Local Information using Stereo Camera in Artificial Potential Field based Path Planning

Hendri Himawan Triharminto, *Member, IAENG,* Oyas Wahyunggoro, Teguh Bharata Adji, Adha Cahyadi, Igi Ardiyanto, and Iswanto

***Abstract*—Generally, a robot acquires environment data from the sensor that is attached to the robot itself and merely obtains local information. Artificial Potential Field (APF) is designed as the path planning with global information. Therefore, local information becomes one of the issues in the APF based path planning. This paper proposes an approach to handle the local information in the APF using framework transformation. With integration of image processing, clustering, and framework transformation, the initial, goal, and obstacles from the real world coordinate can be determined in the APF environment scenario. Transformation of the two-dimensional image is used to generate the APF path planning. The local optima in the local information is set as waypoint for the global optimum in whole environment scenario. In order to test performance of the algorithm, local data set is used. Two scenarios are used in this research, i.e. static environment and dynamic environment with a moving obstacle. The results show that the proposed method can be applied in the real time implementation.**

***Index Terms*—APF, local information, framework transformation.**

I. Introduction

Many applications for either civilian or military purposes used autonomous robots. There are some abilities that have to be fulfilled to warrant an autonomous system in the real implementation. Localization, path planning, and mapping are the abilities that should be have in the autonomous robot system [1]. The hierarchy process of the autonomous robot can be depicted as in Figure 1.

Figure 1 shows that the path planning level is higher than the controller in the autonomous robot system. The path planning is a compulsory component which aims to guide the controller to reach the mission planning and task allocation [2]. Therefore, path planning takes important roles regarding autonomous robot system. In order to compensate the control system limitation, the objective of the path planning is to find the feasible path starting from the initial to goal position which the robot is able to accomplish the task allocation. The connecting lines mean data acquisition gives the environment data to construct path planning. After path planning generation, the controller drives the robot through the waypoints. The process continued until the robot reaches the objective.

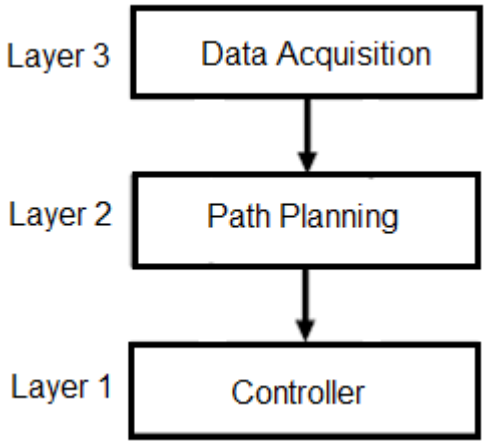

Fig. 1. Hierarchy Process of An Autonomous Robot

Many algorithms were introduced to construct path planning. Some conventional algorithms generate decomposed region such as Heuristic A$*$ and Dijkstra algorithm. Heuristic A$*$ had been proven to be easy and powerful algorithm when using in the path planning field [3]. Similar to A$*$, Dijkstra algorithm was also used as the path planning algorithm [4].

One of the well-known approach of path planning algorithm is based on Artificial Intelligence (AI) algorithm [5]. Genetic Algorithm (GA) [6] [7] [8], Particle Swarm Optimization (PSO) [9] [10] [11], and Ant Colony Optimization (ACO) [12] [13] are the examples of AI algorithm that used as path planning algorithm. The approaches define the chromosome, particle, and ant as candidates of the possible solution. The newest approach based on the natural science such as bee colony was also used in the path planning area [14]. The fitness function is set with some parameters related the feasible path. The feasible path means that the path is the shortest distance but without neglecting kinematic constraint as the robot boundary and safety factor between the robot against the environment. Using the iterative method of optimization, the path will be produced by AI-based algorithm considering the feasible parameters.

The other approach such as RRT (Random Rapidly Tree) was used for robot maneuver in the path planning system [15]. In the research of Yang et. al. [16], RRT was used for the generation of a collision-free piecewise linear path while a path smoothing algorithm was applied which satisfies curvature continuity and non-holonomic constraints.

The curve algorithm is a path planner approach using polynomial equation to build a certain path geometry [17], [18].

Manuscript received February 2, 2017; revised June 22, 2017. This research was supported by PUPT Project grant from DIKTI through Research Directorate, Universitas Gadjah Mada with the contract number: 2368/UN1.P.III/DIT-LIT/LT/2017, awarded to A/P Dr. Oyas Wahyunggoro.

Hendri Himawan Triharminto is with Electrical Engineering and Information Technology Department, Universitas Gadjah Mada, and Electrical Department, Indonesian Airforce Academy, Yogyakarta, Indonesia, e-mail: kanghimawan@gmail.com.

Iswanto is with Electrical Engineering and Information Technology Department, Universitas Gadjah Mada, and Department Electrical Engineering, Universitas Muhammadiyah Yogyakarta.

Oyas Wahyunggoro, Teguh Bharata Adji, Adha Cahyadi, and Igi Ardiyanto are with Electrical Engineering and Information Technology Department, Universitas Gadjah Mada, Yogyakarta, Indonesia.

Generally, the curve algorithm is applied in the static environment. The dynamic curve path algorithm was introduced by [19]. The curve algorithm considers two constraints, i.e. maximum curvature and maximum torsion bound. Based on the constraints, the path is constructed using geometry approach.

All of the path planning approaches that were aforementioned, generally, can solve path planning problem which meets kinematic constraint, but most of them are not designed to become a reactive obstacle avoidance. The conventional and AI-based algorithm have to rearrange and replanning using the iterative method. In spite of rearranging and re-planning, the curve based algorithm has to add one or more points to avoid an obstacle.

The APF is one of the path planning approaches which is based on the idea in term of electric field [20]. There are two important characteristics of force in the APF, i.e. repulsive and attractive force. In the path planning case, the repulsive force is used for obstacle avoidance and imaginary held on the obstacle. The goal takes advantage from the attractive force in the potential field. From the characteristic, the APF has an advantage in the real-time obstacle avoidance despite its weakness in the information acquisition. Environment data is global view which the robot must have a prior knowledge. Therefore, this research focuses on how to handle the local information.

This paper is organized as follows. The second section explains the problem formulation in the APF regarding the local information. The third section describes the concept and idea how to solve the problem of local information in the APF based path planning. The last two sections deliver the results, discussion, and conclusion.

II. Problem Description

The APF follows the natural characteristic of electrostatic potential [21]. The concept of electric field motion is applied to the robot initial position and goal position. The goal position becomes the lowest potential while the initial position is the representative of the highest potential. From nature of the potential field, the potential energy will move from the highest to the lowest. The obstacle set as opposite direction force due to unsafe area of the robot that coined as the repulsive force.

