

International CBRNE Institute

# ISMCR 2018

21th International Symposium on  
Measurement and Control in Robotics  
26-28 September 2018 – International  
CBRNE Institute, MONS, Belgium

ROBOTICS FOR THE CHANGING WORLD



## BEMEKO

[www.imeko.org](http://www.imeko.org) – TC 17



European Society for Defense	Hainaut Sécurité	Belgian Royal Higher Institute of Defense	Climbing and Walking Robots Association	

## PROGRAM

### SESSION 1. WELCOME and INTRODUCTION TO THE MOBILE ROBOTICS 26 September (Room 1)

**CHAIR: Professor L.Van Biesen (VUB/BEMEKO)n Prof Dr Ir Dehombreux (FPMs) Dr Zafar Taqvi (Srt IMEKO/TC17), Professor Y.Baudoin (ICI)**

10.00-10.30H	Welcome, Military Robotics, ELROB Status, ICI aims	Madame Vanwijnsberghe (Director Hainaut Sécurité), Dr F.Schneider (Fraunhofer, Germany), M.Y.Dubucq (Director ICI), Prof dr Ir H. Christensen,
10.30-11.00H	KEY-NOTE Autonomous Robot for Gas and Oil Sites (ARGOS): Total's Lessons Learnt and Next Steps	Kris KYDD Head of Robotics Total ARGOS Challenge
11.00-11.30H	KEY-NOTE Robotics and civilian emergency response : how lessons learned empower Incident command systems organization	LtCol JP Monet (BDFRD, France), Maj E.Rodriguez (BDFRD), Capt S.Mozziconacci (BDFRD), E.Dombre (Emeritus CNRS Research Director, Montpellier University
11.30-12.00H	Robotics for facing CBRNE risks	Dr J.Galatas (KEMEA, Greece, CBRN-KC)m Y.Baudoin (E-KC)
13.30-17.00H	Robotics competition and Exhibition	(ELROB, ICI)
<b>19.30-22.00H</b>	<b>GET TOGETHER ISMCR at the Hotel MONS</b>	

### SESSION 2/1 IMPROVEMENT of ROBOTICS 27 September (Room 1)

**CHAIR : Professor EM Y.Baudoin (ICI/RMA)**

10.00-10.30H	KEY-NOTE Unmanned vehicle systems in unstructured environments : challenges and current status	Prof V.Gradetsky (IPMNET, Moscou, Russia)
10.30-10.50H	<i>IMU based gesture recognition for mobile robot control using Online Lazy Neighborhood Graph search</i>	<i>Padmaja Kulkarni (Fraunhofer-Institute, Germany), B. Illing, B. Gaspers, B. Brüggemann, D. Schulz</i>
10.50-11.10H	<i>A Novel Data Fusion architecture for Unmanned vehicles</i>	<i>I. L. Ermolov, Institute for Problems in Mechanics of the Russian Academy of Sciences, Russia.</i>
11.10-11.30H	<i>Coverage Path Planning by swarm;- of UAV by swarm of UGV for traversability analysis</i>	<i>L. Cantelli, D.C. Guastella, D. Longo, C.D. Melita, G. Muscato University of Catania</i>
11.30-11.50H	<i>Robot control based on human motion analysis with IMU measurements</i>	<i>Robin Pellois, Laura Joris and Olivier Bruls Multibody and Mechatronic Systems Laboratory University of Liege, Belgium</i>
11.50-12.10H	<i>Development of the Modular Platform for Educational Robotics.</i>	<i>A.G. Semenyaka, Moscow State Technological University "STANKIN"</i>

--	--	--

**SESSION 2/2 MEASUREMENT and CONTROL in ROBOTICS**  
**27 September (Room 2)**

**CHAIR: Professor B. Kiss (BME/Hungary)**

10.00-10.30H	KEY-NOTE: JIZAI Body Design of ultra-body fit for super smart society	Prof Masahiko Inami, , Japan)
10.30-10.50H	<i>Enhancing Bodily Expression and communication Capacity of Tele-existence Robot with augmented reality</i>	<i>Yasuyuki Inoue (University of Tokyo, MHD YamenSaraiji (Keio University, Japan), Fumihiro Kato and Susumu Tachi (University of Tokyo)</i>
10.50-11.00H	<i>Semantic Grid Mapping based on Surface Classification with Supervised Learning</i>	<i>Torsten Engler (Universität der Bundeswehr München Institut für Technik Autonomer Systeme (TAS)</i>
11.00-11.20H	<i>Pre-filter to robustify the exact linearization based tracking controller of a SCARA type robot</i>	<i>Na Wang, Balint Kiss Budapest University of Technology and Economics, , Hungary</i>
11.00-11.20H	<i>Feedforward command computation of a3D flexible robot</i>	<i>Arthur Lismonde and Olivier Bruls Department of Aerospace and Mechanical Engineering, University of Liege,Belgium</i>
11.20-11.40H	<i>Effectiveness test of simulator for e-training in carrying out missions with use of tele-operated vehicles</i>	<i>Igor Ostrowski, Andrzej Masłowski NASK Governmental Research Institute Digital Mobile Robotics Department Warsaw, Poland</i>
11.40-12.00H	<i>Training of robots' operators with use of multirobot simulators</i>	<i>Marek Kacprzak, IMMSF, Warsaw, Poland</i>

**SESSION 3/1 ROBOTICS for DEFENSE and SECURITY - STATUS**  
**27 septembre (Room 1)**

**CHAIR: Professor Y.Baudoin (ICI-RMA) – Dr Ir F.Schneider (Fraunhofer)**

13.15-13.40H	UGV-UAV Growing Market for the Defense	Prof EM Y.Baudoin (ICI/RMA)
13.40-14.10H	In-flight launch of unmanned aerial vehicles  Qualitative and quantitative validation of drone detection systems	Dr Ir Geert De Cubber (Royal Military Academy Belgium)
14.10-17.00H	<i>Competition/Exhibition EOD/IED trials</i>	

**SESSION 3/2. CONTROL and SENSOR SYSTEMS in ROBOTICS**  
**27 September (Room 1)**

**CHAIR: Prof A.Maslowski (NASK/Poland)**

14.10-14.40H	KEY NOTE: Measurements for the Forthcoming Future	Dr Zafar Taqvin Scientific Secretary IMEKO TC 17 (USA)
14.40-15.00H	<i>Dedicated simulator for e-training of demining robot "Dromader" operators.</i>	<i>Igor Ostrowski, Andrzej Masłowski NASK Governmental Research Institute Digital Mobile Robotics Department Warsaw, Poland</i>
15.00-15.20H	<i>An active beacon-based Tracking System to be used for Mobile Robot Convoying</i>	<i>Stanislaw Goll, Elena Zakharova, LLC KB Avrora/Ryazan State radio Engineering University, Ryazan, Russia</i>
15.20-15.40H	<i>RADAR-based Through-Wall Mapping</i>	<i>Sedat Dogru, Lino Marques, ISR-University Coimbra, Portugal</i>
15.40-16.00H	<i>Ultrasonic Rangefinder with submillimeter resolution as part of the Rescue Robot's sensor system</i>	<i>Stanislaw Goll, Julia Maximova, LLC KB Avrora/Ryazan State radio Engineering University, Ryazan, Russia</i>

**SESSION 4. TECHNICAL PRESENTATIONS and MODELING**  
**26-27 September ( Exhibition Site)**  
**Demos in competition 24-27 Sep**  
**IMEKO TC17 meeting**  
**And Room 2 (schedule later adapted, depending on the scenarios)**

**CHAIRS: Dr F.Schneider (Fraunhofer/Germany), Dr Z.Taqvi, Y.Baudoin**  
**In RED: exhibitors (updated 15 July)**

14.00-14.20H	<i>ELROB 2018. Convoy and Mule of Team MuCAR</i>	<i>Thorsten Luettel, University of the Bunderswehr, Munich, Germany F. Ebert, P. Berthold, P. Burger, T. Engler, A. Frericks, B. C. Heinrich, J. Kallwies, M. Kusenbach, K. Metzger, M. Michaelis, B. Naujoks, A. Sticht, and H.-J. Wuensche</i>
14.20- 14.40H	<i>Standard Test Methods for Mobile Robots</i>	<i>Andreas Ciossek, Produkt Manager, TELEROB, Germany</i>
14.40-15.00H	<i>Automated Magnetic Field reproducing Stand for Debugging Algorithms Navigation of Mobile Robots.which use on board Magnetometer Sensor</i>	<i>Stanislaw Goll, Alexander Borisov, LLC KB Avrora/Ryazan State radio Engineering University, Ryazan, Russia</i>
15.00-15.20H	<i>Robsim Software for Mobile Robots modeling</i>	<i>O.P. Goidin, S.A. Sobolnikov  FSUE VNIA, Moscow</i>
10.00-17.00H	<b>The ZEUS robot</b>	<b>Steve Wisbey NIC Instruments Ltd</b>
<b>15.30H MEETING IMEKO TC17</b>		
10.00-17.00H	The MSAS vehicle	Janusz Bedkowski, Manadal, Poland
10.00-17.00H	Telex PRO, Hybrid, PLUS	Andreas Ciossek, Telerob, Germany
10.00-17.00H	The Mörri robot	Antti Tikanmäki, BISG Oulu ,

		University of Oulu Finland
10.00-17.00H	The SR-120D System	Patrik Bylin, Brokk AB, Sweden
10.00-17.00H	The Packbot 510 EOD/Kobra 710	Colin Weiss, ELP GmbH, Germany
10.00-17.00H	Tulf/StrAsRob, smart military vehicles	Dr Alexander Wolf, Diehl Defense GmbH, Germany
10.00-17.00H	Milrem THeMIS	Dr Alexandre Wolf, Diehl Defense GmbH&CoKG
10.00-17.00H	Patria AMV, SLO-IFV	Matti Saarikko, Patria land Systems Oy, Finland
10.00-17.00H	The Wombat Leader-Wombatt Follower	Gol Stanislav, LLC KB Avrora, Russia
10.00-17.00H	The robot LongCross	Bastian Gaspers, Fraunhofer, Germany
10.00-17.00H	TAUT	Reinhard stocker, Guenther Tratting, Austria

**NOTE: a SHUTTLE BUS is foreseen during the symposium for transportation of participants to the ELROB stand (standing lunch and competition)**

# Effectiveness test of simulator for e-training in carrying out missions with use of tele-operated vehicles

**I.Ostrowski, A.Masłowski**

Research and Academic Computer Network (NASK),  
Kolska Str. 12 01-045 Warsaw, Poland

<sup>1</sup>E-mail: [iostrowski@wp.pl](mailto:iostrowski@wp.pl)

**Abstract.** This paper describes results of effectiveness test of simulator for e-training operators of tele-operated vehicles. The test has been conducted with 20 participants (Warsaw Military University of Technology students) divided into two groups. First group, numbering 11 persons, operated Dromader robot without previous training on the simulator. Members of the second 9-persons group went through simulator training before operating the robot. The trial was composed of two exercises. First of them was devoted to test driving abilities of the operators. The task for operator i was to cover the distance in a short time and not making errors (such as deviations from the route or collisions with obstacles) that was scored negatively. The second exercise was devoted to test manipulation abilities, and manipulation concrete blocks with robot's manipulator was its substance.

## 1. Introduction

We assumed that training with simulator can make time of training with robot shorter as have been written in [1],[3]. Described in this paper results of test prove that We were right. Dromader is a robot with uneasy user interface. Training of use the interface can be conducted without wasting fuel and even without use of robot.

## 2. The Dromader robot

Dromader robot (Figure 1) has two-segment chassis connected by an articulated joint. The rear section houses a 15kW combustion drive unit while the front section houses the control system and several components of the platform's drive system. Dromader is equipped with manipulator, mounted on the front section, and with three cameras. A typical gamepad serves as control console for robot operation.



Figure 1 Dromader robot. Robot's cameras marked as 1,2,3

Dromader is operated from an operation stand, with use of camera views. The operation stand engaged during validation trial is presented on the Figure 2.



Figure 2 Dromader's operation stand

For training of Dromader's operators serves simulator developed as TIRAMISU tool. The training stand with the simulator, engaged during validation trial, is presented on the Figure 3, and a view on simulator screen is presented on the Figure 4.



Figure 3 Dromader's training Stand



Figure 4 Simulator's screen

### 3. The course of validation trial

The validation trial has been conducted with 20 participants (WAT's students) divided into two groups. First group, numbering 11 persons, operated Dromader without previous training on the simulator. Members of the second 9-persons group went through simulator training before operating the robot.

The trial was composed of two exercises. First of them was devoted to test driving abilities of the operators. The route to go through this exercise is presented on the Fig.5. The task for operator it was to cover the distance in a short time and not making errors (such as deviations from the route or collisions with obstacles) that was scored negatively, corresponding to Maslowski's work [2] .



Figure 5 The route for testing abilities to drive Dromader

The second exercise was devoted to test manipulation abilities, and manipulation concrete blocks with robot's manipulator was its substance. The exercise, presented on the Fig.6, consisted in driving up certain block, localizing it and grasping (I), and then carrying it to the certain distance (II); errors were scored negatively as well. The block being the manipulation subject was located near a vertical wall making up an obstacle, as well as other concrete objects in the neighborhood did.

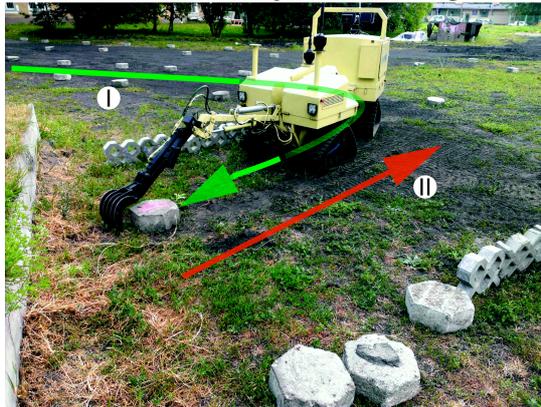


Figure 6 The exercise for testing manipulation abilities

#### 4. Results

Both, in the manipulation and driving exercise, the results have been obtained by comparison of the outcomes of participants of two groups – with and without of previous training on the simulator. In both exercises the time of fulfilling the task, and the number of errors made, were treated as the outcome. Global outcome in relation to the time was calculated in two ways, as an arithmetic mean and as a median, according to the following formulas:

Arithmetic mean:

$$x_{sr} = \sum_1^n \frac{x}{n}$$

$$\text{Median } m_e = \begin{cases} x_{\frac{n+1}{2}} & \text{gdyn jest nieparzyste} \\ \frac{1}{2} \left( x_{\frac{n}{2}} + x_{\frac{n}{2}+1} \right) & \text{gdyn jest parzyste} \end{cases}$$

where:  $n$  – number of participants;  $x$  – time of fulfilling the task.

The results obtained in the driving exercise are shown in the following tables.

Table 1. Driving outcomes of participants trained on the simulator

Participant	Time of fulfilling the task [s]	Time of training [min]	Number of errors	Arithmetic mean $x_{sresz}$ [s]	Median $m_{esz}$ [s]
1	217	40	0	249	243
2	302	20	1		
3	243	30	3		
4	245	30	3		
5	385	30	4		
6	193	60	4		
7	190	60	5		
8	246	30	5		
9	216	30	0		

After rejecting extreme values, arithmetic mean and median amount adequately 232 and 230.

Table 2. Driving outcomes of participants not trained on the simulator

Participant	Time of fulfilling the task [s]	Number of errors	Arithmetic mean $x_{srnsz}$ [s]	Median $m_{ensz}$ [s]
1	280	3	319	308
2	377	4		
3	324	4		
4	348	8		
5	260	4		
6	308	3		
7	262	2		
8	455	7		
9	254	0		
10	247	0		
11	398	8		

After rejecting extreme values, arithmetic mean and median amount adequately 306 and 294.

Percentage differences between outcomes of two groups were calculated for achieved times according to the following formulas.

$$\Delta x_{sr} = \frac{x_{srmsz} - x_{sr\bar{s}z}}{x_{srmsz}} \cdot 100\%$$

$$\Delta m_e = \frac{m_{ensz} - m_{es\bar{z}}}{m_{ensz}} \cdot 100\%$$

Obtained values are the following:

$\Delta x_{sr} = 22\%$ , and for rejected extreme values 24%

$\Delta m_e = 21\%$ , and for rejected extreme values 22%

Percentage differences  $\Delta e$  between outcomes of two groups for errors made during the driving exercise, calculated in the same way, are the following:

$\Delta e = 29\%$  for total number of errors

$\Delta e = 37\%$  for maximal number of errors made by single participant

The results obtained in the manipulation exercise are shown in the following tables.

Table 3. Manipulation outcomes of participants trained on the simulator

Participant	Time of fulfilling the task [s]	Time of training [min]	Number of errors	Arithmetic mean $x_{srmsz}$ [s]	Median $m_{ensz}$ [s]
1	207	40	1	190	192
2	175	20	3		
3	231	30	0		
4	143	30	0		
5	233	30	3		
6	174	60	1		
7	192	60	3		
8	165	30	0		
9	192	30	1		

Table 4. Manipulation outcomes of participants not trained on the simulator

Participant	Time of fulfilling the task [s]	Number of errors	Arithmetic mean $x_{srmsz}$ [s]	Median $m_{ensz}$ [s]
1	305	4	341	334
2	414	2		
3	283	4		
4	334	1		
5	383	1		
6	294	1		
7	316	2		
8	415	5		
9	249	1		
10	403	2		
11	354	0		

Percentage differences between outcomes of two groups for achieved times of fulfilling the task, calculated in the same way as for the driving exercise, are the following:

$$\Delta x_{sr} = 44\%$$

$$\Delta me = 42\%$$

Percentage differences  $\Delta e$  between outcomes of two groups for errors made during the manipulation exercise, calculated in the same way, are the following:

$\Delta e = 40\%$  for total number of errors and for maximal number of errors made by single participant

## 5. Conclusion

The test described in this paper confirm that using simulators in training robot operators is effective way to reduce cost of the training, and let trainee make errors with lower level of stress. Comparing of using real machine during first lessons of robot use it is much faster way to get satisfied results. Validation was conducted by determining how the simulator under consideration meets requirements of its performance threshold. For the simulator triple-value threshold has been set with 3 performance levels:

Level U - unsuitable tool - less than 10% difference between groups' outcomes;

Level I - tool needs improvement - difference between 10% and 30%;

Level S - suitable tool – the difference bigger than 30%.

Differences of outcomes in driving exercises are in the 21%-24% bracket for times of fulfilling the task, and in 29%-37% bracket for number of errors made.

Differences of outcomes in manipulation exercises are in the 42%-44% bracket for times of fulfilling the task, and come to 40% for number of errors made.

Taking into account that time of fulfilling the task of approach in demining missions is of second importance [4], it's justified to decide level S obtained by the trainer-simulator as a result of the validation trial.

## 6. References

- [1] Łuczyński T., Ostrowski I., Będkowski J., Masłowski A., Przybyłko M., Zoppi M., Wojcieszynska P., Szczepaniak M., (2013) Mobile Demining Devices Modeling for the Training Purpose using VORTEX Framework, Proc. of the Int. Conf. on Military Engineering – Problems and Prospective, 23-25 April, 2013, Institute of Military Engineering, Wrocław, Poland, pp. 233-240.
- [2] Masłowski, A. (2014) Training in Military Robotics and EOD Unmanned Systems. Conference and Seminars, 30 September - 2 October 2014, NATO EOD Center of Excellence, Trenčín, Slovakia.
- [3] Methodology of Training and Support for Urban Search and Rescue With Robots Bedkowski J., Majek K., Ostrowski I., Musialik P., Masłowski A., Adamek, A., Coelho, A., De Cubber, G., ICAS 2013, "The Ninth International Conference on Autonomic and Autonomous Systems", Lisbon 2013, Portugal
- [4] Kacprzak, M., Kaczmarczyk, A., Masłowski, A. (2012) Computer training for operators of unmanned vehicles in application to Humanitarian Demining activities. 6th IARP RISE, Warsaw.

# Dedicated simulator for e-training of robot “Dromader”

**I.Ostrowski, A.Masłowski**

Research and Academic Computer Network (NASK),  
Kolska Str. 12 01-045 Warsaw, Poland

E-mail: iostrowski@wp.pl

**Abstract.** Project of robot in simulation system can be divided into two main parts. First is graphical model, and second is physical model. This paper discusses results of using tools for 3D modelling and building physical models. Graphical models are developed using 3dsMax software from CAD model. It is possible to perform straight conversion from the CAD model to a graphical game-ready model but results are poor. The most serious issues were performance, and texturing. Graphical models had to be created from scratch. Mechanical part was developed with Vortex Editor. Vortex Editor is tool which allows creating mechanical system for Vortex physics universe. It is CAD like software, which allows creating project, running test simulation, and tuning parameters without exporting model to simulation framework.

