

Research Article

Open Access

Talgat Islamgozhayev*, Maksat Kalimoldayev, Arman Eleusinov, Shokan Mazhitov, and Orken Mamyrbayev

First results in the development of a mobile robot with trajectory planning and object recognition capabilities

DOI 10.1515/eng-2016-0049

Received Jun 22, 2016; accepted Aug 05, 2016

Abstract: The use of mobile robots is becoming popular in many areas of service because they ensure safety and good performance while working in dangerous or unreachable locations. Areas of application of mobile robots differ from educational research to detection of bombs and their disposal. Based on the mission of the robot they have different configurations and abilities – some of them have additional arms, cranes and other tools, others use sensors and built-in image processing and object recognition systems to perform their missions. The robot that is described in this paper is mobile robot with a turret mounted on top of it. Different approaches have been tested while searching for best method suitable for image processing and template matching goals. Based on the information from image processing unit the system executes appropriate actions for planning motions and trajectory of the mobile robot.

Keywords: mobile robots; trajectory planning; feasible planning; vision system; object detection

1 Introduction

Nowadays detection of objects and taking actions according to its behavior are one of the toughest and broadly re-

searched problems in computer science. According to the needs of specific jobs that will be performed by the robotic system the needs in the actions that they can perform also differ from system to system. Broad range of rescue and military robots use vision systems to detect an object, track and approach it to perform their tasks. There are many examples of using such systems [1, 2] that use image processing together with planning the movements of robots. These works mostly discuss some part of the system and the results of merging vision system results with path planning technique is not clear because the image of the overall system is not presented. In this research we present the first results in merging the works of computer vision team together with the trajectory planning team achievements. Similar system was described in [3], which describes a system with object recognition and trajectory planning that uses sensors and camera, the difference is in the path planning mechanism used in this work that constructs a 3D map of environment and makes decision on movement of the platform according to this map. Our results differ in path planning mechanism and uses simultaneous movements using the information from the camera which is processed by vision processing system that outputs results about the object behavior, so-called leader-follower approach [14, 15], without construction of map.

There are many approaches that have been described for trajectory planning of a mobile robot using monocular cameras. But due to the complexity of the task many authors suggest using combination of multiple sensor systems [12] or construction of the 3D scene of the environment by using stereoscopic cameras [13] which makes the process of obtaining depth data from captured images easier. We present architecture of a system with a monocular camera unit and trajectory planning capabilities.

The system that is discussed in this paper is four wheeled ATV (also known as quadro-cycle) able to move in cross-country terrain driven by a 48V DC motor and powered by a set of DC batteries (Fig. 1). It is featured by a turret on top of the platform and all the steering and turning

*Corresponding Author: **Talgat Islamgozhayev:** Institute of Information and Computational Technologies, Almaty, Kazakhstan; Email: talgat90.07@gmail.com

Maksat Kalimoldayev: Institute of Information and Computational Technologies, Almaty, Kazakhstan; Email: mnk@ipic.kz

Arman Eleusinov: Institute of Information and Computational Technologies, Almaty, Kazakhstan; Email: armankaznu@mail.ru

Shokan Mazhitov: Institute of Information and Computational Technologies, Almaty, Kazakhstan; Email: choka911@mail.ru

Orken Mamyrbayev: Institute of Information and Computational Technologies, Almaty, Kazakhstan; Email: morkenj@mail.ru

 © 2016 T. Islamgozhayev *et al.*, published by De Gruyter Open.
This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 License.



Figure 1: ATV with turret.

mechanisms are automated using DC motors. Movements of the system are controlled by a low-powered CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture ATmega with board and needed motor/sensor controllers. According to the target of research the system should be able to search for, locate and track the object on a video by matching it with a template given by the operator [6, 7]. A C/C++ programming language was used to detect objects and perform computer vision operations. Moreover, the system should be able to plan the trajectory of rotation and follow the detected object. As the computing device iTop tablet PC with Intel Atom CPU was used. The system can receive operator commands by directly connecting PC to a microcontroller board through rs232 port or by sending commands wirelessly using an APC220 module. The article is an overview of a system that provides some approaches that are used in the development of the system – this is not the final results.

The article is organized as follows. Section 2 describes the planning strategy that is used in developed system with introducing the vision systems' algorithm; it also introduces motion and trajectory planning mechanism that constructs the movement strategy of the system based on the information obtained from vision sensor. Section 3 combines the results from different stages of development and shows experimental results. Finally, Section 4 presents conclusions on what have been done.