The attractive forces of APF can be modeled as

$$f_a(x,y) = \nabla V_a(x,y) \tag{1}$$

and the forces in a certain position (x, y) are

$$f_{xa}(x,y) = \frac{\partial V_a(x,y)}{\partial x}, f_{ya}(x,y) = \frac{\partial V_a(x,y)}{\partial y}, \tag{2}$$

where

$$V_a(x,y) = \frac{1}{2} K_a[(x-x_T)^2 + (y-y_T)^2]. \tag{3}$$

Variable K_a is the attractive gain parameter and (x_T, y_T) is the goal position.

The potential surface in a single point of the obstacle is

$$V_o(x,y) = \frac{K_o}{\sqrt{(x-x_o)^2 + (y-y_o)^2}}, \tag{4}$$

where K_o is the repulsive gain parameter and (x_o, y_o) is the point of the obstacle position. From the (4), the repulsive force of APF in certain position of the obstacle (x, y) can be formulated as

$$f_{xo}(x,y) = \frac{\partial V_o(x,y)}{\partial x}, f_{yo}(x,y) = \frac{\partial V_o(x,y)}{\partial y}. \tag{5}$$

If the obstacle has more than one point, then the formula becomes

$$V_o(x,y) = \sum_{i=1}^{j} V_{oi}(x_i, y_i). \tag{6}$$

Variable i is the number of the obstacle's point from $i = 1$ to j, where j is the total number of the obstacles.

The total of the force field can be determined as

$$f(x,y) = f_a(x,y) + f_o(x,y). \tag{7}$$

Consequently, it can be concluded that the total force of APF is the sum of attractive and repulsive force. The example of total forces in the APF can be depicted in Figure 2.

The variable x_T, y_T, and $V_o(x, y)$ as in the (3) and (4) and Figure 2, are known variables in the APF scenario. It means that the prior knowledge of the information becomes the natural characteristic of the APF. However, local information without prior knowledge of the environment is difficult to solve. Thus, one of the main problems in the APF based path planning is to handle local information.

III. The Proposed Method

In this research, the data acquisition process uses a stereo camera to capture the images. The stereo camera obtains depth or distance from the environment. After the distance is obtained, image processing is utilized to manipulate and create the environment scenario for the APF. Result of the image processing is then transformed into a new frame. The transformation yields the position of the obstacle and goal in the new environment scenario. Therefore, the APF can be implemented directly as real-time obstacle avoidance in the path planning using local information. The flowchart of the algorithm can be depicted as in Figure 3.

Figure 3 shows that the feature description uses SURF (Speeded Up Robust Features) [22]. The approximation with respect to repeatability, distinctiveness, and robustness, yet can be computed and compared much faster is consideration of SURF application.

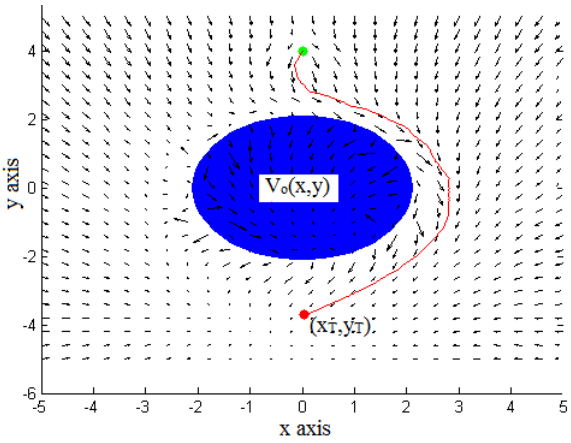

Fig. 2. Example of the Total Force in the APF

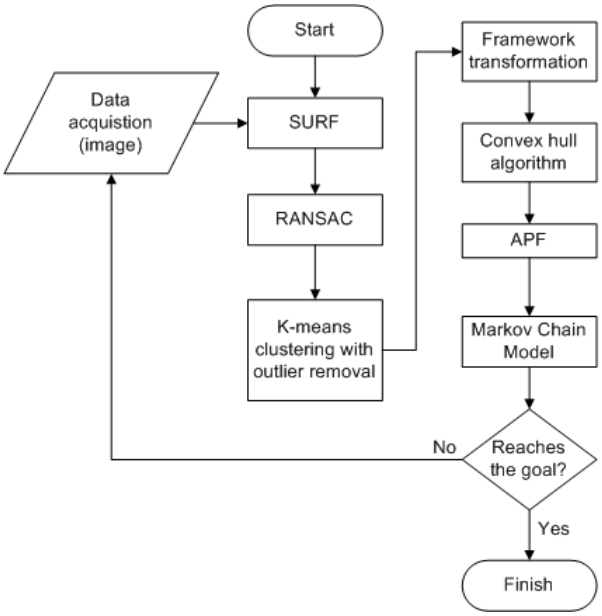

Fig. 3. Algorithm Process of the Proposed Method

A. SURF

SURF is a feature descriptor that uses a Hessian matrix-based measure for the detector. Given a point $x = (x, y)$ in an image I, the Hessian matrix $H(x, \sigma)$ in x at scale σ is defined [22] as

$$H(x,\sigma) = \begin{vmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{vmatrix}, \tag{8}$$

where

$$L_{xx}(x,\sigma) = I(x) * \frac{\partial^2}{\partial x^2} g(\sigma) \tag{9}$$

$$L_{xy}(x,\sigma) = I(x) * \frac{\partial^2}{\partial xy} g(\sigma). \tag{10}$$

In (9), $L_{xx}(x, \sigma)$ is the convolution of the Gaussian second order derivative with the image I in point x, and similarly for $L_{xy}(x, \sigma)$ and $L_{yy}(x, \sigma)$. The relative weights in the expression for the Hessians determinant with

$$\frac{|L_{xy}(1.2)|_F |D_{xx}(9)|_F}{|L_{xx}(1.2)|_F |D_{xy}(9)|_F} = 0.912 \simeq 0.9, \tag{11}$$

where $|x|_F$ is the Frobenius norm[1]. Thus,

$$det(H_{approx}) = D_{xx} D_{yy} (0.9 D_{xy})^2. \tag{12}$$

By the Hessian matrix, the eigen vectors show the direction of curve (gradient) of the image. The SURF descriptor is based on Haar wavelet responses to assess the primary direction of the feature [23]. The direction is determined by the change of intensity.

The result of feature detection is then have to match for both cameras. In order to match the feature of both cameras, Random Sample Consensus (RANSAC) is employed.

[1]Frobenius norm is described as the absolute summation of the diagonal of a matrix

B. RANSAC

RANSAC is a sampling technique to estimate the model parameters by using minimum number of observations [24]. By defining sample of random points from both camera images, the features can be matched by considering the distance of the point. The distance and standard deviation are as the threshold for inlier. Consequently, the probability of the outlier is

$$v = 1 - q, \tag{13}$$

where q represent the probability that any selected data point is an inlier. A confidence number p is used to set a thresholds for acceptance of the points as intermediate results of the solution. It means that the sets of random samples have a very small probability as an outlier. The confidence number p has relation that

$$1 - p = 1 - (1 - v^m)^N, \tag{14}$$

where m is the minimum number of the points or size of the samples, and N is the number of iteration that can be computed as

$$N = \frac{log(1-p)}{log(1-(1-o)^m)}. \tag{15}$$

The termination of the algorithm is based upon the expected number of trials N required to select a subset of m good data points.

