engineering proceedings

Proceeding Paper

A Novel Design of Intelligent Floor Cleaning Robot Using Deep Learning Technique

P. Prakasham [1], T. Sangeetha [2,*], S. Malathy [3], A. Jameer Basha [4] and P. Jeevitha [4]

[1] Visteon Technical and Service Center Pvt Ltd., Chennai, Tamil Nadu, India; prakashampersonaldata@gmail.com
[2] Department of IT, Sri Krishna College of Technology, Coimbatore, Tamil Nadu, India
[3] Department of IT, KPR Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India; malathy.s@kpriet.ac.in
[4] Department of CSE, Hindusthan Institute of Technology, Coimbatore, Tamil Nadu, India; jameer@hit.edu.in (A.J.B.); jeevitha.p@hit.edu.in (P.J.)
* Correspondence: sangeetha131989@gmail.com

Abstract: Floor cleaning plays a major role in all places, like homes, offices, etc., in the olden days by humans. Long-term cleaning makes the person get tired, and they cannot be involved in the deep and neat cleaning process. To make cleaning work easier and tidier, an automatic floor cleaning robot has been introduced through the AI technique. The digital camera makes the autonomous system navigate accordingly based on environmental analysis. The main drawback of the system is that the robot automatically turns in another direction whenever it finds obstacles like different kinds of doors, poles of furniture, cables, small garments lying on the surface of the floor, etc. The objective of the system is to move the autonomous robot freely even if it finds the object to make the system involved in deep cleaning. This paper mainly proposes a vision-based YOLOv5 framework to detect the object during navigation. The custom dataset is annotated from the scrape using different labels for 300 images. A novel approach proposed in this model is multi-class object detection using the YOLOv5 and 6-DOF with the help of a manual dataset. This work proposes a system to develop a highly accurate automated robot system using the Computer Vision Annotation Tool (CVAT) and deep learning algorithm. The high-quality camera is fixed in front of the robot. Video is captured, and it is automatically converted into an image using the Python script. The image is annotated manually using the CVAT Tool which is processed using the deep learning technique of the YOLOv5 algorithm. The incorporation of obstacle avoidance capabilities prevents collisions with furniture and walls, contributing to a hassle-free cleaning experience with the concept of 6-DOF. This system makes the cleaning process efficient even when it finds obstacles, the unsurpassed model achieved a mean average precision (mAP) of 93%.

Keywords: deep cleaning; Computer Vision Annotation; Yolov5; deep learning

Citation: Prakasham, P.; Sangeetha, T.; Basha, A.J.; Jeevitha, P. A Novel Design of Intelligent Floor Cleaning Robot Using Deep Learning Technique. *Eng. Proc.* **2024**, *5*, x. https://doi.org/10.3390/xxxxx

Academic Editor(s):

Published: 26 November 2024

1. Introduction

The driverless floor-cleaning robot is intended to sweep, gather particles, and cleanse the ground. Manual floor cleaning is an unpleasant, monotonous, and time-consuming work. Cleaning plays a major task in day-to-day life to lead a healthy life. People can easily clean their place efficiently when space is less. People with huge crowds in places like hotels, malls, and hospitals require manpower to clean it which is time-consuming and inefficient. Conventional manual cleaning techniques are labor-intensive and inefficient since they frequently take time and energy. 95% of the people are mostly working. They feel it is very tough to maintain the household effortlessly. There are several drawbacks to keeping domestic workers. To keep all these drawbacks in mind the researcher developed an automatic floor-cleaning robot with artificial intelligence. To make the work easy the autonomous floor cleaning robot was introduced in the market with the zero-man effort. These autonomous robots were introduced in the 1990s. The majority of the subsequent versions were able to solve the obstacle-avoidance issues with the earlier models. In recent years, all autonomous robots function better, but certain model updates are still needed to increase performance. Even though it contains many advantages still its nave few disadvantages. The main drawback of the automatic floor cleaning robot is difficult to clean the dust, and dirt under the object in a clean manner. so the accuracy of the cleaning process is reduced. To increase the efficiency, speed, ease of use, limited obstacle recognition, and navigation challenges, various artificial intelligence and deep learning techniques [1] are employed will make the right choice to maintain their environment hygienically.

The robot that cleans the floor with the camera that records video and turns it into an image has a new concept.

Spring cleaning is mainly done through the image using the image processing technique.Recently, floor-cleaning robots have experimented with using camera-based which is the enhancement of existing navigation sensors. Subsequently, these floor-cleaning robots previously had cameras, so it makes it wise to utilize them for missions in addition to navigation. The major problem of the autonomous robot is deep cleaning even if it finds obstacles like different kinds of doors, poles of furniture, cables, small garments on the floor, etc through the concept of six degrees of freedom. This makes cleaning efficient and fulfills diverse cleaning tasks.

The autonomous cleaning robot can identify the cleaning requirement needed to clean the floor based on the dirt residing in it. To guarantee effectiveness and dependability, the inspection robot determines which areas need to be cleaned and directs the appropriate cleaning product to the designated areas while avoiding needless coverage. Robots mainly work with computer vision and artificial intelligence. There are several advantages to creating the floor cleaning robot. The robots are affordable which is designed using machine learning algorithms like Retina Net [2], yolo [3], closed-circuit television (CCTV)-based systems, region-based Convolutional Neural Networks, SSD, etc.… which makes people expect cleaning robots to have the following features: human convenience, reliability and security, energy efficiency, coverage duration, and coverage of the entire area.

Here there is a need for a novel approach to overcome the above-mentioned drawbacks. To keep all the drawbacks in mind this article proposes a novel technique to clean and spot the dirt even if it finds some obstacles in indoor navigation using the computer vision-based deep learning algorithm YOLO. The work is divided into three modules:

- Real-time video has been taken and converted into images using the Python script. The converted image is annotated using the Computer Vision Annotation Tool (CVAT) of rectangular bounding box using the 2-point method.
- The YOLO algorithm is trained using the dataset which is annotated to identify objects like furniture, small Garments, doors, and Cables and maneuver accordingly on the floor
- Robot move turns atomically when it finds obstacles using six degrees of freedom
- Validate the accuracy of the object detection using values obtained in the mean average precision (Map).

