

Mobile Phone as a Sensor for Robot Navigation System

Radek Dorskocil, Vaclav Krivanek
Faculty of Military Technology
University of Defence
Kounicova 65, 612 00 Brno, Czech Republic
radek.dorskocil@unob.cz; vaclav.krivanek@unob.cz

Yves Bergeon
Écoles de Saint-Cyr Coëtquidan
56381 GUER Cedex, France
yves.bergeon@st-cyr.terre-net.defense.gouv.fr

Tomas Kornelly
Army of the Czech Republic
kornellyt@army.cz

Abstract – This article presents the results of research of a robust reconnaissance system. The system presented consists of two or more cooperating robots that use a mobile phone as their control unit. The robot is controlled via sockets from a remote server using the GSM standard, and the control of the hardware peripherals is done with FT312D circuit connected to a microprocessor. This solution seems to be inexpensive and user-friendly. The designed robot can determine its relative position based on photographs made by a general optical sensor. The photographs are subsequently processed on the server. The research work describes a method for detecting and determining the size of an object, while the object detected in a photograph is measured based on precise moves of the robotic device. The design reduces movement inaccuracies caused by chassis slipping on the terrain by using another robotic device. Thanks to the rapid progress of mobile phone technology development and the ever-decreasing purchase costs, it is now possible to use a mobile phone as a source of sensors and as a means of remote communication. This article presents the complete design of the system, the procedure for processing of the photographs, the algorithm for estimating the distance as well as a theoretical vision of autonomous navigation.

Keywords- robot; reconnaissance; mobile phone; optical sensor; remote; control; autonomous navigation

I. INTRODUCTION

This article describes the results of research conducted to design a robotic device that would use a mobile phone sensor for its visual navigation. Use of this sensor type resolves the aforementioned problematic in the context of commonly obtainable optical sensors.

Mobile phones are nowadays equipped with a camera of quality comparable to other commonly used compact cameras. Even though mobile phone cameras have a resolution of a few megapixels, their use demands work with unusual resolution ratio, lower sensitivity and specific distortion of the lens.

This paper has been done expecting additional research and development of a more complex system of autonomous analysis in an unknown environment.

The developed robotic device should be capable of autonomous navigation in an unknown environment, finding the desired object and determining its dimensions. The basic proposed concept is shown in Fig. 1.

The circuit board of the robot communicates with the phone and controls two independent engines on basis of received commands. A mobile phone with Android OS is used to obtain the optical information as well as to allow connection between the remote server and the rolling chassis. Information about the environment and current robot status are sent to the remote server where they are analysed. Other mobile phone sensors – i.e. GPS, motion sensors and sound system – can also be used if additional extensions are needed.

The remote server is used for remote control of the robot, for reception of the optical information and for its processing. In future, it could also be used as an intermediary between the robot and its remote control, done by another application in another mobile phone.

The system proposed in this paper connects several inexpensive robotic devices that communicate with each other to explore unknown environments. The first device is considered static after the initial detection cycle. In its field of view, it controls moves of the second mobile device and prevents slip errors by use of the previously established significantly important element. The second sensor detects and measures unknown object. By using two and more robots, advantages of stereometry can be used. A system

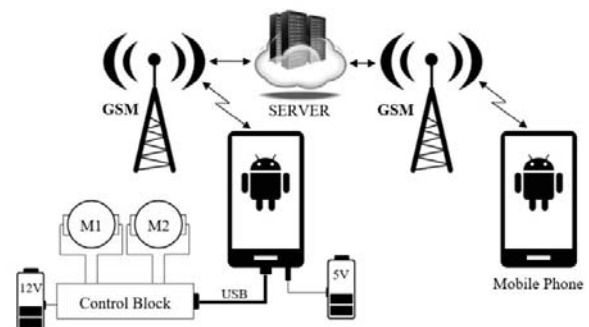


Figure 1. Basic proposition of the control system

designed in this way is then capable of being extended by additional elements and therefore its accuracy and robustness can be ensured.

In the next part of this paper follows an explanation of the selection and the design of the whole system's individual parts.

A. Current State Analysis

There can be found many research papers dedicated to measuring distances of objects from the recording device [1]. These papers are usually concentrating on indirect distance measurements because the actual size of the object can be calculated under the condition of knowing other needed parameters. In the published articles one can mostly find systems that imitate human sight. That means systems that determine distance through spatial vision granted by two separate pictures of the given object. A system designed this way usually includes two identical cameras.

