electronics

MDPI

Article

Working toward Solving Safety Issues in Human–Robot Collaboration: A Case Study for Recognising Collisions Using Machine Learning Algorithms

Justyna Patalas-Maliszewska [1,*], Adam Dudek [2], Grzegorz Pajak [1] and Iwona Pajak [1]

[1] Institute of Mechanical Engineering, University of Zielona Góra, 65-417 Zielona Gora, Poland; g.pajak@iim.uz.zgora.pl (G.P.); i.pajak@iim.uz.zgora.pl (I.P.)
[2] Faculty of Technical Science, University of Applied Sciences in Nysa, 48-300 Nysa, Poland; adam.dudek@pans.nysa.pl
* Correspondence: j.patalas-maliszewska@iim.uz.zgora.pl or j.patalas-malsizewska@iim.uz.zgora.pl

Abstract: The monitoring and early avoidance of collisions in a workspace shared by collaborative robots (cobots) and human operators is crucial for assessing the quality of operations and tasks completed within manufacturing. A gap in the research has been observed regarding effective methods to automatically assess the safety of such collaboration, so that employees can work alongside robots, with trust. The main goal of the study is to build a new method for recognising collisions in workspaces shared by the cobot and human operator. For the purposes of the research, a research unit was built with two UR10e cobots and seven series of subsequent of the operator activities, specifically: (1) entering the cobot's workspace facing forward, (2) turning around in the cobot's workspace and (3) crouching in the cobot's workspace, taken as video recordings from three cameras, totalling 484 images, were analysed. This innovative method involves, firstly, isolating the objects using a Convolutional Neutral Network (CNN), namely the Region-Based CNN (YOLOv8 Tiny) for recognising the objects (stage 1). Next, the Non-Maximum Suppression (NMS) algorithm was used for filtering the objects isolated in previous stage, the *k*-means clustering method and Simple Online Real-Time Tracking (SORT) approach were used for separating and tracking cobots and human operators (stage 2) and the Convolutional Neutral Network (CNN) was used to predict possible collisions (stage 3). The method developed yields 90% accuracy in recognising the object and 96.4% accuracy in predicting collisions accuracy, respectively. The results achieved indicate that understanding human behaviour working with cobots is the new challenge for modern production in the Industry 4.0 and 5.0 concept.

Keywords: human–robot collaboration; collision recognition; video recording; deep learning algorithms

Citation: Patalas-Maliszewska, J.; Dudek, A.; Pajak, G.; Pajak, I. Working toward Solving Safety Issues in Human–Robot Collaboration: A Case Study for Recognising Collisions Using Machine Learning Algorithms. *Electronics* **2024**, *13*, 731. https://doi.org/10.3390/electronics13040731

Academic Editors: Jesús Ángel Román Gallego, María-Luisa Pérez-Delgado, María Concepción Vega Hernández, Alfonso Jose Lopez Rivero and Daniel Hernández De la Iglesia

Received: 10 January 2024
Revised: 7 February 2024
Accepted: 9 February 2024
Published: 11 February 2024

1. Introduction

One of the assumptions of the Industry 4.0 or even Industry 5.0 concept is the use of new technologies in the production process to integrate many areas, such as objects, people and machines, into the same place and time. Nowadays, collaborative robots (cobots) integrate industrial automation possibilities with workers' capabilities in order to enhance the performance of production systems [1]. The Human–Robot Collaboration (HRC) systems enable the collaboration of human operators and cobots in the same workspace in many areas, including manufacturing, logistics, etc. Building user-centric HRC systems, with an emphasis on user safety, requires them to be combined with cyber-physical systems and user-oriented visualisation and interaction technology. There is a risk of collisions between robots and operators occurring when they work together in a common workspace. Accidents occur when the operator cannot anticipate the robot's movement and enters the robot's working area; thus, ensuring the safety of humans in such co-working spaces is

crucial for the further development of such systems and for assigning tasks to humans and cobots.

There is still a need, therefore, to reduce potential physical injuries due to this collaboration [2]. In [3], such physical damage, the incorrect programming of cobots' movements, unsafe worker behaviour and the malfunctioning of safety measures are distinguishable. Early reaction and the avoidance of collisions in a work area shared by collaborative robots (cobots) and operators is crucial for assessing the quality of operations and tasks carried out in production. Therefore, in order to minimise collisions in these workspaces, used by cobots and human operators, researchers are looking for new solutions related to the prediction of and early warnings against such events. In [4], simulation models devoted to expected collisions, based on the depth–image camera installed outside the equipment, were developed. Next, in [5], a dynamic human–robot fusion algorithm, in order to estimate, in real-time, the minimum human–robot distance requires, based on image processing and 3D representation, is proposed. A great challenge for the researcher is to find a solution that would provide early collision warnings with a high efficiency in a realistic environment. HRC, as an element combining human skills with the efficiency and precision of the machine, is based largely on a set of data. The use of machine learning to increase the effectiveness of HRC, which is a deep learning method, seems to be fully justified, as is confirmed by the extensive literature in this area. Study [6] presents research in the field of multimodal robot control interface for human–robot collaboration. Three methods were integrated into the multi-modal interface, including voice recognition, hand motion recognition and body posture recognition. Deep learning was adopted as the algorithm for classification and recognition. In paper [7], it was assumed that fluid, human–robot coexistence in manufacturing requires accurate estimation of the intention of human motion so that the efficiency and safety of HRC can be guaranteed. The authors, in their research, extracted temporal patterns of human motion automatically, thus outputting the predicted result before any motion took place. For this purpose, a deep learning system combining the Convolutional Neural Network (CNN) with the long short-term memory network (LSTM) towards recognising visual signals is explored to predict human motion accurately. The study of deep learning as a data-driven technique for the continuous analysis of human motion and the prediction of future HRC needs, leading to better planning and control of robots while performing a common task, is also presented in the paper [8]. In the context of HRC safety at work, in [9], an intelligent system, based on deep learning and machine learning techniques was proposed. The proposed solution is divided into two modules: detecting collisions between humans and robots and the detection of workers' clothing. The results achieved a sensitivity level greater than 90% in identifying collisions and an accuracy above 94% in identifying workers' clothing. The authors pointed out that the proposed intelligent system effectively supports safe HRC. Next, by applying machine learning algorithms, collisions in shared workspaces can be detected before they occur. In [10] the framework for task and status recognition for HRC in Mold Assembly using YOLOv5 was developed. The results indicated the performance had a mean average precision value of 84.8%. Moreover, in our previous work [11,12], an analysis of the literature in the area of applying Convolutional Neural Network (CNNs) and of CNNs, combined with classifiers for Human Activity Recognition (HAR), was performed.