## 1. Introduction

This paper describes simulator of UGV Dromader. build for training purposes. This UGV is very hard to operate and need long time of training for operators. Our simulator is shortening time of use robot for training purposes.

## 2. UGV Dromader

UGV Dromader robot (Figure 1) was designed and built by Military University of Technology, Warsaw. Robot's structure is characterized by a segmented chassis connected using an articulated joint. The turning angle of the front segment in relation to the rear can reach up to 90°. This provides high maneuverability, allowing for easy turns in confined spaces. In conjunction with the track drive system and both axis drive it allowed the “Dromader” to achieve a very good off-road mobility.



Figure 1 Robot Dromader

The rear section houses a 15kW combustion drive unit while the front section houses the control system, and several components of the platform's drive system. Because of the desire to minimize the weight of the robot, the entire chassis of the robot is made of aluminium.

The robot may be equipped with an experimental protection system against remotely detonated IEDs, which may be mounted in the rear section above the engine inside the chassis. Using this configuration requires the detector and jammer antennas to be mounted in very specific locations to avoid interference and not to reduce their working area due to chassis structure and other equipment mounted on the robot. The rear segment also contains a Wi-Fi transceiver antenna (2.4 GHz) for the remote-control system.

The manipulator had been mounted on the front section of the robot. Part of its structure and location requirements were that in its transport position it would not obscure the operator's field of view registered by the teleoperation system.

If you don't wish to use the Word template provided, please set the margins of your Word document as follows. The manipulator has three degrees of freedom allowing for control of the effector's position in the vertical plane. Due to weight limitations, the horizontal plane control is achieved through turning of the entire front section instead of an additional control mechanism.

### 3. Dromader simulator

#### 3.1. Simulation framework description

Currently developed simulation system is a universal tool for multi-robot operation. Created software solution can be divided into two sub-projects. First is simulation server which is standalone binary package. This program runs a simulation server; this process is responsible for physics solving, multiple-client scene synchronization and watching mission progress. Second is a library which allows communicating with the simulation server. This software is dynamic linked library. This allows separating simulation and trainees' consoles, so simulation process can be run on multiple machines robustly. Another important advantage of presented solution is fact, that rendering scene with multiple cameras can be heavy task for computer, but it no longer affects physics simulation. Scene is synchronized via mixed TCP and UDP protocol. It allows connecting up to ten clients.

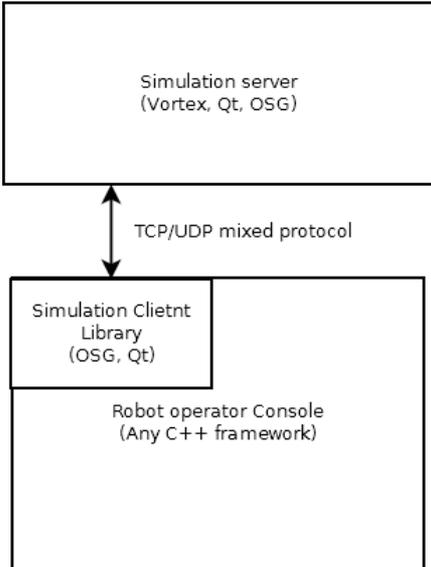


Figure 2 Simulation system concept

For physics simulation CM-Labs Vortex library is used. This library is solution for serious games, industry simulation (like cranes, excavators) and military training. It comes with a mature framework which allows developing simulation of almost any kind of vehicle. Framework allows simulating internal combustions engines, electric motor, variety of transmission systems including classical manual and hydrostatics transmissions, and nearly any kinematic layout. Also it is possible to simulation complex structure like robotics manipulators. Moreover some situation in simulation can be scripted using Python language. Library is proprietary and closed source, it is well integrated with Open Scene Graph.

OSG, another library, is open source, OpenGL based graphics engine. It is C++ library which is very often chosen in serious games and training system. This library provide number of useful tools like: a hierarchical structure of the scene, a hierarchical structure of graphical model with number types of relationship between model parts, graphical models loaders including proprietary formats like FLT, Collada or FBX, levels of details. It also supports OpenGL implementation from 1.1 up to 4.0 including OpenGL ES. Next part of simulation framework is Qt 5 library. This popular LGPL licensed open source C++ framework. It is widely used for GUI application, but it provides plenty solutions for non GUI developers. In project QtXML and QtNetwork modules were heavily used. Last used library is Boost.Python. This library is part of Boost open source set of C++ libraries. Boost.Python allows exposing some dynamically allocated C++ objects in Python name-space. This allows creating tools which can be used during execution of simulated mission scenario. For example, during mission scenario development, some triggers can be prepared. Those triggers can cause number of actions, like adding penalty points, time measurement and so on. Because of Python language, compilation is not needed. Code is loaded and interpreted by integrated Python parser, in runtime.

Simulation client used only OSG and Qt libraries. It makes this solution GUI – ready, multi-platform and lightweight. It can be easily integrated with operator or trainer console. This library deals with recreating graphical scene in client software, rendering camera views, and sending and receiving data from and to robots. It also has methods to receive some messages from simulation server, which can be sent from scenario's script executed on server's side.

### 3.2. Dromader robot simulation

Project of robot in simulation system can be divided into two main parts. First is graphical model, and second is physical model. It is important that second part is based on created graphical model. Graphical models are developed using 3dsMax software from CAD model. It is possible to perform straight conversion from the CAD model to a graphical game-ready model but results are poor. The most serious issues were performance, and texturing. Graphical models had to be created from scratch. Mechanical part was developed with Vortex Editor. Vortex Editor is tool which allows creating mechanical system for Vortex physics universe. It is CAD like software, which allows creating project, running test simulation, and tuning parameters without exporting model to simulation framework.

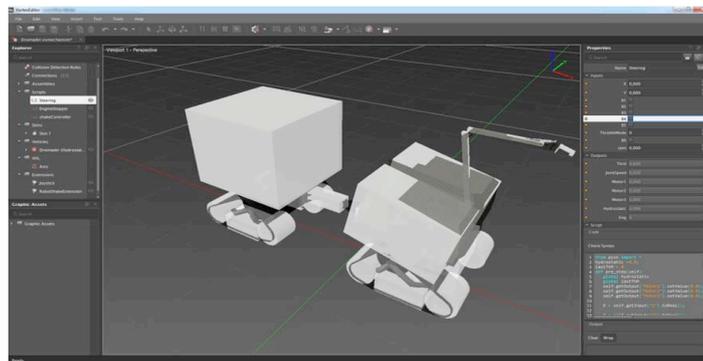


Figure 3 The physical model in Vortex Editor

The graphical model is organized in hierarchical structure, where every moveable part has own node, with own coordinate system. During physics modelling number of properties were added. First of all collision geometries were created. Collision geometries are simple primitives like boxes, cylinders that interact with each other's. Collisions are not resolved between graphical meshes. These shapes are organized in to parts. Every collision shape has own mass, inertia matrix and material. Afterwards, constraints were created. These are objects, which create connection between two or more parts. It could be for example prismatic, or hinge pair. A number of parameters describe constraint, like character (free, motorized or locked), internal friction, damping and stiffness, motor or blockage maximum force, and limits. In this fashion hydraulics actuators were added to the simulation. Next robot-vehicle system was added to physics model. There are some typical vehicle template like cars (AWD, RWD, FWD with auto and manual transmission), tracked vehicles with skid-steering like excavator (hydrostatics torque division) or tank (differential torque division). Robot Dromader is a unique vehicle solution so new template was created.

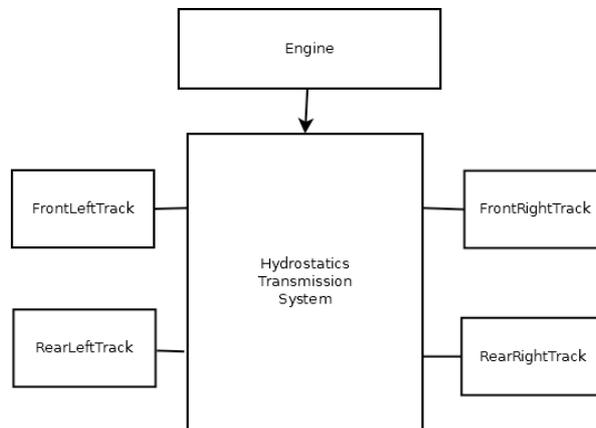


Figure 4 Robot Dromader system template

Template, after loading to Vortex Editor automatically creates such complex structures like AWD truck. Objects like engine, differentials, shafts, transmission system, driving constraints and finally wheels with suspension are introduced, and put in proper relationships. Structure of Dromader robot is pretty simple. The engine is connected to the hydrostatics transmission, and ratio of every output motor can be set separately. The robot doesn't have differential steering (like excavator), so ratio for each track is the same. Driving is realized using hinge constraint with velocity actuator. Final layer of physics model is logic, which does not have to be hard coded in application code. Every robot's object can communicate with Python scripts. These scripts are an elegant way to implement some logics. Example – Dromader robot has three driving modes. Switching it affects motor velocity, but also reconfigures hydrostatics system, connecting motors parallel, thus higher torque is generated. This function cannot be implemented directly, but can be realized as script which would change hydrostatics system ratio in proper way.

Last, but not least is VHL interface. This tool allows creating a universal interface to communicate vehicle model with rest of application. After loading model in final simulation application it allows to write and read parameters from C++ code.

### 3.3. Environment

Sample environment was created from point cloud. It's much easier to create real location model using geodetic data than only photos. Environment also was created using 3dsMax. For buildings floor plan was extracted from point cloud data, for ground shape an elevation map was created. Vegetation was replaced with simple billboard representation. Billboard is a graphics method for creating planar shapes that always face viewer's camera. This method is widely used for simple vegetation rendering

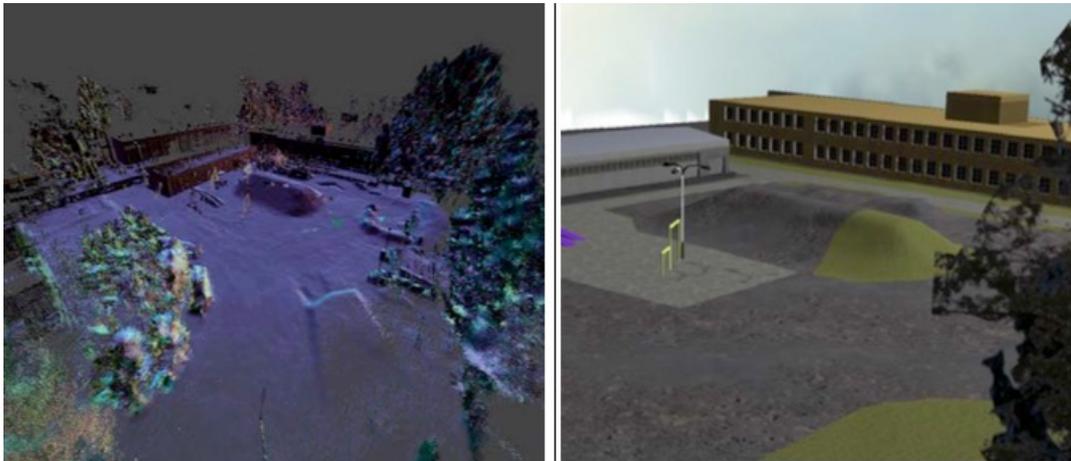


Figure 5 Environment representation - left point cloud, right 3d model in Vortex simulator

Scene was textured, and finally imported to Vortex editor where materials parameters were applied to surface.

Configuration is object orientated XML file which simulation server loads during initialization. These files contain a great deal of information, like spawn position of robots, scene type, position 3rd person view cameras, and so on. It allows to set-up, or modify simulation scene in very convenient way. What is more, XML file contains a scripts coded in Python language, that allows to interact with simulation, so some events can be dynamically created. It can create a lot of opportunities to develop live- like, interactive scenarios.

### 3.4. Operator console

Operator console, as was mentioned before, had been created using Simulation Client Library. The Dromader operator's console is hand-held computer with game-style controller. It is communicate with robot using wireless connection. Operator can observe one camera at the moment, and steering robot and its manipulator using analog sticks and buttons on the controller. Operations like connection, disconnection, switching on and off robot's system are realized via touch screen GUI. Due to lack of access to source code of operator's console, interface was recreated using own framework. This program was successfully integrated with Simulation Client Library, and finally real-like operator's console was introduced. Separation of Simulation Client Library and console's code is great advantage, because it makes simulator independent from used input and output devices and allows performing integration with real equipment.



Figure 6 operators console connected to simulator server

#### 4. References

- [1] Masłowski, A. (2014) Training in Military Robotics and EOD Unmanned Systems. Conference and Seminars, 30 September - 2 October 2014, NATO EOD Center of Excellence, Trenčín, Slovakia.
- [2] Łuczyński T., Ostrowski I., Będkowski J., Maśłowski A., Przybyłko M., Zoppi M., Wojcieszewska P., Szczepaniak M., (2013) Mobile Demining Devices Modeling for the Training Purpose using VORTEX Framework, Proc. of the Int. Conf. on Military Engineering – Problems and Prospective, 23-25 April, 2013, Institute of Military Engineering, Wrocław, Poland, pp. 233-240.
- [3] Kaczmarczyk, A., Kacprzak, M., Maśłowski, A. (2009a) Multi-level simulation training of devices operators. An application to mobile robots operators (in Polish). *Elektronika* 11/2009.
- [4] Będkowski J., Majek K., Ostrowski I., Musialik P., Masłowski A., Adamek, A., Coelho, A., De Cubber, G., "Methodology of training and support for urban search and rescue with robots" ICAS 2013, "The Ninth International Conference on Autonomic and Autonomous Systems", Lisbon 2013, Portugal
- [5] Ostrowski, Igor, et al. "Training tools for TIRAMISU Project." Annual Report fp7-tiramisu.eu
- [6] Musialik, P., Będkowski, J., Ostrowski, I., Łuczyński, T., Majek, K., Maśłowski, A., & Adamek, A. (2015). Virtual environment modeling for training of unmanned vehicles' operators. *Zeszyty Naukowe/Wyższa Szkoła Oficerska Sił Powietrznych*.

# A Novel Data Fusion Architecture for Unmanned Vehicles

**I Ermolov**<sup>1</sup>

Senior Researcher,

Institute for Problems in Mechanics of the Russian Academy of Sciences,  
119526, Moscow, Russia.

E-mail: ermolov@ipmnet.ru

**Abstract.** An effective functioning of unmanned vehicles demands to process large amounts of various data coming from sensors, on-board data bases etc. Therefore data fusion technology is one of key-technologies for autonomous vehicles and systems. In order to systemize such data processing special so-called data fusion architectures are used (e.g. JDL, Waterfall, Boyd etc.). However some of those have a list of restrictions w. A goal of this paper is to present a novel data fusion architecture which could be used on board of unmanned vehicles. This architecture consists of 5 basic layers: parameters identification, state identification, (object type identification), situation identification and task implementation identification. The proposed architecture has some advantages in comparison to those already in use. Author considers that presented architecture has good visibility, intuitive understanding, possibility for deep feedback usage and good potential for automatic reconfiguration and self-learning. The developed data fusion architecture can be used for building complex data fusion systems on board of unmanned vehicles as well as of group of vehicles and even of systems of higher hierarchy.

## 1. Introduction

Data fusion by itself can be defined as a process of information generalization based on more than one source of information. Data fusion and sensor fusion (as its particular case) are among key-technologies towards autonomy of robots [2].

In [8] it was proposed to distinguish following cases of data fusion in Unmanned systems (figure 1):

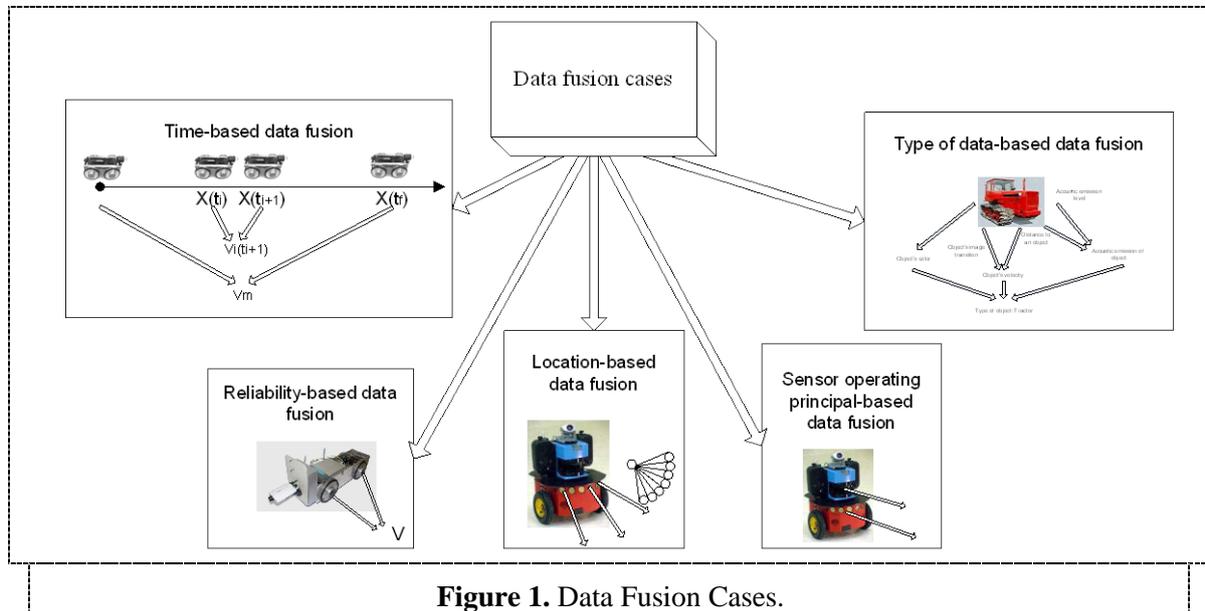
*Time-based data fusion.* While tracking variation of some parameter in time sequences it becomes possible to estimate other parameters of UV. Another case of such data fusion is data filtering. For more details see [3].

*Reliability-based data fusion.* While fusing data from several sensors with low-reliability characteristics it becomes possible to get highly reliable information.

*Location-based data fusion.* By fusing information from several sensors each one with narrow working space one may get information of the large working space [7]. Another case of space-based data fusion is fusing data from dispensed sensors which gives us new information.

---

*Sensor operating principal-based data fusion.* Fusion of data from sensors measuring same parameter but functioning on various principles gives information with higher reliability factor.



**Figure 1. Data Fusion Cases.**

Type of data-based data fusion. This one is used to produce information basing on data of various type (e.g. fusing information from video sensor with information from laser sensor). This type of fusion is intensively used especially for object recognition.

Analysing this classification it can be confirmed that most of the data fusion cases in unmanned systems can be described by these fusion cases or by their combination.

Modern UVs are extremely complex systems with large number of data flows [5]. In order to analyze such amount of data flow it is necessary to adjust those data flows. This is known as data fusion architectures. There exists variety of such architectures: JDL, Boyd model, LAAS, The Omnibus Model, Waterfall model and some others [7].

However according to author these architectures lack some intuitiveness, logic and flexibility.