In such a system, it is highly important to calibrate the cameras to reduce measurement errors [2]. The specifications of the cameras tend to be slightly different intrinsically, and unequal aging of their components then causes significant measurement errors.

Recently a few research projects have started to concentrate on distance estimation by use of more sensors. An interesting solution is published in a paper by Hou et al. [3], where authors propose increasing the safety of transportation vehicles based on using stereo vision to measure safe driving distance. There are also papers utilizing the static stereo system to measure object disparity in separate pictures [4].

On the contrary, only one sensor is used in the system described in this paper and this sensor is commonly available on mobile phone.

Up to this point, research concerning one lens was mostly concerned with the detection and monitoring of certain objects. Authors from Syracuse University have, in their paper [5], introduced a concept of monitoring the object using a network of cheap cameras that are capable to communicate with each other. In the case where the object is detected in the area of interest, the active camera sends a request to monitor the given area to the rest of the cameras in the system. That causes the object to be monitored from multiple points of view. This monitoring algorithm is implemented directly on the motherboard of the CITRIC camera. In their paper, the authors deal with limitations of resolution, memory size, processing speed, and transfer speed. This proposed system is aimed at detection and monitoring of a moving target. They had also introduced a system of a remotely controlled vehicle equipped with the same camera used for detection of people [6]. This system is still limited by its processing capabilities, but these weaknesses could be eliminated using a mobile phone.

Authors have mentioned methods to decrease energy consumption during picture data transfer [7] and methods for camera movement detection that

processes the continuous optical information by use of segmentation of distinct features [8].

A new method to determine the size of an object in a photograph was also introduced for a security system purpose [9]. The method uses the recording from just one camera and has a goal to reduce processing costs caused by false detection by differentiating an animal from a person in respect to their size. An unnecessary detection often happens when using common infrared sensors. A camera hanging over and aimed at an object moving below is not appropriate for an autonomous system designed to explore environment though.

There are two methods used to determine the actual size of an object in a photograph. A commonly used one is putting an object of known dimensions next to the unknown object. On this principle are based for example measurements done by Rahman et al. [10]. It estimates the distance of a person by measuring the size of the person's eyes in pixels. Once the focal length and the distance of the recorded object are known then it is possible to calculate its actual size.

The other method is to determine the object size from two pictures with different points of view. When the focal length is known then the object size can be determined based on trigonometry. One of the solutions using a single optical sensor was published in a paper by Wahab et al. [11]. There they propose a monocular system utilizing the Hough transform. Their system is effective only for small distances though.

Calculations regarding monocular image sequence under a condition of known velocity were introduced in a previous work by Zhuang et al. [12]. This solution does not consider slip errors though.

A localization module based on an objective approach to optical processing has been introduced in often cited later published paper by Sahin et al. [13]. In it, both the size and the eccentricity of given objects was observed during movement of the robot. Elements defined in such a way do not always have to correlate to actual objects, but they must be in regions of enough size which can be separated from the background.

II. PRACTICAL IMPLEMENTATION OF ROBOTIC DEVICE

In the year 2011, Google has released Android version 3.1, part of which was also USB On-The-Go in the form of USB Accessory. USB OTG is a characteristic that allows the mobile phone to appear as a host for its environment and by that allows other USB devices to connect to it. Usage of USB OTG also allows switching the device into client mode. In reaction to this, the company FTDI Chip has released FT312D in April 2013. FT312D allows an Android device to communicate to peripheral hardware which is connected to the phone through USB 2.0 connector. The following communication from FT312D to the hardware is done through universal synchronous/asynchronous serial interface USART.

A. Communication Unit Proposal

The proposal of the control unit circuit board is split into two steps. The first one is the creation of the board with previously mentioned FT312D for communication with the mobile phone. The second one is the creation of universal peripheral for engine control. It was accounted for the need to have extra free space for potential additions to other functions of the robot, e.g. another rotor, a battery voltage sensor, sensor switches, etc. In the final proposal, both mentioned circuit boards are combined into one, but they are kept separated by use of ferrite filters and rectifiers. That is to stop possible transfer of interferences, caused by engines and external crystals, between the units.

Eagle version 6.3.0 for Windows was used for both proposals, in which there was created a custom library of components. That was done to allow adjustment of the component specifications and because not all used components were at disposal in the default set of libraries.

