Deep Learning for Safe Human-Robot Collaboration

Nicolás Duque Suárez, Lina María Amaya Mejía, Carol Martinez, Daniel Jaramillo-Ramirez

Abstract Recent advances in computer vision and deep learning have lead to implementations in different industrial applications such as collaborative robotics, making robots able to perform harder tasks and giving them consciousness of their environment, easing interaction with humans. With the objective of eliminating physical barriers between humans and robots, a security system for industrial collaborative robots based on computer vision and deep learning is proposed, where an RGBD camera is used to detect and track people located inside the robot's workspace. Detection is made with a previously trained convolutional neural network. The position of every detection is fed to the tracker, that identifies the subjects in scene and keeps record of them in case the detector fails. The detected subject's 3D position and height are represented in a simulation of the workspace, where the robot's speed changes depending on its distance to the manipulator following international safety guidelines. This paper shows the implementation of the detector and tracker algorithms, the subject's 3D position, the security zones definition and the integration of the vision system with the robot and workspace. Results show the system's ability to detect and track subjects in scene, and the robot's capacity to change its speed depending on the subject's location.

1 Introduction

Even though nowadays it is increasingly common to find human-robot collaborative work environments in the industry, there is still the fear of being injured by working next to a robot. Collaborative robots count with certified safety systems and follow the international standard ISO 10218-1 / -2 and the technical specification ISO TS 15066. The first, deals with the safety of industrial robots and the second, with industrial robots designed for collaborative operations [8]. However, according to

Nicolás Duque Suárez, Pontificia Universidad Javeriana, e-mail: n-duque@javeriana.edu.co · Lina María Amaya Mejía, Pontificia Universidad Javeriana, e-mail: linaamaya@javeriana.edu.co · Carol Martinez, SnT-Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, e-mail: carol.martinez@uni.lu · Daniel Jaramillo-Ramirez, Pontificia Universidad Javeriana, e-mail: d-jaramillo@javeriana.edu.co

the ISO 10218-1 standard, the robot is only one component of a robotic system and by itself is not sufficient to guarantee safe human-robot interaction [9]. Therefore, the need arises to design an external safety system to avoid as many accidents as possible caused by collisions between humans and robots.

Different implementations in the manufacturing industry have been made with the goal of creating vision-based safety systems that work in environments of collaboration between humans and robots (specifically robotic arms or manipulators) to increase the flexibility of assembly processes. Different approaches related to this human-robot collaboration using computer vision are summarized in [1]. In [2], a ToF (time of flight) camera is used to detect obstacles in the robot's path and recalculate the trajectory to avoid contact. Other sensors such as 2D cameras [3] or a RGBD camera, used to recreate a virtual environment for the robot and the human, and monitor possible collisions [4] have been used. Developments have also been made by using multiple RGBD cameras to monitor the human and robot movements to predict possible collisions in real-time [5]. Others, such as [6] have used different sensors (sensor fusion) like RGBD cameras and ToF sensors to create a "volumetric evidence grid" divided in three different regions. In that way, they can define a danger zone depending on the robot's position and trajectory, that, when met with the human's defined safety zone, stops the robot immediately.

The safety system described in this paper is a continuation of [7], whose goal was to create a safety system for human-robot collaboration in a recycling plant. In their work, people were detected around a robotic manipulator using a RGBD camera and traditional computer vision, and were classified into one of three calculated safety zones. Based on the zone's closeness to the robot, the manipulator stopped its movement.

Compared to the system of [7], the implemented system in this paper, uses a single RGBD camera to detect people with deep learning techniques and is capable of tracking them and estimating their 3D position. This information is used to change the robot's speed according to the security zone where the subject may be located.

The novelty stands on the combination of the detector and the tracker, which made the computer vision system robuster, meaning a more sustained source of information for the safety system. The possibility of calculating the 3D position, hence the height of the subject, lead to the generation of a 3D representation of the body, which handed a closer to reality perception of the environment to the collision avoidance algorithm inside the safety system than using only the 2D position of the subject.

The paper is organized as follows. Section 2 introduces the implemented solution. Section 3 explains the components of the vision system. Section 4 describes the tests made on the vision system and the results obtained. Finally, Section 5 presents the conclusions of this work.

