IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 4, 2021, accepted August 14, 2021, date of publication August 18, 2021, date of current version August 31, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3105720

Application of the ZED Depth Sensor for Painting Robot Vision System Development

VLADIMIR TADIĆ[1], ÁKOS ODRY[2], ERVIN BURKUS[1], ISTVÁN KECSKÉS[1], ZOLTÁN KIRÁLY[1], ZOLTÁN VÍZVÁRI[3], ATTILA TÓTH[4], AND PÉTER ODRY[1], (Member, IEEE)

[1]Institute of Informatics, University of Dunaújváros, 2400 Dunaújváros, Hungary
[2]Technical Department, Faculty of Engineering, University of Szeged, 6724 Szeged, Hungary
[3]Department of Environmental Engineering, Faculty of Engineering and Information Technology, University of Pécs, 7624 Pécs, Hungary
[4]Medical School, Institute of Physiology, University of Pécs, 7624 Pécs, Hungary

Corresponding author: Vladimir Tadić (tadityv@uniduna.hu)

This work was supported by the European Union under Project EFOP-3.6.3-VEKOP-16-2017-00009, Project EFOP-3.6.1-16-2016-00004, Project 2020-4.1.1-TKP2020, Project KFI_16-1-2017-00485, and Project 2020-1.1.2-PIACI-KFI-2020-00173.

ABSTRACT This paper presents the use of the ZED depth sensor in a robot-based painting application. The use of a stereo depth sensor is a very important factor in robotic applications, since it is both the initial and the essential step in a sequence of robotic operations, where the goal is to detect and extract the useful surface and objects or the obstacle on a wall that is not intended for painting. The ZED depth sensor was used for surface recording and navigation of our painting robot. Later, wall extraction was performed using simple image processing and morphological operations in a surface extraction algorithm. The goal was to use well-known, simple, and proven image processing operations in the algorithm to ensure both reliable and smooth operation of the robot's vision system in an industrial environment. The experiments showed that the developed algorithm detects and extracts the wall successfully under various depth measurement conditions.

INDEX TERMS Depth image, image processing, obstacles, painting robot, ZED depth sensor.

I. INTRODUCTION

This paper introduces a procedure for window and obstacle detection from a depth image captured by a ZED depth sensor. The scope of this industrial research is the development of a simple image processing algorithm supported with information from a depth camera. The wall extraction algorithm is based on both simple image processing operations and a sequence of morphological operations. This paper further develops the previous research on a specific robotic application based on depth image processing captured using a low-cost stereo depth camera. This project is a part of the KFI_16-1-2017-00485 project related to the painting robot development for thermal insulation and painting of facades of monument buildings [1]. The research essentially is an industrial project aiming to develop a robot that would automatically—based on information obtained from the depth sensor—paint a building's facades or interior walls with an insulating wall-paint. The directive and the main goal of the project is to use already proven and reliable image processing operations and algorithms to ensure the smooth and reliable operation of a painting robot. Almost no wall painting robots on the market are automated and the operator entirely controls the painting arm and decides what should be painted. In this project, the new requirement was the automation using an image processing vision system. This strategy is quite new in commercial wall painting via robot construction. The basic hardware for the vision system is a Intel RealSense D435 depth camera [1]–[4], which is replaced by a StereoLabs ZED depth camera [5]. The goal is to develop a robust and simple computer vision algorithm to detect and extract mainly rectangular windows and large rectangular obstacles on the wall near the window area using depth images recorded with a stereo camera, as well as notifying the controlling system [1]. The acquired depth image and the point cloud of the environment model serve as resources for the robot to determine which surfaces to paint.

In previous research, the robot's vision system was developed using the real-time appearance-based mapping (RTAB-Map) algorithm and the random sample consensus (RANSAC) algorithm [1]. The main reason for using the mentioned procedures is that the RealSense depth camera

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung.

yields the best depth measuring results from a distance of approximately 60-70 cm [1]–[4]. Unfortunately, the size of the captured wall surface at this distance is only approximately 85 cm × 63 cm. Hence, the RTAB-Map algorithm was used to extend this small wall surface [1]. The input point cloud of the wall surface is formed via the RTAB-Map algorithm using zigzag robot movements along with a built-in RealSense depth sensor by merging recorded point clouds in movement. This mapping operation requires the movement of the robot's arm, and it lasts approximately 2 minutes. Later, by executing the RANSAC algorithms followed by a clustering algorithm, the wall surface is detected, extracted, and prepared for the painting process [1].

All the experiments and testing were performed on the ground and the previous project goal was fulfilled [1].

The next requirement in the project was to avoid moving the robot during the path planning with a mapping algorithm because his intention is to reduce the duration of the entire wall extraction and wall painting procedure to speed up the whole image processing and painting procedure. The new plan is to paint a large number of buildings in the shortest possible time.

Hence, the new request was the development of a simplified detection algorithm, without the unnecessary robot movements for depth image recording. Therefore, the procurement and the study of a new and better depth sensor was the next step. The new and specific requirement was the development of a vision system that captures the image from one position, and later, the wall area that should be painted must be extracted with a much simplified, but reliable, image processing algorithm.

The sensor assessment study culminated with the choice of the StereoLabs ZED depth sensor. The ZED depth sensor is among the best commercial depth sensors on the market [5]–[8], and it measures the depth with very high accuracy. It has numerous advantages over RealSense depth cameras, such as a better depth resolution, better depth accuracy, wider field of view, higher red, green and blue (RGB) resolution, and wider distance range [1]–[8]. However, its price is many times higher. Some of the main advantages are summarized in Table 1.

TABLE 1. Comparison of ZED and RealSense cameras.

Features	ZED	RealSense
Depth resolution	32 bit	16 bit
Max. RGB resolution	2208 × 1242 pixels	1920 × 1080 pixels
Range	1-20 m	0.2-10 m
Field of view	110°	90°

The price of this camera is negligible in relation to the lost time and lost material resources when painting a large number of buildings, and these are the main reasons for this new assignment.

The main part of the equipment developed to move the spray gun and depth sensor is a 5-degree-of-freedom (DOF) robotic arm [1]. The housing of the joints and the segments connecting them are made of aluminum alloys. For the movement of the joints, harmonic drives and servo motors are used. The robot's reach is 1820 mm, and its total weight is approximately 50 kg [1].

The block diagram of the robot-based painting system is given in Fig. 1. The 5-DOF robot is operated through three Delta ASDA series servo drives and two BGE (in German "Bürstenlose Gleichstrommotoren RegelElektroniken" - Brushless Direct Current Motors Controller) controllers manufactured by Dunkermotoren from Bonndorf in Germany (www.dunkermotoren.com). These drivers control the joint motors via the CANopen interface (Device profile DSP402). At the end of the kinematic chain, the ZED camera is attached to the tool coordinate frame. Such robot segment lengths were employed in the mechanical system that enables covering approximately the 2 m × 2 m surface with the minimum distance from the wall. The robot model was initially implemented in the Robot Operating System (ROS) framework with the Unified Robot Description Format (URDF) description format. Then, the CANopen interface was set up, and the communication with motor drivers was established. The inverse kinematics of the robot were derived as a plug-in package for the MoveIt motion planning framework. MoveIt enabled defining the home positions of the robot, incorporating the RGB and depth (RGBD) data into the generation of the occupancy map and establishing the general services to realize motions in the 3D space with obstacle avoidance. A custom graphical user interface (GUI) was developed for the whole painting process, which samples the RGBD data, identifies the wall surface, windows, doors and obstacles, and generates the path to cover the wall and drive the robot according to generated trajectories.

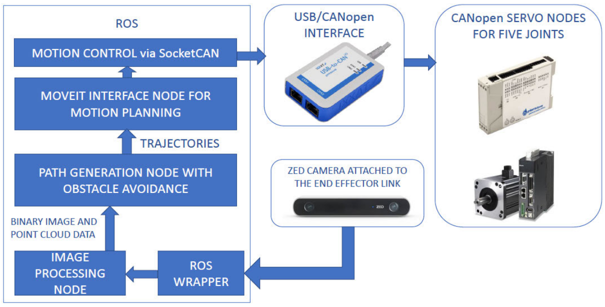

ROS
MOTION CONTROL via SocketCAN
MOVEIT INTERFACE NODE FOR MOTION PLANNING
TRAJECTORIES
PATH GENERATION NODE WITH OBSTACLE AVOIDANCE
BINARY IMAGE AND POINT CLOUD DATA
IMAGE PROCESSING NODE
ROS WRAPPER
USB/CANopen INTERFACE
CANopen SERVO NODES FOR FIVE JOINTS
ZED CAMERA ATTACHED TO THE END EFFECTOR LINK

FIGURE 1. Block diagram of the robot-based painting system.

The RGBD camera was included in the ROS framework via its ROS wrapper. This wrapper enabled obtaining the real-time measurements on multiple ROS topics [1]–[3].

A detailed description of the robot is not within the scope of this paper, and it is not possible at this time since both the robot and its parts are under a patenting process. Therefore, it is not described in this paper to avoid legal infringement. The robot's construction will be fully described in a future research paper.

Finally, using the strict instructions about the simplicity and reliability by the project, a new image processing algorithm was developed for the wall detection and extraction.

This paper is organized as follows. The first section is the introduction, the second section summarizes the related works, and the third section introduces the ZED camera in brief. Section four describes the wall extraction algorithm, section five shows the experiments and results, while the sixth section provides the conclusion.

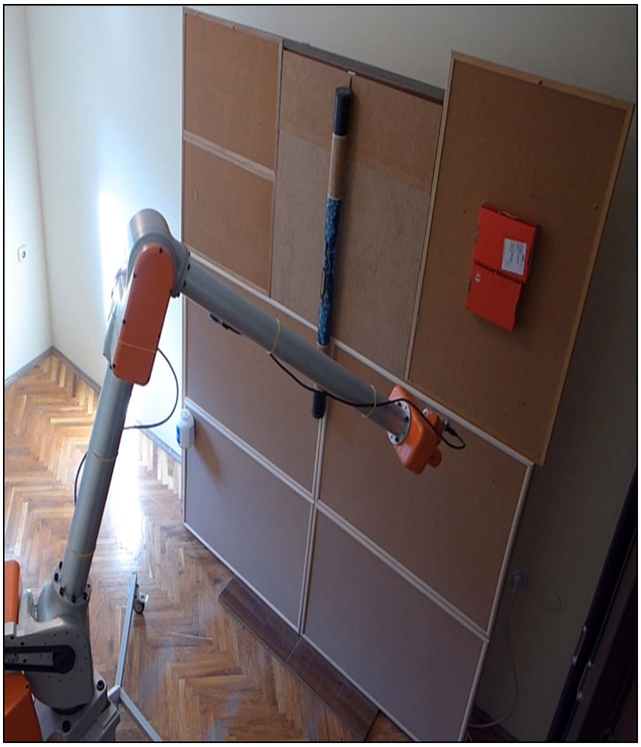


FIGURE 2. The painting robot in the testing process.