## 2. Hierarchical Data Fusion Architecture

Author of this paper proposes following Hierarchical data fusion architecture (Figure 2). Here author describes layers' term as following:

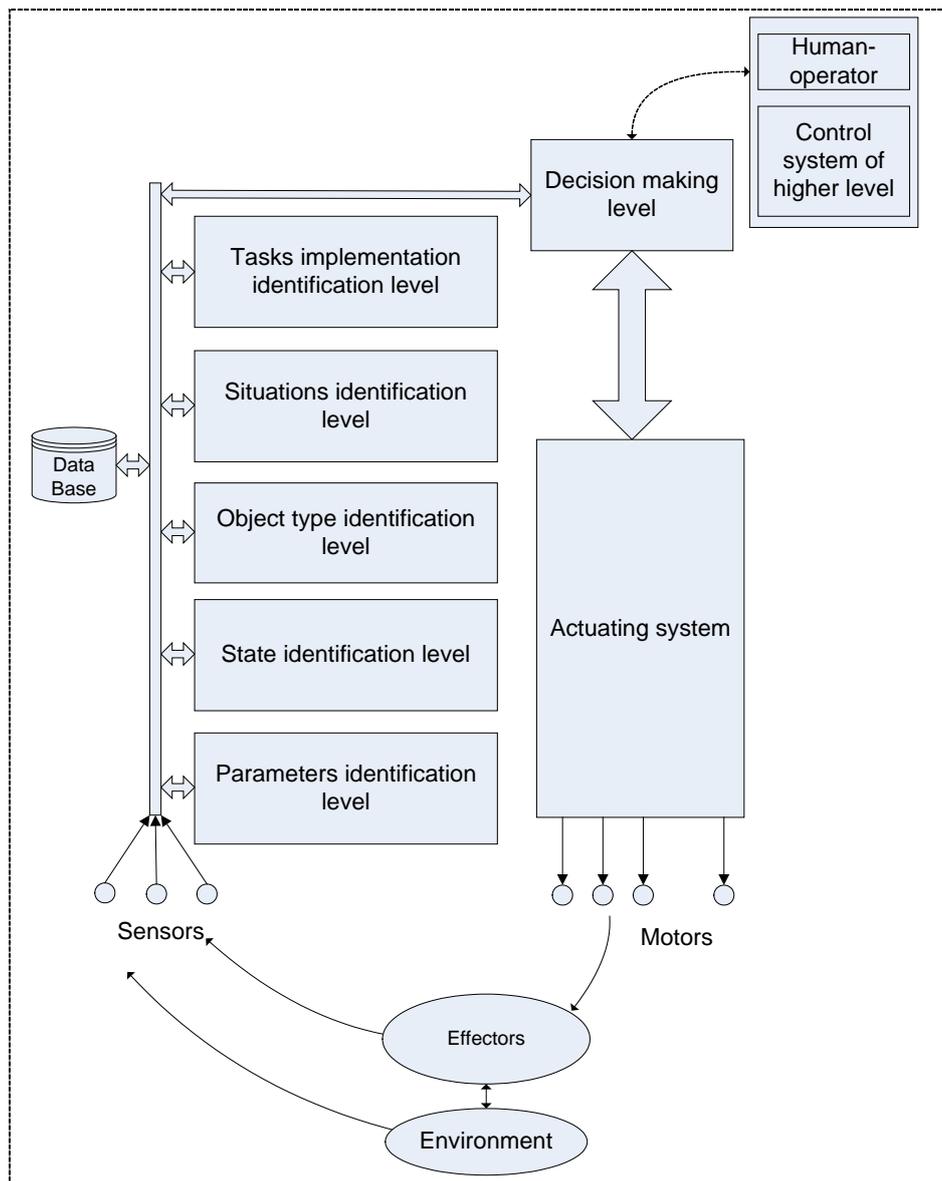
*Parameter* – predicate which describes as a rule quantitatively property of a single component of a UV or of environment. Input of this level usually comes directly from sensors. If you don't wish to use the Word template provided, please set the margins of your Word document as follows.

*State* – predicate which describes quantitatively or relatively property of a whole UV or of environment. Input for this predicate comes from parameters or from sensors directly.

*Object type* – generalized identification of an object present in environment, which is defined by its typical data and by its potential interaction with UV.

*Situation* – is a generalized notion which describes complex of interaction between robot and environment.

*Tasks* – a set of situations in serial-parallel order. These situations must be achieved (implemented) by robot in order for a task to be fulfilled.



**Figure 2.** Hierarchical Data Fusion Architecture.

Data from various sensors of UV and of environment flows to the common bus. An important advantage of this scheme is that it can use data from both on-board and stand-alone sensors.

Usage of common data bus is also an important advantage as it allows to exchange data and information among various levels freely. As it was mentioned before data may flow to the next level

and also to higher levels directly. This also allows in case human-operator needs it to receive information on any predicate directly.

All the data is fused in bottom-up direction. However it's obvious that some more simple or primitive systems may omit some of the higher levels. However highly autonomous systems will involve all the levels. When needed extra data will be acquired from data base. Final output comes to the decision making level which sends instructions to actuators and sub-systems. It also may be built on same approach.

The proposed Hierarchical data fusion architecture has following advantages compared to other architectures:

- it reflects hierarchy of modern UAVs' structure;
- it has clear demonstrable and visual properties, has a high degree of intuition for human-operator;
- this structure allows wide usage of various data flows and feedbacks including transient flows;
- this structure can be effectively implemented with modern control methods [1];
- structure's modularity allows various ready-to-use solutions be transferred from one scheme to another.

According to author's opinion proposed structure can be widely used also for automated programming of complex data fusion systems and algorithms.

### **Acknowledgement**

This research was supported by Program No.29 "Advanced Topics of Robotic Systems" of the Presidium of the Russian Academy of Sciences (AAAA-A17-117121120021-9).

### **References**

1. *Intelligent Control Methods*, Editors: Makarov I.M., Lokhin V.M.– Moscow: Physmatlit, 2001.
2. Ermolov I.L., Robots' Autonomy, its Measures and How To Increase it", *Mechanronics, Automation, Control*, 2008. – №8.
3. Fomichev A.A., Uspensky V.B., Stchastlivets K.Y., et al, Data fusion in integrated navigation system with laser gyroscopes based on generalized Kalman filter, *Proc. Of International Conference on Integrated Navigation Systems*, St. Petersburg, May 26-28m 2003.
4. Gorodetsky V. I., Karsaev O.V., Samoylov V.V., Data Fusion in Complex Situations Analysis and Understanding, *Proc.of "Perspective Systems and Control Tasks"*, Dombai, Russia, 2008.
5. Ermolov I.L., Factors Affecting UGVs Spacious Autonomy Level, *Vestnik YuFU*, №1, 2016.
6. Elmenreich Wilfried, *Sensor Fusion in Time-Triggered Systems*, *Technischen Universitat Wien*, Wien, im Oktober 2002.
7. Zenkevich S.L., Minin A.A., Mapping by UV with a Lidar basing recurrent Filtration Method. *Mechanronics, Automation, Control*,– 2007. - №8.
8. Ermolov I.L., Hierarchical Data Fusion Architecture for Unmanned Vehicles, in *Smart Electromechanical Systems: The Central Nervous System*, Springer, 2017.

# Pre-filter to robustify the exact linearization based tracking controller of a SCARA type robot<sup>1</sup>

Na Wang, Bálint Kiss

Budapest University of Technology and Economics Budapest, Hungary

Email: {wangna,bkiss}@iit.bme.hu

**Abstract.** Robustness against parameter uncertainties is an important requirement for robot controllers to achieve high precision and fast tracking. This paper proposes a procedure to cover the uncertainties remaining after the exact linearization (also known as the computed torque method in robotics), and to design an additional linear compensator  $K(s)$  to ensure robust performance and stability. The design of  $K(s)$  involves standard  $\mathcal{H}_\infty$ . The procedure is presented in details for a model of a four degree-of-freedom SCARA manipulator. The load mass and friction coefficients are considered as uncertain parameters. The simulation shows that the proposed method can increase robustness against parameter uncertainty.

## 1. Introduction

The number of industrial robots is continuously increasing. Their precision and speed determine the quality and efficiency of many manufacturing processes. This is also true for SCARA type (Selective Compliance Assembly Robot Arm) devices which are widely used in the electronics industry for assembly and pick-and-place tasks.

Robot arms are nonlinear, MIMO dynamical systems. Tracking precision and speed are conflicting requirements since controllers need to deal with model uncertainties whose effects increase with speed where coupling effects are more important. Manipulator segment and load inertia, friction coefficients, torque constants of the actuators are known with limited accuracy. Various methods have been already explored for robot controller design and implementation including sliding mode control [1],[2], neural networks [3],[4], fuzzy logic [5],[6], and neuro-fuzzy [7],[8], to name a few.

The so-called computed torque method is also known in the literature. The possible drawback of the method, also referred to as exact linearization, is its sensibility to parameter uncertainty. This paper suggests a novel method to cope with parameter uncertainties while applying the computed torque method. The method is based on the design of a pre-filter for a set of uncertain dynamics. The set is determined by taking the combination of the nonlinear robot dynamics for different parameter values in their uncertainty range with the linearizing feedback which works with the nominal parameter values. By linearizing this set of closed-loop dynamics one gets a set of linear systems which is then covered by an output multiplicative uncertainty structure for which a pre-filter is designed using  $\mathcal{H}_\infty$  techniques.

The design steps to obtain the pre-filter are presented in Section 2. The suggested method is applied to a SCARA type robot in Section 3.

---

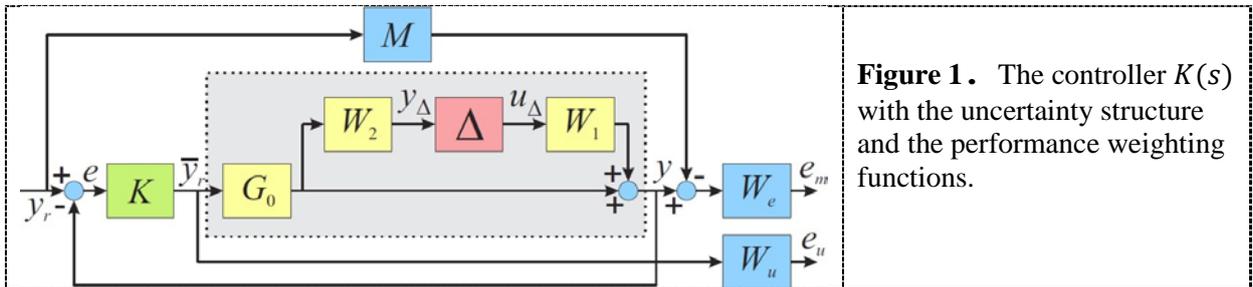
<sup>1</sup> The research reported in this paper was supported by the BME-Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC).

## 2. The robustifying pre-filter and its design

Let us assume that there are  $N$  uncertain parameters  $(p_1, p_2, \dots, p_N)$  such that each parameter may take its value from a bounded set, i.e  $p_i \in Q_i$  and  $Q = Q_1 \times Q_2 \dots \times Q_N$ . Consider a nonlinear dynamical system with an  $n$ -dimensional state vector  $x$  such that the state equation reads  $\dot{x} = f(x, u, p)$  and the output equation is  $y = h(x, p)$  where the dimension  $m$  of the input vector  $u$  equals the dimension of the output vector  $y$ . Suppose that an eventually dynamic linearizing feedback can be calculated for all possible parameter values in  $Q$ . Since the real values of the parameters are unknown, the feedback is applied for some nominal values  $p^0$ . The feedback law is given as  $\dot{\zeta} = \varphi(\zeta, x, v, p^0)$  and  $u = \psi(\zeta, x, v, p^0)$ , where  $\zeta$  is the  $k$ -dimensional state vector of the feedback and  $v$  is the  $m$ -dimensional new input.

The resulting closed-loop system with the nominal plant ( $p = p^0$ ) is  $m$  decoupled chains of integrators  $y_i^{(\mu_i)} = v_i$  with  $i = 1, \dots, m$  and  $n + k = \sum_{i=1}^m \mu_i$ . Recall also that it follows from feedback linearization that the time derivatives of the linearizing outputs (outputs of the integrators in the chains) can be expressed as functions of  $x$  and  $\zeta$ :  $y_i^{(j)} = h_{i,j}(x, \zeta, p^0)$  for  $j = 1, \dots, \mu_i - 1$ . The linearized dynamics (integrator chain) can be used to guarantee the exponential decay of tracking error for any sufficiently smooth reference trajectory, denoted as  $\bar{y}_r$ . The tracking feedback expression can be readily obtained by setting a linear differential equation for each component of the tracking error  $e = \bar{y}_r - y$  as  $0 = e_i^{(\mu_i)} + \lambda_{i,1}e_i^{(\mu_i-1)} + \lambda_{i,2}e_i^{(\mu_i-2)} + \dots + \lambda_{i,\mu_i}e_i$  so that the associated characteristic polynomial is Hurwitz. It follows from the above that the closed-loop transfer between  $\bar{y}_r$  and  $y$ , assuming  $p = p^0$ , reads  $Y(s) = \text{diag}_{i=1, \dots, m} \left\{ \frac{\lambda_{i,\mu_i}}{s^{\mu_i} + \lambda_{i,1}s^{\mu_i-1} + \dots + \lambda_{i,\mu_i}} \right\} \bar{Y}_r(s) = G_0(s) \bar{Y}_r(s)$ .

However, for non-nominal values of the uncertainty parameters, the closed-loop dynamics is different of  $G_0(s)$ . Therefore, a finite grid is chosen to cover the uncertainty range  $Q$ . For each vertex  $p^i$  of the grid, a linear transfer matrix  $G^i(s)$  is obtained between the signals  $\bar{y}_r$  and  $y$  by linearization. This set is covered by an output multiplicative structure with weighting matrices  $W_1(s)$  and  $W_2(s)$  so that  $G^i(s) = (1 + W_1(s)\Delta^i(s)W_2(s))G_0(s)$  holds true for some  $\|\Delta^i(s)\|_\infty \leq 1$ . The weighting functions are numerically determined using Matlab. The associated mixed sensitivity  $\mathcal{H}_\infty$  design problem is also solved with the help of Matlab and an augmented plant is used to design the pre-filter  $K(s)$  with weighting transfer functions  $M(s)$ ,  $W_e(s)$  and  $W_u(s)$ , denoting the desired closed loop model, the error weight and the input weight transfer functions respectively.



**Figure 1.** The controller  $K(s)$  with the uncertainty structure and the performance weighting functions.

## 3. SCARA Robot Control Architecture, Synthesis and Implementation

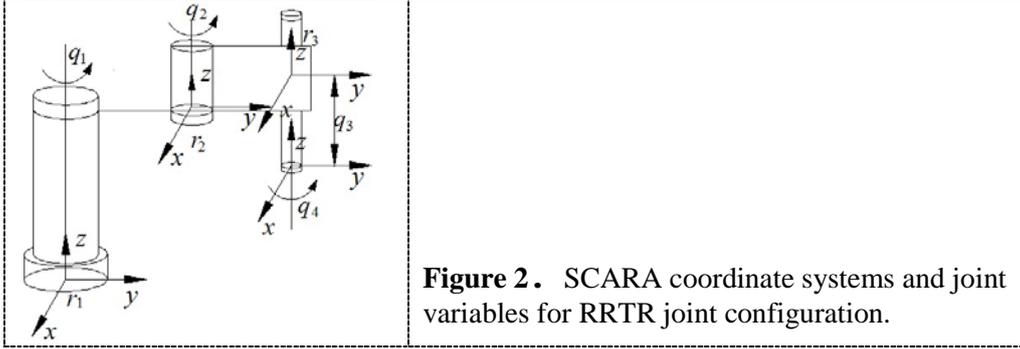
In order to demonstrate the feasibility of the proposed method, the design procedure described in the previous section is applied to a SCARA type robot (Bosch Turboscara SR 60) [9] as shown in Figure 3.

The SCARA robot has three rotary joints, whose axes are parallel with each other, and a prismatic joint. Using the Euler-Lagrange formalism, the dynamics of robot manipulators with rigid links can be written as  $H(q)\ddot{q} + C(q, \dot{q})\dot{q} + B(\dot{q}) + G(q) = \tau$  where  $q$  is the vector of joint variables,  $H(q)$  is the

positive definite inertia matrix,  $C(q, \dot{q})$  is the matrix of Coriolis and centripetal forces,  $B(\dot{q})$  is the vector of joint friction forces,  $G(q)$  is the vector of gravitational forces and  $\tau$  is the vector of actuator forces. In our case, the dynamic equation reads

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} & 0 & D_{14} \\ D_{21} & D_{22} & 0 & D_{24} \\ 0 & 0 & D_{33} & 0 \\ D_{41} & D_{42} & 0 & D_{44} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} + \begin{bmatrix} D_{112}\dot{q}_1\dot{q}_2 + D_{122}\dot{q}_2^2 \\ D_{211}\dot{q}_1^2 \\ -(m_3 + m_4)g \\ 0 \end{bmatrix} + \begin{bmatrix} F_{v1}\dot{q}_1 \\ F_{v2}\dot{q}_2 \\ F_{v3}\dot{q}_3 \\ F_{v4}\dot{q}_4 \end{bmatrix}$$

where the expressions of the inertia parameters are readily available in the literature.



**Figure 2.** SCARA coordinate systems and joint variables for RRTR joint configuration.

In most applications, the load mass can only be estimated with limited accuracy. In our case,  $m_i$  denote the segment masses. Values of  $m_1$ ,  $m_2$  and  $m_3$  are supposed to be known with sufficient accuracy: 15kg, 12kg and 3kg, respectively. The payload mass  $m_4$  remains variable. The lengths of the two first segments are also known  $l_1 = 0.5m$  and  $l_2 = 0.4m$ . The terms  $F_{vi}$  denote the friction coefficients of the joint axes. The uncertain parameters are enumerated in Table 1.

**Table 1.** Uncertain parameters of the SCARA

Variable	Nominal value	Minimum value	Maximum value
$m_4$	3 kg	2 kg	4 kg
$F_{v1}, F_{v2}, F_{v4}$	0.5 kgm/s	0.2 kgm/s	0.8 kgm/s
$F_{v3}$	0.05 kgm/s	0.02 kgm/s	0.08 kgm/s

The design procedure suggested in the previous section is followed. The five-dimensional parameter space is covered by a grid and the a linearized dynamics is determined for each vertex. The resulting set is represented by an uncertainty structure so that the `ucover` function from the Robust Control Toolbox of Matlab is used to compute the bounding diagonal weight transfer matrices  $W_1(s)$  and  $W_2(s)$ .

$$\begin{aligned} W_{1,11}(s) &= \frac{0.2684s^2 + 0.00963s + 2.541 \cdot 10^{-5}}{s^2 + 2.097s + 0.03992} & W_{1,22}(s) &= \frac{0.3504s^2 + 0.6235s + 0.001139}{s^2 + 3.459s + 3.105} \\ W_{1,33}(s) &= \frac{0.4894s^2 + 0.3171s + 0.007531}{s^2 + 3.108s + 3.82} & W_{1,44}(s) &= \frac{1.154s^2 + 0.3297s + 0.001853}{s^2 + 3.905s + 1.819} \\ W_{2,11}(s) &= \frac{0.7388s^2 + 0.3062s + 0.002246}{s^2 + 3.799s + 1.828} & W_{2,22}(s) &= \frac{0.7177s^2 + 8.693s + 0.01301}{s^2 + 15.45s + 32.65} \\ W_{2,33}(s) &= \frac{0.4894s^2 + 0.3171s + 0.007531}{s^2 + 3.108s + 3.82} & W_{2,44}(s) &= \frac{0.755s^2 + 0.3884s + 0.006696}{s^2 + 3.502s + 3.632} \end{aligned}$$

A reference model  $M(s)$  is used to define the desired closed-loop behaviour after the application of the robustifying pre-filter. It gives the transfer between the joint reference signal  $y_r$  and the real joint angle signal  $y$ . Recall that  $\bar{y}_r$  becomes the output of the robustifying pre-filter. The desired closed-loop behaviour is second order and reads

$$M(s) = \text{diag} \left\{ \frac{\omega_m^2}{s^2 + 2\xi_m \omega_m s + \omega_m^2} \right\}$$

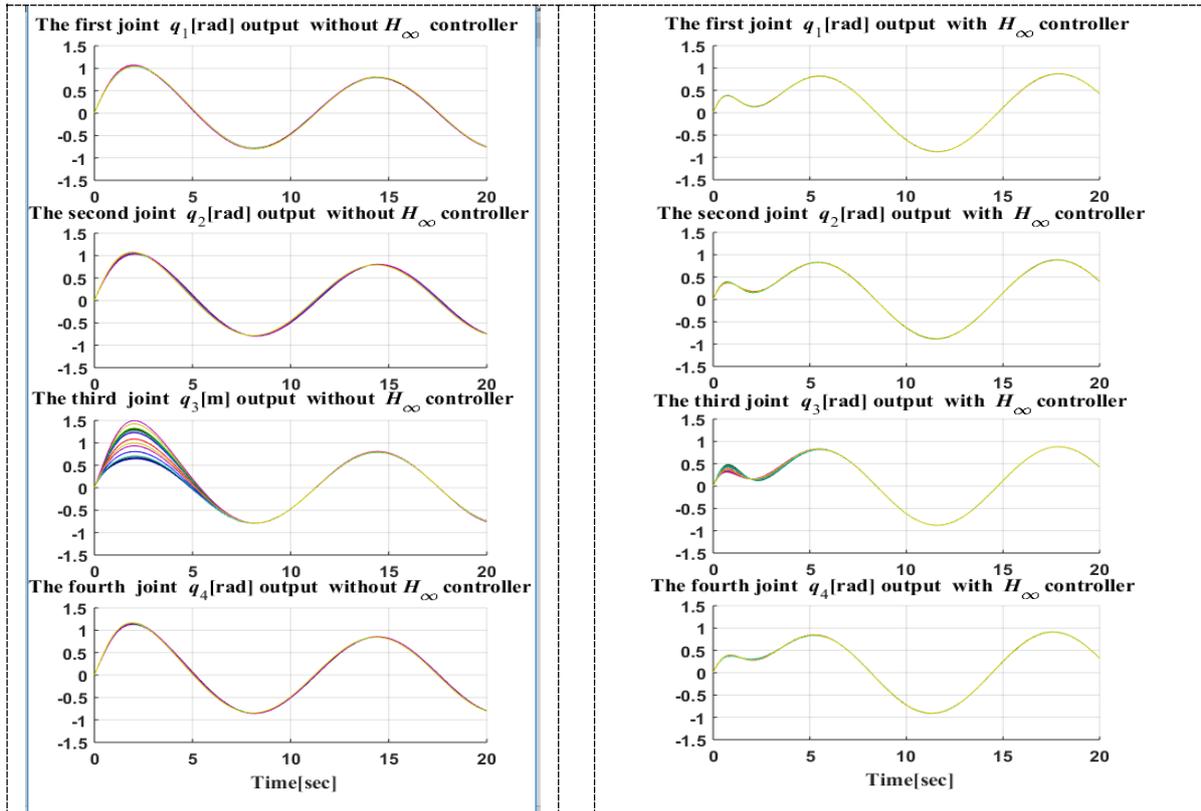
with  $\omega_m = 2\text{rad/s}$  and  $\xi = 0.8$ . The performance weighting transfer matrices (see Figure 1) are also chosen to be diagonal (4-by-4). The weighting function are set to

$$W_u(s) = \text{diag} \left\{ \frac{40(s + 10\omega_m)^2}{(s + 200\omega_m)^2}; \frac{40(s + 10\omega_m)^2}{(s + 200\omega_m)^2}; \frac{40(s + 10\omega_m)^2}{(s + 200\omega_m)^2}; \frac{1}{5} \frac{40(s + 10\omega_m)^2}{(s + 200\omega_m)^2} \right\}$$

$$W_e(s) = \text{diag} \left\{ \frac{1000\omega_m}{s + 2\omega_m}; \frac{1000\omega_m}{s + 2\omega_m}; \frac{9}{20} \frac{1000\omega_m}{s + 2\omega_m}; \frac{19}{40} \frac{1000\omega_m}{s + 2\omega_m} \right\}$$

The robot dynamics and the controller are simulated for different parameter values using Matlab and Simulink. The reference trajectory is set to  $y_r = \sin(t)$ . The state vector of the robot dynamics is  $x = [q, \dot{q}]$ . The initial conditions are set accordingly to  $x(0) = [0, 0, 0, 0, 1, 1, 1, 0]^T$ .

