Transactions of the VŠB – Technical University of Ostrava, Mechanical Series
No. 2, 2016, vol. LXII
article No. 2020

Ján VACHÁLEK[*], Oliver ROVNÝ[**], Rudolf PALENČÁR[***], Stanislav ĎURIŠ[****]

MEASURING DISTANCES USING THE 3D OPTICAL DEVICE FOR THE NEEDS OF MOBILE ROBOTICS

MERANIE VZDIALENOSTÍ ZA POMOCI 3D OPTICKÉHO ZARIADENIA PRE POTREBY MOBILNEJ ROBOTIKY

Abstract

This paper presents the collision-free operation of a robotic arm mounted on a mobile robotic unit, using the MS Windows Kinect 3D optical system. The 3D optical system is used for measuring distances and aiding the collision-free manipulation of objects by means of the robotic arm installed on the mobile robotic system (MRS). Attaching of the optical system directly to the robotic arm, essentially, ensures the autonomy of the overall system. The MRS is capable of recognizing the pre-defined objects in the three-dimensional space as well as automatically approaching and manipulating them using the robotic arm. The aim of this article is to present the algorithms used in the MRS to guarantee the collision-free operation of the system. The novelty of our approach lies in the a priori collision avoidance strategy instead of solving collision states as they occur. In addition, the article presents the problem of the localization of the objects in the robot's surrounding. For this purpose, the balls randomly placed on the floor were localized by the MRS, estimating their real coordinates. Instead of using the standard RGB color space, we propose to utilize the HSV color space in order to enhance the process of objects recognition since the HSV color space provides the more consistent results in the case of varying intensity of the ambient light.

Abstrakt

Príspevok sa zaoberá bezkolíznou manipuláciou ramena mobilného robota s využitím 3D optického kamerového systému MS Windows Kinect. 3D optický systém slúži pre potreby merania vzdialeností a následnou bezkolíznou manipuláciou s nimi na použitom mobilnom robotickom systéme (MRS), zaujímave je jeho konštrukčné umiestnenie priamo na ramene robota pre zachovanie autómnosti celého systemu. MRS je technologický mobilný robotický demonštrátor, kompletne vyvinutý na našom pracovisku a jeho úlohou je oboznámiť študentov a odbornú verejnosť s možnosťami súčasnej mobilnej robotiky. MRS dokáže rozpoznávať v priestore vopred zadefinované objekty priblížiť sa autonómne k nim a za pomoci robotického ramena bezkolízie aj s nimi

[*] doc. Ing., PhD., Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Námestie Slobody 1, Bratislava, tel. (+421) 2 5249 7193, e-mail jan.vachalek@stuba.sk

[**] Ing., Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Námestie Slobody 1, Bratislava, tel. (+421) 2 5249 7193, e-mail oliver.rovny@stuba.sk

[***] Prof. Ing., PhD. Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Námestie Slobody 1, Bratislava, tel. (+421) 2 5249 7193, e-mail rudolf.palencar@stuba.sk

[****] doc. Ing., PhD., Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Námestie Slobody 1, Bratislava, tel. (+421) 2 5249 7193, e-mail stanislav.duris@stuba.sk

manipulovať. V článku budeme riešiť ako boli zadefinované a aplikované konkrétne algoritmy pre garanciu bezkolíznej manipulácie MRS s objektmi, kde na rozdiel od iných prístupov našou snahou je prvotne sa kolízii vyhnuť a nie riešiť stavy ak už kolízia nastane. Riešená bola aj lokalizačná úloha objektov v priestore. Pre potreby riešenia lokalizačných úloh budú v priestore náhodne rozmiestnenené loptičky a úlohou MRS ich bude lokalizovať a stanoviť ich reálne súradnice pre potreby navigačného subsystému MRS. Pre stanovenie pozície nepoužijeme štandardný RGB farebný model, ale HSV farebný model, ktorý dáva lepšie výsledky hlavne pri zmenách intenzity osvetlenia okolného priestoru MRS. Po stanovení reálnych súradníc za pomoci nami zvoleného 3D optického systému, prevedieme záverom štatistický výpočet pre zistenie presnosti nami riešenej lokalizačnej úlohy a výsledných neurčitostí merania v jednotlivých súradných osiach.

Keywords

Measuring distance, objects localization, 3D optical camera system, MS Windows Kinect, RGB colorspace, HSV colorspace

1 INTRODUCTION

Owing to the development of new computing hardware and new generation of sensors, the modern robotic systems may handle the tasks of real-time navigation and localization in the different fields of application. Nowadays, various autonomous mobile robotic systems such as autonomous industrial robotic logistic systems, transportation robots, flying drones or other autonomous robotic systems come into reality.

The aim of this article is to present the algorithms used in our MRS, which guarantee the collision-free manipulation of the objects by means of the robotic arm of the MRS. The novelty of our approach lies in the a priori collision avoidance strategy instead of solving collision states as they occur. In addition, the article presents the problem of localization of the objects in the robot's ambient space. For this purpose, position of the balls randomly placed on the floor has been identified by the MRS. The position coordinates of the balls were estimated by means of algorithms presented in this article. It is worth notation that instead of using the standard RGB color space, we proposed to utilize the HSV color space in order to enhance the process of objects recognition. The reason is that the HSV color space provides the more consistent results in the case of varying intensity of the ambient light.

