CENTRALIZED VISION-BASED CONTROLLER FOR UNMANNED GUIDED VEHICLE DETECTION

Ravindra Thamma, Leela Mohan Kesireddy, and Haoyu Wang, Central Connecticut State University

Abstract

Traditional track-based Unmanned Ground Vehicles (UGV) cannot deviate from their routes due to this track limitation in their navigation methods. Track limitation has to be overcome in order to render a UGV more flexible. It is desirable to have a UGV move without tracks and have the ability to deviate to and from routine routes in order to have flexibility in tasks. In this study, the authors proposed a navigation system to aid multiple UGVs in navigating to various locations without any physical tracks and without colliding with one another. The authors demonstrated an image-recognition-based trackless navigation system to enhance the flexibility of multiple UGVs. To accomplish this feat, an image-recognition algorithm was developed to identify the position and orientation of multiple UGV's using a Centralized Image-Based Controller Unit (CIBCU). This CIBCU is connected to a vision system and radio-frequency (RF) communicator. The CIBCU then implements the image-recognition algorithm, anti-collision and navigation algorithm, and centralized control center to track and navigate multiple UGVs without physical tracks. A prototype was developed to demonstrate and test the Vision-Based Navigation System. Statistical analyses were carried out on this newly developed system in order to find behavior-of-positioning error.

Introduction

Conventionally, controlling Unmanned Ground Vehicles (UGV) and Automatic Guided Vehicles (AGV) has been a challenge. Tracks serve as a key element for navigation systems for a mobile UGV. UGV tracking is a critical component for providing position, directions, and travel information for motion along a trajectory with minimal deviation. Many researchers have proposed different tracking techniques such as dead reckoning, navigation using active beacons, landmark- and map-based navigation, ultrasound, and Global Positioning System (GPS).

Most of the available navigation systems make use of a constant exchange of data between the controller and UGV that is costly in more ways than one. The exchange of data contributes to a slower system, resulting in lower UGV velocity. Furthermore, these systems are complicated by various parts and are often constrained to a pre-defined area. The post-implementation cost can also be a factor against its use. It appears that a newer system that does not carry these drawbacks will be beneficial to this area of study.

Many navigation techniques have been used over the last two decades for tracking a UGV. Dead-reckoning [1] is a process of estimating one's current position based on a previously determined position of the UGV. This is accomplished by advancing a previous position based on a known path and speed over a period of time. However, incremental motion often results in errors. Other navigation systems use active beacons [2] such as laser, sonar, or radio. This technique determines the position of a UGV by drawing a triangle through installed beacons and measuring the distance. The disadvantages of this technique include inaccuracy of distance measurements caused by signal delay, as well as installation and maintenance costs. GPS systems are more advanced and accurate at tracking the position of a UGV, but do not work in an indoor environment where satellite signals are often blocked.

Presently, wireless techniques are extensively used to track the UGV using distance measurement techniques. Radio frequency (RF) and ultrasound [3,4] are extensively used in these navigation techniques. In some cases, both ultrasound and RF are used together for greater precision. All of the aforementioned navigation systems involve a constant exchange of data between the controller and the UGV, resulting in a large amount of overhead, and the employment of more sensors and constraints to the pre-defined landmarks. This results in a higher power consumption and shorter battery life of the UGV. It also results in slower operation and response.

Vision-based navigation systems generally use less data for tracking the UGV position, resulting in faster operation and response. This is different from previous navigation systems like the landmark- and map-based navigation techniques, which rely on predefined landmarks, maps, or pre-information about the environment [5]. No assumptions about the knowledge of the location are made for the vision-based navigation system.

