COMPUTER DESIGN SYSTEMS. THEORY AND PRACTICE

Vol. 6, No. 3, 2024

Igor Nevliudov[1], Vladyslav Yevsieiev[2], Svitlana Maksymova[3], Olena Chala[4], Dmytro Gurin[5]

[1] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine, E-mail: igor.nevliudov@nure.ua, ORCID 0000-0002-9837-2309
[2] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine, E-mail: vladyslav.yevsieiev@nure.ua, ORCID 0000-0002-2590-7085
[3] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv Ukraine,, E-mail: svitlana.milyutina@nure.ua, ORCID 0000-0002-1375-9337
[4] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine,E-mail: olena.chala@nure.ua, ORCID 0000-0003-2454-3774
[5] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, Ukraine, E-mail: dmytro.gurin@nure.ua, ORCID 0000-0002-2272-5227

DEVELOPMENT OF A PROGRAM FOR MODELING AND SIMULATING A COLLABORATIVE ROBOT WORKSPACE

Received: November 12, 2024 / Revised: November 20, 2024 / Accepted: November 30, 2024

https://doi.org/10.23939/cds2024.03.031

Abstract. The article presents the software development for modeling and simulating the workspace of a collaborative robot taking into account the presence of people. This is an important step in creating safe and efficient robotic systems within Industry 5.0 concept. The problem is posed by the need to ensure safety during the interaction of the robot with the operator, which is relevant for modern production processes with high human participation.

The purpose of the study is to create a tool for dynamic modeling of the environment, capable of detecting people in the robot's workspace and avoiding potential collisions. In the process of the study, computer vision methods and image processing algorithms were applied to determine the location of a person in three-dimensional space, using libraries such as PyBullet and OpenCV.

The main results of the work are experimental data confirming the effectiveness of the developed system in detecting objects and preventing collisions. The novelty of the research lies in the application of a potential field model that combines the repulsive force from a person and the gravity force to the target point, which allows adaptively adjusting the robot's trajectory. The practical significance of the work lies in increasing the safety and efficiency of collaborative robots in industrial conditions, which helps reduce risks for the operator.

The scope of further research involves optimizing the algorithm for detecting people, taking into account changes in the environment, in particular, illumination, as well as the introduction of adaptive thresholds for object detection.

Keywords: Collaborative Robot, Modeling, Simulation, Safety, Industry 5.0, People Detection, Computer Vision, Potential Field.

Introduction

In modern manufacturing, Industry 5.0 concept focuses on the robotic systems integration and human operators in a shared workspace. This fact implies the need for new solutions to ensure the safety, efficiency and adaptability of robotic systems. In the context of collaborative robots that work side by side with humans, there is a need to develop software for accurate modeling and simulation of their workspace. This makes it possible to create safe and adaptive systems that can take into account the presence of a person and reduce the risks of accidental collisions or other unwanted incidents. The development of such simulation programs is relevant, as they allow for a preliminary assessment of the behavior of a robotic system in interaction with humans, which is critically important in industries where automation faces a high level of human participation, such as in healthcare institutions, the food industry or on production lines with high requirements for process flexibility [1]–[3].

Industry 5.0 focuses on harmonizing the relationship between humans and machines while ensuring sustainability and inclusiveness, which reinforces the relevance of such research in the context of the humanization of technology. Taking into account the presence of people when modeling collaborative robots is critical not only to reduce accidents, but also to ensure the convenience and efficiency of operators, which in turn contributes to increasing productivity and motivation of personnel. Thanks to simulations, it is possible to ensure that robotic systems comply with the latest safety standards and make the process of implementing robots smoother and more predictable. In addition, such development allows for the optimization of work processes, taking into account the behavioral characteristics of collaborative robots and people, which contributes to the construction of more adaptive, flexible and reliable production environments.

