Road Boundary Detection by a Remote Vehicle Using Radon Transform for Path Map Generation of an Unknown Area

A.S.M Shihavuddin, Kabir Ahmed, Md. Shirajum Munir and Khandakar Rashed Ahmed

Electrical and Electronic Engineering Department, Islamic University of Technology (IUT), OIC, Bangladesh.

Summary
This paper deals with path map generation of an unknown environment by an autonomous vehicle using a proposed trapezoidal approximation of road boundary. Radon transform is used to extract the road boundaries. At first, blind map of the unknown environment is generated in a computer. Image of the unknown environment is captured by the vehicle and sent to the computer using RF transmitter module. The image is pre-processed and the road boundaries are detected using the trapezoidal approximation algorithm and blind map is updated with road positions.
Key words: Edge detection, Sobel filter, Radon transform, map generation.

1. Introduction

This paper is related to the map generation of the roads of an unknown environment and finding shortest path in the between two points using a self guided vehicle. Various methods are proposed in the past for road detection. A single camera with Conic Projection Sensor System has been used in [1], while a two camera system has been used in [2] to get real-time images and the 3-D perspective is removed for vehicle guidance. In [3], Hough Transform has been used to extract 3-D road boundaries in the images coming from a stereo-vision system. Geometrical features are used in [4] to find road boundaries. A multi order line segment description of the road boundary has been achieved through a statistical test, in [5]. In [6], low level structures are extracted via a non-linear transform and it proposes an algorithm for segmentation of images at multiple scales. In [7], a fast edge detection method is proposed. Also different techniques of vehicle guidance system in detected road are proposed in the past [8-13]. In our paper, a different technique is used. We have used a vehicle which is self controlled by detected road parameters. A wireless video camera is installed on the robot which takes the front view video of the robot and transmits it back to a PC. In the PC the video is analyzed to detect the road. At first, the vehicle moves from the source through the unknown environment having approximated boundary. The unknown environment is modeled using a base matrix. Image frames are extracted from the video, converted to grayscale image and noise is reduced by using a 3×3 median filter [14]. This image is converted into binary image. By applying appropriate area threshold, only the road area is kept in the binary image. Then the edge is detected using Sobel method followed by detection of the continuous straight lines and curves of the edges using Hough transform [15]. Then we approximated a trapezium section of the road having defined length. The vehicle then moves through the trapezium and reaches the next approximated trapezium having a tilt angle with the previous one. According to the distances moved by the vehicle and corresponding tilt angles, the base matrix is updated. As the vehicle explores all possible roads of the unknown environment, the complete map of the unknown environment is finally obtained from the base matrix.

To avoid collision with any obstacle of the environment, an IR based obstacle detector is used [20]. As the vehicle can move in the environment without any collision, it can generate the complete map of the environment for subsequent shortest path estimation. For this obstacle detection system, Infra-red light is emitted from the IR-LED and if there is any obstacle nearby then the light will be reflected and picking up the reflected signal, an IR-light detector module will relay the signal to the microcontroller to take the decision to control the vehicle movement.

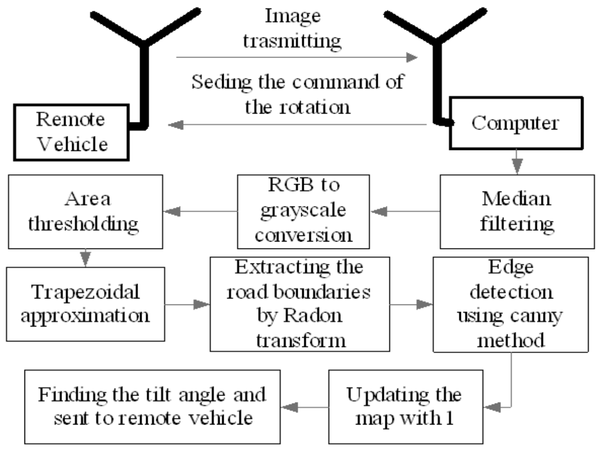


Fig. 1. Block diagram representation of proposed system

Manuscript received August 5, 2008.
Manuscript revised August 20, 2008.