The concept of vision navigation has been in development for the last 20 years [6] in the area of mobile robot navigation. Even though it was introduced to overcome the disadvantages in the previous techniques, it is implemented in accordance with the previous techniques. In many techniques, vision systems (i.e., cameras) are used to visualize the environment and to guide the robot. Vision systems are used to find and measure the location of 3D structures with respect to a CAD-model [7]. The integration of a CAD model to visual measurement and direct feedback of measurement results to the CAD model is a key aspect for this technique. In other techniques, vision systems are used to generate a three-dimensional (3D) environmental map from data taken with stereo vision [8]. Vision systems are used to develop more precise segmentation. From the obtained segmentation, a 3D environment is built using occupancy grid and floor height maps. In another vision-based technique, vehicle position and orientation are determined using panoramic images [9]. Omni-directional sensors are used for obtaining a 360º field of view. Recognizing landmarks in a panoramic image from a prior model of distinct features in a given environment gives information about the robot's location.

Most of these techniques rely on assumptions based on prior knowledge of the scene. Some researchers have proposed using a vision-based system that functions without any prior knowledge of the scene. In this technique, a stereo -based vision system is built from feature correspondences and 3D information from image sequences of the scene [10]. This method uses two cameras for capturing the image frames at a fixed point in time. One camera is used to capture interface images and a second camera is used to collect the stereo image. The relative position of the camera motion is then estimated by registering the 3D feature points from two consecutive image frames.

There are many different vision navigation techniques proposed by prominent researchers. Various techniques utilize different methodologies to track UGVs with a vision system. Some of the vision techniques use a prior model of the environment [11]. Some of the techniques draw imaginary horizontal and vertical lines to find the position of the vehicle [9]. Other techniques use information from gray-scale images to find the path clearance to navigate the vehicles [12].

Some vision techniques use panoramic imagery. Omni-directional sensors are used in obtaining a 360º field of view, permitting the various objects near a robot to be imaged simultaneously. The robot's location is found by recognizing landmarks in a panoramic image from a prior model of distinct features in a given environment [11] (Guerrero, 2001). Other vision techniques find the position of the vehicle using collective measurement data obtained directly from the raw data of gray-level images. Such data is independent of the 3D surface texture, is measured in dimensional units, and requires no 3D reconstruction. The control schemes are based on a set of "if / then" fuzzy rules with almost no knowledge about the vehicle's dynamics, speed, and heading [12]. In some techniques, robot navigation is calibrated based on navigational lines. The position of a robot is based on extracted straight lines, assuming that the robot moves on level ground. The effect in the image of camera rotation is computed from the homography of a line at infinity. The corresponding vertical lines in two uncalibrated images are then used to compute both the robot heading and a region in the image that corresponds to the free space ahead [11].

One more important feature to be considered in vision navigation is the nature of the vision system. The number and placement of vision-system cameras play an important role in the function and performance of the navigation system. Some techniques utilize vision systems that are placed on the vehicle [13], while in others they are placed stationary in the navigation field [8]. Some techniques have only one vision system, while others utilize multiple vision systems placed at different positions in the navigation system.

Although vision-based navigation systems are designed to overcome the disadvantages of the traditional navigation systems, some still depend on traditional techniques like maps and developing of 3D environments from an image system. When such systems use more than one vision system, this further complicates the implementation of vision-based navigation. A vision-based system must be implemented in such a way as to overcome all of these disadvantages, while navigating on level ground without using any tracks.

All of the aforementioned systems need large infrastructure and software, resulting in complex and costly techniques. They are all dependent on the previous techniques and require some type of assumptions about the environment. Therefore, there is a need for the development of a navigation technique which uses less infrastructure and simpler algorithms. Such a system should provide navigation for multiple vehicles with lower overhead and less software and hardware.

In this study, a Vision-Based Navigation System was developed to navigate a UGV from a given position to a predefined final position based solely on this system. Automated software, developed as a Centralized Image-Based Controller Unit (CIBCU), would run the algorithms for vision processing, orientation, anti-collision, and navigation. It was tested using a prototype of the vision-based navigation system and UGVs, and data analysis was carried out on the test data. The goal of this study was to develop a prototype Vision-Based Navigation System to track multiple Unmanned Ground Vehicles.

