this paper, three colours are selected to represent each destination. Red, green and blue colours are selected because they are easy to be differentiated from each other based on their representative values in RGB and HSV colour models. In RGB colour models, colour are represented by mixture of red, green and blue components. The selected colours are easy to be differentiated because they are represented by single component in RGB colour model. Since Hue component represents colour type in HSV colour model, colours which have larger difference in Hue range are more preferable. Red, green and blue colours are more preferable because their ranges in Hue component are further away from each other. Fig. 4.4 shows the design of path routing floor pattern in red, green and blue.

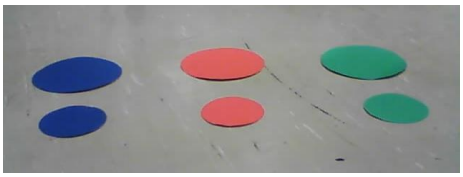


Fig: 4.4: Multi Path routing floor pattern design

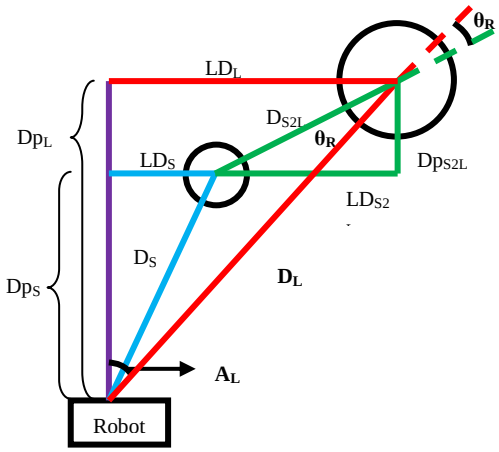


Fig. 4.5: Distance and angle between floor pattern and robot (Top view).

The centroid obtained from blob analysis is applied into trigonometric equation that yields distance and angles calculation. The relationship between both circle and robot's position is shown in Fig. 4.5. The objective of distance and angle calculation is to obtain the distance from robot to large ellipse, D_L, angle from robot to large ellipse, A_L and robot rotation angle to face next path, θ_R as shown in Fig. 4.5. The robot will moves rotate and move to the large ellipse based on the calculated D_L and A_L. When it reaches the large ellipse, the robot will turn towards the new direction based on θ_R and stay there, continue seeking for the next pattern to move on.

Distance for both ellipses are calculated by using Pythagoras theorem as shown in equation (1) and (2).

$$D_L = \sqrt{Dp_L^2 + LD_L^2} \quad (1)$$

$$D_s = \sqrt{Dp_s^2 + LD_s^2} \quad (2)$$

where,

D_L = Distance of large ellipse from robot,
D_S = Distance of small ellipse from robot,
Dp_L = Depth of large ellipse from robot,
Dp_S = Depth of large ellipse from robot,
LD_L = Lateral distance of large ellipse from robot,
LD_S = Lateral Distance of large ellipse from robot,

Angle to the larger circle is

$$A_L = tan^{-1}\left(\frac{LD_L}{Dp_L}\right) \quad (3)$$

Distance between large and small ellipses is

$$D_{S2L} = \sqrt{Dp_{S2L}^2 + LD_{S2L}^2} \quad (4)$$

The robot rotation angle to face next path is

$$D_S^2 = D_{S2L}^2 + D_L^2 - 2(D_{S2L})(D_L)cos\theta_R$$

$$\theta_R = cos^{-1}\frac{D_{S2L}^2 + D_L^2 - D_S^2}{2(D_{S2L})(D_L)} \quad (5)$$

The method to calculate depth and lateral distance will be discussed in subsection C.

B. Floor Pattern design for home and destination

The characteristics of home and destination floor pattern design are similar with floor pattern designed for path navigation which includes the following characteristics:

- Invariance under geometrical transformation,
- Provide information such as direction and distance,
- Different from floor pattern designed for path navigation.

Different sizes of circular pattern apply in floor pattern design for both home and destination to ensure these patterns are detectable from different viewing angle and indicate forward

direction to prevent the reverse movement of mobile robot. The number of circles in the home and destination floor pattern design is increase to differentiate them from the path navigation floor pattern. Fig. 4.6 shows the design for home position pattern and Fig. 4.7 shows the design for destination position pattern. Floor pattern which represents home position consists of a large ellipse lies between two equally small ellipses. In contrast, the destination position pattern has a small ellipse lying in between two equally large ellipses. When a home position or destination position floor pattern is recognized, mobile robot will rotates and moves toward the centered ellipse by using equation (1) and (3) in subsection A.

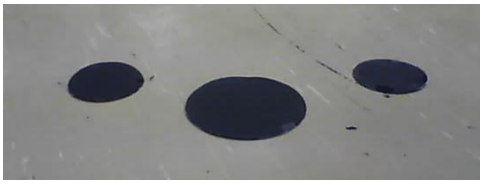


Fig. 4.6: Home position floor pattern design

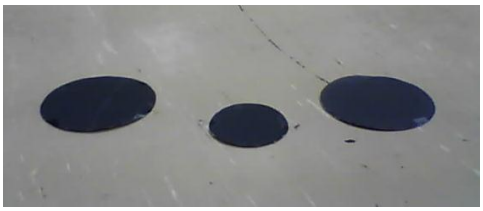


Fig. 4.7: Destination position floor pattern design

C. Pattern Placement Determination

The formula used to determine the distance between the base of the robot and point of interest, and the assisting formulas are to be derived in this subsection. The basic trigonometry layout to determine the relationship between the height of camera from ground, the distance of subject from camera and the tilting angle of the camera are to be designed.

Fig. 4.8 shows the yz-plane view (side view) of the trigonometry layout for the camera mounting on VBFPT robot and the pattern placed on the floor. Fig. 4.9 shows the xy-plane view (top view) of the layout.

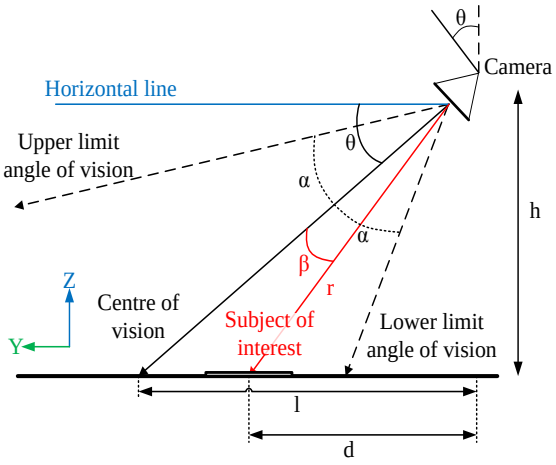


Fig. 4.8: Side view of trigonometry layout

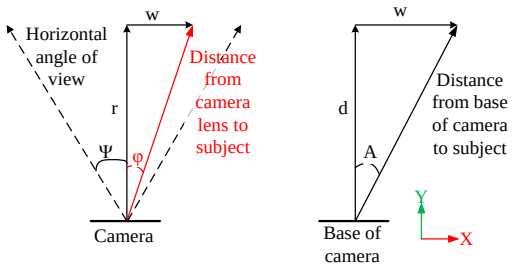


Fig. 4.9: Top view of the layout

To find the vertical angle from the centre of vision to subject of interest (floor pattern), β and the horizontal angle of subject from the centre line of wireless webcam's view φ, the captured image is divided from the wireless webcam into two segments for vertical lines and two segments for horizontal lines as depicted in Fig. 4.10. The right portions of the horizontal lines are positive ratio values, while the left portions are negative ratio values. For vertical lines, the bottom portions are positive ratio values while top portions are negative ratio values. The ratio is the position of the subject with respect to the centre line.