2. Literature Survey

Ramalingam. B et al. [4] propose a method for the floor cleaning robot mostly used in community places to perform regular cleaning processes with various cleaning accessories like the mop pad, scrubber, and brush. The model has been developed using deep learning algorithms, path planning using the optimal way, and a closed-circuit Television (CCTV) network with the RGB-D vision sensor. The main aim of the robot is to clean only the dirt area instead of all the places. The two methods of tracking foot traffic patterns and looking for stains and rubbish on the floor were used to locate the cleaning area. Here, the high-person-traffic area was situated using a deep Simple Online and Real-time Tracking (SORT) humanoid following process, and the unclean zone was positioned with an object detection framework Single Shot Detector (SSD) MobileNet. The optimal shortest path is calculated by using the algorithm evolutionary-based optimization which makes the robot travel efficiently to the designated dirt location. SSD MobileNet algorithm improved the performance of 90%,15% of the steering period, and 10% of the energy utilization.

Ramalingam. B. et al. [5] proposed an autonomous floor-cleaning robot to detect and evade the hard-to-clean robot in the huge fluid expulsion area. The algorithm is required to find the detection and classification is done by using the algorithm deep Convolutional Neural Network (CNN) with Support Vector Machine (SVM) cascaded technique. The image is required for detection and classification is done by using the Single-Shot Multi-Box Detector MobileNet CNN framework. Deep computer vision based on deep learning architecture with many hidden layers is used to find hard solid debris and liquid spills. The object detection algorithm CNN is employed to extract the feature and bounding box. Most of them are related to solid cleaning. This work contains two of solid and liquid cleaning and produces 96% efficiency with previous computer vision-based debris categorization methods and the most recent deep learning approach. Finally, the comparison is done by various algorithms with non-deep learning techniques to show the proposed model produced more accuracy.

Canedo.D et al. [6] introduced a floor-cleaning robot with a digital camera used to identify the dirt in the environment using the computer vision-based technique of YOLOv5. The dirt is classified as two dissimilar kinds of messiness solid and liquid.The main aim of the model is to clean the environment only when it finds dirt spots on the floor to conserve resources and energy. The synthetic data of 141 solid and 15 liquid samples of dirt is taken to train the model instead of the real dataset. The synthetic data is processed using the YOLOv5 models. The main drawback of the system it cleans the dirt only in the spotted area and trains the model with less dirt image in the floor with the object and no object. Sometimes dirt presented under the object makes it difficult to do a deep clean. As the robot used for cleaning must collect the training dataset and manually annotate it afterward, their method is also more expensive. In order to identify dirt based on size, their method also needs an additional SVM classifier. Due to its multi-scale prediction capability, YOLOv5 solves this issue. The model is compared with the various YOLOv5 using the real dataset (ACIN) which attained a mean average precision (mAP) of 87.4% accuracy.

Tian. M et al. [7] evolved a based-on-vision autonomous underwater debris-cleaning robot. During the underwater cleaning, the people may be affected by the marine organism and sometimes they may be hooked by the corals. The autonomous vehicle has a bi-directional camera connected for location tracking and identifying objects. Nearly 6200 images were utilized for instruction and 400 images were used to examine the model using deep learning. single-stage detection YOLOv4 algorithm model is developed with quick and accurate object identification. The algorithm works with the framework of a neural network to detect the debris. When detecting objects accuracy and speed are the two significant factors. The traditional methods cannot meet the expected result. The upgraded YOLOv4 has great detection speed and accuracy, as demonstrated by the experiment result. The mean average precision (mAP) of 95.099% is achieved at a maximum estimated detected speed of 66.67 frames/s.

D. Padilla Carrasco et.al. [8] designed a model to manage the parking in large cities as well as in the large parking area. It takes a long kilometer for the people to find a parking space. The problem of large parking slots was managed using the sensor. However, the magnetometer-based sensor is used to find the slot accurately which minimizes the battery life. so the vacant finding in the parking area became very challenging to the researcher. The main obstacles to vehicle detection include significant light fluctuations, widespread occlusion, and wide differences in object scales.

The YOLOv5 has been released to improve the identification of fragile and delicate objects to overcome all of the disadvantages. One of the advantages is that it works faster due to the different sections to process the data. The proposed model works with the two different sections to extract the tiny object detection like a car. PKLot is the dataset for the suggested model's validation and training with the multi-scale module. The 4474 images almost with the 10 parking slots used to annotate the data with coco format annotations. Furthermore, we deduced a 30 fps slower detection speed in comparison to the YOLO-v5-L/X profiles. Furthermore, there was a noteworthy 33% enhancement in the little vehicle recognition ability when compared to the YOLO-v5-X profile.

3. Proposed Methodology

Cleaning is a routine task given to cutting-edge technology such as robotics and artificial intelligence. Many different types of robots have been built, yet they are unable to work together to locate dirt. The world is moving to automation. People are running towards their goals. so, it isn't easy to find their time to be involved in the cleaning activity. This has led to the researchers designing the automatic floor-cleaning robot. Artificial intelligence is needed to enable autonomous operation of the robot without human involvement. The system should be low-cost. Many researchers used sensors to identify the object. This paper mainly focused on the 360-degree rotating camera instead of the sensor. To identify the object's deep learning algorithm is needed. Object avoidance makes the robot move on the floor freely and quickly even when it finds obstacles computer vision, object detection is the most important and difficult problem require an effective technique. Using deep learning provides several ways of object detection. There are several prevalent algorithms used for object detection SSD [9], R-CNN [10], Fast R-CNN [11], SqueezeDet [12], MobileNet [13], and YOLO [14]. A description of the suggested system is shown in the Figure 1. The system's primary goal is to prevent collisions which ensures the cleaning is dirt-free. The camera is fixed in front of the robot to capture the video. The video is converted into an image. The deep learning algorithm is proposed to detect the obstacles also avoid collision during the cleaning. A robot automatically changes direction using six degrees of freedom when it finds obstacles. so the robot moves to the dirt area without colliding. Several papers discuss dirt identification. This paper mainly focuses on object detection like furniture, small Garments, doors, and Cables and maneuvering accordingly on the floor. The 6-DOF is implemented to avoid collision and make the system utilized for deep cleaning. The accuracy is calculated using the mean average precision (Map). The several metrics are considered and made a comparison with the several algorithms like SSD, SqueezeDet, MobileNet, Fast R-CNN, and R-CNN [10].