2 Vision-Based Safety System

The system is based on an scenario where a robotic manipulator is on a table developing a Pick and Place routine. The robot used is the collaborative robot UR3

[11].The purpose of this system is to facilitate the interaction between an operator and the robot, protecting the operator's safety by avoiding collisions with the robot, and maintaining the robot's functions.

The system was planned under the speed and separation monitoring method (SSM), in which the robot's speed depends on the constant evaluation of the horizontal distance of the operator relative to the robot [18]. Here the distance analysis is performed by a computer vision system that communicates with the UR3 control system to variate its current speed.

2.1 System architecture

Fig. 1 presents the architecture of the proposed vision-based safety system. The proposed system integrates a computer vision system with the robot control system. In the first one, the data from the scene is acquired by a RGBD camera and used to detect and track people with a convolutional neural network (CNN). The 3D position of each person is estimated in real-time and is classified into one of three security zones relative to the robot, that were calculated based on the risk level of a collision between the human and the robot. This information is sent to the control system, where each person is related to a collision object to the robot by creating 3D figures in a simulated space. Finally, an evaluation is made so that the robot varies its speed or stops its movement to avoid a collision, by making a re-planning of the trajectory based on the new speed.

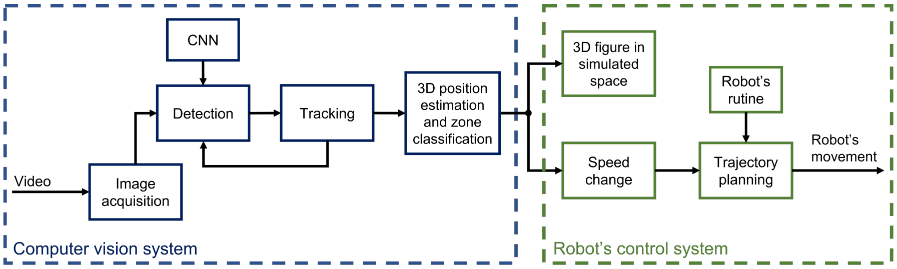


Fig. 1: System architecture

2.2 Security Zones

The security zones dimensions were calculated following the technical specification ISO/TS 15066:2016 and adapted to the characteristics of the UR3 robot to find a minimum separation distance between the robot and the operator, as shown in Fig. 2.

The limits of the security zones are described in Table 1, where *S* represents the operator's distance from the robot's base (for ease of implementation of the vision system, rectangular areas were assumed, keeping the minimum established distance). The speed of the robot when a person is detected at the low-risk zone was determined estimating a speed that would allow it to stop when a person went from the low-risk

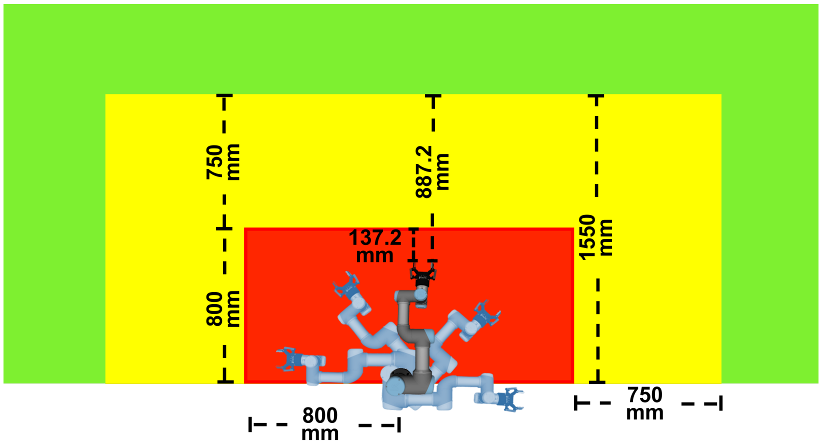


Fig. 2: Security zones dimensions

zone to the high-risk zone. At this speed, the stop time of the robot is lower than the collision time between the robot and a person.

Table 1: Security zones limits

Zone	Color	Limits	Speed
Safe zone	Green	S > 1550 mm	100 %
Low-risk zone	Yellow	800 mm < S ≤ 1550 mm	50 %
High-risk zone	Red	S ≤ 800 mm	0 %

When a person is detected in the safe zone, the robot will move at 100 % of its speed, if he moves to the low-risk zone, the robot will decrease its speed by 50 %, and if he enters to the high-risk zone, the robot will stop its movement. The opposite occurs if the person moves from the high-risk zone to the safe zone. If more than one person is detected in the scene, the robot will move according to the closest person.