Methodology

The vision-based navigation system developed in this study involved processing the image generated by the vision system and navigating multiple UGVs such that they would not collide with each other in accordance with a predefined priority. The basic layout of the proposed solution as shown in Figure 1 helps to illustrate the methodology. The vision system generates the images, and sends them to the vision-processing algorithm. The vision-processing algorithm then processes the image and generates the coordinates of each vehicle. The orientation algorithm then processes the orientation of each vehicle using the coordinates generated by the vision-processing algorithm. An anti-collision algorithm checks the probability of collision and stops the UGV according to its priority. The navigation algorithm navigates the vehicles according to the orientation of each vehicle. The data of the navigation are then transmitted to the vehicles using RF communication.

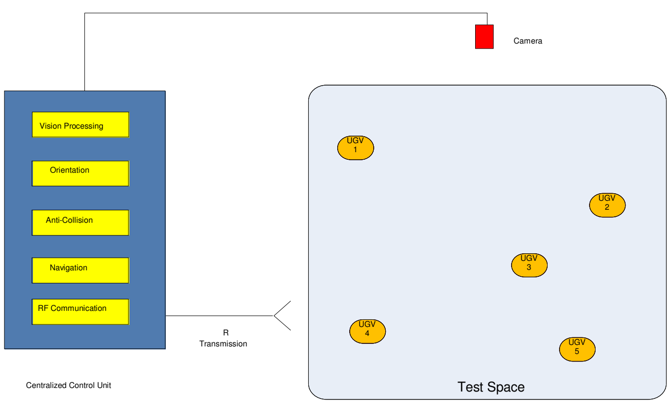
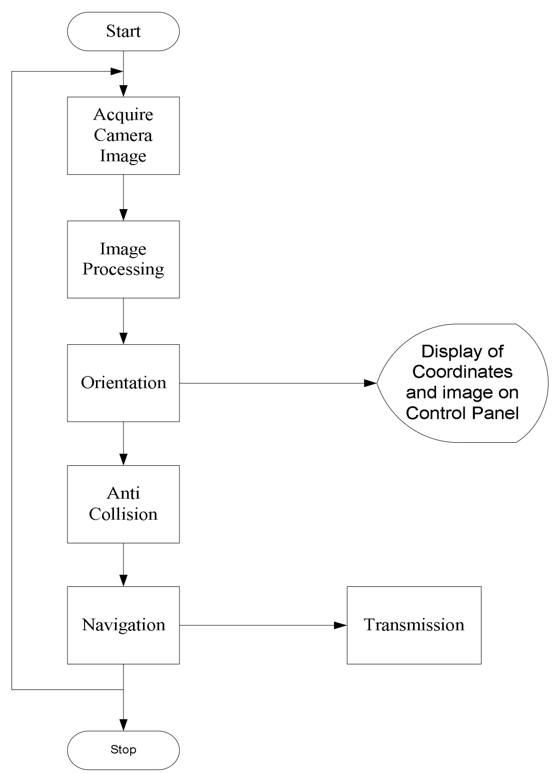


Figure 1. Layout of the Proposed Vision-Based Navigation System

The CIBCU Control Panel is the user interface for operating the vision-based navigation system and works with all five main parts of the system. This control panel provides manual control levers and overriding capabilities to the human user, and has a user-friendly interface that provides user controls for all of the UGVs. The positions of the different UGVs are captured through the vision system camera; these images are then transferred to the vision-processing algorithm. The vision processing system uses the data from the images to provide x and y coordinates for the UGVs. The orientation system calculates the orientation angle of each UGV. The position and orientation information is used by the anti-collision algorithm to find the possibility of any collisions. Finally, the information about the x and y coordinates is fed to the navigation-control algorithm.

The navigation algorithm forms the heart of the entire set up as this system acts as the brain. Navigating all of the UGVs from their present positions to their final positions is the basic responsibility of the navigation algorithm. The RF communication control system handles the transmission of control data to each UGV. A conventional multiplexed radio transmitter serves as the communication medium.