Problem Statement

The task of this research is to solve the problem of safety and effective interaction between a collaborative robot and people in a shared workspace. Modern production processes, which are focused on the integration of robots with operators, require new approaches to detect the presence of a person in the robot’s work area and dynamically adjust its trajectory to avoid collisions. It is important to develop a methodology and algorithms that allow identifying people in the robot's workspace and responding to changes in their position in real time. The main task is to model and simulate interaction, which takes into account not only static objects, but also moving obstacles presented by people. In order to ensure the safe operation of the robot, it is necessary to use computer vision algorithms for object detection and image processing methods for real-time space control. An additional task is to develop a potential field model that combines the gravity field to the target point and the repulsive field from the person, allowing the robot to automatically adjust its trajectory according to changes in the environment. This also requires the creation of an algorithm that adaptively adjusts safety parameters depending on the person’s behavior, speed, and direction of movement.

Review of Modern Information Sources on the Subject of the Paper

The Industry 5.0 concept is much more “human” compared to Industry 4.0. Here, human and robot collaboration appears. However, this leads to increased requirements for human safety. One of the key tasks in ensuring safety is identifying a human in the robot’s work area. Naturally, many scientific papers are devoted to this problem. Let us consider only small part of the recent works on this topic.

Let us begin with the work [4]. There is noted that human-robot collaboration plays a pivotal role in today’s industry by supporting increasingly customized product development. This approach allows to use strengths of robots and humans. Pairing them it became possible to achieve goals more quickly and effectively

In the article [5] the authors consider the problem collision-free human-robot collaboration. They propose a context awareness-based collision-free human-robot collaboration system. And this system allows to plan robotic paths in accordance with the task to reach the goal but avoid collisions with the humans.

The paper [6] note that it is necessary to create collision avoidance control techniques that can improve operator safety and robot flexibility. Such collisions may occur between the robot and humans or with objects inadvertently left in robot workspace. Authors propose to use algorithms already developed by the authors for planar manipulators and to implement some new contributions.

Scientists in [7] present a robot layout that can perform assembly tasks. They propose to use two Microsoft Kinect v2 in order to track a human. There, the robot and operator are in the same the workspace. And there is implemented a real time collision avoidance algorithm that is used to plan robot’s trajectory. Their layout avoids any collision with a human.

Bi, and co-authors in [8] note that if a collaborative robot is used in manufacturing environment it is extremely important to assure humans' safety. Usually, their workload or moving mass is usually large enough to hurt human. Researchers detected some technological bottlenecks. Among them we can distinguish acquiring, processing, and fusing diversified data for risk classification. So, it is necessary to use some sensors [8], [9] and to proceed data from them.

Thus, we see that scientific works consider various solutions to the problem of ensuring human safety. However, they cannot fully guarantee the avoidance of collisions with humans. Further in this paper, we propose our approach to solving this problem.

Objectives and Problems of Research

The aim of this research is to develop an effective program for modeling and simulating the workspace of a collaborative robot taking into account the presence of people to improve the safety and productivity of collaborative work processes. The main problem is the need to create a dynamic model capable of identifying a person in real time and preventing collisions, which is critically important in modern production environments where robots work side by side with people. The challenge is to ensure that the system responds adequately to the presence of a person and to provide adaptive changes in the robot trajectory depending on the operator's actions. The research is also aimed at implementing computer vision methods for accurate detection of people and assessment of possible collision risks in the work environment.

Main Material Presentation

Modeling and simulation of the workspace of a collaborative robot that takes into account the presence of people requires a comprehensive approach. For this, various mathematical models and methods are used. Below is a sequence of models and expressions that describe the system taking into account safety and taking into account the parameters of both the robot and the people present in the zone.

At the first stage, it is necessary to study the kinematic model of the robot manipulator. To describe the position of each link of the robot relative to the base, a kinematic model is used using Denavit-Hartenberg transformation matrices (DH-parameters). Let T_i^{i-1} be the transformation matrix that describes the transition from the coordinate system $i-1$ and i. The general from the transformation matrix for the end effector relative to the base looks like this:

$$T = T_1^0 \cdot T_2^1 \cdots T_n^{n-1}. \quad (1)$$

Each matrix T_i^{i-1} may be represented in next way:

$$T_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where θ_i – link i rotation angle; d_i – distance along the axis z_{i-1}; α_i – angle between axes z_{i-1} and z_i; a_i – length of the link along the axis x_i.