The new method proposed in this article (Figure 1), based on deep learning models, applies YOLOv8 Tiny followed by algorithms for filtering, separating and tracking objects (cobots and humans) in workspaces, to collect real time data, from video cameras, in order to predict collisions in workspaces occupied by both collaborative robots and human operators. Firstly, after creating the research unit and developing scenarios for co-operating a human operator alongside two cobots, a dataset was created with the three cameras observing the workspace. This dataset consisted of 484 images with an initial size of 4096×2160 each. Next, all acquired images were labelled to create a training dataset consisting of five different categorical classes of human operator: arm, forearm, hand, head and torso; three categorical classes of robot: joint, arm, gripper; and, additionally,

two classes of objects that correspond to tools or other items that can be held in a human hand, which will be included in further research; the smaller one is marked as a slim tool, the larger one as a big tool. This dataset was then used to train the YOLO v8 model to detect objects later. Next, in Stage 2, data acquired from YOLO were used in order to eliminate falsely recognised objects, particularly certain categories, and finally we clustered them to create groups representing objects in the workspace (cobots, human operators). For filtering objects isolated by YOLO Non-Maximum Suppression (NMS), an algorithm was used. The *k*-means Clustering method and Simple Online Real-Time Tracking (SORT) approach were utilised for separating and tracking cobots and human operators. Data preformed in this way were applied for predicting possible collisions (Stage 3) with the use of the Convolutional Neutral Network (CNN).

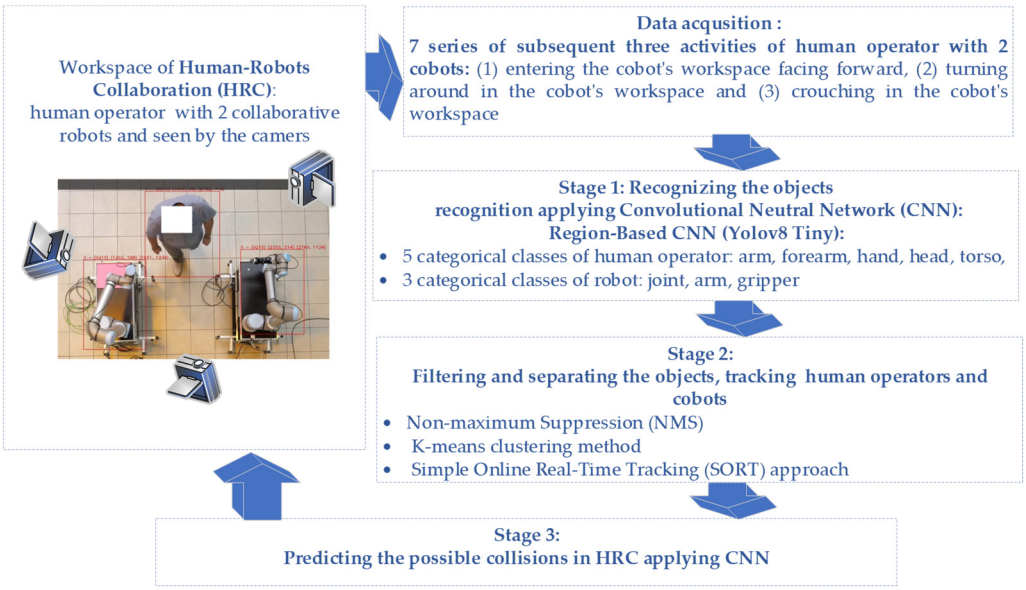


Figure 1. An approach to recognising collisions between collaborative robots and human operators.

The proposed method involves the CNN Region-Based CNN (YOLOv8 Tiny) for isolating and recognising objects followed by the algorithms NMS, *k*-means and SORT for data preparation and CNN for predicting possible collisions.

2. Materials and Methods

2.1. Data Collection Process

The research unit includes 2 UR10e cobots and 3 Panasonic Lumix GH6 cameras, allowing for the human interactions with these cobots to be recorded from three views, viz., from above, side and front, and was built at the Institute of Mechanical Engineering of the University of Zielona Góra. Parameters of cobots and cameras used are shown in Tables 1 and 2, and a schema showing the layout of experimental setup is presented in Figure 2.

Table 1. UR10e technical specifications.

Parameter	Value
Payload	12.5 kg
Reach	1300 mm
Degrees of freedom	6 rotating joints
Force/Torque Sensing	100.0 N 10.0 Nm
Pose Repeatability	±0.05 m
Axis Working Range	±360°
Axis Maximum Speed	±120°/s (arm) ± 180°/s (wrist)
TCP speed	1 m/s

Table 2. Cameras' specification.

Digital Camera	
Model	Panasonic GH6
Type	Digital Single Lens Mirrorless camera
Image sensor	Live MOS sensor 25.2 megapixels
Image stabilization	5 axis
Control	Remote control
Lens	
Model	Venus Optics LAOWA 7.5 mm f/2 MFT
Focal length	7.5 mm
Maximum aperture	f/2
Lens Construction	13 elements/9 groups
Recording Parameters	
Resolution	3840 × 2150
Recording speed	100 fps
Color coding	10 bit
Compression format	MPEG-4 HEVC
Bitrate	min. 50 Mb/s

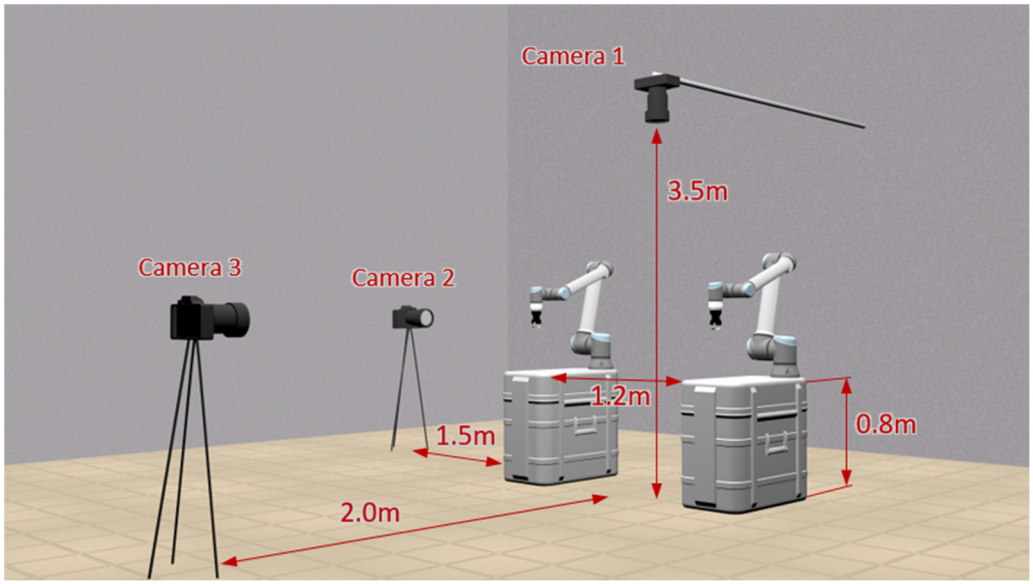


Figure 2. The layout of experimental setup.