2 DESCRIPTION OF THE MOBILE ROBOTIC SYSTEM

The MRS (Fig. 1), which is equipped with an omni-directional (Meccanum) drivetrain, includes a powerful computing module used to control the MRS [1], [2], [3], [4]. The dimensions of the MRS are fixed to be approximately 1000 mm x 800 mm x 400 mm (L x W x H) and the total mass including the robotic arm is 90 kilograms. The prevalent material of the chassis are 6 mm and 2 mm thick aluminium alloy sheets.

As already mentioned, the chassis of the MRS utilizes the omni-directional Meccanum wheels. Such a drivetrain enables the MRS to move with three degrees of freedom so that the two translations and one rotation of the platform can be controlled independently. This principle is referred to as holonomic robotic system. As the said omni-directional drivetrain is only suitable for flat, hard and smooth floor, the Meccanum wheels can be replaced with the traditional wheels and the omni-directional control of the platform can be changed to differential control so that the MRP can be adapted to rough terrain. Each wheel has its own electrical servomotor with planetary gearbox. Such a driving system allows the robot to achieve a maximum speed of 2.6 km/h with a maximum payload of 80 kg. The power subsystem of the robot consists of the eight Lithium Yttrium LiFePO4 batteries, ensuring a sufficient power supply up to 10 hours of operation. The computing subsystem of the MRS uses an x86 architecture computer, providing computing power to solve advanced algorithms in

real-time. The primary task of the built-in computer is to solve the algorithms of navigation and object localization. Its secondary task is to secure the collision-free operation of the robotic arm.

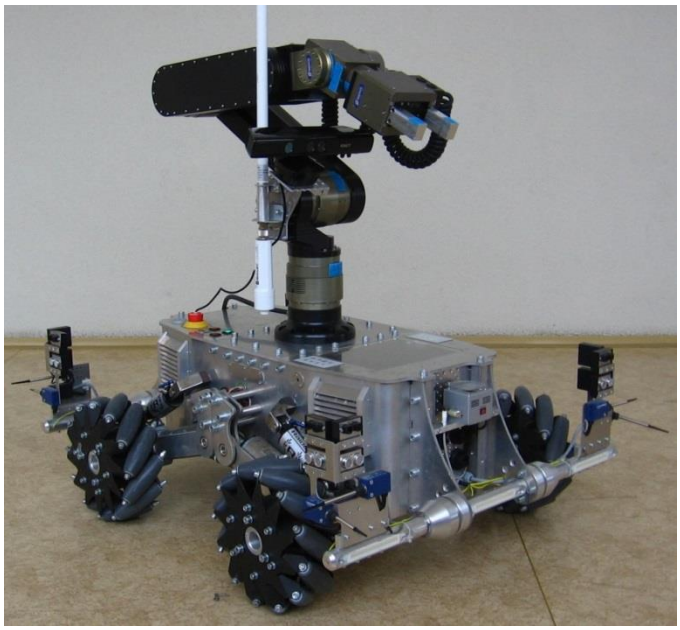

Fig. 1. Autonomous mobile robotic system

The communication subsystem of the robot features traditional cabled Ethernet connections or redundant wireless systems. Redundancy is ensured by the mutual combination of WiFi signals in the 2.4 GHz and 5 GHz bands [5], [6].

Concerning the sensory subsystem, the device is equipped with ultrasound, optical and tactile sensors. These are placed in pairs at the vertices of the chassis. The sensors enable the robot to detect the obstacles, and to provide additional information for navigation. In addition to these, there are more advanced sensors used for spatial localization and navigation. These are mainly the so-called StarGazer device and the 3D optical system, Microsoft Windows Kinect.

The robotic arm has been custom-designed by the company SCHUNK. Mechatronic modules inside the arm are designed and built based on our specifications. The dimensions of the arm itself are designed so that the arm can reach down to the baseline (ground) around the chassis. The arm consists of three parallel units with six degrees of freedom. The first three degrees of freedom are given by the arm, the rest by the end effector. The main parameters of the arm are:

- the end effector is a mechanical device with two fingers,
- the maximal load of the arm in the fully extended state is 1 kg,
- the manipulation radius of the arm is 1.2m,
- the weight is 14.5 kg,
- powered by DC 24 V,
- and the internal communication uses the CAN system.

After this brief introduction to the hardware particulars of the MRS, now we will move to the theoretical definition of our proposed collision-free manipulation system.

3 DESCRIPTION OF THE MS WINDOWS KINECT 3D OPTICAL DEVICE

Microsoft Kinect is probably the most affordable optical device, often used for robotic applications [7], [8], [9], [10], [11], [12], [13], [14], [15]. Its advantage in comparison with professional counterparts lies in relative simplicity and an obvious economic advantage. As this system is primarily meant for the videogame industry, several modifications were necessary for its use on the MRS, including its integration to different software environments, especially Matlab.