3 Computer Vision System

The implemented computer vision system is in charge of 3 main processes: subject detection, tracking, and 3D position estimation.

3.1 Framework

As seen on Fig. 1 the computer vision system begins with the image acquisition using a RGBD camera. Each frame is given to the detector, which uses a trained convolutional neural network to detect heads.

The detector is not able to detect where the person is all the time. This is dangerous because if this happens when the subject is on a collision path with the robot, the safety system will not work. Due to this possible scenario, a tracker was added to the vision system. For every frame, the coordinates of the bounding box of the detected heads are sent to a tracker, which will use the received information to predict the subject's position whenever the detector does not work.

The 3D position of the subject is its x, y and z position relative to the center of the scene and it is calculated by using the coordinates of the bounding box found by the detector and the depth camera data at those same coordinates. With the location of the bounding box, the security zone where the subject is located is defined. In the end, the 3D position and the security zone where the subject is, are sent to the robot's control system where the relative distance between the subject and the robot is obtained and the manipulator's speed is modified.

3.2 Detection

To detect people from above, a CNN needed to be implemented. Tensorbox, which is a TensorFlow implementation that was designed to detect people in crowded environments by decoding the input image into a set of bounding boxes as an output directly, without having to evaluate the bounding boxes with a classifier and merging the results into a complete set of detections [12].

Tensorbox was the chosen pre-trained CNN and it was further trained using footage of people entering and exiting a bus station. This trained network was used in a project where the objective was to count the number of people that entered and exited a bus station every time a bus arrived there by detecting and tracking the people's head movement around the scene. Since the detection system implemented for that project fitted the framework of this security system, it was selected to be employed as a people detector within the robot's workspace.

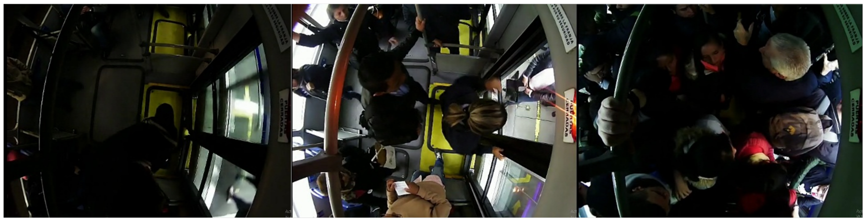


Fig. 3: Examples of images from the dataset

The dataset used to train the CNN was divided in three, where 70 % of the footage was used to train the network, 20 % was used for testing and 10 % for validation. Table 2 shows the division in numbers of the dataset. The dataset consisted of images such as the ones displayed in Fig. 3.

Table 2: Numeric division of dataset

Division	Images	Total Tags
Training	11293	33543
Test	6216	4664
Validation	3713	9452

The dataset was divided into images of each frame and CVAT [13] was used to tag the heads in each image using a bounding box.

The CNN performance was measured, and it was found that it was capable of detecting 92 % of the heads using the test dataset images. due to this result, the detector was considered to be used in a different environment (such as the workspace shown in Fig. 4).

3.3 Tracking

Two different trackers were tested for this system. CSRT, a tracker based on the properties of the pixels inside the bounding box [14] and SORT, which is a tracker-by-detection that predicts the subject's position based on the position of previous detections using a Kalman filter and the Hungarian algorithm to fulfill this objective. [15].

SORT was chosen to be the tracker for this system, because after the tests described in section 4.2 it was confirmed that this tracker had advantages over CSRT regarding tracking speed and path similarities to the reality when compared.

3.4 3D Position Estimation

The neural network used in this project was trained to detect heads from above, thus the camera was located on the ceiling looking downwards. A Kinect sensor [10] was located on the ceiling of the laboratory, pointing downwards to an open space that represents the work zone where the robot is located, as seen in Fig. 4. The camera was located 3.45 m above the floor and its scope was a rectangle of 4.2 m by 3.1 m.

