An Innovative Vision System for Floor-Cleaning Robots Based on YOLOv5

Daniel Canedo(✉), Pedro Fonseca, Petia Georgieva, and António J. R. Neves

IEETA/DETI, University of Aveiro, 3810-193 Aveiro, Portugal
{danielduartecanedo,pf,petia,an}ua.pt

Abstract. The implementation of a robust vision system in floor-cleaning robots enables them to optimize their navigation and analysing the surrounding floor, leading to a reduction on power, water and chemical products' consumption. In this paper, we propose a novel pipeline of a vision system to be integrated into floor-cleaning robots. This vision system was built upon the YOLOv5 framework, and its role is to detect dirty spots on the floor. The vision system is fed by two cameras: one on the front and the other on the back of the floor-cleaning robot. The goal of the front camera is to save energy and resources of the floor-cleaning robot, controlling its speed and how much water and detergent is spent according to the detected dirt. The goal of the back camera is to act as evaluation and aid the navigation node, since it helps the floor-cleaning robot to understand if the cleaning was effective and if it needs to go back later for a second sweep. A self-calibration algorithm was implemented on both cameras to stabilize image intensity and improve the robustness of the vision system. A YOLOv5 model was trained with carefully prepared training data. A new dataset was obtained in an automotive factory using the floor-cleaning robot. A hybrid training dataset was used, consisting on the Automation and Control Institute dataset (ACIN), the automotive factory dataset, and a synthetic dataset. Data augmentation was applied to increase the dataset and to balance the classes. Finally, our vision system attained a mean average precision (mAP) of 0.7 on the testing set.

Keywords: Computer vision · Object detection · Deep learning · Floor-cleaning robots

1 Introduction

Supporting floor-cleaning robots with a robust vision system is getting more popular thanks to the new functionalities it can provide. From camera based

Supported by the project "i-RoCS: Research and Development of an Intelligent Robotic Cleaning System" (Ref. POCI-01-0247-FEDER-039947), co-financed by COMPETE 2020 and Regional Operational Program Lisboa 2020, through Portugal 2020 and FEDER.

A. J. Pinho et al. (Eds.): IbPRIA 2022, LNCS 13256, pp. 378–389, 2022.
https://doi.org/10.1007/978-3-031-04881-4_30

mapping [10] to detecting dirty spots on the floor [5], a robust vision system allows for a more efficient cleaning and navigation. Previous published works that managed to implement a vision system on their floor-cleaning robots, seem to have done it for several purposes: economize cleaning resources [8], distinguishing between different types of dirtiness [13], and distinguishing between dirty spots and useful objects [4]. In this work, we attempt to tackle all of these challenges. The vision system that we propose is able to distinguish between dirty spots and useful objects by carefully selecting training images where objects are present. We tackle three types of dirtiness: solid dirt, liquid dirt and stains. And, finally, we integrated this vision system into an autonomous cleaning robot prototype in order to economize the cleaning resources by controlling water, detergent and mechanical parts of the cleaning system based on the dirty information.

This document is an extension of our previous work [7], and presents the application of the proposed vision system in a real-world robot prototype. In the previous work, we explored the strengths of implementing a Deep Learning solution based on the YOLOv5 framework [9] to detect dirty spots. In that work, we tried to tackle the most relevant challenges pointed by the literature in this application. Those problems revolve around lack of data, complex floor patterns, extreme light intensities, blurred images caused by the robot movement, and dirt/clean discrimination. We tackled these problems by generating a synthetic dataset with complex floors that contain objects other than dirty spots, adding simulated light sources and shadows to the resulting artificial images. The main conclusion that we retrieved from that work was that generating synthetic data using complex floors that contain objects to train a YOLOv5 model is a viable solution to not only detect dirty spots, but also to distinguish between useful objects from dirt, as long as there is enough dirt variety. We also found that stains contributed to a considerable amount of false positives during the testing step, since this type of dirtiness is often overlooked in the literature and, therefore, not labelled.