2 Planning and Vision System

The planning consists of several steps:

1. the environmental information should be gathered to have the view of the surrounding – it can be done by vision system and sensors;

2. the needed objects should be founded on the view of the robot;
3. tracking strategy should be formulated;
4. motion and trajectory planning strategies should be constructed to follow the object by moving the mobile robot;
5. sending the commands according to needed movements.

Steps above are applied on the real robotic system developed by our team for the purpose of testing results of research project. All described algorithms and planning strategies were practically tested on this platform and the applicability in real life is proven.

As the criterion for planning at first we took feasible planning because it will take more time to make an optimal planning strategy, which will eventually be performed in our future works. According to definition, a feasible planning is when a plan causes arrival of the system at a goal state, regardless of its efficiency [5]. The planning problems of our system at the time of investigation were simplest to describe because the state space were finite in most cases. All models are known and predictable, so at first we didn't have any uncertainty and there is no need in probability theory.

As we work with state-space models a discrete feasible planning model was defined. We can describe the discrete feasible planning parameters according to [5] as follows

1. state space X , finite countably infinite set of states
2. x is in a set of X , a finite action space $U(x)$
3. State transition function $f \Rightarrow f(x, u)$ is in a set of X for every x in set X and u in set $U(x)$. The state transition is derived from f by $x' = f(x, u)$.
4. Initial state which is the state at the beginning of the process x_1 in a set of X .
5. And a goal set X_g is a subset of X .

In the described system the steps are performed according to the following order:

- sense the environment by using vision sensors
- process the vision sensor data and derive the current state (or initial state) of the robotic system
- compute the needed movement strategy or actions to use to move according to the movement of the tracked object
- use set of states (actions) that can be performed by our system
- achieve the goal state by achieving the state from the goal states of the X_g .

As the first step of planning described above, we suggested gathering environmental information from camera

and by using a template matching algorithm extracted the features of the tracked object in order to continue on the other steps.

Taking into account the usage of most robots in outdoor environment we have to build a corresponding system that can perform tasks in different conditions. In difficult conditions, the quality of the signal and the result of observation can be improved by using the algorithms for estimating the parameters of geometric transformations of images and methods of spatial and temporal filtering [4]. We can point three main methods out among them:

1. Template matching methods can be used to measure the position of fixed and moving objects observed in a uniform and non-uniform background.
2. Using the information on the statistical properties of the object and the background we can isolate moving and stationary objects observed at a relatively uniform background. Such methods are called methods of statistical segmentation.
3. Methods of distribution of dynamic changes based on allocation of changes that occur over time in the observed group of images. Such methods are used for solving the problem of extraction of moving objects.

Despite many other approaches [6–8] we used template matching method to detect and extract features from the object. Template matching is done by using normalized square difference template matching algorithm modified by adding the time constraint and is described by

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y')^2 - I(x + x', y + y'))^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \sum_{x', y'} I(x + x', y + y')^2}} \quad (1)$$

Where each location (x, y) contains the match metric needed for construction of the result matrix R . Here T is the template image and I is the initial image from the video camera. To complete these steps have been written a program for capturing a video from monocular vision system installed on the turret of the vehicle. For filtering purposes we used the smoothing with Gaussian kernel. The following formula 2 describes the Gaussian standard deviation or sigma (σ):

$$\sigma = 0.3(n/2 - 1) + 0.8 \quad (2)$$

where n can either be size 1 for horizontal kernel or size 2 for vertical kernel. Using σ for small kernels reduces the time required for calculation. The following images show the results obtained from real-time image processing unit using threshold and normalized template matching algorithm with time optimization for better results.

2.1 Planning the trajectory of rotation of turret. Tracking the object

Most of automated vehicles have traditional design. Turret is located on top of the vehicle to make it able to rotate around axis z (in Fig. 4 it is an axis y). The turret consists of 2-DOF system: 1 – for rotating around z axis and 2 – for moving the unit with camera up and down in some range around x and y axis depending to which side the turret is pointing. The horizontal azimuth rotation of the turret is realized by using a bearing to the rotating part of which the metal basement of the turret was welded. Rotation is done by DC motor with gear divided into small gearings to ensure accurate positioning of the turret while turning to a dedicated target. The x axis rotation is done by the same principle and differs from horizontal one only in amount of bearings – it has two bearing on both sides of the rotating mechanism and can only turn to maximum of 60° angle. In this case a motor fixed to the turret drives a gear box with small gearing to make the camera basement move up and down elevating it in pitch. Motor is controlled by a switching mechanism to turn it on or off on time while the signal is sent from the controller.

