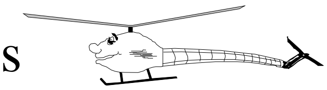
Cooperative Autonomous Mission Planning and Execution for the Flying Robot MARVIN



Marek Musial, Uwe Wolfgang Brandenburg, Günter Hommel
Technische Universität Berlin
Real-Time Systems & Robotics Group
http://pdv.cs.tu-berlin.de/

Abstract

MARVIN is an autonomously flying robot based on a model helicopter. It has been designed for participation in the International Aerial Robotics Competition (*IARC*) Millennial Event 1998 – 2000. The competition task consists of a search operation in an unknown disaster scene with threats such as fires, water fountains, and smoke. The overall system consists of the helicopter itself and the ground station hosting most of the computing power. This paper describes how the system components cooperate in planning, executing and re-planning the search mission guided by a variety of sensor data. The most important sensor is a digital photo camera, its images being analyzed by a number of independent computer vision modules. In 1999, MARVIN has taken the lead of the three-year competition.

1 Introduction

MARVIN is an abbreviation for **M**ulti-purpose **A**erial **R**obot **V**ehicle with **I**ntelligent **N**avigation. This name expresses that MARVIN is an autonomously flying robot that can fulfill a mission purely on the basis of aquired sensor data, without *any* human interaction.

The requirements to be fulfilled by MARVIN are defined by the mission of the 1998–2000 International Aerial Robotics Competition, as described in [1]. The mission task consists of finding and classifying hazards and human bodies in a simulated disaster area. There are black drums containing dangerous materials, with the contents distinguishable by symbols on the drums' surface. There may be fires, water fountains, and smoke threatening the operation of the robot.

The historic background of MARVIN involves TU Berlin's participation in the 1995 IARC with the blimp robot TubRob [9, 10], the presentation of MARVIN's technical conception at the 1998 event [11, 6] and the most successful participation in the IARC of 1999 [5, 7], with the result of TU Berlin's leadership in the qualification score and its most promising starting point for the 2000 finale, which will finally take place on June 29th, 2000.

This paper deals with the task of mission planning and execution, which is split between the robot and the ground station and solely relies on sensor data aquired during the mission – with no human interaction at all, as required by the competition rules. Sections 2 provides an overview of the complete system. The remaining sections deal with the components and information involved in the task of mission planning and execution, the mission strategy and its implementation, and the different image analysis algorithms in use.

Figure 1: MARVIN in flight

2 System Overview

The basis of the MARVIN air vehicle is a conventional model helicopter, the "Petrol Trainer" 180 by SSM, Germany. It has a rotor diameter of 1.8 m and is equipped with a two-stroke petrol engine producing about 2 kW. Figure 1 shows MARVIN in flight.

The robot's position is determined by a carrier phase differential GPS receiver, NovAtel RT-2, which is lent by NovAtel to all IARC teams on request. Its accuracy relative to the reference antenna amounts to 2 cm. Additionally, the helicopter is equipped with a three-dimensional electronic flux-gate compass, three semiconductor acceleration sensors and three piezo-electric rotation sensors ("gyroscopes") for measuring the orientation. The orientation measurement unit has been handmade in the MARVIN project at material costs below $200.

In order to detect obstacles and threats in the area of operation, MARVIN has an ultrasonic sensor facing down to measure the actual altitude over the surface or an obstacle, and a radar sensor looking ahead to detect obstacles in the flight path. Further there is a flame sensor, which detects flames through their characteristic UV spectrum.

On-board computational power is provided by a single chip, a Siemens 80C167 microcontroller. The 80C167 can, for example, measure or generate up to 32 pulse-width modulated (PWM) signals without consuming a single CPU cycle, and has a 16-channel analog digital converter on chip. The selection of this chip has proven almost ideal with respect to power consumption, overall weight, and system complexity.

A pair of Siemens Gigaset M101 Data modules serve as a data communication link between the robot and the ground station. These modules are based on DECT technology known from cordless telephones, but provide some sort of wireless "null-modem cable" at 115.200 bps. A conventional remote control unit provides a backup system, allowing a human pilot to take over control instantly at any time.

Image acquisition takes place via a digital photo camera. The camera, a Sanyo X350, is connected via RS232 to the 80C167, which allows the transmission of a 640x480 pixel image, already JPEG compressed, to the ground station every 8 seconds. This is done using the DECT link that is required anyway.