2. Data Set

We first of all make the analysis of all the data concerned with the vehicle like its motion, dimension, decision and detected road through which it is moving. We need following two data matrixes to represent all those data,

1. Blind matrix
2. Progression matrix

The constant terms used throughout this paper are as followed,

The units which will be constant throughout the paper is, *UNIT = 1cm*. The environment will be represented by four states - *SOURCE = "*"*, *DESTINATION = "#"*, *UNKNOWN = "-"* and *ROAD = "1"*.

2.1 Base Matrix

Every point of the Environment where this vehicle will work is represented as Base matrix set in terms of X and Y Coordinates. Initially all the points of Base Set will be set to "-" except "*" and "#". As the vehicle moves through the road and generates Progression Set, those data will be converted into X and Y coordinates values starting from "*" and each detected road coordinate will be marked as "1" in the Blind set. When the robot will proceed through the detected path, the rotation of the stepper motor of the driving wheel will give the distance moved. As it will face a division of the road, it will go through one path and will save another uninvestigated path position to detect on next stage. From the rotation of the stepper motor of the steering, the rotation angle of the robot will be found hence the accurate road position which will be used to update corresponding blind matrix position with "1".

2.2 Progression Matrix

Initially the vehicle will move from the source to destination by detecting the unknown road. Each time, a segment of the road having length *l* will be selected through which the vehicle will move and will enter into the next segment having a tilt angle with the previous segment. Every tilt angle is stored in Progression matrix.

3. Map Generation

Path map of an unknown environment is generated by the vehicle as it moves along the road with and creates a database of the motion. The movement has the memory of order 1, i.e. the next step of the decision is based upon the last one. The path is assumed to be consisting of very small linear approximated segment of length *l*. The vehicle advances *l* which is calculated from the image sequence taken at every decision moment 't_i'.

With edge detection algorithms even road edges can be extracted from the image of unknown environment. However, it is difficult to do so when the road surface is not homogeneous. These inhomogeneities appear as noise while finding the edges. In order to eliminate them, we ran a Median filtering [21] on the road image. Let the image be represented as $I(x, y)$, with x and y being the two coordinate directions. Median filter can be represented as,

$$v(m,n) = median\{I(m-\text{k}, n-1), (\text{k},1) \in W\} \quad (1)$$

Where, W is a suitably chosen window.

The median filtered image is then converted into binary image where only the road, plants having significant color discontinuity and sky areas are detected. The camera is positioned in such way on the vehicle that the area of the road is greatest on the binary image. Using the road area as threshold, we eliminated all other smaller detected areas as,

$$\sum I_{ROAD}(x, y) > \sum I_{SKY}(x, y), \sum I_{PLANT}(x, y) \quad (2)$$

Sobel filter [22], are used for boundary extraction. Performances of both the filters are almost similar. The output image after boundary extraction is represented as:

$$E(x, y) = I(x, y) \times h(x, y) \quad (3)$$

Where, $E(x, y)$ is the image data containing extracted edge, $I(x, y)$ is the area thresholded binary image data, $h(x, y)$ is the filter matrix.

To apply our proposed trapezium approximation method for road segmentation, we need straight road boundaries. Analytical detection of parameterized geometrical primitives like lines, circles and ellipses, can be done by Radon transform[23] [24]. The Radon transform of a function f(x,y) is defined as the integral along a straight line defined by its distance from the origin and its angle of inclination , a definition very close to that of the Hough transform. Suppose a 2-D function $f(x, y)$ (Fig. 2). Integrating along the line, whose normal vector is in θ direction, results in the $g(s,\theta)$ function which is the projection of the 2D function $f(x, y)$ on the axis s of θ direction. When s is zero, the g function has the value $g(0,\theta)$ which is obtained by the integration along the line passing the origin of (x, y)-coordinate. The points on the line whose normal vector is in θ direction and passes the origin of (x, y) -coordinate satisfy the equation:

$$\frac{y}{x} = \tan(\theta + \frac{\pi}{2}) = \frac{-\cos\theta}{\sin\theta} \quad (4)$$

$$\text{Or, } x\cos\theta + y\sin\theta = 0 \quad (5)$$

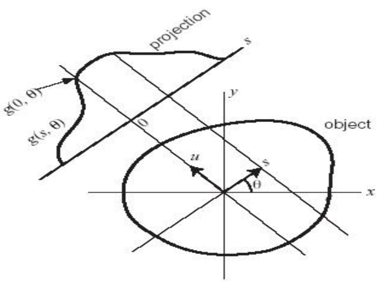


Fig. 2 The Radon Transform computation.