In this document, the stains on the ACIN dataset [1] are annotated, complementing the annotations proposed in our previous work [7]. A real-world dataset was captured in an automotive factory using the floor-cleaning robot, and part of it was annotated. A synthetic dataset is generated using the tool [2] provided by [4], that we had to improve to use in this application. A data augmentation pipeline is proposed to balance the number of samples per class. Then, a YOLOv5 model is trained using a hybrid dataset consisting of the ACIN dataset, the automotive factory dataset and the synthetic dataset. The new annotations and the hybrid dataset are public, and the links provided at the end of the document. A self-calibration algorithm is adapted from [12] to stabilize the image intensity from the cameras installed on the floor-cleaning robot. Finally, the main results and conclusions are discussed.

This document is structured as follows: Sect. 2 presents the related work; Sect. 3 presents the methodology; Sect. 4 presents the results and discussion; Sect. 5 presents the conclusion.

2 Related Work

Building a vision system to aid floor-cleaning robots has been tackled in several different ways in the literature. Some works try to approach the problem with pre-processing and unsupervised techniques, which main advantage lies in avoiding a learning step. This approach, while not needing previous knowledge to be used, have some problems, such as detecting everything that is not within the floor pattern as dirt. Wires, objects, walls, doors, shoes, carpets, just to name a few examples, have a high chance of being detected as a dirty spot by this approach. More problems revolve around blurred images, uneven illumination, and floors with multiple patterns. Other works try to approach the problem with object detection techniques, mainly based on Deep Learning. The literature seems to indicate that this approach is more successful, however it also has some problems, specially if one does not know where the floor-cleaning robot will operate. It is quite difficult to organize a robust and varied training data that covers most real-word scenarios that the floor-cleaning robot might encounter. If the area and dirty spots that the floor-cleaning robot needs to cover is known, this problem can be overcome by capturing a dataset for the training step in that area. However, if it is unknown, the training dataset must be diverse both in floor patterns and dirt variety to enable the vision system to handle unknown circumstances with as much accuracy as possible.

Grünauer et al. [8] proposed an unsupervised approach based on Gaussian Mixture Models (GMMs) to detect dirty spots. Firstly, they do several pre-processing steps such as converting the captured images to the CIELAB color space, which main advantage lies in separating colour information from illumination. Then, the gradient is calculated, and the images are divided into blocks. The mean and standard deviation are calculated for each block, and those values are used by the GMMs to learn the floor pattern. If something in a given image breaks this pattern, it is considered as dirt. This approach suffers from some problems described above. However, it is a viable solution that does not require a learning step.

Ramalingam et al. [13] proposed a multi-stage approach to detect solid and liquid dirt based on a Single-Shot MultiBox Detector (SSD), a MobileNet, and a Support Vector Machine. The MobileNet extracts features, the SSD detects the dirty spots, and the SVM classifies liquid dirt based on size to identify spots that are harder to clean. Their strategy was to collect data that the robot might face during its cleaning operations, manually label it and use it to train the robot's vision system. This strategy allowed the robot to attain an accuracy higher than 96% in detecting solid and liquid dirt. The same first author proposed a three-layer filtering framework which includes a periodic pattern detection filter, edge detection, and noise filtering to detect dirty spots on complex floor patterns in another work [14]. The periodic pattern detection filter is able to identify the floor pattern and dirty spots, since floors generally have a defined pattern. The edge detection step is performed on the background subtracted images to sharpen the edges that may get blurred in the previous step. Finally, they filter the residual noise through a median filter. This work then proceeds to show promising results on some challenging images.

Bormann et al. [4] tackled the lack of data in this particular application by developing a tool to artificially generate data [2]. This tool is able to blend dirty spots into clean floors in random locations, add simulated light sources and shadows, and apply geometric transformations to both floors and dirty spots. Consequently, this tool enables the generation of large datasets from a small amount of images. Then, the authors proposed using the YOLOv3 framework to detect dirty spots. The YOLO family is basically an object detection algorithm supported by a CNN. This algorithm divides the image into a grid and outputs a bounding box and its probability of belonging to a certain class for each block of the grid. This approach allowed the YOLO framework to attain state-of-the-art results measured through the mAP in several benchmarks, such as COCO. And for the application of detecting dirty spots on the floor, this proposal also managed to obtain state-of-the-art results, demonstrating better performance than GMMs [8].