The ground station, consisting of a network of laptop PCs running under Linux, runs the image analysis programs and the "mission control" module. The latter implements the mission strategy and serves as a user interface for observation, testing, and emergency situations.

Figure 2 presents a diagram of all the components of the MARVIN system.

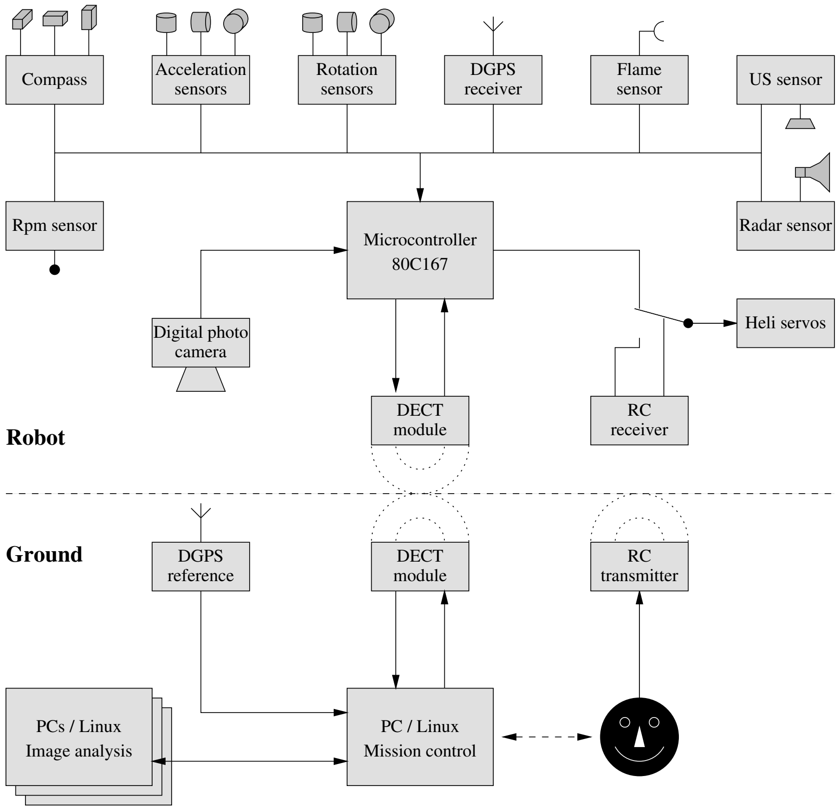


Figure 2: Overview of the MARVIN system

3 Mission Execution Overview

This section gives a detailed description of the interaction between the components directly involved in mission planning and execution, in the sense of this paper.

3.1 Components

The following list enumerates the components involved in mission planning and execution and describes their specific contribution to the overall process:

- The *GPS receiver* on board continously determines the robot's (x,y,z)-position.
- A couple of *on-board sensors* detect potential threats to the robot: the flame sensor, the sonar, and the radar sensor.
- A couple of *image analysis software modules* running on the ground station computers look for target objects in the images taken by the on-board camera.

  Every recognition of an object within an image is attributed with a measure of reliability, roughly approximating the probability of the recognition's being correct. The image analysis modules are basically independent, each of them designed to detect a certain class of objects. Therefore, it is possible to use more than one module for a given object class. This introduces redundancy into object recognition – if one algorithm encounters difficulties due to adverse environmental conditions, it is highly probable that another algorithm will not suffer the very same problem.
- The *camera module* on board issues an immediate notification about every image taken, containing the location, altitude, and orientation of the robot at the very moment of