Figure 3 shows the tracking performance without the use of the pre-filter whereas Figure 4 shows the joint trajectories with the pre-filter. For each joint, a family of curves are presented, each for a different vertex of the grid spanned over the parameter space. It can be observed that the use of the pre-filter reduces the effect of uncertainty.



**Figure 3.** Joint trajectories without  $\mathcal{H}_\infty$  pre-filter

**Figure 4.** Joint trajectories with the application of  $\mathcal{H}_\infty$  pre-filter

#### 4. References

- [1] F. Adelhedi, A. Jribi, Y. Bouteraa, and N. Derbel, "Adaptive Sliding Mode Control Design of a SCARA Robot Manipulator System Under Parametric Variations," *J. Eng. Sci. Technol. Rev.*, vol. 8, no. 5, pp. 117–123, 2015.
- [2] C. J. Fallaha, M. Saad, H. Y. Kanaan, and K. Al-Haddad, "Sliding-mode robot control with

- exponential reaching law,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 2, pp. 600–610, 2011.
- [3] N. Goléa, G. Debbache, and A. Goléa, “Neural network-based adaptive sliding mode control for uncertain non-linear MIMO systems,” *Int. J. Model. Identif. Control*, vol. 16, no. 4, pp. 334–344, 2012.
  - [4] R. Köker, “A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization,” *Inf. Sci. (Ny)*, vol. 222, pp. 528–543, 2013.
  - [5] K. Lochan and B. K. Roy, “Control of two-link 2-dof robot manipulator using fuzzy logic techniques: A review,” In *Proc. of 4th Int. Conf. on Soft Comp. for Problem Solving*, 2015, pp. 499–511.
  - [6] M.-C. Popescu, I. Borcosi, O. Olaru, L. Popescu, and F. Grofu, “The simulation hybrid fuzzy control of SCARA robot,” *Wseas Trans. Syst. Control*, vol. 3, no. 2, pp. 105–114, 2008.
  - [7] Y. Sahin, M. Tinkir, and A. Ankarali, “Neuro-Fuzzy trajectory control of a scara robot,” in *Computer and Automation Engineering (ICCAE), 2nd Int. Conf. on*, 2010, vol. 2, pp. 298–302.
  - [8] J. Tavoosi, A. S. Jokandan, and M. A. Daneshwar, “A new method for position control of a 2-DOF robot arm using neuro-fuzzy controller,” *Indian J. Sci. Technol.*, vol. 5, no. 3, pp. 2253–2257, 2012.
  - [9] G. Incerti, “Trajectory tracking for SCARA robots with compliant transmissions: a technique to improve the positioning precision,” in *Proceedings of the 12th IFToMM World Congress*, 2007.

# Automated magnetic field reconstruction stand for mobile robot navigation algorithms debugging which use magnetometer data

S Goll<sup>1,2</sup> and A Borisov<sup>1,2</sup>

<sup>1</sup>Department of Information-Measuring and Biomedical Engineering, Ryazan State Radio Engineering University (RSREU), Gagarin Street, 59/1, Ryazan, RU

<sup>2</sup>LLC KB Avrora, Skomoroshinskaja Street 9, of.3, Ryazan, RU

**Abstract.** An important addition to the inertial navigation system are magnetometers. Areas with magnetic field anomalies serve to determine the reference points. But the magnetometers are influenced from both the electrical equipment of the robot itself and robot's parts configuration. Compensation of the robot's self-influence on the readings of the magnetometers is carried out by computer tools. In order to obtain the initial data, live experiments are required in a natural environment. For simplifying data acquisition about the behavior of magnetometric systems of a mobile robot a facility is used, which allow to compensate the Earth's magnetic field in a working space and to create an artificial magnetic field that varies according to a predetermined algorithm and simulates a magnetic field in the intended environment of application of the robot. The facility features: a working space, sufficient to place the mobile robot; a coil temperature drift correction; uniformity of the frequency response in operating frequency range; compensation of power supply interference and similar disturbances; sensitivity equalization of control channels; compensation of misalignment coordinate systems of the sensor and coil system. An interactive Simulink model designed and evaluated. The automated stand created as experimental facility, its parameters prove proposed model adequacy.

## 1. Introduction

SLAM based on magnetometer data is a rapidly growing field in the robotics [1] – [8]. The unperturbed natural Earth's magnetic field (EMF) is used to determine the orientation of the mobile robot, and local anomalies (disturbances) of the EMF are used as features for positioning and navigation algorithms [4], [6]. Magnetometer data based algorithms and methods for navigation and mapping are especially relevant for indoor usage and for usage in urbanized environment. Such areas have a large number of objects perturbing the EMF, which in the mobile robots operation areas is expected to be homogeneous and stable. Modern researches create systems for building maps of magnetic environment along with maps based on data from lidars and cameras. For magnetic environment map building single- or multi-axes magnetometers (electronic compass) of magnetometer systems, integrated with exterior receptive sensor system of the mobile robot are used.

Even a carefully calibrated magnetometer system, in combination with sensors of spatial position estimation (accelerometers) is subject to the influence of static, folding, retractable and other elements of the robot, as well as its flowing currents. Distortion of magnetometer estimations caused by such influence may be compensated by algorithms based on machine learning with redundant data, which is recorded by proprioceptive mobile robot sensor system [9]. The quality of compensation depends on the model of formation and propagation of disturbances. The perturbation model is created in a special environment called "magnetic silence", which significantly excludes external, i.e. not related to the

robot, influences on magnetometers. Such influences are caused by different objects like indoor engineering networks (power supply – first). Even the influence from reinforced concrete beams and metal coated floor may be significant. For model evaluation in [9] authors were obliged to build experimental site in the Utah desert.

## **2. Magnetometers calibration stand**

For magnetometers calibration it is a common practice to use facilities based on the Helmholtz coil system (or other similar coil systems) and functional generators. These facilities allow creating the artificial magnetic field (MF), comparable with EMF. These systems form a homogeneous MF which is constant or varies according to a given function, with or without simultaneous compensation of the EMF, as well as its perturbations of both anthropogenic nature and magnetic storms. The control channel is implemented both open [10] and closed [11].

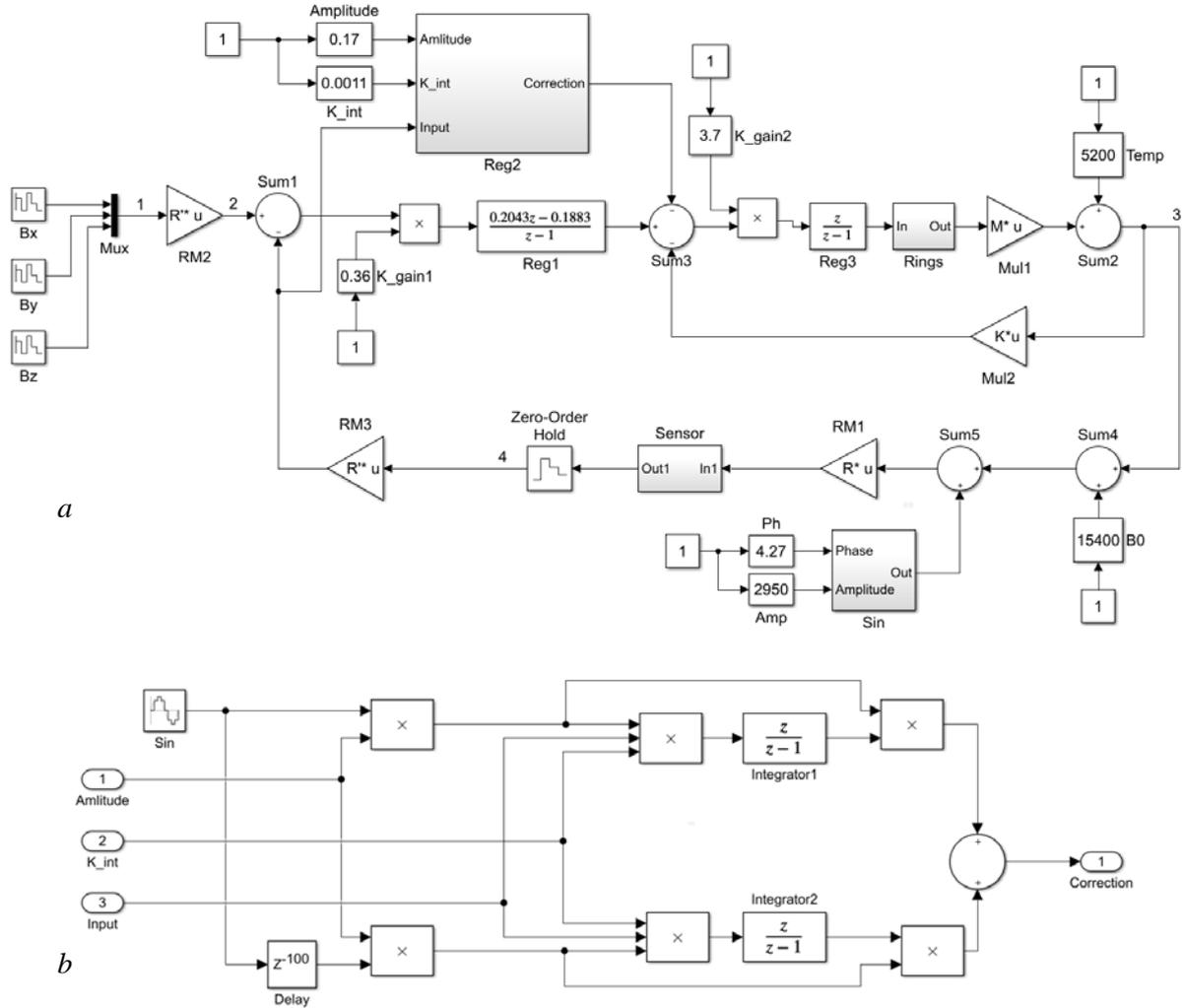
The authors used this approach while designing an automated stand for reconstructing the MF for solving the problem of determining perturbations caused by mobile robot elements. A mobile robot is placed in the test zone of a three-component contour system, where a series of experiments is performed using various operating modes of its executive subsystems. The reproducible MF is either constant or changing according to the function which were specified in a dedicated editor (e.g., simulating rotations of a mobile robot). The prerecorded set of magnetic induction vectors serves as a training set for designing and evaluating various models which are invariant to the self-influence of the robot. Another task of automated stand consists in representing variation of the MF vector along the real route to the robot, which is situated in the contour system, with further benchmark of the features recognition algorithms. The MF parameters may be previously recorded using precision three-component magnetic induction sensor.

Most of emergency mobile robots are fit in space about  $1 \text{ m}^3$ , but building a coil system with such test zone space caused a series of problems. By increasing test site space we decrease the current transmission coefficient. Compensation of the coefficient drop by increasing the number of coil turns causes an increase in inductance of the coils and a narrowing of the operating frequency range consequently. Increasing the control currents in its turn requires non-trivial solutions for their formation, with preserving low levels of interference. In addition, it results in the fact that it is no longer possible to neglect changes in currents caused by the temperature drift of the coil resistance, which further leads to deviations of the MF, comparable in magnitude with the EMF. The MF stabilization in the test site space is impossible without using of a closed loop system. Feedback can be realized either with a separate MF sensor or with the robot on-board sensor. The coaxial arrangement of the sensor and the Helmholtz coil circuit system also ensures correct operation of the closed loop control system. The mechanical arrangement of the coordinate systems cannot be sufficiently accurate because mobile robots usually are not equipped with tools for precise orientation estimation. In addition, in a sequence of experiments, a robot is not always installed in a similar way, but takes many different positions. The residual misalignment should be compensated by the program setting of the test site.

## **3. Automated stand model**

Let us dwell on details of solving the problem of stabilizing the MF in the test site volume, compensating for the residual misalignment of the coordinate systems of the sensor and the contour system, eliminating power supply interference, and the influence of temperature drift of the ring resistances.

The proposed solution is explained using the Simulink model, convenient from the point of view of interactive verification of the automated stand for the reconstruction of the MF is shown in figure 1.



**Figure 1.** a – Simulink-model of the automated stand for the reconstruction of the magnetic field, b – the structure of the adaptive regulator Reg2.

The Rings block defines the model of the contour system as a transmission function  $W(s) = \frac{1.4406}{0.0008s + 0.8}$  for each reproducing channel. This transmission function is obtained as a result of identification of a real Helmholtz coil having a diameter of 2 m with seven turns of copper wire with a cross-sectional area of 2 mm<sup>2</sup>. The Mul1 block determines the differences in the current transmission coefficients of the coils by multiplying the input signals by the matrix.

$M = \begin{bmatrix} 0.9 & 0 & 0 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.1 \end{bmatrix}$ . The Sensor module simulates the transmission function of a three-component

magnetic induction transducer  $H(s) = \frac{I}{0.000514^2 s^2 + 0.0019s + 1}$  for each of the channels  $H(s)$  as an estimate of the transmission function of the serial sensor HB0302.61A, the time step of the measurement data is 1 ms and is defined by the Zero-Order Hold block. It should be noted that the sampling interval of the entire Simulink model is 50 μs, therefore, the channel for reproducing the MF

is 20 times faster than the measuring channel. The signals of the magnetic induction vector  $B_x, B_y, B_z$  assigned for reproduction are combined by the Mux block into a vector signal, therefore all other signals of the Simulink model are three-component ones. The RM1 block specifies the misalignment of the contour coordinate systems and the magnetic induction sensor by multiplying the coil current

vector  $[I_x \ I_y \ I_z]^T$  by the rotation matrix  $R = \begin{bmatrix} 0.9998 & -0.0118 & 0.0146 \\ 0.01 & 0.9933 & 0.1152 \\ -0.0158 & -0.115 & 0.9932 \end{bmatrix}$ . Identification of this

matrix in practice is performed using the formula  $R = DS^T(SS^T)^{-1}$ , where  $D$  – the matrix of  $N$  test current vectors sequentially fed to the contour system,  $S$  – the corresponding matrix of the components of the magnetic induction vectors measured by the sensor  $[B_x \ B_y \ B_z]^T$ . For an accurate estimation of  $R$  it is necessary to reproduce currents close to the borders of the ranges of the test MF, without causing saturation of the coils with respect to the current. It is necessary to take into account the displacement of the dynamic ranges of the channels by the magnetic induction, which is connected with the presence of a constant EMF in the region of system installation.

The main feedback loop for the MF sensor closes on Sum1, the input signals of which are firstly multiplied by the matrix  $R^T$  (in Simulink notation  $R'$ ) by the blocks RM2 and RM3. These blocks are designed to compensate axis misalignment of the sensor and the contour system.

The difference signal from Sum1 is fed to the regulator Reg1 with a transmission function  $V(z) = \frac{0.2043z - 0.1883}{z - 1}$ , whose amplification factor is interactively tuned with the K\_gain1 block.

In addition to K\_gain1 block there are also Amplitude, K\_int, Ph, Amp, K\_gain2, Temp and B0 interactive tuning blocks. All of them are built using the standard Simulink block – Slider Gain.

The background MF level and its perturbations is interactively set by the B0 block and is added by Sum4 block to the field created by the coil system. The Sum5 block introduces a change in the MF into the coil system caused by power supply interference. It is a 50 Hz sine wave, set by the Sin block with interactively arranged amplitude (Amp block) and phase (Ph block).

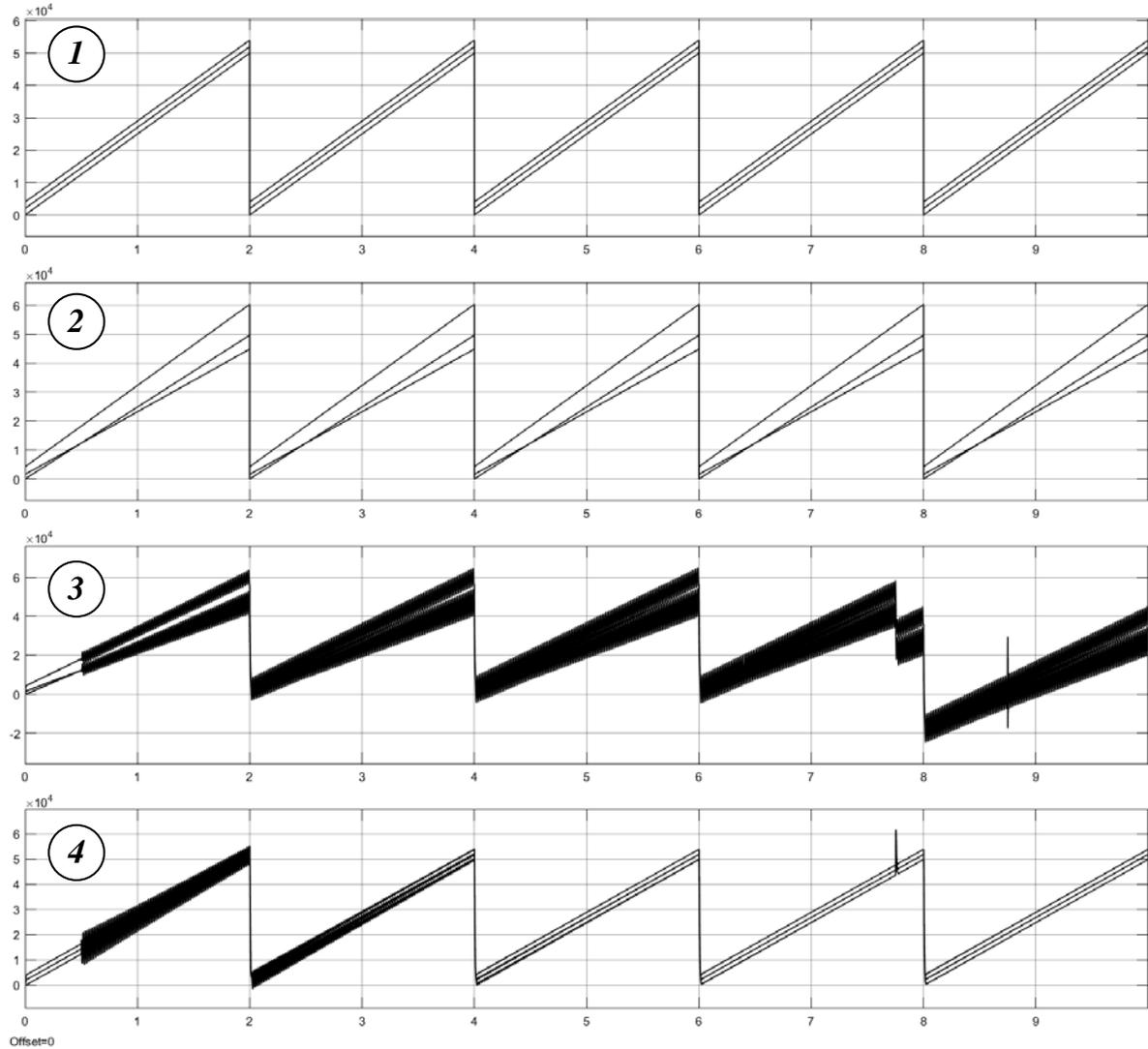
The main feedback loop compensates a constant and slowly changing external MF, but it does not satisfactorily compensate the power supply interference. That's why we introduced another feedback loop with the regulator Reg2, built as a modification of the adaptive rejection filter for suppressing this interference [12]. The regulator structure is represented in figure 2b.