All of the above discussion is developed using a VB.net program. The main function of this program is to navigate the UGVs according to the positions and orientations derived from the latest acquired images. The flow of the program is to acquire the latest image from the vision system, resize and compress the image for faster processing, generate the coordinates of the vehicles by comparing the acquired image with reference images, calculate the orientation of the vehicles from the generated coordinates, and navigate the vehicles to their final positions. This flow is implemented by using the previously described algorithms.



The program is divided into five main parts, based on these algorithms (see Figure 2):

Figure 2. Flowchart for the Vision Navigation Program

RF Communication

The signals from the CIBCU are received by the transmission hardware and send navigation signals to the UGVs for navigation. A Parallax Basic Stamp (BS2SX) microcontroller in the transmission hardware is programmed to receive a signal from the CIBCU and transmit corresponding navigation signals. Transmission hardware is connected to the controller's serial port to receive serial commands, as well as being connected to the RF transmitter to transmit radio signals. The BS2SX microcontroller is capable of receiving and sending serial data.

The BS2SX in the transmission hardware is programmed to receive a unique command and transmit a corresponding unique set of navigation signals. As the BS2SX receives a command from the serial port, it analyses the command using an "if else" loop. Each UGV has a predefined set of signals for navigation of each UGV and some universal signals, which provide specific navigation control for every UGV. Universal signals are in the range of 68 – 99, with 66 being the universal stop command. Table 1 gives the range of signals for each UGV.

UGV Number	Signal Range
1	11 – 15
2	21 – 25
3	31 – 35
4	41 – 45
5	51 – 55

Table 1. Signal Ranges for Each UGV

The BS2SX program receives a command through a serial input (SERIN) in the form of numbers. The number received is assigned to a variable, which is first checked to see if it falls in the range of universal signals. If the signal falls in the range of universal signals, the corresponding signal is transmitted via the RF transmitter (Figure 9). If the variable is not in the range of universal signals, then it is verified as to which UGV it belongs. Once it falls into a specific UGV range, it then checks for corresponding signals and sends it out. The transmit signal is a combination of three numbers.
The BS2SX receives its commands through the serial port at 9600 baud rates on pin 16, which is connected to the serial port. A PULSOUT signal is sent out to the transmitter to place it in a wake-up state before sending the signal. Then the actual signal is sent out to the transmitter from pin 7. The basic stamp is programmed to receive the signal continuously from the serial port by using a loop.

RF receivers are placed on every UGV to receive the signals. The BS2SX receives the signals through another RF receiver and navigates the UGVs. The receiver on each UGV is connected to a BS2SX, which controls the UGV. All of the UGVs receive all of the signals but only respond to signals assigned to them.

Every UGV has a unique signal for every movement (see Table 2), all of which are preprogrammed on the UGV. The RF receiver on each UGV is always in the wake-up state to receive the signals. It receives every signal at its frequency and sends it to the BS2SX for processing.

UGV -1		UGV -2		UGV -3	
Signal	**Movement**	**Signal**	**Movement**	**Signal**	**Movement**
11	Forward	21	Forward	31	Forward
12	Backward	22	Backward	32	Backward
13	Clockwise	23	Clockwise	33	Clockwise
14	Counterclockwise	24	Counterclockwise	34	Counterclockwise
15	Stop	25	Stop	35	Stop

UGV -4		UGV -5	
Signal	**Movement**	**Signal**	**Movement**
41	Forward	51	Forward
42	Backward	52	Backward
43	Clockwise	53	Clockwise
44	Counterclockwise	54	Counterclockwise
45	Stop	55	Stop

Table 2. Signals Specific to each UGV

Each Boe-Bot UGV (Figure 3) is programmed for five important movement functions: forward, backward, clockwise, counterclockwise, and stop. The BS2SX sends out the pulses to the servo motors according to the signal it receives. Every UGV is programmed to make uniform movements. The movement of the UGV is controlled by the PULSOUT signals to the servos. The servos on every UGV are connected to pins 12 and 13 of the BS2SX.