The turret with the camera operates on three dimensional spaces. The camera is placed like in Figure 2 on the turning mechanism on horizontal axis. There are many investigations in developing such a system that can detect and track an object on some point $P = (x, y, z)$. Papers [6, 7, 7] describe different strategies for tracking the movement of an object in a space and camera rotation according to these tasks. These methods mainly describe the camera movement with its base on the back of the camera. In this work we propose a method to track an object with tacking into account the camera movement when it is hold and turned by the center of it ensuring the stability and little

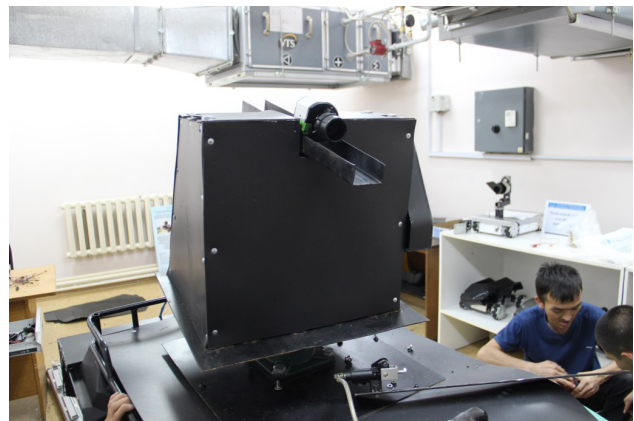


Figure 2: Turret with camera (front view).

angles deviation. Next results, describing the movement of the camera which takes into account the movement of the platform where camera is placed will be described in next paper.

So, suppose the camera is placed on some uniform platform and directed to some point $P_0 = (x_0, y_0, 0)$. Z is 0 because the point lies on x and y plane, z is only to move the camera up and down which we do not take into account because we are not interested in the objects that are out of the range of the camera in an angle of 90 degrees. For this purpose we firstly need to turn the camera around l_1 axis to some angle α_0 ($|\alpha_0| < \pi/2$), then around l_2 angle α_0 ($|\beta_0| < \pi/2$). Let's calculate turning angle α_0 and β_0 .

$$\alpha_0 = \pm \arctan \sqrt{\frac{x_0^2}{(y_0 + h)^2 + H^2 - h^2}} \quad (3)$$

Here H is the distance from the initial position of the camera to the surface of observation and h – the distance between two axes of rotation of the turret on which the camera is mounted. Rotate the camera from the initial position around the axis l_1 to some angle α ($|\alpha| < \pi/2$) and denote the optical axis of the camera lens as d_α . If we fix angle α and rotate the camera around the axis l_2 , then the line d_α will rotate around the axis l_2 , plotting space sheeted hyperboloid of rotation (Fig. 4) [4, 5].

Let's turn the camera from its initial position around the l_2 axis to some angle β ($|\beta| < \pi/2$) and denote the optical axis of the lens as d_β . Because of the first gear is fixed to second gear this revolves will not only change its direction but will also move on $Oxyz$ plane by circle with its center on point D and radius h . We will denote new position of the camera as C_β . l_1 axis will also rotate around the l_2 axis

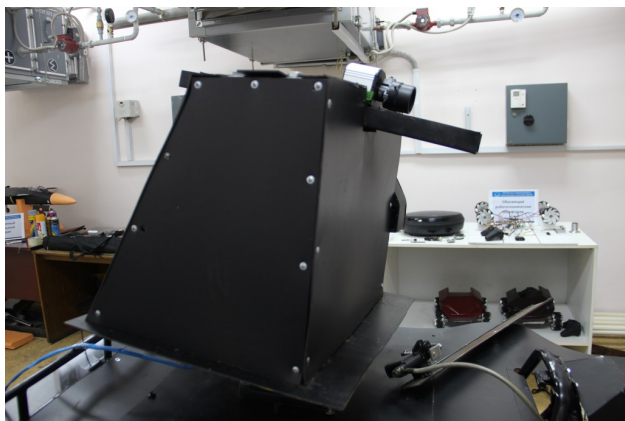


Figure 3: Turret with camera (side view).