3 Methodology

This section is divided into subsections addressing the several steps carried out in this work, from data preparation to the experiments.

3.1 Vision System

The vision system of the robot was built based on the Robot Operating System (ROS). ROS provides a set of libraries and tools to build robot applications. We have built two nodes for our vision system: a node to access the cameras and a node to detect dirty spots. The node to access the cameras starts them up, calibrates the colormetric parameters of both cameras, accesses the captured images, and publishes them to the dirt detection node. Figure 1 illustrates the importance of calibrating the cameras.

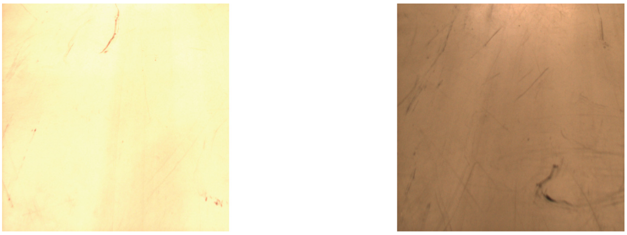

Fig. 1. An example of an overexposed image on the left and a calibrated image on the right.

This is a result of a self-calibration algorithm adapted from [12] which is based on the image luminance histogram. With this histogram, it is possible to indicate if the image is underexposed or overexposed by dividing the histogram of the grayscale image into five regions to calculate its mean sample value (MSV). The histogram is uniform if MSV ≈ 2.5. Based on this value, we implemented a Proportional-Integral controller (PI) to regulate the gain and exposure of both cameras. This results in uniform images, which makes dirty spots more visible.

Afterwards, the dirt detection node receives those calibrated images, detects the dirty spots, calculates the dirty area present in the images, and publishes that information. Based on the dirty area calculated from the images captured by the front camera, the floor-cleaning robot will regulate its speed, water, and detergent. If significant dirty spots are detected on images captured by the back camera, it is an indication that the cleaning was not successful. All the regions that were unsuccessfully cleaned are mapped such that the floor-cleaning robot goes back for a second sweep at the end of its cleaning procedure. Figure 2 shows our floor-cleaning robot prototype.

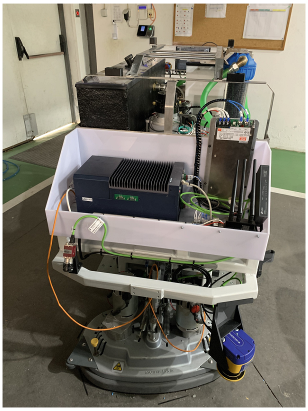

Fig. 2. Floor-cleaning robot prototype where the vision system was implemented and tested.

3.2 Automotive Factory Dataset

A real-world dataset was captured using the floor-cleaning robot prototype under development. This dataset was captured in a challenging environment: an automotive factory. It was noted a huge variety in dirty spots, particularly in size. Since the overall dirty spots on the ACIN dataset are minuscule, part of the automotive factory dataset was used to train the YOLOv5 model. It is expected that by doing so, the model is capable of accurately detecting dirty spots independently of their size. 39 captured images were annotated using the LabelImg tool [3], resulting in 92 instances of solid dirt, 3 instances of liquid dirt, and 804 instances of stains. Figure 3 illustrates some samples of this dataset.

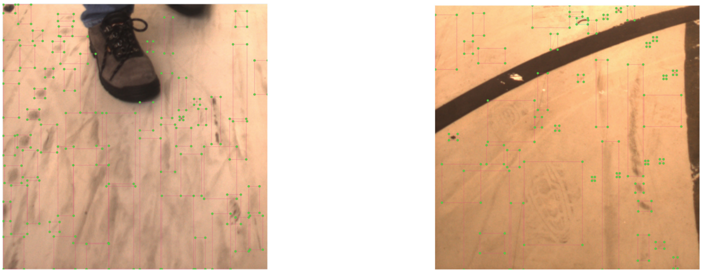

Fig. 3. Automotive factory dataset annotations using LabelImg.