B. Control Algorithm for ATmega16 Proposal

AVR Studio 4 was used to create the proposal for a control program for ATmega16. The compiled project can be uploaded into the microprocessor through the SV1 connector. The prepared circuit board can be connected to PC and be debugged through preferred terminal. For our work, a Terminal 1.9b was used for debugging.

C. Android Application Proposal

An open source development platform Eclipse JUNO for Mobile Developers, extended by SDK Manager, was selected for programming the Android application. SDK contains tools for application development and operation system emulator. This Android emulator is unfortunately too slow and there is no guarantee of all its functions working correctly. For those reasons, it is practically unusable, and all the debugging was done on an actual device. The development and debugging have started on Samsung Galaxy S2 GT-I9100 (original ROM version 4.1.2). This phone is equipped with a camera of 8 Mpx (3964×2448 px).

The sharpness of photos is ensured by its autofocus function and appropriate lighting conditions can be established by use of its flash. Later down the development, it was found out that the Samsung-adjusted Android contains certain bugs. One of them being the inability to debug through Wi-Fi. For that reason, a new phone was selected for the continuation of our work. That being LG Nexus 5 for which Google guarantees the functionality of all its documented operating system features.

When Android Studio reached version 1.0, Google has announced that it is their official development tool and advised all developers to start using it. Later in the year 2015, a statement was issued, that the support of Eclipse has run its course and that is why the application for the robot's control has also been created in Android Studio.

D. Remote Server Proposal

One of the most important characteristics for selection of operating system was its general popularity. For that reason, and based on statistics of usage share, it was decided to use OS Windows. WinXP SP3 and Win7 SP1 were used for primary testing purposes. They were accessed through remote desktop by use of VNC Enterprise Edition for Windows on the default port 5900. Development environment Microsoft Visual Studio 2010 extended by JetBrains ReSharper 8.2 was used for the design proposal. A new project Visual C# Console Application was created in it based on NET Framework 4.

A robot with a mobile phone connected to a network is usually hidden behind NAT and so it is not possible to achieve direct communication from the server. That is with exception of the times when a static IP service is enabled on the phone number (this service is provided by cell phone providers for a fee). Due to the price of the service, a more conventional method of static server IP address was selected. It was then possible to proceed with debugging on any preferred router.

III. EXPERIMENTAL MEASURING OF DISTANCE

The goal was to confirm the hypothesis that it is possible to use a common optical sensor to measure distance and therefore to determine dimensions of the unknown object. The measuring is based on assumption that with increasing distance there is going to be increasing error in measurements – that is at least by an error caused by the resolution limitations. Another assumption is that with decreasing lighting conditions, it will be more difficult to apply appropriate detection method for accurate dimension estimation. An LG Nexus 5 mobile phone positioned above the working space was used for the measurements. The phone was connected through a designed application to a remote server to which it sends photos in regular time intervals. The server then processed and evaluated the photos according to its settings. Both the processing and the measurements are described in more details in the next paragraphs. The measurement was done in an enclosed hallway under stable lighting conditions as can be seen in Fig. 2.

In the picture there can be seen that due to the fluorescent lights there are many light reflection areas that make the detection more complicated. The phone was positioned in 1m step distances ranging from 2 to 15 meters. 50 images in 10 second interval were taken on each position. The whole measurement contains a constant error caused by using a tape measure limited to mm scale to determine the camera positions.

The error ε is the difference between the measured value x_m and the real value of the measured variable x . Therefore their relation is $\varepsilon = x_m - x$. One measurement is not enough to estimate the error, so multiple measurements should be done. That is to find the best estimate x_0 of the real value x , and to find out how exactly this estimate is. The accuracy of the estimate is based on an absolute error. Absolute error



Figure 2. 1st measurement test environment

in its essence characterizes the interval in which the real value can be expected within a certain probability margin.

The constant (systematic) errors afflict the measurement results in one specific way and with certain regularity. The systematic character of this error can be seen in the measured values being either constantly lower or constantly higher than the real values. These errors are based on the measurement method. They can occur due to an inappropriate method of measurements, varied quality of the measurement tools or a human error. The results can be corrected by the value of the error only if it is known. The problem of its identification and quantification remains to be solved by the measurements being done by a different method, by measurement tools of higher quality or by a different person. In our case the error is caused by using a tape measure of 1m length used to establish each measurement location. With each and every location, the error caused by the tape measure then adds up. It is also necessary to consider the methodical error occurring during averaging of the object edges that happens at the nearby point optimization step. In Fig. 3 is shown a layout of measuring an object of 100×10 cm dimensions where the highlighted angle $\varphi = 2/\sqrt{VVA}$.