II. RELATED WORK

Plane and obstacle detection and extraction are common problems in robot vision systems [19]–[30]. This action can be determining in certain applications where it is necessary to separate some objects from the background. There are various ways to perform this separation. In this section, a brief overview of the related works is given. Notably, there are practically no such solutions for painting robots in the literature. Therefore, this industrial research presents a completely new approach.

Tadic *et al.* [1] proposed a painting robot path planning procedure based on the RTAB-Map algorithm. Later, wall extraction was performed using the RANSAC iterative algorithm. They stated that the developed method worked well. In [2] the authors introduced an edge-preserving and fuzzy enhancement method in the wall detection and extraction algorithm to improve the low-quality depth image of the captured wall-window area with obstacles. The experiments showed that the proposed method extracts the wall surface successfully under strict depth image capturing conditions. Cunha *et al.* [12] introduced the results of the application of a Kinect sensor on a wheeled indoor service robot for elderly assistance. The robot used the depth metric map of the environment and the depth information of the depth camera to extract the wall areas and localize itself in the environment. The authors used an error minimization method to provide real-time efficient robot pose estimation. During the experiments, they successfully navigated the developed robot in a domestic environment, across different rooms without colliding with obstacles. Flacco *et al.* [13] proposed a real-time collision avoidance procedure for safe human-robot coexistence. They claimed that the main contribution of their research is a fast method to evaluate distances between the robot and possibly moving obstacles, based on the depth images. The distances are utilized to generate repulsive vectors that are used to control the robot during a generic motion task. Jawale *et al.* [14] proposed a painting robot system for interior wall painting. The robot is constructed using a few steel components, a spray gun, a conveyor shaft, and a controller unit to control the robot system. The developed robot does not possess a computer image system for navigation. The authors claim that the robot is compact, has a high speed and pressure capabilities, and is very reliable during the painting process. In [15], the authors developed an industrial painting robot programmed for painting of alphabets. FlexPendant was used to manually program the robot to track the paths for specific targets of letters. This robot is constructed without a vision system. The experiments demonstrate that the implementation of such a system helps to increase the quality of painting and reduce paint consumption. Keerthanaa *et al.* [16] proposed an automatic wall painting robot design with the goal of realizing low-cost painting equipment. The authors used an infrared (IR) transmitter and IR receiver to detect the presence of the wall, and a microcontroller unit was used to control the movement of the embedded direct current (DC) motor. They stated that the developed robot is cost effective, reduces the work force for human workers, and reduces the time consumption. In [17], the authors introduced a FlexPaint project that created a methodology to automatically generate robot programs for spray painting unknown objects. The solution consists of four steps: laser triangulation sensing, geometric feature detection, tool path planning, and the generation of the collision-free executable robot program. The authors used a MAPP 2500 Smart Vision Camera to generate 3D point clouds of an object that should be painted. During the operation, all steps are fully automatic, and no intervention of an operator is needed during the whole process. The authors claim that convex parts can be painted automatically, but that complex concave shapes, such as the truck chassis, are not possible to paint with the developed painting robot. Borhade *et al.* [18] developed a low-cost painting robot for automated exterior wall painting. They constructed a climbing robot with the ability to maneuver on vertical surfaces, i.e., for painting in vertical robot movements. The authors reported that the robot is cost effective, eliminates work on scaffolds, reduces the need for human workers and finally reduces the painting time.

III. THE ZED DEPTH SENSOR

This section briefly introduces the technology and some of the more important characteristics of the ZED sensor, alongside its working principle.

The ZED depth sensor is a passive device without an active ranging device. This depth sensor uses a binocular camera to capture 3D scene information, extracts the disparity of the object applying a stereo matching algorithm and finally calculates the depth information according to the camera parameters [5]–[10].

The ZED depth camera is composed of stereo 2K cameras with dual 4 MP RGB sensors. The two RGB sensors/cameras have a fixed base distance of 120 mm, which can generate depth images up to 20 m (40 m is the maximum distance on the new updated firmware according to the manufacturer) [5]. It has a universal serial bus (USB) video device class (UVC) compliant USB 3.0 port backward compatible with the USB 2.0 standard. It is very important to note that the ZED 3D depth camera is optimized for real-time calculation using NVIDIA Compute Unified Device Architecture (CUDA) technology. Therefore, an appropriate graphical processing unit (GPU) and computer hardware are required for the ZED camera to be used [5]–[10].

FIGURE 3. The ZED depth sensor.

The ZED device has wide angle lenses, a field of view of 110° and it can stream uncompressed video at a rate up to 100 fps in Wide Video Graphics Array (WVGA) format. The WVGA is a standardized display resolution with the same 480 pixel height as VGA but wider, such as 720 × 480 (3:2 aspect ratio), 800 × 480 (5:3), 848 × 480, 852 × 480, 853 × 480, or 854 × 480 (16:9). The depth map image is expressed with 32-bit resolution, which results in a very accurate and precise depth image that depicts the depth differences, i.e., different distances from the plane of the camera. The left and right video frames are synchronized and streamed as a single uncompressed video frame in the side-by-side format. Several configuration parameters of the on-board image signal processor (ISP), such as resolution, brightness, contrast, and saturation can be adjusted through the software development kit (SDK) that is provided by the StereoLabs development team [5]. Furthermore, ZED cameras support various software packages, called "wrappers" such as MATLAB, ROS, and Python wrappers. All of these software packages allow the user to modify different parameters depending on the user needs, such as the quality of depth, the quality of the image, the sensing mode, the number of frames per second, and the name of topics [5]–[10].

Fig. 3 shows the accuracy of the ZED depth sensor depending on the distance of an object from the depth camera. The depth resolution, i.e., the depth precision, deteriorates with the increasing distance [5].

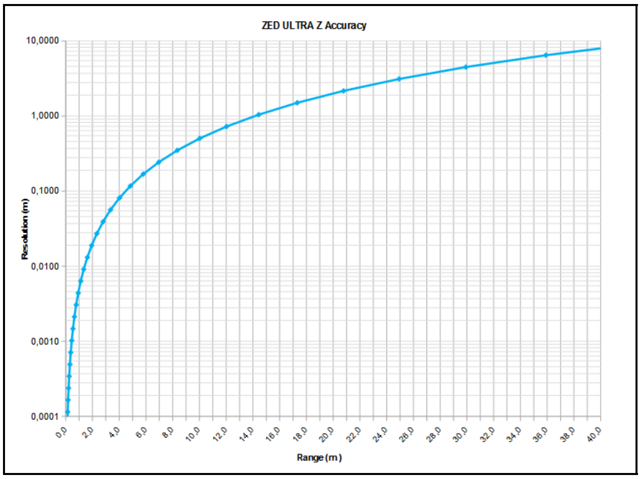


FIGURE 4. The accuracy graph of the ZED depth sensor (courtesy of StereoLabs) [5].

TABLE 2. Relevant features of the ZED depth camera.

Size and weight	Dimensions: 175 × 30 × 33 mm Weight: 159 g
Depth	Range: 1-20 m Format: 32 bits Baseline: 120 mm
Sensors	Size: 1/3 Format: 16:9 Pixel Size: 2-u pixels
Lens	Field of View: 110° *f*/2.0 aperture
Individual image and depth resolution in pixels	HD2K: 2208 × 1242 (15 fps) HD1080: 1920 × 1080 (30, 15 fps) HD720: 1280 × 720 (60, 30, 15 fps) WVGA: 672 × 376 (100, 60, 30, 15 fps)
Connectivity and working temperature	USB 3.0 (5 V/380 mA) 0°C to +45°C
SDK System minimal requirements	Windows or Linux Dual-core 2.3 GHz CPU 4 GB RAM NVIDIA GPU with compute capability > 3.0

Here, at the end of this brief description, it should be noted that each ZED depth camera comes with a unique factory calibration file, which is downloaded automatically. While using factory settings is recommended, users can also calibrate the ZED device with the ZED SDK software package [5]. The relevant features of the ZED depth camera are summarized in Table 2 [5].

IV. WALL EXTRACTION ALGORITHM

In the following section, the wall extraction algorithm is described.

Fig. 5 displays the block diagram of the proposed image processing procedure and algorithms. The input of the algorithm is the grayscale depth image in 32-bit resolution. Similar image processing operations are executed in parallel. The upper branch of the algorithm refers to the extraction of the wall surface together with the possible obstacle, while the lower branch of the algorithm performs the extraction of the possible obstacle surface. After the depth image is loaded, the ranking of the distance values (Z coordinate values) is performed. This ranging operation eliminates all the unnecessary depth image components that have no effect on the algorithm and the application itself. The ranging operation is followed by a binarization operation using a simple thresholding operation [18]–[21]. After the thresholding operation is applied, a sequence of morphological operations is performed to extract the wall area that should be painted. The wall surface intended for painting is determined using simple binary image subtraction, where the extracted obstacle area is subtracted from the extracted wall and obstacle area [21]. Finally, the extracted binary wall surface, i.e., the binary mask, is paired with the point cloud coordinates obtained by the ZED depth sensor, and based on this information, the robot's navigation system generates the trajectory for the painting.

The extracted surface and the obstacles are described in the system for the robot painting operation as follows. The processing node generates a binary image based on the input depth image using the proposed image processing algorithm. This binary map is used to generate the spatial trajectories for the robot: the 3D coordinates of the occupied cells in the binary map (i.e., the obstacle) were extracted from the point cloud data, and the zigzag painting trajectory is generated in such a way that those spatial coordinates that belong to the obstacles discontinued the painting path. As a result, the zigzag painting trajectories cover all free areas on the wall and the obstacles are successfully omitted during the painting operation.

First, it should be noted that the distance of the camera from the wall and the position of the camera's sensor play crucial roles in the formulation of the size, orientation, and shape of the structuring elements (SEs) [18]–[22]. Considering these facts, the specified SEs were formed. The dimensions, shape, and orientation of the SEs were designed based on experimentally determined conditions for the morphological operations used in the algorithm.

