Application of the Image Processing Technique for Powerline Robot

Ha Quang Thinh Ngo[1,2](✉) and The Tri Bui[1,2]

[1] Department of Mechatronics, Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam
nhqthinh@hcmut.edu.vn

[2] Vietnam National University Ho Chi Minh City (VNU-HCM), Linh Trung Ward, Thu Duc City, Ho Chi Minh City, Vietnam

Abstract. Applying image processing to electromechanical systems is a problem of interest to scientists, in order to serve humans in many fields. To do that, there needs to be a connection between image processing and mechanical construction to create complete mobile cameras. One of the research directions is about mobile cameras, specifically a system consisting of dual cameras that detect and track moving objects, and at the same time calculate the distance from the dual camera system to the target, this system can be application in object tracking robot. In this paper, the research object includes the camera system designed according to the pan-tilt structure, the algorithm used for object detection is YOLO-based on CNN, estimating the distance from the camera system to the object. By means of stereo vision, control the pan-tilt system to automatically track objects.

Keywords: Image processing · Powerline robot · Machine learning

1 Introduction

In the context of the world economy rapidly expanding to promote global competition in all industrial sectors. It leads to optimization of production operations, efficient process management or the integration of multiple functions into a single unit [1–5]. There are various ways to enhance the competition in the market where efficiency in material handling tasks has been given considerable attention. As it increases industrial productivity and reduces labor costs associated with logistics and distribution jobs, most businesses focus on improving various raw material supply technologies. They often use automated guided vehicles (AGVs) for industrial warehouses to overcome the mentioned problems. In [6–8], the authors developed a PID-based algorithm for a wheelless AGV to follow a reference trajectory to prevent oscillations during travel. Some camera apps help the robot track [9–12] - instead of using an infrared sensor - to determine if it should keep moving forward or turn left and right. Compared with simplified models

T. T. Bui—Graduate student.

C. V. Phan and T. D. Nguyen (Eds.): ICCASA 2022, LNICST 475, pp. 178–189, 2023.
https://doi.org/10.1007/978-3-031-28816-6_14

and other classical control techniques [13–15], the results show stability at lower speeds. The researchers [16–20] have designed an AGV control system based on the fuzzy PID diagram that helps the AGV trolley to have robustness and stable operation for a long time. Main controller chip like STM32F207 is a powerful microprocessor with high speed, large memory and low cost [21, 22].

The content of this paper proposes an image processing method that is widely applied in all fields today. The technique used is mainly an artificial intelligence algorithm consisting of many layers of neurons. The training sample set is selected in accordance with the actual conditions. After the training process, the algorithm is able to recognize the object within the effective range of the camera. The second part is followed by the presentation of the mechanical design for the camera frame structure. The distance calculation method is described in Sect. 3. Some explanations about neural networks in Sect. 4 are applied in the algorithm. Next, the content of part 5 revolves around object recognition based on some characteristics of the object. Experimental results are presented in Sect. 6. Finally, some conclusions are made after applying the algorithm in practice.

2 Computational Method for Distance

Usually, the method used to extract 3D information is to use multiple images, also known as multiple images method, of which a simple method is stereo vision. This method uses two cameras to reconstruct a 3D scene. To determine distance information, the features of the object in one image (or more) are first matched in another image (simultaneous images of the object from separate cameras). Then, the difference in the features of the object in the two images will be used to calculate the distance.

The stereo vision method requires the two optical axes of the two cameras to be parallel. This method is geometrically illustrated as shown below:

- p and p′ is the intersection of two rays CP and CP′ with the image plane I′I. C,C′ is the center of the lens, P is the object
- f is the focal length of a lens
- b is called the baseline, which is the distance between the 2 lens centers. With the same Z distance, increasing baseline will lead to increased accuracy when determining Z distance because of limited camera resolution
- Disparity D is the horizontal displacement of the same object on 2 images taken from 2 cameras (Fig. 1).