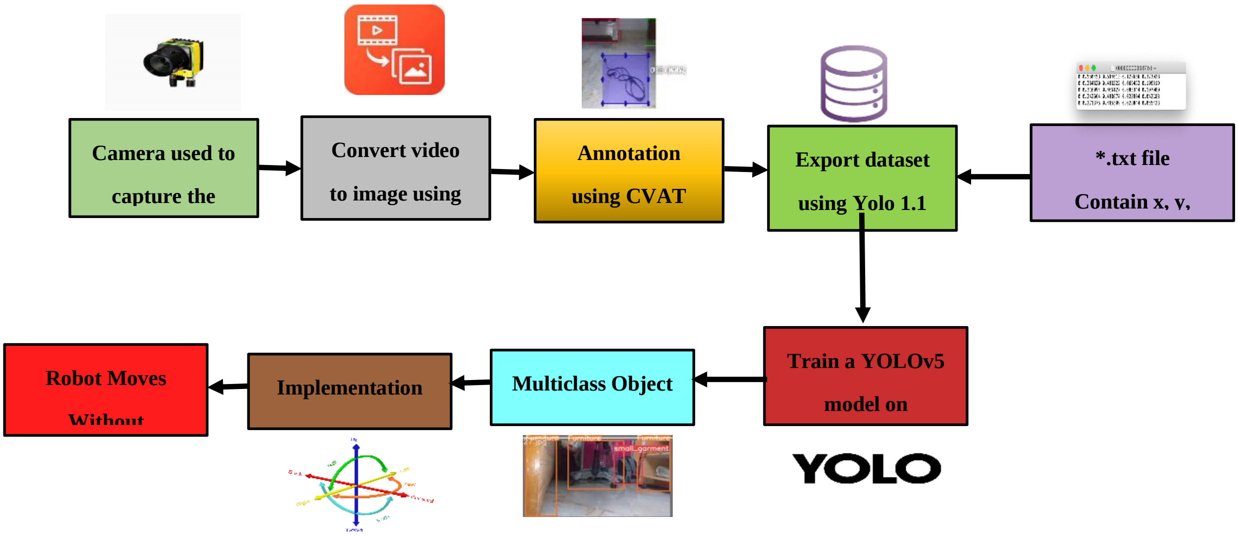

Figure 1. Overall Proposed Model.

3.1. Custom Data Preparation

Video is nothing but several images combined that form a video. Video is captured in various rooms in the home. In the multimedia processing video is transformed to images using the python script. Python script provides an efficient way of converting a video into the images using the OpenCV library. These images are saved into a separate folder and numbered continuously. The test dataset is formed with more than 3000 images used to train the model using YOLOV5.

3.2. Drawing Bounding Box

Image annotation can be done with the open-source Computer Vision Annotation Tool (CVAT) depicted in Figure 2. It supports the supervised learning of object classification, segmentation, detection, and 3D segmentation. To train the deep neural network AI required large annotated images to train the model. Manual annotation required more time. CVAT offers automatic annotation to speed up the annotation process. CVAT offers different types of annotation cuboid, rectangle, polygon, points, polyline, and tag. Create a new task. To begin the annotation, create the labels required for annotation. Choose the shape and color used in the label. Add an attribute to set up the property. While annotating choose the required annotation method. Here we use a 2-point rectangle box to annotate the image. finally, we have to export the data based on the algorithm used once the annotation is over. Here we going to use the yolov5 which should export the dataset through the CVAT. The bounding box takes x&y coordinates from the lower right and upper left corner which calculates the box center to corner and box corner to center.

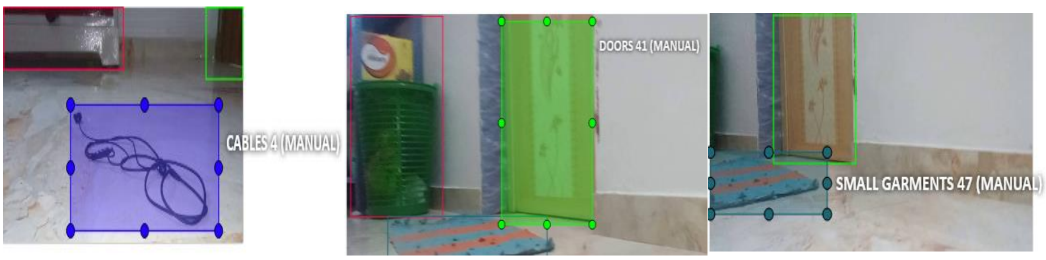

Figure 2. Annotation Using CVAT.