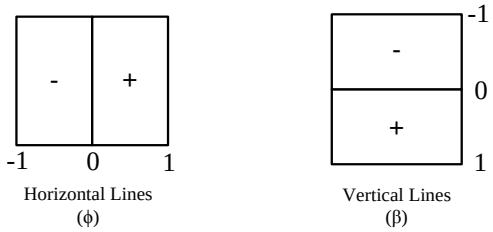


Fig. 4.10: Horizontal segments and vertical segments for captured image

for captured image

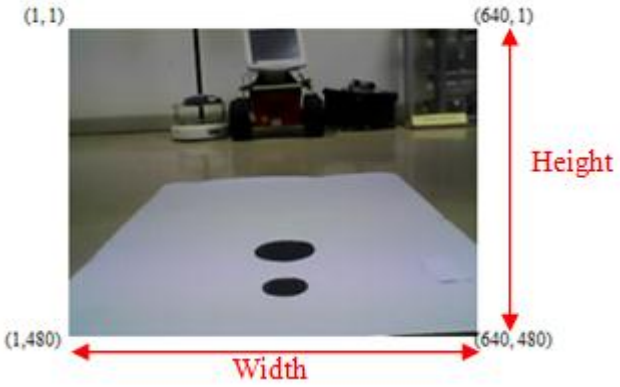


Fig. 4.11: Height and width indication of a captured image

The ratios are calculated as below:

$$\text{ratio of vertical line, } r_1 = \frac{(2 \times Pixel\ Coordinate\ y) - height\ of\ image}{height\ of\ image}$$

$$\text{ratio of horizontal line, } r_2 = \frac{(2 \times Pixel\ Coordinate\ x) - width\ of\ image}{width\ of\ image}$$

Some basic trigonometry formulas are also formed from the layouts:

The length from the base of camera to the centre of its vision,

$$l = \frac{h}{\tan\theta} \quad (6)$$

The distance from subject to the base of the camera,

$$d = \frac{h}{\tan(\theta+\beta)} \quad (7)$$

The vertical angle from the centre of vision to subject,

$$\beta = r_1 \times \alpha \quad (8)$$

The horizontal distance of subject from the centre line of camera's view,

$$w = r\tan\varphi \quad (9)$$

The horizontal angle of subject from the centre line of camera's view,

$$\varphi = r_2 \times \Psi$$

The distance from the lens to subject,

$$r = \sqrt{d^2 + h^2}$$

Where:

r_1 is the ratio of vertical lines,
r_2 is the ration of horizontal lines,
h is the height of the lens from ground,
θ is the camera's tilt angle,
α is the vertical viewing angle of camera,
Ψ is the vertical viewing angle of camera,
A is the horizontal angle of subject from the centre vertical line from camera's base.

The distance of target area from the robot ,

$$D = \sqrt{d^2 + w^2} = \sqrt{(d)^2 + \left(\sqrt{d^2+h^2}\tan(r_2\Psi)\right)^2} \quad (10)$$

and the distance from subject to the base of the camera as in eq(7) and eq(8):

$$d = \frac{h}{\tan(\theta+(r_1\alpha))} \quad (11)$$

Based on Pythagoras Theorem, derived from equation (7) and (9), the angle of the target from the centre vertical line of the robot is:

$$A = \tan^{-1}\frac{w}{d} \quad (12)$$

Equation (10) consists of six variables, two of them (r_1 and r_2) changes throughout the robot's operation, while the other four (θ, α, Ψ and h) are supposedly fixed all along. These four constant variables are determined before the robot's operation starts and should remain unchanged during the operation.

The height of the lens from ground, h is decided by the human operator where it is the height from the lens to ground. Three remaining parameters: θ, α and Ψ are needed to determine. Some imaging tool manufacturers will provide the viewing angle parameters in their product datasheets, and they can be used to determine θ, α and Ψ without going through their formulation process. However, if these information is unavailable, the following methods can be performed to obtain θ, α and Ψ respectively.

Method to determine vertical angle of vision, α

To obtain the imaging tool's angle of vision, α at vertical axis, two reference points on the ground are needed. One reference point will be exactly focus on the centre of the imaging tool's viewing frame, and another reference point is k distance away from the first reference point, along the vertical axis as shown in Fig. 4.12 and Fig. 4.13. The distance of second reference point from the base of imaging tool will be

$$d_2 = \frac{h}{\tan(\theta+\beta)}$$

Using trigonometry properties

$$\tan(A+B) = \frac{\tan A + \tan B}{1 - \tan A \tan B}$$

$$d_2 = \frac{hl - h^2\tan\beta}{h + l\tan\beta} \quad (13)$$

Let $d_2 - d_1 = k$, where d_1 is the distance from the first reference point to the base of imaging tool and k is the distance between the first and the second reference points. l the distance between the first reference point and base of imaging tool. Since the first reference point is at the centre of the imaging tool's view, d_1 is equal to l.

$$d_2 - d_1 = \frac{hl - h^2\tan\beta}{h + l\tan\beta} - = k \ ,$$

$$where\ d_1 = l$$

Rearranging the expression and from equation (8), the vertical angle of vision

$$\alpha = \left|\frac{\tan^{-1}\left(\frac{kh}{-h^2+l^2-kl}\right)}{r_1}\right| \quad (14)$$

where k is the distance between the first and the second reference points, h is the height of imaging tool position measured from the ground level, l is the distance between the first reference point and base of imaging tool, and r_1 is the ratio of the second reference point's pixel coordinate y in the capture image to the height of the image. Independent variables k, h and l can be preset by human operator while r_1 extracted from the captured image.

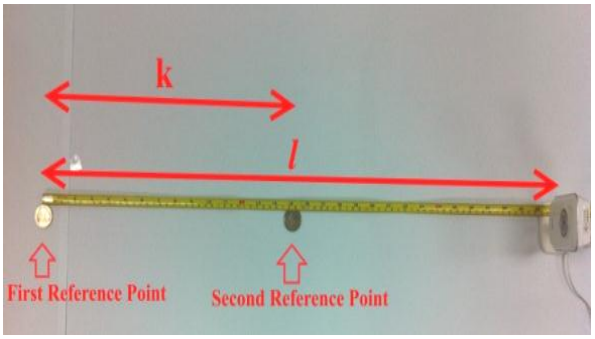


Fig. 4.12: Top view of setup to find α

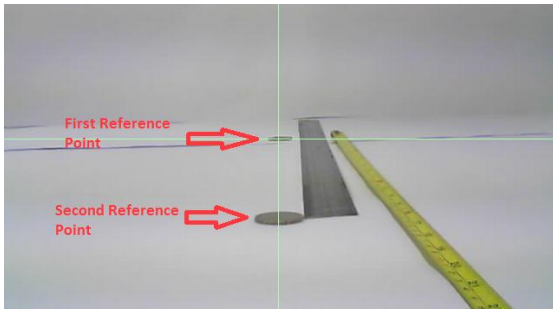


Fig. 4.13: Image view of setup to find α

Method to determine horizontal angle of vision, Ψ

Similar to the method of finding α, two reference points are needed to determine horizontal angle of vision, *Ψ*. The first reference point will be exactly focus on the imaging tool viewing frame whereas the second point is place at *k* distance along the horizontal axis, as shown in Fig. 4.14 and Fig. 4.15.

From equation (9),

$$\Psi = \frac{\tan^{-1}\frac{w}{\sqrt{d^2+h^2}}}{r_2}$$

$$let\ w = k\ and\ d = l$$

Since the first point is at the centre, *w* is equal to *k* and *d* is equal to *l*. hence, the horizontal angle of vision

$$\Psi = \left| \frac{\tan^{-1}\frac{k}{\sqrt{l^2+h^2}}}{r_2} \right| \qquad (15)$$

where *k*, *l* and *h* are the same parameters applied in the method of finding α. r_2 is the ratio of the second reference point's pixel coordinate x in the capture image to the width of the image.