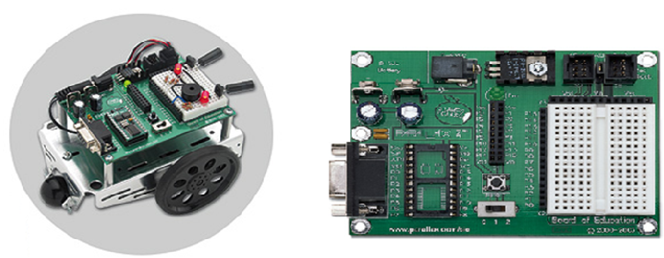
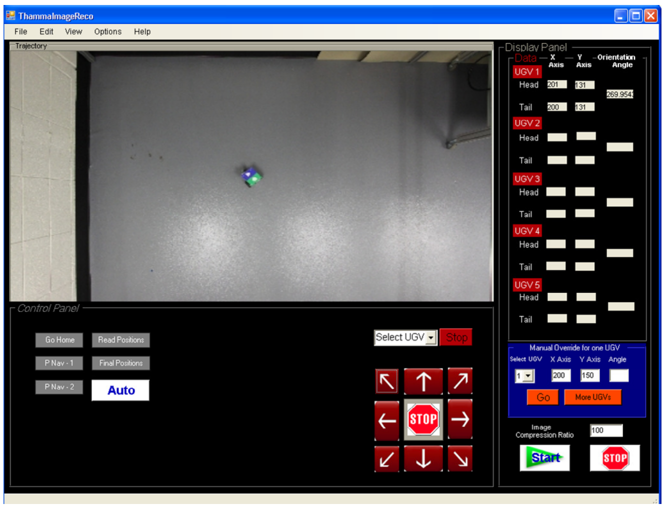


Figure 3. Parallax Boe-Bot and Board of Education

All of the UGVs receive all of the signals transmitted from the transmitter. Once the signal is received by the UGV, the BS2SX on each UGV analyzes the signal it received. If the signal received belongs to that UGV, then it responds with that specific movement.

Centralized Image-Based Controller Unit

The main window of the vision-based navigation system



is shown in Figure 4. This screen shows the image of the bots which move relative to the x, y coordinates and the orientation angles for all of the UGVs.

Figure 4. Snapshot of the Main Screen

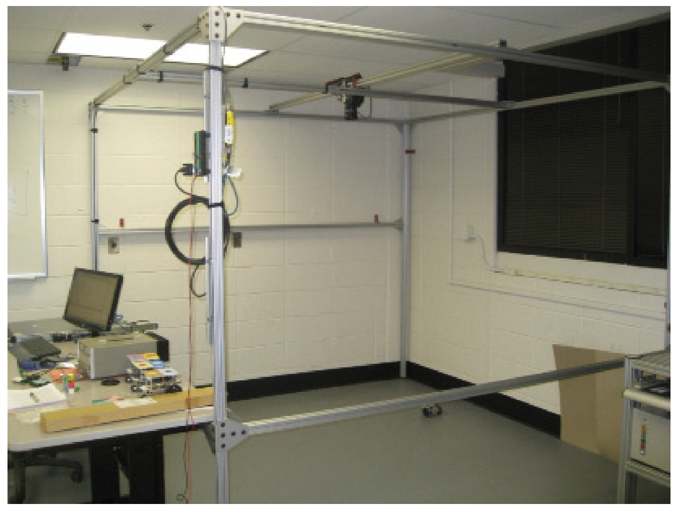
Test Environment

Verification of this vision-based navigation system was accomplished with a test environment set up to navigate the bots according to the images acquired from the vision system. The setup needed a predetermined space in which the bots were to be navigated, a frame to hold the camera at the top center of the predetermined space, a server to run the program and process the images and send the navigation signals, five UGVs with RF receivers, and transmission hardware.