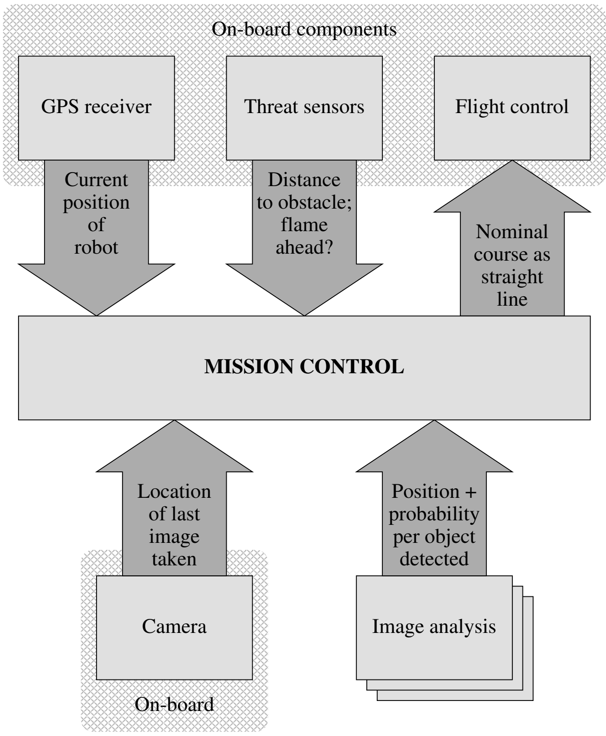


Figure 3: Flow of information between the components involved in mission execution

triggering the digital photo camera. This notification allows to make reasonable use of the 8 s time delay required for the transmission of the actual image data to the ground station

- The *mission control software module* implements the mission strategy. It collects all the sensor data and uses it for planning the next mission steps.
- The *flight control module* running on-board drives the servos such that the robot follows the desired flight path. This module actually executes the activities planned.

3.2 Flow of Information

Figure 3 gives an overview of the information exchange between the components enumerated in the preceding section.

The mission control module acts as the only link between the other components. Mission control receives the current position, the threat data, the notifications about the images taken, and the findings of the image analysis modules. It decides about the next flight path segment, which is propagated to flight control.

Information exchange in this scheme always constitutes, with the only exception of the image analysis results, communication between the robot and the ground station. With regard to communication, the MARVIN system shows a blackboard architecture implemented as *virtual shared memory*: The system state is defined as a globally unique data structure, which can be manipulated locally by every process. The communication software transparently propagates state changes throughout the system.

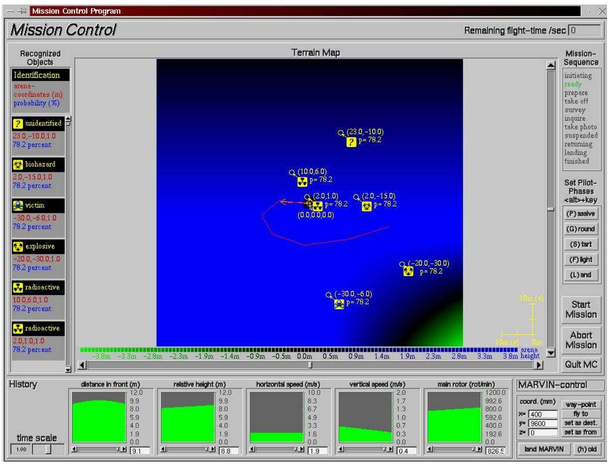


Figure 4: Screen-shot of the mission control software

4 Mission Strategy

This section deals with the mission strategy and its implementation in the mission control module.

4.1 *Dynamic Map and Image Targets*

Mission control operates on the basis of a dynamic object map covering the area of operation. This map contains the following three types of entries: recognized target objects, obstacles resp. threats, and image targets.

Target objects are dangerous drums and human bodies. All target objects may be recognized with certainty or just potential, based on the probability of recognition issued from the respective image analysis module. Mission control creates new entries of this type according to the image analysis results and takes care of fusing multiple recognitions of the same object.

Obstacles and threats are entered into the map according to the data from the threat sensors. Obstacles are modeled as cylinders, with x-y-center-point, height, and radius. Solid obstacles can be roughly located due to the distance measured by the radar and the robot's current heading. Flames have to be represented separately, since the flame sensor does not give any distance information – it is unable to distinguish between far big flames and near small ones. Therefore, mission control uses a *negative* map with a 1 m^2 raster, which is initialized to “fire everywhere”. Whenever the flame sensor detects *no* flame ahead, a view sector of 10° in width and 15 m in radius is cleared of “potential dangerous fire”. MARVIN's flight path planning is restricted to safe areas obtained in this way.