Using similar triangles obtained,

$$\frac{b}{Z} = \frac{b-(d+d')}{Z-f} = \frac{d+d'}{f} \tag{1}$$

We have,

$$Z = \frac{f \cdot b}{D} \tag{2}$$

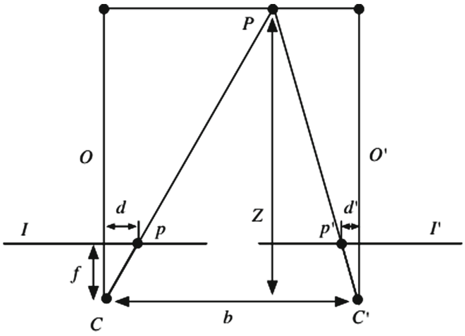

Fig. 1. Geometric description of stereo vision

The parameters D, f, b are calculated through the process of calibrating the camera system, rectification and un-distortion.

In reality, the camera can never be set up perfectly to achieve the frontal parallel as shown in the figure. Instead, we often have to compute, find projections, and correct distortions (rectify). Left and right images so that they align (row-aligned) (Fig. 2).

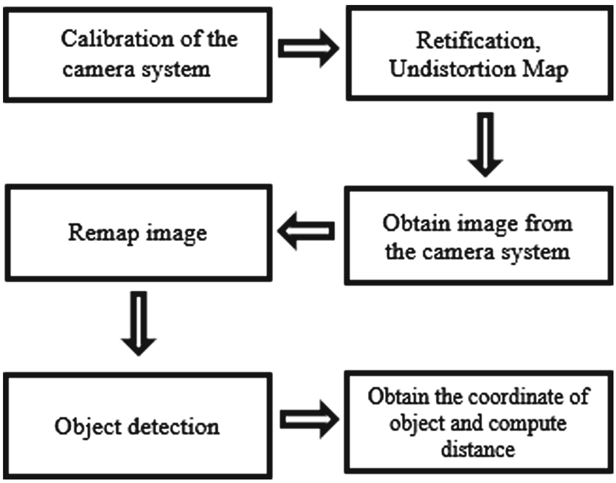

Fig. 2. Block diagram of the image processing method

Stereo calibrate is used to calculate the internal and external parameters of the camera system. These parameters can be measured manually or using the cv::stereoCalibrate function of the OpenCV library. This function returns the internal and external parameter matrices of the camera in which the focal length f and the baseline b are used for the distance calculation step, in addition, the function also returns the matrices used for the rectification, un-distortion process.

Rectification is the step of projecting 2 image planes onto a plane parallel to the line joining the 2 lens centers, each pixel or object in one image can be found on the same row in another image. This process also makes the 2 optical axes of the 2 cameras parallel. Un-distortion is the step that removes radial and tangential distortion.

The functions in OpenCV used for the 2 steps Rectification and Un-Distortion will return the matrices used for the remap process. The remap process uses these matrices on the original pixel matrix to create a rectify and undistorted pixel matrix. The object detection step uses the object detection algorithm to determine the pixel coordinates of the object's position on the 2 images, these coordinates are used to calculate the distance D. Later, the value of f, D and b which are gained, would be substituted into $Z = \frac{f \cdot b}{D}$ to calculate the distance.

3 Convolutional Neural Network - CNN

The CNN network processes the image through a number of layers. Here's an overview of the layers and their purposes:

- Convolutional layer – Used to detect features.
- Non-linearity layer – Use non-linearity in the system.
- Pooling (Down sampling) layer – Reduce the amount of weights and control overfitting.
- Flattening layer – Prepares the data for the Fully-Connected layer.
- Fully-Connected layer – Used for classification.

Basically, in the end, CNN is a neural network used to solve classification problems, but it uses other layers to prepare data and detect certain features beforehand.

1	0	1
0	1	0
1	0	1

Fig. 3. Filter with size 3 × 3

Convolutional layer is the main layer of the CNN network, responsible for detecting features such as straight edges, curves, and simple colors. This is done by using a filter on the image to extract some low- and high-level features on the image. The filter is usually a multidimensional array containing the pixel values. Example: Consider a 5 × 5 image channel where each pixel has a value of 1 or 0. And use the following simple 3 × 3 filter as Fig. 3.