3.3. System Design for Object Detection

The technique used in the system is YOLOV5 which is computer vision-based object detection. Computer vision performs the tasks of object detection, image Classification, Captioning, and reconstruction. Object detection is the identification of the object in an image or object in a camera accurately. It is divided into two types such as single-shot detectors and two-stage detectors. "You Only Look Once" known as the YOLO is a well-liked object detection model because of its quick response time and effectiveness. A fully convolutional neural network (CNN) is used by the single-shot detector to process the image. Here the YOLO used the single-stage detector. Single shot detector best one for real-time application. One advantage of using the YOLO is it learns the feature automatically through the input image during training. We do not need to provide any features to train the model. YOLOV5 is simple, faster, and easy to detect the different objects by their class. YOLOV5 is proposed to identify the object and obstacles during cleaning and move the robot to an environment with no obstacle which follows the computer vision-based object detection method. It is enhanced from the YOLOV4 and works with a speed of 140 fps. It consists of five types large, extra-large, nano, small, and medium. This algorithm detects the object with great accuracy. Algorithm used in the various applications with the box prediction and the class prediction. It consists of three parts backbone, head, and neck. The features from the picture are aggregated by Backbone at various granularities. The visual features are combined and mixed in the head to advance it to the forecast. Head detects different sizes of images as small, medium, and large. The extracted feature map is sent to the neck works based on the PANet. The result of the head is moved to the next to perform bounding box and class prediction. The input image is separated into an S*S grid. The grid cell is used to forecast the object based on the center of an object in the grid cell. This forecasts the confidence score and bounding box. Based on the confidence score the exactness of the bounding box to the object detection can be done. Even if it contains the multiple bounding box the YOLO predicts only one bounding box created on the highest current IOU. The confidence calculation processes each pixel to calculate the multidimensional array which includes object class, height, width, box coordinates, etc. This makes the YOLO algorithm find the object in the given image. This algorithm was implemented using the Google Colab. The model should train with a lot of images which is used to find the different kinds of things in the image. By using the YOLO, the output is predicted with a rectangle box and object label prediction. If the robot finds any obstacles it automatically moves around the poles using the concept called six degrees of freedom (6-DoF) [15]. The human head is the best example. It operates mainly on translation and rotation.6-DoF is the representation of the movement of the object in 3D space and is controlled through the translation which are the x-axis, y-axis, and z-axis. The 6DoF translates in either of the directions as up/down, right/left, and forward/backward. Hence the 6DoF makes the robot without collision.

3.4. Training and Validation

The model was trained using the Google Colab. Figure 3 shows the correlogram label which explores the connection between the different class labels in the custom dataset. It is the representation of a 2D histogram showing the data in the x-axis against the y-axis of a matrix or heatmap. The label can help to identify the pattern in the image across the different labels. Every value in the matrix denotes the correlation between the pair of labels. Calculation of correlation is accomplished with a statistical formula. Correlograms consist of high-intensity and low-intensity colors. The high intensity shows a positive correlation, which indicates the co-occurrence between the labels. The low intensity suggests the reduced likelihood of the dataset of small garments, doors, furniture, and cables. The information is presented in Figure 3 to understand the characteristics of the dataset that is employed in the model's training and assessment. It is a multi-variate analysis to check the randomness.

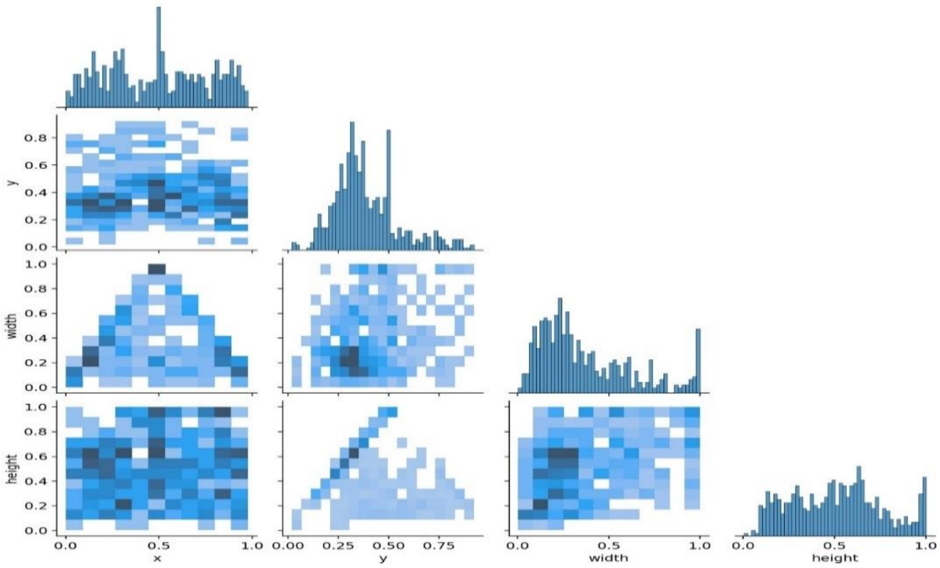

Figure 3. Representation of correlogram to the custom dataset.

Figure 4 shows the 2D histogram representation of the custom dataset. Figure 4a shows the occurrence of the annotation of labels used in the manual dataset. Figure 4b Illustrations of spatial distribution and dimension of bounding boxes in the given dataset. It shows Figure 4c,d depicts the distribution pattern of the dataset in terms of the X, width, and Height. It provides the representation of the bounding boxes in the image and their sizes. This study aims to bring the arrangement of the bounding boxes in the dataset. Some region has a higher concentration. with the help of the model, the object can be recognized easily by the given input images.3000 images are involved to train the model by using the object detection of the YOLOv5.

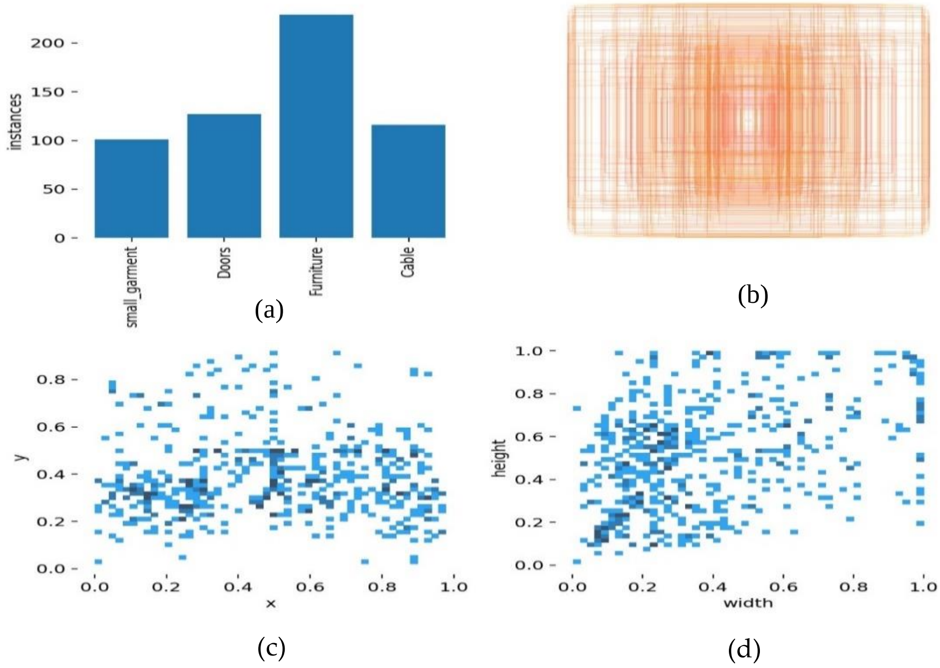

Figure 4. 2D representation of the individual custom dataset.