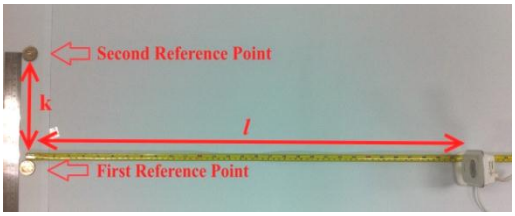


Fig. 4.14: Top view of setup to find Ψ

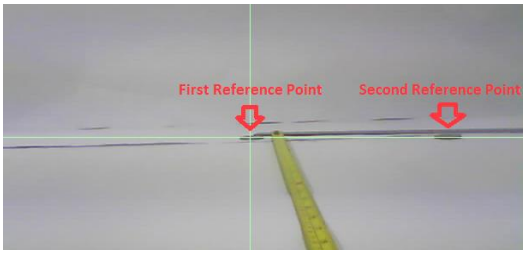


Fig. 4.15: Image view of setup to find Ψ

Method to determine imaging tool's tilt angle, θ

To obtain the imaging tool's tilt angle, *θ*, the imaging tool is first tilted to the outlined angle before applying the following formula to get *θ*. The distance between a pre-set reference point and the base of imaging tool is

$$d = \frac{h}{\tan(\theta+\beta)}$$

$$using\ \tan(A+B) = \frac{\tan A + \tan B}{1 - \tan A \tan B}$$

$$\theta = \tan^{-1}\left(\frac{h - d\tan\beta}{d + h\tan\beta}\right) \qquad (16)$$

Single reference point is needed along the vertical axis, where *h* is the height of camera from ground and $\beta = r_1 \times \alpha$. *θ* should be fixed from this section onwards as slight variation in the angle will cause the distance results to be inaccurate.

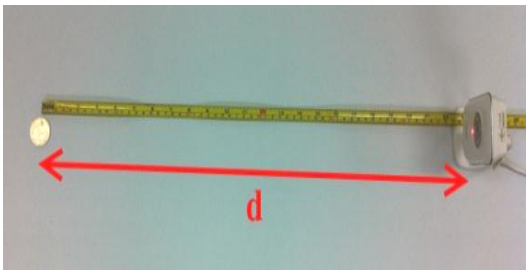


Fig. 4.16: Top view of setup to find θ

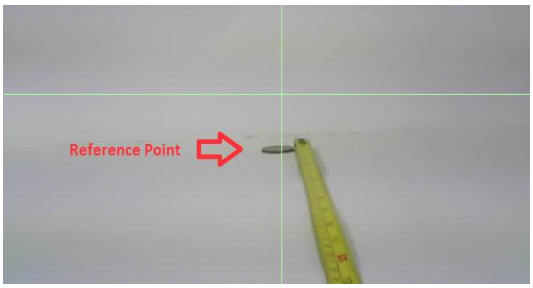


Fig. 4.17: Image view of setup to find θ

Summary steps to determine α, Ψ and θ

1. Set values for k, h and *l*. Setup the equipments following the values.
2. Find α and Ψ.
3. Set value for d. Set and fix the values for h and θ. Setup the equipments following the values.
4. Find θ.

Any changes in *h* and *θ* after this process will result in inaccurate *D* (the distance from robot to point of interest) and *A* (the angle from robot to point of interest) values; therefore proper setting must be taken in order to avoid altering these two parameters.

Determining D and A

After obtaining *θ*, *α*, *Ψ* and *h*, the distance from robot to targeted area, *D* and its angle, *A* can now calculated based on r_1 and r_2 using equation (10) and (11).

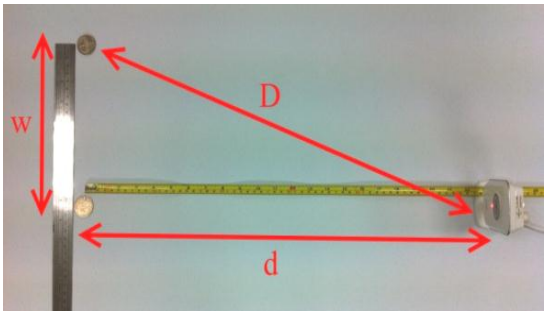


Fig. 4.18: Top view of setup to find point of interest

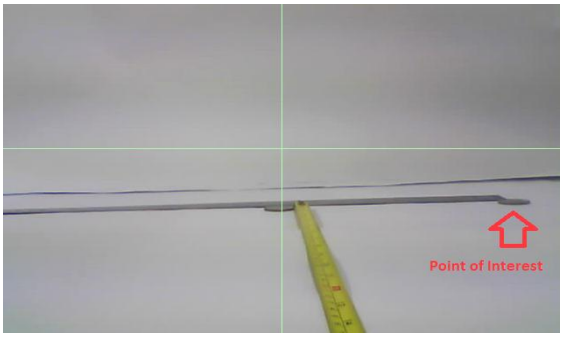


Fig. 4.19: Imaging tool viewing frame, setup to find point of interest

V. Visual Based Floor Pattern Tracking (VBFPT) Algorithm

In this section, the VBFPT algorithm that is implemented in the fabricated mobile robot in Section III is elaborated in the following steps.

Step 1: *Capture image*: An on-site image is captured by using the recommended wireless imaging tool.

Step 2: *Ellipse recognition*:

(i) The RGB values of captured image are transformed to HSV colour model.

(ii) Denote S_{Red}, S_{Green} and S_{Blue} as upper filtering limit for Saturation component of red, green and blue colours.

(iii) Filter objects from background by using S_{Red}, S_{Green} and S_{Blue}.

(iv) Filter non-target colour by using hue component of HSV colour model.

(v) Set background and non-target colour to 0 while the remaining to 1.

(vi) Set S_{SE} as structuring element of closing process and apply closing technique with S_{SE} =1.

(vii) Denote a threshold value, T, which indicates the minimum object size and eliminate those which are less than T. Fig. 5.1 shows the resultant image after Step 2(vii).

(viii) Obtain the two dimensional centroid's coordinate and boundaries for each detected object in the image. The boundaries include the coordinate of top, bottom, left and right pixels of each object.

(ix) Apply the characteristic of ellipse [16],

$$m_1 + n_1 = m_2 + n_2 \qquad (17)$$

as shown in Fig. 5.2 to each blob to identify ellipse shaped object, where m_1 and n_1 are the distance from M to P_1 and P_1 to N respectively and m_2 and n_2 are the distance from M to P_2 and P_2 to N respectively. M and N are the foci of ellipse. Blob that did not satisfy equation (17) is eliminated.

(x) In the case where no ellipse or no object detected, proceed to Step 4.


Fig. 5.1: Resultant Image after Step 2(vii)

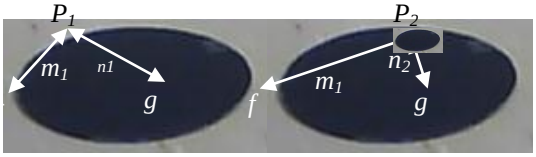


Fig. 5.2: Characteristic of ellipse

Step 3: *Identification of forward pattern from all detected ellipse:*

(i) Group ellipses with $Dp_{S2L} < Dp_{max}$ and $LD_{S2L} < LD_{max}$ as shown in Fig. 5.3, where:

Threshold depth between two ellipses, $Dp_{max} = V_R / n$, Threshold lateral distance between two ellipses, $LD_{max} = H_R / n$, with n is the maximum number of paired ellipses exist in the image frame. V_R and H_R denoted in Fig. 4.11 are the height and width indication of a captured image.

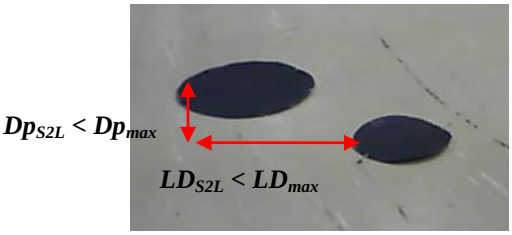


Fig. 5.3: Allowable distance between ellipses

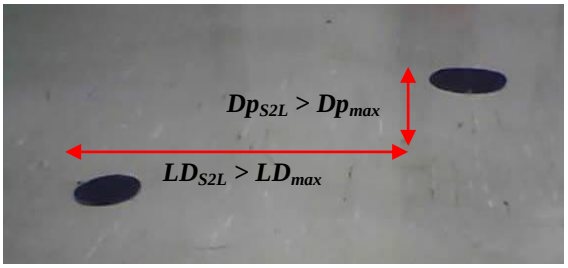


Fig. 5.4: Prohibited distance between ellipses

(ii) *Extract paired ellipses*: Denote number of paired ellipses as *G*. Extracted *G's* ellipses, store them in a data pool. If no paired ellipses detected, $G = 0$ and proceed to searching ellipse by scanning surrounding floor in Step 4.