After each convolutional layer there is usually a nonlinear layer. This class uses one of the activation functions. The commonly used function is the rectifier function, so this layer is also called the ReLU (Rectify Linear Units) layer as Fig. 4. This rectifier function f(x) = max(0,x) will return the values in the image less than 0 to 0. The figure below illustrates the use of this function.

Fig. 4. Application of ReLU technique on image

Pooling layer (composite layer) is the layer inserted between consecutive convolutional layers in the CNN network. The function of this layer is to reduce the spatial size of the array, the number of parameters and computation in the network. There are different types of pooling like L2 pooling, mean pooling, max pooling. Use the filter again. The image below uses a 2 × 2 max pooling filter onto a 4 × 4 image. This filter selects the largest number in the part of the image that it covers. In this way, a smaller image is obtained but still contains enough information for the neural network to make an accurate decision. However, many models replace the pooling layer with additional convolutional layers with a larger stride. Also, newer generation models, such as VAEs or GANs, eliminate the Pooling layer altogether.

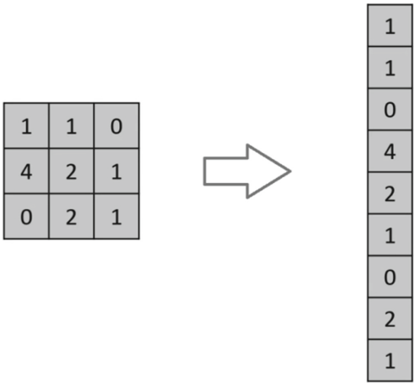

Fig. 5. Example of matrix conversion

The flattening layer is the layer used to prepare the input data of the fully-connected layer. Since neural networks receive data in one dimension as an array of values, this layer uses the data passed from the pooling layer or convolutional layer and compresses the matrices into 1-dimensional arrays as Fig. 5. Below is a visual image of the matrix pressing process.

The fully-connected layer as Fig. 6 is the last layer and the layer that actually performs the classification. This layer basically takes an input, whether it is the output of a convolutional, ReLu, or pool layer, and outputs an N-dimensional vector, where N is the number of classes that the program has to classify. For example, if you want a program that classifies digits from 0 to 9, N would be 10. Each number in this N-dimensional vector represents the probability of a given class. For example, if the resulting vector of a numerical classifier program is [0 .1 .1 .75 0 0 0 0 .05] then 10% chance of image is

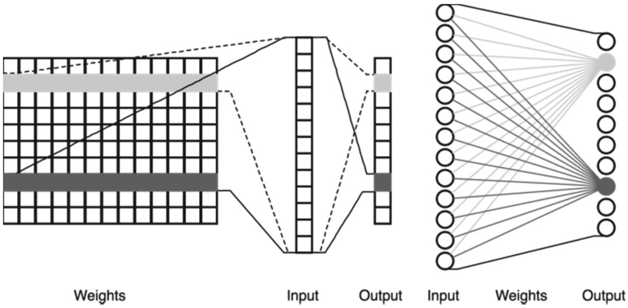

Fig. 6. Example of fully-connected layer

1.10% chance of image is 2.75% the image likelihood is 3 and the image probability is 9. The way the fully-connected layer works is that it looks at the output of the previous layer (this output is the feature map of the high-level features) and determine which feature best corresponds to a particular class. The fully-connected layer can be imagined as a matrix of weights multiplied by the input feature map, resulting in the probabilities of the different classes.

4 An Algorithm for Recognition

The input of the YOLO network is pre-labeled images, the output corresponding to each image is a feature map in the form of a grid of size N × N cells. Corresponding to each cell, the network predicts class probabilities. The input of the YOLO network is pre-labeled images, the output corresponding to each image is a feature map in the form of a grid of size N × N cells. Corresponding to each cell, the network predicts the class probabilities, bounding boxes and confidence scores of each bounding box.