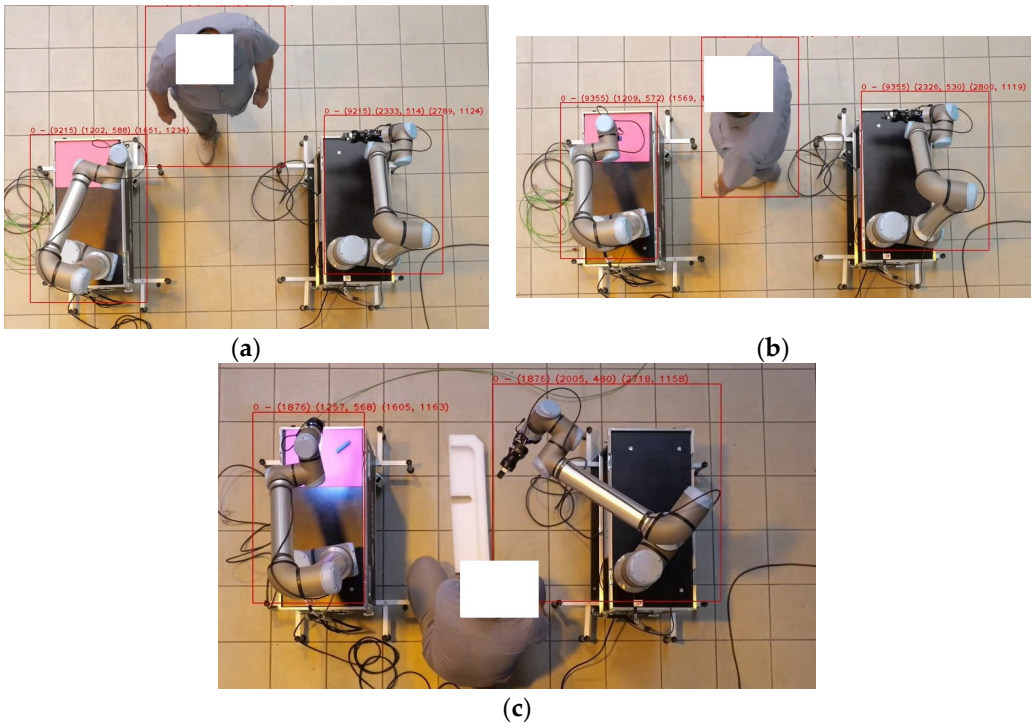
The first robot performed a cyclic task typical for industrial applications, moving the TCP between intermediate points (coordinates expressed in the base system): $[-0.17, -0.52, 0.27]^T$, $[-0.18, -0.53, 0.73\,]^T$, $[-0.18, -0.8, 0.73]^T$, $[\,0.28, -0.8, 0.73]^T$, $[\,0.28, -0.53, 0.73,\,]^T$ and maintaining orientation $[\pi, 0, 3\pi/4\,]^T$ (expressed in Euler angles Z-Y-X). The second robot used a vision system (wrist camera) to search for a specific object in the workspace. The found object was picked up, moved towards the first robot and then dropped in the workspace. The position of an object dropped from a certain height changed in each case, which introduced an element of randomness to the task.

The operator subsequently performed a series of planned activities in co-operation with cobots in a common workspace. The study was approved by the Bioethical Committee of University of Zielona Góra, Poland. The cameras have lenses with the lowest possible geometric distortions. All of them recorded images with the following parameters:

- Width of 1 frame: 3840 px;
- Height of 1 frame: 2150 px;

- Recording speed: 100 frames/s;
- Data recording format: MPEG-4 HEVC.

The research experiment consisted of three, defined, subsequent activities of a human operator with cobots (Figure 3a–c) specifically: (1) entering the cobot's workspace facing forward, (2) crouching in the cobot's workspace, (3) turning around in the cobot's workspace.



(**a**) (**b**)

(**c**)

Figure 3. Co-operation of the human operator with cobots in the recording from the camera from above: (**a**) entering the cobot's workspace facing forward, (**b**) crouching in the cobot's workspace, (**c**) turning around in the cobot's workspace.

2.2. Learning Patterns

The first stage of the experiment presented was the YOLO v8 network for recognising and indicating individual elements of cobots and the operator's body parts in the videoed material. Since the experiment used recordings from 3 different views, depending on the direction of the camera, each of the elements recognised has a completely different shape. Therefore, it was proposed that, in the first stage of our research (Figure 1), we use three independent artificial neural networks, which would first be trained with a dedicated sets of patterns and then used for the parallel detection of objects in the same scene from three different directions. Thanks to this, it is possible to indicate the relationships between detected objects, taking into account their location in 3D space while, at the same time, confirming the occurrence of a collision in the view from a minimum of 2 of them. In the second stage of the experiemnt (Figure 1), research is only provided using data from the "top" view, because this dataset allows for the largest number of potential collisions to be detected.

For each of these three networks, it was necessary to prepare independent sets of training patterns. Based on the analysis of the records recorded by individual cameras, series of images were selected that showed objects that individual networks then learned to recognise. This analysis looked for scenes in which individual objects were visible from different angles, in different positions and in different relationships to other objects. Ultimately, the following were selected as images in which training patterns were later indicated:

- 215 images for the view corresponding to camera position 1;
- 149 images for the view corresponding to camera position 2;
- 120 images for the view corresponding to camera position 3.

As mentioned in the introduction, in these images were marked occurrences of 8 classes of objects were marked in these images: cobot joint, cobot arm, cobot gripper, human head, human torso, human arm, human forearm, human hand.

There are many tools available that enable the process of annotating objects in images and then exporting information about these selections to a format that is accepted by the YOLO neural network model, which was used to perform the object's learning and detection tasks in this experiment. Among those available, a tool called Computer Vision Annotation Tool was selected, available as an online application at https://www.cvat.ai/ (accessed on 25 October 2023). It is a versatile tool used by many researchers to help point to objects in images and video recordings and its free version provides the functionality required to perform the experiment presented. Using this tool, rectangular markings, indicating individual types of objects were manually selected in each of the images from the 3 training sets. Figure 4a–c present example images with such markings, respectively, for the training sets for cameras 1, 2 and 3.

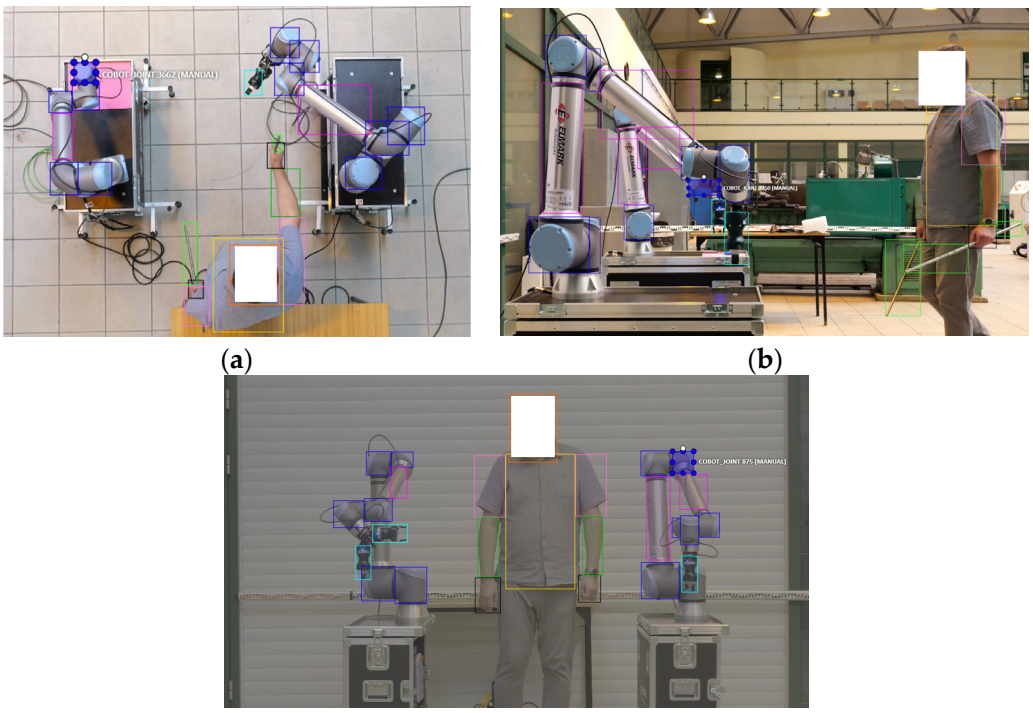


Figure 4. Example of selecting objects for a scene from the view of cameras 1, 2 and 3, namely: (**a**) from above, (**b**) from side and (**c**) from front.