(iii) Remove similar sized paired ellipse, $R_m < R$ from *G* in data pool, where $R_m = M_L - M_s$, is the difference between length of major axis of ellipses in the group and *R* is the maximum acceptable threshold value of R_m, by preference set equal to $M_L/2$ because the diameter of large circle in the floor pattern designed is twice of the diameter of small circle in the floor pattern. M_L is the length of major axis for larger ellipse and M_s is the length of major axis for smaller ellipse as shown in Fig. 5.5. If no paired ellipses is left in the data pool ($G=0$), then proceed to searching ellipse by scanning surrounding floor in Step 4.

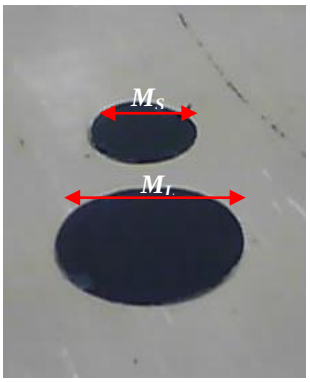


Fig. 5.5: Major axis of large ellipse and small ellipse smaller than *R* .

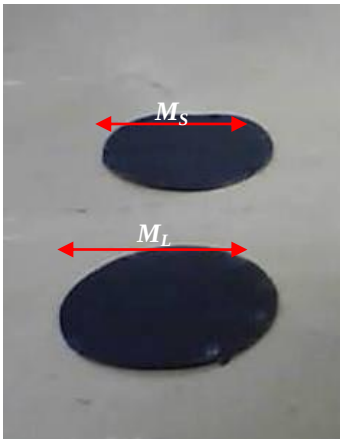


Fig. 5.6: Major axis of large ellipse and small ellipse larger than *R*.

(iv) *Home and destination pattern recognition*. From the remaining paired ellipses left in data pool, if two paired ellipses group of forward pattern shared the same small ellipse or large ellipse which indicates home or destination pattern, then denote the center shared ellipse as large ellipse in *P* and proceed to Step 5.

(v) If no home or destination pattern recognized, eliminate pairs which does not indicate forward direction form data pool. $X_L < X_S$ indicates forward direction for home to destination as shown in Fig. 5.7 while $X_L > X_S$ indicates forward direction for destination to home Fig. 5.8. X_L and X_S are the horizontal centroid coordinates of large and small ellipse respectively.

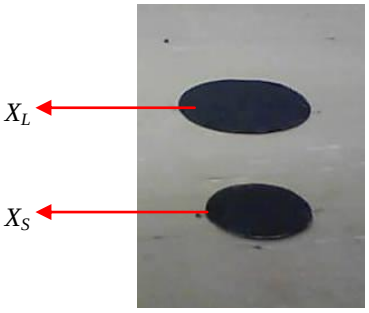


Fig. 5.7: Floor pattern indicating forward direction for home to destination routing ($X_L < X_S$).

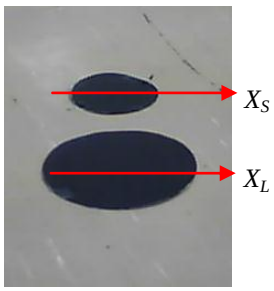
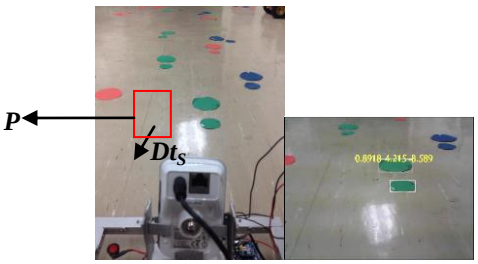


Fig. 5.8: Floor pattern indicating forward direction for destination to home routing ($X_L > X_S$).

(vi) *Find nearest paired ellipse from data pool:* If there is remaining paired ellipse in data pool, $G \neq 0$, denote the paired ellipses with smallest Ds_I as P as shown in Fig. 5.9, where D_{SI} is the imaginary distance of small ellipse in each group of forward pattern from robot and proceed to real distance and angle calculation in Step 5.

(vii) If no remaining paired ellipse in data pool, $G = 0$, proceed to searching ellipse by scanning surrounding floor in Step 4 when no remaining paired ellipse in data pool.



(a) (b)

Fig. 5.9: (a) Floor pattern placement from camera's direction (b) Closest floor pattern is recognized

Step 4: Search ellipse by scanning surrounding floor:

When there is no forward pattern in capture zone, denote θ_{Max} as the maximum acceptable range for camera rotation to the right. Denote θ_{search} as the angle of rotation for each search which is 30°, SR_Z as indicator of robot position in diamond shape path finding and SR_T as a indicator of camera rotation in hemisphere path finding. Proceed with diamond-shape path finding in step 4(i) when SR_T is smaller than 6, else proceed to hemisphere path finding in step 4(ii).

(i) Diamond shape path finding,

Case 1: $SR_Z = 0$

1) Rotate camera to origin.
2) Rotate robot to the right by 45° to avoid obstacle and send command to move robot for a distance, *M*.
3) Rotate robot back to centre position by -45°.
4) Set $SR_T = 0$ and $SR_Z = SR_Z + 1$.
5) Repeat step 1 for new cycle.

Case 2: $SR_Z = 1$

1) Rotate camera to origin.
2) Rotate robot to the left by 90° to avoid obstacle and send command to move robot for a distance, $2M_D$*sin45°.
3) Rotate robot back to centre position by 90°
4) Set $SR_T = 0$ and $SR_Z = SR_Z + 1$.
5) Repeat step 1 for new cycle.

Case 1: $SR_Z = 2$

1) Rotate camera to origin.
2) Rotate robot to the right by 45° to avoid obstacle and send command to move robot for a distance, *M*.
3) Rotate robot back to centre position by -45°.
4) Set $SR_T = 0$ and $SR_Z = 0$.
5) Repeat step 1 for new cycle.

(ii) Hemisphere path finding,

Case 1: SR_T is odd value

1) Determine camera's turning angle, θ_{Right}, to the next searching range on right side with equation (18).

$$\theta_{Right} = \theta_{search} \times SR_T + \theta_{search} \quad (18)$$

2) Transmit turning angle to main control to turn camera.
3) Increment SR_T by 1.

Case 2: SR_T is even value

1) Determine camera's turning angle, θ_{Left}, to the next searching range on left side with equation (19).

$$\theta_{Left} = -\theta_{search} \times SR_T - \theta_{search} \quad (19)$$

2) Transmit turning angle to main control to turn camera.
3) Increment SR_Z by 1.

Step 5: *Distance and angles calculation*:

(i) Calculate real lateral distance, *LD* and depth, *Dp*, of both ellipses in *P* from robot's position by using the equation (9) and (11).

(ii) Calculate real distance of ellipses in P by using Pythagoras theorem as in equation (1) and/or (2).

(iii) Calculate angle of large ellipse in P from robot's position by using equation (3).

(iv) If number of ellipse in P is more than 1, calculate distance between ellipses, D_{S2L} by using equation (4).

a. Calculate angle to face next guided path, θ_R , by using Equation (5).

b. Determine turning direction of θ_R by verifying the following conditions,

1. When $A_L > A_S$, direction of θ_R: Clockwise.
2. When $A_L < A_S$, direction of θ_R: anticlockwise

(v) Draw bounding box around the targeted pattern and label with the distance of each ellipse of the identified floor pattern as shown in Fig. 5.10.