A test environment was set up with all of the requirements for full verification. The space for the UGV navigation was determined and a frame was built covering the predetermined area and a vision system was hung from the top of the frame. A server with high processing power was used to handle the overhead caused by the program. The vision system and transmission hardware were hardwired to the server. All of the UGV's were given unique labels which had the symbols of the head and tail of each vehicle. Each component of the test environment is discussed in detail below.

The UGVs' test space was the predetermined space (see Figure 5) in which the bots would be navigated. The test space was recognized by the frame built to surround the test area. The frame was built using aluminum bars, and was 8 feet in length, 8 feet wide, and 7 feet high (8'x8'x7'). The frame held the vision system at the top of the mid-center of the predetermined area.

A high-quality vision system was needed to provide robust images of the environment. The vision system had to be capable of capturing images at regular intervals and be capable of operating remotely. The vision system also had to cover the total test space and provide high-quality imag-



es. A Canon 50D SLR camera was used as it fulfilled all of the above requirements. It was fitted with a wide-angle lens to cover the test area.

Figure 5. Test Space

The vision system was connected to the server using a USB cable. It was operated by the EOS utility provided by Canon. It could be programmed to capture continuous images with a predefined delay between the images and to save them in a specific location on the server. It could also be operated manually. The transmission hardware was built using a Board of Education (BOE) component carrier board, a BS2SX, and an RF transmitter. The BOE provided the interface for the BS2SX to connect to the serial port and to hold the transmitter module. The BS2SX was programmed to receive the signals from the CIBCU and transmit signals using an RF transmitter. The RF transmitter used in this project was a Parallax 433.92MHz RF transmitter module. This module comes with a transmitter chip, an antenna, and four connection pins. It operates at a baud rate of 12.0k – 19.2k and transmits up to 500 feet.

Testing and Data Analysis

All of the algorithms and programming discussed above were validated by testing in the test environment. At first, synchronization between the different parts of the setup was tested After which the different modes of operation of the program were tested. The vision system was tested for remote shooting by connecting it to the controller and taking test shots. Zoom and focus of the lens were adjusted to cover the whole test area. The vision system can be programmed for different photo formats and image orientation. The vision system was set to save images in the jpeg format with date and time stamps in the specified folder by the user. The vision system was also set to take images in a continuous mode with the image capture interval and number of images being controlled from the remote-control panel.

RF communications of both the transmitter and receiver was tested for functionality. The RF transmission was tested by checking which signal the RF hardware was receiving and which signal it was sending out. The RF receiver was tested by placing a DEBUG code in the receiver program and checking which signals it was receiving. The RF communication was tested by sending signals to the vehicles to move them forward, backward, right, and left. It was tested on all vehicles for continuous signal transmission and transmission range. The synchronization test was conducted to test the synchronization between the controller, vision system, and RF transmission. It was tested by running the program to navigate the vehicles to verify proper performance. It was also tested for all modes of operation and at different speeds of the UGV and camera intervals.

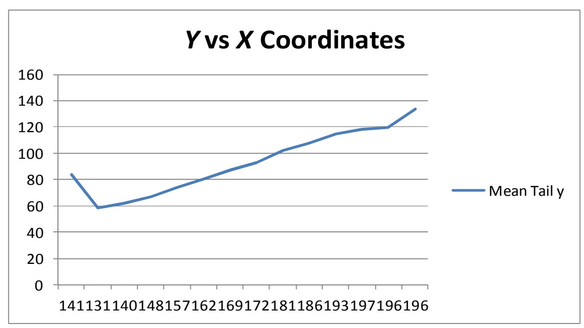
The vision navigation was first tested by the navigation of one vehicle from its present position to a given position. This was done by running the program in the individual-UGV mode. Only one vehicle was placed in the test field and navigated to a final position specified by the coordinates. In this mode, an image was taken and processed and the UGV navigated towards the final position. After every movement in its trajectory, another image was taken and processed to check if the vehicle had deviated from its path. If the vehicle deviated from its original path it was rotated to get it back to its final position. If the vehicle was near the final position within an acceptable tolerance, the vehicle navigation was stopped. The data for every position was recorded automatically in a ".csv" file.