The additive mixture of the useful signal and the power supply interference from the main feedback loop output is fed to the input of the adaptive regulator Reg2. The reference 50 Hz sine wave is generated by the Sin generator with 50  $\mu$ s sampling interval of the Simulink model. The reference signal with 100 sampling intervals delay, created by the Delay block is converted into a signal shifted in phase by 90°. The reference signal and its shifted copy are multiplied with the input signal of the regulator Reg2, the resulting products are fed to the digital integrators Integrator1 and Integrator2. The corresponding integration results are the weight coefficients of the adaptive regulator. The weighted reference signal and its shifted copy are summed, resulting in the regulator output, which is subtracted from the output signal of the main regulator Reg1. Adjustable weighting values allow changing the reference signal in amplitude and phase by any method necessary to suppress the power supply interference. The interactive amplitude adjustment of the reference sine wave and the integration factor, which is common for both integrators using the Amplitude and K\_int blocks, affects the adaptation speed.

The temperature stabilization loop includes system blocks associated with Helmholtz coil's currents generation. The output value – the coil current for each of the spatial axes – is converted with the scale factor (defined by Mul2 block) and is fed to the summation point Sum3. A signal compensating for the power supply interference also enters this point. The feedback transfer ratio depends on the parameters of the coils and the electronics of the regulated current source, so it must be selected during the initial system calibration.

The regulator in the direct transfer circuit (Reg3) eliminates the residual steady-state error of the control channels, and the block K\_gain2 allows interactively changing the parameters of the current generating unit. The temperature error is modeled by the interactive Temp block and is summed with the current signals of the loop system.

In figure 2 the results obtained by using the Simulink model at points 1, 2, 3 and 4 are presented. For an effective demonstration of the proposed solution, let us designate three sawtooth signals offset from each other as changes in the component of the magnetic induction vector (point 1). The sawtooth signals at point 2 are no longer parallel to each other and show the degree of misalignment of the sensor and the contour system. At time  $t = 0.5$  s the amplitude and phase of the sine wave of the power supply interference are abruptly changed, and at the time  $t = 7.75$  s the level of the background MF is abruptly changed. Transient processes as a result of these disturbances may be estimated on the graph of the field reproduced by the coils (point 3) and superposition of this field and perturbations at the sensor output (point 4).



**Figure 2.** Signals at points 1, 2, 3 and 4 of the Simulink model. X-axes in seconds, Y-axes in nanotesla.

Adaptation to the new value of the external field performed in 4 s, and the elimination of power supply interference ended after 30 ms. Transient process as a result of temperature step change located

on the graph of the field reproduced by the coils (point 3) at 8.85 s. System stabilization took about 15 ms. At point 4 temperature disturbance is imperceptible due to sensor sampling rate. Taking into account that these are not stressful effects typical for practice, the values obtained can be considered sufficient for solving the tasks set. The restored parallelism of the sawtooth signals at point 4 indicates that the compensation of misalignment of the sensor and the contour system is performed. Resulting MF magnitude (point 4) equivalent the programmed one (point 1).

#### 4. Automated stand implementation

The automated stand for the MF reconstruction was made as an experimental facility, is presented in figure 3. The facility performance result is proved the adequacy of the proposed Simulink-model. A small number of independent variables of the model made it possible to abandon the analytical search for stability and the selection of parameters close to optimal values of regulators in favor of their interactive tuning. Effectiveness of proposed solution in a variety of mobile robots and magnetometers used as research objects was shown in practice.

The automated stand is used to perform experiments to evaluate the robot self-influence on the magnetometer system following the way described in [9], but without searching for a special “magnetic silence” terrain. To do this, first, a field that simulates the EMF is created in the empty coil system. For creating the initial MF, a separate sensor or robot on-board sensor or both can be used. The currents flowing through the coils are fixed as reference values. At the second stage, the robot is installed in the coil system and its magnetometer is used as a feedback sensor. Distortions of the MF caused by the self-influence of the robot, leads to other values of the control currents. The difference of the reference and observable currents serves as the basis for a training set for self-influence compensation algorithms.

The second use case is the determination of the frequency characteristics of the magnetometers. Simulink model point 1 switches on the automatic identification block. After transmitting a test sinusoidal signal with a linear frequency modulation to the coil system and simultaneously maintaining the specified amplitude of the coil currents, the unit will perceive all the frequency properties of the Sensor block.



**Figure 3.** Automated stand for reproducing magnetic field: on the left – a three-component coil system of Helmholtz coils, on the right – a functional block.

- [1] Solin A, Kok M , Wahlström N , Schön T B 2015 Modeling and interpolation of the ambient magnetic field by gaussian processes, *CoRR*, vol. abs/1509.04634.
- [2] Haverinen J and Kemppainen A 2009 Global indoor self-localization based on the ambient magnetic field, *Robotics and Autonomous Systems*, vol. 57, no. 10, pp. 1028-1035.
- [3] Vallivaara I, Haverinen J, Kemppainen A and Rönning J 2011 Magnetic field-based SLAM method for solving the localization problem in mobile robot floor-cleaning task, *15th IEEE International Conference on Advanced Robotics: New Boundaries for Robotics, ICAR 2011*, Tallinn, Estonia, 20-23 June 2011., pp. 198-203.
- [4] Angermann M, Frassl M, Doniec M, Julian B J and Robertson P 2012 Characterization of the indoor magnetic field for applications in localization and mapping, *IEEE International Conference on Indoor Positioning and Indoor Navigation, IPIN 2012*, Sydney, Australia, 13-15 November 2012, pp. 1-9.
- [5] Grand E L and Thrun S 2012 3-axis magnetic field mapping and fusion for indoor localization, *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2012*, Hamburg, Germany, 13-15 September 2012, pp. 358-364.
- [6] Frassl M, Angermann M, Lichtenstern M, Robertson P, Julian B J and Doniec M 2013 Magnetic maps of indoor environments for precise localization of legged and non-legged locomotion, *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan, 3-7 November 2013, pp. 913-920.
- [7] Lee N, Ahn S and Han D 2018 AMID: accurate magnetic indoor localization using deep learning, *Sensors*, vol. 18, no. 5, p. 1598.
- [8] Kemppainen A, Vallivaara I and Rönning J 2015 Magnetic field SLAM exploration: Frequency domain gaussian processes and informative route planning, *2015 IEEE European Conference on Mobile Robots, ECMR 2015*, Lincoln, United Kingdom, 2-4 September 2015, pp. 1-7.
- [9] Christensen L, Krell M M and Kirchner F 2017 Learning magnetic field distortion compensation for robotic systems, *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017*, Vancouver, BC, Canada, 24-28 September 2017, pp. 3516-3521.
- [10] Roman S, Jacek S, Adam B, Rafal K and Marcin S 2009 Testing of the three axis magnetometers for measurements of the earth magnetic field, *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 3, no. 4, pp. 96-98.
- [11] Alvarez A R, Mejía E F, Ramírez H C and Jaramillo C P 2017 A simple geomagnetic field compensation system for uniform magnetic field applications, *Revista Facultad de Ingeniería*, vol. 0, no. 83, pp. 65-71.
- [12] Widrow B and Stearns S 1985 Adaptive signal processing. Englewood Cliffs, N.J: Prentice-Hall.

# An Active Beacon-Based Tracking System To Be Used For Mobile Robot Convoying

S Goll<sup>1,2</sup> and E Zakharova<sup>1,2</sup>

<sup>1</sup>Department of Information-Measuring and Biomedical Engineering, Ryazan State Radio Engineering University (RSREU), Gagarin Street, 59/1, Ryazan, RU

<sup>2</sup>LLC KB Avrora, Skomoroshinskaja Street 9, of.3, Ryazan, RU

**Abstract.** The paper is focused on the mobile robot convoying along the path travelled by some leader carrying the active ultrasonic beacon. The robot is equipped with the three-dimensional receiver array in order to receive both the ultrasonic wave and the RF wave marking the beginning of the measurement cycle. To increase measurement reliability each receiver contains two independent measurement channels with the automatic gain control. The distance measurements are preprocessed to remove identify the artifacts and either remove them or substitute with the interpolated value. To estimate the position of the beacon in the robot's local coordinate system several methods are used including least squares method with the subsequent exponential smoothing, linear Kalman Filter, Rauch-Tung-Striebel smoother, Extended Kalman Filter, Unscented Kalman Filter and the Particle Filter. The experiments were made in order to estimate the estimation method preferable for the leader's path following.

## 1. Introduction

The convoying scenario is one of the most important tasks for the modern robotics. A mobile robot autonomously following a leader can be widely used in such areas as agriculture, transportation and military. Many well-known international trials, such as ELROB, include this scenario either as an independent one or as a part of a more complex scenario.

The scenario is focused on the mobile robot autonomously following some leader who can be either a human operator or another vehicle, autonomous or remote controlled. To detect the leader different systems can be used, including GNSS systems [1, 2] video cameras [3, 4], infrared cameras [5], lidars [6], radars and ultrasonic rangefinders [7, 8]. The nature of these systems, however, can impose various restrictions on the conditions of their use. GNSS systems perform poorly in the urban areas and inside of the buildings. Lidars and radars can be used to measure distance between the vehicle and the surrounding objects, but it is a non-trivial task to detect the leader based on range data. Cameras, both video and infrared, can help with leader detection, but depend greatly on the environmental conditions. Moreover, if the obstacle appears between the leader and the robot the accuracy of their relative positions estimate can be severely decreased.

The mathematical methods used for localization, obstacle detection and occupancy grid mapping are not reliable enough to build a robust convoying system. Such methods usually require the redundant sensor data of different nature. The leader detection suffers from the same restraints and, to make things even more difficult, the rough weather conditions, dense vegetation, smoke and other factors should also be considered.

## 2. The approach

In this paper we propose an ultrasonic-based leader detection system which includes an active beacon carried by the leader and a set of  $N$  ultrasonic receivers mounted on the convoyed robot (figure 1).

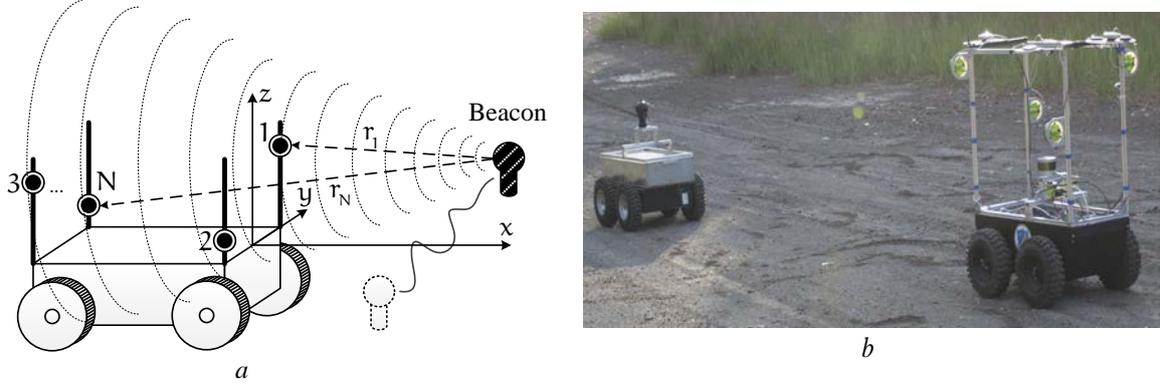


Figure 1. a – active beacon and the receiver array mounted on the mobile robot, b – convoying scenario with the two robots.

The active beacon transmits the ultrasonic waves with the constant time intervals between the waves. At the same time as the ultrasonic wave is sent the radio-frequency wave marking the beginning of the measurement cycle is also transmitted. For the each of the  $N$  receivers the time interval between the moment of radio-frequency wave and ultrasonic wave arrivals is measured according to the Time-of-Flight principle. These intervals are proportional to the distances between the beacon and the respective receivers and can be calculated as  $r_n = c\tau_n$ ,  $n = 1, \dots, N$ , where  $c$  is the speed of the ultrasound in the air,  $\tau_n$  is the time interval measured for the  $n^{\text{th}}$  receiver.

To estimate the beacon's coordinates in the robot's local coordinate system the system of  $N$  nonlinear equations is solved.

$$r_n^2 = (x^m - x_n)^2 + (y^m - y_n)^2 + (z^m - z_n)^2, \quad n = 1, \dots, N, \quad (1)$$

where  $\vec{u} = [x^m \quad y^m \quad z^m]^T$  is a vector containing coordinates of the beacon and the

$[x_n \quad y_n \quad z_n]^T$  are the coordinates of the  $n^{\text{th}}$  receiver in the convoyed robot's local coordinate system.

During the system operation an obstacle can cause the line-of-sight loss between one or several receivers and the beacon. Moreover, the ultrasonic wave can be reflected by the surrounding objects causing the multipath propagation problem. For these reasons acquired measurements can contain artifact distances  $r_n$  (figure 2a) which are identified using the threshold constant  $\rho$ . The distance measurement  $r_{n,k}$  is considered an artifact if the following condition is met

$$|r_{n,k} - r_{n,k-1}| > \rho, \quad (2)$$

where  $\rho$  is an adjustable threshold constant,  $k$  – consequent measurement number.

If the artifact is detected, the linear least square extrapolation is used to calculate the substitute estimate based on the last  $W$  estimates  $\{r_{n,k-1}, \dots, r_{n,k-W}\}$ .

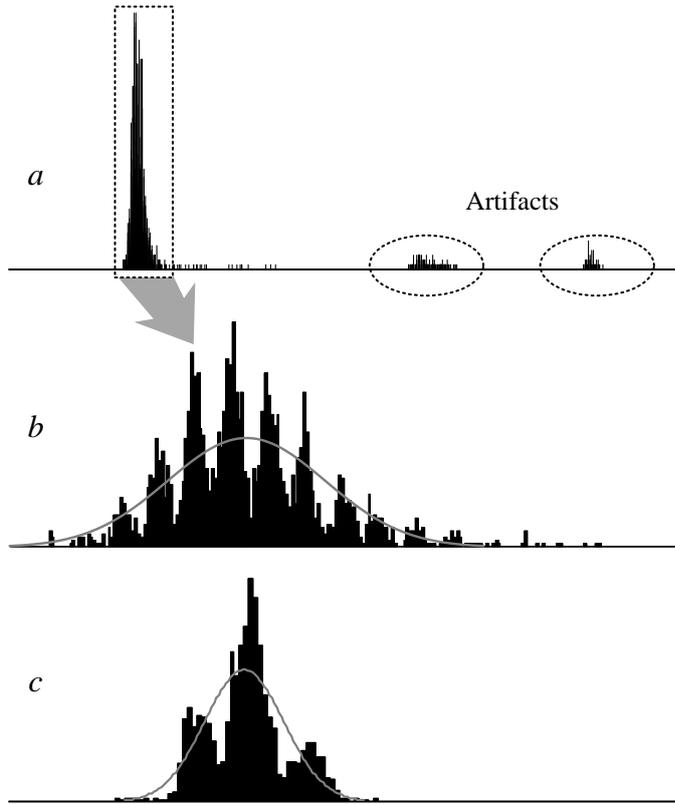


Figure 2. Histogram of centered values for the single receiver's measurements:  
a – measurement series containing artifacts,  
b – 11 meter distance between the beacon and the receiver,  
c – 3 meter distance between the beacon and the receiver.

If the series of  $M$  artifacts is detected in the receiver's measurements or the substitute estimate cannot be calculated due to the unreliable measurements present in the last  $W$  estimates, the corresponding equations are excluded from the system (3). When the reliable measurements (i.e., the condition (2) is not met for the two consecutive measurements) arrive the receiver is included back in the system (3) and the artifact removal procedure becomes applicable again. The system must contain at least three reliable measurements from different receivers to calculate the beacon's position estimate  $\vec{u}$ . If there are not enough reliable measurements available the robot stops until the beacon's position can be calculated again.

To make more robust measurements each receiver contains two independent measurement channels with partly overlapping beam patterns and automatic gain control. If both channels succeeded at the distance measurement at some moment of time  $k$ , then the average value is used as a resulting measurement. If one of the channels failed to provide a measurement then the over channel's measurement is used as a resulting measurement.

### 3. Position estimation

Many of the effective estimation methods are based on the linear models and use normally distributed values. The figures 2b and 2c show the histograms of centered values for the measured distances  $r_n$  between the beacon and the  $n^{\text{th}}$  receiver with the beacon positioned along the axis of the receiver's beam pattern at the distance of 11 and 3 meters accordingly. Clearly, the histograms are multimodal with the constant distances  $c \cdot \Delta\tau$  between the modes, where  $\Delta\tau$  is the period of ultrasound.

Since the envelope's shape can be approximated with the Gaussian, we can assume that the distances  $r_n$  are distributed normally with the variance depending on the beacon's position. However, if we assume that the  $\vec{u}$  values are normally distributed as well, then the  $r_n^2$ , as it appears in the (1),

complies with the single-sided exponential distribution (figure 3). This means that distances  $r_n$  are not distributed normally.

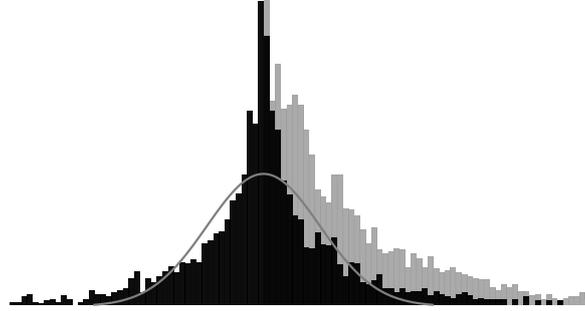


Figure 3. Distribution histograms for the  $r_n^2$  (grey columns) and the  $g_l$  (black columns).

We can use linear combination of the equations to transform the nonlinear equations of (1) into the linear ones. For the each pair of receivers the difference of the squared distances to the beacon is calculated as follows

$r_i^2 - r_j^2 = (x^m - x_i)^2 + (y^m - y_i)^2 + (z^m - z_i)^2 - (x^m - x_j)^2 - (y^m - y_j)^2 - (z^m - z_j)^2$ , where  $i, j = 1, \dots, N$ ,  $i \neq j$ ,  $C_N^2$  – binomial factor,  $L = C_N^2$  – number of receiver combinations.

As a result, the system (1) can be written in the form of the linear system

$$B\vec{u} = \vec{g}, \quad (3)$$

where  $B$  – system matrix with the  $L$  rows containing  $[2(x_j - x_i) \quad 2(y_j - y_i) \quad 2(z_j - z_i)]$ ,  $\vec{g}$  – column vector of  $L$  constant terms  $(r_i^2 - r_j^2 + x_j^2 - x_i^2 + y_j^2 - y_i^2 + z_j^2 - z_i^2)$ ,  $i, j = 1, \dots, N$ ,  $i \neq j$ .

Generally, the system (3) is inconsistent due to the measurement noise. Hence, to estimate the position of the beacon the least squares method can be used  $\vec{u} = (B^T B)^{-1} B^T \vec{g}$ .

If  $\vec{u}$  is distributed normally, then  $g_l$ ,  $l = 1, \dots, L$  in (3) complies with the two-sided exponential distribution as shown in the figure 3 with the black-colored columns. This distribution can be approximated by the Gaussian more accurately than the single-sided one, increasing the accuracy of the estimate obtained with most of the effective methods.

To compute the estimate  $\vec{u}$  of the beacon's position different methods can be used, therefore several estimates were computed in the course of research. Since all the methods are well-known and widely used it seem unnecessary to describe them. For that reason only the implementation details are provided for the each method.

During the research the receiver array of  $N = 4$  receivers was used, therefore, the system (3) contains  $L = C_4^2 = 6$  equations.