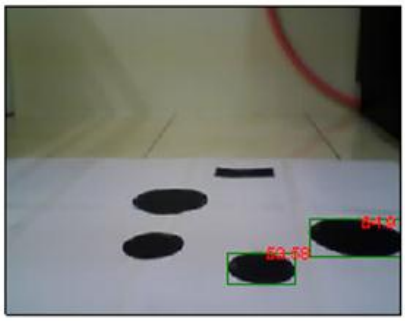


Fig. 5.10: Resultant image after step 5(v)

Step 6: *Align floor pattern to centre axis of image frame:* If A_L equals to 0° proceed to Step 7, else,

(i) Transmit A_L to rotate robot until the large ellipse is aligned at the centre axis.

(ii) Repeat Step 1 for new cycle.

The purpose of aligning the floor pattern to centre is to ensure home or destination pattern is not treated as guided path pattern. Besides, this step can also improve the accuracy of the calculated distance and angles of ellipse from robot.

Step 7: *Output distance and angles to robot*:
Transmits D_L, A_L and θ_R to the control terminal.

Step 8: *Arrival at home or destination*:

A. Case1: arrival at destination: Robot rotates 180° and repeat Step 1 for new cycle to move back to home.

B. Case 2: arrival at home: Process ended.

C. Case 3: none of the above: repeat Step 1 for new cycle.

VI. EXPERIMENTAL RESULTS

In this section, physical experiments are carried out on the proposed VBFPT algorithm on robot routing. Several experiments are conducted to determine the optimum parameters stated within the algorithm. The setting of the experiment is at Center for Robotic and Automation Lab, Faculty of Engineering and Technology, Multimedia University, 75450 Melaka, Malaysia. Subsection A discusses the optimum setting range for tilting angle and height of camera. Subsection B discusses the single path routing conducted on clean background. Subsection C discusses the multiple paths routing conducted on harsh background. Subsection D discusses the performance of the robot routing system in different light luminance level. Subsection E discusses the capability of guided path finding on VBFPT algorithm.

A. Setting Optimum Range for Tilting Angle and Height of Camera

For the robot system to work efficiently, tilting angle, θ and height of camera, h have to be set with optimum values. Based on the vertical angle of view of the camera α, the camera should be focus at the ground at the angle $\theta > \alpha$. This would bring two benefits to the system: 1) the camera would not capture image of horizon, thus the image capturing can be focused on object on the ground much closer to the camera without immense divergence; 2) unrelated objects at far distance would not be captured and thus would not distort the image processing.

DCS-930L wireless webcam is used by the robot structure as imaging tool. It has vertical angle of view at 17.25° and lens height of 6.5cm. The imaging tool is tilted for more than 17.25° to make it focus on the ground level image capturement. As the tilting angle θ increases, the maximum viewing range of the camera decreases and thus floor pattern need to be put closer to each other. This increases the number of symbols needed between home and destination positions. Therefore optimum range of tilting angle set will be between 20° and 45°. Hence, a tilted angle θ =20° is well selected in this paper for maximize the coverage of nearest symbols placed.

The height of imaging tool h is important variable and it should be set realistically. Since the imaging tool should have a base to operate on, the minimum imaging tool height from ground h should be around 30cm. As for maximum height 1.5m was set to replicate the height of a normal humanoid robot.

The horizontal viewing angle of camera is 34.5° and the vertical viewing angle of camera is 45.3°, height of camera from ground fixed with 0.37meter. This allows the maximum number of paired ellipse, n, existed in the image frame is 4. From there, the camera has resolution H_R (640 pixels) x V_R (480 pixels). From there, it can be further calculated that Dp_{max} = 120 and LD_{max} = 160. The parameters, Dp_{S2L}, LD_{S2L}, M_L, M_S, X_L, X_S, Dp_L, LD_L, D_L, A_L, D_S, A_S and θ_R, are all measured from the captured image.

To evaluate the proposed VBFPT algorithm described in Section V, a series of physical experiment test was conducted

using randomly organized Home-Destination-Home guided path test cases. Each test case consisted of one Home floor pattern, 10 guided path directed pattern and one destination pattern in a. research lab test-site with the dimension of 12 meters x 12 meters. 1000 different path cases were generated. An example test site was shown in Fig. 6.1.

Fig. 6.1: Example test site.

B. Single Path Clean Background Routing

1000 experiments were conducted to study the performance of algorithm for the robot to travel from Home to Destination on clean background. In this test, single destination was used and the floor pattern selected is black in colour. Out of the 1000 experiments conducted, 942 success cases and 58 failed cases were recorded, results in success rate of 94.2%. The 58 cases can be further classified into three main cause categories, which are: misrecognition of floor pattern, partial capture of path routing floor pattern and wrong ellipse grouping.

Cause Category 1: Misrecognition of Floor Pattern

45% of the 58 failed cases were caused by misrecognition of destination floor pattern as path routing floor pattern when the destination floor pattern was not fully capture in the image. An example is shown in Fig. 6.2. In the figure, partial (only 2 ellipses) of destination floor pattern were captured, results in misrecognition of these 2 ellipses as path routing floor pattern. Since destination floor pattern is misrecognized, the robot failed to identify its correct position and perform the right task.

This problem was resolved by aligning the centre ellipse to the centre of image to ensure that the whole destination floor pattern lies within the capture range. Fig. 6.3 shows the correct recognition of destination floor pattern application of Step 6 in the proposed VBFPT algorithm.

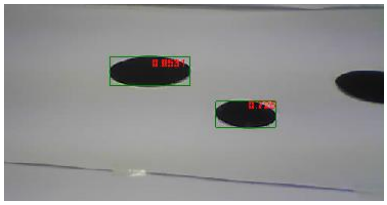


Fig. 6.2: Misrecognition of destination floor pattern

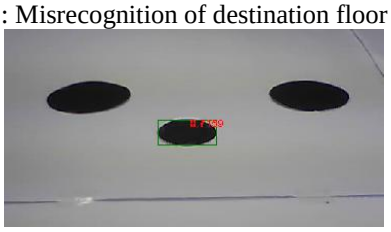


Fig. 6.3: Corrected recognition of destination floor pattern recognition

Cause Category 2: Partial Capture of Path Routing Floor Pattern

39% of 58 failed cases were caused by failure in floor pattern detection during path finding process. Fig. 6.4(a) shows an example for one of the path routing floor patterns is partially captured. Fig. 6.4(b) shows the image captured after camera rotated by angle to next searching range, θ_{search} during hemisphere path finding process. After performed camera rotation, the captured floor pattern was still failed to fully lies within the capture range because there was no overlapped range between these two images when θ_{search} equals to or larger than twice of the horizontal viewing angle. In order to overcome this problem, a smaller value of θ_{search} was chosen to ensure that the floor pattern could be fully detected during path finding process. Fig. 6.4(c) shows that the floor pattern lies within the searching range when the same path finding process is conducted with smaller θ_{search} selected, which is 30^0 in these cases. A bigger θ_{search} selection will cause overlooked cases as in Fig. 6.4(a) and Fig. 6.4(b), whereas a smaller θ_{search} selection will cause spending longer searching time. A justified selection of θ_{search} might balance both under coverage and optimum time expenditure.

(a) (b) (c)

Fig. 6.4: (a) Ellipse partially captured (b) Floor pattern partially captured during path finding (c) Corrected floor pattern recognition

Cause Category 3: Wrong Ellipse Grouping

16% of 58 failed cases were caused by wrong ellipse grouping when the grouping process is conducted based on the resultant distance between ellipses. Fig. 6.5(a) shows an example of grouping the ellipses pairs. The small ellipse is grouped with two ellipses labelled with 1 on its right and left sides. The small ellipse supposed to be grouped with ellipse on the right and recognized as path routing floor pattern. However, due to the wrong grouping, the floor pattern is recognized as home position. The small ellipse is wrongly grouped with the left

ellipse because it has resultant distance which is similar to the correct pair in Fig. 6.5(b).

Instead of resultant distance, to resolve such problem, depth and lateral distance between ellipses are used to determine ellipses grouping. Although both pairs (small and left ellipses in Fig. 6.5(a) and pair in Fig. 6.5(b)) have similar resultant distance, but the depth difference between ellipses in both pairs has big contrast. The pair in Fig. 6.5(a) has smaller depth difference while pair in Fig. 6.5(b) has larger depth difference. Therefore, the small and left ellipses were not be grouped when depth and lateral distance were used to replace resultant distance as the criteria in ellipse grouping. Fig. 6.5(c) shows correct floor pattern recognition after the new proposed method (with depth and lateral distance info embedded) is applied in Step 3(i) of the proposed VBFPT algorithm.