Each cell has B * 5 + C elements. In there:

B is the number of bounding boxes of each cell.
C is the number of class probabilities.
5 is the number of elements of each bounding box (including x,y: coordinates of the center point of the bounding box corresponding to the cell in which the point lies, w - the width of the bounding box corresponding to the original image, h -height of the bounding box relative to the original image, confidence score: the probability that the object is present in the bounding box.

The confidence score is calculated as follows,

$$\Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} \tag{3}$$

The IOU is used to evaluate the detection, the IOU is calculated by dividing the intersection area by the union of the predicted bounding box and the true box (Fig. 7).

The center coordinates, width, and height of the bounding box are converted to segment [0, 1]. The figure below illustrates how these coordinates are calculated (Fig. 8):

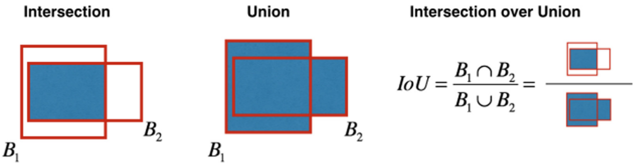

Fig. 7. Example of computational method for IOU

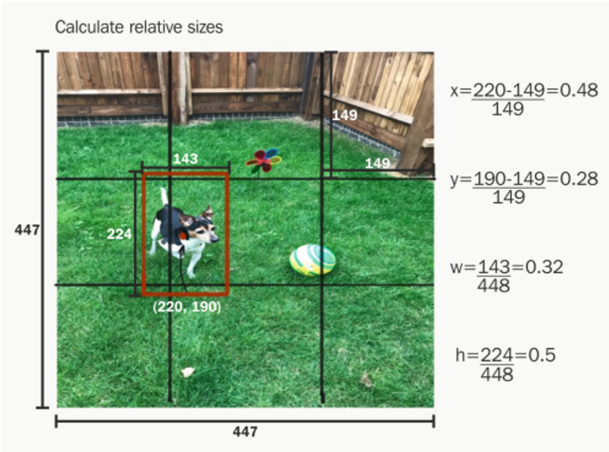

Fig. 8. Example of computational method for coordinates in box

C are class probabilities, Pr(Classi|Object). These probabilities are considered only if the cell contains an object. The network predicts only a unique set of class probabilities for each cell, regardless of the number of bounding boxes B.

At the time of using the images used for the network detection test (time-mem test). Multiply confidence score and class probability,

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} = \Pr(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}} \tag{4}$$

Pr(Classi) includes the probability of a class appearing in the bounding box and the confidence score.

5 Results of Research

In the practical application, the proposed scheme is embedded into our powerline robot. From the needs of the power corporation of Vietnam, the power transmission lines are required to be cleaned after a period of time. Otherwise, some unexpected problems could be occurred such decreasing the quality of power line due to the chemical corrosion, short-circuit or discharge phenomenon when the weather condition is bad, and bird nesting avoidance. Since the natural factors would impact on powerline, worker must do the cleaning job once every two or three months. In previous time, those jobs were entirely handled by manual which requires various skills and faces the potential danger. In addition, the power must be cut on the line when the cleaning job is processing. This will cause great economic losses, inconvenience to people's daily lives, and disrupt communications.

The automated solution is introduced as powerline robot that controlled by host computer. Instead of human, robot would reach to power line alone and manipulate the cleaning task. This method has many advantages such as not violating the electrical safety, improving the precision and ensuring the quality of cleaning service. In many