In this research, RANSAC is used to remove outlier in the feature matching of the SURF approach. Figure 4 is the application of RANSAC that reduces the outlier in the feature matching. Consequently, it will eliminate the features itself. The main point of RANSAC in this research is to obtain the feature matching of two cameras thus it can measure the depth or distance of each feature.

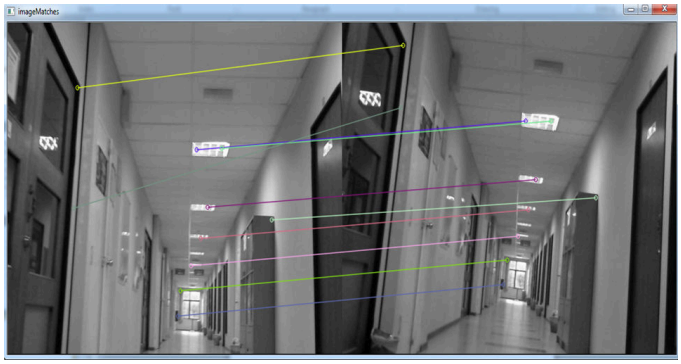

Fig. 4. The Final Result of SURF+RANSAC

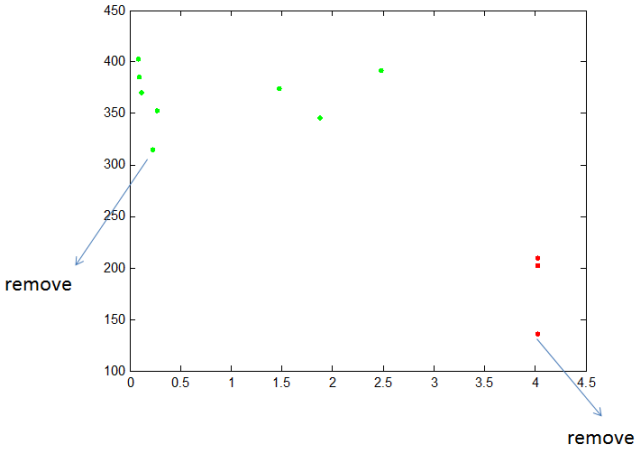

Fig. 5. Example of K-means Clustering with Outlier Removal

C. K-means Clustering

The next stage of proposed method is K-means clustering with outlier removal [27]. This stage is based on the observation that the environment density of an object is greater than the environment that is not an object. The mathematical model can be described as

$$\sigma_{object} \geq \sigma_{\sim object}, \tag{16}$$

where σ_{object} is density of the environment of an object while in contrast, $\sigma_{\sim object}$ is density of the environment that is not an object.

Therefore, the strategy of this stage makes a grouping of the feature detections into each cluster and defining the rules when a particular point is more than outlier threshold. The strategy is divided into two steps. First, every feature must be grouped into each cluster. Secondly, based on (16), the point that has a distance less than outlier threshold is determined as part of the object. It means that if distance of the points is larger than the outlier threshold then it can be neglected since the point is not part of an object.

The clustering stage performs K-means algorithm until it meets convergence. Division of the cluster depends on the distance from a point x_i to the centroid (c) which is modeled as

$$D = \sum_{i=1}^{n} \sqrt{(x_i - c_k)^2}. \tag{17}$$

The update of the centroid of k is as follows

$$c_k = \frac{1}{N_i} \sum_{i=1}^{N_i} x_i \ , \tag{18}$$

where N_i is total data on each cluster.

The second step is to assign an outlyingness factor. The factor depends on the distance from the cluster centroid. The formula of the outlyingness factor is described as

$$Outlier = \frac{||x_i - c_k||}{D_{max}}, \ i = 1, ..., N \tag{19}$$

and

$$D_{max} = max||x_i - c_k||, \ i = 1, ..., N. \tag{20}$$

The outlyingness factor is normalized within scale [0,1]. In this research, the total number of clustering is two clusters. The total number is to meet goal point convergent in local information. If there are more than two obstacles, then the goal point can be obtained more than one as well. Since the APF merely needs one goal point as optimum solution, then it is difficult to determine the optimum solution in local information.

Figure 5 is the example of K-means clustering with outlier removal. It shows that the outlier is removed from the cluster while it means the outlier is not part of neither an object nor an obstacle. It means that the point in the outlier can be neglected as part of the environment in the APF scenario. This stage yields a certain obstacle in the image environment. The example how to define the obstacle is shown in Figure 6 and 7 which uses data set from IMAGE PROCESSING LAB (IPLAB) [25]. Figure 6 is the initial condition when the SURF algorithm was previously applied without K-Means clustering. Then, the blue and red dots are the obstacle detection after K-Means clustering with outlier removal process. It proves that the method can be used for obstacle detection and removing the non-obstacle points.

D. Framework Transformation

The transformation stage is the key to handle the local information. The simple idea is by transforming the image into planar (with a fixed altitude). The first step is to find depth from the result of K-means clustering with outlier removal. By using a simple triangulation in Figure 8, the depth measurement can be obtained easily [26].

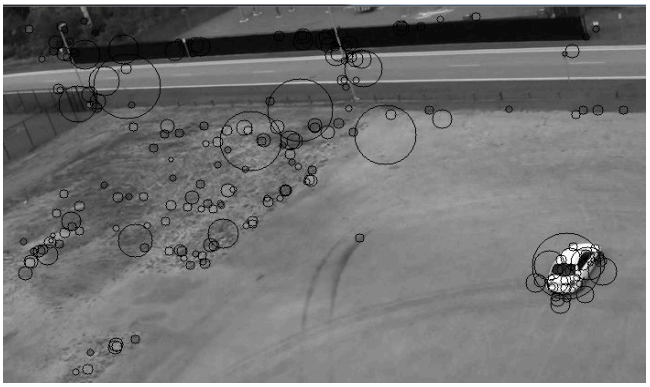

Fig. 6. Example of SURF Application, black round circles are the features detection

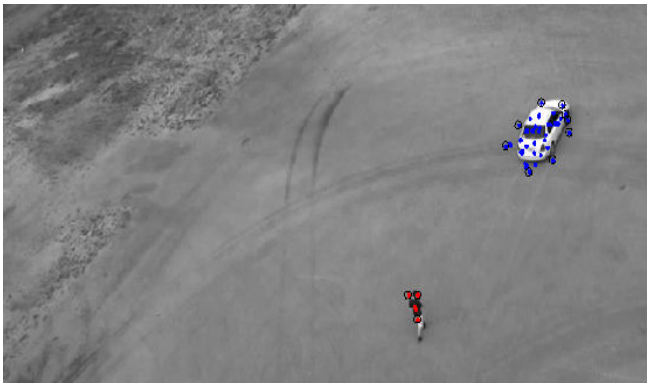

Fig. 7. Example of the Obstacle Detection [25], red and blue dots are the obstacle detection

From Figure 8, f is focal length, P is a point (x,y) in the image, b is disparity camera and the formula of triangulation is

$$\frac{b}{Z} = \frac{b-(d+d')}{Z-f} = \frac{d+d'}{f}. \tag{21}$$

Thus, (21) can be simplified as

$$Z = \frac{f.b}{D}. \tag{22}$$

The implementation of the triangulation usually cannot obtain accurate depth measurement. The approximation of disparity and intrinsic value of focal length from the both camera are the most problem in stereo vision. It is difficult to acquire the exact value of camera disparity since it cannot define the real position of lens from the both cameras. In order to compensate the problem, a fitting method based on logarithmic equation is introduced as

$$D_{act} = a\ ln(Z) - b, \tag{23}$$

where a and b are constant values which set of 17.502 and 37.194 respectively. Variable D_{act} is the approximation of real distance value. By using (23), the distance from the measurement can be refined and obtained an accurate result of depth measurement.