(a) (b) (c)

Fig. 6.5: (a) Wrong ellipse grouping (b) Permitted resultant distance (c) Corrected ellipse grouping and floor pattern recognition

C. Multi Paths Harsh Background Routing

1000 experiments were conducted to study the performance of the algorithm. The test was conducted on harsh background. The experiments involved three paths of red, green and blue colour floor patterns respectively. The end result shows 953 succeed and 47 failed cases out of 1000 attempts, measuring up to 95.3% success rate. The 47 cases can be further classified into three main cause categories, which are: incomplete ellipse detection by using RGB colour model, incomplete background filter by using Hue component and incomplete ellipse recognition under dark condition.

Cause Category 1: Incomplete Ellipse Detection by Using RGB Colour Model

52% of the 47 failed cases were due to incomplete ellipse detection when the image was segmented based on RGB colour model. Initially, RGB colour model was used for colour detection and segmentation in the proposed VBFPT algorithm. However, results showed that RGB colour model did not performed very well in recognizing the difference in colours. Fig. 6.6(a) shows an example of the image of floor patterns without reflection of sunlight while Fig. 6.7(a) shows the image of the patterns with reflection caused by sunlight.

RGB colour model was used in both images to detect the green colour ellipses' path. The green ellipses can be detected without any problem at the absence of the reflection of sunlight (e.g. in Fig. 6.6(b)). However, the same ellipses cannot be detected clearly when there were reflections of sunlight on the patterns (e.g. in Fig 6.7(b)). The undesired presence of sunlight on the patterns affected the colour detection using RGB colour model. This problem was solved by replacing RGB colour model with HSV (Hue-Saturation-Value) colour model. The advantage of HSV over RGB is instead of recognizing colours based on the mixtures of Red, Green and Blue, HSV differentiates colour based on colour type (Hue), luminance (Value) and chrominance (Hue & Saturation). Fig. 6.6(c) and Fig. 6.7(c) show that green ellipses extracted from both Fig. 6.6(a) and Fig. 6.7(a). They are correctly detected using HSV colour models. Number of failed cases decreased after HSV colour model is applied in Step 2(ii) of the proposed VBFPT algorithm.

(a) (b) (c)

Fig. 6.6: (a) Image captured with indoor lighting (b) Image filtered by using RGB colour model (c) Image filtered by using HSV colour model

(a) (b) (c)

Fig. 6.7: (a) Image captured with sunlight interception (b) Image filtered by using RGB colour model (c) Image filtered by using HSV colour model.

Cause Category 2: Incomplete Background Filter by Using Hue Component.

37% of the 47 failed cases were due to incomplete filtration of background from ellipses when Hue component of HSV colour model is used. Despite the ability of Hue component to clearly differentiate specific colour from numerous colours, the application of Hue component alone still insignificant to completely separate the background from the patterns and thus the filtered image appeared distorted. Fig. 6.8(a) shows the resultant image filtered from Fig. 6.6(a) using hue component alone.

By combining the Hue and Saturation components of HSV colour model in colour detection, the green ellipses can be clearly differentiated from other ellipses as well as the floor background. Fig. 6.8(b) shows the image filtered by using Saturation Component only. The image shows that the Saturation component was able to determine the saturation level with the most number of image pixels and classify them as the background. However, Saturation component cannot differentiate the difference in colours; therefore Hue component is applied in conjunction with Saturation component to identify the green ellipses from the rest of the image. Fig. 6.8(c) shows image filtered with Hue and

Saturation components. This method is applied in Step 2(iii) and (iv) of the proposed VBFPT algorithm.

(a) (b) (c)

Fig. 6.8: (a) Image filtered by using Hue component (b) Image filtered by using Saturation component (c) Image filtered by using Hue and Saturation components.

Cause Category 3: Incomplete Ellipses Recognition under Dark Condition

11% of the 47 failed cases were due to incomplete ellipses recognition under dark condition. The lack of light during image capturing weakened the performance of ellipse detection due to the increase of noise presented in the image. Despite having excellent results, the ellipse detection by using the combination of Hue and Saturation components (the solution as mentioned in cause category 2) still tends to fail under low light condition due to flicker noises on the images. Fig. 6.9(a) shows an example image captured on red colour patterns under low light environment. Fig. 6.9(b) is the filtered image of the red ellipses, where several areas on the red ellipses were mistakenly filtered out. This resulted in the larger red ellipse to be incorrectly treated as two ellipses instead of one big ellipse as shown in Fig. 6.9(c).

(a) (b) (c)

Fig. 6.9: (a) Image captured under dark condition (b) Image filtered without closing process (c) Single ellipse detected as two objects.

This problem were solved by introducing a process named "Closing Process" to remove the noises presence on the ellipses. Closing process is a process of dilation followed by erosion using the defined structuring element. Fig. 6.10(a) shows the successful detection of the red ellipses on the image. The red ellipses were recognized faultlessly as shown in Fig. 6.10 (b). The closing process is applied in the proposed VBFPT algorithm at Step 2(vi).

(a) (b)

Fig. 6.10: (a) Image filtered with closing process (b) Accurate recognition of ellipses.

D. Robot Routing With Different Luminance Level

This section discusses the performance of the robot routing system in different light luminance level. The total number of experiment that had been carried out is 800 attempts. The experiments are carried out on harsh background environment consists of three routing paths which are differentiated by red, green and blue colours. This test is divided into four group based on the luminance level and each group is conducted for 200 times. The luminance level is measured by Lux (lx). In previous section, the robot routing for multiple paths is done with full brightness test which ranged from 1300lx to 1600lx. Therefore, the luminance levels set for this section's test are classified into bright, medium, dim and dark conditions. Bright condition is ranged from 1060lx to 1250lx, medium condition is ranged from 423lx to 923lx, dim condition is ranged from 40lx to 50lx and lastly, dark condition is ranged from 4 to 10lx. The values of Hue component for red, green and blue are set based on their range specified by HSV colour model. Whereas, threshold values of Saturation component for each colour are set differently. Saturation threshold for red colour (S_{Red}) is 0.5, green colour (S_{Green}) is 0.35 and blue colour (S_{Blue}) is 0.5. Structuring element (S_{SE}) of closing process is set to 1. S_{Red}, S_{Green}, S_{Blue} and S_{SE} have to be varied based on difference luminance level at the site.

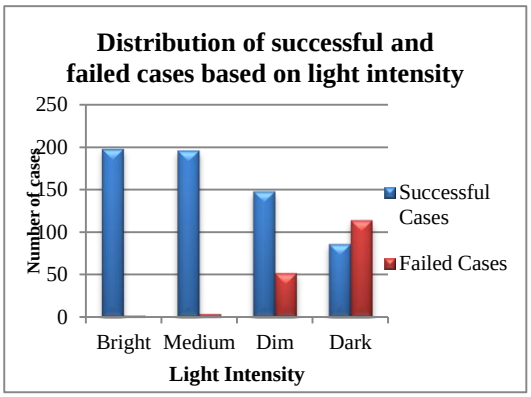


Fig. 6.11: Distribution of successful and failed cases under different light intensity

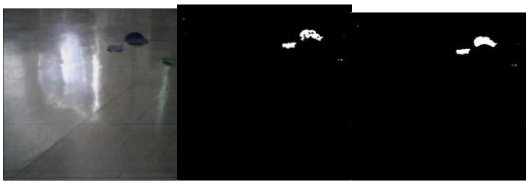
Out of the 800 attemps, 628 of success cases and 172 of failure were recorded for Home-Destination-Home routing which results in 78.5% success rate. The number of success and failure cases for each condition is plotted in Fig. 6.11. From the plot, it can be observed that most of the recorded failures happened in dim and dark conditions. There are two cases determined from the failures which are: 1) wrong ellipse detection due to structuring element, S_{SE}, and 2) partial ellipse detection due to thresholds of Saturation component.