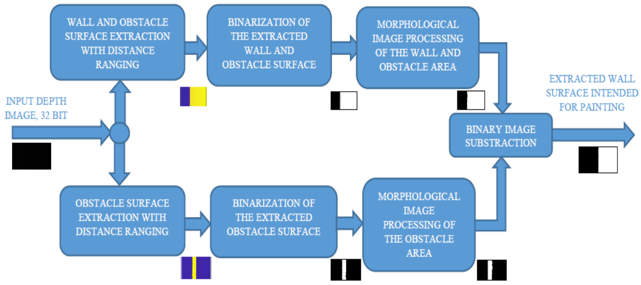

WALL AND OBSTACLE SURFACE EXTRACTION WITH DISTANCE RANGING
BINARIZATION OF THE EXTRACTED WALL AND OBSTACLE SURFACE
MORPHOLOGICAL IMAGE PROCESSING OF THE WALL AND OBSTACLE AREA
INPUT DEPTH IMAGE, 32 BIT
EXTRACTED WALL SURFACE INTENDED FOR PAINTING
BINARY IMAGE SUBSTRACTION
OBSTACLE SURFACE EXTRACTION WITH DISTANCE RANGING
BINARIZATION OF THE EXTRACTED OBSTACLE SURFACE
MORPHOLOGICAL IMAGE PROCESSING OF THE OBSTACLE AREA

FIGURE 5. Block diagram of the image processing algorithm of the robotized painting system.

The term SE is related to morphological image processing. Morphological operations are determined in terms of sets [22]. In image processing algorithms, morphology is used with two types of sets of pixels, the objects and SEs. The objects are determined as sets of foreground pixels (white pixels on black background). SEs can be defined in terms of foreground and background pixels. The SEs are used in a form similar to spatial convolution masks and kernels. Therefore, SEs have the role of an operator mask in morphology by which morphological operations are performed in morphological image processing [21], [22]. The most important requirement was that the depth image of the wall or obstacle needs to be taken from a specific distance, in this application 1 m, and the baseline and the lens of the depth camera must be positioned parallel to the plane of the wall/obstacle [5]. This is very important, since the robotic arm with the paint gun also moves parallel to the plane of the wall based on the coordinates from the captured point cloud related to the depth image. The distance is determined based on the properties of the camera lens itself, since the lens is factory made/calibrated so that the depth camera gives the most accurate depth image from a distance of 1 m [5]. This distance was suggested by the StereoLabs Engineering Support Team after consultations and test measurements performed by the StereoLabs team to present us with the capabilities of the ZED camera [5]. Hence, according to the StereoLabs engineering team, the 1 m distance is best suited for solving the problem of depth in measuring the surface of the wall and the environment to separate the wall itself due to later painting operation [5]. Based on this determined distance from the wall or obstacle, the measured wall surface (field of view) is approximately 160 cm × 90 cm. This surface is significantly larger than the surface captured by the RealSense camera [1]–[4]. Therefore, the wall extraction algorithm can be developed without robot arm manipulations. The distance of 1 m can vary approximately ±1 cm due to the influence of a slight wind, which can slightly move the robot's arm with the depth camera, but these variations do not influence the image acquisition and the accuracy. However, in the case of strong wind, it would not be possible to carry out the painting since the paint is being sprayed with a spray gun, as in any other case utilizing a manual painting operation. The plan is to have the depth camera mounted under the spray gun on a robotic arm, or it can potentially be mounted on a separate arm to avoid having the spray paint smear the camera. This separate arm would be small-sized component and would move in the opposite direction from the wall during the wall painting to avoid the paint. However, this part of the project will be tested after the patenting of the robot itself and the completion of the development of the painting equipment. Therefore, this part of the project will be the topic of a separate paper focusing on the robot's description. The focus of this paper is on the image processing algorithm.

Likewise, the angle of the depth camera is not a significant factor in the algorithm if the camera's lens and sensor remain parallel to the wall plane. However, the slant of the depth camera can produce a distorted depth image of the wall surface and can also create faulty coordinates of the surface that should be painted [5]. The result would be the incorrect movement of the robot with the painting equipment. Additionally, the lighting conditions are significant for creating the depth image. The worst case is the direct impact of sunlight or strong light directly into the camera sensor. These effects would blind the ZED camera [5]. Furthermore, the impact of a strong shadow on the surface being captured further impairs the visibility to the camera and may result in a damaged depth image in the form of invisibility of some parts of the depth image [5]. In essence, the mentioned factors always result in artifacts, when using any camera. One solution could be the use of optical filters, but in practice and in the literature, this has not been confirmed to have a positive impact on the aforementioned problems during the use of depth cameras in industrial applications [5]. The effects of the specified prerequisites are illustrated in later examples.

Since a depth sensor is used, the main information for algorithm development is the depth distance in the direction of the Z coordinate, i.e., the distance between the wall and the camera. This means that it is necessary to find the range within which the distance of the wall and possible obstacles on the wall to the sensor vary. This variation is a consequence of the optics of the camera itself [5]. The distance at the height of the camera from the wall directly at a given point is exactly 1 m, and this distance slowly increases by distancing from that point due to the oval shape of the lens [5]. In the case of a ZED camera, this deviation from a distance of 1 m from the wall is approximately 3 cm over the entire captured surface for a given field of view of the depth camera. These data were determined experimentally by measurements.

By increasing the distance between the camera and the wall, the scattering increases, which makes it practically impossible to separate the wall from the window or obstacle.

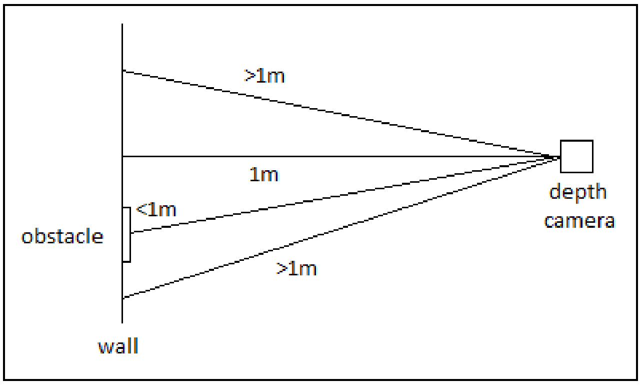


FIGURE 6. A sketch of the position of the depth camera in relation to the wall and obstacle.

The experimental results show that the depth camera from a 1 m distance can reliably separate objects with a difference in depth distance of a maximum of 2 cm. This result is clearly acceptable, as the depth difference between the wall and the window in most cases is at least 4-5 cm or more, and in this way, it is possible to perform the separation without any problems. Furthermore, this reliable 2 cm distance sensing is also sufficient for obstacle detection, since the rectangular obstacles are most often thicker than 2-3 cm and the depth is detected with high precision by the ZED camera [5]. Additionally, the slant of the depth sensor can cause wall distances from the depth camera in the Z coordinate direction than are longer than the maximum allowable measurements of 3 cm, which can also make it impossible to separate the wall from the window, since the distance range in some parts of the wall would be confused with the distance to a window. Therefore, the depth sensing values when the depth camera is properly set 1 m from but parallel to the wall surface are used in the algorithm for the Z value thresholding to extract the wall.

A. WALL EXTRACTION WITH OBSTACLE

In the next section, the procedure of the wall surface extraction with the potential obstacle is described.

First, the depth image is captured in 32-bit resolution, and this depth image is loaded as the input of the proposed image processing algorithm. The original input depth image is shown in Fig. 7 (b). After the depth image is captured, the first operation is to find the Z value, i.e., the distance thresholding in the range of 930 mm $< Z <$ 1030 mm to separate the wall area with possible obstacles from the window [20]. As mentioned, the maximum spread in Z distance is 3 cm, or 30 mm since the depth camera records the Z values in mm. The value of 930 mm is empirically determined since all the obstacles that are significant to the client are approximately 2-6 cm thick. Naturally, this value can be increased, but the maximum value of 70 mm is definitely sufficient for applications where the customer wants to use the painting robot. The ZED sensor software records the depth image in grayscale, but for a better view, the depth map with distances was analyzed via a color map. The reason for this is that the depth is expressed using 32 bits, and therefore, a basically black image is displayed via the monitor, as shown in Fig. 7 (b) [5].

Fig. 7 shows the RGB image of the scene, the original input depth image in 32-bit resolution, and the visible depth map of the depth image. Using the colored depth map, the distribution of the Z values over the whole depth image can be seen. The green-yellow shade of color represents the 1 m = 1000 mm distance from the depth sensor. The yellow color represents the window area that is approximately 1060 mm from the camera. The bright green part is the obstacle near the window. The blue line in the left part of the image is the dead zone due to the baseline of the depth camera [5].

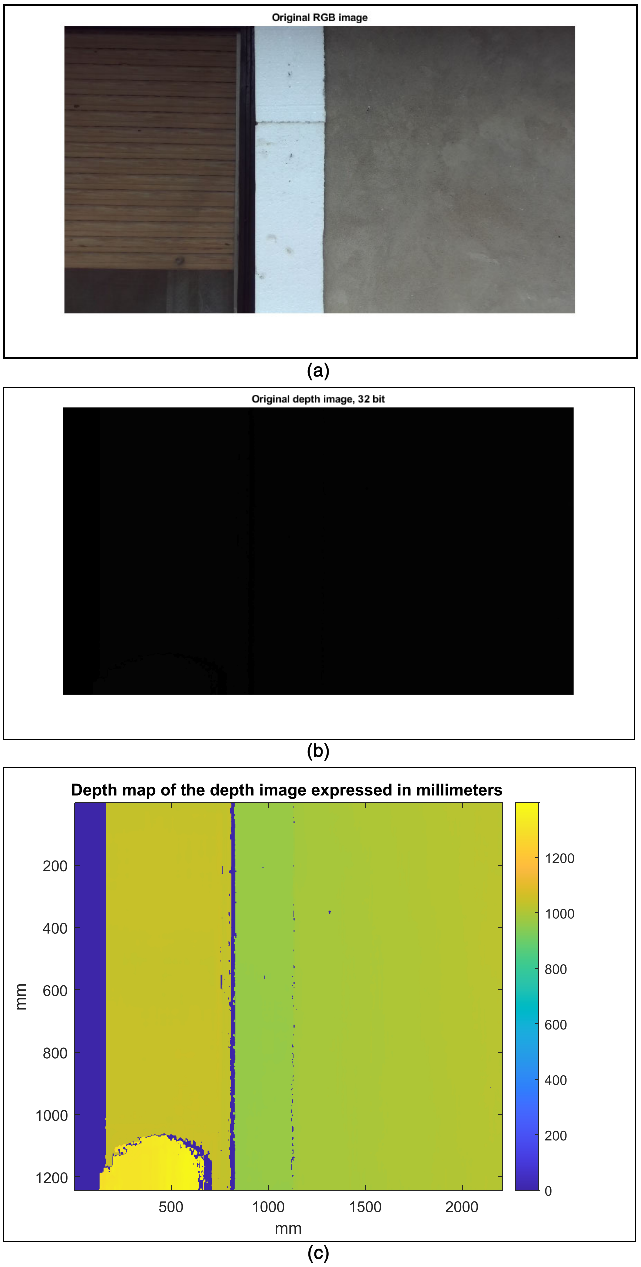


FIGURE 7. (a) Original RGB image; (b) the original input depth image in 32-bit resolution; (c) the depth map of the image expressed in millimeters.