The height of each feature is measured by congruent triangle concept. The illustration of height measurement can be depicted in the Figure 9. The formula is defined as follows

$$tg\,\theta = h'/Z',\ and \tag{24}$$

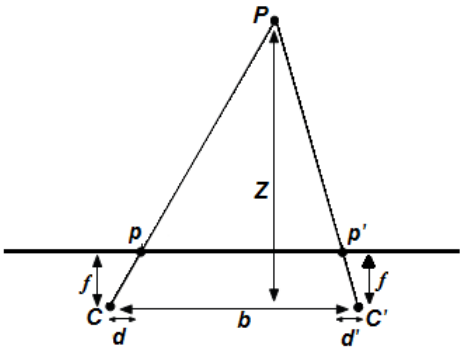

Fig. 8. Triangulation for Depth Measurement

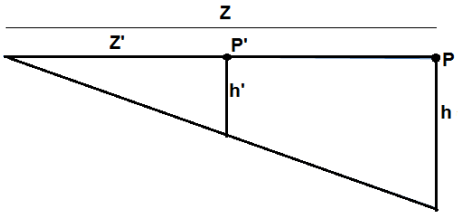

Fig. 9. Height Measurement using Congruent Triangle

$$tg\,\theta = h/Z, \tag{25}$$

then,

$$h' = \frac{h.Z'}{Z} \tag{26}$$

The variable Z and Z' are obtained by distance measurement of triangulation method (22) and (23). Variable h is a constant altitude of the robot.

After the height measurement of each feature can be determined, each feature can be represented by 3D position. Then, the 3D position transforms into the new frame. The transformation is illustrated in Figure 10.

The distance is obtained from the fitting method based on logarithmic equation of the stereo camera as in (23). From Figure 10, the coordinate camera of an image (x_I, y_I) will be used as the parameter of the new coordinates. The mathematical model can be defined as

$$Q = \begin{pmatrix} y_I \\ x_I \\ D_{act} \end{pmatrix}, \tag{27}$$

thus $Q' = RQ$ when

$$R = \begin{pmatrix} 0 & 0 & \frac{x_I}{D_{act}} \\ 0 & \frac{D_{act}}{x_I} & 0 \\ \frac{h}{y_I} & 0 & 0 \end{pmatrix}, \tag{28}$$

where Q is 3D coordinate in the image environment and Q' is the new 3D coordinate in the APF scenario environment.

The framework transformation yields the points which part of the obstacle in the APF scenario environment. In the APF scenario environment, the 2D plane consists of initial, goal, and $Q(h, D_{act})$ which can be modeled as

$$\{(x_{init}, y_{init}) \cup (x_T, y_T) \cup Q'(h, D_{act})\} \in \{APF_{(x,y)}\}. \tag{29}$$

After the points can be defined, the obstacle has to be reconstructed regarding the approximation of the geometry shape of an obstacle. The obstacle shape converts from the unstructured to polygon shape [28]. The conversion of polygon shape is coined as virtual geometry.

E. Convex Hull

The first step of virtual geometry is created by convex hull algorithm. The virtual geometry reconstructs the features to become an object. Based on K-means clustering approach, the reconstruction is divided into two clusters as well. Each cluster will become a virtual geometry.

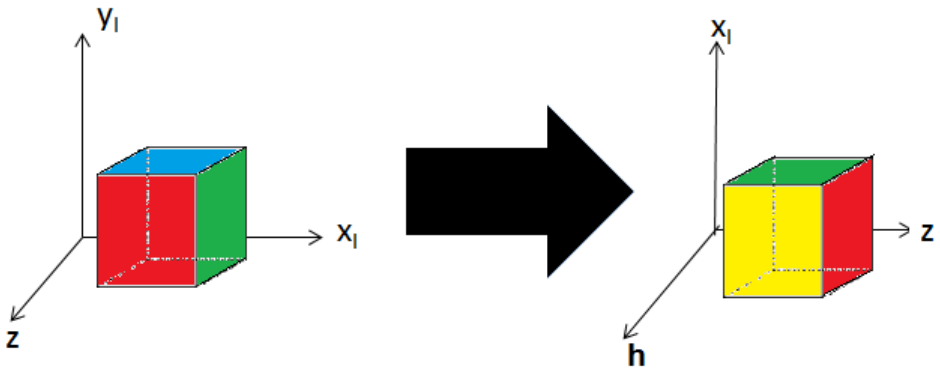

Fig. 10. Framework Transformation

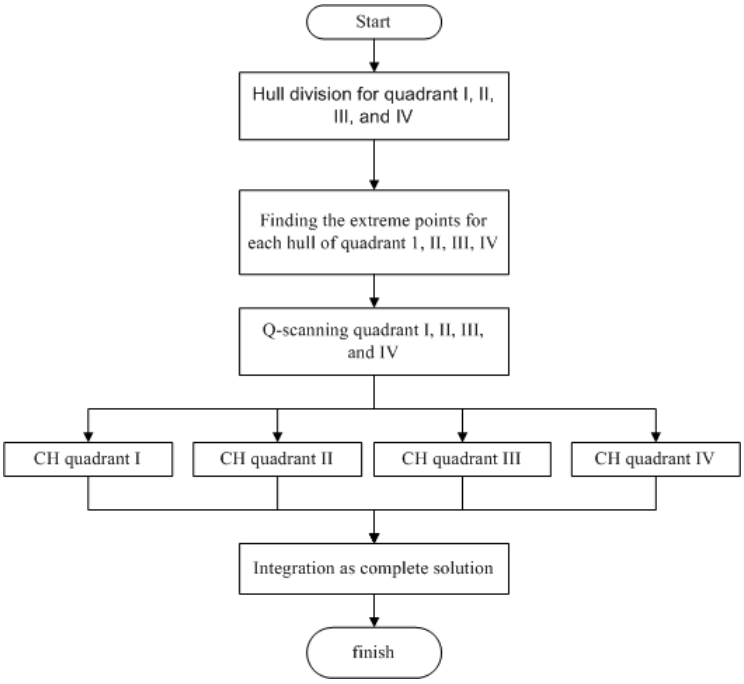

Fig. 11. Q-Scanning Algorithm

The convex hull algorithm uses the Q-scanning approach [29]. The block diagram of Q-scanning algorithm can be illustrated as in Figure 11.