Image targets are desired locations for the taking of more images. Whenever a potential or uncertain object detection takes place, a new image target point above the object in question, at an altitude of 5 m, is entered into the map. The purpose of this is to take an image of the potential object that should allow to verify the first detection. Whenever the camera has notified mission control that an image has been taken close enough to the current desired image target location, mission control selects the next image target location based on the Euclidean distance from the robot's position. When the list of image targets becomes empty, there is nothing more to do for MARVIN – in this case, mission control guides the robot back to the starting point and then initiates an autonomous landing maneuver.

Figure 4 depicts the X-Window user interface of the mission control software with a number of object and target positions displayed.

4.2 Initialization

Initially, the list of image targets is set up as a rough search pattern with target points at an altitude of 50 m for overview images. From this altitude, potential drum and body shapes can be found, but the classification of the drum labels remains to the detailed images taken after the findings in the overview images.

4.3 Collision Avoidance

When mission control generates flight path segments to reach an image target location, it checks whether a segment intersects one of the obstacle cylinders. In this case, a new intermediate way-point is generated by lengthening the cylinder radius perpendicular to the intersecting path segment by 2 m, a safety distance for passing the obstacle. This process is repeated until no more intersections take place.

This scheme of collision avoidance involves communication back and forth between the ground station and the robot. Since the one-way communication delay between the robot and mission control can be shown to be below 400 ms [4] and mission control is able to calculate a new flight path in no more than 200 ms, 1 s is an upper bound on the response time in cooperative collision avoidance situations. MARVIN's travel speed is limited to 4 m/s, yielding a maximum "braking distance" of 10 m to a full stop. Thus, the radar's minimum detection range of 20 m is safe compared to a resulting stopping distance of 14 m ($4\frac{m}{s} \cdot 1\,s + 10\,m$).

Additionally, some obstacle avoidance behavior is provided in the flight control software for last-second measures: If the down-looking sonar detects a surface less than 1.5 m beneath the robot, the main rotor pitch is increased in order to gain height instantly. If the radar detects an obstacle no more than 12 m ahead, flight control switches to hovering mode immediately. In both cases, re-planning is not time-critical and left to mission control. Thus, collision avoidance is implemented cooperatively between the robot and the ground station.

5 Image Analysis

This section outlines the image analysis modules implemented and the methods they are based upon. The description is organized with respect to the classes of target objects.

5.1 Potential Objects in Overview Images

For the evaluation of overview images – which do not allow the recognition of drum labels – an algorithm has been implemented that searches for compact segments of a distinguishable color. The algorithm is based on Spann-Wilson-segmentation [8]. The desired range of segment sizes is computed from the altitude at that the image has been taken. Fortunately, drums and bodies are roughly similar in size.

5.2 Drum Labels

One algorithm for the detection of labels computes correlations between pre-defined label masks and the image at all possible positions and orientations. For efficiency, only binary images are used, generated from the camera images applying a locally adaptive threshold, and sparse pixel masks are utilized in the comparisons. Figure 5 shows a part of a real aerial picture with two recognized labels and a blind label.

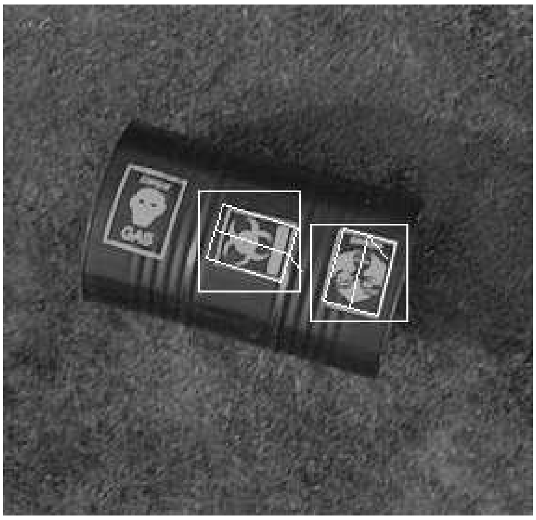


Figure 5: Example of label recognition in an aerial picture

The second approach to label detection collects segments within the binary image by edge tracing and checks if they are suitably sized and rectangular. The remaining segments are used to cut out the potential labels from the original image, which are finally validated through cross-correlation with example labels.

The third algorithm for finding labels calculates a number of central moments for every image segment, which are invariant to rotation and scaling. The resulting 7-dimensional feature vectors are classified by an euclidean distance classifier in order to identify them as complete labels or parts thereof.