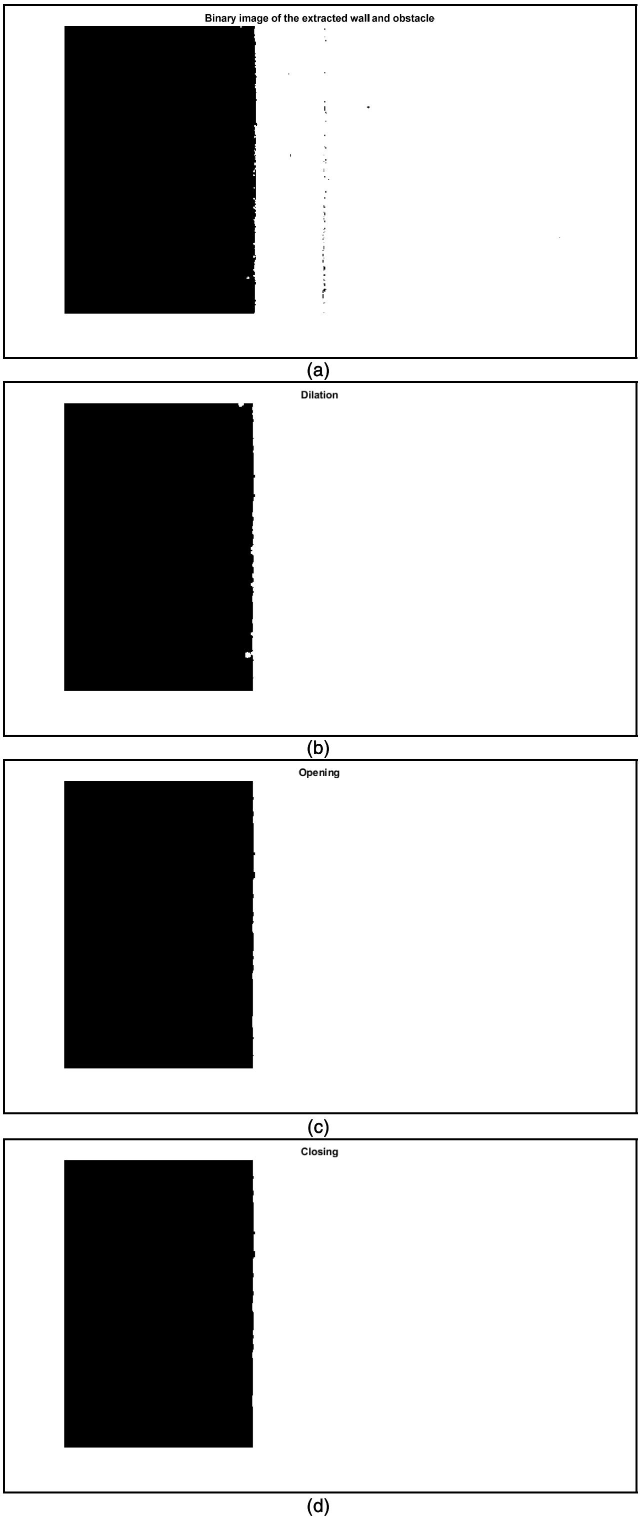


FIGURE 8. Sequence of binary and morphological operations for the wall and the possible obstacle extraction: (a) binary image; (b) dilation; (c) opening; (d) closing.

The dead zone is approximately 12 cm in width, and this is a constant occurrence during recording. In that area, the depth camera does not see anything, and the Z values there are 0 [5].

After the ranging operation in the 930 mm $< Z <$ 1030 mm interval, the binarization operation is performed using global thresholding [20]. The binary image contains both the wall and the obstacle. After the binarization, a sequence of morphological operations is performed to enhance the extracted surface. In the proposed image processing procedure, the applied disk-shaped SEs are determined empirically during the development and testing. Notably, the solution is not unique, and the shape and size of SEs can be easily adjusted according to the properties of other depth cameras that would give a different depth image, or according to other image processing problems [22]. The radii of the applied disk-shaped SEs in the algorithm are 8, 20 and 10 pixels [22].

Hence, the use of the appropriate SEs can affect the accuracy, but these SEs are often determined empirically, the choice of SEs is usually case dependent, and sometimes it is possible to use different SEs to solve the same problem [21], [22].

The first operation is the dilation with a disk-shaped structural element [21]. The goal of this operation is to fill in the holes, gaps, and crevices in the white part of the binary image. The dilation operation widens the object in the image [18]–[22]. The result of the dilation is shown in Fig. 8 (b). The small black gaps in the white area of the image are removed or filled in. This dilation is followed by a morphological opening operation [20] with larger disk-shaped SEs [21] to eliminate small protrusions along the edge of the white colored area, i.e., both the wall and obstacle areas. The opening operation smooths the contour of the object, repairs narrow cracks and eliminates thin protrusions. This operation is used for analyzing the dimensions of the objects [18]–[22]. Fig. 6 (c) shows the result of the opening operation. The edge of the white area is almost smooth, but with small recesses. To remove these small recesses, the morphological closing operation [20] is performed again with a disk-shaped SE [21] that is slightly larger than the SE in the performed dilation operation. Closing also makes some parts of the contour smooth, but in contrast to opening, it widens the narrow cracks and long thin depressions, eliminates small holes, and fills cracks in the contour. It is used for analyzing the distance between the objects [18]–[20]. Fig. 8 (d) shows the result of the closing operation, the extracted wall area with the possible obstacle. The edge of the white area is smoother, and only very small cracks remain that are not significant for further processing.

B. OBSTACLE EXTRACTION

The next step in the algorithm is the obstacle detection and extraction. Here, the same 32-bit resolution depth image is used as the input for the obstacle detection part of the image processing algorithm from Fig. 7 (b). Since the obstacle is closer to the depth camera, the Z distance values of its area are smaller than 1 m.

As mentioned before in the paper, the minimum distance that the ZED camera can separate reliably is 2 cm. Hence, this is the value that is significant for the algorithm. The determined interval of Z values for the obstacle is 930 mm < Z < 980 mm, and this is the area that is ranged from in the original depth image. It is the bright green part of the depth map. After the ranging operation [18]–[20], the binarization is performed again with the global thresholding method [20]. As illustrated in Fig. 9 (a), the obstacle area is extracted with high accuracy. Additionally, the extracted obstacle area possesses the same shortcomings as before in the wall and obstacle separation, i.e., holes, gaps, and cracks, around its contour or boundary. All of these small deformations and irregularities along the contour are eliminated with the same sequence of morphological operations as before during the wall and obstacle separation postprocessing [21]. The results of these operations are presented in Fig. 9 (b)-(d). Finally, after the closing operation is performed, the obstacle area is extracted with high accuracy. This is a very important result since the obstacle area is needed for the wall extraction.

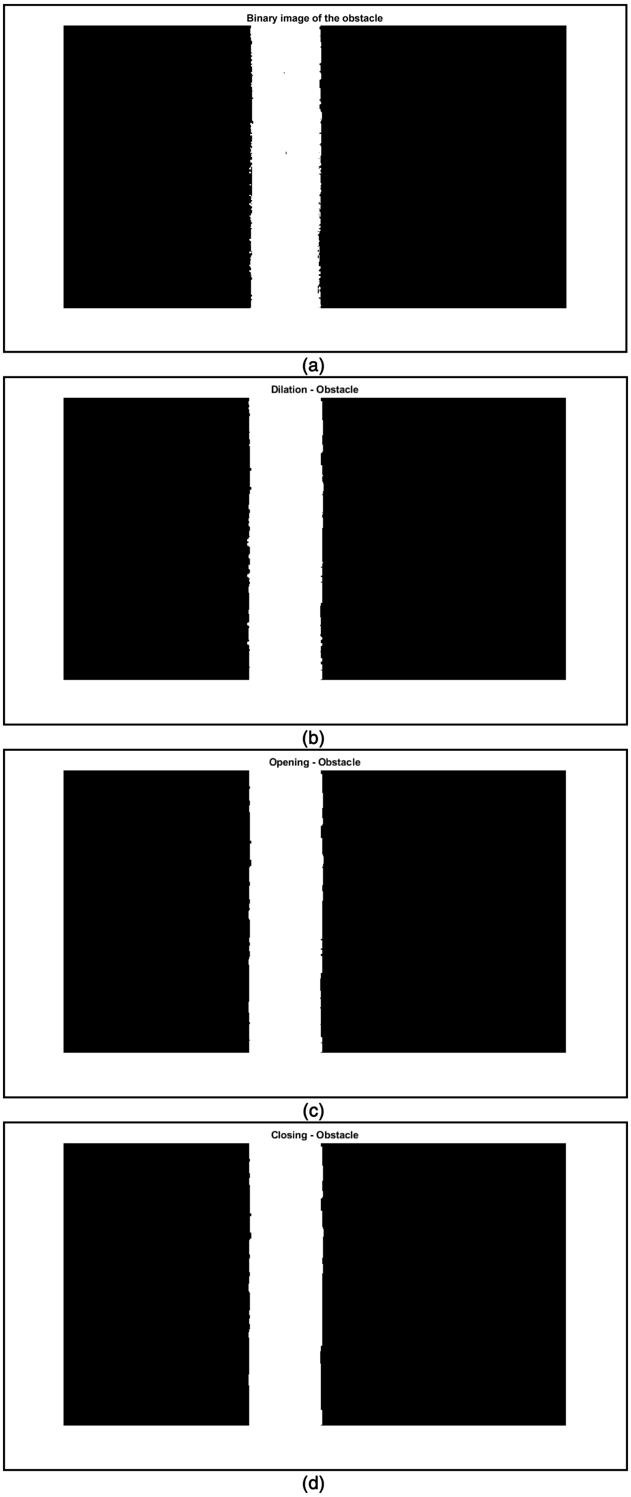


FIGURE 9. Sequence of binary and morphological operations for the possible obstacle extraction: (a) binary image; (b) dilation; (c) opening; (d) closing.

