Detection and localization strategy based on YOLO for robot sorting under complex lighting conditions

Article *in* International Journal of Intelligent Robotics and Applications · May 2023

DOI: 10.1007/s41315-023-00285-z

7 authors, including:

Silu Chen
Chinese Academy of Sciences
132 PUBLICATIONS **1,389** CITATIONS
SEE PROFILE

Zaojun Fang
Chinese Academy of Sciences
83 PUBLICATIONS **775** CITATIONS
SEE PROFILE

Chi Zhang
Johns Hopkins University
105 PUBLICATIONS **2,354** CITATIONS
SEE PROFILE

Guilin Yang
Ningbo Institute of Industrial Technology, CAS
140 PUBLICATIONS **1,065** CITATIONS
SEE PROFILE

International Journal of Intelligent Robotics and Applications
https://doi.org/10.1007/s41315-023-00285-z

REGULAR PAPER

Detection and localization strategy based on YOLO for robot sorting under complex lighting conditions

Wujie Ge[1,2] · Silu Chen[1,2] · Hua Hu[2] · Tianjiang Zheng[2] · Zaojun Fang[2] · Chi Zhang[2] · Guilin Yang[2]

Received: 27 January 2023 / Accepted: 5 May 2023

Abstract

Many studies on the object detection emphasizes the accuracy of the algorithms themselves, while the requirement of real-time processing can be addressed by the usage of "you only look once" (YOLO) model. However, the reliably of machine vision is still a problem since some practical issues are not addressed properly, such as variation of light intensity, reflection of light on the surface and interference of shooting background. In this paper, we address above problems by developing a vision system with YOLO algorithm for object detection, segmentation and localization. A segmentation approach is adopted on the model outputs to extract the object to be detected from the background, under the premise of enhancing the adaptability of the YOLO model to environmental changes. Thus, the influence of background and light-sensitive factors on localization is removed even in extreme lighting conditions. An experimental platform is built based on a pair of low-cost cameras, which verifies the effectiveness of proposed method.

Keywords Object detection · Object localization · Object classification · Illumination · You look only once · Deep learning · Binocular vision

1 Introduction

The machine vision has been widely applied on automated sorting systems (Batchelor and Waltz 2001). The current automated sorting by robots has stringent requirements on the hardware, such as the stable and harsh lighting source, high accuracy of vision devices. These raise the cost of system integration (Zou et al. 2021). The hardware cost can be reduced if the sorting system is designed from an algorithmic perspective, where the detection and localization of objects in real time are critical steps.

In recent years, the machine learning methods have been applied to the detection and classification of objects, such as oriented fast and rotated brief (ORB) based on shape matching (Rublee et al. 2011), scale-invariant feature transform (SIFT) (Ng and Henikoff 2003) and histogram of oriented gradients (HOG) features (Wang et al. 2009). However, ORB requires accurate extraction of edge based on features, and it requires consistent and harsh lighting requirements. HOG and SIFT generally utilize support vector machine (SVM) for classification after manually extracting features. However, these methods have a low processing speed of 4 frames per second (FPS) when detecting the objects (Machaca Arceda and Laura Riveros 2018). These machine learning algorithms have limitations in the presence of occlusion and changes of illumination (Yin et al. 2022; Liu et al. 2020). Therefore, they cannot meet the requirement of real-time and stable sorting operation.

Notably, with the rapid development of deep learning, convolutional neural network (CNN) has been widely applied to object detection. Such methods are mainly divided into two categories: two-stage and single-stage. The two-stage algorithm utilizes faster region-based convolutional neural network (Fast R-CNN) for the detection and classification of objects (Girshick 2015), and extracts the region of interest (ROI) through the region proposal network (RPN), then classifies ROI by CNN. Toward a more efficient automated sorting, the speed of the detection becomes an increasingly important attribute. Although above methods usually have high detection accuracy, but they take long

✉ Silu Chen
chensilu@nimte.ac.cn

[1] College of Integrated Science and Education, Ningbo University, Ningbo 315211, Zhejiang, China

[2] Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo 315201, Zhejiang, China

Published online: 24 May 2023

processing time, which do not meet the real-time sorting application. The single-stage algorithm includes single shot multibox detector (SSD) (Liu et al. 2016), RetinaNet (Wang et al. 2019), "you look only once" (YOLO) (Redmon and Farhadi 2017, 2018; Alexey Bochkovskiy and Chien-Yao Wang 2020) series. They treat localization and classification as a regression problem to achieve end-to-end detection. These methods make detection speed fast enough for real-time sorting. However, their detection accuracy is degraded in the face of dramatic changes in the light intensity (Mirhaji et al. 2021).

With the aid of machine vision, the object's positional information can be acquired by binocular machine vision. The binocular machine vision has the capability to acquire the depth information (Sun et al. 2019). It matches the pixel coordinates of the object center in two imaging objects by polar alignment, and calculates the object poses relative to the camera's coordinate system by the similar triangle principle. Some existing algorithm detect the outlines of a limited number of specific objects (Modi and Desai 2011; Prasetyo et al. 2020), which cannot be applicable to the sorting line with a large number of different objects. Meanwhile, in sorting applications, the improper lighting will cause reflective areas on the background of image, which misleads the algorithm to identify the reflective position as the object. To solve the above issue, the industry will deploy hardware with less reflection (Tsang and Tsang 1997), which increases the cost of the system.

In this paper, a vision system based on the YOLOv5 network model for detection and localization is established to strengthen the insensitivity of the network model. Thereby, the unreliable localization of the object due to light reflection is solved from an algorithmic approach. The main contributions are as follows:

1. In order to make the detection model applicable under both low and high illumination, we add a module for image enhancement at the input of the YOLOv5 model, which comply with the need for real-time detection of sorting. Specially, the predicted frames at the output of this model are identified as an ROI for class detection and rough localization. Compared with the existing method that is only applicable to low illumination (Yin et al. 2022; Liu et al. 2020), our detection model can still maintain a good accuracy rate even when the background contains reflection area due to high illumination.
2. To reduce the portion of background in the bounding box, the anchor parameters of the model are updated by a clustering algorithm, so that the ROI fits the object well. Then, the segmentation of ROI is applicable to a variety of objects being expendable in the YOLOv5 model, rather than being limited to predefine objects as in Modi and Desai (2011); Prasetyo et al. (2020). Thereafter, the measurement algorithm by binocular machine vision is applied in the ROI to achieve the accurate calculation of object position.