The algorithm starts with dividing the hull into four set points which are called q_1, q_2, q_3, and q_4 referring for the position of four quadrants. The complete solution of convex hull algorithm is the integration of convex hull on each quadrant. The extreme points are determined by each vertex of each quadrant. The maximum value of y with maximum value of x is one of the extreme points of vertex q_1 in quadrant I as

$$\{\epsilon_{tr} \subset q_1 | \epsilon_{tr} = p(max(x), max(y))\}. \quad (30)$$

The other extreme point in quadrant I is the point that has a maximum value of y. The mathematical model is

$$\{\epsilon_r \subset q_1 | \epsilon_r = p(max(x), y))\}. \quad (31)$$

The lowest point of x and y value is the extreme point in quadrant II,

$$\{\epsilon_{tl} \subset q_2 | \epsilon_{tl} = p(min(x), max(y))\}. \quad (32)$$

The left most point is determined by the minimum value of x,

$$\{\epsilon_l \subset q_2 | \epsilon_l = p(min(x), y))\}. \quad (33)$$

It has to be noted that if the extreme points of quadrant I and II are found similar, then the extreme points of quadrant I and II are an identical point.

By contradicting process of quadrant I and II, the extreme points of quadrant III and IV are obtained from the lowest value of x in the lowest and highest value of y. The extreme point formula of quadrant III and IV can be described as follows.

$$\{\epsilon_{bl} \subset q_3 | \epsilon_{bl} = p(min(x), min(y))\}. \quad (34)$$

$$\{\epsilon_{br} \subset q_4 | \epsilon_{br} = p(max(x), min(y))\}. \quad (35)$$

Then, Q-scanning can be implemented to obtain the complete convex hull. The Q-scanning compares the point value from the $max(y)$ until meets the most left and right points. Otherwise, point value from the $min(y)$ has to be compared to obtain q_3 and q_4 hull.

It can be concluded that the Q-scanning yields the vertex of each hull in quadrant I, II, III, and IV respectively. Integration of the hull from the Q-scanning method is the complete solution of the convex hull.

The objective of this stage is to determine the obstacle in the opposite direction of the robot during the robot movement toward the goal. Since the hull points cannot cover all the surface of the obstacle, an interpolation method is used to fill the blank points. The prediction of line that connects a point (x_1, y_1) with a point (x_2, y_2) can be stated with formula

$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1}. \tag{36}$$

Based on the (36), the points (x, y) that fill between the point (x_1, y_1) and the point (x_2, y_2) can be obtained easily. This research sets the distance to one unit between original point and additional point. The completion of additional points is to avoid local optima phenomena in the APF that exists due to the concave shape of an obstacle. On the other hands, the completion points will generate the shortest path since it consider the edge of the obstacle. Determining the repulsive forces surrounding an obstacle that equal to the distance of c-obstacle will yield the shortest path.

F. Goal Point on Local Information

Due to the local information, the problem solver obtains the local optima. It means the proposed method merely solves the problem of local information in certain time. Related local optima, the goal point has to meet safety constraint that assumes as equilibrium point in certain time. The goal point is defined as the center point among the two obstacles [30] as described as

$$x_{loc} = \frac{|c_{obsx1} + c_{obsx2}|}{2} \tag{37}$$

and

$$y_{loc} = \frac{|c_{obsy1} + c_{obsy2}|}{2}, \tag{38}$$

where c_{obsx} and c_{obsy} are the centroid of each cluster that represent the center of each obstacle. Variable x_{loc} and $y_{loc}$are the goal point that reflects the solution of local optima from local information.

G. Global Optimum

All the stages of the proposed method achieve the transformation of the real world environment. The initial, goal, and obstacles have transformation points in the APF scenario environment. Then, the APF can employ directly. In addition, the obstacle has been manipulated to avoid the local optima phenomena from the concave obstacle.

The manner of the proposed method is to cope the problem of local information which means obtain local optima from local information. However, the destination of the robot is the global optimum point from the inertial coordinate. The relation between local optima point from local information and global optimum point from the whole environment scenario is depicted in Figure 12.

Figure 12 means that the global optimum point follows Markov Chain Model which means the next state depends on the current state. Thus, the mathematical model of global optimum is

$$p(s_z|(x_T, y_T)) = \prod_{j=1}^{z} p((x_{locj}, y_{locj})|(odom, Q'(h, D_{act})), \tag{39}$$

where z is total of the goal points from initial condition until the robot reaches global optimum. Equation (39) means probability of the last state (s_z) depends on the position of the robot in the goal point. Robot's belief on the goal position is product rule of positions on the goal point from a certain local information which depends on the robot odometry and image acquisition. The total distance from initial to global optimum point is sum of the distance from initial point travels to next local optima until reaches the goal point. The global optimum is reached when condition $x_{loc} = x_T$ and $y_{loc} = y_T$ are satisfied.

From (39), the global optimum point from the whole environment can be attained by connecting the goal point from local information. For the static environment, the solving problem of local information is obtained from every single frame. Every frame has its own local optima. It means that the global optimum point in the inertial coordinate is met by following the goal of local optima point from one or more than one frames. On the other hands, a similar concept is applied in the moving obstacle. The different is during the robot movement, the acquisition data sensor perceives the alteration of the environment in the real time. If the new data insists to the robot to re-planning the path, then the robot will follow result of the path planning from the new data acquisition. For instance, if during the movement, an obstacle moves toward to the robot, then the previous path will change to the new path with a new local optima from the local information. The condition will continue until the robot reaches the global optimum point in the inertial coordinate.

IV. Experimental Setup

A. Research Material

A Personal Computer (PC) laid on the chair is assumed as a robot with 60 cm height. Here, this research assumes the altitude is in a constant condition. Integration of two web cameras Logitech C270 as a stereo camera is used for data acquisition. The laptop specification is AMD A6 processor, 4 DDR3 1333 MHz SDRAM, AMD Radeon HD 6520G2+HD 7670M with Dual Graphics. For the robot, this research uses Kobuki Yunjin Robot which is differential drive robot as seen in Figure 13. The robot has circle shape with 34 cm of radius. The control system uses integration of closed loop control

Fig. 12. Markov Chain Model of Global Optimum

system and APF based on [31]. Since the control system considers non-holonomic constraint, then it has curvature and acceleration bound. The inertial coordinate refers to odometry that is attached to the robot. The robot and camera frame are set similar to the inertial coordinate.

B. Data Set

This research uses local data set. The environment is in the corridor of Block H at Fakulti Teknologi and Sains Maklumat (FTSM) University Kebangsaan Malaysia and Microwave Laboratorium Indonesian Airforce Academy.

V. Results and Discussion

The experimental setup is divided into two scenarios. Each scenario consists of two different kind of obstacles, i.e. static and moving obstacle. The first scenario tests the proposed method in the real stereo camera without the robot in order to check performance of the proposed method before integrating with the real robot. The second scenario uses the robot with stereo camera for obstacle avoidance and the proposed method is applied in the camera.