Table 3 presents a quantitative summary of indications of individual types of objects in each of the data sets, corresponding to individual views from 3 cameras.

Table 3. Quantitative summary of object type designations in training patterns.

Object Class	Set 1	Set 3	Set 3
cobot joint	2431	1214	1382
cobot arm	640	488	458
cobot gripper	245	311	318
human head	170	153	101
human torso	213	170	108
human arm	320	261	154
human forearm	271	256	160
human hand	280	226	165
total	4570	3079	2846

Finally, for each of the images in each of the three sets, text files were obtained that contain information about the location of rectangular areas corresponding to individual selections in a format consistent with the format used by the YOLO library [13].

2.3. Data Preprocessing

Analysis of the data detected by YOLO revealed that it is not possible to use it directly to prepare the algorithm for predicting collisions based on CNN. In particular some objects were detected multiple times leading to the multiple rectangular markings indicating the same object. It was impossible to use such data when the number of network inputs had to be fixed. Moreover, due to the fact that there were two cobots and up to two human operators in the workspace, there was a problem with separating rectangular markings into groups representing, individually, the cobot and operator. Finally, each marking had to be assigned a unique identifier, the same as in subsequent frames of the video, to enable tracking changes in the marking position over time. Subsequent stages of data preprocessing, used in the approach presented are shown in Figure 5.

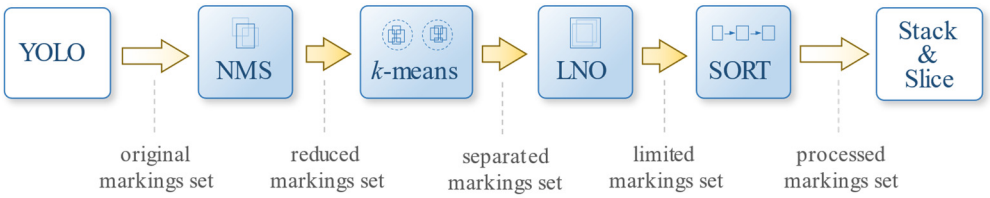


Figure 5. Stages of data preprocessing.

In order to eliminate multiple detections of the same objects, the technique Non-Maximum Suppression (NMS) was used [14]. Due to ensuring the reliability of the collision prediction algorithm, it was important to preserve the largest markings identified by YOLO; therefore, in implementing the NMS algorithm proposed, the criterion describing the markings area was adopted as a key criterion. For this reason, the list of true detections was created by selecting the largest rectangles representing the objects on a single frame and eliminating markings whose overlapping measure exceeded the threshold values adopted. In the proposed approach to assess the degree of overlapping the Intersection over Union (*IoU*) (*IoU*) the index defined by Formula (1) was used.

$$IoU = \frac{R_i \cap R_j}{R_i \cup R_j}, \tag{1}$$

where R_i, R_j are rectangles for the i-th and the j-th object. Such an approach allowed most of the multiple detections to be removed, but a certain number of unnecessary markings remained and they were eliminated in the next stages of the preprocessing process.

The reduced set of rectangles was subjected to cluster analysis in order to separate markings into groups representing individual cobots and operators in the workspace. For

this reason, the *k*-means Clustering method, minimizing distances between objects and centroids of their clusters, was used. In those experiments whose results are presented in this work, there were two cobots in the workspace; however, the number of operators may vary; in all scenarios, at the beginning, the operator is outside the cameras field of view; then, one or two operators may appear on the scene. Application of the *k*-means method combined with assessment of clustering results additionally allowed the number of human operators in the workspace to be determined. In the solutions proposed, the markings representing the operator were always separated into two clusters and the results obtained were analysed using the Silhouette score metric, defined for the centre of *i*-th rectangular markings as follows

$$s_i = \begin{cases} \frac{b_i - a_i}{\max(a_i, b_i)} & if\ |C_I| > 1 \\ 0 & otherwise \end{cases}, \tag{2}$$

where $a_i = \frac{1}{|C_I|-1}\sum_{j \in C_I, i \neq j} d(i,j)$, $b_i = \min_{J \neq I}\left(\frac{1}{|C_J|}\sum_{j \in C_J} d(i,j)\right)$, C_I is the cluster to which the *i*-th marking belongs, C_J means other clusters. $|\cdot|$ is the number of markings in cluster and $d(i,j)$ is the distance between the centre of the *i*-th and the *j*-th markings. The average value of s_i over all markings was taken to assess separation of the clusters. When the value of this coefficient exceeded the threshold value adopted it was assumed that there were two operators in the workspace, otherwise there was one operator. If no markings indicated object classes connected to humans were detected the absence of an operator in the workspace was registered.

In the next stage, when the number of robots and operators in the workspace was known and objects were assigned to appropriate clusters, the elimination of unnecessary markings which remained after using NMS was possible. Taking into account that each cobot is composed of 10 objects from 3 classes (7 joints, 2 arms, 1 gripper) while the operator has 8 objects from 5 classes (1 head, 1 body, 2 arms, 2 forearms, 2 hands), then Limiting Number of Objects (LNO) could be carried out. For this reason, the redundant rectangles of the smallest sizes were combined with the closest rectangle representing an object of the same type. After completing this stage, the number of objects recognised did not exceed the maximum number of objects resulting from the state of the workspace.

The last stage of the preprocessing process assigned unique identifiers to rectangular markings allowing them to be tracked between the frames of the video. For this purpose, the Simple Online Real-Time Tracking (SORT) algorithm, a widely used algorithm for tracking objects in real-time applications, was applied [15]. This algorithm, using the Kalman filter, predicts the positions of rectangular markings in subsequent frames and, comparing the predictions with the closest objects detected, in the sense of IoU measure, assigns them appropriate identifiers. After this stage of processing, the set of markings was limited, individual rectangles were assigned to appropriate clusters and labelled in such a way that it is possible to track them between video frames.

2.4. Neural Network Prediction

In order to predict the possibility of collision between human operators and cobots operating in the common workspace, utilisation of the Convolutional Neural Network (CNN) binary classifier was proposed. The task of the classifier proposed in the solution presented is to assess the current situation in the workspace as close to—or as far from—collision. To prepare input data for this classifier, the data preprocessed in stage 2 were stacked and sliced using the overlapping window technique. In stacking step 112, coordinates describing positions of rectangular markings representing both cobots and human operator objects from one frame ((2 cobots × 10 objects + 1 operator × 8 objects) × 4 coordinates) were combined into one vector and then arranged in a stack containing vectors corresponding to subsequent time moments of the movie. In this way, 112 time series, which can be interpreted as signals describing the changes of markings positions, were obtained. In the sliding step such signals were sliced in a time domain to form 2D windows including

100 successive vectors. Segmentation of data was performed using the overlapping window approach and 50 samples of the overlapping half window size.