Since the vision system needs to work in a 3-dimensional environment, height of the subject in scene and their 3D position must be calculated in order to represent it on a 3D simulation. By using the intrinsic parameters of the RGB camera (Kinect) and the distance between the sensor and the subject obtained by the depth camera, 3D position of the subject relative to the scene and height were calculated using Eq. 1 as used in [16].

$$P = (X_r, Y_r, Z_r) = \left(Z\left(\frac{u - x_0}{f_x}\right), Z\left(\frac{v - y_0}{f_y}\right), 3.45 - Z \right) \tag{1}$$

Variables u and v are the position of the central pixel of the bounding box with the image's top left corner as reference. x_0 and y_o are the image's central pixel position, f_x and f_y are the horizontal and vertical focal distance of the camera and Z is the distance between the lens and the subject.

The results obtained in Eq. 1 correspond to the position of the subject relative to the center of the scene. In order to obtain the relative position of the subject to the robot, a coordinate transform was made using ROS [17] that allowed to translate the coordinate system of the camera to the coordinate system of the robot's base.

Fig. 4 shows the setting and explains graphically how the height or Z position of the subject was calculated.

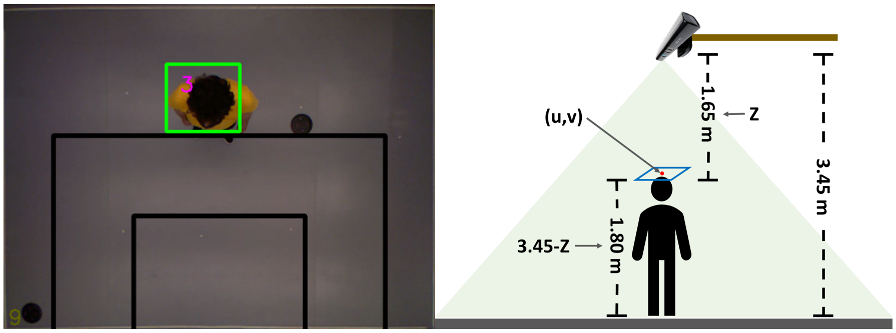


Fig. 4: Detection obtained by Tensorbox (left) and graphic calculation of a subject's 3D position

4 Experiments and Results

Tests were made to find out the performance of the vision system. They are divided in 2 sections: Detector and integration of the vision system with the robot system.

4.1 Detector

The detector went through two tests to find its performance under different conditions (light and subject's occlusion) as well as with a different number of subjects in scene.

4.1.1 Detection performance under different conditions

This test consisted of finding the number of detections obtained, while using 3 different lighting conditions on the scene and changing the subject's head coverage (uncovered, partially, and fully covered). Nine videos with 200 frames each for every combination of conditions were recorded. The subject made a similar path and was always inside the scene.

A luxmeter was used to measure the illumination conditions of the scene, where 43.5 lux, 320 lux, 613 lux corresponded to low, medium and high illumination conditions respectively. Regarding the head's coverage, a white helmet was used as full cover and a piece of paper was used as a partial cover.

Performance was measured by comparing the number of frames where the person was detected with the number of total frames the subject was actually on scene. The results are summarized in Table 3.

Table 3 shows that the lowest error happens when the head is uncovered and there is low illumination. These results may be due to the fact that the detector was trained using images of a bus station, whose illumination conditions were not controlled and oftentimes were similar to a low lighting scene. Also, people had their head uncovered in most of the captures used to train the model.

4.1.2 Detector performance with multiple subjects in scene

A possibility of more than one subject in scene must be evaluated. Three videos of 200 frames each were recorded, using medium lighting conditions (same used in

Table 3: Behavior of the system under different test conditions

		Low ilumination	**Medium ilumination**	**High ilumination**	**Heads total**
Uncovered head	Real frames	148	154	153	455
	Detected frames	128	114	103	245
	% Error	**13.5%**	25.9%	32.7%	**46.1%**
Partially covered head	Real frames	143	146	194	483
	Detected frames	53	64	26	143
	% Error	62.9%	56.2%	**86.6%**	**70.4%**
Covered head	Real frames	113	158	146	417
	Detected frames	96	46	49	191
	% Error	15.3%	70.9%	66.4%	54.2%
Ilumination total	Real frames	404	458	493	**1355**
	Detected frames	277	224	178	**679**
	% Error	**31.4%**	51.1%	**63.3%**	**49.9%**

Sect. 4.1.1). Each video had a different number of subjects moving. Fig. 5 shows captured frames of every video.