The experiment is performed on the testbed to validate the effectiveness of the proposed algorithm.

2 Methodology

In this paper, we build an experimental platform based on a pair of HIKVSION industrial cameras. The depth information is useful for online programming of the grasping robot, which has four degree-of-freedom. This enables it to pick and place objects on the conveyor belt with arbitrary thickness and orientation. This is shown in the experimental testbed as in Fig. 1. For the reliable detection of the location and type of an object on a sorting line, there are four important steps, which are the judging the appearance of objects, optimizing detection model, recognizing under extreme light conditions, and positioning by the binocular camera.

2.1 The judgment for the appearance of an object

To save the computational resource, the dynamic localization with the binocular camera is only performed when an object passes by. Conventionally, hardware triggers such as photoelectric sensors have been used, but this increases the cost and complexity. In this paper, we will make the judgment by software algorithm.

In the experimental platform, the cameras are stationary and the objects move with a conveyor belt. We record the image without object as background. Obviously, the image

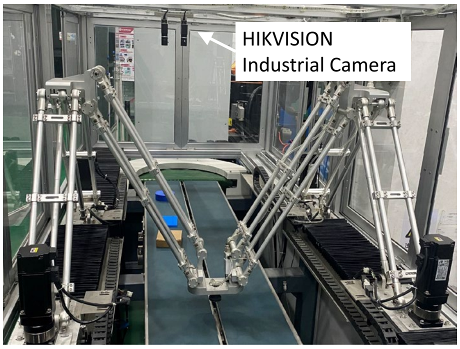


Fig. 1 The experimental testbed

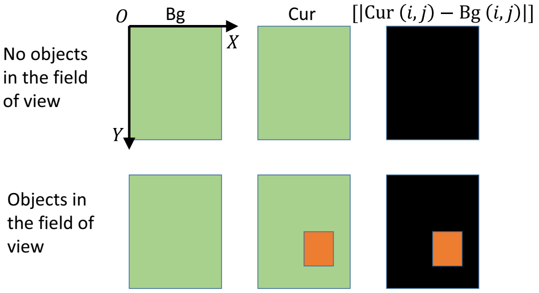

O
Bg
X
Y
Cur
[|Cur (i, j) − Bg (i, j)|]
No objects in the field of view
Objects in the field of view

Fig. 2 Detection of foreground and background

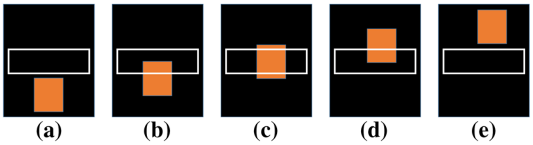


Fig. 3 Schematic diagram of object movement

with object passing the view of the cameras is difference from the background. Therefore, we can achieve the segmentation of foreground and background by comparing the pixels of these two images and doing a simple subtraction.

As shown in Fig. 2, before starting detection, the background image is saved, which is denoted as Bg. The image acquired at the beginning of detection is noted as Cur, and a new image is generated as $\left[|\mathrm{Cur}(i,j) - \mathrm{Bg}(i,j)|\right]$, where (i, j) is the index of pixel. When there is an object in the field of view, Bg and Cur are different, and the pixel difference between the two images is not 0. Otherwise, the value is 0. So that it can be used for judging the appearance of an object. We will divide a region in the captured image, which is close to the side where the object enters the view of the cameras, and to monitor whether the object appears in the region by calculating the standard deviation of corresponding image pixels by (1).

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(p_i - \bar{p})^2}{N}} \tag{1}$$

where p_i is the pixel point in the listening area.

As shown in Fig. 3a, the object is outside the listening area when the system initializes. At this moment, the average of pixels in the area is calculated and its value is noted as $\bar{p}$. And σ is calculated continuously before the sorting process.

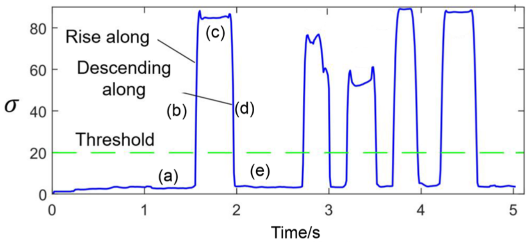


Fig. 4 Variation of σ with object passage

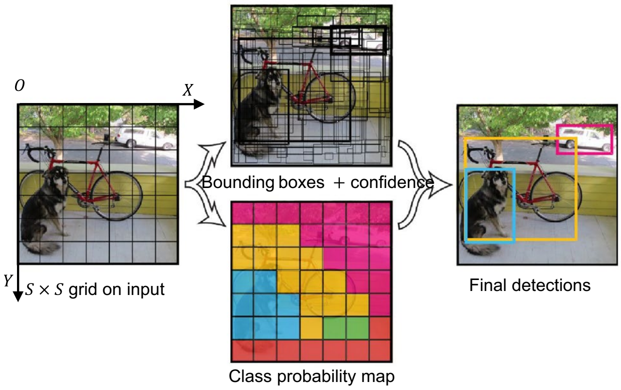


Fig. 5 The concept of YOLO

The test is done for five objects of different shapes and varying spacing passing through the view of the cameras, and the results are shown in Fig. 4. Figure 3a to e correspond to (a) to (e) in Fig. 4, and they represent the object not yet entered, starting to enter, fully staying in, starting to leave and being fully outside the listening area, respectively. When the object enters the listening area, σ moves upward and stays at a high value. When the object leaves the listening area, σ behaves inversely. We set the threshold of σ to be 20 to indicate the appearance of the object.