The general structure of the CNN classifier used is shown in Figure 6. The structure of the classifier adopted, results from a series of experiments wherein the tuning of hyper-parameters, describing the number of convolutional layers, the number of filters and the size of the kernels, was performed.

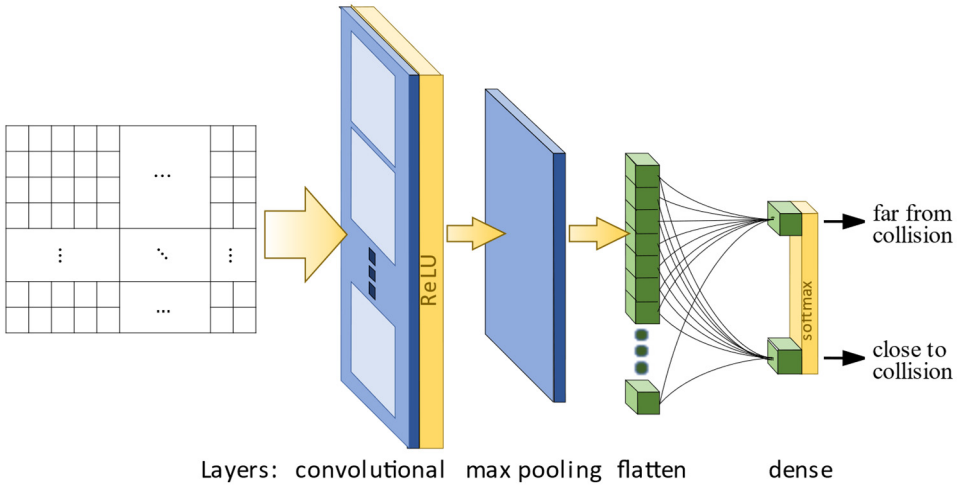


Figure 6. General structure of CNN.

To prepare data for training and testing a CNN for each window (input data), the corresponding class (output data), 'far from collision' or 'close to collision', had to be determined. For this purpose, each frame was labelled with collision while the distance to collision expressed in time instants for each window was calculated. Those windows whose distance from the collision was smaller than the adopted threshold were assigned to the class 'close to collision' with the rest being assigned to the class 'far from collision'. The distribution of the data samples obtained is presented in Table 4.

Table 4. The distribution of samples.

Distance to Collision (Measured in Time Instants)	Number of Samples	Percentages
far from collision	2958	72.4%
close to collision	1126	27.6%
Total	4084	100.0%

3. Results

As mentioned in the introduction, the aim of the research is to develop a method for recognising and predicting possible collisions in the collaboration of human operators and cobots in the same workspace. The proposed solution assumes the use of artificial neural networks in two stages:

- Stage 1: detection the position of individual parts of the operator's body and cobot elements in real time;
- Stage 2: filtering objects, separating and tracking human operators and cobots;
- Stage 3: prediction of the risk of collision between them, taking into account their current direction and speed of movement.

3.1. Object Detection Results

Firstly, the Convolutional Neutral Network (CNN), namely Region-Based CNN (YOLOv8 Tiny) for recognizing objects was applied. For the purposes of the experiment, a

network model called YOLO version 8, very popular both among researchers and in commercial applications, was used [16]. The choice of this solution results from the assumption of the practical applications of the experiment discussed, namely active monitoring of the production station in real time and the use of the YOLO model as one of the best connections in terms of high efficiency and speed in the detection of objects, namely over 30 images per second using a personal computer. The YOLO model uses many convolutional networks and its architecture is constantly evolving with subsequent versions. YOLO is provided to programmers as a ready-made programming library, where default operating parameters allow for effective training and then the detection of objects without conducting additional experiments. Additionally, this model was implemented in Python, which facilitates its use in further stages of work. The constantly growing popularity of the YOLO model is also due to the fact that its implementation is based on the PyTorch library, which can use CUDA technology, which is available in popular graphics cards with NVIDIA processors. This allows the hardware of machine learning computations to accelerate significantly on a regular personal computer.

Based on the training patterns prepared for each of the three networks, learning processes were carried out for each of them. Thanks to the capabilities of the YOLO model, an approach was used here in which we pass training patterns and a number of learning epochs to the model and it returns a file containing the definition of the trained network and information about the course of the learning process and its final effect. Based on the previous experience of the research team members, at the planning stage of this process, a number of experiments were carried out to determine the minimum number of training epochs so that, on the one hand, the network training process was not too long, and on the other hand, to obtain satisfactory network results. The YOLO model itself is very helpful in this matter, as the result returns two versions of the trained network: the variant where the network achieved the best results, and the variant after all training epochs. The experiment showed that, for the data from this experiment, with 400 epochs, the variant indicated as the best is obtained below this value, while increasing this number does not result in a significant increase in the effectiveness of the network.

The learning effectiveness for the YOLO model can be presented using a graph called a Confusion Matrix. "The confusion matrix provides a detailed view of the outcomes, showcasing the counts of true positives, true negatives, false positives and false negatives for each class" [17]. Such a chart is automatically generated by YOLO. Figures 7–9 present standardised versions of a Confusion Matrix appropriate for networks 1, 2 and 3. The normalised version represents the data in proportions and is simpler to compare the performance across classes. Figures 7–9 use the "human_body" label, which corresponds to "human torso" as used in the article.

This study illustrates Confusion Matrix graphs for those three views of cameras. It was observed that the main areas of misclassification are, excluding tools: (1) the human forearm, top and front view and (2) the human hand, left and front view. Analysis of the data presented in Figures 7–9 confirms that both the patterns enabling learners to perform the experiment and the selection of the YOLO tool itself were correct. For all three networks selected, the correctness of recognizing the problems identified was on average 80%, in the normalised graph, 1 corresponding to 100%, taking into account all classes of objects. Their data also show that the average correctness of classic slim tools is up to 36%; however, we do not take the class of objects, such as tools, into account in further stages of our research. So, the results indicated that the mean average precision value for all three networks was over 90% excluding this class.