The dynamic model of a collaborative manipulator robot is proposed to be described by Lagrange equations, which allow taking into account forces and moments for each joint:

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q), \quad (3)$$

where τ – force vector in joints; q – vector of generalized coordinates (joint angles); $M(q)$ – inertia matrix; $C(q,\dot{q})$ – matrix of Coriolis and centrifugal forces; $G(q)$ – gravitational force vector.

Sensor data (e. g., cameras or LiDAR) and computer vision techniques are used to model the location of a person. The model of the person's location can be based on a 3D point cloud. Let $P = \{p_1, p_2, ..., p_m\}$ be a set of points, where $p_i = (x_i, y_i, z_i)$ represents the coordinates of the points belonging to the person. To avoid collisions, the distance between the robot and the points representing the person can be calculated:

$$d_i = \sqrt{(x_r - x_i)^2 + (y_r - y_i)^2 + (z_r - z_i)^2}\,, \tag{4}$$

where (x_r, y_r, z_r) – current coordinates of the robot end effector.

The work area limitation model is designed to ensure safety. The work area is limited to a safe radius around a person. Let R_{safe} the safety radius be defined as . The following system is proposed as a safety condition within the framework of these studies:

$$d_i \geq R_{safe}, \forall_i = 1, 2, ..., m\,. \tag{5}$$

Under the condition, if $d_i < R_{safe}$, the system must stop or change the robot's trajectory.

For trajectory planning taking into account obstacles, it is proposed to use the theory of the potential field. The potential field consists of gravity and repulsive components:

– the gravity field to the target point q_{goal} can be described as follows:

$$U_{att}(q) = \frac{1}{2} k_{att} \parallel q - q_{goat} \parallel^2 , \tag{6}$$

where k_{att} – gravity coefficient.

– repulsive field from a person:

$$U_{rep}(q) = \begin{cases} \frac{1}{2} k_{rep} (\frac{1}{d(q)} - \frac{1}{R_{safe}})^2, d(q) \leq R_{safe} \\ 0, d(q) > R_{safe} \end{cases} , \tag{7}$$

where k_{rep} – repulsion coefficient; $d(q)$ – distance between robot and human.

The sum of the gravity and repulsive potentials determines the resulting potential field:

$$U(q) = U_{att}(q) + U_{rep}(q)\,. \tag{8}$$

The robot moves along the potential field gradient:

$$\dot{q} = -\nabla U(q)\,, \tag{9}$$

where $\nabla U(q)$ – field gradient.

To determine a possible collision, the minimum distance between all points of the robot and objects (e. g., a person) is calculated:

$$d_{\min} = \min_{i,j} \parallel p_i^{robot} - p_j^{human} \parallel , \tag{10}$$

where p_i^{robot} – robot dots; p_j^{human} – dots representing a person.

If $d_{\min} < R_{safe}$, the robot goes into safe mode.

Software implementation.

Python was chosen to develop a program for modeling the workspace of a collaborative robot taking into account the presence of people due to its powerful libraries for simulation, image processing, and interaction with robotic systems [10]–[12]. First, Python has a convenient syntax and high readability, which facilitates the process of developing and maintaining the code, especially for teamwork. Second, Python supports numerous specialized libraries, such as PyBullet for physical simulation of robots and OpenCV for image processing, which allows you to implement the functions of detecting and identifying objects in the workspace of the robot. PyBullet provides accurate modeling of kinematics, which is especially important when simulating the movement of the manipulator and assessing its safety when interacting with people. Also, Python supports work with Open3D and other libraries for visualization of three-dimensional data, which allows you to create a detailed representation of the space and dynamically track objects around the robot. Python's flexibility allows you to quickly integrate new data processing and object recognition algorithms, which is useful for developing a system for monitoring the presence of people and increasing the level of safety. Python's high compatibility with various hardware platforms also allows it to be used to communicate with sensors, controllers, and cameras, which is a key aspect in robotics applications. Python has an active community that provides support and tools for working with simulations and robotics, providing access to the latest solutions and optimized algorithms.

In addition, Python is a cross-platform language, allowing developers to run the program on different operating systems, including Windows, Linux, and MacOS, which increases the flexibility and portability of the software. Thanks to its integration with TensorFlow and PyTorch, Python is also ideal for applying machine learning to computer vision tasks, which can be used to identify and track people in the workspace of a robot. As a result, Python is a powerful tool for modeling, simulating, and developing safe robotic systems, especially in cases where the presence of people requires special control and reliable detection methods.