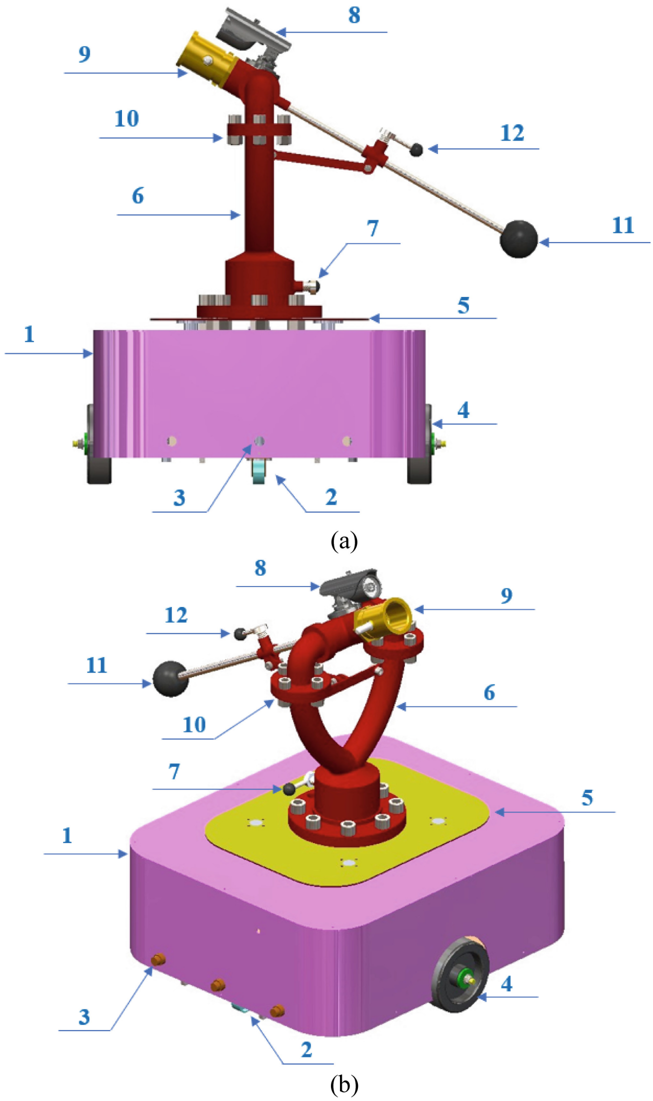

Fig. 9. Computer-based design of the proposed system, (a) front view and (b) 3D view

countries, the powerline robot becomes an indispensable partner when the air condition is unfavorable. As a result, in this paper, the autonomous platform of powerline robot based on the image processing technique is investigated as Fig. 9. List of components in this system is depicted as Table 1. The function of each component and its specification are also explained.

Table 1. List of components for powerline robot.

No.	Specification	Description
1	Body of mobile platform	Its shape might be rectangle or square made by metal material
2	Castor wheel	It moves freely and must be driven by the other wheels
3	Proximity sensor	In the closed distance, this sensing device is to detect any obstacle
4	Driving wheel	It is directly connected with DC motor which manipulate the driving mission by the differences in velocities of left wheel and right wheel
5	Lifting part	The function of this part is to elevate vertically according to the desired height. It is energized by an electric cylinder
6	Body of water gun	Its material is hard enough to suffer the high pressure of water
7	Basement of water gun	It is possibly rotated around z axis in order to provide the wide angle for gun
8	Digital camera	It is used to implement the image processing technique to recognize and measure distance
9	Muzzle of water gun	It could be adjusted by the pressure of water
10	Intermediate coupling	The connection between lower part and upper part
11	Knob of driving hand	In the case of manual control, it is useful for an operator to drive. Additionally, it plays a role as counterweight to balance
12	Navigation lock	Its usage is to fix an angle under the unexpected effect of highly water pressure

With those developments, the application of our approach is clearly stated. Later, the vision-based techniques are implemented. The training for optional object detection will use the YOLOv3-tiny pre-designed network, which has about half the mAP of the YOLOv3 version but has a higher FPS, suitable for training with laptops. no GPU. To increase the mAP to the maximum possible extent of the network, we will use a large and diverse dataset. Detection requires good image information so the input image resolution should be large, be it 416 $\times$ 416 or 608 $\times$ 608. Lowering the resolution will increase the FPS. Training will use Darknet - an open-source neural network framework written in C and CUDA that supports both CPU and GPU computation. Training with GPU will have faster training speed than CPU. Because the thesis uses a computer without a GPU, this training is done using Google Colab. This is a free GPU from Google and we can train YOLO on it. During training, after each iteration, the training program will show the average loss. If this number reaches 0.xxxxx and does not decrease further after many iterations then we will stop the training. After finishing training, we get the weight file.weights which will be used for object detection code.