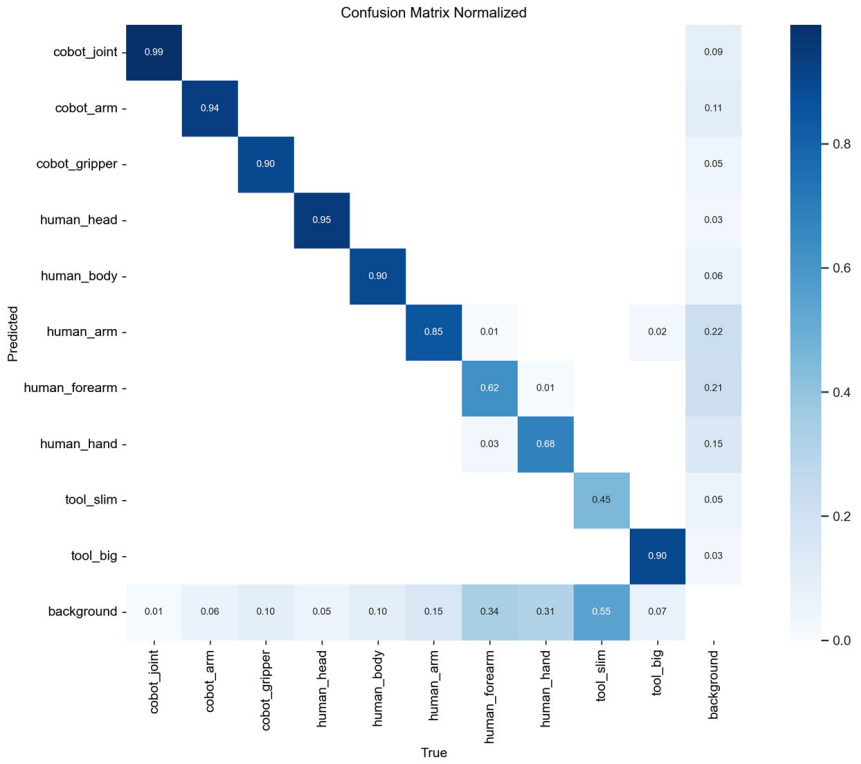


Figure 7. Normalised Confusion Matrix for network 1 (top view).

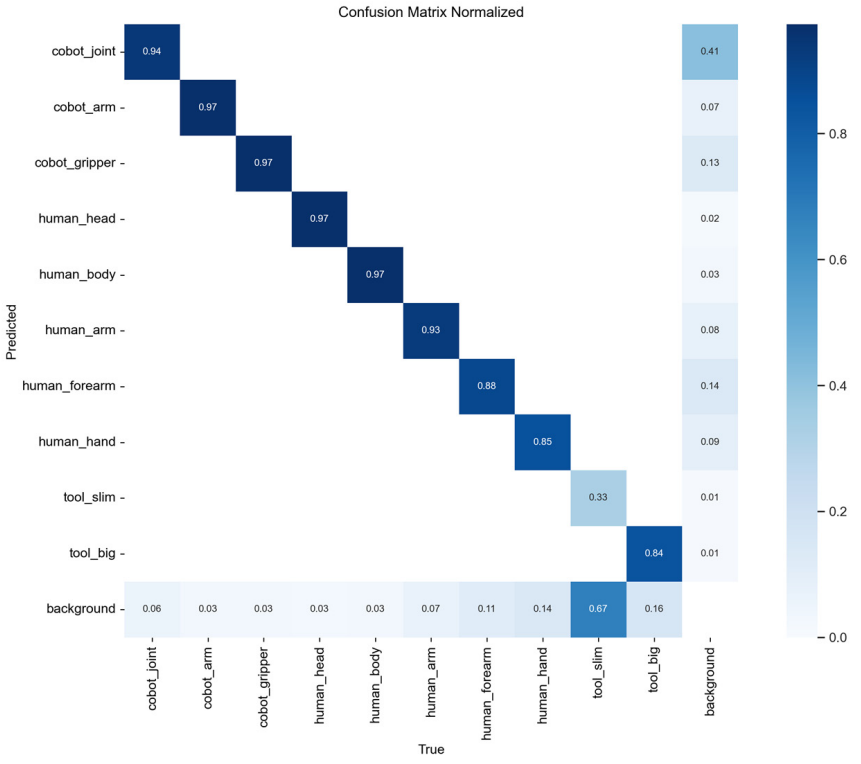


Figure 8. Normalised Confusion Matrix for network 2 (left view).

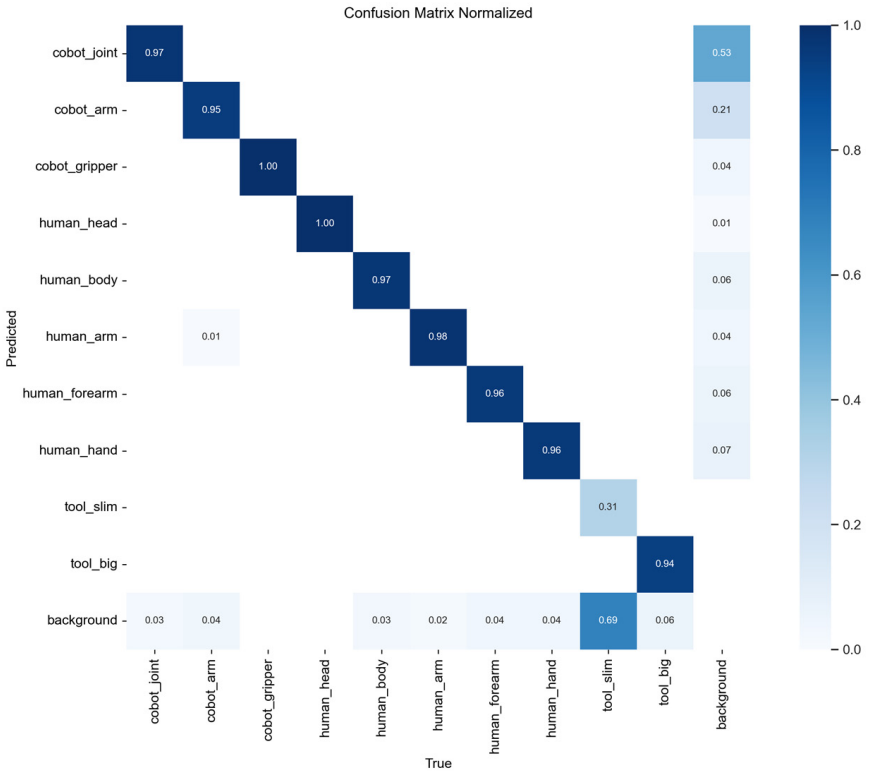


Figure 9. Normalised Confusion Matrix for network 3 (front view).

To implement the second and third stages of our research (Figure 1), it is necessary to prepare appropriate training patterns for the CNN, corresponding to the sequence of positions of individual cobot elements and body parts for situations where a collision occurred and when it was avoided. For this stage the data were only used from one view (top) of the camera.

3.2. Filtering the Objects, Separating and Tracking Cobots and Human Operators

Firstly, an application in Python was developed for recognising the set of objects expected for each frame of the video. Its effect consists of a text file with information about the frame number, the type of object recognised, the location of the rectangle surrounding it, namely the co-ordinates of its upper left and lower right vertex the unique object ID that was generated by the YOLOv8 model and graphic files corresponding to each frame of the movie analysed, marked with colours consistent with those used in the annotation process, the sequence number of the frame in the movie, the class of the object and its unique object ID. An example of such an image is presented in Figure 10.

The dataset obtained from YOLO, before presenting to a CNN classifier, was preprocessed and stacked according to the approaches described in Sections 2.3 and 2.4. At first, the algorithms Non-Maximum Suppression (NMS) for filtering the objects isolated in the previous stage, *k*-means Clustering method and Simple Online Real-Time Tracking (SORT) approach for separating and tracking cobots and human operators, were applied. Next, stacking and slicing techniques were used to prepare the CNN input data. All algorithms transforming data were implemented in Python 3.10.12.