### 3.1. Least squares method with the exponential smoothing

$$\vec{u}_k^s = \alpha \vec{u}_k + (1 - \alpha) \vec{u}_{k-1}^s, \quad \alpha \in (0; 1), \quad (4)$$

where  $\alpha$  is the adjustable smoothing factor.

### 3.2. Kalman Filter with the non-stationary measurement noise matrix

The overall filter design is done according to [9]. Still, several implementation details should be clarified. According to (3) the stochastic system model can be described as follows

$$\begin{cases} \vec{x}_{k+1} = A\vec{x}_k + \vec{\xi}_k, \\ \vec{g}_k = C\vec{x}_k + \vec{\eta}_k, \end{cases} \quad (5)$$

$\vec{x}_k = [x_k^m \quad y_k^m \quad z_k^m \quad v_k^x \quad v_k^y \quad v_k^z]^T$  – state vector containing beacon’s coordinates and velocity

components;  $A = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$  – stationary transition matrix of the dynamic model,

where  $\Delta t$  – time interval between the consecutive measurements;  $\vec{g}_k = [g_{1,k} \quad \dots \quad g_{L,k}]^T$  – measurement vector;

$C$  – stationary measurement model with  $L$  rows  $[2(x_j - x_i) \quad 2(y_j - y_i) \quad 2(z_j - z_i) \quad 0 \quad 0 \quad 0]$ ,  $i, j = 1, \dots, N$ ,  $i \neq j$ ;

$\vec{\xi}_k = [0 \quad 0 \quad 0 \quad a_k^x \quad a_k^y \quad a_k^z]$  – process noise,  $\vec{\xi}_k \sim \mathbb{N}(\vec{\xi}_k | \vec{0}, Q)$ ,  $Q = \text{diag}(0, 0, 0, \sigma_a^2, \sigma_a^2, \sigma_a^2)$  –

stationary process noise covariance matrix;  $\vec{\eta}_k$  – observation noise vector,  $\vec{\eta}_k \sim \mathbb{N}(\vec{\eta}_k | \vec{0}, R_k)$ ,  $R_k$  – non-stationary measurement noise covariance matrix.

To estimate the measurement noise covariance [10] we assume that

$R_k = E[\vec{s}_k \vec{s}_k^T] + CP_k C^T$ , where  $\vec{s}_k$  – measurement residual (also called innovation), when

$$R_k = \begin{cases} R_0, & k < D \\ \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + CP_k C^T, & k \geq D \end{cases}, \quad D - \text{the estimation window size.}$$

The initial state vector is assumed to be  $\vec{x}_0 = [x_0^m \quad y_0^m \quad z_0^m \quad 0 \quad 0 \quad 0]^T$ , where  $[x_0^m \quad y_0^m \quad z_0^m]^T = (B^T B)^{-1} B^T \vec{g}_0$ . The posteriori error covariance matrix and the measurement noise covariance matrix are written as  $P_0 = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2)$  and  $R_0 = \text{diag}(\underbrace{\sigma_g^2, \dots, \sigma_g^2}_L)$  respectively.

### 3.3. Rauch-Tung-Striebel smoother

The sequence of beacon’s state estimates  $\{\vec{x}_k, \vec{x}_{k-1}, \dots, \vec{x}_{k-T}\}$  can be used to set the desired path for the mobile robot. If the corresponding covariance matrices  $\{P_k, P_{k-1}, \dots, P_{k-T}\}$  are also known, then this path-to-be can be smoothed using the Rauch-Tung-Striebel smoother. The estimate and covariance sequences are rewritten as  $\{\vec{x}_T, \vec{x}_{T-1}, \dots, \vec{x}_0\}$  and  $\{P_T, P_{T-1}, \dots, P_0\}$  accordingly. The initial smoothed estimate is  $\vec{x}_T^s = \vec{x}_T$  with the covariance  $P_T^s = P_T$ . Then the smoother is applied from the last time step to the first (i.e.,  $t = T - 1, \dots, 0$ ) according to the description given in [9]

### 3.4. Extended Kalman Filter

According to (1) the stochastic system model can be described as follows:

$$\begin{cases} \vec{x}_{k+1} = A\vec{x}_k + \vec{\xi}_k, \\ \vec{r}_k = \vec{h}(\vec{x}_k) + \vec{\eta}_k, \end{cases} \quad (6)$$

$\vec{h}(\vec{x}_k)$  – vector-valued function containing  $N$  elements

$$\sqrt{(x_k^m - x_n)^2 + (y_k^m - y_n)^2 + (z_k^m - z_n)^2}, \quad n = 1, \dots, N,$$

$\vec{r}_k = [r_{1,k} \ \dots \ r_{N,k}]^T$  – measurement vector. The measurement noise covariance matrix

$$R_k = \begin{cases} R_0, & k < D \\ \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + H_k P_k H_k^T, & k \geq D \end{cases} \quad \text{is non-stationary with the initial value of}$$

$$R_0 = \text{diag}\left(\underbrace{\sigma_r^2, \dots, \sigma_r^2}_N\right), \quad H_k - \text{Jacobian matrix. As for the other parameters used in (6), they are the}$$

same as for the linear filter described above. The implementation is done according to the [9].

### 3.5. Unscented Kalman Filter

The problem-dependent filter parameters are the same as for the Extended Kalman Filter and were described above. The measurement noise covariance matrix

$$R_k = \begin{cases} R_0, & k < D \\ \frac{1}{D} \sum_{d=k-D}^k \vec{s}_d \vec{s}_d^T + \Upsilon_k W \Upsilon_k^T, & k \geq D \end{cases} \quad \text{is non-stationary with the initial value of}$$

$$R_0 = \text{diag}\left(\underbrace{\sigma_r^2, \dots, \sigma_r^2}_N\right),$$

$\Upsilon_k = \vec{h}(X_k)$ ,  $X_k$  – array of sigma points,  $W$  – matrix of weights. The filter implementation complies with the [9].

### 3.6. Particle Filter

To describe the beacon's position the following state vector  $\vec{x} = [x^m \ y^m \ z^m \ v^x \ v^y \ v^z]^T$  is used, where  $\vec{x} \sim \mathbb{N}(\vec{x} | \vec{x}_0, \Sigma)$ . The initial estimate is  $\vec{x}_0 = [x_0^m \ y_0^m \ z_0^m \ 0 \ 0 \ 0]^T$ , where  $\vec{u}_0 = [x_0^m \ y_0^m \ z_0^m]^T = (B^T B)^{-1} B^T \vec{g}_0$ ,  $\Sigma = \text{diag}(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_v^2, \sigma_v^2, \sigma_v^2)$ .

Based on the distribution  $\mathbb{N}(\vec{x} | \vec{x}_0, \Sigma)$  the set of particles  $X_0$  is generated. For the each particle

$$\vec{x}_{j,k} \quad \text{the weight } w_{j,0} = 1/J \text{ is set, with the sum of weights } w_\Sigma = \sum_{j=1}^J w_{j,0} = 1, \quad j = 1, \dots, J.$$

The prediction step is done according to  $X_k = A X_{k-1}$ . During the update step for the each particle  $\vec{x}_{j,k}$  distances  $\hat{r}_{n,j,k}^2$  to the each of the receivers are calculated according to the equation

$$\hat{r}_{n,j,k}^2 = (x_{j,k}^m - x_n)^2 + (y_{j,k}^m - y_n)^2 + (z_{j,k}^m - z_n)^2, \quad n = 1, \dots, N, \quad j = 1, \dots, J$$

We assume that distances between the  $n^{\text{th}}$  receiver and the beacon are distributed normally  $p(\hat{r}_{n,j,k}^2) = \mathbb{N}(\hat{r}_{n,j,k}^2 | r_{n,j,k}^2, \sigma)$ , where  $r_{n,j,k}^2$  – measurement of  $n^{\text{th}}$  receiver. Weight for each particle calculate according to the equation  $w_{j,k} = p(\hat{r}_{1,j,k}^2) \cdot \dots \cdot p(\hat{r}_{N,j,k}^2)$ .

Then the particles with the weights less than  $\beta$  are removed from the set and the weights are normalized according to the equation  $\vec{w}_k = \frac{1}{\sum_{j=1}^{\Omega} w_{j,k}} \vec{w}_k$ , where  $\Omega$  is the size of the new particle set.

The state vector is computed as  $\vec{x}_k = \sum_{j=1}^{\Omega} w_{j,k} \vec{x}_{j,k}$ . After that new particles are added to the set according to the normal distribution  $\mathbb{N}(\vec{x} | \vec{x}_k, \Sigma_x)$  up to the number  $J$ . Then the next prediction step is executed.

#### 4. Results

During the experiment the mobile robot equipped with the active beacon moved along the  $4.6 \times 7$  m rectangular path (figure 4). The array of  $N = 4$  receivers was statically placed outside of that path with the center at  $(0;0)$ .

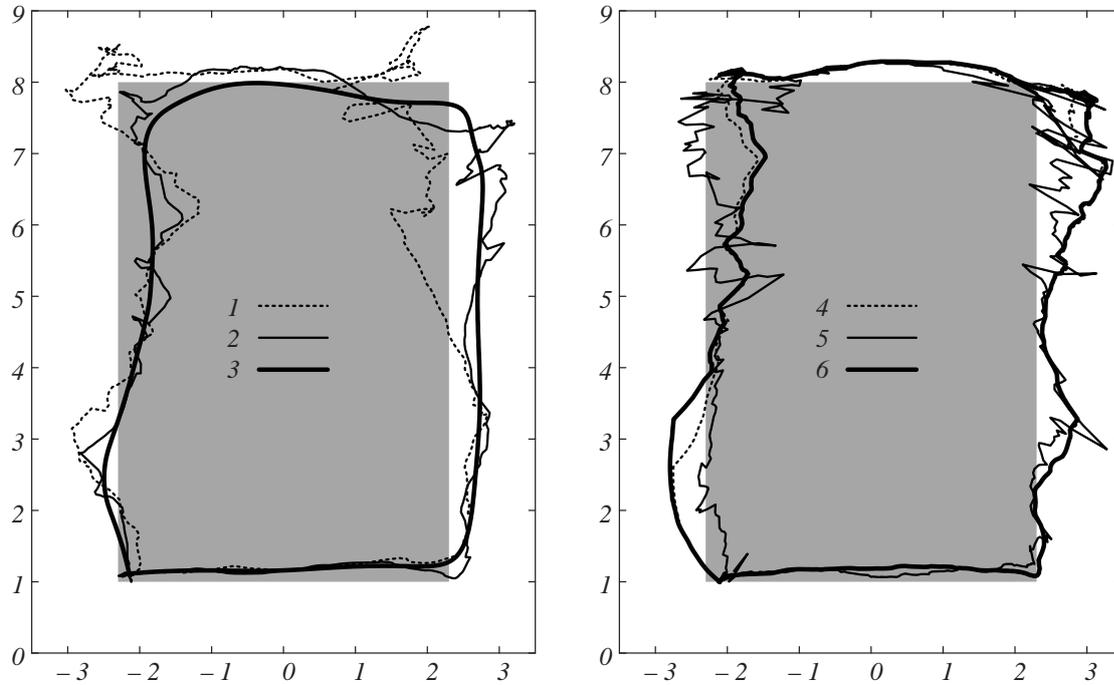


Figure 4. Beacon's path estimates: 1 – least squares method with the exponential smoothing, 2 – Kalman Filter, 3 – Kalman Filter with Rauch-Tung-Striebel smoother, 4 – Unscented Kalman Filter, 5 – Particle Filter, 6 – Extended Kalman Filter.

For the each beacon's position a total of 6 estimates were computed using the methods described above. To make a comparison between the performances of different methods the average of root mean square errors (RMSE) was used. The results are shown in the table 1. Clearly, the Rauch-Tung-Striebel smoother is preferable to estimate the path made by the active beacon which a mobile robot has to follow.

**Table 1.** A comparison between different methods using root mean square errors (RMSE)

	RMSE( $x^m$ ), m	RMSE( $y^m$ ), m
ES	0.4617	0.5737
KF	0.2910	0.5433
RTS	0.2230	0.4223

<b>EKF</b>	0.2478	0.4987
<b>UKF</b>	0.2305	0.4349
<b>PF</b>	0.1918	0.4958

## 5. References

- [1] Spencer M, Jones D, Kraehling M and Stol K 2007 Trajectory based autonomous vehicle following using a robotic driver, *Australasian conference on robotics and automation, Brisbane: Australian Robotics and Automation Association*, Sydney, Australia, pp. 1-10.
- [2] Wong N, Stol K A, Chambers C and Halkyard R 2009 Autonomous vehicle following using a robotic driver, *IEEE 15th international conference on mechatronics and machine vision in practice*, Auckland, New Zealand, 2-4 December 2008, pp.115-120.
- [3] Changchen Z, Weihai C, Zhiwen Z and Jingmeng L 2013 An RGBD data based vehicle detection algorithm for vehicle following systems, *Industrial Electronics and Applications (ICIEA) 2013 8th IEEE Conference on*, Melbourne, Australia, 19-21 June 2013, pp. 1506-1511.
- [4] Vidal R, Shakernia O and Sastry S 2004 Following the flock: distributed formation control with omnidirectional vision-based motion segmentation and visual servoing, *IEEE Robotics & Automation Magazine*, December 2004, vol. 11, no. 4, pp. 14-20.
- [5] Krejsa J and Vechet S 2012 Infrared beacons based localization of mobile robot, *Electronics And Electrical Engineering*, no. 1(117), pp. 17-22.
- [6] Ng T C, Ibanez-Guzman J, Shen J, Gong Z, Wang H, and Cheng C 2004 Vehicle following with obstacle avoidance capabilities in natural environments, *Proc. IEEE Int. Conf. Robotics and Automation ICRA '04*, New Orleans, LA, USA, April 2004, vol. 5, pp. 4283-4288.
- [7] Shoval S and Borenstein J 2001 Measuring the relative position and orientation between two mobile robots with binaural sonar, *Proceedings of the ANS 9th International Topical Meeting on Robotics and Remote Systems*, Seattle, Washington, USA, 4-8 March 2001.
- [8] Hoflinger F and Reindl L M 2012 TDOA based localization using interacting multiple model estimator and ultrasonic transmitter/receiver, *IEEE 9th International Multi-Conference on Systems, Signals and Devices*, Chemnitz, Germany, 20-23 March 2012, vol.1, pp.1063-1069.
- [9] Hartikainen J and Sumo S 2011 Optimal filtering with Kalman filters and smoothers, *A Manual for the Matlab toolbox EKF/UKF*, Department of Biomedical Engineering and Computational Science, Aalto University School of Science, Espoo, Finland, August 2011, p.129.
- [10] Wang J, Ding W and Wang J 2007 Improving adaptive Kalman filter in GPS/SDINS integration with neural network, *Proceedings of ION GNSS 2007*, Fort Worth, TX, USA, 25-28 September 2007, vol.1, pp.571-579.

# Ultrasonic rangefinder with the submillimeter resolution as a part of the rescue robot's sensor system

S Goll<sup>1,2</sup> and J Maximova<sup>1,2</sup>

<sup>1</sup>Department of Information-Measuring and Biomedical Engineering, Ryazan State Radio Engineering University (RSREU), Gagarin Street, 59/1, Ryazan, RU

<sup>2</sup>LLC KB Avrora, Skomoroshinskaja Street 9, of.3, Ryazan, RU

**Abstract.** The main goal of the research is to increase the measurement resolution of the ultrasonic rangefinders to meet the needs of vital signs noncontact registration based on the chest movements. The two-phase method is proposed to make distance estimates by sending the probe pulse trains, calculating the phase spectrum of the echo signal's envelope and tracking its relevant components. During the first phase the rough TOF based estimate is made. During the second phase this estimate is corrected based on the phase spectrum of the echo signal's envelope, the phase ambiguity is removed and the relevant components are determined. The final estimate of the human chest displacement is calculated based on these relevant components. The output data rate is the same as for the TOF measurements, but the measurement resolution is increased to the one hundredth of the ultrasonic wavelength. The experiment results are provided for the both model and the real human chest displacements caused by the respiration and heartbeat processes.

## 1. Introduction

The extreme mobile robotic systems, the safety-critical systems (particularly, the ones designed for the drivers, pilots, motormen) as well as medical equipment (e.g., diagnostic and therapy instruments) need the low-cost precision rangefinders with the effective range up to 1 meter and the output data rate of at least 20 Hz to measure both absolute and relative distances. Such rangefinder sensors can be used to provide the noncontact measurements of the person's pulse and respiratory rates – the information critical for the rescue operations planning, human-factor accidents prevention and the medical equipment synchronization with the patient's biorhythms [1].

Search and Rescue (SAR) operations caused by industrial disasters and military conflicts can be dangerous for the SAR crew members. The main tendency here is to minimize the human presence in the dangerous zone by relying on the telemetry data (i.e., victims' vital signs and locations) to optimally plan and execute the rescue operation, including the SAR crew members' efforts and the rescue robots, both autonomous and remote controlled.

Detection of the injured person's vital signs, mainly pulse and respiratory rates, poses a nontrivial task even for a remote controlled rescue robot. For that task video and thermal cameras are considered to be the most informative sensors. Due to amount of data broadcasted by these sensors, they require wired connection (e.g., LAN) between the robot and the human operator or the broadband wireless connection in case the former is not possible. However, the radio wave-based data transmission can lead to detection mistakes due to interferences and non-line-of-sight and near-line-of-sight transmissions. Moreover, the usage of thermal camera to detect the body temperature can sometimes lead to the dead body being mistaken for the living person.

Another way to measure the human vitals is to make contact measurements. In that case robot must place the electrodes on the particular parts of the human body, which leads to a handful of problems like the process of electrodes placement or detection and recognition of the body parts where the electrodes should be placed. Clearly, these can be quite dangerous operations to be performed on an injured person.

## 2. Related work

The noncontact measurements of the pulse and respiratory rates can be taken by the means of the short-range radars since it is possible for the single integrated circuit to contain two 77 GHz UHF transmission lines (one transmitting line and one receiving line) as well as the low-frequency processing channel and the ADC [2]. The data gathered that way can be affected by the electromagnetic interference and other factors; hence, we propose to increase the robustness of the radar measurements by the means of the secondary noncontact measurement channel based on the ultrasonic sensors.

The commonly used Time-of-Flight Principle [3] is based on the measurement of the time interval between the start of the ultrasonic wave emission and the moment when the reflected wave causes the difference between the current air pressure and the air pressure at rest to exceed the given threshold for the first time. The measurement resolution of the ToF-based sensor depends both on the properties of the Timer/Counter used and the ultrasonic signal wavelength. The ToF measurement resolution for the 40 kHz ultrasound is, at a rough estimate, of 8 mm. At the same time, a lot of the ToF-based sensors are featured with the resolution up to the one fourth of the ultrasonic wavelength. To achieve such resolution various methods can be used: the adaptive threshold, comparisons for the positive and negative half-waves, etc. However, all these modifications are not enough to achieve the submillimeter resolution.

Another attempts to improve the measurement resolution of the ultrasonic rangefinders are based on the Linear Frequency Modulation or Composite Modulation [4]. However, these methods require broadband ultrasonic receivers and transmitters sufficiently increasing the hardware cost.

## 3. The approach

In this paper we propose an algorithm to improve the measurement resolution of the low-cost single-tone 40 kHz ultrasonic rangefinders. These inverse piezoeffect based sensors are commonly used in the automotive industry to build Advanced Driver Assistance Systems (ADAS).

The cyclic movements of the human chest are caused by both the respiration process and the cardiac activity with the movement amplitudes of 4 to 12 mm and 0.5 mm accordingly [5]. The cardiac cycle is more frequent than the respiration cycle and able to occur up to 200 times per minute. According to the Nyquist–Shannon–Kotelnikov sampling theorem the measurement rate of 20 Hz is sufficient to perform the vital signs detection. Due to the comparatively high sound speed in the air medium it is possible to obtain up to 100 measurements per second at the distance of up to 1 meter using the ToF method. Despite this, the measurement resolution of one fourth of the 40 kHz ultrasonic wavelength is deficient in pulse and respiratory rates registration.