2.2 The model for object detection

The real time is hard requirement for the efficient sorting. The YOLO framework is designed for real-time object detection, which emphasizes on the speed with acceptable accuracy. The object detection task is performed as a regression problem, and the YOLO framework divides the input image into an $S \times S$ grid through the backbone network. For each cell in the grid, only one object can be detected. To detect object, a fixed number of predicted bounding boxes are predicted. Each predicted bounding box has five parameters,

(x, y) in the coordinate system with the upper left corner as the origin, height (h), width (w), and a confidence score. The confidence score defines how large the box is likely to contain an object. The height and width of the bounding box are normalized according to the resolution of the input image. The coordinate (x, y) is the offset of the cell containing object. In addition, the c-conditional probability of each cell is predicted. Subsequently, the conditional probability of each category is predicted. The YOLO framework maps the output characteristics into a 7 × 7 grid. The main concept of the original YOLO framework is shown in Fig. 5.

The class confidence score $p(c)$ can be calculated by multiplying the conditional probability $p(\text{object})$ and the bounding box confidence score, which is equivalent to intersection over union (IOU) between truth and prediction $IOU_{\text{pred}}^{\text{truth}}$. The concept of the YOLO framework allows predicting both bounding box and class score through the regression layer. As a result, the calculation speed of the model is very fast and can achieve real-time processing.

In order to optimize the model, a hybrid loss function combining local loss and categorical loss is proposed. Rather than predicting the height and width of bounding boxes, YOLO proposes to predict their square roots, so the errors will be reduced especially on large bounding boxes. In addition, to emphasize the accuracy of the bounding box, the loss is multiplied by a coefficient λ_{coord}. By default, the value of λ_{coord} is fixed to 5. This can lead to the problem of class imbalance if many prediction boxes do not contain object. To solve this problem, the classification loss is multiplied by a factor λ_{noobj}. The default value of λ_{noobj} is 0.5. The YOLO loss function can be computed as (2).

$$\begin{aligned} l =& \lambda_{\text{coord}} \sum_{i=0}^{s^2} \sum_{j=0}^{k} q_{ij}^{\text{obj}} \left((x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right) \\ &+ \lambda_{\text{coord}} \sum_{i=0}^{s^2} \sum_{j=0}^{k} q_{ij}^{\text{obj}} ((\sqrt{w_i} - \sqrt{\hat{w}_i}) + (\sqrt{h_i} \\ &- \sqrt{\hat{h}_i})^2) + \sum_{i=0}^{s^2} \sum_{j=0}^{k} q_{ij}^{\text{obj}} (c_i - \hat{c}_i)^2 \\ &+ \lambda_{\text{noobj}} \sum_{i=0}^{s^2} \sum_{j=0}^{k} (c_i - \hat{c}_i)^2 \\ &+ \sum_{i=0}^{s^2} q_i^{\text{obj}} \sum_{c \epsilon \mathrm{C}} (p_i(c) - \hat{p}_i(c))^2 \end{aligned} \tag{2}$$

where various symbols are listed in Table 1.

In order to avoid repeated prediction for the same object, YOLO proposes the use of non-maximum suppression (NMS) technique, which works as the following steps.

1. Rank the predictions based on confidence score.
2. For the current prediction, if any prediction with IOU being greater than 0.5 toward the same class is found, the prediction is updated by the one with the highest confidence.
3. Repeat step 2 until all predictions are checked.

2.3 Optimization of the anchor parameter in the network model

The YOLO algorithm draws on the idea of R-CNN and introduces the anchor mechanism, which is replacing the direct prediction of the object anchor by predicting the offset of the object anchor. An anchor is a set of pre-determined width and height of the bounding box for the detected object.

Table 1 Description of symbols

Symbols	Description
(x_i, y_i)	The ith center coordinate of the ground truth bounding box
$(\hat{x}_i, \hat{y}_i)$	The ith center coordinate of the predicted bounding box
w_i	The width of the ith ground truth bounding box
$\hat{w}_i$	The width of the ith predicted bounding box
h_i	The height of the ith ground truth bounding box
$\hat{h}_i$	The height of the ith predicted bounding box
c_i	The target confidence score
$\hat{c}_i$	The confidence score of the bounding box j in cell i
s	The grid size
k	The number of the predicted bounding boxes
p_i	The ground truth conditional probability of object belongs to a class c
$\hat{p}_i$	The predicted conditional probability of object belongs to a class c
C	All object classes

Table 2 Width (w_A) and height (h_A) of anchors before and after clustering

Type of anchors	Small anchors	Medium anchors	Large anchors	Accuracy
Our anchors	$w_A = 147, h_A = 213$	$w_A = 183, h_A = 228$	$w_A = 233, h_A = 272$	94.87%
	$w_A = 159, h_A = 161$	$w_A = 206, h_A = 290$	$w_A = 253, h_A = 308$	
	$w_A = 162, h_A = 259$	$w_A = 219, h_A = 239$	$w_A = 271, h_A = 345$	
COCO's anchors	$w_A = 10, h_A = 13$	$w_A = 30, h_A = 61$	$w_A = 116, h_A = 90$	66.84%
	$w_A = 16, h_A = 30$	$w_A = 62, h_A = 45$	$w_A = 156, h_A = 198$	
	$w_A = 33, h_A = 23$	$w_A = 59, h_A = 119$	$w_A = 373, h_A = 326$	

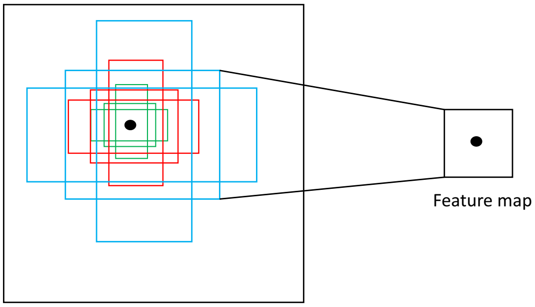


Fig. 6 Anchors suitable for multi-scale detection

YOLO algorithm improves the anchor size manually set in the Fast R-CNN algorithm, and obtains a set of anchors for the training dataset through the K-Means clustering algorithm. The anchor size in the YOLO model from Github is trained according to the COCO dataset, which is suitable for detection of objects such as car, people and cat. For the high-speed sorting in the assembly line, the objects to be detected are regular shape such as rectangle and circle, which are different from the object in the COCO dataset.

In order to make the anchor size more suitable for the object to be detected, K-Means clustering algorithm is adopted to regenerate the anchor size. The objective function of K-Means uses distance commonly, such as Euclidean distance and Manhattan distance, as an index of similarity (Li and Wu 2012). It groups objects that are close in distance into a cluster, and the compact and independent clusters are ultimately formed.