C. FINAL WALL EXTRACTION

Finally, the wall extraction is performed with a simple binary image subtraction operation [18]–[22]. The detected binary image of the obstacle is subtracted from the detected binary wall and obstacle image using the ordinary subtraction operation for two matrices of the same dimensions [21].

The extracted wall area is shown in Fig. 10. The wall area is perfectly extracted observing the original RGB image of the scene and the colored depth map of the original depth image. After this subtraction, the binary image is assigned appropriate coordinates from the captured point cloud to navigate the painting robot. Finally, the extracted wall surface is intended for the painting with the painting robot with the mounted spray gun.

The whole painting process is automated, and the robot's arm with the mounted painting equipment is navigated via control software using the information about the extracted useful wall surface. The painting process can be briefly summarized as follows. First, the robot is started at its home position. Then, a trajectory is generated to the scanning-recording position and sent to the joint drivers; once the robot is finished with the execution of this motion, the ZED camera data are recorded. The obtained depth image and point cloud data contain the scene of the wall in front of the robot along with its characteristics, i.e., the depth information and the coordinates about the recorded scene with potential obstacles. Once the point cloud data of the wall are obtained, the processing of the measurement is executed in the GUI. This image processing algorithm identifies the wall surface. Based on the location of the wall and the window, door and/or obstacles in 3D space, the painting trajectory is generated, which is composed of zigzag paths. These zigzag paths are constructed in such a way to cover all the free areas on the wall and omit the windows, doors and other obstacles. The painting paths are sent to the motor drivers, and the painting is performed using the spray gun. At the end of this process, the robot is actuated back to its home position.

Notably, the coordinates and dimensions of the image processing results for further robot painting operations are generated automatically via the ZED camera's built-in software [5]. According to the StereoLabs documentation, the relation between the image coordinates (2D) and the real coordinates (3D) for a given pixel (u, v), knowing the depth (z) at this point, the (x, y) coordinates in the real world are determined using the following formulas:

$$x = \frac{(u - c_x)\, z}{f_x} \tag{1}$$

$$y = \frac{(v - c_y)\, z}{f_y} \tag{2}$$

where c_x, c_y, f_x, f_y are the camera intrinsic parameters, that are accessed through the left camera of the ZED depth sensor [5]. f_x and f_y are the focal lengths in pixels along the x- and y-axes, respectively. c_x is the optical center along the x-axis defined in pixels (usually close to width/2), and c_y is the optical center along y-axis defined in pixels (usually close to height/2). The units in which the coordinates are expressed are defined by default in millimeters [5].

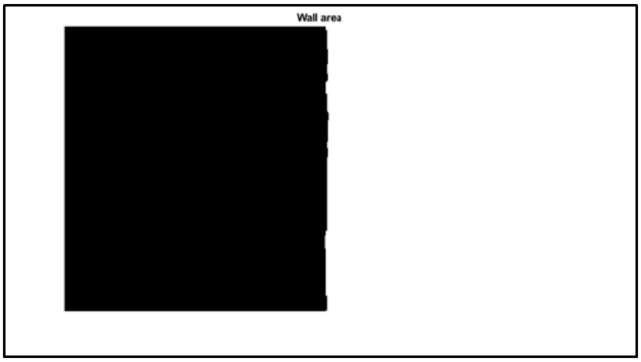


FIGURE 10. The extracted wall area that should be painted.

Finally, to better review the accuracy of the algorithm, the overlapped images of the detected obstacle with the original RGB image of the scene, and the overlapped image of the wall area with the original RGB of the scene are provided in Fig. 11 (a)-(b). In both images, the pink area represents the overlap of the extracted obstacle and the wall area. These pink surfaces are the useful surfaces intended for the painting (or for avoiding in the case of the obstacle) and serve to navigate the robotic arm. The green area is the surface not intended for painting. As can be observed, the overlap is considered perfect. These extracted pink surfaces are intended for robot movements with the spray gun. The painting process and the painting precision depend on the quality of the used gun and its spreading quality. Since the painting is performed with the spray gun, all surfaces not intended for painting are covered with a protective foil, as with any painting job with a compressor spray gun. Additionally, there is a possibility to paint the obstacle with a different color, since the obstacle area is also extracted. The obstacle painting procedure would be the same as the wall painting process.

Further, it should be noted, that the painting process is intended to be performed during daylight outdoors (if the building facades would be painted), or under indoor lighting (if the indoor environments would be painted). Since the ZED camera is a passive depth detector (it uses stereovision for depth detection, and it does not have a built-in infrared detector), it can operate under only good light conditions (daylight and indoor lighting) [5]. Additionally, the painting is planned to be performed in the light, similar to any other manual painting job. Finally, the speed of the robot's arm motion has no effect on the depth image capture performance, since image capturing is performed when the robot is not moving. Later, after the image processing algorithm is executed, the painting process starts.

Therefore, this algorithm works well although only the very simple operations requested for the project were used. Finally, the project was fulfilled, and the next step is the

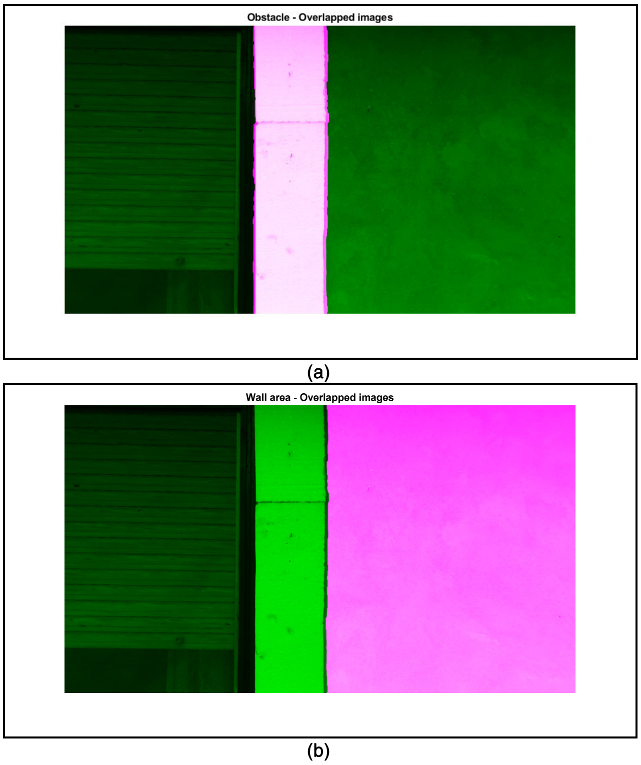


FIGURE 11. The overlap of the RGB image: (a) with the obstacle; (b) with the extracted wall area that should be painted.

testing with the painting equipment after the patenting process of the robot is finished.

In the remainder of the paper, a number of experiments and results are presented, as well as the verification of the algorithm itself.

V. EXPERIMENTS AND RESULTS

In this section, various experiments and results are introduced, and finally the verification results of the algorithm are presented.

A. TESTING RESULTS

During the experiments, the algorithm was tested on various wall-window examples that were in the scope of the project requirements. As previously mentioned, the request and goal are to separate the rectangular walls and windows and to extract and avoid the rectangular obstacles near the window area. Additionally, there is also an option to paint the obstacles with a different paint color. These obstacles are often considered decoration frames around the window and door surface, as shown in Fig. 7 (a), and homeowners often wish to paint them with a diverse color. The depth of these decorative frames is almost always thicker than 2 cm, and since the ZED camera can accurately distinguish the 2 cm depth, the criteria for obstacle detection is that the obstacles should be at least 2 cm thick and have a rectangular boundary, and this was one of the project's main requirements. Finally, the dead zone with a width of 12 cm on the left side of the images is avoided during robot movements. Thus, this surface is considered during the painting process.

The first columns in Figs. 12 and 13 show the original RGB images, the second columns show the images of the extracted wall areas that should be painted in binary form and the third columns show the overlaps of RGB images with the extracted wall areas intended for painting. This representation of the experimental results is chosen for a better view and clarity. The testing was conducted under different lighting conditions and on different terrains, in some cases on extremely uneven ground surfaces. Additionally, sometimes the wall surface and the window frame were of poor quality with uneven depth distances in the direction of the Z coordinate from the depth camera. This inequality of the ground is one of the most substantial problems during the process of image capturing and painting because the whole system is affected by the aforementioned problems.

In the first example in Fig. 12, the lighting conditions are satisfactory and there are 2 rectangular obstacles on the wall. The obstacle with a drawing is approximately 5 cm thick; the other obstacle is 2 cm thick. The second image shows the extracted wall area. The obstacles are successfully avoided, and the third image shows the overlapped original image with the image of the extracted wall. The success of the wall extraction algorithm is obvious, since the area intended for painting is clearly separated from the window and from the obstacles. The second example presents a case where the wall has a window and a door on the left side. The depth difference between the wall and the window is approximately 6 cm. Additionally, a strong influence of the sunrays on the wall and window can be observed. These sunrays can cause unfavorable reflection, and they can blind the depth camera if the reflection of rays directly enters the sensor. However, in this example, the reflection did not affect the depth sensor and the test was successful. Clearly, the wall is distinguished from the window and door very well. The third example presents a wall area with a door in the middle. An old building is captured in this case. The depth difference is approximately 9-10 cm, and it varies. The depth image is captured in the shadow of trees. Again, the wall extraction was successful with high accuracy. Naturally, the light switch will be covered during the painting process. The fourth example presents a wall area near a metal door. The depth difference in this case is approximately 7 cm. The door region is under a roof, and the wall region is next to the area covered by the roof. Additionally, sunlight is reflecting in the door window.

The second image shows the extracted wall surface, and the third image shows the comparison of the original image and wall extracted image by overlapping. The extraction is considered perfect. The next example presents a very old and ruined building. One of the authors is captured in the image since the ground around the building was very uneven and there was a need for additional holding of the equipment by a person. The difference between the wall and the door window is approximately 4 cm.