4. Results and Discussion

Metrics for Evaluation

Utilizing evaluation metrics, the training's completion is assessed. These evaluation metrics are used to identify the accuracy in the given input image. Here we going to detect the evaluation metrics such as precision, confusion matrix, Average Precision, recall, Mean Average Precision (mAP), and F1 Score.

Figure 5 confusion matrix is mainly used to visualize in which the classified a value as compared to the ground truth. The algorithm classifies the image based on the user-defined labels. It is a kind of table layout that plots all the predicted against the actual value. The row represents the actual class in the dataset. The column represents the predicted class. Every image produces a pair of classification tags used to find out the frequency of the occurrences. A number of the total count is mapped to the corresponding index of the confusion matrix. These values are used to calculate the table of the True Negative (TN), True Positive (TP), False Negative (FN), and False Positive (FP). The confusion matrix which offers Recall, Precision, and F1-score. A matrix grid is formed based on the number of the class labels. Accuracy is estimated using the sum of all true values divided by the total value. It mainly assesses the performance of the object detection model. It shows how the model predicts the classes rightly and wrongly based on the prediction.3000 images are used for the multi-class classification.

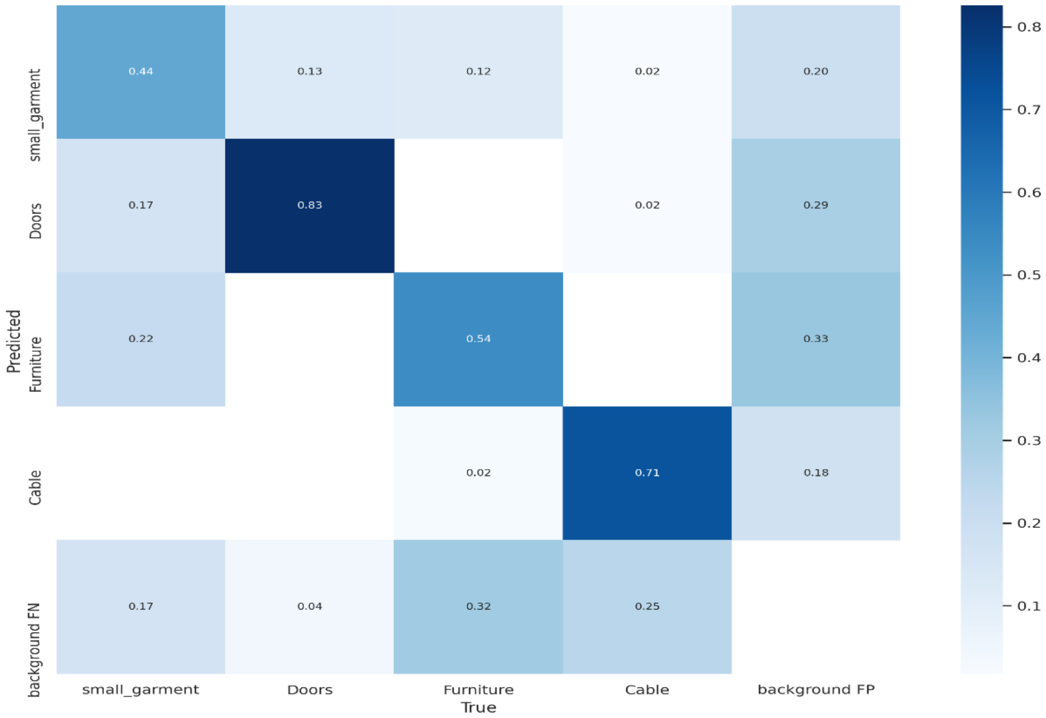

Figure 5. Confusion matrix using YOLOv5 for object detection.

The below Equation (1) calculates the value of accuracy. The wrong prediction is calculated using the formula called error rate. It represents the number of incorrect predictions calculated using Equation (2).

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (1)$$

$$\text{Error rate} = (\text{FP} + \text{FN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (2)$$

Figure 6 shows the results of various metrics. Model training parameters are epochs and stack size. The prepared dataset is iterated based on the epochs to detect the class. Epochs are mainly used to reduce the loss function. This is mostly used in the training phase. The result produced both training and validation sets. The three different types of loss produced in the results are box, objectness, and classification. The box loss shows how successfully the method locates an object's center and how efficiently the object is covered by the expected bounding box. In essence, objectness is an indicator of probability for the presence of an object in a suggested area of interest. Classification loss determines how well the object detects the correct class to the given input.

Figure 6. Loss and mAP results.

Figure 7 measures how the model predicts the positive class correctly. precision measured with the value range from 0 to 1 or in percentage. The value derived from the positive by the total positive present in true and false value. The accuracy of the model is correctly determined with a percentage of 0. 874. It detects and diminishes the false positive. It determines how accurate the prediction is. Value is considered with the Equation (3).