The Vision Navigation test was run continuously ten times by navigating the same UGV between the same start and final positions. All of these data were recorded for subsequent analysis using the save-data option, as seen in Table 3. The x, y coordinates of head and tail orientation of the vehicle for every image were recorded across ten sets of data. The error in the data will be different every time the bots are navigated, thus the mean value was calculated from the ten data sets. These mean data were then plotted to illustrate performance of the system.

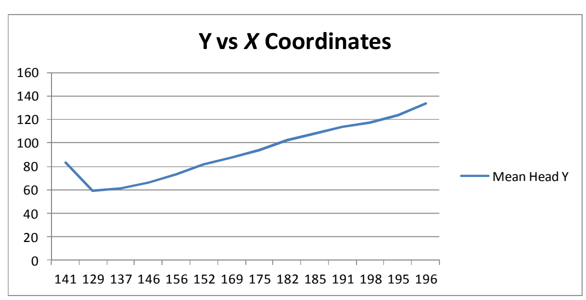
Mean Values				
Tail X	**Tail y**	**Head X**	**Head Y**	**Orientation Angle**
170	108	170	105	261
127	58	122	58	266
136	61	130	60	260
144	66	141	65	253
154	74	150	72	244
160	80	152	80	240
187	105	164	85	242
171	93	170	92	245
178	101	178	101	248
203	127	182	107	252
213	136	188	112	258
191	114	192	115	305
229	154	191	122	245
194	130	194	130	258

Table 3. Mean Values Table of Test Data Recorded

Figure 6 is a plot of the mean tail data showing the relationship between the *x* and *y* coordinates. As can be seen, there was not much deviation from the expected path. Figure 7 is



a plot of the mean head data showing the relationship be-

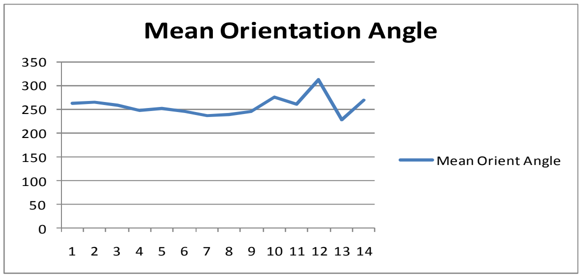


tween the *x* and *y* coordinates. There is not much deviation from the expected path.

Figure 6. *y* versus *x* Coordinates

Figure 7. *y* versus *x* Coordinates

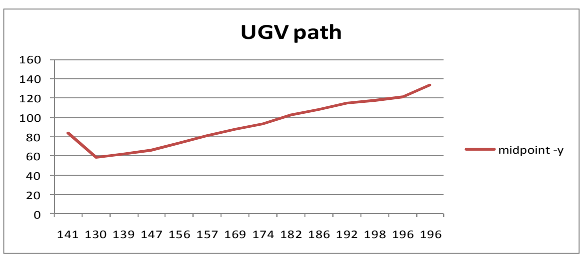
Figure 8 is a plot of the mean orientation data showing the deviation from expected values of the orientation angle. The orientation angle graph should also be a straight line. However, it was observed that the UGVs deviated from their expected path. The deviation was the effect of a change in



the coordinates on heads and tails of the vehicles. Furthermore, the deviation was only in the area indicated in the graphs. This is because most of the deviation in the orientation angle was dependent on both head and tail coordinates.

Figure 8. Deviation in Orientation Angle

Figure 9 is a plot of the data calculated from the mid-



points of the mean data to illustrate the deviation from expected values. This graph gives the actual path of the bot. There is not much deviation as the UGV moved from its present position to its final position.