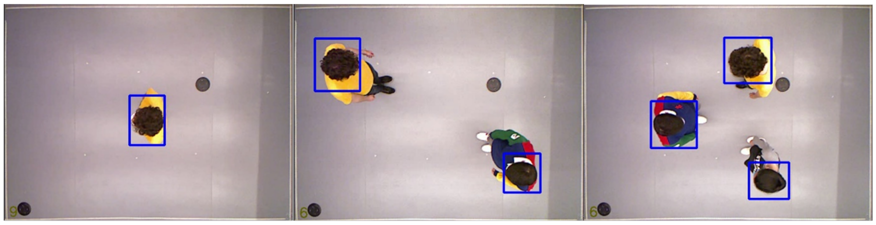


Fig. 5: Frames of footage with different number of subjects.

Each video had a confusion matrix, comparing how many people were detected versus how many people were actually in scene. Performance results are summarized on Table 4.

Table 4: Summary of performance measurements

Measure	**Case: 1 subject**		**Case: 2 subjects**			**Case: 3 subjects**			
Accuracy	88.40%		34.67%			22.61%			
	0 subjects	1 subject	0 subjects	1 subject	2 subjects	0 subjects	1 subject	2 subjects	3 subjects
Precision	62.50%	98.60%	34.55%	7.84%	100%	43.90%	14.95%	2.44%	100%
Recall	94.59%	87.04%	100%	25.00%	23.38%	48.65%	64.00%	5.56%	8.40%
F-score	75.27%	92.46%	51.35%	11.94%	44.21%	46.15%	24.24%	3.39%	15.50%

The accuracy displayed on Table 4, shows that the detector on its own can constantly detect a subject but struggles when there are more people in the scene. This is reflected on the precision scores, where the score was higher for the largest number of subjects of each case. Recall values show that the number of frames where an amount of people were detected, was lower than the number of frames with that number of people in scene. These results reinforce the idea that a tracking system is needed to keep a constant notion of the location of the subject.

4.2 Tracker

For the tracking system, a test was performed where two different trackers (SORT and CSRT) were given the same footage as in the test described in Sect. 4.1.2, and their performance was evaluated by calculating tracker speed and their RMSE and IoU which was compared with the ground truth (manually tagged bounding boxes). root mean square error (RMSE) between the paths obtained by both trackers with the real path, and the Intersection over union (IoU) between the bounding boxes obtained by both trackers and the ground truth (manually tagged bounding boxes per frame), as well as observing the tracking speed of both trackers.

Table 5 summarizes the measurements made. It can be noticed that although CSRT has a higher tracking speed with one subject, this value decreases when the number of subjects in scene rises, while SORT keeps this value constant independently of the subjects. This happens because while SORT only depends on the bounding box location, CSRT needs the properties of the pixels within the bounding box, which has a higher cost in processing speed.

Table 5 also shows that the RMSE between the SORT path and the ground truth path is in average, lower than the one obtained for CSRT. This is due to the way both trackers work. While SORT depends on the detections made in every frame, which are located similarly to the real subject, CSRT uses the first detection bounding box properties to track the subject, this means that as time passes, the bounding box obtained by the tracker will drift from the real position of the head and the path will differ from the real one.

The IoU results show that the bounding box obtained by the trackers (and detector to some extent) are not similar to those of the ground truth, meaning that the trackers bounding boxes either they were bigger than the head of the subject, or that they drifted, which would explain why CSRT has a lower IoU in general.

Table 5: Results obtained for each tracker

Tracker		**Speed (fps)**	**RMSE between trackers and ground truth (pixels)**	**IoU of bounding boxes (%)**
CSRT	1 subject	25	9.13	16.8%
	2 subjects	13	6.53	26.9%
	3 subjects	9	14.62	16.9%
SORT	1 subject	13	9.53	33.0%
	2 subjects	13	8.08	41.2%
	3 subjects	13	8.37	37.1%