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (3)$$

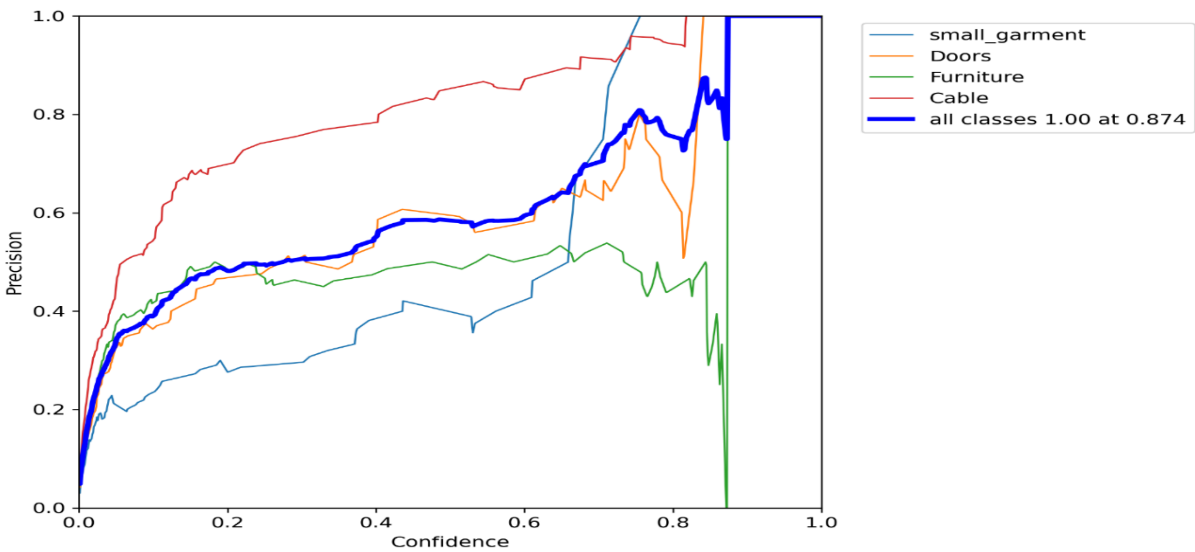

Figure 7. Confidence vs. precision result.

Figure 8 which shows the result between the confidence and recall. The result is derived with the Equation (4). It checks all the relevant objects are detected. The ability of the model to identify every occurrence of an object in the images. It detects the threshold of 0.94 for all classes confidence threshold of 0.0. This model detects TP detections across all classes 94% of the time with the Equation (4)

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (4)$$

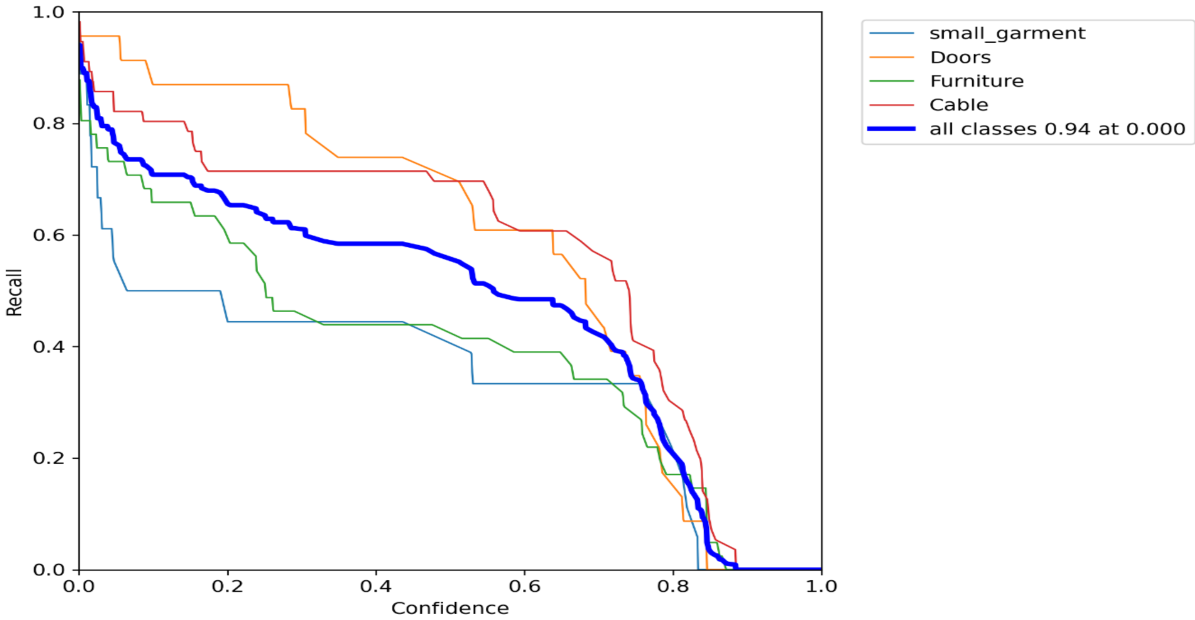

Figure 8. Confidence vs. recall result.

Figure 9 deals with Average precision (AP) can be calculated by comparing the recall curve at various confidence levels to the area under precision. Increasing the threshold reduces the detection risk of over-detecting objects. It denotes how many true positive detects in the analysis were detected by the model. The range for average precision is between 0 to 1. The average precision is calculated by using the below Equation (5)

$$\text{Average Precision} = \int_{a=0}^{1} p(a)da \quad (5)$$

The mean average precision is determined from average precision across all the classes denoted by Equation (6)