3.3 Stain Annotations on the ACIN Dataset

As mentioned in Sect. 1, stains are also a type of dirtiness that is often overlooked during labeling. In this work, all stains on the ACIN dataset were annotated using the LabelImg tool. These new annotations complemented the ones proposed in our previous work [7], resulting in 1785 instances of solid dirt, 634 instances of liquid dirt, and 4162 instances of stains. The ACIN dataset consists of 968 images. Figure 4 illustrates some examples of stain annotations.

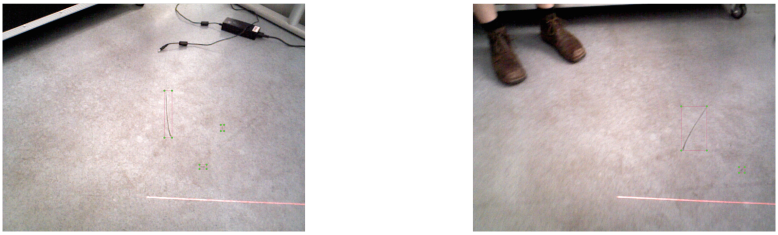

Fig. 4. Stain annotations using LabelImg.

3.4 Data Augmentation

Data augmentation is a well known technique used in Machine Learning to increase and enhance training data. This can be done by applying geometric transformations, color space augmentations, feature space augmentations, random erasing, and so on. This is particularly useful in tackling overfitting. Overfitting occurs whenever the network learns a function with very high variance to fit the training data, which makes it unreliable when facing new data. This phenomenon can happen when the network is too complex and/or the training data is too small.

Since the YOLOv5 network is complex, the ACIN dataset only has 968 images, and the automotive factory dataset only has 39 annotated images, it was expected to occur overfitting during the training step. Therefore, we performed data augmentation on both datasets. For this, we used a tool called

Albumentations [6]. This tool offers several image transformations. However, we had to make some modifications to the original code to fit our needs. Whenever we applied a perspective transformation to the image, sometimes dirty spots on the transformed image were out of bounds. The tool deals with these scenarios with a minimum visibility threshold, deleting any bounding box if the threshold is not met, however it was not working properly for our case. For this reason, changes were implemented in the tool such that no bounding box is deleted, and its position is returned even if it completely goes out of bounds. This change gave us the possibility to handle these situations better. Since some dirty spots are quite small, sometimes during a perspective transformation, dirty spots that were near the image borders hardly became visible. To avoid this partial visibility problem, we only generated images where dirty spots were fully visible or fully invisible.

We divided the ACIN dataset and the automotive factory dataset into training sets and validation sets. We only applied data augmentation on the training sets. From each ACIN training image, four were generated, and from each automotive factory training image, five were generated. This data augmentation was performed by applying the following transformations with a probability p:

- Flip either horizontally, vertically, or both horizontally and vertically. (p = 0.75).
- Randomly change hue, saturation and value, or randomly change gamma, or randomly change brightness and contrast (p = 1).
- Randomly shift values for each channel of the input RGB image (p = 0.5).
- Perform a random four point perspective transform (p = 0.75),

Figure 5 illustrates an augmentation example.

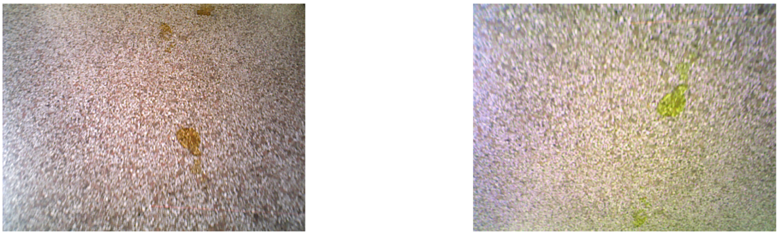

Fig. 5. Original image on the left, augmented image on the right.