The Kinect module is mounted on the first joint of the arm of the MRS, as it is shown in Fig. 1. This solution is more complex in terms of mechanical construction; however, considering the autonomous use of the MRS, it is more desirable. The arm can turn around its vertical axis, enabling to sense a 360° panoramic view around the robot. The main components of this device are a proprietary single use DSP chip, a CCD camera and a depth-sensing camera. The depth-sensing camera is created by an infrared camera and an infrared emitter module. The Kinect device uses the USB 2.0 interface to communicate with the computer. The first optical camera allows a user set resolution of either 640 x 480 pixels or 1280 x 960 pixels. In addition, it allows the user to switch between different colorspaces. The depth measurement implemented by the second camera is possible in two working regimes: the first is used by the X-box gaming console, having an approximate range of 0.8 to 4 m. The second mode of operation is the so-called "near" regime, having a range from 0.4 m up to 3 m. This is available only on a special modified version of the Kinect, named Windows Kinect (used for the MS Windows operating system), which is also considered in this work. The resulting distance information is the perpendicular position of the measured object from the face of the Kinect sensor. The output data are matrices measuring 640 x 480 for both cameras, which are then fed and evaluated inside the Matlab environment (Fig. 2).

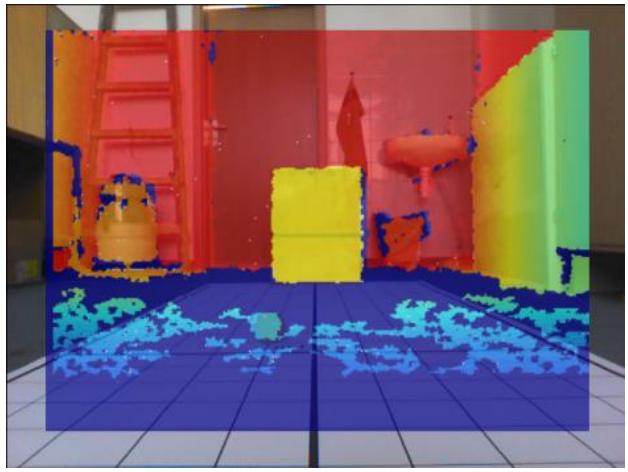

Fig. 2. Output from Kinect, overlaid with a pseudo color representation of the measured depth field in Matlab

It is then necessary to extract the object of interest from these matrices, and to calculate its position coordinates in relation to the face of the Kinect sensor. For this reason, we have chosen the location of the coordinate system as it is shown in Fig. 3. It is actually a pair of coordinate systems: one local and coincident with the MRS, and one local, given by the Kinect sensor. This choice can be rationalized, since the Kinect sensor will rotate along with the arm, resulting in simple transformation procedures to get its global coordinates.

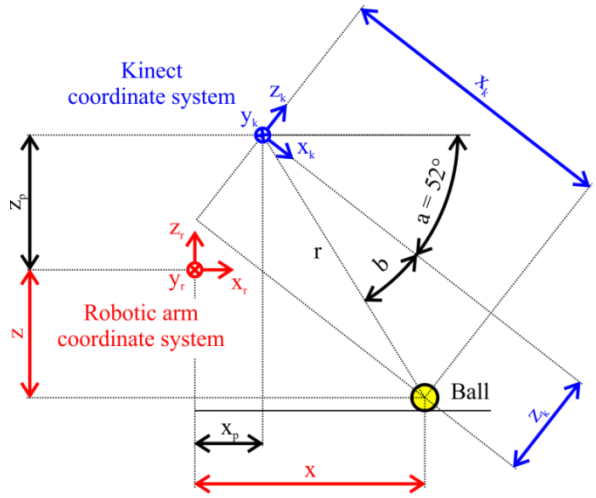

Fig. 3. Definition of the position of the Kinect sensor in local and global coordinates

Fig. 4.

4 MEASURING DISTANCES WITH MS KINECT CCD CAMERAS

For the exact evaluation of coordinates, the internal parameters of the CCD camera must be calibrated. Fortunately, a ready-made tool called the Camera Calibrator Toolbox is available in the Matlab environment. This toolbox was designed to compute focal distances and it requires a complex calibration procedure [16], [17], [18], [19], [20], [21]. The approach in the heart of the procedure is the use of a specific calibration screen with a certain shape and dimensions. This is black and white screen, which must be sensed from different pre-determined angles. The number of images taken with the camera equal to 36. One field of the calibration screen has the dimensions of 30 mm by 30 mm. After using the resulting matrices in the mentioned toolbox, the focal distances f along with the center of the frame c will be computed. Using this procedure, we have calibrated the camera according the data shown in Tab. I.

TABLE I. OPTICAL PARAMETERS OF THE FIRST CAMERA

Focal points	
f_y	517.3
f_z	517.4
Center	
c_y	318.5
c_z	239.4

The next step is to calibrate the position of the Kinect device in relation to the global coordinate system of the MRS, which was placed into its hypothetic center. For this, a program was created to transform different coordinate systems inside Matlab. Amongst others, this program automatically visualizes a crosshair into the footage, with its center given at c_y and c_z. This crosshair is necessary for the correct configuration of the coordinate system, since it helps to align the field of view on the calibration screens (Fig. 4). This procedure ensures that the Kinect system is turned in the correct angle in relation to the global coordinate system. The object localized in this work is a small yellow sponge ball with a radius of $r_l = 34mm$. The localization procedure is realized by the color of the object. The color is set using the average color of the ball, based on several frames from the

primary CCD camera. The frames containing the ball were taken at several places with different lightning conditions. The RGB values the robot is looking for are shown in Tab. II. These values need to be re-configured, in case the color of the ball changes or there are severe deviations from the nominal lightning conditions.