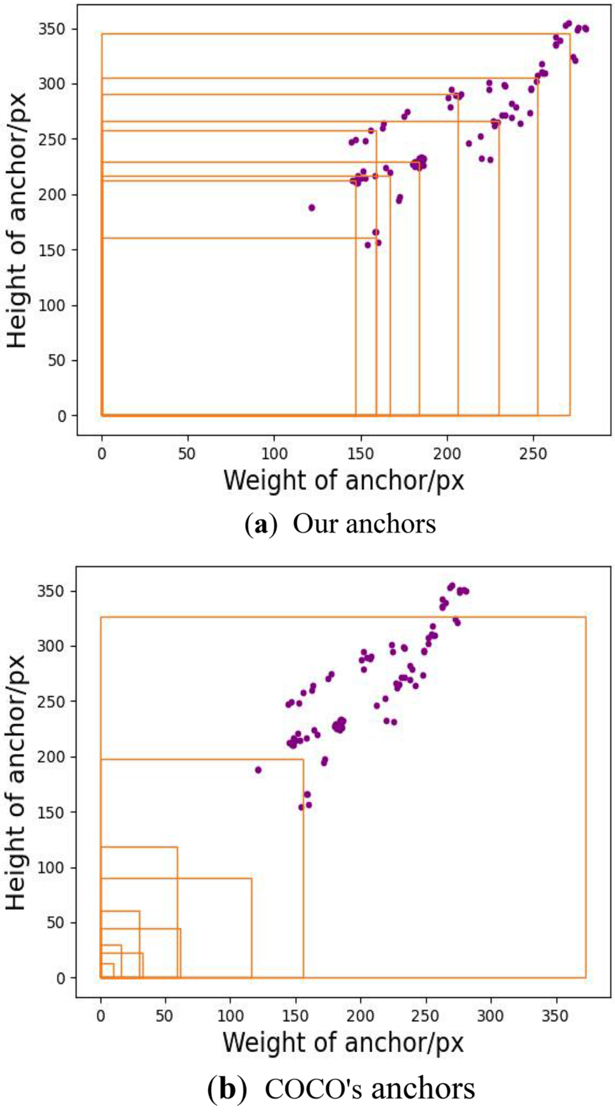
If Euclidean distance is used in K-means clustering algorithm, the anchor obtained in the YOLO model will lead to large estimation error of bounding box, especially when the object is large. To let IOU be closed to unity, regardless of the size of the bounding box, the objective function D as in (4) in the clustering process.

$$IOU(s_b, c_b) = \frac{s_b \cap c_b}{s_b \cup c_b} \tag{3}$$

$$D(s_b, c_b) = 1 - IOU(s_b, c_b) \tag{4}$$

where s_b represents the sample, c_b represents the cluster center, $IOU(s_b, c_b)$ represents the intersection ratio of the cluster center and the sample.

The anchor in YOLOv5 can be used to achieve multi-scale detection, so the anchor parameters are also applicable to large, medium and small targets, with 3 shapes of anchor



(a) Our anchors

(b) COCO's anchors

Fig. 7 Distribution of anchors generated by K-Means and anchors of COCO dataset in our dataset

per category, as shown in Fig. 6. In this paper, K-Means algorithm is performed on the dataset of object to be sorted, with the number of cluster center $K = 9$. The changes before and after clustering are shown in Fig. 7, where we can see that the width and height of the anchor parameters generated by K-Means are suitable for 94.87% of images in our dataset. The anchor parameters generated by the clustering algorithm are shown in Table 2.

2.4 Improvement of model reliability under extreme light intensity

In the automated sorting line, the images of object are often taken under complex and unfavorable environment, such as lighting and background. It is desirable to enhance the original image by algorithm. This will enhance the visual effect of the image or emphasize its overall or local features, so that the reliability of detection is improved.

In industrial practice, images are sometimes taken with low-brightness. In this situation, using the method of image smoothing alone cannot effectively remove the noise and extract object features. Besides, the methods of histogram equalization change the image contrast by calculating the histogram of the image and redistributing the brightness to make the object be more visible. The popular methods of histogram equalization include adaptive histogram equalization (AHE) (Pizer et al. 1987) and contrast limited adaptive histogram equalization(CLAHE) (Reza 2004). CLAHE is used in this paper, which can effectively limit the unfavorable amplification of noise. The effect after image enhancement is shown in Fig. 8.

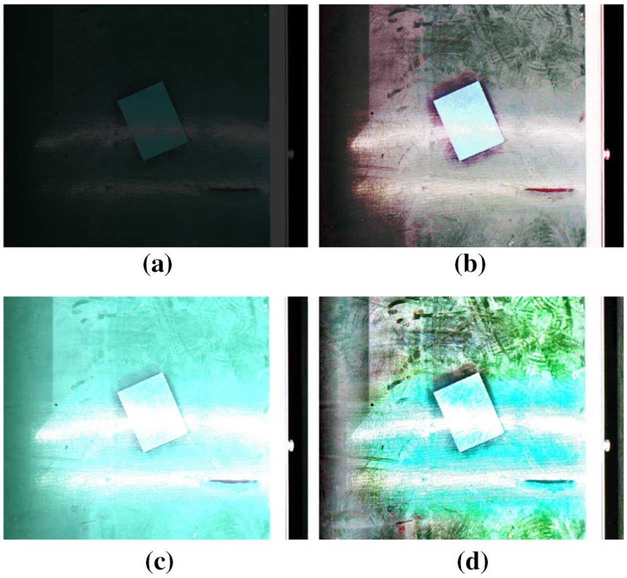


Fig. 8 The effect after image enhancement: **a** and **c** are taken under extreme lighting. **b** and **d** are the images after image enhancement

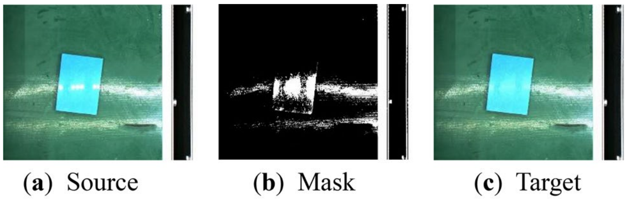


Fig. 9 Images before and after reflection removal

The contrast of images is effectively enhanced through image histogram, so that the undetectable problem is solved. However, there is still the problem that object with smooth surface is prone to specular reflection under bright light. This can lead to wrongly extracting the reflective area of the object surface as the edge of the object. To solve this problem, a mask image is synthesized as a pixel value of 0 or 1 generated from the original image. And it is fused with the original image using the Poisson fusion method, so that the reflective area is removed in the image. The results are shown in Fig. 9.