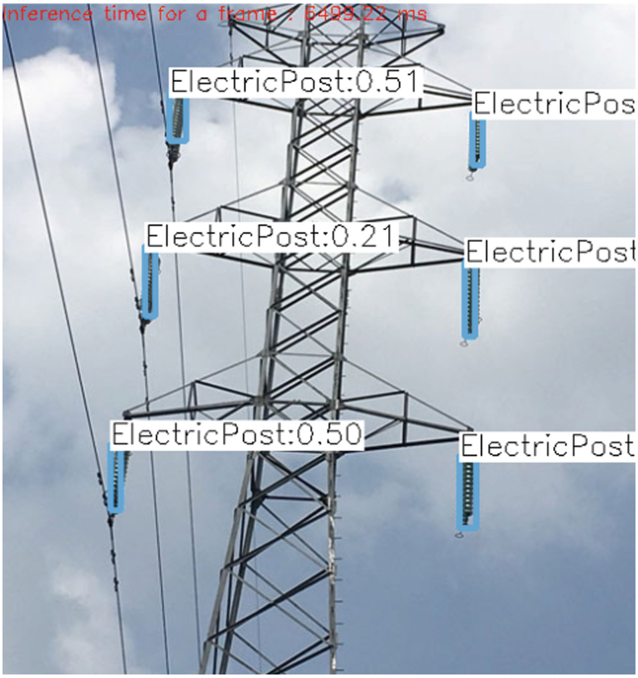

Fig. 10. Experimental result of the proposed method in powerline.

The practical result by using vision approach is shown as Fig. 10. The training process has been trialed in many times so that the weights were as high as possible. There were approximately one thousand of pictures which captured in the outdoor environment. Then, all of them have been labeled and marked in order to feed the machine learning scheme. Along with these results, it also exists some limitations that could affect on the quality of output. One of them is the weather condition or the time of day. Additionally, both terrain and location of powerline are the important factors for the high quality of recognition.

6 Conclusions

In this paper, a novel approach by using vision technique was investigated in the powerline robot. This application is actually essential in the field of electricity transmission due to its advantages such as safe maintenance, high productivity and uniform quality. Initially, a method to estimate distance from robot to powerline was mentioned. It is one of the basis computation to maintain the suitable pressure of water. Later, some definitions of convolutional network were described to provide the knowledge and information. Then, a model of YOLO scheme was recommended to train according to the dataset. The results of both simulation and experiment were clearly shown. It could be seen that our approach is effective, feasible and applicable in this field.

Acknowledgements. We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.