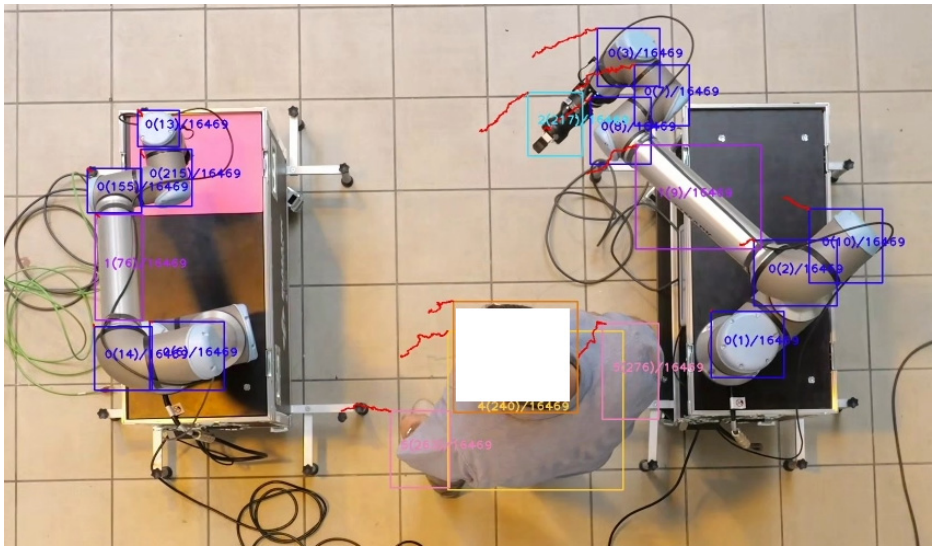


Figure 10. Sample image with the recognised object classes indicated.

3.3. Predicting Possible Collisions in HRC

A CNN for predicting possible collisions (stage 3 in Figure 1) was implemented in Python 3.10.12 using TensorFlow and Keras libraries in version 2.15. The code was run in the Google Colab environment with GPU support. In all experiments, the dataset containing preprocessed samples (Table 4) was split in the ratio 70:15:15 between training, validation and test sets. The validation set with an early stopping approach was used to avoid classifier overtraining. In order to determine the hyperparameters of a classifier, a series of experiments involving single- and double-layer CNN was carried out. Finally, CNN with one convolutional layer with 5 filters and 3×4 kernel size was chosen to predict collisions.

The efficiency of the proposed approach was confirmed by performing experiments involving the dataset collected in accordance with the procedure described in Section 2.1 and processed using the methods presented in Sections 2.2 and 2.3. Input data in the form of overlapping windows describing changes in the position of objects from the workspace over time and output data specifying the possibility of collisions, for training and testing the CNN classifier, were prepared as described in Section 2.4. A summary of the experimental results is presented in Table 5, containing the accuracies achieved in the training and testing phases. A more accurate assessment of the classifier's effectiveness is provided by the confusion matrices shown in Figure 11. As can be seen, the number of true positives and true negatives, namely the percentage of samples classified correctly, is high while the number of false positives and false negatives, namely the percentage of samples classified incorrectly, is low in both the training and testing phases.

Table 5. Training and testing accuracies of CNN classifier.

Phase	Accuracy
Training	97.2%
Testing	95.6%
Average	96.4%

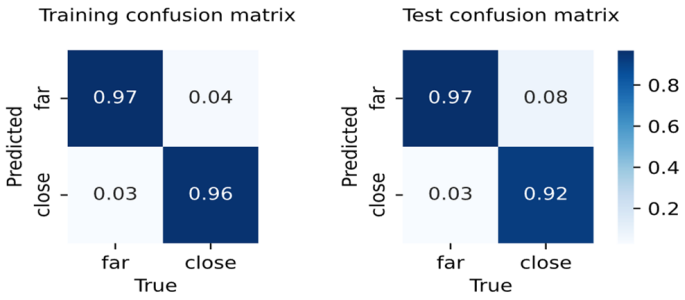


Figure 11. Confusion matrices summarising the performance of the classifier.

4. Discussion

According to [18], one of the challenges of HRC research is the acquisition of data vis à vis human behaviour in the workplace. In this article, we built three scenarios of possible operator behaviour in co-operation with cobots and then collected data via video recordings from three cameras (total 484 images) and performed research on predicting potential collisions in a given workplace. Based on our research results, it can be stated that:

- The YOLO v8 network showed a good level of recognising and indicating individual elements of cobots and operators in common HRC workplaces with an accuracy of 90%.
- The integration of YOLOv8 deep learning approach and CNN classifier enables early warning of potential collisions in common HRC workplace with an accuracy of 96.4%.
- The proposed new method significantly increases the effectiveness of HRC due to the possibility of predicting the occurrence of a collision, instead of detecting the presence of the operator in a certain safety zone, on the one hand by preventing the operator from colliding with the robot's arm and, on the other hand, eliminating the need to perform a safety stop, which disturbs the continuity of the ongoing process.
- Increasing safety levels within HRC is an important factor in the development of HRC, which allows the unique abilities and skills of people and robots to be combined in order to optimise production processes [19]. Therefore, this research was undertaken in a specific defined HRC area, in order to present how early warning can be given against a potential collision in such an HRC co-operation environment when a human accidentally enters such a space. The work shows three cases of the potential danger, i.e., unexpected entry of a human into the HRC zone: (1) entering the cobot's workspace facing forward, (2) crouching in the cobot's workspace, (3) turning around in the cobot's workspace. In works [20–22], comparisons of visual systems used for recognising human and robotic poses can be found, e.g., Azure Kinect Body Tracking, Intel RealSense D435i and also the YOLO network, Azure Kinect and Intel RealSense. The authors of this work used data from video recordings from three cameras due to the possibility of obtaining recordings of HRC from three points: from the front, from the side and from above.
- Currently it is difficult to compare the effectiveness of the results obtained, due to the specific and strictly defined common workplace of humans and cobots with specific behavioural scenarios. According to the notes in the introduction, the results in works [9,10] achieved an accuracy of 90% in recognising an object and 96.4% in predicting collisions, which can be said to be satisfactory. In [23] the scenario covers the HRC space, where an operator exchanges components with one cobot. The results yielded 91% accuracy with R-CNN when trained with synthetic data but only 78% accuracy with real data accuracy, respectively. In our research only real data were used for both training and testing deep learning models. This is particularly important because many works use synthetic or synthetic and real data, because there is no public dataset available [19].

The main limitation of our research is the dataset used for the second stage of our research, which was collected in a given period under strictly three scenarios. We plan to create the synthetic dataset in order to increase the amount of data for a model in order to predict the possible collisions in HRC, which should result in greater accuracy when predicting collision. Moreover, in the next stages of research, in order to improve the model for predicting potential collisions, we plan to work with data from subsequent views: side and front. In addition, further objects, such as tools, will also be considered in further work. The next limitation is the use of specific artificial network structures. We plan to conduct further research experiments with various neural network architectures. Moreover, based on the research results, we will continue our work on the scalable system that can be used in an environment containing various numbers of cobots. We plan to apply an individual artificial network for each cobot, which will increase the flexibility of the system.