Figure 9. Deviation in UGV Path

Conclusions

The implementation and testing of this vision-based navigation system allowed for some conclusions to be drawn, which would have a positive effect on the efficiency of the system. Efficient operation of the vision-processing system results in better operation of the navigation system. Thus, vision processing is the most important part of the project. If the coordinates generated by the vision-processing unit are incorrect, all other parts of the system result in error, as all other parts of the system depend on the values of the coordinates generated by the vision-processing unit.

Vision processing depends on the image generated by the vision system and the reference images. Both the image and reference image directly depend on the image labels on the UGVs, therefore the image labels on the bots are the key element for the success of the project. This means more work and time are spent coming up with better image labels that can help in generating exact coordinates of the vehicles.

Recommendations

After working extensively with the reference images, the authors came up with some recommendations for making image labels:

1. All of the images should have a square background with different shapes on them.
2. All of the shapes on the images should be as sharp as possible.
3. Both background and shape on a label should be of different colors.
4. Care should be taken to have different colors on the labels for backgrounds and shapes.

References

[1] Bowditch, N. (1995). Dead Reckoning. In *The American Practical Navigator* (pp. 113-118). Bethesda, MD: National Imagery and Mapping Agency. Retrieved from http://www.irbs.com/bowditch/pdf/chapt07.pdf.

[2] Rathbone, R. R. (2000). Beacon-referenced dead reckoning: a versatile guidance system. *Robotics Engineers,* 11-16.

[3] Thamma, R. (2008). Navigational aids to predict the position of automated guided vehicles with ultrasound and radio frequency sensing. *Proceedings of the IAJC-IJME International Conference,* (pp. 154-165).

[4] Thamma, R. (2009, Winter). Estimating Position and Orientation of an Unmanned Guided Vehicle with Ultrasound and Radio Frequency Sensing. *Technology Interface Journal*. Special Edition, 10(2).

[5] Saeedi, P. D. (2006). Vision-Based 3D trajectory tracking for unknown environments. *IEEE Transactions on Robotics, 22*(1), 119-136.

[6] DeSouza, G., & Kak, A. C. (2002, February). Vision for mobile robot navigation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 24(2), 237-267.

[7] Gasteratos, A., Beltran, C., Metta, G., & Giulio, S. (2002). PRONTO: a system for mobile robot navigation via CAD-model guidance. *Microprocessors and Microsystems,* 26, 17-26.

[8] Jens-Steffen Gutmann, M. F. (2008). 3D perception and environment map generation for humanoid robot navigation. *The International Journal of Robotics Research, 27*(10), 1117-1134.

[9] Fiala, M., & Basu, A. (2004). Robot navigation using panoramic tracking. *Pattern Recognition,* 34, 2195-2215.

[10] Huei-Yung, J. H. L. (2006). A Visual Positioning system for vehicle or mobile robot navigation. *IEEE Transactions on Information & Systems, 89*, 2109-2116.

[11] Guerrero, J. J., & Sagues, C. (2001). Uncalibrated vision based on lines for robot navigation. *Mechatronics, 11*, 759-777.

[12] Kundur, S. R., & Raviv, D (2000). Active vision-based schemes for autonomous navigation tasks. *Pattern Recognition,* 33, 295-308.

[13] Wu, C. J., & Tsai, W. H. (2009). Location estimation for indoor autonomous vehicle navigation by omni-directional vision using circular landmarks on ceilings. *Robotics and Autonomous Systems, 57,* 546-555.

Biographies

RAVINDRA THAMMA is with Department of Manufacturing and Construction Management, Central Connecticut State University, New Britain, CT 06053 USA (Tel: 860 -832-3516; fax: 860-832-1806; (e-mail: thammarav@ccsu.edu).

LEELA MOHAN KESIREDDY is a graduate student at Central Connecticut State University, New Britain, CT 06053 USA (Tel: 860-329-2373; (e-mail: kesireddylem@ccsu.edu).

HAOYU WANG is with Department of Manufacturing and Construction Management, Central Connecticut State University, New Britain, CT 06053 USA (Tel: 860-832-1824; fax: 860-832-1806; (e-mail: wanghao@ccsu.edu).