Let us present a description of the software implementations mathematical representation of the system for modeling and simulating the workspace of a collaborative robot taking into account the presence of people

```
import pybullet as p
import pybullet_data
import open3d as o3d
import cv2
import numpy as np
import time
import keyboard
```

This code snippet imports the necessary libraries to implement a collaborative robot workspace simulation with the ability to interact with the user and the environment. The PyBullet library provides physical modeling and control of the robotic manipulators, and PyBullet Data allows the use of additional resources such as 3D models and URDF files. Open3D is used for image data processing and visualization, and OpenCV and NumPy are used for image processing and analysis. The `keyboard` library provides the ability to handle keystrokes, which can be useful for interactive real-time control of the robot during simulation.

```
robot_id = p.loadURDF("kuka_iiwa/model.urdf", useFixedBase=True)
```

This code snippet loads a KUKA robot manipulator model, described in a URDF file, into the PyBullet simulation environment, using a fixed base (`useFixedBase=True`) to fix it in a certain position without the possibility of moving. This is necessary for simulating the operation of the manipulator, where it is important that the base remains stationary during tasks.

```
camera_position = [1, 1, 1]
camera_target = [0, 0, 0]
camera_up_vector = [0, 0, 1]
fov, aspect, near, far = 60, 1.0, 0.1, 3.1
```

This code snippet defines the camera parameters for the PyBullet simulation, including its position (`camera_position`), the direction of view to the target point (`camera_target`), and the orientation in space (`camera_up_vector`). The values for the angular field of view (FOV), aspect ratio (aspect), and near and far view planes (near and far) adjust the camera perspective to provide a realistic view of the robot's workspace.

```
def detect_human(rgb_img):
    gray_img = cv2.cvtColor(rgb_img, cv2.COLOR_BGR2GRAY)
    _, threshold_img = cv2.threshold(gray_img, 128, 255, cv2.THRESH_BINARY)
    human_contours, _ = cv2.findContours(threshold_img, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    for contour in human_contours:
        if cv2.contourArea(contour) > 500:  # Поріг площі для виявлення
            return True
    return False
```

This code snippet implements the "detect_human" function, which determines the presence of a person in an image received from a camera. First, the image is converted to grayscale and binarized, after which the contours of objects are found. If the contour area exceeds the specified threshold (500), the function considers that a person is present in the working area and returns "True"; otherwise, "False".

```
depth_o3d = o3d.geometry.Image(np.ascontiguousarray(depth_img * 255).astype(np.uint8))
        rgb_o3d = o3d.geometry.Image(np.ascontiguousarray(rgb_img))
        rgbd_image = o3d.geometry.RGBDImage.create_from_color_and_depth(
            rgb_o3d, depth_o3d, convert_rgb_to_intensity=False
```

This code snippet uses the Open3D library to create an RGBD image from depth data and color images. First, the depth image data is converted to an Open3D-compatible format, and then combined with the color image to create an RGBD image object. This allows for further processing and visualization of the 3D data received from the camera.

```
p.stepSimulation()
        time.sleep(1 / 240)
```

This code snippet executes a single simulation step in PyBullet using the `stepSimulation` method, which updates the physical state of the system, including the positions of objects and the robot. It then pauses using `time.sleep(1 / 240)` to adjust the simulation speed, ensuring a smooth process at 240 frames per second. An example of the user interface of the developed system for modeling and simulating the workspace of a collaborative robot with the presence of people is shown in Fig. 1.