A. First Scenario

The test is conducted in the night time using light from the lamp to keep the illumination in a stable condition. The objects are static and moving in opposite direction (along the z axis). The static objects in surrounding environment are the obstacles. On the other hands, the K-means clustering outlier threshold is set to 0.75. The scenario of the experiment is explained as follows. The moving obstacle is assumed as a human that moves toward to the robot and the robot has fixed altitude of 60 cm. The goal is to avoid the moving obstacle and find a safety path to the goal point. The goal point considers the maximum accurate distance of stereo vision and the safety area. The maximum accurate distance is 5 m. Results of the scenario can be seen in Figure 14 and 15.

Figure 14 and 15 show that the proposed method can be used to implement in the APF using local information. The red and white lines are the goal position which meet safety constraint. In the scenario, the robot is assumed as a point mass and the safety constraint is met if the robot is not passing one of the obstacle's point.

Due to the fixed altitude assumption in the scenario, the transformation follows Figure 10 and (27). Consequently, the height of each obstacle's point has been neglected and the height measurement is merely used to avoid the floor detection as an obstacle. The transformation from the local

Fig. 13. Kobuki Yunjin Robot

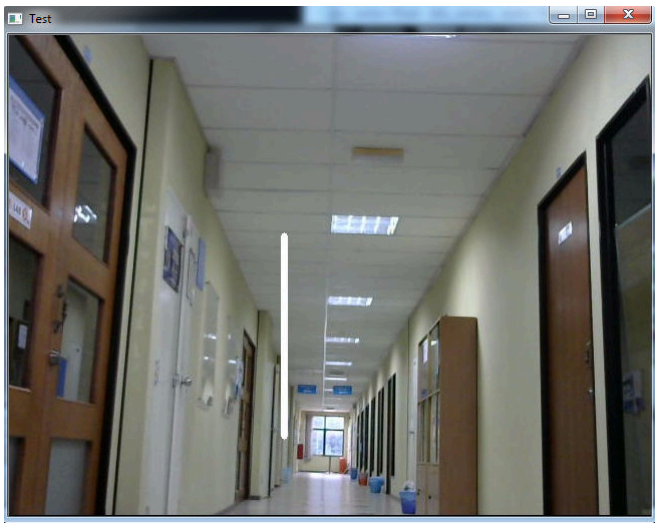

Fig. 14. Example of the final results for static object, the line in white color is robot's direction

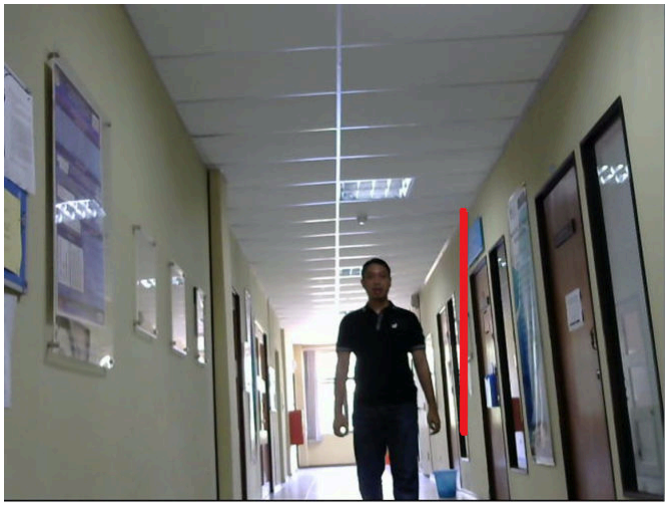

Fig. 15. Example of the final results for moving object, the line in red color is robot's direction

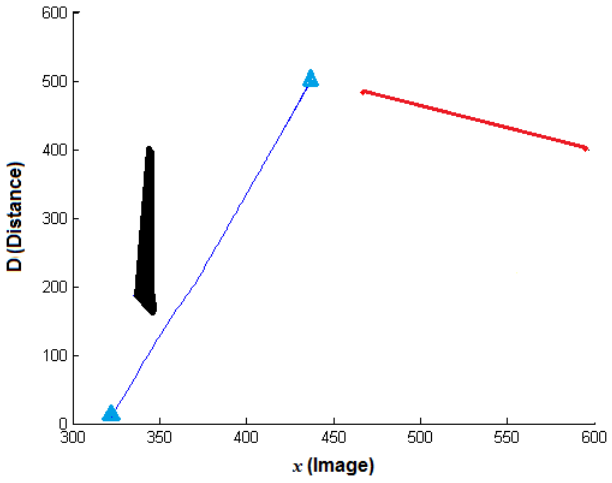

Fig. 16. Example of the final results in the APF scenario, blue triangle is the robot, black and red part are the obstacles, blue line is the path

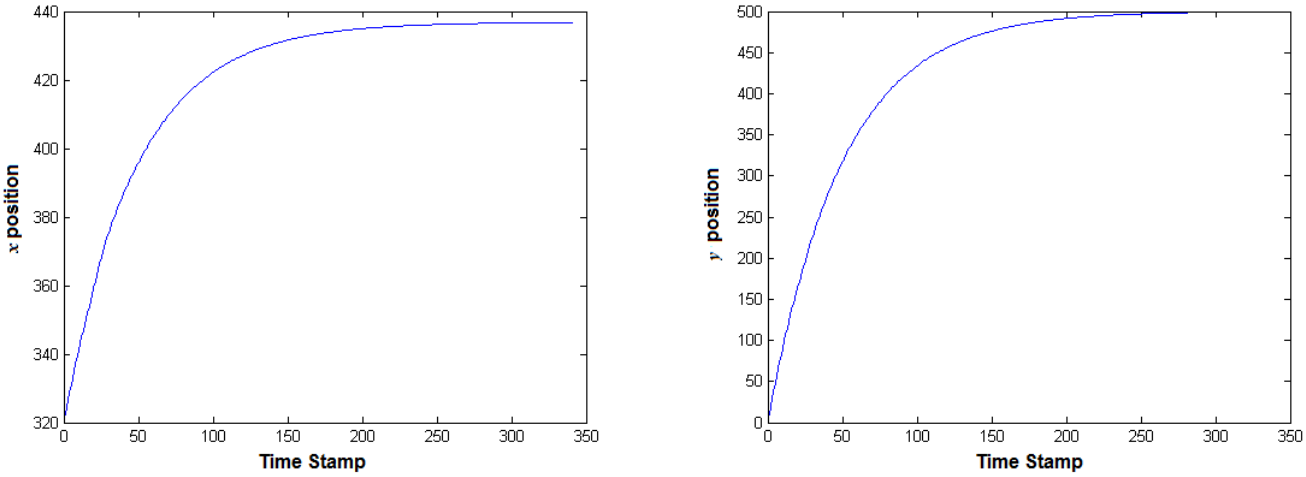

Fig. 17. Time vs robot pose on x and y axes of the APF scenario environment

information to APF scenario environment can be depicted in $\Re^2$ and seen in the Figure 16.