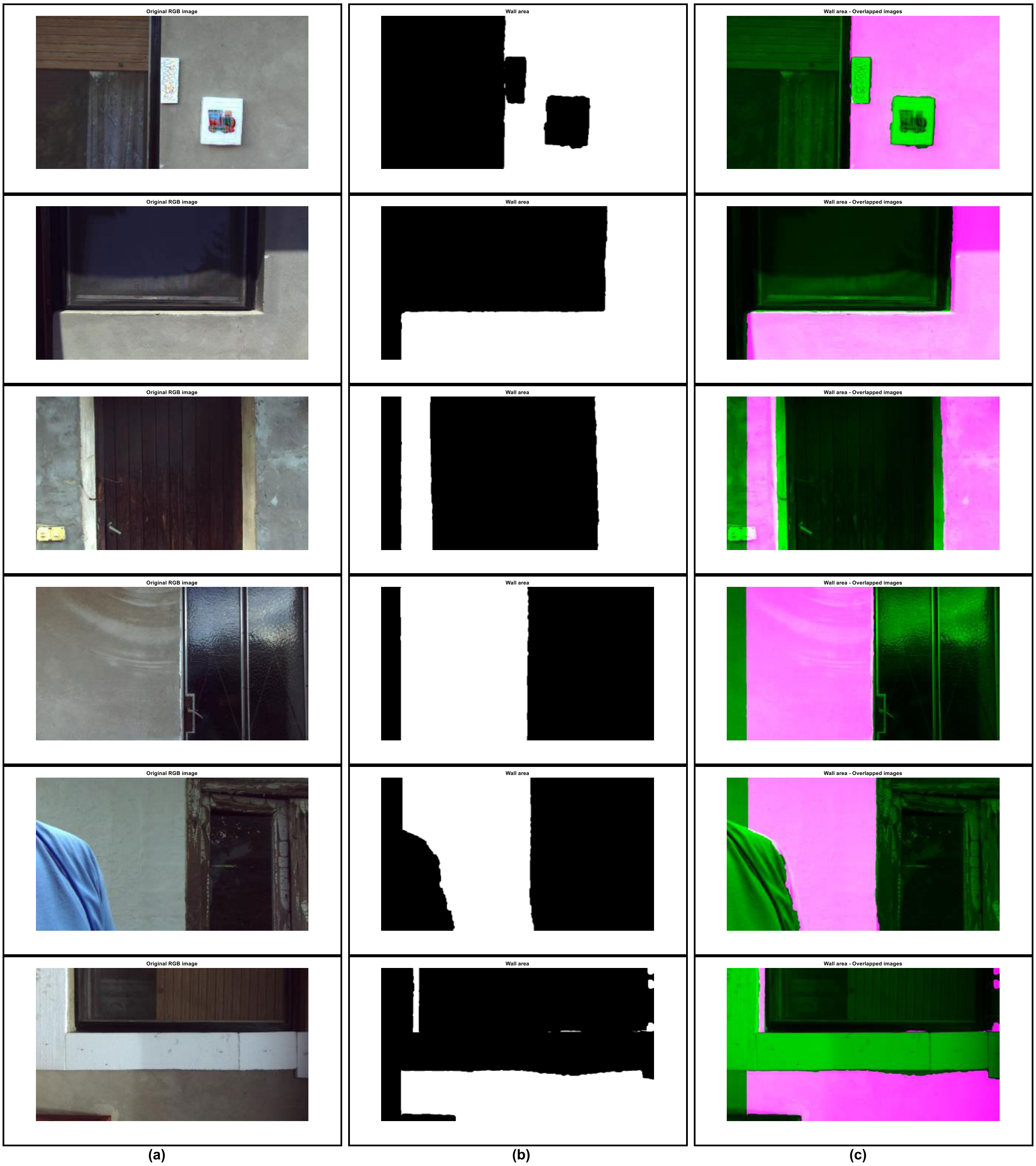


FIGURE 12. Testing examples: (a) original RGB image; (b) extracted wall area intended for painting; (c) overlapped RGB image with the extracted wall area.

As seen in the second and third images in the fifth example of Fig. 12, the algorithm successfully extracted the wall area by eliminating the body of the person since he was considered an obstacle in this test. The final example presents a wall with an obstacle and a part of the terrace fence in the left bottom part of the image in red color. The obstacle is 2 cm thick. The image is captured in the evening under poor lighting conditions. Additionally, in this example, the camera was rotated 90° vertically with a small slant. The second image shows the extracted wall area.

Notably, the obstacle and a small part of the fence are avoided successfully. However, there are small artifacts, including remaining parts of the window's edge. The reason for this is the small slant of the camera. This slant causes the sensors not to be at the same distance from the wall, and this situation can produce artifacts and false values for the Z coordinates. By eliminating the slant and the poor lighting conditions, it should be possible to overcome these problems.

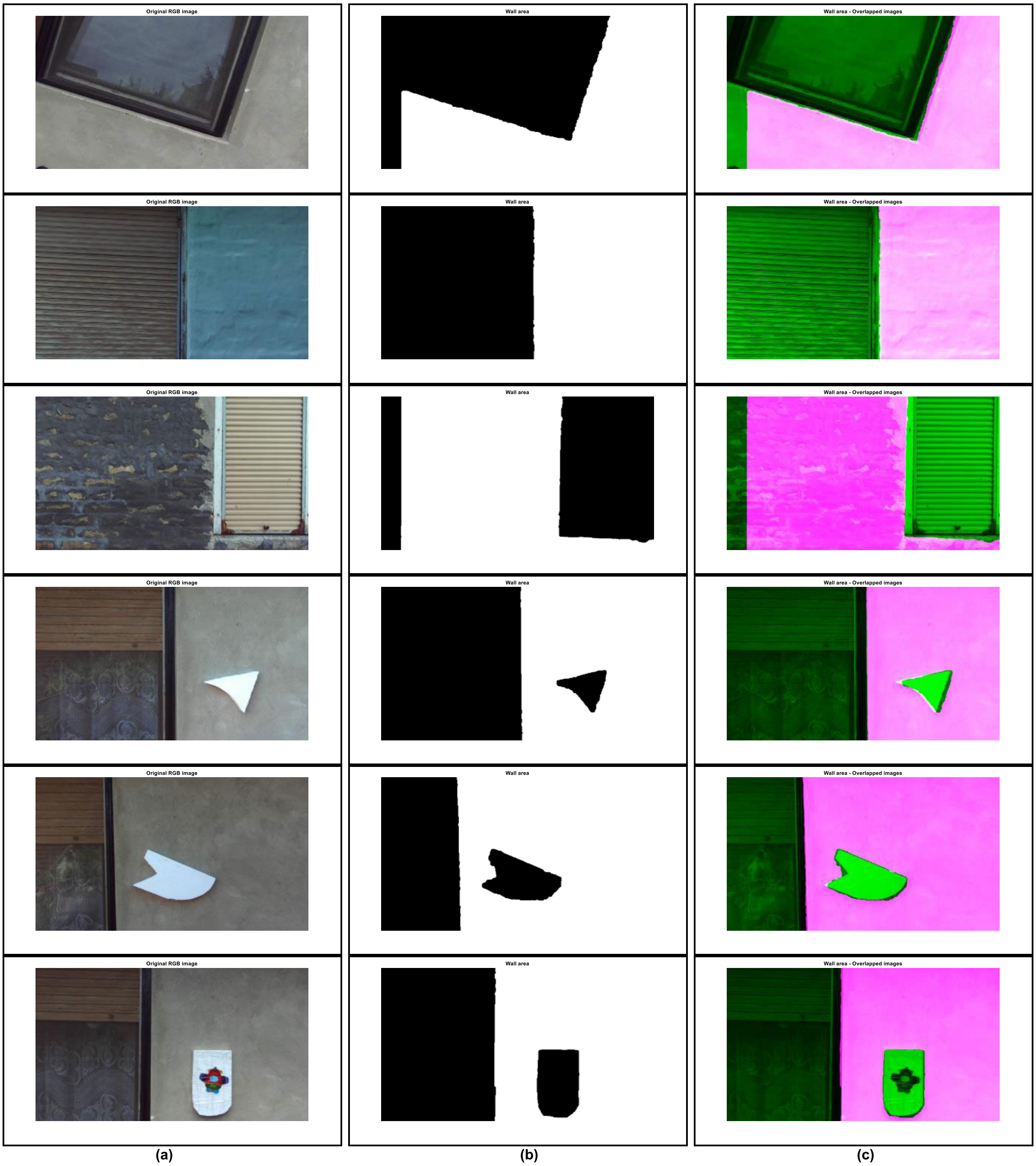

Original RGB image
Wall area
Wall area - Overlapped images
Original RGB image
Wall area
Wall area - Overlapped images
Original RGB image
Wall area
Wall area - Overlapped images
Original RGB image
Wall area
Wall area - Overlapped images
Original RGB image
Wall area
Wall area - Overlapped images
Original RGB image
Wall area
Wall area - Overlapped images
(a)
(b)
(c)

FIGURE 13. **Verification examples: (a) original RGB image; (b) extracted wall area intended for painting; (c) overlapped RGB image with the extracted wall area.**

B. VERIFICATION OF THE ALGORITHM

Fig. 13 presents the verification of the wall extraction algorithm. To verify the experiments and results, a series of examples are presented. Several examples show instances that are not considered by the project, but these examples emphasize the power of this simple image processing method. These examples include undefined obstacle shapes and a triangular window. Additionally, a set of examples is shown where the difference in depth between the wall and the window varies along their boundary, and in addition, images were captured on uneven surfaces. The first example in Fig. 13 shows a triangular window. The example was made by rotating the depth camera. Additionally, the depth image was captured in the evening, and the shadow from a fir tree can be noticed. The second image shows the extracted wall area followed by the comparison of the images in the first and second column. The wall extraction was performed perfectly. The second example presents the wall-window boundary with varying Z distance values. This was caused by the imperfections of the wall and the window frame. The distance varies between 2 and 4 cm. Additionally, the ground near the building is very uneven, and there are also shadows from trees.

As shown in the second and third images, the wall extraction was performed extraordinarily well with high accuracy in relation to the shooting conditions. The third example shows an imperfect wall-window again. In this case, not only the depth difference along the boundary varies but also the depth difference on the wall surface varies. The depth variation along the wall-window boundary is 3-4 cm, and the depth variation on the wall surface is approximately 1 cm. Additionally, the ground is also uneven in this example. As shown in the second and third columns, the wall area is extracted again with high precision relative to the depth image conditions. The next three examples show the behavior of the algorithm in the case of undefined shapes. Although the detection of such shapes was not required for the project, these examples are presented to better verify the algorithm. The fourth example shows a pointed shape similar to a triangle. This image was captured in the evening. The shape is approximately 4 cm thick. The obstacle is successfully avoided by the wall extraction algorithm, although slight problems may be observed in the detection of the shape spikes themselves. This kind of obstacle detection is considered successful. The next example shows an undefined shape with a recess. The obstacle is approximately 5 cm thick. This image was captured under satisfactory lighting conditions. Again, the obstacle detection and avoidance are successfully accomplished using the developed algorithm. As in the previous example, the problems are the spikes of the shape since they are very thin at their ends.