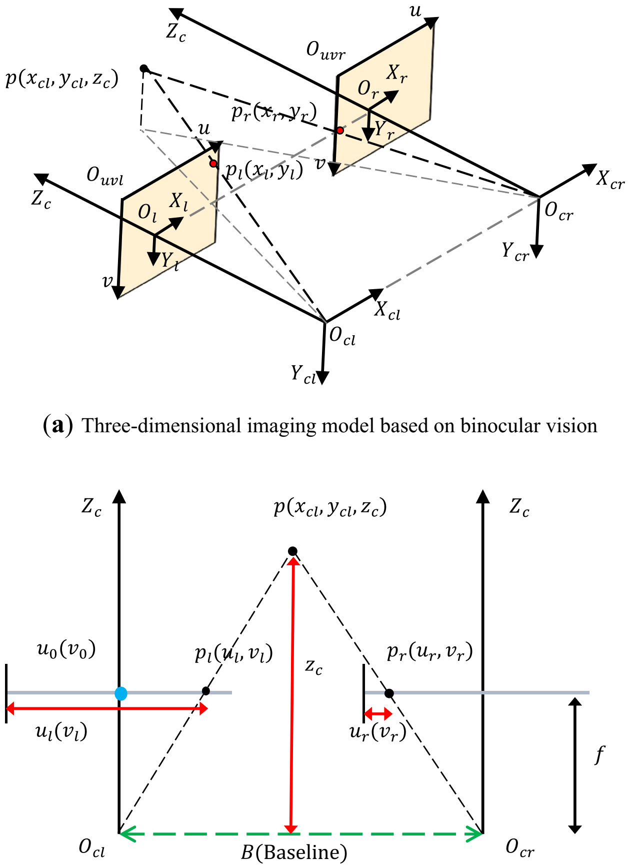
2.5 Position of objects with binocular camera

After optimizing the parameters of YOLOv5 model and processing the images, the YOLOv5 model will output a predictive box of object $B_{\text{roi}}(x, y, w, h)$ as the ROI for the posture measurement. The ROI is segmented from the original image to reduce the interference of various environmental factors such as background. In this step, we complete the classification of object and obtain the approximate position of the object in image. Then, we will focus on implementing stereo matching of objects to achieve the purpose for obtaining posture of object.

In this paper, a method based on binarization is used to extract the edge of imaging object captured with a pair of calibrated cameras, and the minimum outer rectangle can be generated. $p_l(x_l, y_l)$, $p_r(x_r, y_r)$ are the center points of the minimum outer rectangle, and $p(x_{cl}, y_{cl}, z_c)$ or (x_{cr}, y_{cr}, z_c) is the point of the object in the camera coordinate system. As shown in Fig. 10a, the equations for solving the coordinates of the object in the left/right camera coordinate system by the triangle similarity principle are given as

$$\begin{aligned} \frac{x_l}{x_{cl}} = \frac{y_l}{y_{cl}} = \frac{f}{z_c} \\ \frac{x_r}{x_{cr}} = \frac{y_r}{y_{cr}} = \frac{f}{z_c} \end{aligned} \tag{5}$$

where f is the focal length of the camera.



(a) Three-dimensional imaging model based on binocular vision

(b) Two-dimensional imaging model based on binocular vision

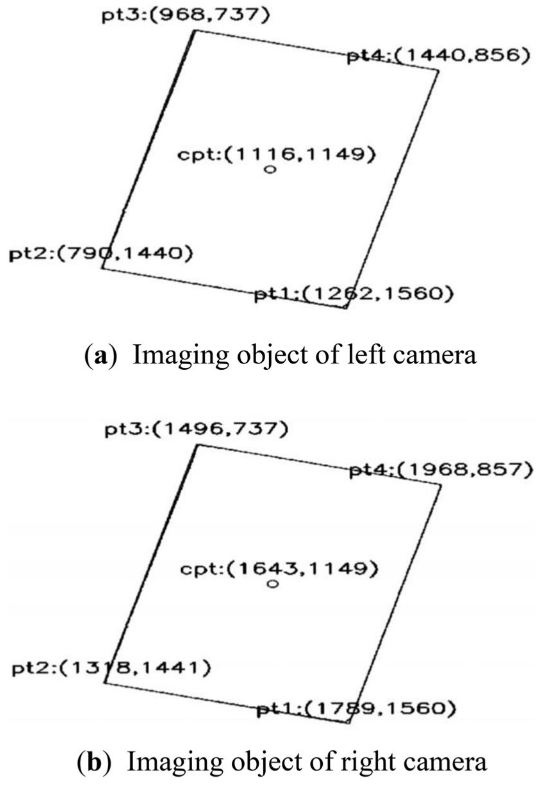
Fig. 10 Distance measurement based on binocular vision

The two-dimensional imaging model of point p is shown in Fig. 10b. The coordinate of p is expressed in pixels as $p_l(u_l, v_l)$. The constraint formulas for $p_l(u_l, v_l)$ and $p_l(x_l, y_l)$ are given as

$$
\begin{aligned}
u_l &= \frac{x_l}{d_x} + u_0 \\
v_l &= \frac{y_l}{d_y} + v_0
\end{aligned}
\tag{6}
$$

where (u_0, v_0) is the center of the imaging plane and d_x, d_y denote the physical size of a single pixel point in the X_l and Y_l directions of the $O_l X_l Y_l$ coordinate system. This is also true for $p_r(u_r, v_r)$ and $p_r(x_r, y_r)$.