3.5 Synthetic Dataset

In this work, we use an adapted version of the data generation tool proposed by [4]. As mentioned in Sect. 2, this tool is able to blend dirty spots into clean floors in random locations, add simulated light sources and shadows, and apply geometric transformations to both floors and dirty spots. We made some adaptations to this tool as mentioned in our previous work [7]. It is now able to generate liquid dirt by manipulating their transparency, it checks if there is a mask associated with the floor image such that dirty spots are only generated on the floor, and the output labels are converted to the YOLO format.

In order to generate these images, we needed floor images, solid dirt samples, and liquid dirt samples. Regarding floor images, 496 samples were obtained from Google Search. It was given priority to floor images that approximately simulated the distance from the floor (0.7 m) and angle (45 to 70 °C downwards) of the cameras placed on our floor-cleaning robot. Some images only represented a clean floor, while others had some objects in it such as shoes, wires, doors, carpets, walls, and so on. This helps the network to distinguish between a dirty spot and an object, reducing false positives. Regarding solid dirt, 141 samples were obtained from the Trashnet dataset [15] and Google Search. As for liquid dirt, 45 samples were obtained from Google Search. All of these samples were segmented using a tool proposed by Marcu et al. [11].

Before the data generation, we divided the floor images, solid dirt samples, and liquid dirt samples so the training set and validation set do not use the same images. It is possible to adjust the parameters of the synthetic data generation tool such as the number of dirty spots per image and the number of augmentations. These parameters were adjusted considering the amount of dirt instances on the ACIN dataset and the automotive factory dataset. This was done so that when we created the hybrid dataset by combining the three datasets, it should have a balanced number of samples per class. We have also increased the dirt size ceiling to compensate for the ACIN dataset small dirt size.

3.6 Training

For the experiment, we created a hybrid dataset. Specifically, the previously created training sets were combined, as well as the validation sets. Table 1 aims to provide a better insight on the training and validation sets.

Table 1. Training and validation sets.

Set	Images	Solid dirt	Liquid dirt	Stains
Training	9470	23670	23426	22367
Validation	388	506	584	599

There are several YOLOv5 models with different complexities. We chose the medium network (YOLOv5m6) since it was the best performing one for this type of data, as concluded in our previous work [7]. Transfer Learning was implemented by freezing the backbone, which helps to tackle overfitting. We used the Stochastic Gradient Descent (SGD) optimizer with a learning rate of 0.01 and a decay of 0.0005. The image size was set to 640 × 640, the batch size was set to 32, and the training was done over 50 epochs with early stopping, saving the best weights. This was performed with an Nvidia GeForce RTX 3080 GPU and an AMD Ryzen 5 5600X 6-Core 3.7GHz CPU.

3.7 Testing

We created a small testing set to test our network and the network from our previous proposal [7] for comparison. For this, we annotated a few images from the automotive factory dataset that are different from the ones used in the training, as well as a few images from the IPA dataset [1]. The following Table 2 aims to provide a better insight on the testing set.

Table 2. Testing set.

Set	Images	Solid dirt	Liquid dirt	Stains
Testing	24	80	2	212

The type of dirtiness on the automotive factory dataset mainly consists of stains and solid dirt, but only a few liquid dirt instances. That is why the classes on the testing set are not balanced. However, the testing step will provide an overall view of how the proposed vision system is capable of handling real-world settings. The testing was done for a binary classification (dirty or not dirty) since what is important in our application is the detection of dirty spots, and not so much the classification of those given spots.

4 Results and Discussion

Table 3 shows the mAP on the testing set of this work compared with our previous work. Figure 6 shows the respective precision-recall curves.

Table 3. Comparing the network of this work with our previous work.