Figure 17 show that the robot's path planning meets convergent for both x and y position. It means that the APF is satisfied to apply in the scenario. The APF has been avoided the obstacles in front and met the goal point.

Since the moving obstacle moves toward to the robot, then trajectory model of the moving obstacle does not necessary to be calculated. By using the proposed method, the robot moves forward until the obstacle has been detected and the robot creates a path for avoiding the moving obstacle. Therefore, random movement of the obstacle needs a further prediction method which is used to determine the APF path planning that meets safety and optimum constraints.

B. Second Scenario

In the second scenario, the experiment uses Kobuki as the robot platform. Stereo camera are installed on the robot and the proposed algorithm is implemented. The initial and goal position of the robot are set on the position (0, 0) and (2.5, 0.3) respectively. A fire extinguisher as the obstacle is arranged in between the initial and goal at (1.5, 0). Scenario with the static obstacle can be seen in Figure 18.

Fig. 18. Scenario 1 using the real robot

In the beginning, the distance of the camera is set to 10 cm following the first scenario. Since the camera has a very limited beam and the robot has a certain size, then result of the experiment showed the robot hits the obstacles. Thus, this research sets the distance of the camera at the maximum distance 22 cm referring size of the camera place. The distance of safety constraint is set to 90 cm. It means that if the distance between the robot and obstacle is equal or below 90 cm then the repulsive force will active and the robot will avoid the obstacle.

Result of the experiment can be seen in Figure 19. It shows the proposed method can be used in the real time system considering kinematic constraints of the robot. The robot avoids the obstacle and reaches its goal.

In order to confirm the experimental result, the position of the robot is plotted in two-dimensional as seen in Figure 20. Figure 20 proves the robot avoids the obstacle in an

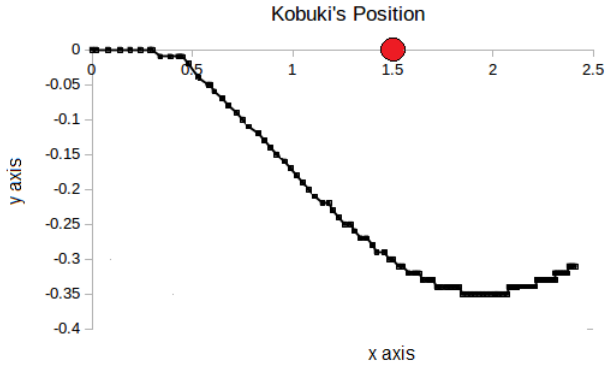

Fig. 20. 2D Plot of Robot's Position, Obstacle - red circle

accurate distance. At 90 cm of distance between the robot and obstacle, the repulsive force is active which is indicated by the robot turn to right avoiding the obstacle. Finally, the robot reaches destination at the goal position.

The next experiment uses a human as the moving obstacle. Similar to first scenario, the obstacle moves toward to the robot with a certain velocity and the robot tries to avoid the obstacle. Figure 21 show the robot avoids the moving obstacle. The trajectory of the robot can be depicted in Figure 22. It can be seen that the robot avoids the moving obstacle at position (1.1, 0).

Regarding the feasible path which relates to the shortest

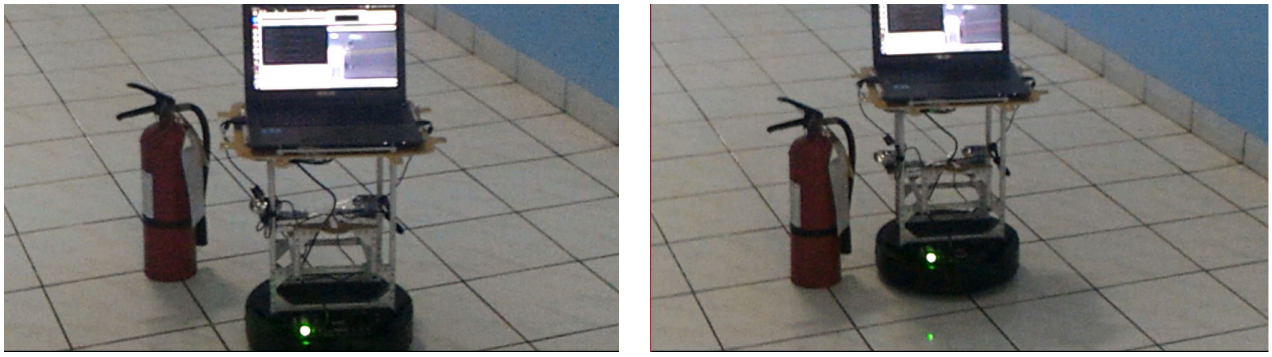

Fig. 19. Result experiment in the real robot with static obstacle

Fig. 21. Result experiment in the real robot with moving obstacle

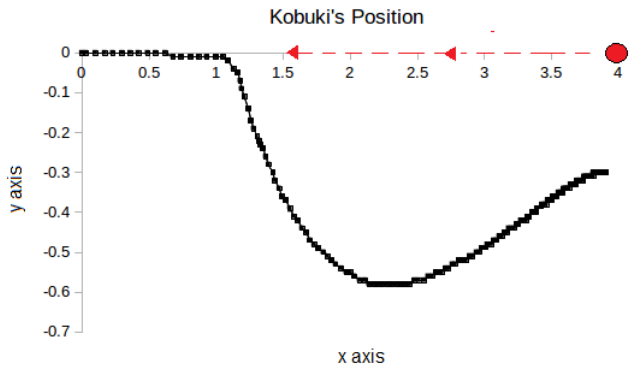

Fig. 22. 2D Plot of Robot's Position, Obstacle - red circle, Red dash line - obstacle's movement

distance of the path, distances of robot trajectory are shown in Table I.

TABLE I
DISTANCE OF TRAJECTORY

Scenario	**Distance (m)**	**Shortest Distance (m)**	**Diff (%)**
Static osbtacle	2.64	2.52	0.96
Moving obstacle	4.18	4	1.04

Table I describes the distance of robot's trajectory is different with the shortest distance. It means that the generated paths from the proposed algorithm are not the shortest paths distance and needs further research to obtain the shortest path.

Another limitation of the experiment is that the moving obstacle is assigned in linear movement. Therefore, it cannot be guaranteed the robot can avoid the obstacle if the moving obstacle has the other modeling of obstacle's movement. The result of the trajectory depicts the robot's movement is not smooth and needs further research to make smooth trajectory.

VI. CONCLUSION

APF is designed successfully using global information. The position of the obstacle, goal, and initial are a prior knowledge. Thus, local information is an issue in the APF. To deal with that, this research proposes a method using framework transformation. The method transforms the sensor framework into the new frame. The image that was acquired by the sensor is used as APF environment scenario with the assumption of a constant altitude. The new frame result determines the goal position, initial and obstacle in the APF scenario. It has been seen that the algorithm can compensate the local information with the assumption of constant altitude level and then, the APF can be applied easily. Furthermore, this research merely considers the linear movement of the obstacle and neglects the other movement of moving obstacle. On the other hands, smooth trajectory must consider due to kinematic and dynamic constraints of the robot.