Note that B is the baseline distance between the two cameras, d is the parallax and its value is $(u_l - u_r)d_x$. So the formula for z_c is given by the similar triangle principle as



(a) Imaging object of left camera

(b) Imaging object of right camera

Fig. 11 Coordinates of each eigenvalue after extracting the edge

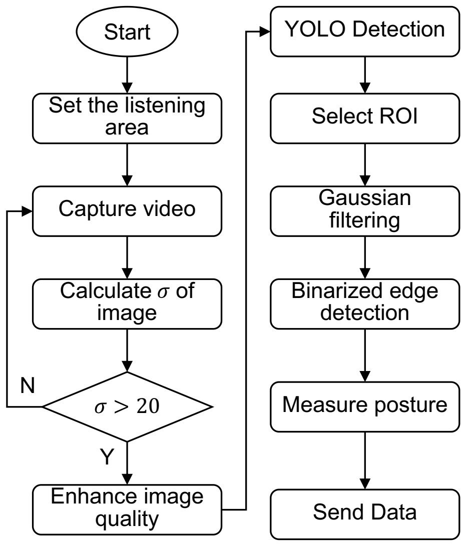


Fig. 12 The procedures of our method

$$\frac{B-(u_l-u_r)d_x}{B}=\frac{z_c-f}{z_c} \tag{7}$$

where $z_c=\frac{Bf}{d}$.

Finally, the equations for the coordinate of $p(x_{cl},y_{cl},z_c)$ in the left camera coordinate system are given as

$$\begin{aligned} z_c &= \frac{f_x B}{d} \\ x_{cl} &= \frac{z_c}{f_x}(u_l-u_0)d_x \\ y_{cl} &= \frac{z_c}{f_y}(v_l-v_0)d_y \end{aligned} \tag{8}$$

In (8), f_x,f_y are the focal length of the camera in the X and Y.

As shown in Fig. 11, a, b show the results of the left and right camera imaging objects after edge detection, respectively. $pt_n(n=1,2,3,4)$ are the coordinates of the corner points in the minimum outer rectangle, whose center point is cpt. The coordinates of the center point are brought into the formula (5) to calculate the position information of the object in the camera coordinate system. Also we define θ as the angle between the minimum outer rectangle and the horizontal line.

The overview of workflow for the proposed sorting and positioning method is shown in Fig. 12.

3 Experiments

3.1 Training dataset

We install a pair of HIKVISION cameras above the conveyor belt to capture five type objects, which are paper box, round plastic block, rectangular plastic block, hexagon blocks and triangle blocks at random angles under different lighting conditions. A total of 304 images, including 60 white paper boxes, 74 round plastic blocks, 70 rectangular plastic blocks, 50 hexagon blocks and 50 triangle blocks are selected for annotation and divided into 70% training set, 20% validation set and 10% test set. In order to simulate the images under a wide range of illumination and increase the size of the experimental data, a variety of images under extreme illumination are generated by changing the light contrast of images.

3.2 Training

We use the pre-trained weight file of COCO dataset, add our dataset and adjust the anchor parameters of the model, which generated by K-Means clustering 2.3. The YOLOv5 model is trained using Google Colab, which provides free access to powerful GPUs.

We set the number of epoch to be 300 to train the upper layers of the model to classify the objects. It takes about 70 min to train this model. Figure 13 shows various performance metrics of our model with both the training and validating sets.

There are three different type of loss shown in Fig 13: box loss, objectness loss and classification loss. The box loss represents how well the algorithm can locate the center of an object and how well the predicted bounding box covers an object. Objectness is essentially a measure of the probability that an object exists in a proposed ROI. The high objectivity implies that the image window is likely to contain an object. Classification loss gives an idea of how well the algorithm can predict the correct class of a given object. The model improved swiftly in terms of precision, recall and average precision before stabilizing after about 200 epochs. The box, objectness and classification losses of the validation data also show a rapid decline until around epoch 200. So we select the weight file of optimized model at epoch 200.