The integration along the line whose normal vector is in θ direction and that passes the origin of (x, y) -coordinate means the integration of $f(x, y)$ only at the points satisfying the previous equation. With the help of the Dirac function - δ , which is zero for every argument except to 0 and its integral is one, $g(0,\theta)$ is expressed as:

$$g(0,\theta) = \iint f(x, y) \cdot \delta(x\cos\theta + y\sin\theta)\, dxdy \quad (6)$$

Similarly, the line with normal vector in θ direction and distance *s* from the origin is satisfying the following equation:

$$(x - s\cdot\cos\theta)\cdot\cos\theta + (y - s\cdot\sin\theta)\cdot\sin\theta = 0 \quad (7)$$

$$\text{or, } x\cos\theta + y\sin\theta - s = 0 \quad (8)$$

So the general equation of the Radon transformation is acquired:

$$g(s,\theta) = \iint f(x, y)\cdot\delta(x\cos\theta + y\sin\theta - s)\, dxdy \quad (9)$$

Like in the Hough transform, the Radon operator maps the spatial domain (x,y) to the projection domain (s,θ), in which each point corresponds to a straight line in the spatial domain. Using this property, we used Radon transform for detecting straight lines describing the road boundaries in the edge detected binary image. To move along the detected road, rotation angle is what we required. For this, we have taken a precise portion from the processed image of particular length *l*. Road view is then approximated as trapezium.

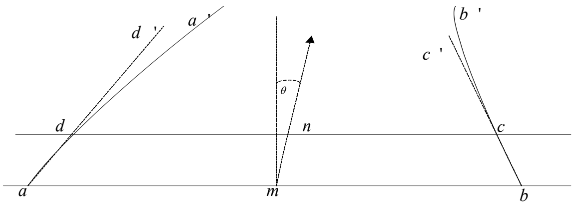


Fig. 3. Linear straight line approximation of road boundaries.

In the Fig. 3, *ada'* and *bcb'* are the road boundaries. The precise portion we take for the consideration is the trapezium *abcd*. The direction vector for the motion of the robot is given by the vector indicated by *mn* and the angle to be rotate is the angle between *mn* and the normal drawn at the point *'m'*. With the constant distance *l* and angle, the vehicle will be guided through the path. And according to the movement, the corresponding map will be generated in the remote computer.