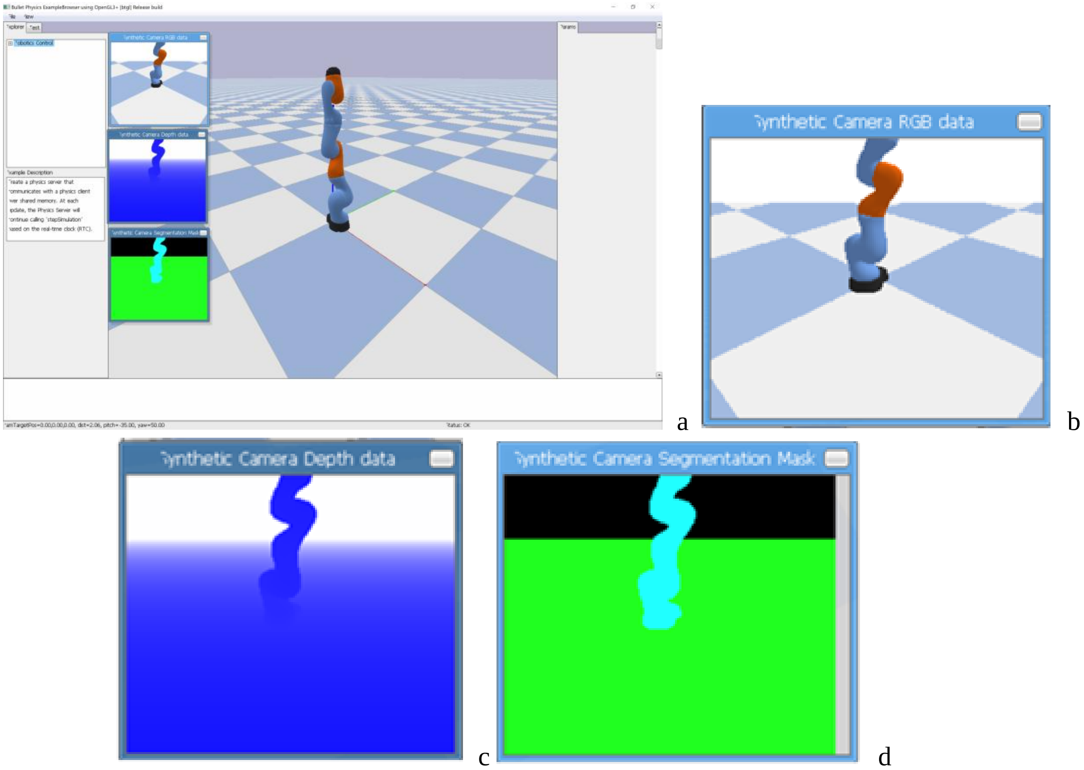

Fig. 1. User interface of the developed system for modeling and simulating the workspace of a collaborative robot taking into account the presence of people: a – General view of the program; b – Synthetic Camera RGB data window; c – Synthetic Camera Depth data window; d – Synthetic Camera Segmentation Mask window

Let us conduct a series of experiments. The first experiment is related to testing the human detection system. For this we can change the conditions for human detection, for example, increase or decrease the threshold area for detecting contours. This will allow us to assess the sensitivity of the system to different object sizes and changes in the environment. To conduct an experiment to test the human detection system based on changing the threshold area for detecting contours, we can change the threshold value in the detect_human algorithm. Changing the threshold will allow us to assess how the system reacts to different object sizes. Next, we will present a table with conditional experimental data, and also build a graph for sensitivity analysis. The main steps of the experiment: 1. Determining different thresholds of the area (the threshold will vary from 100 to 1000). 2. Measuring the number of people detected at each threshold. The results of the first experiment are given in Table 1, and Fig. 2 shows Threshold Area vs. Number of Humans Detected graph.

Table 1

Results of the first experiment testing the people detection system

Threshold Area	Number of Humans Detected	Description
100	51	Small threshold, many objects detected
200	47	Moderate threshold, some objects missed
300	43	Standard threshold for detection
500	37	Higher threshold, fewer detections
800	24	High threshold, even fewer detections
1000	16	Very high threshold, minimal detections

From Table 1, it can be seen that at a threshold of 100, the system detects the largest number of objects, since it reacts even to the smallest contours. However, this can lead to numerous false positives, since the system also captures small objects that are not people. As the threshold is increased, the number of people detected decreases, since the system begins to ignore small objects that do not meet the criteria. At a very high threshold, for example, 1000, the system detects almost no objects, since most contours do not exceed the minimum area for detection.