The last example shows a semirectangular-semicircular obstacle of 5 cm thickness. The image capturing was performed under satisfactory lighting conditions. The wall area is detected and extracted almost perfectly, and the obstacle is avoided.

As is shown in the algorithm verification, the geometry of the obstacles is not relevant to the algorithm since the procedure is based on the Z depth distance from the depth camera. These verification results prove that the main conditions for successful wall detection are the accurate and precise positioning of the depth camera from the wall surface and the proper lighting conditions. Naturally, the strong and direct influence of sunrays and other harmful influences must be avoided, as for any camera or recording process.

Furthermore, the structural elements in the morphological operations were determined empirically during the development. These SEs are dimension-dependent elements, which means they can be easily adjusted depending on the size of the objects in the image, and the size of the objects in the image depends on the type of the depth camera as well as on the distance of the camera from the wall. These modifications are very easy to make during the reproduction of this algorithm. Additionally, the range for the detection of the wall surface and the obstacle can vary depending on the distance of the depth camera from the wall and on the lens quality of the depth camera. The high-quality lenses introduce less scatter when mapping the depth of the wall surface; therefore, the wall extraction can be performed with high accuracy. In the case of a poor quality depth camera with poor quality lenses, this separation cannot be successfully performed because the scattering of Z values is very high, and it can lead to mixing of the depth values of the surfaces that represent the wall, window and obstacles. In this situation, the separation cannot be established. From this perspective, the StereoLabs ZED depth camera met all the expectations for this project.

VI. CONCLUSION

Herein, we introduced both the working principles and the properties of an algorithm for detection and extraction of useful surfaces and objects in a painting robot application. The main steps of the wall surface extraction process were presented, i.e., distance ranging, binarization, and a sequence of morphological operations. The goal of this industrial research project was to build a simple robot vision system with known and reliable procedures to ensure the reliability of the robot's working process. Appropriate experiments were conducted, on certain wall-window surfaces. All experiments were performed successfully. During the development of the robot vision system, all of the project requirements and instructions were followed. As a result, the project's goal was fully achieved.

FUTURE WORKS

In the future, a special painting kit with a spray gun should be mounted on the robot, and the robot should be tested in real painting applications. Additionally, possibilities for improvement exist by considering a visual system with a depth sensor that has better performance or by adapting the robot's control system to the technological process during practical use. The start of a production series of the painting robot is also planned. Finally, all refinements are client-dependent considerations, and the development will be accomplished according to the requirements of the customer.

REFERENCES

[1] V. Tadic, A. Odry, E. Burkus, I. Kecskes, Z. Kiraly, M. Klincsik, Z. Sari, Z. Vizvari, A. Toth, and P. Odry, "Painting path planning for a painting robot with a RealSense depth sensor," *Appl. Sci.*, vol. 11, no. 4, p. 1467, Feb. 2021, doi: 10.3390/app11041467.

[2] V. Tadic, A. Odry, E. Burkus, I. Kecskes, Z. Kiraly, and P. Odry, "Edge-preserving filtering and fuzzy image enhancement in depth images captured by realsense cameras in robotic applications," *Adv. Electr. Comput. Eng.*, vol. 20, no. 3, pp. 83–92, 2020, doi: 10.4316/AECE.2020.03010.

[3] V. Tadic, E. Burkus, A. Odry, I. Kecskes, Z. Kiraly, and P. Odry, "Effects of the post-processing on depth value accuracy of the images captured by RealSense cameras," *Contemp. Eng. Sci.*, vol. 13, no. 1, pp. 149–156, 2020, doi: 10.12988/ces.2020.91454.

[4] V. Tadic, A. Odry, I. Kecskes, E. Burkus, Z. Kiraly, and P. Odry, "Application of Intel realsense cameras for depth image generation in robotics," *WSEAS Trans. Comput.*, vol. 18, pp. 107–112, 2019, Art. no. #14.

[5] Accessed: Apr. 10, 2021. [Online]. Available: https://www.stereolabs.com/zed/

[6] E. L. Ortiz, V. E. Cabrera, and M. L. Goncalves, "Depth data error modeling of the ZED 3D vision sensor from stereolabs," *Electron. Lett. Comput. Vis. Image Anal.*, vol. 17, no. 1, pp. 1–15, 2018, doi: 10.5565/rev/elcvia.1084.

[7] P. Burdziakowski, "Low cost real time UAV stereo photogrammetry modelling technique—Accuracy considerations," *E3S Web Conf.*, vol. 63, pp. 1–5, Nov. 2018, doi: 10.1051/e3sconf/20186300020.

[8] M. Senthilvela, K. R. Somanb, and K. Varghesea, "Comparison of handheld devices for 3D reconstruction in construction," in *Proc. Int. Symp. Automat. Robot. Construct.*, 2017, pp. 1–8.

[9] J. Teizer, M. Wolf, O. Golovina, M. Perschewski, M. Propach, M. Neges, and M. König, "Internet of Things (IoT) for integrating environmental and localization data in Building Information Modeling (BIM)," in *Proc. 34th Int. Symp. Automat. Robot. Construct., (ISARC)*, 2017, pp. 1–7.

[10] H. Ren, N. Gao, and Z. Su, "Color-guided depth recovery based on ZED," in *Proc. Int. Conf. Comput. Technol., Electron. Commun. (ICCTEC)*, Dec. 2017, pp. 733–737, doi: 10.1109/ICCTEC.2017.00162.

[11] E. Cabrera, L. Ortiz, B. Silva, E. Clua, and L. Gonçalves, "A versatile method for depth data error estimation in RGB-D sensors," *Sensors*, vol. 18, no. 9, p. 3122, Sep. 2018, doi: 10.3390/s18093122.

[12] J. Cunha, E. Pedrosa, C. Cruz, J. R. Antonio Neves, and N. Lau, "Using a depth camera for indoor robot localization and navigation," in *Proc. Conf. RGB-D, Adv. Reasoning With Depth Cameras Workshop, Robot. Sci. Syst. Conf. (RSS)*, 2011, pp. 1–6.

[13] F. Flacco, T. Kroger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 338–345.

[14] K. Jawale, R. Kumar, and V. Kale, "Design and development of a wall painting robot for the houses wall," *Int. J. Multidisciplinary Res. Develop.*, vol. 2, no. 4, pp. 397–401, 2015.

[15] W. I. Muzan, T. Faisal, H. M. A. A. Al-Assadi, and M. Iwan, "Implementation of industrial robot for painting applications," in *Proc. Int. Symp. Robot. Intell. Sensors*, 2012, pp. 1329–1335, doi: 10.1016/j.proeng.2012.07.318.

[16] P. Keerthanaa, K. Jeevitha, V. Navina, G. Indira, and S. Jayamani, "Automatic wall painting robot," *Int. J. Innov. Res. Sci., Eng. Technol.*, vol. 2, no. 7, 2013.

[17] M. Vincze, A. Pichler, G. Biegelbauer, K. Häusler, H. Andersen, O. Madsen, and M. Kristiansen, "Automatic robotic spray painting of low volume high variant parts," in *Proc. 33rd Int. Symp. Robot.*, Oct. 2002, p. 11.

[18] A. P. Borhade, A. N. Patil, and S. K. Patil, "Automatic wall painting robot," *Int. J. Sci. Eng. Manage. Res.*, vol. 7, no. 5, 2019.

[19] V. Tadic, M. Popovic, and P. Odry, "Fuzzified Gabor filter for license plate detection," *Eng. Appl. Artif. Intell.*, vol. 48, pp. 40–58, Feb. 2016, doi: 10.1016/j.engappai.2015.09.009.

[20] V. Tadic, Z. Kiraly, P. Odry, Z. Trpovski, T. Loncar-Turukalo, and U. of Novi Sad, "Comparison of Gabor filter bank and fuzzified Gabor filter for license plate detection," *Acta Polytechnica Hungarica*, vol. 17, no. 1, pp. 61–81, 2020, doi: 10.12700/APH.17.1.2020.1.4.

[21] R. C. Gonzales, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*, 2nd ed. Knoxville, TN, USA: Gatesmark, 2009.

[22] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, 3rd ed. Pearson, NJ, USA: Prentice-Hall, 2008.

[23] A. Saxena, S. H. Chung, and A. Y. Ng, "3-D depth reconstruction from a single still image," *Int. J. Comput. Vis.*, vol. 76, no. 1, pp. 53–69, Dec. 2007.

[24] H. J. Hemmat, E. Bondarev, and H. N. P. de With, "Real-time planar segmentation of depth images: From 3D edges to segmented planes," Dept. Elect. Eng., Eindhoven Univ. Technol., Eindhoven, The Netherlands, 2015.

[25] V. Sterzentsenko, A. Karakottas, A. Papachristou, N. Zioulis, A. Doumanoglou, D. Zarpalas, and P. Daras. *A Low-Cost, Flexible and Portable Volumetric Capturing System*. Accessed: Apr. 25, 2021. [Online]. Available: https://vcl.iti.gr

[26] N. Carey, J. Werfel, and R. Nagpal, "Fast, accurate, small-scale 3D scene capture using a low-cost depth sensor," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 1268–1276, doi: 10.1109/WACV.2017.146.

[27] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Robot. Auto. Syst.*, vol. 56, no. 11, pp. 927–941, Nov. 2008, doi: 10.1016/j.robot.2008.08.005.

[28] T. Schwarze and M. Lauer, "Wall estimation from StereoVision in urban street canyons," in *Proc. 10th Int. Conf. Inform. Control, Automat. Robot.*, 2013, pp. 83–90, doi: 10.5220/0004484600830090.

[29] S. Zhou, F. Kang, W. Li, J. Kan, Y. Zheng, and G. He, "Extracting diameter at breast height with a handheld mobile LiDAR system in an outdoor environment," *Sensors*, vol. 19, no. 14, p. 3212, Jul. 2019, doi: 10.3390/s19143212.

[30] C. Garnica, F. Boochs, and M. Twardochlib, "A new approach to edge-preserving smoothing for edge extraction and image segmentation," *Int. Arch. Photogramm. Remote Sens.*, vol. 33, Jul. 2000.