TABLE II. COLOR TO BE FOUND

Channel	Value
r	200
g	180
b	45

First, the ball is localized in the RGB colorspace. We will use a method to look for the ball in the individual RGB channel. The coordinates of the ball found by the primary CCD camera will be noted as "s" a "r". The symbol s (column) will mark the points along the y-axis, while symbol r (row) will denote the points found along the z-axis. The coordinates of such points s and r are to be found, which are close to the nominal color in the individual channel with some added tolerance. In this way, each channel has its own tolerance value, with the nominal settings shown in Tab. IV. Points are to be found in the RGB colorspace so that they fulfill the following conditions:

$$\mathbf{R} < r + tol \;\&\; \mathbf{R} > r - tol \tag{1}$$

$$\mathbf{G} < g + tol \;\&\; \mathbf{G} > g - tol \tag{2}$$

$$\mathbf{B} < b + tol \;\&\; \mathbf{B} > b - tol \tag{3}$$

where matrices **R**, **G**, **B** represent the matrices of each channel within the colorspace, while "r", "g", "b" are the nominal values for the color of the ball within the channel and "tol" is the pre-determined tolerance in which the color must be contained. The tolerance values used in this work are shown in this work are shown in Tab. IV.

TABLE IV. RGB COLOR TOLERANCES

Channel	Tolerance
r	40
g	40
b	20

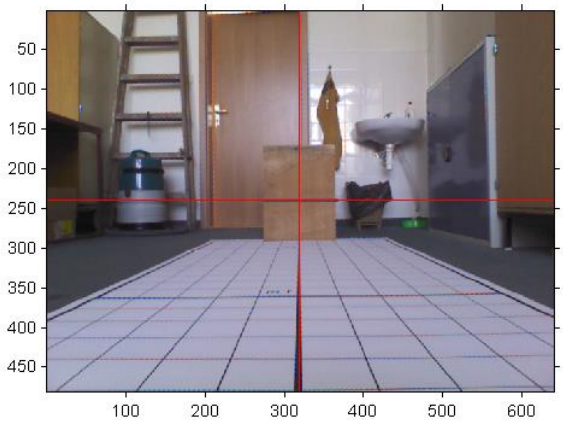

Fig. 4. Virtual crosshair for the correct configuration of the coordinate system

In order to remove the problems that arise from the changing light intensity in the room, causing imprecision of object detection when using the RGB colorspace, we have also used the so-called HSV (Hue, Saturation, Value) colorspace. In contrary to the color-mixing scheme of RGB, our primary choice for colorspace, the HSV colorspace models the colors in parameters, which are closer to our natural way of seeing light. This actually results in detection that is more precise and eliminates the problems associated with variable light intensity in the room. The HSV color space consists of the hue of the color, characterizing its wavelength, which is traditionally visualized in a circle using values ranging from 0° to 360°. The saturation parameter characterizes the ratio of the color component in relation to grayscale, where 100% saturation means that the color is fully saturated, while 0% means that only a grey color is left without any saturation. The final, third, parameter provides information about the brightness value ranging from 0% to 100%. The difference between these visualized colorspaces is given in Fig. 5.

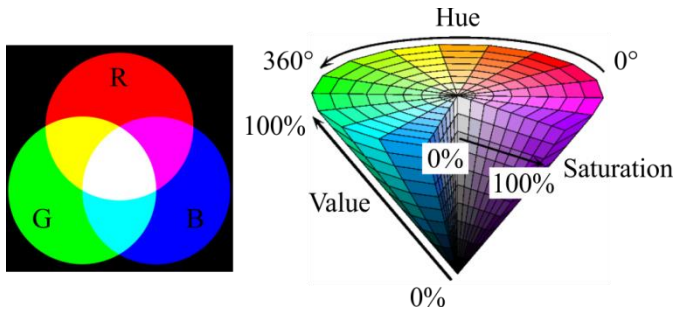

Fig. 5. Principle of the RGB (left) and HSV (right) color models

The advantage of the HSV model in comparison with the RGB is that the HSV is able to capture a greater detail in moderate lightning conditions, which actually suits our needs.

The coordinates of the points are then store into two vectors. Vector **r** contains all values of r, while vector **s** contains all values of s, having equal lengths. We compute the average value for each of these vectors according to equations (4) and (5), obtaining the final position y_{rgb} and z_{rgb} of the ball with the given color using the primary CCD camera. These coordinates represent the position of the ball in the vision field according to

$$y_{rgb} = \frac{\sum_{i=1}^{n} s_i}{n}, \tag{4}$$

$$z_{rgb} = \frac{\sum_{i=1}^{n} r_i}{n}. \tag{5}$$

The difference between the less and more intense lightning of the ball causes the color of the ball to vary so much in the RGB colorspace, that it is actually impossible to locate it using this procedure. This is the reason why we augmented the RGB search method with the HSV colorspace. The image is converted into the HSV from the original RGB colorspace (Fig. 6), ensuring a better localization of the ball in bad ambient lightning conditions.