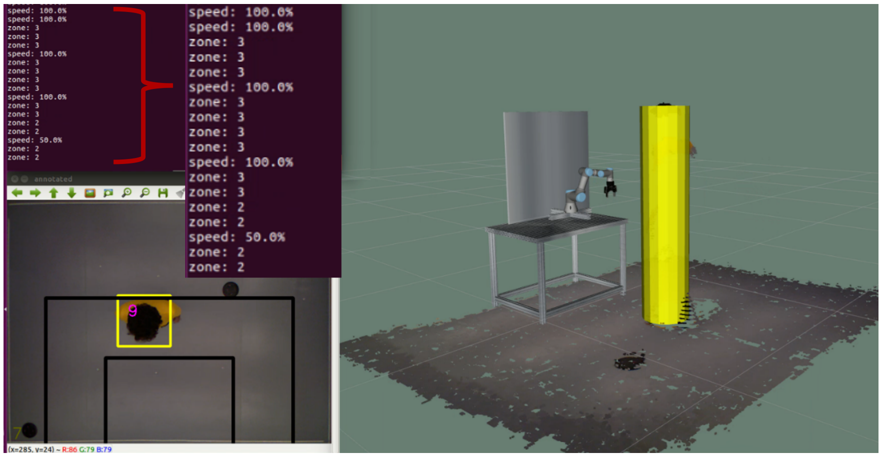
Because of these results, SORT was chosen to be the tracker for the implemented vision system.

4.3 Integration with ROS

4.3.1 Interaction with the simulated space

The system was implemented in ROS (Robot Operating System) in order to be able to handle and communicate with the robot. The detected subjects are used to generate 3D figures in the simulated space of the robot, with their estimated real height and the color of the zone where they were in. These figures are considered by the robot as collision objects. This way the robot “sees” if there is a person around it.

Fig. 6 shows the interaction of the vision system with the robot’s control system in the simulated space. The scene with the security zones is seen in the lower-left window. The system has detected a person, which has been marked with a yellow box (low-risk area). This generates a change in speed and creates a cylinder in the simulated space, in the same position and with the person’s real height.



```
speed: 100.0%
speed: 100.0%
zone: 3
zone: 3
zone: 3
speed: 100.0%
zone: 3
zone: 3
zone: 3
zone: 3
speed: 100.0%
zone: 3
zone: 3
zone: 2
zone: 2
speed: 50.0%
zone: 2
zone: 2
```

```
speed: 100.0%
speed: 100.0%
zone: 3
zone: 3
zone: 3
speed: 100.0%
zone: 3
zone: 3
zone: 3
zone: 3
speed: 100.0%
zone: 3
zone: 3
zone: 2
zone: 2
speed: 50.0%
zone: 2
zone: 2
```

Fig. 6: Frames of footage for every possible lighting and head coverage combination

4.3.2 Occupied zones analysis

It is determined by a test how accurate the vision system classifies people in scene in the security zones in comparison to the real zones where they are. With a correct classification of occupied areas, the correct speed change of the robot is obtained, ensuring the person’s safety. For this test, a 900 frames video was employed. A confusion matrix was made for each zone and with the information registered, different performance measurements were calculated and summarized on Table 6.

Table 6 shows that the three situations exceed 85 % for accuracy and 97 % for precision. This means that overall, the system classifies correctly the security zone of the detected subjects.

Table 6: Results obtained for each security zone

Measure	Green zone	Yellow zone	Red zone
Accuracy	85.6 %	91.9 %	97.9 %
Precision	97.7 %	99.4 %	99.5 %
Recall	57.4 %	81.9 %	91.5 %
F-score	72.3 %	89.8 %	95.3 %

A low recall percentage was obtained for the green zone; indicating that, only 57.4 % of the cases were detected correctly. Most of false negatives are caused by the absence of detections and not by the classification in another area. This is attributed to the Kinect's disparity effect, which reduces the usable field of view (no depth data on the borders of the image), reducing the number of correct detections in the green zone and therefore decreasing the recall score.

5 Conclusions

A vision-based safety system was designed and implemented on a scene where a collaborative industrial robot was located. The vision system can detect and track multiple individuals while they are in the scene of interest under controlled lighting conditions. The security zones were designed applying the standards of industrial robots and collaborative industrial robots.

Using a tracker improved the performance of the vision system when more than one subject is in the scene. The vision system can also estimate the position of these people and classify them in safety zones relative to the robot, being accurate 98 % of the time inside the most critical zone.