The random errors are caused by outside interferences. They are hard to control and to them is attributed to the fact that the measurement values

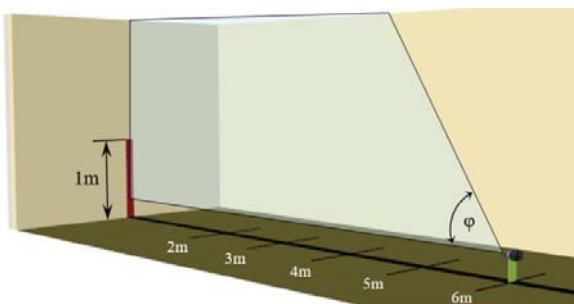


Figure 3. The layout of measurements at changing distances

somewhat differ even at the same conditions. Because of random errors, the measurement itself is a statistical process with a random variable. The probable value of the measured physical quantity and its error can then be determined by statistical methods.

A. Image Processing Algorithm

The image processing process is added to the design of the control server. There can be found a diagram of the whole process in Fig. 4. AForge.NET external libraries are used for some of the steps. That is an open-source C# framework meant for developers in the fields of computer vision, artificial intelligence and image processing. The following three libraries were used in version 2.2.5 (www.aforgenet.com) and placed into the project references:

- Aforge.dll
- AForge.Imaging.dll
- AForge.Math.dll

The photo first loads in (through a specifically defined protocol), its dimensions are determined and then colour correction filters are applied to it – as to remove any undesired shapes. The actual settings for the object detection and the following single step optimizations are described below.

1) Filter Application

For testing purposes, a board of 1000×100 mm dimensions in ruby red colour was chosen as the object of interest. The colour was chosen intentionally to be as it occurs in a standard working environment, so that the capabilities of the filter can be tested. Ruby red colour is defined by RAL 3003, i.e. RGB 134, 26, 34, or HSL 356, 68, 31, and it can be filtered by more than one possible filter. Filtering by colour in RGB colour space seems to be the simplest one. Pixels outside of the defined RGB ranges are filtered out and replaced by a selected colour. Another option is filtering by channel, where it is not the pixels that are filtered out but just their RGB values. Another mention-worthy option is Euclidean filtering, where

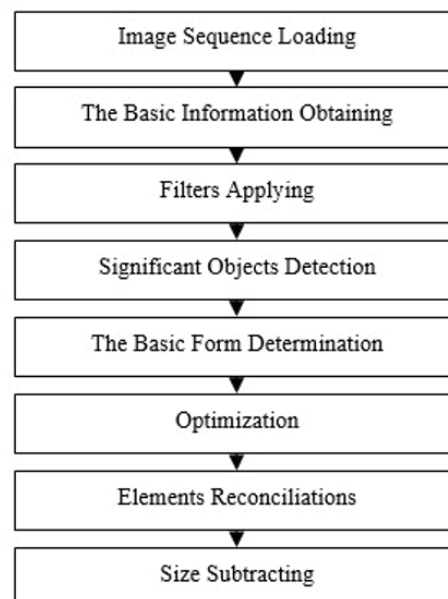


Figure 4. Image processing block sequence

the pixels meant for filtering are determined by RGB values of the central colour and a maximum range of the Euclidean distance. While working in HSL colour space, pixels outside certain value ranges can be filtered out again. These ranges define the shade, saturation, and brightness. The finally applicable colour models is the YCbCr model, which is defined by the brightness and two chrominance signals. Considering all of this, we selected the HSL filtering.

The HSL colour model consists of three parts:

- the colour hue H, i.e. the colour measured on a standard RGB colour wheel;
- -the colour saturation S (strength or pureness) represents the amount of grey compared to the hue and is measured in a percentage ranging from 0 to 100 %;
- the value of the colour lightness L.

Appropriate visualization geometry of the HSL model is a bicone where the hue represented as a colour wheel is conically prolonged into 3D space. The saturation (or in this case of conical representation – the chroma) is represented by the distance of the point from the centre of a circular segment. And the lightness is represented by its position between the cylinder top and base. The proposed filter settings do not have to be suitable for every environment, so it might be necessary to adjust them for different applications.