	Previous YOLOv5m6 [7]	Current YOLOv5m6
mAP	0.29	**0.70**

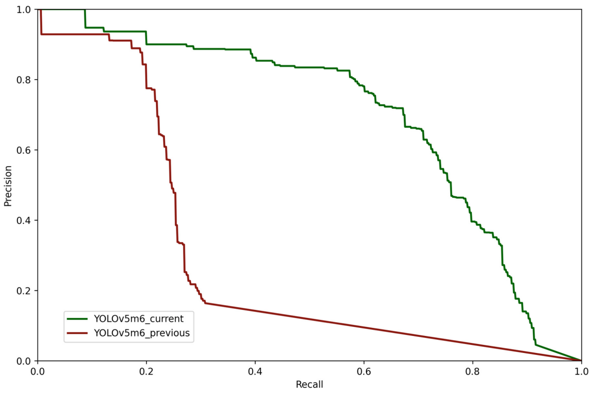

Fig. 6. Precision-recall curves on the testing set.

These results show how relevant stains are. This type of dirtiness is quite common because it is mostly caused by shoes. Although the previous network is able to detect some stains, it cannot outperform a network that was trained considering this type of dirtiness. Figure 7 illustrates the results of some images of the testing set for both networks, side by side.

It is possible to observe that the current network not only performs better in detecting stains, but also in detecting the other types of dirtiness. This was expected, since the previous network was only trained with a synthetic dataset with two classes: solid dirt and liquid dirt. The current network was trained on a hybrid dataset with one more class than the previous network: stains. Class balance and data augmentation was also considered in this work. Therefore, these results were expected and desired, since this is the network that will be used by our floor-cleaning robot.

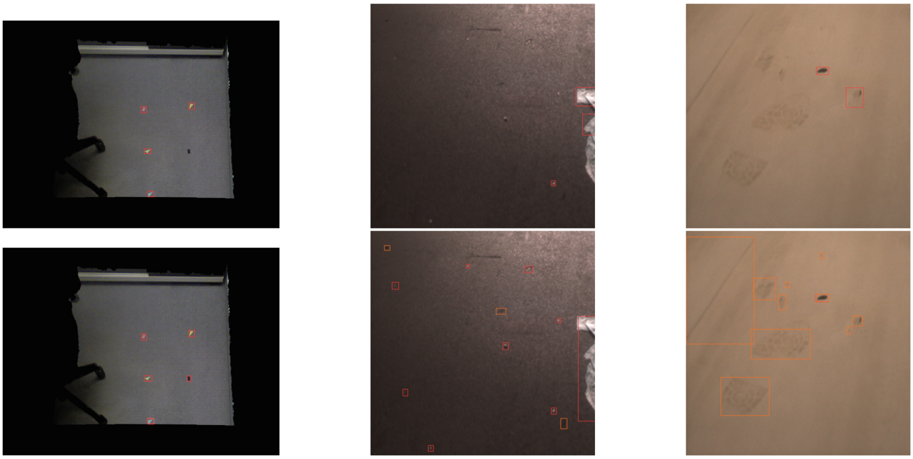

Fig. 7. Results on the testing set. Previous network on the top, current network on the bottom.

However, these results can still be improved by increasing the variety of floor images, solid dirt, liquid dirt, and stains of the training set. Dirty spots can come in different sizes, different colours, and complex shapes. Adding this to the limitless amount of floor patterns makes it sometimes difficult to detect dirty spots. During the testing step the network encountered some problems, mainly when the floor is worn out. Figure 8 illustrates an image where the network struggled to distinguish between holes and dirty spots.

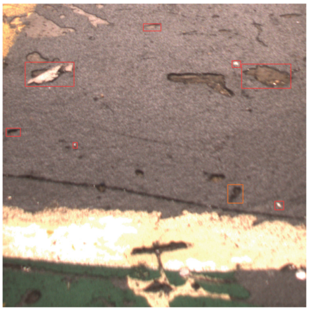

Fig. 8. Results on an image from the testing set.

As it is possible to observe, there are some holes in this image that are detected as dirty spots. We did not consider adding worn out floors to the training set, and therefore the network struggles to distinguish worn out features from actual dirty spots. However, this is one of the problems of trying to have a network that strives to generalize every single scenario. Generally, the application of a floor-cleaning robot is self-contained, meaning that the floor patterns and expected dirty spots are known in advance. In those cases, one can build a vision system that has a close to perfect efficiency.