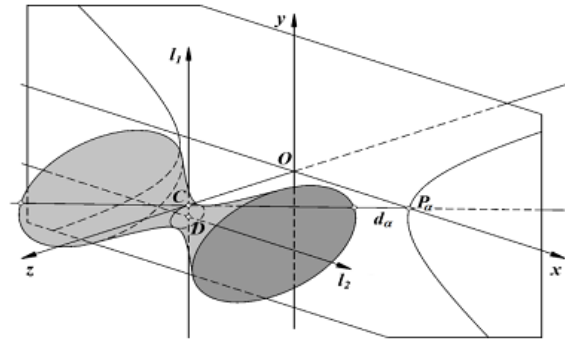


Figure 4: Scheme of the rotation of camera platform ($a = 0$).

to an angle β . Denote the l_1 axis after turning as $l_1\beta$.

$$\beta = \beta(y) = \begin{cases} 2 \arctan(h/H) & , y = -2h \\ 2 \arctan \frac{H - \sqrt{H^2 + y(y+2h)}}{y+2h} & , y' = -2h. \end{cases} \quad (4)$$

Turning angles and orientation of the turret was calculated by the above formula and appropriate motion planning class was written for this purposes. After knowing the angles of rotation of the camera and position of the object related to the center of the camera we calculated the initial radius of the circular object. The difference in the radiuses of real-time image from camera and template image will give us the value that can be used to measure the distance to the object from platform. Then to achieve the movements of the platform from side to side according to the image from camera we set the borders of the movement of the object in the view – if the object passes the border of view the system gives the platform a signal to rotate left or right – depending which border the object have passed while leaving the view. The following Fig. 5 shows the location of the tracked object relative to the resolution of the image from camera (the resolution of our camera is 640×480 – width*height). For each movement of the platform the dedicated variable was used: x-axis,

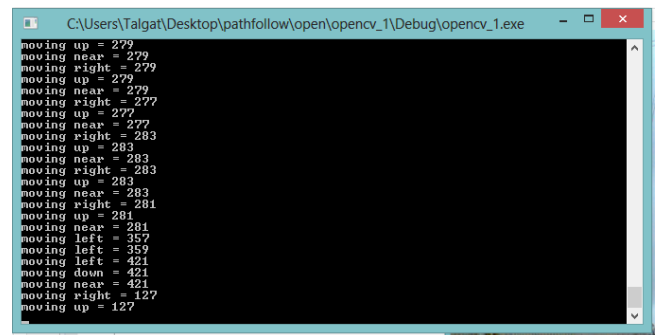


Figure 5: Reports on movements of the object.

y-axis, right turn of the platform, and left turn of the platform, move forward or backwards. The following part of the code calculates the turning angle for the platform – we take the angle of view (camera specific), x resolution of camera, flag on reaching the pivot point, current position of the object.

```
int turningAngle(int angleOfView, int sizeX,
bool pivot, int curX)
{
    // half of the width of the image (640*480)
    int turnGrade = angleOfView / 2;

    int turnTo = 0;

    // calculated by dividing the width of view /
    // AOV - 640 / 110 = 5.82;
    double oneGrade = sizeX / angleOfView;

    // turnTo - degree to turn the platform to, by
    // default 0 degrees
    // if pivot == true - angle is more than pivot
    // point (central line)
    if (pivot)
    {
        turnTo = (curX - 320)/oneGrade;
    }
    else
    {
        turnTo = ((320 - curX)/oneGrade)*(-1);
    }
    return turnTo;
}
```

For each variable mentioned, we used PID controller control scheme with closed loop to bring it to a particular value stabilizing against the position of the object from the camera. Differing from other previously described systems, platform while turning according to a turret position uses not the precalculated angle of rotation but the parameter marked as a flag – while rotating the turret the parameter is updated according to its position, and the platform just turns left of right side to reach this flag. Described solution lowers the cost of platform development saving money that could be spent on buying angle estimation sensors and makes the adjustment of angle of rotation be fined programmatically.

As the system developed for practical usage in outdoor environment practical results were more useful to us than computer modeled ones. To achieve the goal we have tested the system in cross-country environment by using

as an object for detection and tracking a moving rectangular or circular objects hold by a human and moving in some average speed. The vision system was tested using six template matching methods: square difference matching, correlation matching, correlation coefficient matching, normalized square difference matching, normalized correlation matching, and normalized correlation coefficient matching [11]. As the best suitable method for experimental results in outdoor environment normalized coefficient matching method was chosen because it has a lowest occurrence of mismatching. For the experimental tests of proposed system the TVS consisting two cameras was created – one for platform and second for turret – in this paper we described the camera mounted on turret. Overall scheme of the system is shown in Figure 6. The micro-