2) Interest Points Detection

Interest points in the image are searched for during the detection process. Locally significant interest points are such points that differ from their surrounding environment in the aspects of colour, intensity or texture. It is important to realize that an interesting point can be a blob, a ridge or a corner. From those, corner detectors are used the most often. The corner is a point that lies on the intersection of two or more edges. On the contrary, a blob is a whole area in the image that significantly differs from its surrounding environment but has constant characteristics inside the area. Due to the lack of edges in the non-structured environment, the most used method in visual odometry is then blob detection.

During the whole existence of this problematic, an extensive selection of detectors was created. The most known authors and their detectors are: Moravec [14], Förstner [15], Harris [16], Shi [17] a Rosten [18]. Into the line of blob detectors belongs for example Lowe [19].

Each detector has its advantages and disadvantages. A good one should have exact localization, good resolution capability and noise, compressions and blurring resistances. Another important aspect is also its photometric stability under nonlinear lightning conditions and its geometric stability under changing rotation, scale or perspective disturbances. Generally speaking, corner detectors have faster computing speeds compared to blob detectors, but they have lower resolution capabilities. Detected corners are located more easily in the image but compared to blobs they might not be possible to

locate in the next image. The choice of the detector should be considered carefully depending on the calculation possibilities, type of environment and frequency of incoming images. One detector is not suitable for everything; it all depends on the targeted navigation environment. For example, the SIFT detector automatically does not pay attention to corners that the metropolitan area is filled with. This detector was designed for object and area recognition and great results were achieved with it due to extensive descriptions of found interest points [20]. The area around each detected interest point is converted into mathematical descriptor during the description creation. Each of those descriptors can then be compared to descriptors from a previous image. The simplest description of an interest point is its appearance which is represented by the intensity of remote pixels. But in many cases the description of the interest point by use of local appearance is not the most suitable solution because the appearance changes depending on the capture orientation, scale and position.

A blob detector with a setting for the smallest possible detectable object taking up at least 1/35 of the image was selected for this research. This range alongside other options can be changed in the settings application window.

3) Determination of the Basic Shape

The result of the preceding detection is a set of n points $S = \{p_1, p_2, \dots, p_n\}$ in R^2 , where $p_i = [x_i, y_i]$. From that set we need to obtain the smallest convex set C , i.e. Euclidean space subset $\forall p_i \in C$, where for any line segment that connects any two points of the convex set, the points lying on that line segment also belong to the set itself. For this we use Graham Scan algorithm [21], which can be described in the following five steps:

First, we find a point defined by $p_1 = \min(y_i)$ inside a set C . In case of a match it is also true that $p_1 = \min(x_i)$. Secondly, for each point $p_i \in S$ we express polar coordinates with centre exactly at point p_1 , $\varphi = 0$.

Thirdly, we sort these points in ascending order depending on the angle φ , and we obtain $S = \{p_1 \exp(\varphi_1), p_2 \exp(\varphi_2), \dots, p_n \exp(\varphi_n)\}$ where $0 \leq \varphi_1 \leq \dots \leq \varphi_n \leq 2\pi$.

Fourthly, if $\varphi_i = \varphi_{i+1}$, then we can remove the point with the lowest amplitude from the final list.

Lastly, we test the three consecutive points $p_k \exp(\varphi_k)$, $p_{k+1} \exp(\varphi_{k+1})$, $p_{k+2} \exp(\varphi_{k+2})$ with $\varphi_k < \varphi_{k+1} < \varphi_{k+2}$, which can result in two possible results. Either the point lies to the right, $\alpha + \beta \geq \pi$, then we remove the point $p_{k+1} \exp(\varphi_{k+1})$ and return to the beginning of the last step to new testing. Or the point lies to the left $\alpha + \beta < \pi$, and then the point is left in the final set and we return to the beginning of the last step to new testing.

4) Optimization and shape determination

In Fig. 5 are shown all single steps required for object determination, while all view segments with exception of the first one, are colour inverted for clarification purposes.

There is the main optical input in the first segment. Segment B shows the output of the detector where the ridge marks are left out because otherwise, they would cause the whole object to be outlined in red due to their numbers. Information from this step to the last one is represented as a set of all ridges with defined coordinates. The object is left drawn out again for clarity purposes. Segment C shows the result after application of Graham scan. Segments D to F are outputs of single optimization processes. The first one is optimization of near corners. That is an algorithm where two points, which are in a specific range defined by their relation

$$t = \frac{\sqrt{\max(h) \cdot \max(w)}}{15}, \quad (1)$$

are merged into their average. The replacement is done during the consecutive iteration of the whole set, which is why the distance between the extreme points of the merged points set can be higher than the present range. An error is created due to this optimization, which can be considered a method based error.