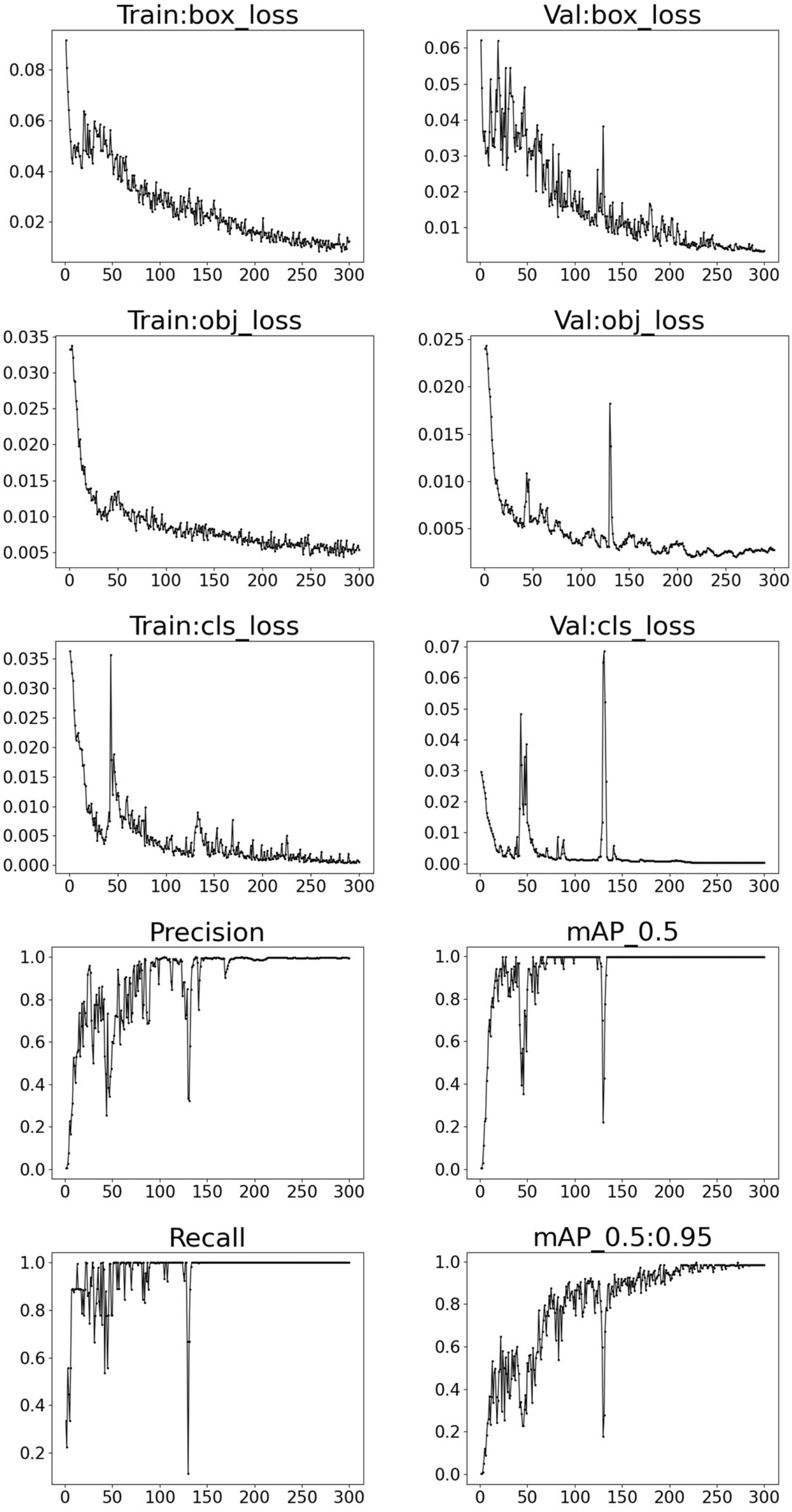
3.3 Experimental analysis

After training our model, we use the TensorRt framework, which enables the model trained in Python to run in C++. We implant the model into the vision system. To evaluate the performance of our detection strategy, we compare it with the YOLOv5 model + edge extraction method (Arjun et al. 2020), without adding image enhancement and anchor parameter optimization, YOLOv5 with anchor parameter optimization but no image enhancement and the only edge extraction method. The edge extraction method is represented by binarization based method (Yu and Yan 1997), because this method is used commonly in edge extraction.

In order to simulate illumination changes in real environment, we generate images under different lighting by changing the luminance contrast of images taken under natural light. We set the range of lighting intensity λ from 0.2 to 2.0 with the step of 0.2. Notably, when $\lambda=1.0$, the image is under natural light. $\lambda \in [0.2,1.0)$ implies that the image is under low light, $\lambda \in (1.0,2.0]$ implies that the image is under strong light. The number of datasets for each λ is 190, giving total 1900 images.

Four methods are tested with 1900 images, and the accuracies at different light intensities are shown in Fig. 14. The blue line indicates the accuracy when only edge detection is used, which is the worst. The yellow line indicates the

Fig. 13 Various metrics of our model over the training epochs for the training and validating sets



result when both the edge detection and YOLOv5 are used, which has the improved accuracy under different illumination compared with the first method. The green line indicates the result with addition of anchor optimization in YOLOv5 on top of the second method, and the accuracy is further improved. The red line is the result with our method, which adds image processing on top of the third method, and it has the best accuracy among the four methods. Subsequently, the experiment is conducted with mixed samples taken under different light intensities, and the results are summarized in Table 3, which shows that the anchor optimization makes an average accuracy improvement by 12.63%, while the image processing makes an average accuracy improvement by 14.21%. Some images of detection for the test data of our method are shown in Fig. 15.

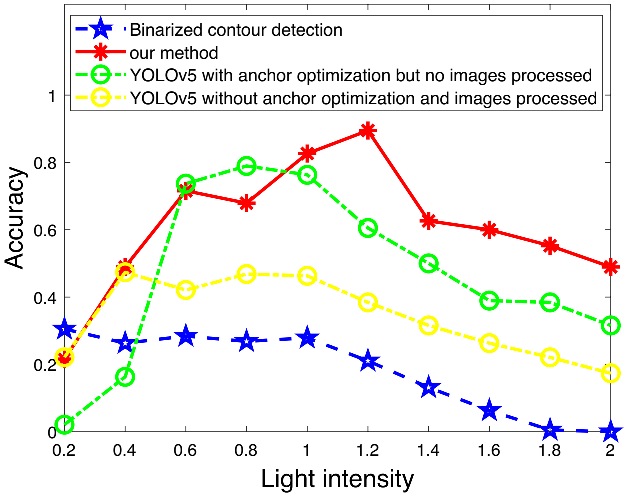
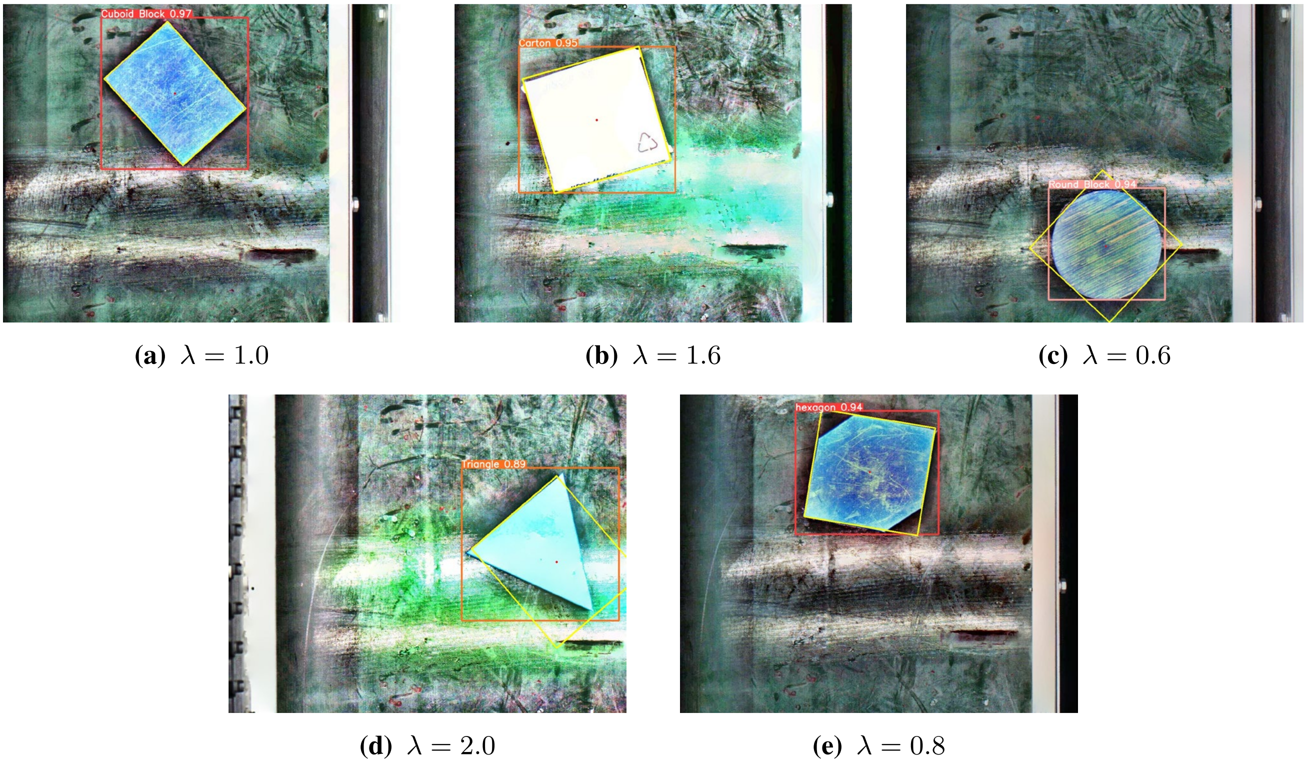