$$\text{mAP} = \sum_{i}^{k} AP_{\mathrm{i}} \quad (6)$$

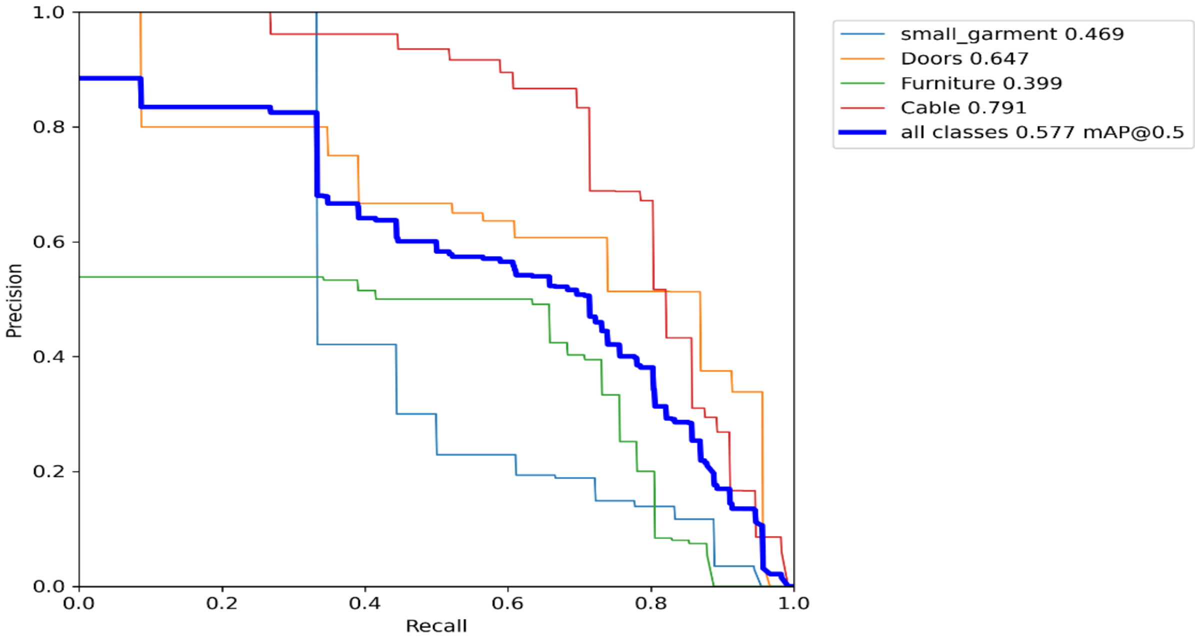

Figure 9. Precision vs. Recall result.

The F1 score is calculated for all the classes with the below-mentioned Equation (7). The value is measured with the mean value of recall and precision. It balances between the false positive and false negative. The value ranges between 0 to 1. This is shown in Figure 10.

$$F1 = 2 * (\text{precision} * \text{recall})/(\text{precision recall}) \quad (7)$$

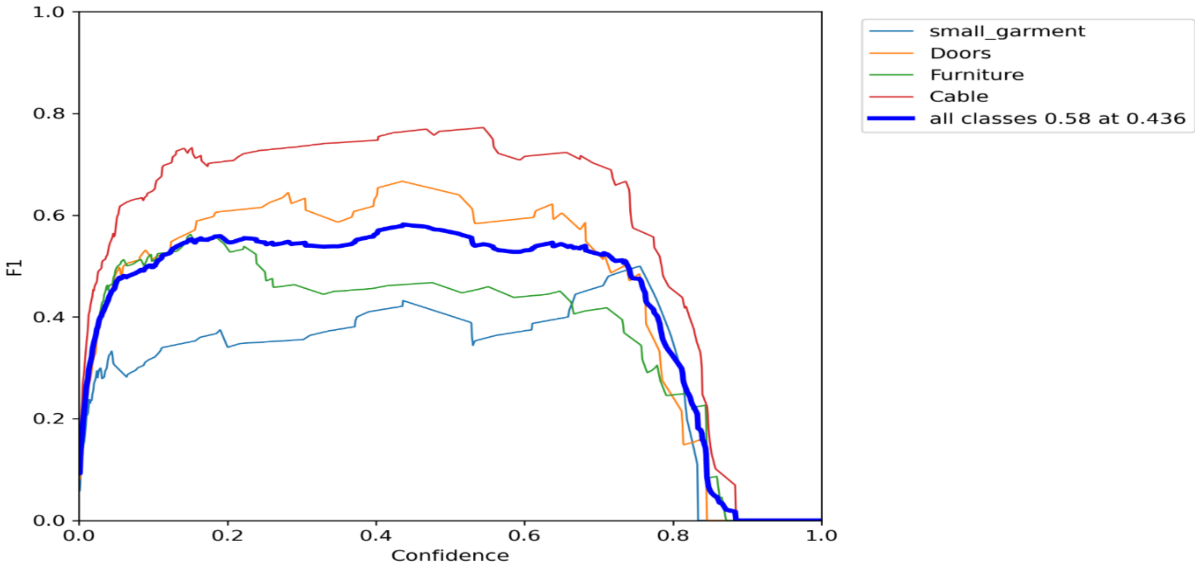

Figure 10. F1 Score Result.

Annotation is done with the CVAT Tool mainly utilized to train the model. Figure 11 illustrates the batch of detection results to some test images of the proposed work. The test data is mainly used to identify the accuracy of the model through performance. The model is tested with more than 3000 images it accurately identifies the annotated classes. This shows the YOLOV5 object detection output. Due to the test dataset's relatively low noise and visible objects, our model performs quite well. However, to reach good accuracy, you might need to train the model for a significant quantity of epochs and a sizable number of training photos.

Figure 11. Test batch using the Proposed model of YOLOv5.

The YoloV5 is compared with various algorithms such as Faster RCNN ResNet, Inception, SSD V2, YoloV4 and YoloV5.By comparing all the evaluation metrics YoloV5 shows better results compared to all algorithms shown in Table 1 and Figure 12.

Table 1. Experimental analysis result.

	Faster RCNN ResNet	Faster RCNN Inception	SSD V2	Yolov4	Yolo v5
Precision	95.96	94.45	96.63	96.2	97.35
Recall	94.34	94.13	96.29	95.91	96.92
F1Curve	94.11	94.01	96.17	95.43	96.74
Map@0.5	95.04	94.24	96.44	96.03	97