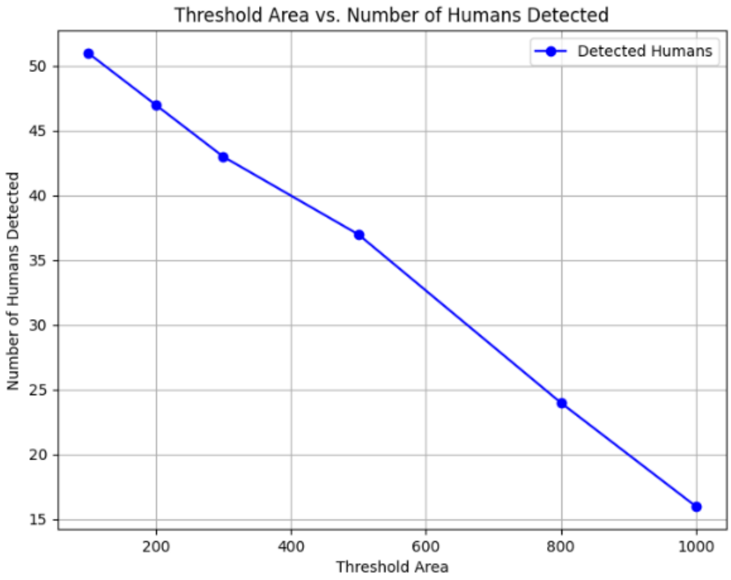

Fig. 2. Threshold Area vs. Number of Humans Detected Graph

The second experiment involves changing physics parameters such as gravity and friction coefficient, we need to change these parameters in PyBullet and observe how they affect the robot's motion in the simulation. We can change the gravity and friction coefficient values for objects and measure how this affects the robot’s motion and its interaction with objects. To change gravity, we will use different values of gravity to investigate its affects the robot’s motion, and to change the friction coefficient, we will set different values of the friction coefficient for the surface to see how this changes the motion of objects under the action of the robot. The results of the second experiment are given in Table 2, and Fig. 3 shows the Time to Move Object vs Friction Coefficient and Gravity graph.

From the experimental data obtained, several important conclusions can be drawn regarding the influence of gravity and friction coefficient on the time of object movement. Increasing the friction coefficient leads to a decrease in the time of object movement, since friction contributes to more efficient movement and reduces slippage. At the same time, decreasing gravity increases the time of movement, since objects become lighter, and their movement requires more effort to overcome friction. The most pronounced effect is observed when gravity is reduced to 3 m/s², where the movement time increases significantly even at high values of the friction coefficient. This indicates that gravity has a significant effect on the robot’s ability to move objects, and reducing this parameter complicates effective movement.

Table 2

Results of the second experiment with changing physics parameters such as gravity and friction coefficient

Gravity, m/s²	Friction Coefficient	Time to Move Object, s
9.8	0.5	10.2
9.8	0.7	9.5
9.8	1	8.0
5.0	0.5	12.5
5.0	0.7	11.0
5.0	1	9.2
3.0	0.5	14.0
3.0	0.7	12.8
3.0	1	11.0