Case 1: Wrong Ellipse Detection Due to Structuring Element, S_{SE}.

59% of the 172 failure cases were caused by wrong ellipse detection due to low setting of S_{SE} in dim and dark conditions. The closing process which was applied in case 3 of previous test is useful to eliminate unnecessary noise in the ellipse from colour recognition. However, the value of structuring element, S_{SE}, will determine the effectiveness of closing process. A large S_{SE} could cause negative effect such as combining two

different ellipses while a small setting would be insufficient to eliminate all the noises.

In dim and dark condition, most ellipses are not able to be recognized due to the presence of noise in the ellipses as shown in Fig. 6.12(a) and when the original image captured in dim condition as shown in Fig. 6.12(b) is filtered with S_{SE} equals to 1. The noises are filtered by increasing S_{SE} to 2. During dark condition test, more noises are detected from the captured image. Therefore, S_{SE} is further increases to 5 to improve accuracy in detecting ellipses. The improvement is shown in Fig. 6.12(c).



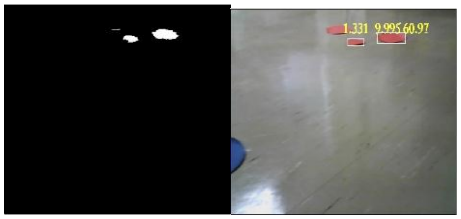
(a) (b) (c)

Fig. 6.12: (a) Image captured in *Dim* condition (b) Image filtered by S_{SE} = 1 (c) Image filtered with S_{SE} = 2.

Case 2: Partial Ellipse Detection Due to Thresholds of Saturation Component

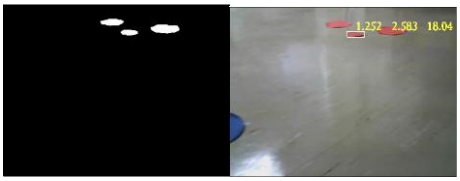
41% of the 172 failure cases were caused by wrong ellipse detection due to wrong threshold setting for Saturation component. Saturation component is used to separate objects from background. This separation could be done because background has same saturation level and it occupied the largest area in the image. In low luminance level, the saturation level of each pixel in the image would decreases and falls into the filtering range when same saturation thresholds were used. In other words, portion of the ellipse would be recognized as the background and become noise in ellipse detection.

By using the same Saturation threshold in bright condition, red ellipses are partially detected during routing in medium and dim conditions. However, detection of green and blue colours are not affected. Due to partial detection of red ellipse in destination pattern as shown in Fig. 6.13(a), the floor pattern misrecognized as pattern for path routing as illustrated in Fig. 6.13(b). This problem is solved by decreasing S_{Red} to 0.4 and the improvement is shown in Fig. 6.14(a) and Fig. 6.14(b).



(a) (b)

Fig. 6.13: (a) Image filtered with S_{Red} = 0.5 (b) Misrecognition of floor pattern



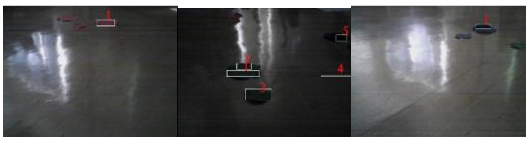
(a) (b)

Fig. 6.14: (a) Image filtered with S_{Red} = 0.4 (b) Corrected floor pattern recognition

When the light intensity dropped to dark condition, all three colours recognition are affected as shown in Fig. 6.15 and the drawback of these effects are shown in Fig. 6.16. Most of the ellipses in the images are either not recognizable or recognized as two objects. Therefore, thresholds ae further decreases to S_{Red}=*0.3*, S_{Green}=0.25 and S_{Blue}=0.3.

(a) (b) (c)

Fig. 6.15: (a) Image filtered with S_{Red} = 0.4 (b) Image filtered with S_{Green} = 0.35 (c) Image filtered with S_{Blue} = 0.5.



(a) (b) (c)

Fig. 6.16: (a) One out of three ellipse detected (b) Large ellipse detected as two objects (c) Half of an ellipse detected.

After the changes are applied to threshold of Saturation component, the colour recognition is improved. Fig. 6.17 shows the improvement in filtering process with decreased thresholds and Fig. 6.18 shows the improvement in ellipse detection. The performance of colour recognition in medium condition is less accurate as compared to bright condition. However, this performance dropped drastically in dim and dark condition. Since most indoor environment is lighted with bright or medium conditions, the designed robot is suitable to be applied.

(a) (b) (c)

Fig. 6.17: (a) Image filtered with S_{Red} = 0.3 (b) Image filtered with S_{Green} = 0.25 (c) Image filtered with S_{Blue} = 0.3

(a) (b) (c)

Fig. 6.18: Corrected ellipse detection for (a) red (b) green and (c) blue coloured ellipses.

E. Capability of Guided Path Finding

The total number of experiment that had been carried out is 500 attempts. The experiments are carried out on harsh background environment consists of three paths which are differentiated by red, green and blue colours under bright condition with brightness of 1060lx to 1250lx. Out of 50 attempts, 449 were the successful cases and another 51 were failures for a Home-Destination-Home routing which means 89.8% success rate. There are two cases which were determined from the failure cases namely undetectable zone for guided path finding by using zig-zag pattern and failure in recognizing home or destination floor pattern with vertical orientation.

Case 1: Undetectable Zone for Guided Path Finding by Using Zig-zag Pattern

56% of the 51 failures were caused by undetectable zone during guided path finding by using zig-zag pattern. During the initial stage of path finding algorithm design, zig-zag pattern is used to search for floor pattern which guide the robot back to the guided path. This movement pattern is used to widen the area of search but there is a drawback by using zig-zag pattern. There is a certain zone that is not included in the searching as shown in Fig. 6.19. The robot would missed the floor pattern which situated in that zone.

In order to encounter this problem, diamond pattern path finding (as shown in Fig. 6.20) is used to replace the zig-zag pattern in guided path finding. Diamond pattern path finding has advantage of larger covering range as compared to zig-zag pattern. The undetectable zone in zig-zag pattern is solved by using diamond pattern. This change is applied into Step 4(i) of the proposed VBFPT algorithm to minimize failure caused by this problem.

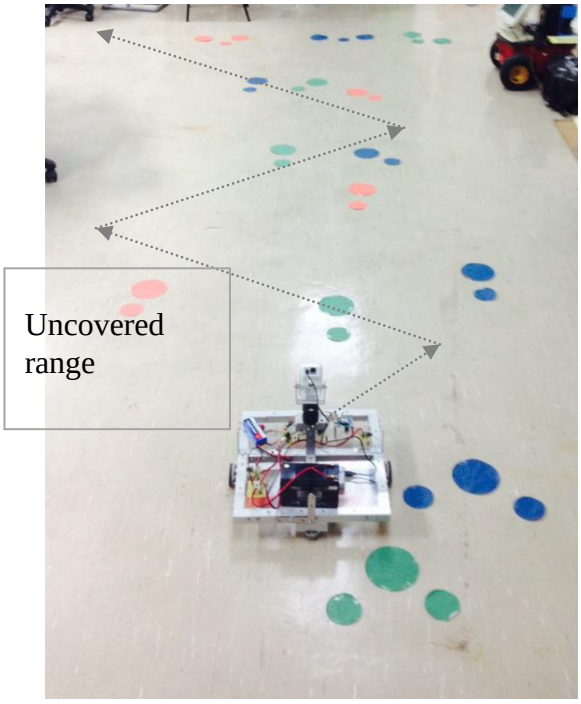


Fig. 6.19: Uncovered range in path finding using zig-zag pattern.