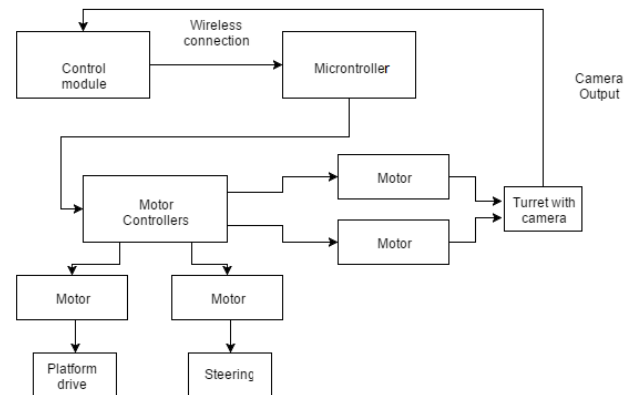


Figure 6: Scheme of the system.

controller acquires the actual values from control module and according to a state space derives a control signal from the difference of previous state and required final state. Then it sends the control commands to motor controller that is responsible for driving the motors. By selecting forward or reverse rotation, and regulating speeds of motors they are turned to a specified angle and the maximum and minimum angles are also given programmatically to prevent the usage of extra circuitry. Further results will be discussed in next report.

3 Conclusion

We presented first results of a development of a system of autonomous mobile robot with trajectory planning and object recognition capabilities. Monocular vision system was used for environment observation and as an input device for image processing system. Testing different ap-

proaches we decided to build image processing module upon the normalized correlation coefficient matching algorithm, which have shown the best template matching properties: it can match images in a changing environment. The movement strategy was made using leader-follower approach with the help of a virtual leader, or, in other words, the matched image (according to the given template) from the vision processing system. This module performs the calculation of the position of the target, turns the platform, and approaches it if it moves away and moves back if it is too close. Despite the difficulties with image stabilization and calculation lags the developed system can near-to-accurate track and approach the moving object – the big problem is the factor of light, but we will try to solve it in our future work. So, future works will be dedicated on optimization of vision and creation of advanced trajectory planning algorithms. Moreover, the idea of using multiple controllers to achieve on-time signal processing is also one of the targets in the way of making a fully functional fast responding autonomous vehicle.

References

- [1] Weyrich M., Laurowski M., Klein P., Wang Y., High speed vision based automatic inspection and path planning for processing conveyed objects, *Procedia CIRP*, 2012, 3, 442–447
- [2] Mohammed A., Wang L., Gao R., Integrated image processing and path planning for robotic sketching, *Procedia CIRP*, 2013, 12, 199–204
- [3] Lemaire T., Berger C., Jung I., Lacroix S., Vision-based SLAM: stereo and monocular approaches, *Int J Comput Vision*, 2007, 74(3), 343–364
- [4] Boyko I.A., Guryanov R.A., Recognition of objects based on the video obtained from camera mounted on a mobile platform, *Young scientist*, 2013, 6, 26–34 (Russian)
- [5] LaValle S.M., *Planning algorithms*, Cambridge University Press, Cambridge, 2006
- [6] Weibo W., *Study on automatic object detection and tracking based on image*, Nanjing University of Science and Technology, Nanjing, Jiangsu, China, 2006
- [7] Hao Z., Zhu M., Design and implementation of target intelligent detecting and tracking system, *Opto-Electronic Engineering*, 2007, 34(1), 27–31
- [8] Abramov N.S., Romankin V.A., Methods of control of rotating video camera (in Russian), *Izvestiya SFedU, Engineering Sciences*, 2013, 7(144), 173–179
- [9] Aleksandrov P.S., *Lectures on Analytical Geometry* (in Russian), Nauka, Moscow, 1968
- [10] Vygodsky M., *Mathematical Handbook: Higher Mathematics*, Mir Publishers, Moscow, 1975
- [11] Martin J., Crowley J.L., Experimental Comparison of correlation techniques, In: *Proceedings of the Conference on Intelligent Autonomous Systems* (March 1995, Karlsruhe, Germany), IAS '95, 1995, 86–93
- [12] Pollefeys M., Nister D., Detailed real-time urban 3D reconstruction from video, *IJCV*, 2008, 78, no. 2–3, 143–167
- [13] Wen Chung C., Shu-An L., Real-time feature-based 3D map reconstruction for stereo visual guidance and control of mobile robots in indoor environments, In: *Proceedings of 2004 IEEE International Conference on Systems, Man and Cybernetics* (10–13 October 2005, Hague, Netherlands), IEEE Xplore 2005, vol. 6, 5386–5391
- [14] Wang P.K.C., Hadaegh F.Y., Lau K., Synchronized formation rotation and attitude control of multiple free-flying spacecraft, *AIAA J. Guidance, Contr. Dynamics*, 1999, 22(1), 28–35
- [15] Ren W., Sorensen N., Distributed coordination architecture for multi-robot formation control, *Robot. Auton. Syst.*, 2008, 56, 324–333