5 Conclusion

In this work we proposed a vision system for a floor-cleaning robot. This is a novel approach that uses two cameras, one on the front and one on the back. The front camera is responsible for adjusting the speed, water, and detergent of the floor-cleaning robot. The back camera is responsible for mapping regions that were not successfully cleaned, such that the floor-cleaning robot goes back for a second sweep later on. The colormetric parameters of the cameras are autonomously calibrated to adapt the light conditions and floor type, a major contribution to spot dirt. A hybrid dataset was built using the ACIN dataset, the automotive factory dataset which was captured using our floor-cleaning robot, and a synthetic dataset. In this work, we paid special attention to a type of dirtiness that is often overlooked in the literature: stains. For this reason, we annotated all the stains on the ACIN dataset.

Finally, we trained a YOLOv5m6 network on the hybrid dataset. We then tested that network on a testing set which was built using a few images from the automotive factory dataset and the IPA dataset. We attained a mAP of 0.7, which was a considerable improvement over the result of our previous work: 0.29.

The ACIN annotations are available at https://tinyurl.com/3hjpxehw and the built Hybrid dataset is available at https://tinyurl.com/2p8ryr7s.

References

1. ACIN and IPA datasets. https://goo.gl/6UCBpR. Accessed 17 Jan 2022
2. IPA dirt detection. http://wiki.ros.org/ipa_dirt_detection. Accessed 12 Jan 2022
3. Tzutalin. labelimg. https://github.com/tzutalin/labelImg. Accessed 14 Jan 2022
4. Bormann, R., Wang, X., Xu, J., Schmidt, J.: DirtNet: visual dirt detection for autonomous cleaning robots. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 1977–1983. IEEE (2020)
5. Bormann, R., Weisshardt, F., Arbeiter, G., Fischer, J.: Autonomous dirt detection for cleaning in office environments. In: 2013 IEEE International Conference on Robotics and Automation, pp. 1260–1267. IEEE (2013)
6. Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A.: Albumentations: fast and flexible image augmentations. Information **11**(2), 125 (2020)
7. Canedo, D., Fonseca, P., Georgieva, P., Neves, A.J.: A deep learning-based dirt detection computer vision system for floor-cleaning robots with improved data collection. Technologies **9**(4), 94 (2021)
8. Grünauer, A., Halmetschlager-Funek, G., Prankl, J., Vincze, M.: The power of GMMs: unsupervised dirt spot detection for industrial floor cleaning robots. In: Gao, Y., Fallah, S., Jin, Y., Lekakou, C. (eds.) TAROS 2017. LNCS (LNAI), vol. 10454, pp. 436–449. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-64107-2_34
9. Jocher, G., et al.: ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations, April 2021. https://doi.org/10.5281/zenodo.4679653
10. Kang, M.C., Kim, K.S., Noh, D.K., Han, J.W., Ko, S.J.: A robust obstacle detection method for robotic vacuum cleaners. IEEE Trans. Consum. Electron. **60**(4), 587–595 (2014)
11. Marcu, A., Licaret, V., Costea, D., Leordeanu, M.: Semantics through time: semi-supervised segmentation of aerial videos with iterative label propagation. In: Proceedings of the Asian Conference on Computer Vision (2020)
12. Neves, A.J.R., Trifan, A., Cunha, B.: Self-calibration of colormetric parameters in vision systems for autonomous soccer robots. In: Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.) RoboCup 2013. LNCS (LNAI), vol. 8371, pp. 183–194. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44468-9_17
13. 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. **8**(12), 2649 (2018)
14. Ramalingam, B., Veerajagadheswar, P., Ilyas, M., Elara, M.R., Manimuthu, A.: Vision-based dirt detection and adaptive tiling scheme for selective area coverage. J. Sens. **2018** (2018)
15. Yang, M., Thung, G.: Classification of trash for recyclability status. CS229 Project Report 2016 (2016)