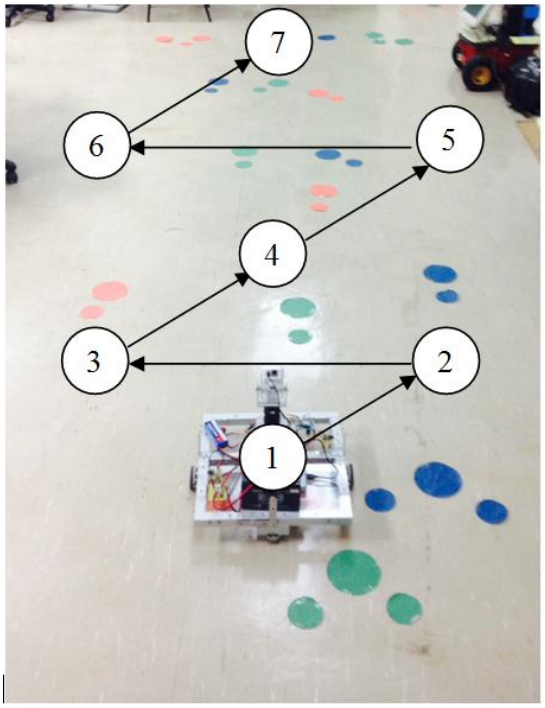


Fig. 6.20: Movement of robot in diamond-pattern path finding

Case 2: Failure in Recognizing Home or Destination Floor Pattern with Vertical Orientation.

44% of the 51 recorded failures were caused by failure to recognize home or destination pattern when these patterns are in vertical orientation. These floor patterns are usually arranged in horizontal orientation as shown in Fig. 6.21(a). However, they were captured in vertical orientation during path finding process since the movement of robot is guided by the diamond pattern instead of the guided path. At the initial stage, the recognition of these patterns was done based on x-axis centroid coordinates of ellipses in the pattern. When the pattern is orientated, vertically, the x-axis centroid coordinate of the center ellipse is not necessary lying between the coordinate of the other two ellipses as shown in Fig. 6.21(b). Therefore, the robot failed to navigate to the pattern since the pattern is not recognizable.

This problem is solved by adding new design into the algorithm for home and destination floor pattern recognition. Y-axis centroid coordinate of each ellipse in the pattern is also taken into consideration to encounter the problem. When the pattern is oriented vertically, the y-axis centroid coordinate of center ellipse would lies between the coordinate of the other two ellipses. Fig. 6.21(c) shows the improvement after modification is implemented in Step 3(iv) of the proposed VBFPT algorithm.

(a) (b) (c)

Fig. 6.21: (a) Home position floor pattern in horizontal orientation (b) Undetected home position floor pattern in vertical orientation and (c) Corrected recognition of floor pattern in (b)

VII. CONCLUSION

This paper described the design of a new vision based robot routing method utilizing elliptical geometry approach in floor pattern tracking. The proposed vision based floor pattern tracking method has been implemented in a real world robot routing scenario. Remarkable floor pattern is depicted using circular geometry, treated as path guided and home-destination symbols. A robot platform had been constructed to evaluate the performance of the vision based floor pattern tracking algorithm for robotic routing purposes. Experimental results shown that the proposed vision based floor pattern tracking algorithm supplies solutions to single path, multiple paths robotic routing under various floor background and light intensity conditions. The robotic routing method achieves expandability compared to conventional sensory based and virtual map based approach because both users and researchers might use the proposed algorithm and the configurable modular robot hardware as a mobile logistic robot or as several other autonomous functional robots. In future, the algorithm will further considered tested in bumpy outdoor environment, and embed with image processing based obstacle avoidance features.

REFERENCES

[1] D. Botturi, P. Fiorini and S. Lonardi (2002). "Functional and Economic Justification for Service Robotics in the Pharmaceutical Industry" Proceedings of the 33rd ISR (International Symposium on Robotics), p.p. 1-6.
[2] K. N. Faress, A. A. El-Kousy and M. T. El-Hagry (2006). "Mobile Robot Navigation In Warehouses For Materials Handling", the Fourth Saudi Technical Conference and Exhibition, STCEX 2006, Vol. III, pp. 144-150.
[3] S. I. Amer , M. N. Eskander, A. M. Zaki (2012). "Positioning And Motion Control For Mobile Robot", International Journal of Emerging Tec hnology and Advanced Engineering, Vol. 2, Iss. 11, p.p. 498-504.
[4] T. Samad and A.M. Annaswamy (2011). "Success Stories For Control: Mobile-Robot-Enabled Smart Warehouses", The Impact of Control Technology, p.p. 1-2.
[5] C. Story (2005)."Using Robotics to Enhance Logistics Solution", Report of AHRRM of the American Hospital Association p.p. 1-32
[6] P.C. Loh , A. Johari, M. H. AbdWahab, D. Md. Nor, N. S. A. Md.Taujuddin (2007). An RFID Warehouse Robot. 2007 IEEE Int. Conf. on Intelligent and Advanced System (ICIAS2007), KLCC, p.p. 451-456.
[7] S. Liawatimena, B.T. Felix, A. Nugraha, R. Evans (2011). "A Mini Forklift Robot", 2011 The 2nd Next Generation Information Technology (ICNIT), p.p. 127 - 131.
[8] Wolfram Burgard Andreas Derr Dieter Fox Armin B. Cremers(1998) Integrating Global Position Estimation and Position Tracking for Mobile Robots: The Dynamic Markov Localization Approach Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'98), Victoria, BC, Vol.2, p.p. 730-735.
[9] I. Susnea, G. Vasiliu, A. Filipescu, A. Serbencu and A. Radaschin (2009). "Virtual Pheromones to Control Mobile Robots. A Neural Network Approach", International Conference on Automation and Logistics, Shenyang, China p.p. 1962-1967.
[10] M. Ballesta, A. Gil, O. Reinoso, L. Paya and L. M. Jimenez (2010). "Map fusion in an independent multi- robot approach", Journal WSEAS Trans. on Systems archive, Vol. 9 Iss. 9, p.p. 959-968.
[11] P. Alves, H. Costelha and C. Neves (2013). "Localization and Navigation of a Mobile Robot in an Office-like Environment", 2013 13th International Conference on Autonomous Robot Systems (Robotica), p.p. 1-6.
[12] R. Jarvis and N. Ho (2010). "Robotic Cybernavigation in Natural Known Environments", 2010 International Conference on Cyberworlds, p.p. 338 – 345.
[13] H. Bay, A. Ess T. Tuytelaars and L. Van Gool, (2008). "Speeded Up Robust Features", Computer Vision and Image Understanding, Elsevier, Vol. 110, Iss. 3, p.p. 346–359
[14] J. W. Yoon and S. J. Roberts (2010). "Robust Measurement Validation in Target Tracking Using Geometric Structure", IEEE Signal Processing Letters, Vol. 17, No. 5, p.p. 493-496.
[15] M. K. Hu (1979). "Visual Pattern Recognition by Moment Invariants", IRE Transactions on Information Theory, p.p. 179-187.
[16] D.A. Brannan., M.F. Esplen, J.J. Gray (2012). Geometry, Cambridge University Press, p.p 14-19.
[17] Kahn, J., & Barry, J. Wireless infrared communications. Proceedings of the IEEE, 85, 265-298.
[18] Carruthers, J. B. Wireless Infrared Communications. Wiley Encyclopedia of Telecommunications, 1.
[19] Ton, D. Radio Frequency (RF) Data Communications.
[20] John Donovan. (2010). Bluetooth low-energy: An introduction.
[21] Martin Hebel, George Bricker & Daniel Harris. Parallax Inc. (2010). Getting Started with XBee RF Modules.
[22] Paramasivam, C. & Sornakumar, T. (2009). Simulation and Development of Mobile Robot Optimal Path Planning Based on Computer Vision Technique. International Journal of Applied Engineering Research, 4(6), 1009-1029.
[23] Asami, K., Hagiwara, H. & Komori, M. (2012). Visual Navigation System Based on Evolutionary Computation on FPGA for Patrol Service Robot.
[24] Morales, B., Roberti, F., Toibero, J. M. & Carelli, R. (2011). Passitivity based visual servoing of mobile robots with dynamics compensation. Mechatronics, 22, 481-490.
[25] Callister, W. D. & Rethwisch, D. G. (2011). Materials Science and Engineering. Hoboken, NJ: John Wiley & Sons Inc.