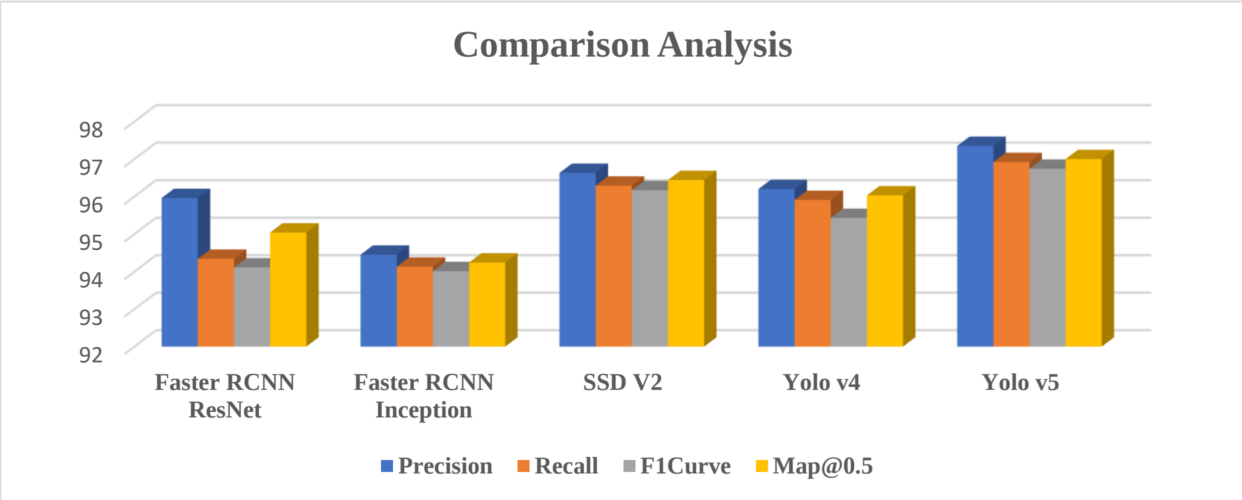

Figure 12. A comparison of the proposed model with various techniques.

5. Conclusions

Artificial intelligence plays a major role in autonomous cleaning. Good detection is required for the floor cleaning robot to clean it perfectly. The floor cleaning robot fails due to the object collision. The traditional approach has few drawbacks. The study investigation made in the proposed model of Yolov5 is mainly used to determine the object and prevent a collision during cleaning with the help of 6-DOF. The model is well trained with the custom data to solve the unique challenge in the autonomous floor cleaning robot. It overcomes the drawbacks of the existing system. Some difficulties were initiated through this work and will be attempted in the future. The proposed item detects the accuracy of the autonomous moving of the vehicle. The model produced an accuracy of 93%.

Author Contributions:

Funding:

Institutional Review Board Statement:

Informed Consent Statement:

Data Availability Statement:

Conflicts of Interest:

References

1. Prabakaran, V.; Le, A.V.; Kyaw, P.T.; Kandasamy, P.; Paing, A.; Mohan, R.E. sTetro-D: A deep learning based autonomous descending-stair cleaning robot. *Eng. Appl. Artif. Intell.* **2023**, *120*, 105844. https://doi.org/10.1016/j.engappai.2023.105844.

2. Singh, I.S.; Wijegunawardana, I.D.; Samarakoon, S.B.P.; Muthugala, M.V.J.; Elara, M.R. Vision-based dirt distribution mapping using deep learning. *Sci. Rep*. **2023**, *13*, 12741. https://doi.org/10.1038/s41598-023-38538-3.
3. Tian, M.; Li, X.; Kong, S.; Wu, L.; Yu, J. A modified YOLOv4 detection method for a vision-based underwater garbage cleaning robot. *Front. Inform. Technol. Electron. Eng.* **2022**, *23*, 1217–1228. https://doi.org/10.1631/FITEE.2100473.
4. Ramalingam, B.; Le, A.V.; Lin, Z.; Weng, Z.; Mohan, R.E.; Pookkuttath, S. Optimal selective floor cleaning using deep learning algorithms and reconfigurable robot hTetro. *Sci. Rep.* **2022**, *12*, 15938.
5. Ramalingam, B.; Lakshmanan, A.K.; Ilyas, M.; Le, A.V.; Elara, M.R. Cascaded Machine-Learning Technique for Debris Classification in Floor-Cleaning Robot Application. *Appl. Sci*. **2018**, *8*, 2649.https://doi.org/10.3390/app8122649.
6. Canedo, D.; Fonseca, P.; Georgieva, P.; Neves, A.J.R. A Deep Learning-Based Dirt Detection Computer Vision System for Floor-Cleaning Robots with Improved Data Collection. *Technologies* **2021**, *9*, 94. https://doi.org/10.3390/technologies9040094.
7. Tian, M.; Li, X.; Kong, S.; Wu, L.; Yu, J.. A modified YOLOv4 detection method for a vision-based underwater garbage cleaning robot. *Front. Inf. Technol. Electron. Eng.* **2022**, *23*, 1217–1228. https://doi.org/10.1631/FITEE.2100473.
8. Carrasco, D.P.; Rashwan, H.A.; García, M.Á.; Puig, D. T-YOLO: Tiny Vehicle Detection Based on YOLO and Multi-Scale Convolutional Neural Networks. *IEEE Access* **2023**, *11*, 22430–22440. https://doi.org/10.1109/ACCESS.2021.3137638.
9. Kumar, S.; Kumar, R. Saad. Real-Time Detection of Road-Based Objects using SSD MobileNet-v2 FPNlite with a new Benchmark Dataset. In Proceedings of the 2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 17–18 March 2023; pp. 1–5. https://doi.org/10.1109/iCoMET57998.2023.10099364.
10. Murthy, C.B.; Hashmi, M.F.; Bokde, N.D.; Geem, Z.W. Investigations of Object Detection in Images/Videos Using Various Deep Learning Techniques and Embedded Platforms—A Comprehensive Review. *Appl. Sci.* **2020**, *10*, 3280. https://doi.org/10.3390/app10093280.
11. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.
12. Palwankar, T.; Kothari, K. Real time object detection using ssd and mobilenet. *Int. J. Res. Appl. Sci. Eng. Technol.* **2022**, *10*, 831–834.
13. Li, W.; Liu, K. Confidence-Aware Object Detection Based on MobileNetv2 for Autonomous Driving. *Sensors* **2021**, *21*, 2380. https://doi.org/10.3390/s21072380.
14. Womg, A.; Shafiee, M.J.; Li, F.; Chwyl, B. Tiny SSD: A Tiny Single-Shot Detection Deep Convolutional Neural Network for Real-Time Embedded Object Detection. In Proceedings of the 2018 15th Conference on Computer and Robot Vision (CRV), Toronto, ON, Canada, 8–10 May 2018; pp. 95–101. https://doi.org/10.1109/CRV.2018.00023.
15. Pathmakumar, T.; Muthugala, M.A.V.J.; Samarakoon, S.M.B.P.; Gómez, B.F.; Elara, M.R. A Novel Path Planning Strategy for a Cleaning Audit Robot Using Geometrical Features and Swarm Algorithms. *Sensors* **2022**, *22*, 5317. https://doi.org/10.3390/s22145317.