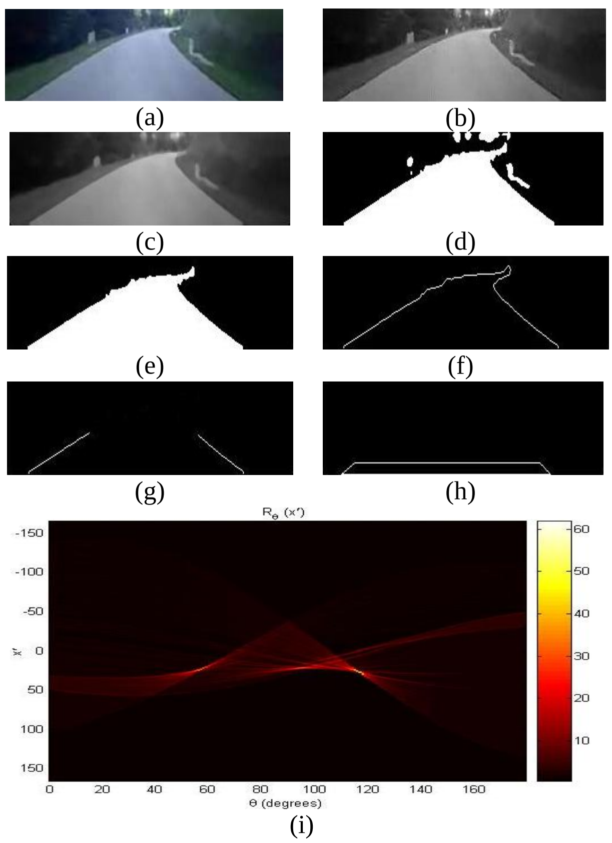


Fig. 4. (a) Original image, (b) Gray scale image, (c) Median filtered image, (d) Binary image, (e) Area thresholded image, (f) Edge detected image, (g) Detected straight line by Radon transform, (h) Trapezoidal approximation, (i) Radon transformation space for straight line detection.

When the trapezium is available from the road image, we can extract the decision parameter for the vehicle movement according to the following geometrical analysis:

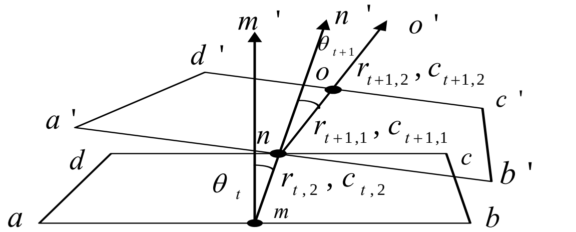


Fig. 5. Calculation of the decision parameter for the vehicle movement.

At Fig. 5. two consecutive road frame are placed together to make the analysis. The trapezium *abcd* represent the the road segment at time '*t*' and *a'b'c'd'* represents for time '*t+1*'.'r' and 'c' denotes the row and column number of the image matrix. For *t* time frame the angular motion towards mnn' can be derived from the *'m'* and *'n'* pixels row and column position $(r_{t,1},c_{t,1})$ and $(r_{t,2}, c_{t,2})$ according to below relation

$$\theta_t = 90 - \tan^{-1}\left(\frac{r_{t,2} - r_{t,1}}{c_{t,2} - c_{t,1}}\right) \quad (10)$$

Where, θ_t is the angle of the rotation. As the length to be advance is already predefined and fixed so no further calculation is required for the wheel motion of the vehicle.
Similarly after the completion of the auto generated command for the time frame '*t*', the next road frame for the decision moment of *t+1* will be like the trapezium *a'b'c'd'* and the angle to be rotated will be equal to

$$\theta_{t+1} = 90 - \tan^{-1}\left(\frac{r_{t+1,2} - r_{t+1,1}}{c_{t+1,2} - c_{t+1,1}}\right) \quad (11)$$

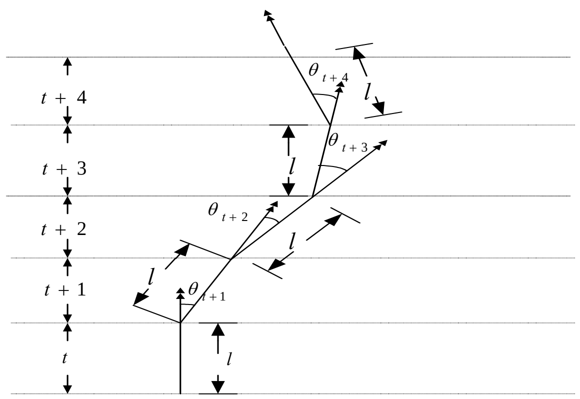


Fig. 6. Calculation of the value of Progression matrix.

At every decision moment we have the angle at which a straight line of length *l* will tilted. Along this line of movement, the map initiated in the base matrix will be updated with "1" values.
Say the current position of the robot in XY coordinate is (x_i,y_i) and we assume it to be in (x_{i+1},y_{i+1}) position after the current movement. So to generate the map we need to find the values of (x_{i+1},y_{i+1}) while the (x_i,y_i) is already known. And for these two unknown variables we can use the following two equations.