References

1. Chen, R.C.: Automatic license plate recognition via sliding-window darknet-YOLO deep learning. Image Vis. Comput. **87**, 47–56 (2019)
2. Tian, Y., Yang, G., Wang, Z., Wang, H., Li, E., Liang, Z.: Apple detection during different growth stages in orchards using the improved YOLO-V3 model. Comput. Electron. Agric. **157**, 417–426 (2019)
3. Tran, H.A.M., Ngo, H.Q.T., Nguyen, T.P., Nguyen, H.: Implementation of vision-based autonomous mobile platform to control by A∗ algorithm. In: 2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications & Computing (SigTelCom), pp. 39–44. IEEE (2018)
4. Fang, W., Wang, L., Ren, P.: Tinier-YOLO: a real-time object detection method for constrained environments. IEEE Access **8**, 1935–1944 (2019)
5. Lan, W., Dang, J., Wang, Y., Wang, S.: Pedestrian detection based on YOLO network model. In: 2018 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 1547–1551. IEEE (2018)
6. Bhuiyan, M.R., Khushbu, S.A., Islam, M.S.: A deep learning based assistive system to classify COVID-19 face mask for human safety with YOLOv3. In: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1–5. IEEE (2020)
7. Laroca, R., et al.: A robust real-time automatic license plate recognition based on the YOLO detector. In: 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–10. IEEE (2018)
8. Adarsh, P., Rathi, P., Kumar, M.: YOLO v3-Tiny: object detection and recognition using one stage improved model. In: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 687–694. IEEE (2020)
9. Hurtik, P., Molek, V., Hula, J., Vajgl, M., Vlasanek, P., Nejezchleba, T.: Poly-YOLO: higher speed, more precise detection and instance segmentation for YOLOv3. Neural Comput. Appl. **34**(10), 8275–8290 (2022)
10. Zhang, H., Watanabe, K., Motegi, K., Shiraishi, Y.: ROS based framework for autonomous driving of AGVs. In: Proceedings of the IPS6-04, ICMEMIS, Kiryu, Japan, pp. 4–6 (2019)
11. Linder, T., Pfeiffer, K.Y., Vaskevicius, N., Schirmer, R., Arras, K.O.: Accurate detection and 3D localization of humans using a novel YOLO-based RGB-D fusion approach and synthetic training data. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 1000–1006. IEEE (2020)
12. Mi, C., Huang, Y., Fu, C., Zhang, Z., Postolache, O.: Vision-based measurement: actualities and developing trends in automated container terminals. IEEE Instrum. Meas. Mag. **24**(4), 65–76 (2021)
13. Cao, Z., Liao, T., Song, W., Chen, Z., Li, C.: Detecting the shuttlecock for a badminton robot: a YOLO based approach. Expert Syst. Appl. **164**, 113833 (2021)

14. Dos Reis, D.H., Welfer, D., De Souza Leite Cuadros, M.A., Gamarra, D.F.T.: Mobile robot navigation using an object recognition software with RGBD images and the YOLO algorithm. Appl. Artif. Intell. **33**(14), 1290–1305 (2019)
15. Yu, Y., Zhang, K., Liu, H., Yang, L., Zhang, D.: Real-time visual localization of the picking points for a ridge-planting strawberry harvesting robot. IEEE Access **8**, 116556–116568 (2020)
16. Kim, H., Kim, H., Lee, S.I., Lee, H.: Autonomous exploration in a cluttered environment for a mobile robot with 2D-map segmentation and object detection. IEEE Robot Autom. Lett. **7**, 6343–6350 (2022)
17. Kulik, S.D., Shtanko, A.N.: Experiments with neural net object detection system YOLO on small training datasets for intelligent robotics. In: Misyurin, S.Y., Arakelian, V., Avetisyan, A.I. (eds.) Advanced Technologies in Robotics and Intelligent Systems. MMS, vol. 80, pp. 157–162. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-33491-8_19
18. Vemula, S., Frye, M.: Real-time powerline detection system for an unmanned aircraft system. In: 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 4493–4497. IEEE (2020)
19. Wang, D., Li, W., Liu, X., Li, N., Zhang, C.: UAV environmental perception and autonomous obstacle avoidance: a deep learning and depth camera combined solution. Comput. Electron. Agric. **175**, 105523 (2020)
20. Ohta, H., Sato, Y., Mori, T., Takaya, K., Kroumov, V.: Image acquisition of power line transmission towers using UAV and deep learning technique for insulators localization and recognition. In: 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC), pp. 125–130. IEEE (2019)
21. Lopez Lopez, R., Batista Sanchez, M.J., Perez Jimenez, M., Arrue, B.C., Ollero, A.: Autonomous UAV system for cleaning insulators in power line inspection and maintenance. Sensors **21**(24), 8488 (2021)
22. Sadykova, D., Pernebayeva, D., Bagheri, M., James, A.: IN-YOLO: real-time detection of outdoor high voltage insulators using UAV imaging. IEEE Trans. Power Delivery **35**(3), 1599–1601 (2019)