The vision system interacts with the robot control system, generating 3D figures in the simulated space of the robot to represent the people detected as collision objects and make the robot stop or vary its working speed according to the area of the detected subjects. With this work it was confirmed that it is feasible to eliminate the use of physical barriers and multiple sensors around an industrial robot, to promote collaborative work between robots and humans.

References

1. Halme, R. J., Lanz, M., Kämäräinen, J., Pieters, R., Latokartano, J., Hietanen, A. (2018). Review of vision-based safety systems for human-robot collaboration. Procedia CIRP, 72, 111–116. https://doi.org/10.1016/j.procir.2018.03.043
2. Ahmad, R., Plapper, P. (2015). Human-Robot Collaboration: Twofold Strategy Algorithm to Avoid Collisions Using ToF Sensor. International Journal of Materials, Mechanics and Manufacturing, 4(2), 144–147. https://doi.org/10.7763/ijmmm.2016.v4.243
3. M. Lešo, J. Žilková and M. Vacek, "Robotic manipulator with optical safety system," 2015 International Conference on Electrical Drives and Power Electronics (EDPE), 2015, pp. 389-393, doi: 10.1109/EDPE.2015.7325326.
4. Mohammed, A., Schmidt, B., Wang, L. (2016). Active collision avoidance for human–robot collaboration driven by vision sensors. International Journal of Computer Integrated Manufacturing, 30(9), 970–980. https://doi.org/10.1080/0951192x.2016.1268269
5. Morato, C., Kaipa, K. N., Zhao, B., Gupta, S. K. (2014). Toward Safe Human Robot Collaboration by Using Multiple Kinects Based Real-Time Human Tracking. Journal of Computing and Information Science in Engineering, 14(1). https://doi.org/10.1115/1.4025810

6. P. Rybski, P. Anderson-Sprecher, D. Huber, C. Niessl and R. Simmons, "Sensor fusion for human safety in industrial workcells," 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 3612-3619, doi: 10.1109/IROS.2012.6386034.
7. Medina, A. C., Mora, J. F., Martinez, C., Barrero, N., Hernandez, W. (2019). Safety Protocol for Collaborative Human-Robot Recycling Tasks. IFAC-PapersOnLine, 52(13), 2008–2013. https://doi.org/10.1016/j.ifacol.2019.11.498
8. Info PLC. (2017, February 27). La seguridad en la colaboración inteligente hombre-robot. Retrieved May 21, 2020, from https://www.infoplc.net/plus-plus/tecnologia/item/104052-seguridad-colaboracion-inteligente-hombre-robot
9. Beaupre, M. (2014). Collaborative Robot Technology and Applications [Slides]. Retrieved from https://www.robotics.org/userAssets/riaUploads/file/4-KUKA_Beaupre.pdf
10. Al-Naji, AA, Gibson, K, Lee, S-H, Chahl, J. (2017). Real Time Apnoea Monitoring of Children Using the Microsoft Kinect Sensor: A Pilot Study.Sensors.17.286.10.3390/s17020286.
11. Universal Robots. (2018). e-Series DE UNIVERSAL ROBOTS. Retrieved from https://www.universal-robots.com/cb3/
12. Stewart, R., Andriluka, M., Ng, A. (2016). End-to-End People Detection in Crowded Scenes. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2016.255
13. Computer Vision Annotation Tool (CVAT). (2020). Software. Retrieved from https://cvat.org
14. Mallick, S. (2017, February 13). Object Tracking using OpenCV (C++/Python). Retrieved May 23, 2020, from https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/
15. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B. (2016). Simple online and realtime tracking. 2016 IEEE International Conference on Image Processing (ICIP). https://doi.org/10.1109/icip.2016.7533003
16. Mora, JF., Medina AC., Ramírez, E., Mártinez, CV. (2018). Human Recognition Algorithm for Industrial Collaborative Robots in Automated Waste Separations Tasks (Thesis).
17. (n.d). In ROS.org. tf, Static Transform Publisher - ROS Wiki. Retrieved June 5, 2020, from http://wiki.ros.org/Parameter%20Server
18. Beaupre, M. (2014). Collaborative Robot Technology and Applications [Slides]. Retrieved from https://www.robotics.org/userAssets/riaUploads/file/4-KUKA_Beaupre.pdf