The second step of optimization is alignment optimization. This algorithm checks all points and removes those that are at a certain distance from the previous and following point while on a line connecting those two. In other words, it goes through all the edges of the shape and checks the distance between each considered edge and a corner created by the two other edges. In the case that the distance is lower or equal than the pre-set value according to Eq. (1), then the point is removed so both of the edges get replaced by one.

The last step that is presented on the segment F, is the creation of arithmetic mean between the resulting points. That again causes a systemic error which is part of the overall aggregated method error.

B. Measurement Errors

An analysis of errors that can occur during the measurements is added to the measurement results themselves. The following text will pay more attention in detail to establish the inaccuracies of measurement. The conclusions relating to the random error theory are in theory true only for infinite amount measurements and are in practice applicable only for the significantly large amount of measurements. Due to the fact, that performing many measurements (approx. 1000) is quite time intensive, the sum of all errors ε_k can be considered as null. Therefore, during the n -th measurement, it is true that:

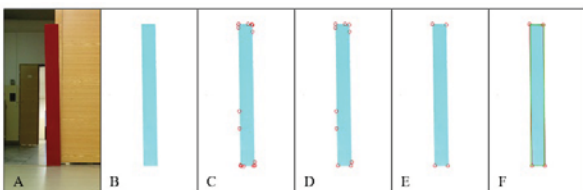


Figure 5. Optical information processing steps

$$x_0 \approx \bar{x} = \frac{1}{n} \sum_{k=1}^n x_k. \quad (2)$$

The set of all measured values n is a random selection from the set of all possible values of the measured quantity and is a Gaussian distribution. If the set of n measurements would be done multiple times, then a different value of an arithmetic mean would be obtained for each of these sets. That is why the arithmetic means according to Eq. (2) will be from now on denoted as a sample mean.

To evaluate the precision was used standard deviation, where the same problem as with calculation of the arithmetic mean has occurred. Therefore, we will also denote the standard deviation as a sample standard deviation s and it is true that:

$$s = \sqrt{\frac{1}{n-1} \sum_{k=1}^n \Delta_k^2} = \sqrt{\frac{1}{n-1} \sum_{k=1}^n (x_k - \bar{x})^2}. \quad (3)$$

That means that s is the sample standard deviation of one measurement. But for evaluation, it is more important to know how much will be influenced the sample mean of the measured values, see Eq. (2). The sample standard deviation of the sample mean is

$$s_x = \frac{s}{\sqrt{n}} = \sqrt{\frac{\sum_{k=1}^n \Delta_k^2}{n(n-1)}} \quad (4)$$

From the dependency of ratio s_x/s , as a function of measurement amount n , it is obvious that high measurement quantities are inefficient. For realized $n = 50$, it is true that, $s_x = s / \sqrt{50} \approx 0.14 \cdot s$. Tab. I. shows the calculations made from all the measurement results. In the first column are approximate distance values d in meters.

Then there is the arithmetic mean in millimetres, the sum of squares of the standard deviation, sample standard deviation calculated from the Eq. (4) and resulting size estimate in millimetre.

TABLE I. FIRST MEASUREMENT RESULTS

d	\bar{x}	$\sum \Delta_k^2$	s_x	$3s_x$	x
2	990	239	15	46	990 ± 15
3	988	352	19	56	988 ± 19
4	988	508	22	67	988 ± 22
5	983	544	23	69	983 ± 23
6	983	798	28	84	983 ± 28
7	977	1100	33	98	977 ± 33
8	969	1983	44	132	969 ± 44
9	970	2469	49	148	970 ± 49
10	966	2280	47	142	966 ± 47
11	961	3893	62	185	961 ± 62
12	958	3194	56	168	958 ± 56
13	956	4167	64	192	956 ± 64
14	958	6124	77	232	958 ± 77
15	949	6903	82	247	949 ± 82

These graph results, shown in Fig. 6, confirm the hypothesis, that with increasing distance, the detection error increases as well.