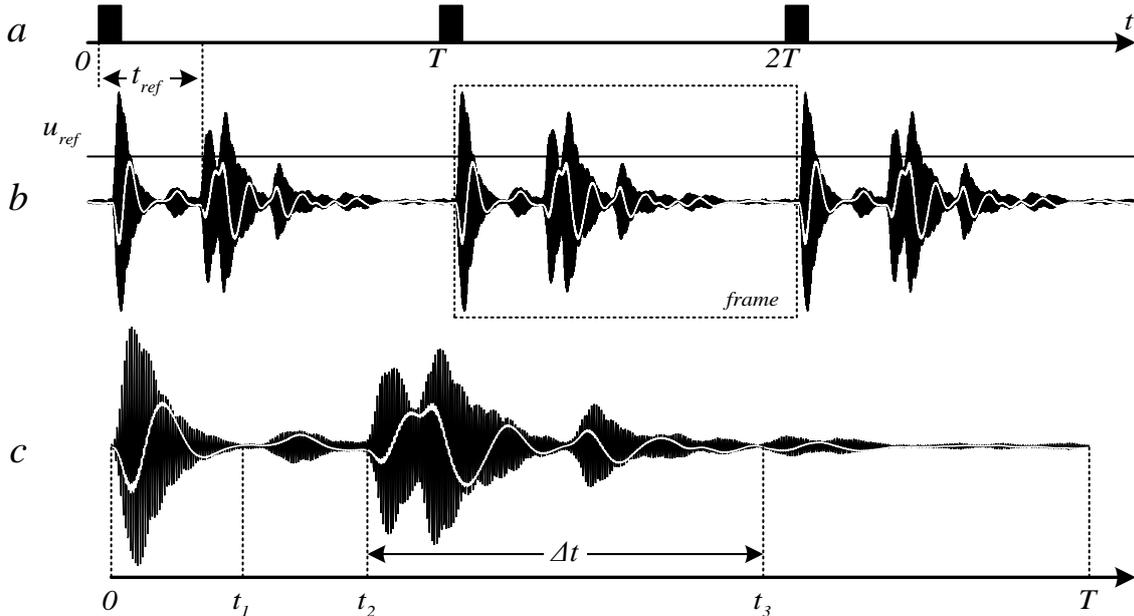
We suggest an approach to improve the measurement resolution without changing the output data rate. First, some (for example, ToF) method is used to obtain a rough distance estimate  $d_{ref}$ . Second, the proposed method is used to  $d$  estimate the displacement relative to the rough estimate  $d_{ref}$ . Opposing to the ToF method which uses the point in time when the threshold is exceeded as the input data, the proposed method uses all the readings of the echo signal gathered with the sampling rate  $f_d = 200 \text{ kHz}$ .

The figure 1a shows pulse trains containing  $N = 8$  probe pulses each with the probe pulse frequency set to  $f_{us} = 40 \text{ kHz}$ . The pulse trains are transferred to ultrasonic transmitter with the frame period  $T = 10 \text{ ms}$ . The black-colored regions in the figure 1b represent the digital signal  $u$  which is formed from the received frame sequence and can be described as the amplitude overmodulated

signal. The white line in the figure 1b represents the demodulated envelope of the receiver's signal. To compute the envelope the modulated signal is multiplied by the sinusoidal carrier signal  $v = V\sin(2\pi f_{us}t + \phi_v)$  sampled with the rate  $f_d$ ; then the low pass recursive filter with the transfer function  $\frac{b_0z^2+b_1z+b_2}{z^2+a_1z+a_2}$  is applied to the multiplication result. The figure 1c shows the enlarged version of the frame surrounded with the dotted rectangle in the figure 1b. Each frame contains the deterministic transmitted pulse received when the transmitter emits the ultrasonic wave. This signal starts from the beginning of the frame and ends at time  $t_1$ . In the case some object or group of objects is present in the area covered by the rangefinder's beam pattern and these objects are large enough to reflect the ultrasonic wave with the frequency  $f_{us}$ , the frames also contain the reflected echo signal at the time interval  $\Delta t = [t_2 \dots t_3]$ .

The moment  $t_1$  when the transmitted pulse ends is constant for every frame and is computed as  $t_1 = \frac{N + n_1}{f_{us}}$ . We define the frame containing the rough ToF-based measurement of the distance  $d_{ref}$  to the closest object in the sensor's beam pattern as a reference frame. The rough estimate  $d_{ref}$  is proportional to the time interval  $t_{ref}$  which is determined when the echo signal exceeds the comparison threshold  $u_{ref}$  after the time  $t_1$ . For the every frame received after the reference frame time  $t_2 = t_{ref} - \frac{n_2}{f_{us}}$  remains constant until the new reference frames arrives changing the value of  $t_{ref}$ . The coefficients  $n_1$  and  $n_2$  represent the adjustable offsets with the integer values.

The reference frame is used as a starting point for the proposed method to compute its distance estimate and can be set (and changed later) either by the human operator's request or automatically if the absolute distance measurements acquired by the means of the ToF method lie in some predetermined intervals. The moment  $t_3 = t_2 + \Delta t$  sets the right boundary of the echo signal's time interval of the constant duration  $\Delta t = const$ .



**Figure 1.** a – probe pulse trains, b – received reflected signal, c – a single frame as it appears in the receiver.

On the  $\Delta t = [t_2 \dots t_3]$  interval of the each frame the Fast Fourier Transform is applied to the echo signal's envelope and then for the each of its frames the phase spectrum is computed.

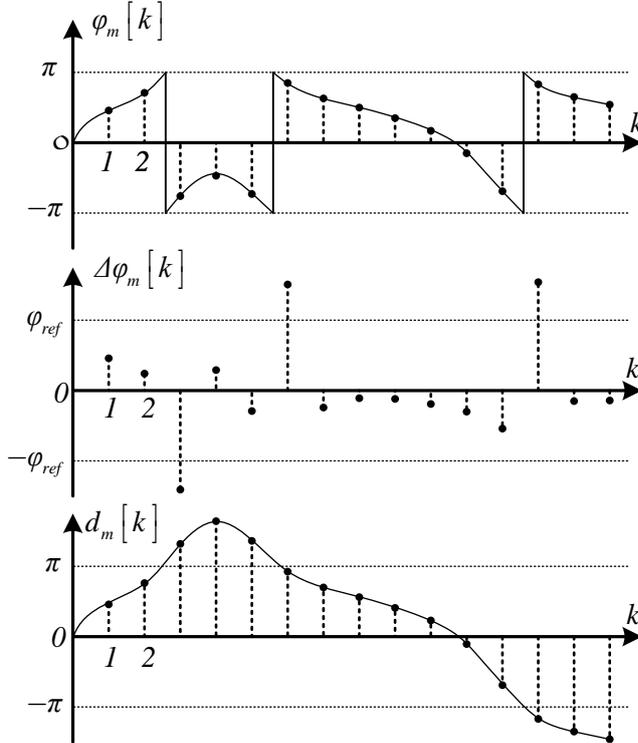
The next step is the selection of the  $M = 10$  sequential spectrum components starting with the 2<sup>nd</sup> component which are expressed as  $\vec{\varphi}[k] = (\varphi_1[k] \ \varphi_2[k] \ \dots \ \varphi_M[k])^T$ , where  $k$  is the number of frame relative to the reference frame. This selection takes place at the end of the each frame period  $T$ . Each component  $\varphi_m[k]$  is used to estimate the displacement  $d_m[k]$  relative to the rough estimate  $d_{ref}$ . This estimate is calculated as  $d_m[k] = \varphi_m[k] + \lambda_m[k] \cdot \pi$ , where  $\lambda_m[k]$  is the number of  $m^{\text{th}}$  component's phase unwrappings and is calculated as follows:

$$\lambda_m[k] = \lambda_m[k-1] + g(\Delta\varphi_m[k]), \quad (1)$$

where  $\Delta\varphi_m[k] = \varphi_m[k] - \varphi_m[k-1]$ ,  $g(\zeta) = \begin{cases} -1, & \zeta < -\varphi_{ref} \\ 1, & \zeta > \varphi_{ref} \\ 0, & \text{otherwise} \end{cases}$ ,  $\varphi_{ref}$  is the adjustable threshold of the

$\Delta\varphi_m[k]$ .

The figure 2 illustrates the computation of the displacement  $d_m[k]$ . When the new reference frame is set the number of phase unwrappings  $\lambda_m[0]$  is zeroed for the each phase component which provides the recurrent expression (1) with the initial condition. Therefore, for the each received frame the  $M = 10$  estimates are calculated for the absolute distance  $h_m$  between the sensor and the object of interest. Each of these estimates is calculated as  $h_m[k] = d_{ref} + d_m[k]$ , where  $m = 1, 2, \dots, M$ , and, apparently, represents the sum of the reference distance  $d_{ref}$  and the displacement  $d_m[k]$  computed using the proposed method. Let us note that  $N$ ,  $n_1$ ,  $n_2$ ,  $u_{ref}$ ,  $\Delta t$ ,  $\varphi_{ref}$ ,  $\vec{a} = [a_1 \ a_2]^T$ ,  $\vec{b} = [b_1 \ b_2 \ b_3]^T$ ,  $M$  are the structural parameters of the algorithm and are estimated empirically before the algorithm is applied.



**Figure 2.** Applying the phase ambiguity resolution to improve the dynamic range of the displacement estimation.

Algorithm: displacement estimation

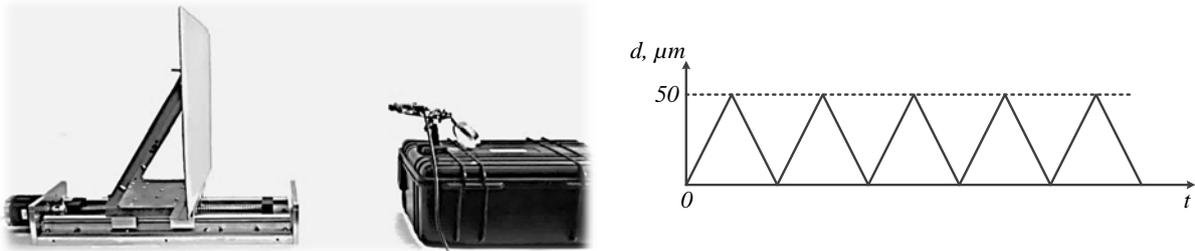
```

1 input:  $u$ 
2 output:  $\vec{d}=(d_1[k] \ d_2[k] \ \dots \ d_M[k])^T$ 
3 begin
4   multiply modulated signal by the carrier signal  $x_i=u_i \cdot v_i$ 
5   perform low pass filtering  $y_i=\vec{b}^T \vec{x}_i - \vec{a}^T \vec{y}_i$ 
6   if the  $k^{th}$  frame is ready then
7     if the  $k^{th}$  frame is the reference frame then
8       set the initial conditions
9        $\lambda_1[0] = \lambda_2[0] = \dots \lambda_M[0] = 0$ 
10       $\varphi_1[0] = \varphi_2[0] = \dots = \varphi_M[0] = 0$ 
11       $k=1$ 
12     else
13        $j$  – index of the reading corresponding to the moment  $t_2$  in the current frame,
14        $L$  – number of the readings in the interval  $\Delta t=[t_2 \dots t_3]$ 
15       extract the echo signal  $\vec{z}=(y_j \ y_{j+1} \ \dots \ y_{j+L})^T$ 
16        $\vec{\varphi}[k] = arg(FFT[\vec{z}])$ 
17       for the  $m=1,2,\dots,M$  phase spectrum components do
18          $d_m[k] = \varphi_m[k] + \lambda_m[k] \cdot \pi$ 
19          $\Delta\varphi_m[k] = \varphi_m[k] - \varphi_m[k-1]$ 
20          $\lambda_m[k] = \lambda_m[k-1] + g(\Delta\varphi_m[k])$ 
21       end
22     end
23   end
24 end

```

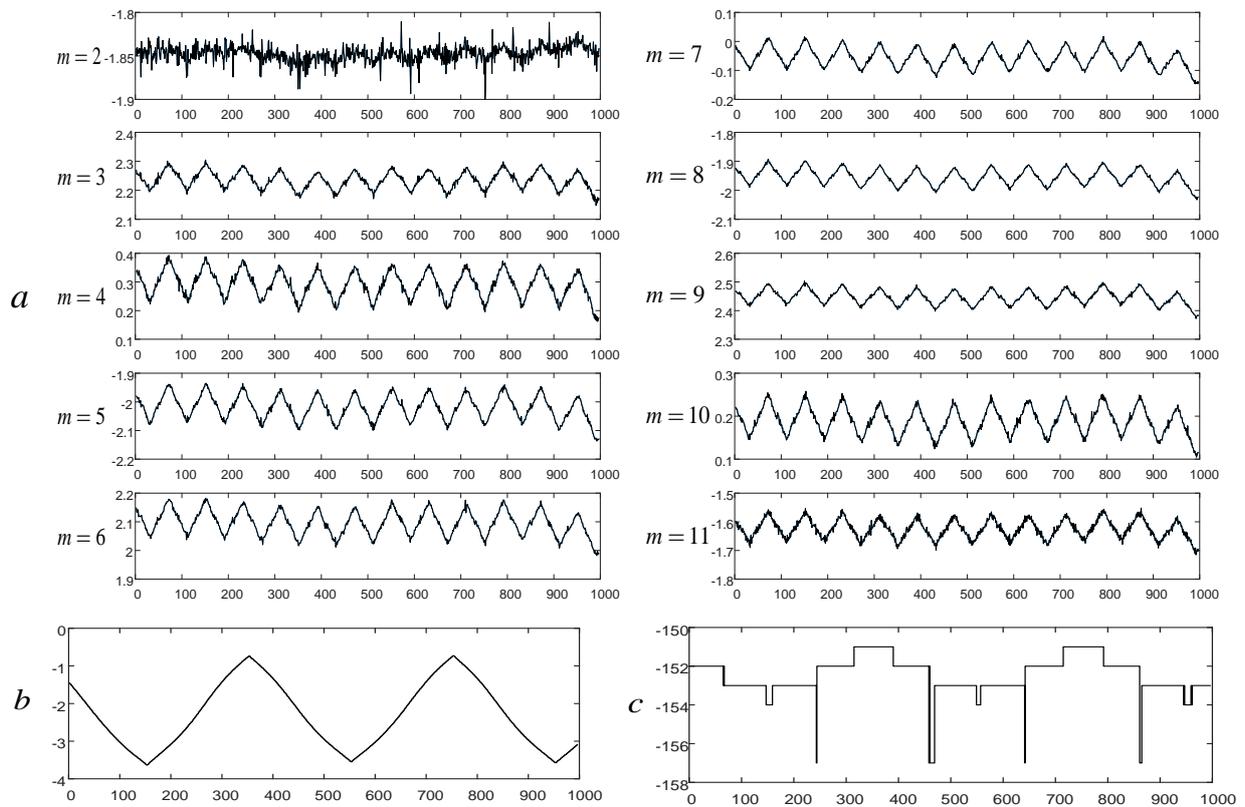
#### 4. Results

The proposed method was used to process data gathered from the ultrasonic rangefinder in three experiments. For the first experiment a rectangular aluminum sheet was used as the object of interest (figure 3). A step motor with the step of the  $1.56 \mu\text{m}$  was used to cause the periodic sawtooth-like motion of the sheet with the amplitude of  $50 \mu\text{m}$  along the axis of the rangefinder's beam pattern.



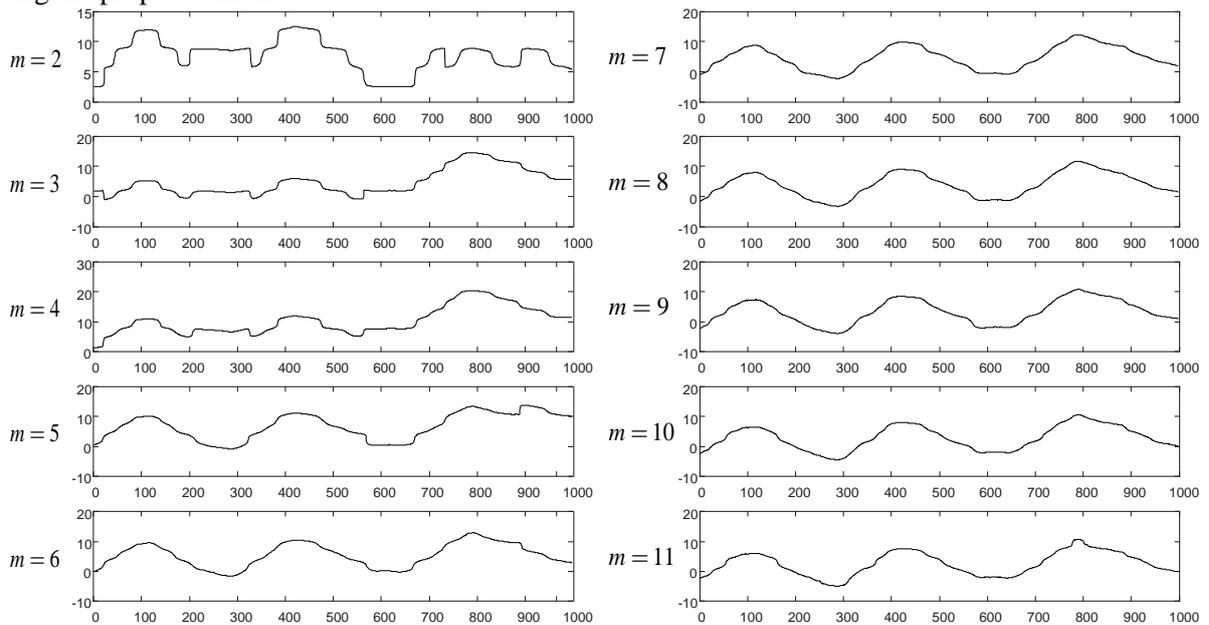
**Figure 3.** The experimental set-up and the sawtooth-like displacements of the sheet.

The 10 estimates of the aluminum sheet displacement with the amplitude of  $50 \mu\text{m}$  are shown in figure 4a. The figures 4b, 4c show the displacement estimates for the movements with the amplitude of  $4 \text{ mm}$  made by the proposed and the ToF methods respectively.



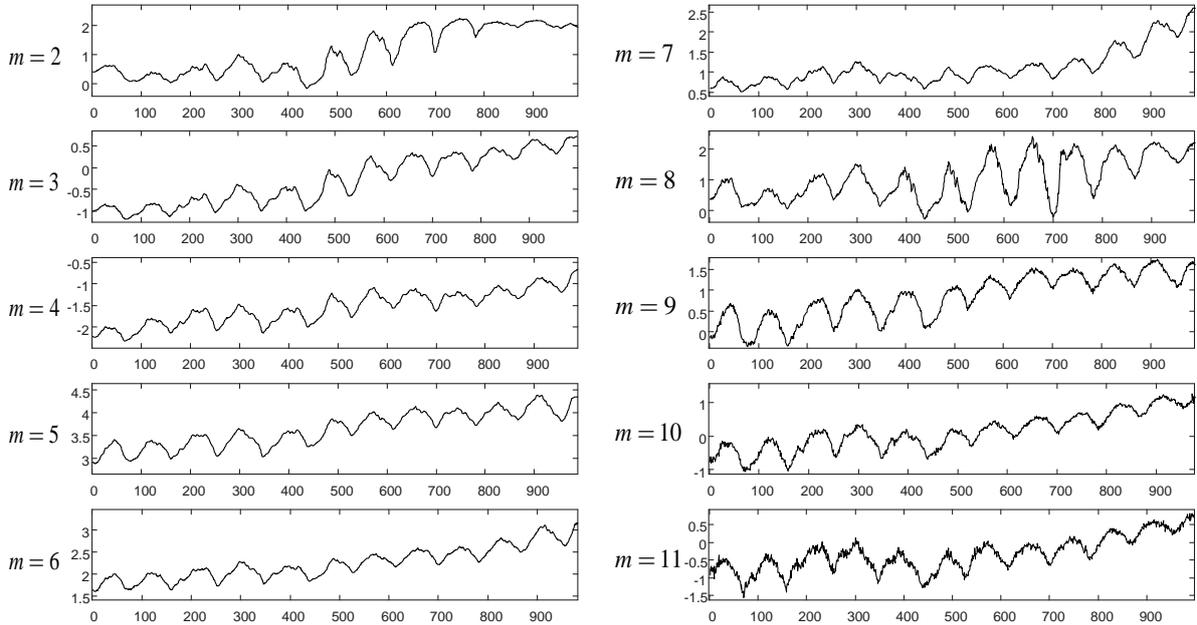
**Figure 4.** Aluminum sheet displacement estimates : a – 50  $\mu\text{m}$  amplitude, proposed algorithm, b – 4 mm amplitude, proposed algorithm, c – 4 mm amplitude, ToF method.

For the 2<sup>nd</sup> and the 3<sup>rd</sup> experiments the human chest was used as the object of interest. Figure 5 shows the 10 estimates of the human chest displacement during the respiration process computed using the proposed method.



**Figure 5.** Human chest displacement during the respiration process.

Figure 6 contains the 10 estimates of the chest displacement caused by the cardiac activity while the person was holding their breath.