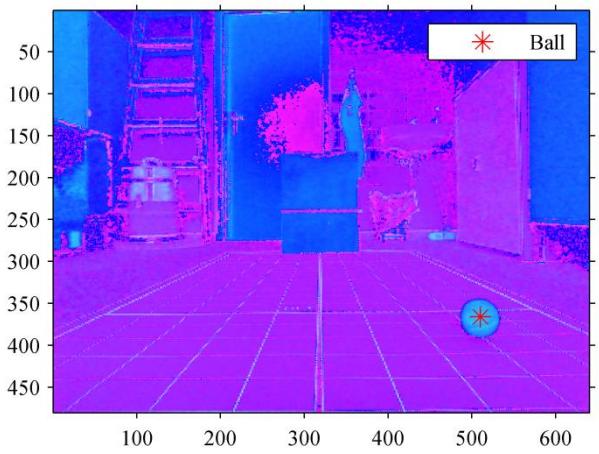

Fig. 6. Localization in the HSV colorspace

The way in which the ball is localized in the HSV colorspace is the same as in the case of RGB. All points are found, which fulfill the certain condition with the given tolerance. Then we compute the average value from all these points along the y and z-axes, which fulfill similar criteria to the case of RGB. The coordinates of the ball in the HSV colorspace are denoted by y_{hsv} and z_{hsv} with the reference values listed in Tab. V.

TABLE V. TOLERANCE VALUES FOR THE HSV COLORSPACE

Channel	Tolerance
h	0.03
s	0.4
v	80

If we visualize the points found in the HSV colorspace, we unfortunately see several misdetected locations. These locations are either individual points scattered across the screens or are groups of coordinates as it is illustrated in Fig. 7.

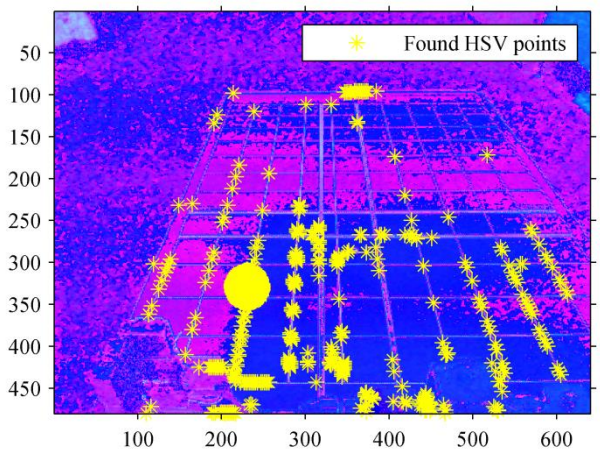

Fig. 7. Coordinates found in the HSV colorspace

A customized filter was created to remove the misdetected coordinates from the HSV analysis from the primary camera. For each point, a control algorithm is run in eight directions. If the filter detects that some of the eight search directions is not marked in the image, the given point is removed from the detected set. Fig. 8. shows all detected coordinates (yellow), the coordinates that remain after the above described ad-hoc filtering procedure (green), leaving the center of the ball marked by the coordinates y_c and z_c.

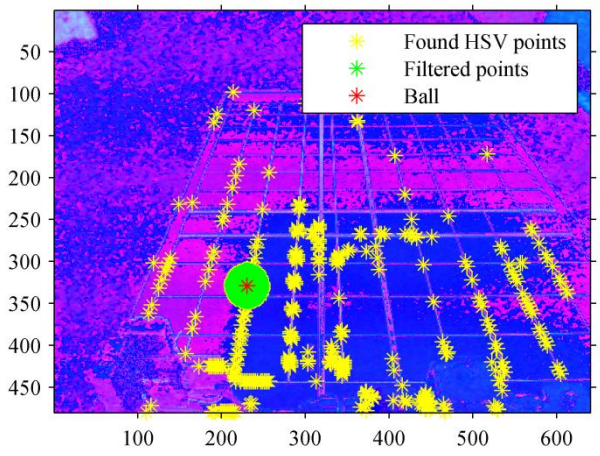

Fig. 8. Filtered coordinates and the center of the ball

To find the coordinates of the ball for the needs of its localization, the logically simplest solution is to transform the two-dimensional coordinates into the secondary infrared imaged, in order to gain depth information. This way, the x coordinate of the ball can be obtained. Unfortunately, this direct solution is not feasible as the field of view of the two cameras slightly differs. This means, that the coordinates of obtained using the primary CCD camera do not actually correspond with the coordinates of the secondary camera used to get the needed depth information. This is the reason why it is necessary to calibrate the secondary camera, then to perform a correction between the primary and secondary images.

A custom analytical correction method has been created to solve the issue with this transformation. We have computed the horizontal and vertical angles of the ball based on the detected image from the primary camera, and re-computed this angle to fit the image obtained by the

secondary infrared camera. This way, it is possible to transform the coordinates of the ball from the first to the second camera within a certain tolerance band. It is necessary to know the viewing angle of both cameras to perform this transformation. The viewing angle of the secondary camera is actually given by the manufacturer, so that we have a horizontal angle marked by the index „h“ and an angle α_h ; and an angle β_h . The unknown horizontal angle of the first α_{rgb} with the index “rgb” can be computed by:

$$\alpha_{rgb} = \arctan(320/f_y).2\,, \tag{6}$$

while the vertical angle β_{rgb} of the primary camera can be computed from the equation

$$\beta_{rgb} = \arctan(240/f_z).2\,. \tag{7}$$

The coordinate transformation is carried out in four quadrants, as shown in Fig. 9. The first quadrant uses the transformation given by