It can also be seen, that the result of measurement on all positions is smaller than the real value of 1000 mm. That is due to the systematic error of the used detector's corner detection method. The dotted line signifies the linear trend. Quality of the selected regression model is given by the coefficient of determination R^2 , which is sometimes interpreted as a match of the model with the data itself. The closer the coefficient is to value of 1 m the less are the points spread around the regression curve. Value of $R^2 > 0.95$ is often considered as enough for accepting the chosen model in the technical literature. The R^2 for the measurements presented in Fig. 6 is obtained by the linear regression function of MS EXCEL and is of value $R^2 = 0.9662$. The increase of the standard deviation with distance is equivalent to $5.1319d + 5.8681$, where d denotes the distance from the measured object. The linear regression model can still be considered as suitable with this result.

IV. CONCLUSION

The measurement results prove that a basic optical sensor of 1/3.2" dimensions, that can be found on the used LG Nexus 5 phone, can be successfully used for object size measurements. The object size determination error is a combination of camera resolution error and a random error. In comparison to other common phones, it can be said that the measures made by the sensor belong to the cheaper category of phones. Therefore, as newer phones equipped with higher quality sensors will be used, the overall precision of the estimation will improve as well.

To remove the error caused by the detector, it is necessary to apply different decision-making algorithm. The solution of this particular task would be possible by a simple selection of extreme points from the whole set and consecutive estimation based on their position. But it is necessary to keep the current application settings so that this paper is applicable even to more complex objects. The first and second step of optimization, near corner optimization and alignment optimization respectively, were interchanged for the purposes of this paper. Even though these solutions seemed to result in a more accurate estimate of extreme points, it also showed more frequent errors in recognition of the final shape. That is why in the end it was not used. This particular

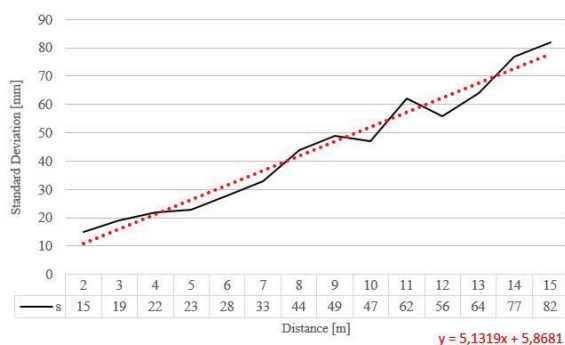


Figure 6. 2nd processing results

setting option depends on the utilization type of the device and so it is possible that the discarded proposal will be more suitable for other applications.

ACKNOWLEDGMENT

The work presented in this article has been supported by the Czech Republic Ministry of Defence – University of Defence development program “Advanced Automated Command and Control System II - PASVR II”