5. Conclusions

The research results presented are important in the field of safety issues in human–robot collaboration. Thanks to applying our developed method combining applications of the Region-Based CNN (YOLOv8 Tiny) for recognising objects (stage 1), the NMS for filtering the objects isolated, the *k*-means Clustering method and SORT approach for separating and tracking cobots and human operators (stage 2) and finally CNN for predicting possible collisions (stage 3), it is possible to detect collisions early in the common work space of the cobot and the operator. This is particularly important in the context of building an enterprise according to the Industry 5.0 concept, where harmony and sustainability are very important in the implementation of production. Promising research results indicate that this is undoubtedly an area of research that should be continued in subsequent work. However, acquiring real data is a very expensive and time-consuming process. In further work, we will conduct research experiments regarding the evaluation of deep learning models with both synthetic and real data sets and will compare the results regarding the accuracy of such models received.

Author Contributions: Conceptualization, J.P.-M., A.D., G.P. and I.P.; methodology, J.P.-M., A.D., G.P. and I.P; software, A.D., G.P. and I.P.; validation, J.P.-M., A.D., G.P. and I.P.; formal analysis, A.D., G.P. and I.P.; investigation, A.D., G.P. and I.P.; resources, J.P.-M., A.D., G.P. and I.P.; data curation, J.P.-M., A.D., G.P. and I.P.; writing—original draft preparation, J.P.-M., A.D., G.P. and I.P.; writing—review and editing, J.P.-M., A.D., G.P. and I.P.; visualization, A.D., G.P. and I.P.; supervision, J.P.-M.; project administration, J.P.-M.; funding acquisition, J.P.-M. and A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by a program of the Polish Ministry of Science under the title 'Regional Excellence Initiative', project no. RID/SP/0050/2024/1.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Bioethical Committee of University of Zielona Góra, Poland (Nr.19/2023 by 25 October 2023).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Siciliano, B.; Khatib, O. *Springer Handbook of Robotics*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2016.
2. La Fata, C.M.; Adelfio, L.; Micale, R.; La Scalia, G. Human error contribution to accidents in the manufacturing sector: A structured approach to evaluate the interdependence among performance shaping factors. *Saf. Sci.* **2023**, *161*, 106067. [CrossRef]
3. Giallanza, A.; La Scalia, G.; Micale, R.; La Fata, C.M. Occupational health and safety issues in human-robot collaboration: State of the art and open challenges. *Saf. Sci.* **2024**, *169*, 106313. [CrossRef]
4. Ko, D.; Lee, S.; Park, J. A study on manufacturing facility safety system using multimedia tools for cyber physical systems. *Tools Appl.* **2021**, *80*, 34553–34570. [CrossRef]

5. Zhang, S.; Li, S.; Li, X.; Xiong, Y.; Xie, Z. A Human-Robot Dynamic Fusion Safety Algorithm for Collaborative Operations of Cobots. *J. Intell. Robot. Syst. Theory Appl.* **2022**, *104*, 18. [CrossRef]
6. Liu, H. Deep Learning-based Multimodal Control Interface for Human-Robot Collaboration. *Procedia CIRP* **2018**, *72*, 3–8. [CrossRef]
7. Liu, Z.; Liu, Q.; Xu, W.; Liu, Z.; Zhou, Z.; Chen, J. Deep Learning-based Human Motion Prediction considering Context Awareness for Human-Robot Collaboration in Manufacturing. *Procedia CIRP* **2019**, *83*, 272–278. [CrossRef]
8. Wang, P.; Liu, H.; Wang, L.; Gao, R.X. Deep learning-based human motion recognition for predictive context-aware human-robot collaboration. *CIRP Ann.* **2018**, *67*, 17–20. [CrossRef]
9. Rodrigues, L.R.; Barbosa, G.; Filho, A.O.; Cani, C.; Dantas, M.; Sadok, D.; Kener, J.; Souza, R.S.; Marquezini, M.V.; Lins, S. Modeling and assessing an intelligent system for safety in human-robot collaboration using deep and machine learning techniques. In *Multimedia Tools and Applications*; Springer: Berlin/Heidelberg, Germany, 2022; p. 81.
10. Liau, Y.Y.; Ryu, K. Status Recognition Using Pre-Trained YOLOv5 for Sustainable Human-Robot Collaboration (HRC) System in Mold Assembly. *Sustainability* **2021**, *13*, 12044. [CrossRef]
11. Pajak, G.; Krutz, P.; Patalas-Maliszewska, J.; Rehm, M.; Pajak, I.; Dix, M. An approach to sport activities recognition based on an inertial sensor and deep learning. *Sens. Actuators A Phys.* **2022**, *345*, 113773. [CrossRef]
12. Pajak, I.; Krutz, P.; Patalas-Maliszewska, J.; Rehm, M.; Pajak, G.; Schlegel, H.; Dix, M. Sports activity recognition with UWB and inertial sensors using deep learning approach. In Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Padua, Italy, 18–23 July 2022; pp. 1–8. [CrossRef]
13. Ultralytics, Introducing Ultralytics YOLOv8. Available online: https://docs.ultralytics.com (accessed on 10 October 2023).
14. Neubeck, A.; Van Gool, L. Efficient Non-Maximum Suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; Volume 3, pp. 850–855.
15. Bewley, A.; Ge, Z.Y.; Ott, L.; Ramov, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 23rd IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468.
16. Sirisha, U.; Praveen, S.P.; Srinivasu, P.N.; Barsocchi, P.; Bhoi, A.K. Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection. *Int. J. Comput. Intell. Syst.* **2023**, *126*, 18. [CrossRef]
17. Ultralytics. Performance Metrics Deep Dive. Available online: https://docs.ultralytics.com/guides/yolo-performance-metrics/ (accessed on 10 October 2023).
18. Mukherjee, D.; Gupta, K.; Chang, L.H.; Najjaran, H. A Survey of Robot Learning Strategies for Human-Robot Collaboration. *Ind. Settings Robot. Comput. Integr. Manuf.* **2022**, *73*, 102231. [CrossRef]
19. Gross, S.; Krenn, B. A Communicative Perspective on Human–Robot Collaboration in Industry: Mapping Communicative Modes on Collaborative Scenarios. *Int. J. Soc. Robot.* **2023**. [CrossRef]
20. Ramasubramanian, A.K.; Kazasidis, M.; Fay, B.; Papakostas, N. On the Evaluation of Diverse Vision Systems towards Detecting Human Pose in Collaborative Robot Applications. *Sensors* **2024**, *24*, 578. [CrossRef] [PubMed]
21. De Feudis, I.; Buongiorno, D.; Grossi, S.; Losito, G.; Brunetti, A.; Longo, N.; Di Stefano, G.; Bevilacqua, V. Evaluation of Vision-Based Hand Tool Tracking Methods for Quality Assessment and Training in Human-Centered Industry 4.0. *Appl. Sci.* **2022**, *12*, 1796. [CrossRef]
22. Rijal, S.; Pokhrel, S.; Om, M.; Ojha, V.P. Comparing Depth Estimation of Azure Kinect and Realsense D435i Cameras. *Ann. Ig.* **2023**.
23. Wang, S.; Zhang, J.; Wang, P.; Law, J.; Calinescu, R.; Mihaylova, L. A deep learning-enhanced Digital Twin framework for improving safety and reliability in human–robot collaborative manufacturing. *Robot. Comput. Integr. Manuf.* **2024**, *85*, 102608. [CrossRef]