[31] S. Reich, F. Wörgötter, and B. Dellen, "A real-time edge-preserving denoising filter," in *Proc. 13th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2018, pp. 85–94, doi: 10.5220/0006509000850094.

[32] R. Abiko and M. Ikehara, "Fast edge preserving 2D smoothing filter using indicator function," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1877–1881, doi: 10.1109/ICASSP.2019.8683544.

[33] J. Choi, H. Park, and D. Seo, "Pansharpening using guided filtering to improve the spatial clarity of VHR satellite imagery," *Remote Sens.*, vol. 11, no. 6, p. 633, 2019, doi: 10.3390/rs11060633.

[34] N. Iqbal, S. Ali, I. Khan, and B. Lee, "Adaptive edge preserving weighted mean filter for removing random-valued impulse noise," *Symmetry*, vol. 11, no. 3, p. 395, Mar. 2019, doi: 10.3390/sym11030395.

[35] P. M. Khandekar, S. S. Chiddarwar, and A. Jha, "Programming of an industrial robot using demonstrations and soft computing techniques," *J. Sci. Ind. Res.*, vol. 77, pp. 156–163, Mar. 2018.

[36] D. Ristic-Durrant, S. M. Grigorescu, A. Graser, Z. Cojbasic, and V. Nikolic, "Robust stereo-vision based 3D object reconstruction for the assistive robot FRIEND," *Adv. Electr. Comput. Eng.*, vol. 11, no. 4, pp. 15–22, 2011, doi: 10.4316/AECE.2011.04003.

[37] X. Ning, G. Tian, and Y. Wang, "Top-down approach to the automatic extraction of individual trees from scanned scene point cloud data," *Adv. Electr. Comput. Eng.*, vol. 19, no. 3, pp. 11–18, 2019.

[38] E. Asadi, B. Li, and I.-M. Chen, "Pictobot: A cooperative painting robot for interior finishing of industrial developments," *IEEE Robot. Autom. Mag.*, vol. 25, no. 2, pp. 82–94, Jun. 2018, doi: 10.1109/MRA.2018.2816972.

[39] I. M. Chen, E. Asadi, J. Nie, R. J. Yan, W. C. Law, E. Kayacan, S. H. Yeo, K. H. Low, G. Seet, and R. Tiong, "Innovations in infrastructure service robots," in *ROMANSY 21—Robot Design, Dynamics and Control* (CISM International Centre for Mechanical Sciences), vol. 569, V. Parenti-Castelli and W. Schiehlen, Eds. Udine, Italy, 2016, doi: 10.1007/978-3-319-33714-2_1.

[40] L. Somlyai and Z. Vámossy, "SLAM algorithm for mobile robot localization with RGB-D camera," in *Fluids, Heat and Mass Transfer, Mechanical and Civil Engineering, WSEAS*. Budapest, Hungary, 2015.

[41] G. Kertesz, S. Szenasi, and Z. Vamossy, "Multi-directional image projections with fixed resolution for object matching," *Acta Polytechnica Hungarica*, vol. 15, no. 2, pp. 211–229, 2018, doi: 10.12700/aph.15.1.2018.2.11.

[42] T. Haidegger, S. G. Virk, C. Herman, R. Bostelman, P. Galambos, G. Györök, and J. I. Rudas, "Industrial and medical cyber-physical systems: Tackling user requirements and challenges in robotics," *Recent Advances in Intelligent Engineering*, vol. 14. Cham, Switzerland: Springer, 2020, doi: 10.1007/978-3-030-14350-3_13.

[43] R. Annamária and A. R. Várkonyi-Kóczy, "Soft computing based point correspondence matching for automatic 3D reconstruction," *Acta Polytechnica Hungarica*, vol. 2, no. 1, pp. 33–44, 2005.

[44] E. Burkus and P. Odry, "Autonomous hexapod walker robot 'Szabad(ka),'" *Acta Polytechnica Hungarica*, vol. 5, no. 1, pp. 69–85, 2008.

[45] R. Szabo and A. Gontean, "Robotic arm control algorithm based on stereo vision using RoboRealm vision," *Adv. Electr. Comput. Eng.*, vol. 15, no. 2, pp. 65–74, 2015, doi: 10.4316/aece.2015.02009.

[46] I. Kecskés, E. Burkus, F. Bazsó, and P. Odry, "Model validation of a hexapod Walker robot," *Robotica*, vol. 35, no. 2, pp. 419–462, Feb. 2017, doi: 10.1017/S0263574715000673.

[47] A. Koubaa, *Robot Operating System (ROS)*. Cham, Switzerland: Springer, 2016, doi: 10.1007/978-3-319-26054-9.

[48] J. Kramer and M. Scheutz, "Development environments for autonomous mobile robots: A survey," *Auto. Robots*, vol. 22, no. 2, pp. 101–132, Jan. 2007.

[49] A. Martinez and E. Fernández, *Learning ROS for Robotics Programming*. Birmingham, U.K.: Packt Publishing, 2013.

[50] J. Kerr and K. Nickels, "Robot operating systems: Bridging the gap between human and robot," in *Proc. 44th Southeastern Symp. Syst. Theory (SSST)*, Mar. 2012, pp. 99–104.

[51] M. Quigley, B. Gerkey, and W. D. Smart, *Programming Robots With ROS*. Sebastopol, CA, USA: O'Reilly Media, Inc., 2015.

[52] V. Tadic, A. Odry, A. Toth, Z. Vizvari, and P. Odry, "Fuzzified circular Gabor filter for circular and near-circular object detection," *IEEE Access*, vol. 8, pp. 96706–96713, 2020, doi: 10.1109/ACCESS.2020.2995553.

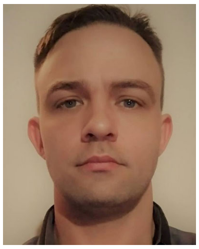
[53] J. C. Jáuregui, J. R. Reséndiz, S. Thenozhi, T. Szalay, Á. Jacsó, and M. Takács, "Frequency and time-frequency analysis of cutting force and vibration signals for tool condition monitoring," *IEEE Access*, vol. 6, pp. 6400–6410, 2018, doi: 10.1109/ACCESS.2018.2797003.

[54] E. A. Padilla-Garcia, A. Rodriguez-Angeles, J. R. Resendiz, and C. A. Cruz-Villar, "Concurrent optimization for selection and control of AC servomotors on the powertrain of industrial robots," *IEEE Access*, vol. 6, pp. 27923–27938, 2018, doi: 10.1109/ACCESS.2018.2840537.

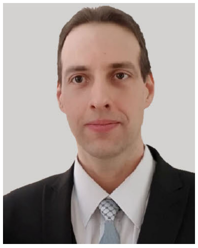
[55] M.-A. Martinez-Prado, J. Rodriguez-Resendiz, R.-A. Gomez-Loenzo, G. Herrera-Ruiz, and L.-A. Franco-Gasca, "An FPGA-based open architecture industrial robot controller," *IEEE Access*, vol. 6, pp. 13407–13417, 2018, doi: 10.1109/ACCESS.2018.2797803.

VLADIMIR TADIĆ received the Dipl.Ing. (Graduate Engineer), M.Sc., and Ph.D. degrees in electrical engineering from the Faculty of Technical Sciences, University of Novi Sad, in 2004, 2009, and 2018, respectively. He is currently a Professor and a Researcher with the Department of Control Engineering and Information Technology, University of Dunaújváros. His research interests include image processing, computer vision, signal processing, speech processing, telecommunications, electronics, robotics, and soft computing methods.

ÁKOS ODRY received the B.Sc. and M.Sc. degrees in electrical engineering from Budapest University of Technology and Economics, in 2012 and 2015, respectively, and the Ph.D. degree from the Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University, in 2020. He is currently a College Associate Professor with the Technical Department, Faculty of Engineering, University of Szeged. His research interests include nonlinear control systems, robotics, engineering education, soft computing, and adaptive control methods.



ERVIN BURKUS received the B.Sc. degree in computer engineering from Dennis Gabor College, in 2006, and the M.Sc. degree in computer engineering from the Technical Faculty Mihajlo Pupin, University of Novi Sad, in 2011. He is currently pursuing the Ph.D. degree with the Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University. He is currently a Junior Assistant Professor and a Researcher with the Department of Computer Systems and Control Engineering, University of Dunaújváros. His research interests include mechanical structural optimization, robotics, embedded systems, and mechanics.



ISTVÁN KECSKÉS received the M.Sc. degree in informatics from Belgrade—Singidunum University, in 2012, and the Ph.D. degree from the Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University, in 2018. He is currently a Research Engineer with the Department of Control Engineering and Information Technology, University of Dunaújváros. His research interests include digital signal processing, soft computing, machine learning, deep learning, robotics, sensors, bio-signals, and heart diagnosis.

ZOLTÁN KIRÁLY graduated as a Technical Teacher from Dunaújváros College, in 1992, and a Computer Science Teacher from Eötvös Lóránd University, in 1995. He received the Ph.D. degree from Budapest University of Technology, in 2010. He is currently an Associate Professor and a Researcher with the Department of Software Development and Application, University of Dunaújváros. His research interests include robotics, computer vision, signal processing, speech processing, telecommunications, IT security, and soft computing methods.

ZOLTÁN VÍZVÁRI received the Dipl.Ing. degree in environmental engineering from the Faculty of Engineering and Information Technology, University of Pécs, Hungary, in 2006, and the M.Sc. degree in environmental engineering from Budapest University of Technology and Economics, Hungary, in 2009. He is currently a Graduate Student with the Doctoral School of Applied Informatics and Applied Mathematics and a Researcher with the Faculty of Engineering and Information Technology in Pécs. His research interests include measurement sciences and associated mathematical physical modeling, data processing methods, differential equation solvers, and associated inverse problems solving procedures. The results of his research activity are applied in field of biotechnology and medicine in Hungary.

ATTILA TÓTH received the master's degree in biology from the University of Pécs, in 2001, and the Ph.D. degree from the School of Medicine, Institute of Behavioural Sciences, University of Pécs, in 2019. After that, he was an Intern at the School of Medicine, Institute of Behavioural Sciences, University of Pécs. Then he has been an Assistant Professor at the School of Medicine, Physiology Institute, University of Pécs, since 2008. He is currently working as a Senior Researcher with the Metabolic Regulation and Bioimpedance Research Group.

PÉTER ODRY (Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering from the University of Belgrade, in 1986 and 1992, respectively. He is currently a Professor with the Department of Control Engineering and Information Technology, University of Dunaújváros. His research interests include impedance tomography, legged robotic systems, computer vision, robust control techniques, and soft computing methods.