**Figure 6.** Human chest displacement caused by cardiac activity.

## 5. Conclusion

It is possible to empirically select the relevant (i.e., the most correlated with the object's movements) displacement estimates from the series computed by the proposed method. For the 1<sup>st</sup> experiment these estimates are under numbers 5, 6, 10, for the 2<sup>nd</sup> experiment these are estimates under numbers 7, 8, 9 and for the 3<sup>rd</sup> experiment these are estimates under numbers 5, 6, 8.

Since the aluminum sheet used in the 1<sup>st</sup> experiment was moving with the amplitude of the  $50 \mu\text{m}$ , we can make a rough estimate of the proposed method's resolution which is less than  $50 \mu\text{m}$  and, consequently, less than one hundredth of the ultrasonic wavelength with the frequency of  $f_{us} = 40 \text{ kHz}$  making it sufficient for the purposes of respiration and heartbeat detection. However, the aforementioned estimate is highly conservative and is a subject of the future research.

According to the experimental results, the ToF method is inadequate for the respiration and heartbeat graphs plotting due to its the measurement resolution. It should be also noted that in all of the experiments a considerable part of the target object was covered by the sensor's beam pattern and the waveform of the echo signal's envelope depends highly upon object's shape. So, in the 1<sup>st</sup> experiment all the surface points move equally linear, but in the 2<sup>nd</sup> and 3<sup>rd</sup> experiments the shape of the surface is a subject to the nonlinear three-dimensional changes causing sufficient changes in the waveform of the corresponding echo signal's envelope. For that reason the ToF-based results contain more noise for the 2<sup>nd</sup> and the 3<sup>rd</sup> experiments than for the 1<sup>st</sup>. Consequently, for the most of the applications the measurement resolution of this method is in close proximity to the wavelength.

Since the displacements caused by the respiration and heartbeat can be considered to be concentrated in the small neighborhood of  $d_{ref}$  it is possible to provide the accurate estimates based on the phase components. The proposed method is robust to such kind of the displacements since the only thing that changes during the process is the set of the relevant phase spectrum components. The main source of such robustness is the fact that both the time when the envelope exceeds the comparison threshold  $u_{ref}$  and all the information it contains are used to perform calculations.

However, to survive the significant nonlinear changes in the object's surface over time some combinations of the specific phase components should be used.

Therefore, the proposed method is suitable for the small movement detection happening in any region of the sensor's beam pattern and can be applied by mobile robots to detect human's vitals during the SAR operation.

An algorithm for the automatic selection of the most relevant phase spectrum components or combinations of these components based on the a-priori information about the target object and the measuring conditions poses another subject for a future research. Topics of interest also include the impact of the rough estimate  $d_{ref}$  on the relevant phase components selection and the compensation of the distortions caused by the air streams on the territory covered by the sensor's beam pattern.

- [1] Grigoryev E M, Gurzhin S G, Zhulev V I, Kaplan M B, Kryakov V G and Proshin E M 2016 Technology and methods for formation of the complex magnetotherapy impact by the inductor array, *5TH Mediterranean conference on embedded computing, MECO 2016*, Bar, Montenegro, June 2016, pp. 419-422.
- [2] Texas Instruments datasheet IWR1443 Single-Chip 76- to 81-GHz mmWave Sensor, SWRS211B –may 2017–revised February 2018.
- [3] Huang S S, Huang C F, Huang K N and Young M S 2002 A high accuracy ultrasonic distance measurement system using binary frequency shift-keyed signal and phase detection, *The Review of scientific instruments*, vol. 73, no. 10, pp. 3671-3677.
- [4] Saad A, Radwan A, Sadek S, Obeid S, Zaharia G and El Zein G 2009 Remote monitoring of heart and respiration rate using a wireless microwave sensor, *12th International Healthcare, Hospital Supplies and Medical Equipment Shows (Saudi Medicare 2009)*, Riad, Saudi Arabia, May 2009.
- [5] Huang Y P, Wang J S, Huang K N, Ho C T, Huang J D and Young M S 2007 Envelope pulsed ultrasonic distance measurement system based upon amplitude modulation and phase modulation, *The Review of scientific instruments*, vol.78, no. 6, 06510.

# Semantic Grid Mapping based on Surface Classification with Supervised Learning

Torsten Engler, Felix Ebert and Hans-Joachim Wuensche

**Abstract**—LiDAR-based occupancy grid mapping can lead to overly conservative detection of obstacles in non-urban autonomous driving scenarios, e.g. grass in the middle of the lane is often interpreted as obstacle although it is actually driveable. We therefore aim to augment our current grid-based environment representation with additional information derived from pixel-level semantic segmentation in camera images. We project the resulting segmentation map onto an additional semantic layer in the environment grid representation by utilizing LiDAR data for pixel-to-cell association to improve our driveability analysis.

We apply supervised machine learning techniques for pixel-wise prediction of class labels. Datasets for non-urban environments are rare. Therefore, we created a custom dataset. Due to the huge effort necessary to create such a dataset, its size is relatively small and hence neural networks might not be able to train effectively. Thus, low numbers of training samples require a careful choice of the classifier and/or data augmentation techniques. We therefore compare classification performance of neural networks with random forest classifiers.

## I. INTRODUCTION

A semantic understanding of the current surroundings is a fundamental ability to improve autonomous driving performance. Thus, we enhance our current environment representation by semantic segmentation of camera images to improve our driveability analysis. We apply the segmentation results to distinguish between areas that are detected as occupied but actually are driveable (e.g. high grass) and those that are correctly detected as non-driveable (e.g. tree, hedge, stones covered with grass).

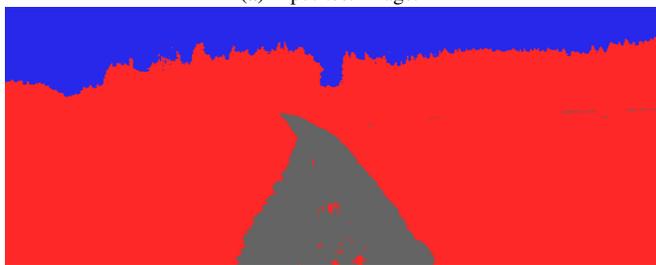
Our current environment representation is a multi-modal local terrain map obtained from Vision and LiDAR data [1]. It consists of several distinct layers where each layer corresponds to a specific environmental feature such as obstacle probabilities, colors and heights. The sensors used are a Velodyne HDL-64 LiDAR with 360° field of view, a front-facing color camera and a high-precision inertial navigation system. The resulting terrain map is obtained by spatiotemporal fusion of depth measurements, color information and motion estimation. Due to temporal accumulation, outliers are effectively filtered.

While this terrain map provides a conservative basis for obstacle-free path planning, we observe a high false positive rate of cell occupancy classification. Therefore, we propose a classification based approach to semantically distinguish between different surfaces and driveabilities as shown in Figure 1.

All authors are with the Institute for Autonomous Systems Technology (TAS) of the University of the Bundeswehr Munich, Neubiberg, Germany. Contact author email: torsten.engler@unibw.de



(a) Input test image.



(b) Result of the segmentation with the random forest with hand-crafted features.



(c) Segmentation result of the Pyramid Scene Parsing Network.

**Fig. 1:** Segmentation result for the random forest (b) and the neural network classifier (c). Blue corresponds to label *sky*, red to *vegetation* (undriveable), green to *grass* (driveable vegetation) and gray to *dirt road*.

Acquiring datasets of sufficient size poses one of the main challenges to apply supervised machine learning algorithms. Moreover, most readily available datasets for autonomous driving are designed for urban scenarios and therefore not directly applicable. It is necessary to create a distinct dataset for the situations and vegetations we encounter during offroad driving. Due to the high effort of manually creating large datasets, our dataset is comparably small. While deep neural networks perform very well in pixel-wise semantic classification, their performance degrades when trained with insufficient training data. In order to achieve good classification results, it is necessary to artificially increase the training data size by augmentation and/or transfer learning. In addition to neural networks, we evaluate random forests which don't require

large datasets and compare their classification performance to a neural network classifier.

This document is structured as follows: Section II gives a short overview of related work in the field of semantic segmentation with random forest or deep neural network classifiers. Section III details the dataset deployed for training. After covering our approaches to semantic segmentation in Section IV, we present the fusion approach in Section V and finally provide results and conclusions in Section VI.

## II. RELATED WORK

Semantic segmentation is the process of assigning a label to each pixel of an input image resulting in a segmentation map with same width and height as the input image. This is an essential task for image analysis and scene understanding in autonomous driving tasks. With the emergence of deep neural networks in recent years, significant progress has been made in the field of semantic segmentation, surpassing traditional methods. For a recent overview, the reader is referred to [2]. The Pascal Visual Object Classes [3] was the first challenge to incorporate semantic image segmentation. More recently, the Robust Vision Challenge provides good comparisons [4]. In this work, we compare a traditional segmentation method, the random forest classifier, with a deep neural network classifier.

### A. Random Forests

The random forest classifier [5] is an ensemble classifier grown from a number of random binary decision trees. Given a training set  $\{X, y\}$  with input  $X$  and corresponding class label  $y$ , each tree node selects a feature function  $h(X)$  and a threshold to split the input set to maximize the information gain. In this way, several trees are learned and together form the random forest. Prediction is achieved by applying the query input  $x$  to each tree individually, branching down until a leaf node is reached. A majority vote over all trees gives the final classification result.

Due to randomization, the random forest is able to mitigate the problem of overfitting and lack of generalization connected to single tree classifiers. Randomization is achieved by selecting a random subset of the input data either for each tree or each node and a random subset of the feature functions available for each node splitting.

Due to the general nature of feature functions applicable for this classifier, several approaches have been proposed for semantic image segmentation like structured class-labels [6], semantic textons [7], or a combination of multiple features like textons, color, filterbanks and HOG features [8].

### B. Deep Neural Networks for Semantic Image Segmentation

Deep neural networks have their origin in object classification. By introducing masks for convolutions and learning the weights of the masks, the network is able to differentiate image regions based on their context. The task of pixel-level image segmentation is inherently more challenging and can be done in different ways. The information gathered by the network has to be upsampled again to obtain a pixel-wise information. The first possibility are R-CNNs (Regions with

CNN Features) [9]. In a first step, the network extracts image regions based on a trained object detection. Afterwards, the classification decision is made based on a majority of all image regions where the pixel is part of.

Another popular approach is the encoder-decoder architecture. The encoder is used to extract discriminative features of the image region and the decoder is used to output dense pixel-wise labels based on the features. The category of these networks is also called fully convolutional networks (FCN) [10]. In contrast to object detection networks, these networks don't have fully connected layers and consist solely of convolutional layers. Therefore, there are no constraints on the size of the input image. However, the output of those networks usually is of low resolution and necessary upsampling leads to fuzzy object boundaries. Different strategies are applied to reduced this issue, e.g. SegNet [11]. Most modern segmentation networks differ in terms of the upsampling part, while using proven techniques for the feature extraction.

## III. DATASET

We deploy a handcrafted dataset for training the random forest and the neural network classifier. Data was recorded for five different dirt roads where camera images, LiDAR scans and egomotion data was captured. Special attention was paid to recording data at different times of day and different exposure conditions. Four of these five recordings are used for training and the fifth exclusively for testing. A random set of around 200 images with low correlation between the images was selected for training data generation. Data is split into 80% training data, 20% validation data and 15 images from the test stream that are not used during training. In order to keep inference times small and be able to process images from different sources, the images are downscaled to  $512 \text{ px} \times 512 \text{ px}$  which proved sufficient for our classification task. These images were labeled manually pixel-wise with four classes: *dirt road*, *sky*, *vegetation* and *grass*, where *grass* is driveable and *vegetation* is not. The intention to distinguish between driveable and undriveable vegetation is to classify patches of grass in the middle of the lane that might be misclassified as obstacle by the environment mapping module.

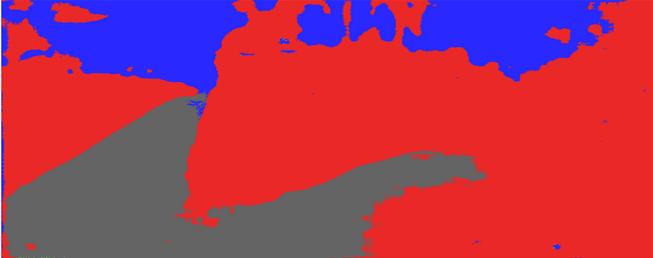
## IV. SEMANTIC IMAGE CLASSIFICATION

We apply the Extra-Trees classifier [12] as a representative random forest algorithm. It selects a random subset of the training data and feature functions for each node split. Features are computed as a weighted sum of responses of a rectangular region around the respective pixel location. These include hue, saturation, the green channel, the *ndi* and *exg* vegetation indices [13] as well as parts of the MR8 filter bank [14]. Analysis of feature importances along the trees showed that edge features of the filter bank had no significant impact on the classification result. Therefore, these were left out to reduce computation time of the feature vectors. The final forest consist of 36 random trees.

Correctly implementing and comparing different architectures of deep neural networks still poses a major challenge.



(a) Input testing image to the network.



(b) Segmentation result of the Pyramid Scene Parsing Network.

**Fig. 2:** Scene of the testing dataset with a forking road. The training data doesn't contain a visually similar scene, yet the network is able to at least partially generalize from the training data.

To alleviate network selection, training and performance comparison of different network architectures, we adopt the Semantic Segmentation Suite [15]. This suite provides methods to train different state-of-the-art network models with any dataset using the deep learning framework *Tensorflow*. After comparing different models trained with our custom dataset, the Pyramid Scene Parsing Network (PSP) [16] turned out to be best performing and hence was selected for further application and comparison with the random forest approach.

Our dataset is comparatively small for training of deep neural networks. Therefore, we applied data augmentation to generate additional training data. During data augmentation it is important to alter the data based on changes that are likely to occur in real-world scenarios. This includes vertical flipping of the image as well as small rotations up to  $10^\circ$ . Other alternations like flipping horizontally, or brightness changes of the images need to be applied with caution. For instance a brightness change of the scene cannot be easily simulated by altering the resulting picture.

## V. FUSION APPROACH

The fusion strategy of the camera and the LiDAR as described in [1] is crucial for all approaches. The fundamentals of the fusion are equal in each tested scenario and only depend on the relative pose of camera and LiDAR and on the egomotion. Each 3D point of the LiDAR is associated with a cell in the occupancy grid and projected back to the camera image. Afterwards the pixel's color is used to color the associated cell. Additionally, each cell has a decay rate to reduce the impact of wrongly accumulated colors. Special consideration is necessary in terms of occlusion. The LiDAR sensor and the camera are mounted on different parts of the vehicle. Thus, points visible by the LiDAR are not necessarily visible in the camera image.

## VI. RESULTS AND CONCLUSION

Both approaches, random forests and neural networks, were able to achieve a reasonable classification rate for our dataset. Table I shows a comparison of several performance scores for both approaches. Besides precision and recall, two further performance metrics are given which provide a combined measure. The F1 score, the harmonic mean of precision and recall, and the Intersection over Union (IoU) measure. We achieve average IoU values of 0.933 for the PSP network and 0.920 for the random forest.

**TABLE I:** Classification rates and scores of the random forest (RF) and the Pyramid Scene Parsing Network (PSP). The scores are averaged over 15 different images from the validation set (same images for each approach).

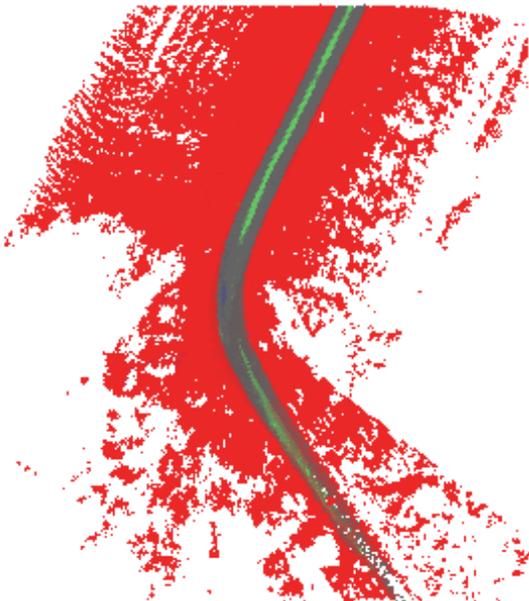
Vegetation	PSP	RF
Precision	0.995	0.987
Recall	0.992	0.993
F1 Score	0.993	0.990
IoU	0.987	0.980
Road	PSP	RF
Precision	0.981	0.989
Recall	0.987	0.990
F1 Score	0.984	0.990
IoU	0.968	0.980
Sky	PSP	RF
Precision	0.967	0.937
Recall	0.964	0.919
F1 Score	0.966	0.928
IoU	0.934	0.866
Grass	PSP	RF
Precision	0.883	0.854
Recall	0.948	0.698
F1 Score	0.914	0.768
IoU	0.842	0.624

Note that these values are comparatively high. This is due to the low number of classes. Furthermore we consider relatively simple scenes, where substantial parts of the images consist of sky and vegetation and hence misclassifications in the other two classes have less impact on the overall average values. It can be seen, that the segmentation quality of both approaches is about the same for *vegetation* and *road*. However, the classification scores for *grass* indicate that the neural network is better at distinguishing grass in the middle of the road from other vegetation.

Despite having similar IoU values on our test dataset, the random forest approach shows some disadvantages. The size and thus the inference time grows with the size of the training set as well as with the number of classes. Hence, having more data leads to larger trees with larger memory footprint and inference times, whereas these values are constant for the deep learning approach. This is the limiting factor for generalization to other road types or more classes. Furthermore, due to the



(a) Grid colored with the color camera image as described in Section V.



(b) Grid colored with the resulting segmentation map of the Pyramid Scene Parsing Network.

**Fig. 3:** Accumulation of colors in the grid surrounding the ego vehicle.

current feature selection, the forest approach is not able to incorporate spatial context into the inference. The low recall value for grass in Table I indicates that the amount of false negatives is high which coincides with the observation in Figure 1b that the vegetation at the center of the lane is misclassified as *road* or *vegetation*. The main advantage of random forests is that for a very small training set of around 20 images, it already leads to reasonable classification results.

Despite a small data set, the network is able to learn a good representation of the data. It seems to be helpful to apply data augmentation to artificially increase the training data size. In contrast to the random forest, the segmentation of

the network takes spatial information into account. Grass in the middle of the lane is classified as driveable even though it looks visually similar to vegetation in other parts of the image. More importantly, the network is able to generalize very well. In the scene with a parting road (see Figure 2) the segmentation is good even though the training data did not contain a similar case.

In terms of segmentation quality the network outperforms the random forest approach. Therefore, the segmentation results of the CNN are used subsequently to color our environment representation as seen in Figure 3. The segmented grid around the vehicle can be used to mark obstacle cells due to grass in the middle of the road as driveable. In the future, our results can be used to assist road network tracking and crossroad detection.

## REFERENCES

- [1] H. Jaspers, M. Himmelsbach, and H.-J. Wuensche, "Multi-modal Local Terrain Maps from Vision and LiDAR," in *Proceedings of IEEE Intelligent Vehicles Symposium (IV)*, Redondo Beach, CA, USA, Jun. 2017.
- [2] X. Liu, Z. Deng, and Y. Yang, "Recent Progress in Semantic Image Segmentation," *Artificial Intelligence Review*, 2018.
- [3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
- [4] "Robust Vision Challenge," <http://www.robustvision.net/>, 2018.
- [5] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [6] P. Kotschieder, S. R. Bulò, H. Bischof, and M. Pelillo, "Structured Class-Labels in Random Forests for Semantic Image Labelling," *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2190–2197, 2011.
- [7] J. Shotton, M. Johnson, and R. Cipolla, "Semantic Texton Forests for Image Categorization and Segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2008, pp. 1–8.
- [8] F. Schroff, A. Criminisi, and A. Zisserman, "Object Class Segmentation using Random Forests," *Proceedings of the British Machine Vision Conference*, pp. 54.1–54.10, 2008.
- [9] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CoRR*, vol. abs/1311.2524, 2013. [Online]. Available: <http://arxiv.org/abs/1311.2524>
- [10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3431–3440.
- [11] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *CoRR*, vol. abs/1511.00561, 2015. [Online]. Available: <http://arxiv.org/abs/1511.00561>
- [12] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely Randomized Trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, 2006.
- [13] E. Hamuda, M. Glavin, and E. Jones, "A Survey of Image Processing Techniques for Plant Extraction and Segmentation in the Field," *Computers and Electronics in Agriculture*, vol. 125, pp. 184–199, 2016.
- [14] M. Varma and A. Zisserman, "Classifying Images of Materials: Achieving Viewpoint and Illumination Independence," in *Computer Vision — ECCV 2002*, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 255–271.
- [15] "Semantic Segmentation Suite," <https://github.com/GeorgeSeif/Semantic-Segmentation-Suite>, 2018.
- [16] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6230–6239, 2017.