REFERENCES

- [1] P. PUGET, and T. SKORDAS, “An optimal solution for mobile camera calibration”, in International Conference on Robotics and Automation. Cincinnati: IEEE, 1990, pp. 34-39. ISBN: 0-8186-9061-5, DOI: 10.1109/ROBOT.1990.125942.
- [2] Z. ZHENGYOU, and O.D., FAUGERAS, “Calibration of a mobile robot with application to visual navigation”, in Workshop of Visual Motion. Irvine: IEEE, 1989, pp. 306-313. ISBN: 0-8186-1903-1, DOI: 10.1109/WVM.1989.47123.
- [3] A. HOU, X. CUI, Y., GENG, W. YUAN, and J. HOU, “Measurement of safe driving distance based on stereo vision”, in International Conference on Image and Graphics. Hefei: IEEE, 2011, pp. 902-907. ISBN: 978-0-7695-4541-7. DOI: 10.1109/ICIG.2011.27.
- [4] Y.M. MUSTAFAH, R. NOOR, H. HASBI, and A.W. AZMA, “Stereo vision images processing for real-time object distance and size measurements”, in International Conference Computer and Communication Engineering. Kuala Lumpur: IEEE, 2012, pp. 659-663. ISBN: 978-1-4673-0478-8. DOI: 10.1109/ICCCE.2012.6271270.
- [5] W. YOU LU, M. CASARES, and S. VELIPASARLAR, “Cooperative object tracking and event detection with wireless smart cameras”, in International Conference on Advanced Video and Signal Based Surveillance. Genova: IEEE, 2009, pp. 394-399. ISBN: 978-1-4244-4755-8. DOI: 10.1109/AVSS.2009.77.
- [6] H. LI, W. YOU LU, S. VELIPASARLAR, and M.C. GURSOY, “Human detection using mobile embedded smart cameras”, in International Conference on Distributed Smart Cameras. Ghent: IEEE, 2011, pp. 1-6. ISBN: 978-1-4577-1707-9. DOI: 10.1109/ICDSC.2011.6042924.
- [7] Z. YU, Y. CHUANG, S. VELIPASARLAR, and M.C. GURSOY, “Energy efficient image transmission using wireless embedded smart cameras”, in International Conference on Advanced Video and Signal Based Surveillance. Seoul: IEEE, 2014, pp. 62-67. ISBN: 978-1-4799-4871-0. DOI: 10.1109/AVSS.2014.6918645.
- [8] A. MAHABALAGIRI, and S. VELIPASARLAR, “Camera motion detection for mobile smart cameras using segmented edge-based optical flow”, in International Conference on Advanced Video and Signal Based Surveillance. Seoul: IEEE, 2014, pp. 271-276. ISBN: 978-1-4799-4871-0. DOI: 10.1109/AVSS.2014.6918645.
- [9] P. MOOKYUNG, M. NAMSU, R. SANGRIM, K. JEONGPYO, L. YONGJIN, and M. WANGKIN, “A pixel-weighting method for discriminating objects of different sizes” in The 3rd Canadian Conference Computer and Robot Vision. Quebec: IEEE, 2006, pp. 36-36. ISBN: 0-7695-2542-3. DOI: 10.1109/CRV.2006.6.
- [10] K.A. RAHMAN, M.S. HOSSAIN, M.A. BHUIYAN, Z. TAO, M. HASANUZZAMAN, and H. UENO, “Person to camera distance measurement based on eye-distance”, in International Conference on Multimedia and Ubiquitous Engineer. Qingda: IEEE, 2009, pp. 137-141. ISBN: 978-0-7695-3658-3. DOI: 10.1109/MUE.2009.34.
- [11] M.N.A. WAHAB, N. SIVADEV, and K. SUNDARAJ, “Target distance estimation using monocular vision system for mobile robot”, in International Conference on Open Systems. Langkawi: IEEE, 2011, pp. 11-15. ISBN: 978-1-61284-931-7. DOI: 10.1109/ICOS.2011.6079296.
- [12] H. ZHUANG, R. SUDHAKAR, and J. SHIEH, “Depth estimation from a sequence of monocular images with known camera motion”, in Robotics and Autonomous Systems.

- Florida: Atlantic University, vol. 13, issue 2, pp. 87-95, 1994. DOI: 10.1016/0921-8890(94)90051-5.
- [13] E. SAHIN, and P. GAUDIANO, "Development of a visual object localization module for mobile robots", in Third European Workshop on Advanced Mobile Robots. Zurich: IEEE, 1999, pp. 65-72. ISBN: 0-7803-5672-1. DOI: 10.1109/EURBOT.1999.827623.
- [14] H. MORAVEC, "Obstacle avoidance and navigation in the real world by a seeing robot rover". Robotics Institute, Carnegie Mellon University. Doctoral dissertation. Stanford University, 1980. Supervisor: Rod Brooks.
- [15] W. FÖRSTNER, "A feature based correspondence algorithm for image matching", *Int. Arch. Photogrammetry and remote sensing*. Vol. 26, issue 3, pp. 150-166. Stuttgart University, 1986.
- [16] Ch, HARRIS, and M. STEPHENS, "A combined corner and edge detector", In *Proc. of Fourth Alvey Vision Conference*. United Kingdom: 1988.
- [17] J. SHI, and C. TOMASI, "Good features to track", in *Computer Society Conference on Computer Vision and Pattern Recognition*. Seattle: IEEE, 1994, pp. 593-600. ISSN: 1063-6919. DOI: 10.1109/CVPR.1994.323794.
- [18] E. ROSTEN, and T. DRUMMOND, "Machine learning for high-speed corner detection", in *9th European Conference on Computer Vision*. Graz: Springer-Verlag, 2006, pp. 430-443. ISBN: 978-3-540-33832-1.
- [19] D.G. LOWE, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*. 2004, vol. 60, issue 2, pp. 91-110. ISSN: 0920-5691.
- [20] F. FRAUNDORFE, and D. SCATRAMUZZA, "Visual Odometry; Part II: Matching, Robustness, Optimization, and Applications", *Robotics & Automation Magazine*. 2012, vol.19, no.2, pp.78-90. DOI: 10.1109 /MRA. 2012. 2182810.
- [21] R.L. GRAHAM, "An Efficient Algorithm for Determining the Convex Hull of a Finite Planar Set", *Information Processing Letters* 1, 132-133, 1972.