REFERENCES

[1] M. R. Walter,"*Sparse Bayesian Information Filters for Localization and Mapping*," PhD Thesis, Massachusetts Institute of Technology, 2008.

[2] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. Grixa, F. Ruess, M. Suppa, and D. Burschka,"*Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue*," IEEE Robotic Automation Magazine, 2012, vol. 19(3), pp. 46-56.

[3] A. Sgorbissa and R. Zaccaria," *Planning and obstacle avoidance in mobile robotics*," Robotic Autonomous System, 2012, vol. 60(4), pp. 628-638.

[4] H. Wang and Y. Yu,"*Application of Dijkstra algorithm in robot path-planning*," 2nd International Conference on Mechanic Automation and Control Engineering, MACE, 2011, pp. 1067-1069.

[5] A. Mohammad Pourmahmood, "*3D path planning for underwater vehicles using five evolutionary optimization algorithms avoiding static and energetic obstacles*," Journal of Applied Ocean Research, 2012, vol. 38, pp. 48-62.

[6] A. Tuncer and M. Yildirim.,"*Dynamic path planning of mobile robots with improved genetic algorithm*," Comput. Electr. Eng, 2012, vol. 38(6), pp. 1564-1572.

[7] P. Xuan Zou and S. Bin Ge,"*Improved Genetic Algorithm for Dynamic Path Planning*," International Journal Information Computer Science IJICS, 2012, vol. 1(2), pp. 16-20.

[8] H. Qu, K. Xing, and T. Alexander,"*An improved genetic algorithm with co-evolutionary strategy for global path planning of multiple mobile robots*," Journal of Neurocomputing, 2013, vol. 120, pp. 509-517.

[9] Y. Liu, M. Li, C. Xie, M. Peng, and F. Xie,"*Path-planning research in radioactive environment based on particle swarm algorithm*," Prog. Nucl. Energy. 2014. vol. 74, pp. 184-192.

[10] H. Mo and L. Xu,"*Neurocomputing Research of biogeography particle swarm optimization for robot path planning*," Neurocomputing, 2015, vol. 148, pp. 91-99.

[11] D. Yazdani, B. Nasiri, A. Sepas-Moghaddam, and R. Meybodi Mohammad,"*A novel multi-swarm algorithm for optimization in dynamic environments based on particle swarm optimization*,"Journal of Applied Soft Computing, 2013, vol. 13(4), pp. 2144-2158.

[12] S. Chakraborty,"*Ant Colony System: A New Concept to Robot Path Planning*," International Journal Hybrid Information Technology, 2013, vol. 6(6), pp. 11-30.

[13] W. Zhangqi, Z. Xiaoguang, and H. Qingyao,"*Mobile Robot Path Planning based on Parameter Optimization Ant Colony Algorithm*," Procedia Engineering, 2011, vol. 15, pp. 2738-2741.

[14] K. Mustafa Servet, G. Mesut,"*A recombination-based hybridization of particle swarm optimization and artificial bee colony algorithm for continuous optimization problems*," Applied Soft Computing, 2013, vol. 13(4), pp. 2188-2203.

[15] J. M. Martin Ramos, D. Lpez Garca, F. Gmez-Bravo, and A. Blanco Morn.,"*Application of multicriteria decision-making techniques to manoeuvre planning in nonholonomic robots*, Expert System Application, vol. 37(5), pp.39623976, 2010.

[16] Y. Kwangjin Gan., S. Keat, and S. Salah,"*An efficient path planning and control algorithm for RUAV's in unknown and cluttered environments*," Journal of Intelligent and Robotic Systems: Theory and Applications, vol. 57, pp. 101-122, 2010.

[17] M. Shanmugavel, A. Tsourdos, B. White, and R. Zbikowski, "*Co-operative path planning of multiple UAVs using Dubins paths with clothoid arcs*," Control Engineering Practice, vol. 18(9), pp. 10841092, 2010.

[18] M. Shanmugavel, A. Tsourdos, and B. A. White,"*Collision avoidance and path planning of multiple UAVs using flyable paths in 3D*," 15th International Conference on Methods and Models in Automation and Robotics, pp. 218222, 2010.

[19] H. H. Triharminto, A. S. Prabuwono, T. B. Adji, and N. A. Setiawan,"*Adaptive Dynamic Path Planning Algorithm for Interception of a Moving Target*," International Journal Mobile Computing Multimedia Communnication, vol.5(3), pp.1933, 2013.

[20] M. Kalavsky and Z. Ferkova,"*Harmonic Potential Field Method for Path Planning of Mobile Robot*," International Virtual Conference (ICTIC 2012), Slovakia, 2012, vol. 1, pp. 4146.

[21] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, MIT Press, 2004.

[22] L. Li,"*Image Matching Algorithm based on Feature-point and DAISY Descriptor*,"Journal of Multimedia, vol. 9(6), pp. 829-834, 2014.

[23] P. M. Panchal, S. R. Panchal, S. K. Shah,"*A Comparison of SIFT and SURF*,"International Journal of Innovative Research in Computer and Communication Engineering, vol. 1(2), pp.53-60, 2013.

[24] M. A. Fischler and R. C. Bolles,"*Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography*," Communications of the ACM, vol. 24(6), pp. 381395, 1981.

[25] IMAGE PROCESSING LAB (IPLAB). [Online]. Available: http://www.aforge.net/

[26] J. Davis, D. Nehab, R. Ramamoorthi, Rusinkiewicz, and Szymon, "*Spacetime stereo: A unifying framework for depth from triangulation*," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27(2), pp. 296-302, 2005.

[27] V. Hautamaki, S. Cherednichenko, I. Karkkainen, T. Kinnunen, and P. Franti,"*Improving K-Means by Outlier Removal*,"SCIA 2005, LNCS 3540, pp. 978987, 2005.

[28] A. Gorbenko, V. Popov"*Visual landmark selection for mobile robot navigation*," IAENG International Journal of Computer Science, vol. 40(3), pp. 134-142, 2013.

[29] H. H. Triharminto, A. W. Wasisto, O. Wahyunggoro, T. B. Adji, A. I. Cahyadi,"*A Novel Q-Scanning for Convex Hull Algorithm*," Joint International Conference on Electric Vehicular Technology and Industrial, Mechanical, Electrical and Chemical Engineering (ICEVT and IMECE), 2015.

[30] Y. Volkan Pehlivanoglu,"*New vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV*,"Journal of Aerospace Science and Technology, vol. 16(1), pp.47-55, 2012.

[31] H. H. Triharminto, O. Wahyunggoro, T. B. Adji, A. I. Cahyadi, "*An Integrated Artificial Potential Field Path Planning with Kinematic Control for Nonholonomic Mobile Robot*," International Journal on Advance Science Engineering and Technology, vol. 6(4), pp. 410-418, 2016.