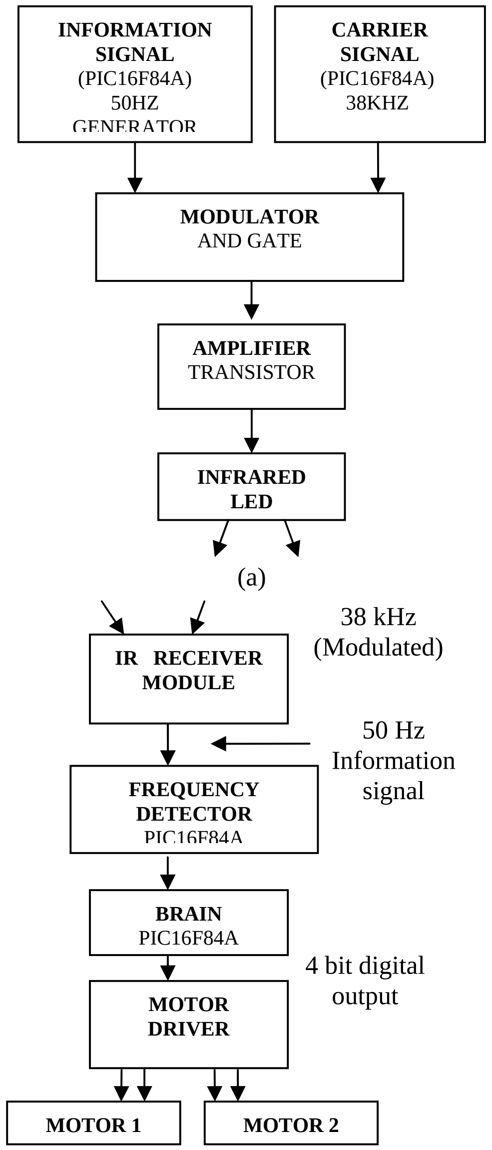
$$\sqrt{((x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} = l \quad (12)$$

$$\frac{y_{i+1} - y_i}{x_{i+1} - x_i} = \tan(90 - \theta_i) \quad (13)$$

Along the line bounded by (x_i,y_i) and (x_{i+1},y_{i+1}) the corresponding pixels of the base matrix will be updated as 1. So while the robot reaches to its destination it will be automatically leaving its history of movement which is basically the path along which it explored.
This generated map of the unknown environment ultimately would be useful for later purposes like finding the shortest path.

4. Obstacle Detection

In the obstacle detection system, one microcontroller is used to generate a 50 Hz pulse signal that is to be sent by the Infra-red LED. Another microcontroller was programmed to generate 38 kHz carrier signal. These two signals were modulated using an AND gate and then amplified using an 8050 transistor and finally sent through an IR-LED. If there is any obstacle then the signal will reflect back to the receiving module of the sensor. At the receiving side, the IR receiver module picks up the 38 kHz modulated signal and then demodulates it, to give the original 50 Hz signal to another microcontroller. This microcontroller then matches the received signal with the sent signal. If the microcontroller can match the received signal with the sent signal then it takes the decision that there is an obstacle. Then necessary decisions are taken and control signal is sent to the motor driver and which moves the vehicle in such a way that the obstacle is avoided. In the obstacle detection system, two sensors are used for the left side and right side of the vehicle. If any obstacle is detected at the left side, then the robot will move to the right and if the obstacle is at right then the robot will move to the left. And if both the sensors detect obstacles then the robot will move backward.
The block diagram of the obstacle detection system is shown bellow,

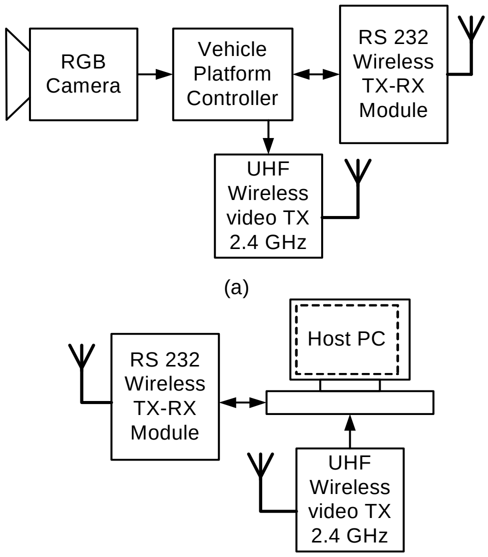


(b)

Fig. 7 Block diagram of the obstacle detection system- (a) Transmitter unit, (b) Receiver unit.