$$y_h = 320 - ((\alpha_c / 2)/320).(320 - y_c)).(320/(\alpha_h / 2) \tag{8}$$

$$z_h = 240 - ((\beta_c / 2)/240).(240 - z_c)).(240/(\beta_h / 2) \tag{9}$$

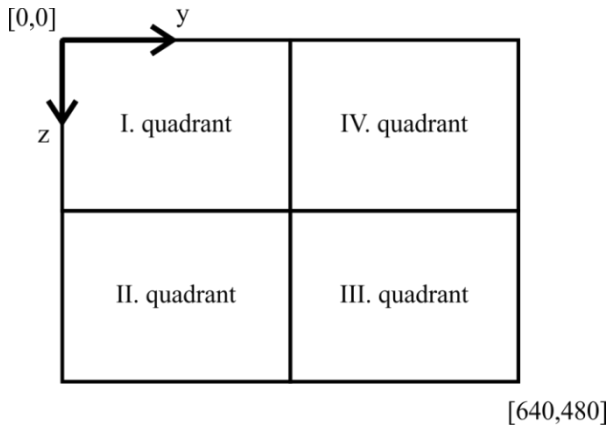

Fig. 9. Dividing the image into four quadrants

The computation for the second quadrant is:

$$y_h = 320 - ((\alpha_c / 2)/320).(320 - y_c)).(320/(\alpha_h / 2) \tag{10}$$

$$z_h = ((\beta_c / 2)/240).(z_c - 240)).(240/(\beta_h / 2) + 240 \tag{11}$$

for the third quadrant it is:

$$y_h = ((\alpha_c / 2)/320).(y_c - 320)).(320/(\alpha_h / 2) + 320 \tag{12}$$

$$z_h = ((\beta_c / 2)/240).(z_c - 240)).(240/(\beta_h / 2) + 240 \tag{13}$$

and finally for the fourth quadrant is:

$$y_h = ((\alpha_c / 2)/320).(y_c - 320)).(320/(\alpha_h / 2) + 320 \tag{14}$$

$$z_h = 240 - ((\beta_c / 2) / 240).(240 - z_c)).(240 / (\beta_h / 2) \quad (15)$$

where y_h and z_h are the coordinates of the object in the depth image obtained from the footage of the secondary infrared imaging device. As it has been already discussed, the coordinates of the Kinect device do not correspond to the global coordinates of the mobile robotic system, which has its origin in the hypothetical center of the MRS (see Fig. 7). This is why it is necessary to perform a transformation procedure, where the coordinates based on the focal point of the camera are converted into the real coordinates of the object given by x_k, y_k a z_k. The procedure is solved as follows: coordinate x is the value directly read by the secondary infrared imaging device, as this value is actually identical for both the global and the local coordinate systems, that is:

$$x_k = x + r_l \quad (16)$$

$$y_k = (y_c - c_y).x / f_y \quad (17)$$

$$z_k = (c_z - z_c).x / f_z \quad (18)$$

where:

r_l - is the radius of the ball,

y_c , z_c - are the y and z coordinates of the camera in mm,

c_y , c_z – are the coordinates of the image centers based on the image resolution of the given camera,

f_y , f_z –are the focal distances within the given image resolution of the given camera.

5 EXPERIMENTS SETTING FOR DETERMINING ACQUIRE ACCURATE USED 3D OPTICAL SYSTEM

Measurement accuracy control Kinect-in took place in two modes - manual and automatic. For the needs of the experiment was initially randomly selected 10 position on the map calibration, these positions were for both modes follow the same. In each position was then carried out 10 measurements in a row. The first mode has been selected, provided that the measurement point Wednesday localized balls from the first camera is selected manually by a cross-hairs. The second mode has been selected for a change so that the center of the balls were collected automatically by the intensity of the RGB color spectrum first camera. Always the first five positions close to the reference ball was turned on with the "Near" mode and five other remote measurements with a normal regime. The measurement was carried out on a screen distance of 800 mm to 1200 mm in steps of 100 mm, and each length was the experiment remeasured twice that the final measurement 10 and the number of considered within 5 (in the sequence: 800 mm, 900 mm, 1000 mm, 1100 mm and 1200 mm). Maximum length 1200 mm was chosen because it is the maximum operating distance arm together with effector.

6 METHODS USED IN AN EXPERIMENT TO DETERMINE THE ACCURACY OF THE 3D SCANNING OPTICAL SYSTEM KINECT

To actually verify the accuracy of the 3D scanning optical system, we calculated for each axis separately coordinate of the measured values of polynomial regression 6th order and overall uncertainty. Form of polynomial regression we obtained as [22]:

$$\Delta = b_0 + b_1 \cdot P + b_2 \cdot P^2 + b_3 \cdot P^3 + \cdots + b_n \cdot P^n \tag{19}$$

where:

Δ - polynomial regression,

b_i – parameters of the polynomial for i = 1, ..., n,

P – calculated deviation in point.

Confidence interval for the regression polynomial obtained by:

$$\left(\hat{\eta}(x) - t_{1-\frac{\alpha}{2}}(v)\sqrt{\widehat{D}\left(\hat{\eta}(x)\right)}\,;\hat{\eta}(x) + t_{1-\frac{\alpha}{2}}(v)\sqrt{\widehat{D}\left(\hat{\eta}(x)\right)}\right) \tag{20}$$

where:

$\hat{\eta}(x)$ – polynomial regression,

$\widehat{D}\left(\hat{\eta}(x)\right)$ – variance estimation,

t_1 – quantile of the Student distribution,

α – level of significance.