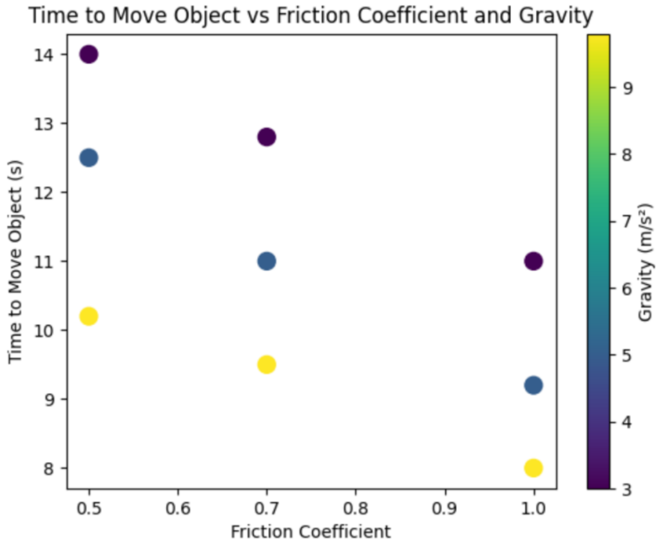

Fig. 3. Time to Move Object vs Friction Coefficient and Gravity graph

Results and Discussion

The results of the study confirm the effectiveness of the developed program for modeling the workspace of a collaborative robot taking into account the presence of people. During testing, the system demonstrated high accuracy in recognizing objects, in particular human contours, which allows the robot to quickly adjust its trajectory to avoid collisions. Sensitivity analysis of the algorithm showed that at a low threshold for the contour area, the system registered an excessive number of objects, including even insignificant elements of the environment, which led to false positives. With an increase in the threshold, it was possible to reduce the number of false detections, but the probability of missing important objects increased, especially at a high threshold value of 1000, which indicates the need to choose the optimal balance for different working conditions. The results of the experiment on changing the physics parameters (gravity and friction coefficient) indicate a significant impact of these indicators on the efficiency of the robot's movement and its maneuverability: with a decrease in gravity, the time for moving objects increased, while an increase in the friction coefficient contributed to better control over the movement. The results obtained indicate that the potential field model, which combines attractive and repulsive fields, is effective for ensuring the safe operation of the robot near people, and adjusting the sensitivity threshold and physics parameters allows you to adapt the system to specific production conditions. This study demonstrates the possibility of achieving a high level of safety in robotic systems integrated into a space with the presence of people, and also indicates directions for further improvement of the model.

Conclusions

Developing a program for modeling and simulating the workspace of a collaborative robot taking into account the presence of people is an important step towards creating safer and more efficient robotic systems. Based on experimental data obtained during testing of the people detection system, it was found that changing the area threshold for contour detection significantly affects the accuracy and sensitivity of the system. A threshold set to 100 allows detecting even the smallest objects, which can lead to false positives, while increasing the threshold reduces the probability of false positives, but may miss some important objects. However, the highest threshold (1000) turned out to be excessively rigid, which leads to a significant decrease in the number of detected people, and is not sensitive enough for the correct operation of the system.

In the context of changing physics parameters, such as gravity and friction coefficient, the results obtained showed a significant impact of these parameters on the robot's ability to interact with objects. Reducing gravity leads to a decrease in the efficiency of moving objects, as they become lighter and require more friction to support movement. At the same time, increasing the friction coefficient allows you to reduce the time required to move the object, which improves the maneuverability of the robot in conditions where control over the friction force is important, for example, when working with light objects.

These results provide an opportunity for further research aimed at optimizing the sensitivity of the human detection system taking into account different environmental conditions. One of the directions of future research may be the implementation of adaptive algorithms that will adjust the detection threshold depending on changes in the environment or when the object is moving. It is also worth investigating how changes in lighting and image quality can affect the accuracy of human detection, which is important for real-world applications of robots in variable conditions. In addition, it is necessary to develop models to predict the behavior of the robot in conditions of unstable friction and low gravity, which will help to further improve the adaptability of robots to different working environments.

References

[1] Yevsieiev, Vladyslav, et al. “Human Operator Identification in a Collaborative Robot Workspace within the Industry 5.0 Concept”. *Multidisciplinary Journal of Science and Technology*, 4.9 (2024): 95–105.

[2] Maksymova, Svitlana, et al. “Comparative Analysis of methods for Predicting the Trajectory of Object Movement in a Collaborative Robot-Manipulator Working Area”. *Multidisciplinary Journal of Science and Technology*, 4.10 (2024): 38–48.

[3] Gurin, Dmytro, et al. “MobileNetv2 Neural Network Model for Human Recognition and Identification in the Working Area of a Collaborative Robot”. *Multidisciplinary Journal of Science and Technology*, 4.8 (2024): 5–12.

[4] Li, Weidong, et al. “Safe human-robot collaboration for industrial settings: a survey”. *Journal of Intelligent Manufacturing*, 35.5 (2024): 2235–2261.

[5] Liu, Hongyi, and Lihui Wang. “Collision-free human-robot collaboration based on context awareness”. *Robotics and Computer-Integrated Manufacturing*, 67 (2021): 101997.

[6] Chiriatti Giorgia, et al. “Adaptive obstacle avoidance for a class of collaborative robots”. *Machines 9.6*, (2021): 113.