5. Hardware Implementation

The project was implemented on a mobile robot vehicle platform called MORPHEOUS [20]. This platform was driven by two stepper motors 12V 0.12Kwatt each. The microcontroller used is PIC 16F84A. A CCD camera was mounted on the mobile robot platform, and an analog output was connected to the UHF Audio/Video Transmitter unit. While the RS232 RF receiver module connected to a PIC 16F84A I/O ports was used as remote controller.



(b)

Fig. 8. Block diagram of (a) Vehicle hardware (b) PC hardware

The wireless video system used a 2.4 GHz carrier frequency and does not interfere with the RS232 RF data module transmission due to different carrier frequencies. Analog video images from the receiver unit then connected to the USB video frame grabber. After the images were received into the computer, map is generated and shortest path position is sent via RS232 serial port COM1 to an RS232 RF transmitter unit as a command to the vehicle.

6. Result

Initially the boundary of the unknown environment, source position and destination is specified to generate the blind matrix. Then the vehicle will start exploring all the connected roads and the map will be generated accordingly by the proposed trapezoidal approximation method. A typical generated map using MORPHEOUS robot vehicle platform is shown bellow,

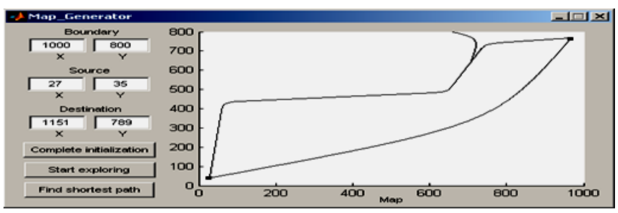


Fig. 7. Generated map of the unknown environment.

7. Conclusion

In this paper we have applied Radon transform for road boundary detection which is used later on to generate the map of an unknown area. As the vehicle initially explores all possible road branches, so the complete map of the environment is generated. Later the tilt angle of the road is calculated for the vehicle to move. During the whole process, the vehicle can operate independently avoiding all obstacles and practically it is implemented using MORPHEOUS mobile robot test platform. Practically, the proposed algorithm generates map with 90% accurate road position. An experimental result is also shown in the result section of this paper.

References

[1] Z. Zhu, S. Yang, G. Xu, X. Lin, D. Shi, "Fast Road Classification and Orientation Estimation Using Omni- View Images and Neural Networks," *IEEE Trans. On Image Processing*, vol. 7, no. 8, pp 1182-1197, Aug. 1998.

[2] M. Bertozzi, and A. Broggi, "GOLD: A parallel realtime stereo vision system for generic obstacle and lane detection," *IEEE Trans. on Image Processing*, vol. 7, no. 1, pp. 62-81, Jan. 1998.

[3] U. K. Sharma, and L. S. Davis, "Road boundary detection in range imagery for an autonomous robot," *IEEE Journal of Robotics and Automation*, vol. 4, no. 5, pp. 515-523, Oct. 1988.

[4] J. P. Tarel, and F. Guichard, "Combined dynamic tracking and recognition of curves with application to road detection," *Procc. of IEEE Int. Conf. on Image Processing*, Sept. 2000, vol. 1, pp. 216-219.

[5] G. Tang, X. Liu, X. Liu, H. Yang, "A road boundary detection method for autonomous land vehicle," *Procc. of the 4th World Congress on Intelligent Control and Automation*, June 2002, Shanghai, vol. 4, pp. 2949- 2951.

[6] M. Tabb, and N. Ahuja, "Multiscale Image Segmentation by Integrated Edge and Region Detection," *IEEE Trans. on Image Processing*, vol. 6, no. 5, pp. 642-655, May 1997.