During calculations which are based on the assumption that $v = n - 2$, where n is the number of considered equations. For calculation was used quantile of the Student distribution t_1 = 1,533. The overall uncertainty we calculate, according to the following formula as:

$$\boldsymbol{Y} = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{U}^{-1}(\Delta)\boldsymbol{A})^{-1}\boldsymbol{A}^{T}\boldsymbol{U}^{-1}(\Delta)\Delta \tag{21}$$

where:

$\boldsymbol{Y}$ – overall uncertainty,

$\boldsymbol{A}^{\mathrm{T}}$ – matrix of the sensitivity coefficients,

$\boldsymbol{U}$ – covariance matrix,

Δ matrix of measured values.

Covariance matrix $\boldsymbol{U}(\Delta)$ is in the form:

$$\boldsymbol{U}(\Delta) = \begin{pmatrix} u^2(\Delta_1) & u(\Delta_1,\Delta_2) & u(\Delta_1,\Delta_3) & \cdots & u(\Delta_1,\Delta_m) \\ u(\Delta_2,\Delta_1) & u^2(\Delta_2) & u(\Delta_2,\Delta_3) & \cdots & u(\Delta_2,\Delta_m) \\ u(\Delta_3,\Delta_1) & u(\Delta_3,\Delta_2) & u^2(\Delta_3) & \cdots & u(\Delta_3,\Delta_m) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ u(\Delta_m,\Delta_1) & u(\Delta_m,\Delta_2) & u(\Delta_m,\Delta_3) & \cdots & u^2(\Delta_m) \end{pmatrix} \tag{22}$$

where:

$u^2(\Delta_1)$, ..., $u^2(\Delta_m)$ – uncertainty of individual measurement elements,
$u(\Delta_1, \Delta_2)$, ..., $u(\Delta_m, \Delta_m)$ – covariance between measuring points..

Term Δ it is a matrix measured values and the form is:

$$\Delta = \begin{pmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \\ \vdots \\ \Delta_n \end{pmatrix} \tag{23}$$

where:

$\Delta_1 - \Delta_n$ – measured values.

MAtrix $\boldsymbol{A}$ are matrix of experiments design (25), that occurs through (26):

$$\boldsymbol{P_n} = \frac{\partial f(\Delta)(b_0 + b_1 \cdot x + b_2 \cdot x_m^2 + b_3 \cdot x_m^2 + \cdots + b_n \cdot x_m^n)}{\partial b_{\mathrm{n}}} \tag{24}$$

where:

$\partial f(\Delta)$ – partial derivative of model,

$\partial f(\Delta)$ – partial derivative according to each variable.

$$\boldsymbol{A} = \begin{pmatrix} 1 & P_1 & P_1^2 & \cdots & P_1^n \\ 1 & P_2 & P_2^2 & \cdots & P_2^n \\ 1 & P_3 & P_3^2 & \cdots & P_3^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & P_m & P_m^2 & \cdots & P_m^n \end{pmatrix} \tag{25}$$

where:

$P_1 - P_m$ – deviation at a given point and a $\boldsymbol{b}$ is the matrix of parameters in the form**:**

$$\boldsymbol{b} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \tag{26}$$

where:

$b_0 - b_n$ - parameters of the polynomial for i = 1, ..., n.

8 CONCLUSION

The collision-free operation of the robotic system is a strategy which ensures that the potential collision states are identified in advance and thus, the risk of collision is minimized. The constructional solution and placement of the 3D optical system was verified for its use in the manipulation of the robotic arm. All the proposed hardware, localization and software solutions were carried out and proved to be fully functional. Based on the experiments and their evaluation, the precision of the localization using the MS Windows Kinect optical system in the HSV color space for our usage is satisfactory. The final average measurement precision for the localization of the object is ± 1,34 mm for the x axis, ± 1,47 mm for the y axis and ± 2,48 mm for the z axis, which considering the used low-cost technology indicates a fairly good precision, its use in mobile robotic can be well-justified.

ACKNOWLEDGMENT

The authors would like to thank the VEGA Ministry of Education, Science, Research and Sport of the Slovak Republic for financial support, project VEGA ID: 1/0144/15, VEGA ID: 1/0317/17. This support is very gratefully acknowledged.

REFERENCES

[1] VACHÁLEK, J., TÓTH, F., KRASŇANSKÝ, P., ČAPUCHA, Ľ.: Design and construction of a robotic vehicle with omni-directional mecanum wheels. In Sborník vědeckých prací Vysoké školy báňské - Technické univerzity Ostrava. Roč. 60, č. 1 (2014), s. 97-103. ISSN 1210-0471.

[2] VACHÁLEK, J., ČAPUCHA, Ľ., KRASŇANSKÝ, P., TÓTH, F.: Collision-free manipulation of a robotic arm using the MS Windows Kinect 3D optical system. In Process control 2015 : 20th International Conference on Process Control. Štrbské Pleso, Slovak Republic. June 9-12, 2015. [s.l.] : IEEE, 2015, S. 96-106. ISBN 978-1-4673-6627-4.