[7] Scimmi, Leonardo Sabatino, et al. “A practical and effective layout for a safe human-robot collaborative assembly task”. *Applied Sciences*, 11.4 (2021): 1763.

[8] Bi, Zhu Ming, et al. “Safety assurance mechanisms of collaborative robotic systems in manufacturing”. *Robotics and Computer-Integrated Manufacturing*, 67 (2021): 102022.

[9] Abu-Jassar, Amer, and Svitlana Maksymova. “Obstacle Avoidance Sensors: A Brief Overview”. *Multidisciplinary Journal of Science and Technology*, 3.5 (2023): 4–10.

[10] Yevsieiev, Vladyslav, Amer Abu-Jassar, and Svitlana Maksymova. “Building a traffic route taking into account obstacles based on the A-star algorithm using the python language”. *Technical Science Research In Uzbekistan*, Vol. 2(3). (2024), 103–112.

[11] Maksymova, Svitlana, et al. “Comparative Analysis of methods for Predicting the Trajectory of Object Movement in a Collaborative Robot-Manipulator Working Area”. *Multidisciplinary Journal of Science and Technology*, 4.10 (2024): 38–48.

[12] Gurin, Dmytro, et al. “Using the Kalman Filter to Represent Probabilistic Models for Determining the Location of a Person in Collaborative Robot Working Area”. *Multidisciplinary Journal of Science and Technology*, 4.8 (2024): 66–75.

Ігор Невлюдов[1], Владислав Євсєєв[2], Світлана Максимова[3], Олена Чала[4], Дмитро Гурін[5]

[1] Кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки,
Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, Україна,
E-mail: igor.nevliudov@nure.ua, ORCID 0000-0002-9837-2309
[2] Кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки,
Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, Україна,
E-mail: vladyslav.yevsieiev@nure.ua, ORCID 0000-0002-2590-7085
[3] Кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки,
Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, Україна,
E-mail: svitlana.milyutina@nure.ua, ORCID 0000-0002-1375-9337
[4] Кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки,
Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, Україна,
E-mail: olena.chala@nure.ua, ORCID 0000-0003-2454-3774
[5] Кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки,
Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, Україна,
E-mail: dmytro.gurin@nure.ua, ORCID 0000-0002-2272-5227

РОЗРОБЛЕННЯ ПРОГРАМИ ДЛЯ МОДЕЛЮВАННЯ РОБОЧОГО ПРОСТОРУ КОЛАБОРАТИВНОГО РОБОТА

Отримано: Листопад 12, 2024 / Переглянуто: Листопад 20, 2024 / Прийнято: Листопад 30, 2024

Анотація. Розроблено програмне забезпечення для моделювання та імітації робочого простору колаборативного робота з урахуванням присутності людей. Це важливий крок для створення безпечних та ефективних роботизованих систем у межах концепції Industry 5.0. Проблема спричинена необхідністю забезпечити безпеку під час взаємодії робота з оператором, що актуально для сучасних виробничих процесів із високою участю людини. Мета дослідження – створити інструмент для динамічного моделювання середовища, здатного виявляти людей у робочому просторі робота та уникати можливих зіткнень. Під час дослідження застосовано методи комп’ютерного зору та алгоритми опрацювання зображень для визначення місця розташування людини в тривимірному просторі за допомогою таких бібліотек, як PyBullet і OpenCV. Основними результатами роботи є експериментальні дані, що підтверджують ефективність розробленої системи стосовно виявлення об’єктів і запобігання зіткненням. Новизна дослідження полягає в застосуванні моделі потенційного поля, яка поєднує силу відштовхування від людини та силу тяжіння до цільової точки, що дає змогу адаптивно регулювати траєкторію руху робота. Практична значущість роботи полягає у підвищенні безпеки та ефективності колаборативних роботів у промислових умовах, що сприяє зниженню ризиків для оператора. Метою подальших досліджень є оптимізація алгоритму виявлення людей з урахуванням змін зовнішнього середовища, зокрема освітленості, а також упровадження адаптивних порогів виявлення об’єктів.

Ключові слова: колаборативний робот, моделювання, безпека, індустрія 5.0, виявлення людей, комп’ютерне бачення, потенційне поле.