5.3 Human Bodies

The recognition of persons in images and, especially, in image sequences has been subject to many research activities shortly. [3] presents a very useful survey of objectives and methods.

The first algorithm implemented to detect human bodies makes use of the empirical result that most of the dummies used in the competition wear red T-shirts. It simply looks for suitably sized image areas that are suitably red. This is a certainly incomplete solution, but nevertheless adequate to find *some* of the bodies.

The second, more sophisticated approach starts with edge detection and collects straight lines in parameter representation using a new specialized algorithm. The lines being edges of arms and legs are then determined according to their relative positions and angles.

The third algorithm determines segments of colors that are rare in the image, then performs edge-detection on these segments. The resulting contours are compared to a large number of sample contours by means of the chamfer distance [2].

6 Conclusion and Acknowledgements

This paper has described the strategy and implementation of mission planning and execution in the distributed autonomous robot system MARVIN. Additionally, the various image analysis algorithms, constituting the most important source of information for mission planning, have been explained.

Since the MARVIN system has been designed and built primarily by students during a number of project courses, the success of MARVIN is highly related to their commitment. We wish to thank the core team, consisting of Marc Batholomäus, Eike Berg, Carsten Deeg, Christian Fleischer, Christian Reinicke, Volker Remuß, Andreas Rose, Roland Stahn, Andreas Wege, and Stefan Wilhelmy. Of critical importance is Matthias Jeserich, our safety pilot.

The MARVIN project would not have been possible without financial, technical, and logistic support from various sources, including NovAtel Inc., Siemens AG, DaimlerChrysler Aerospace AG, SSM-Technik Peter Schröppel, Rotor Modellsport-Center, Tasking B.V., NLV Saarmund model airport, and TFH Berlin.

References

[1] THE MILLENNIAL EVENT: Rules for the 1999 International Aerial Robotics Competition qualifier. http://avdil.gtri.gatech.edu/AUVS/CurrentIARC/1999CollegiateRules.html.

[2] D. Gavrila and L. Davis. 3-D model-based tracking of humans in action: a multi-view approach. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pages 73–80, San Francisco, 1996.

[3] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, 1999.

[4] G. Hommel M. Musial, U. W. Brandenburg. Das Kommunikationskonzept für MARVIN, den autonom fliegenden Erkundungsroboter der TU Berlin. In *Robotik 2000*, Berlin, 2000. VDI GMA.

[5] M. Musial, G. Hommel, U. W. Brandenburg, E. Berg, M. Christmann, C. Fleischer, C. Reinicke, V. Remuß, S. Rönnecke, A. Wege. MARVIN – Technische Universität Berlin's flying robot competing at the IARC'99. In *Proc. AUVS Symposium*, Washington D.C., USA, 1999.

[6] Marek Musial, Uwe Wolfgang Brandenburg, Günter Hommel. MARVIN - Ein autonom fliegender Erkundungsroboter. In *Autonome Mobile Systeme 1998, 14. Fachgespräch*, pages 226–233, Karlsruhe, 1998. Springer-Verlag.

[7] Marek Musial, Uwe Wolfgang Brandenburg, Günter Hommel. MARVIN - Der autonom fliegende Erkundungsroboter der TU Berlin und sein Erfolg beim Wettbewerb IARC'99. In *15. Fachgespräch Autonome Mobile Systeme (AMS99)*, München, 1999. Springer-Verlag.

[8] M. Spann and R. Wilson. A quad-tree approach to image segmentation which combines statistical and spatial information. *Pattern Recognition*, 18(3/4):257–269, 1985.

[9] U. W. Brandenburg, M. Finke, D. Hanisch, M. Musial, R. Stenzel. TUBROB – An autonomously flying robot. In *Proc. AUVS Symposium*, pages 627–636, Washington D.C., USA, 1995.

[10] U. W. Brandenburg, M. Finke, M. Musial. Aufbau und Steuerung des fliegenden Roboters TUBROB. In *Tagungsband 11. Fachgespräch Autonome Mobile Systeme*, pages 100–109, Karlsruhe, 1995. Springer-Verlag.

[11] U. W. Brandenburg, M. Musial, G. Hommel. MARVIN – Technische Universität Berlin's never-depressed flying robot for the IARC'98. In *Proc. AUVS Symposium*, Washington D.C., USA, 1998.