[3] VACHÁLEK, J., KRASŇANSKÝ, P., VAJSÁBEL, M.: Využitie 3D optického kamerového systému MS Windows Kinect pre potreby manipulácie robotického ramena mobilného robotického systému. In Metrológia a skúšobníctvo. Roč. 19, č. 2 (2014), s. 7-13. ISSN 1335-2768

[4] VACHÁLEK, J., TAKÁCS, G.: Robotika. 1. vyd. Bratislava: Nakladateľstvo STU, 2014. 166 s., 96 obr., 2 tab. ISBN 978-80-227-4163-7.

[5] CHUNG-LIANG, CH. AND BO-HAN, W., "A Dynamic Cooperative Scheme with Multiple Antennas for Indoor Mobile Robot Localization," Abstract and Applied Analysis, vol. 2013, Article ID 972371, 14 pages, 2013. doi:10.1155/2013/972371

[6] LIPING, Z., CHENG-CHEW, L., YIPING, C., AND HAMID, R. K., "Tracking Mobile Robot in Indoor Wireless Sensor Networks," Mathematical Problems in Engineering, vol. 2014, Article ID 837050, 8 pages, 2014. doi:10.1155/2014/837050

[7] TÖLGYESSY, M., "Kinect, hacked sensor in mobile robotics (Kinect, hacknutý snímač v mobilnej robotike in Slovak)." Posterus. [online]. 2011, r. 4, č. 5., Available on the Internet: http://www.posterus.sk/?p=10623

[8] MICROSOFT. KINECT FOR WINDOWS BLOG, "Use the Power of Kinect for Windows to Change the World. MSDN. [online]. 2012." Available on the Internet: http://blogs.msdn.com/b/kinectforwindows/archive/2012/01/09/kinect-for-windows-commercial-program-announced.aspx

[9] MICROSOFT. KINECT FOR WINDOWS, “Sensor MSDN. [online]. 2014.” Available on the Internet: http://msdn.microsoft.com/en-us/library/hh855355.aspx

[10] MATHWORKS. MICROSOFT KINECT FOR WINDOWS, “Support from Image Acquisition Toolbox. [online]. 2014.” Available on the Internet: http://www.mathworks.com/hardware-support/kinect-windows.html

[11] MATHWORKS. MICROSOFT KINECT, “Support from Simulink. [online]. 2014.” Available on the Internet: http://www.mathworks.com/hardware-support/kinect-simulink.html

[12] MICROSOFT. KINECT FOR WINDOWS BLOG, “Pre-order your Kinect for Windows v2 sensor starting today. MSDN. [online]. 2014.” Available on the Internet: http://blogs.msdn.com/b/kinectforwindows/default.aspx?PageIndex=1

[13] MICROSOFT. KINECT FOR WINDOWS, “Kinect for Windows features. MSDN. [online]. 2014.” Available on the Internet: http://www.microsoft.com/en-us/kinectforwindows/discover/features.aspx

[14] TÖLGYESSY, M., CHOVANEC, L. PÁSZTÓ, P. AND HUBINSKÝ, P., “A Plane Based Real-Time Algorithm For Controlling A Semi-Autonomous Robot With Hand Gestures Using The Kinect.” International Journal of Imaging & Robotics, vol.13, no.2, 2014

[15] PÁSZTÓ, P., KĽÚČIK, M., CHOVANEC, L., TÖLGYESSY, M., HANZEL, J., DUONG, Q. K. AND HUBINSKÝ, P., “Object Relative Position Estimation Based on Hough Transform Using One Camera.” International Journal of Imaging and Robotics Vol. 13, Iss. 2. s. 1--11. ISSN 2231-525X., 2014

[16] HAGGAG, H., HOSSNY, M., FILIPPIDIS, D., CREIGHTON, D., NAHAVANDI, S., PURI, V., “Measuring depth accuracy in RGBD cameras.” IEEE, Signal Processing and Communication Systems (ICSPCS), 2013 7th International Conference on, vol., no., pp.1,7, 16-18 Dec. 2013

[17] MATHWORKS, “Computer Vision System Toolbox: Camera Calibration. [online]. 2014.” Available on the Internet: http://www.mathworks.com/products/computer-vision/description5.html

[18] JEAN, Y. B., “Camera Calibration Toolbox for Matlab. [online]. 2013.” Available on the Internet: http://www.vision.caltech.edu/bouguetj/calib_doc/

[19] FETIC, A., JURIC, D. AND OSMANKOVIC, D., “The procedure of a camera calibration using Camera Calibration Toolbox for MATLAB.” IEEE, MIPRO, 2012 Proceedings of the 35th International Convention , vol., no., pp.1752,1757, 21-25 May 2012

[20] HERRERA, C. D., KANNALA, J. AND HEIKKILÄ, J., “Joint Depth and Color Camera Calibration with Distortion Correction.” IEEE, Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.34, no.10, pp.2058,2064, Oct. 2012

[21] ŠKUTA, J., BABIUCH, M. AND TUMA, J., “Design of hardware of inercial navigation systems.” In: Engineering mechanics 2007, Engineering Academy of the Czech Republic, May 14-17, 2007, p.277, ISBN 978-80-87012-06

[22] PALENČÁR, R. (2005): Measurement models.In: Measurement in Technology: A textbook from the multimedia courseware METROMEDIA-ONLINE II. - Bratislava: Ing. Peter Juriga - Grafické štúdio, 2005. - ISBN 80-89112-05-6. - S. 594-640