[7] Aurobinda Routray, Kanungo Barada Mohanty. "A Fast Edge Detection Algorithm for Road Boundary Extraction Under Nonuniform Light Condition". Procc. of 10th International Conference on Information. December 17-20, 2007, Rourkela, pp. 38-40

[8] Davis, L. and Kushner, T. Road boundary detection for autonomous vehicle navigation. SPIE Vol. 579[Intelligent Robots and Computer Vision]. 1985.

[9] Hong, T., Abrams, M., Chang, T., and Shneier, M. O., "An Intelligent World Model for Autonomous Off-Road Driving," *Computer Vision and Image Understanding*, 2000.

[10] Rasmussen, C. "Combining Laser Range, Color, and Texture Cues for Autonomous Road Following." Proceedings of the International Conference on Robotics and Automation. 2002.

[11] R. Gregor, M. L¨utzeler, and E. D. Dickmanns, *EMS-Vision: Combining on- and off-road driving.* Proc. SPIE Conf. on Unmanned Ground Vehicle Technology III, AeroSense 2001, Orlando, FL, USA, April 16-17 2001.

[12] D. Lieb, A. Lookingbill, S. Thrun: *Adaptive Road Following using Self- Supervised Learning and Reverse Optical Flow,* Proc. Robotics Science and Systems, Cambridge, MA, USA, June 8-11 2005.

[13] D. Stavens and S. Thrun: *A Self-Supervised Terrain Roughness Estimator for Off-Road Autonomous Driving,* In Proc. Conference on Uncertainty in AI (UAI), Cambridge, MA, USA, July 13-16 2006.

[14] H. Hwang, R. A. Haddad, "Adaptive Median Filters: New Algorithms and Results," *IEEE Trans. on ImageProcessing*, vol. 4, no. 4, pp. 499-502, April 1995.

[15] Duda R.O. – Hart E.P. (1972): Use of the Hough Transformation to detect lines and curves in images. *Grapics and Image Processing. 3, pp. 11-15.*

[16] Giorgio Ausiello , Giuseppe F. Italiano , Alberto Marchetti Spaccamela , Umberto Nanni, Incremental algorithms for minimal length paths, Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms, p.12-21, January 22-24, 1990, San Francisco, California, United States.

[17] Ning Jing , Yun-Wu Huang , Elke A. Rundensteiner, Hierarchical optimization of optimal path finding for transportation applications, Proceedings of the fifth international conference on Information and knowledge management, p.261-268, November 12-16, 1996, Rockville, Maryland, United States

[18] Michelle R. Hribar, Valerie E. Taylor. Termination detection for parallel shortest path algorithms". December 1998, Journal of Parallel and Distributed Computing, Volume 55 Issue 2

[19] M.R. Hribar, V.E. Taylor, D.E. Boyce, Choosing a shortest path algorithm, Technical Report CSE-95-004, Computer Science and Engineering, EECS Department, Northern University, 1995.

[20] A.S.M.Shihavuddin; Numair MD Fadlullah; Kazi Khairul Islam. Micro-controller operated robot using photo sensor guiding system, 1st international conference on control, instrumentation and mechatronics, CIM '07, Johor Bahru, Johor, Malaysia, 28th - 29th May 2007.

[21] H. Hwang, R. A. Haddad, "Adaptive Median Filters: New Algorithms and Results," *IEEE Trans. on Image Processing*, vol. 4, no. 4, pp. 499-502, April 1995.

[22] J. Scharcanski, and A. N. Venetsanopoulos, "Edge Detection of Color Images Using Directional Operators," IEEE Trans on Circuits and Systems for Video Technology, vol. 7, no. 2, pp. 397-401, April 1997.

[23] E. Magli, L. Lo Presti, and G. Olmo, "A pattern detection and compression algorithm based on the joint wavelet and Radon transform," in Proc. IEEE 13th Int. Conf. Digital Signal Processing, 1997, vol. 2, pp. 559-562.

[24] A.L. Warrick and P.A. Delaney, "Detection of linear features using a localized Radon transform with a wavelet filter," in Proc. ICASSP, 1997, vol. 4, pp. 2769-2772.