Fig. 14 Comparison of detection accuracy by various methods

Table 3 Accuracy of mixed detection

Method	Accuracy %
Binarization	18.11
YOLOv5 without anchor optimization and image enhancement + Binarization	34.05
YOLOv5 with anchor optimization but no image enhancement + Binarization	46.68
YOLOv5 with anchor optimization and image enhancement + Binarization	60.89



(a) $\lambda = 1.0$

(b) $\lambda = 1.6$

(c) $\lambda = 0.6$

(d) $\lambda = 2.0$

(e) $\lambda = 0.8$

Fig. 15 Detection results under different lighting intensity λ, and the yellow rectangle is the minimum outer rectangle for calculating positioning

4 Conclusion

In this work, we aim to develop a vision system for real-time sorting in the automation lines. Combining the YOLOv5 framework of object detection and position by binocular machine vision, a series of methods such as image enhancement and *K*-Means clustering are proposed to achieve the classifying and positioning of the object, so that the influence such as change of illumination and reflection on the surface of object are reduced. This improves the robustness of the sorting system based on machine vision. Through experimental verification, our method gets good detection accuracy on a low-cost hardware such as a pair of cameras. Finally, the authors would like to acknowlege Xiaonan Fan for his help on setting up the testbed.

Funding This work was supported in part by the "Leading Goose" R&D Program of Zhejiang Province under Grant 2022C01114, in part by the National Natural Science Foundation of China under Grant U20A20282 and Grant 51905523, in part by the Key Research and Development Plan of Zhejiang Province under Grant 2021C01069, in part by the Ningbo Science and Technology Innovation Key Projects under Grant 2018B10069, in part by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant XDA22000000.

References

Alexey Bochkovskiy, H.-Y.M.L. Chien-Yao Wang: YOLOv4: Optimal speed and accuracy of object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

Arjun, B., Hari, V., Chandran, D., Varghese, A.B.: Packing automation in a high variety conveyor line via image classification (2020)

Batchelor, B., Waltz, F.: Machine vision for industrial applications. Intelligent Machine Vision: Techniques, Implementations and Applications, 1–29 (2001)

Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2015)

Li, Y., Wu, H.: A clustering method based on K-Means algorithm. Phys. Proced. **25**, 1104–1109 (2012)

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: Single shot multibox detector. In: Computer Vision – ECCV 2016, pp. 21–37 (2016)

Liu, T., Chen, Z., Yang, Y., Wu, Z., Li, H.: Lane detection in low-light conditions using an efficient data enhancement: Light conditions style transfer. In: 2020 IEEE Intelligent Vehicles Symposium (IV), pp. 1394–1399 (2020)

Machaca Arceda, V., Laura Riveros, E.: Fast car crash detection in video. In: 2018 XLIV Latin American Computer Conference (CLEI), pp. 632–637 (2018)

Mirhaji, H., Soleymani, M., Asakereh, A., Mehdizadeh, S.A.: Fruit detection and load estimation of an orange orchard using the yolo models through simple approaches in different imaging and illumination conditions. Comput. Electron. Agric **191**, 106533 (2021)

Modi, C.K., Desai, N.P.: A simple and novel algorithm for automatic selection of roi for dental radiograph segmentation. In: 2011 24th Canadian Conference on Electrical and Computer Engineering(CCECE), pp. 504–507 (2011)

Ng, P.C., Henikoff, S.: SIFT: predicting amino acid changes that affect protein function. Nucl. Acids Res. **31**(13), 3812–3814 (2003)

Pizer, S.M., Amburn, E.P., Austin, J.D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J.B., Zuiderveld, K.: Adaptive histogram equalization and its variations. Comput. Vis. Graph. Image Process. **39**(3), 355–368 (1987)

Prasetyo, E., Suciati, N., Fatichah, C.: A comparison of yolo and mask r-cnn for segmenting head and tail of fish. In: 2020 4th International Conference on Informatics and Computational Sciences (ICICoS), pp. 1–6 (2020)

Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

Redmon, J., Farhadi, A.: YOLOv3: An incremental improvement. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

Reza, A.M.: Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. J. VLSI Signal Process. Syst. Signal Image Video Technol. **38**, 35–44 (2004)

Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. In: 2011 International Conference on Computer Vision, pp. 2564–2571 (2011)

Sun, X., Jiang, Y., Ji, Y., Fu, W., Yan, S., Chen, Q., Yu, B., Gan, X.: Distance measurement system based on binocular stereo vision. IOP Conf. Ser. Earth Environ. Sci. **252**(5), 052051 (2019)

Tsang, W.H., Tsang, P.W.M.: Suppression of false edge detection due to specular reflection in color images. Pattern Recogn. Lett. **18**(2), 165–171 (1997)

Wang, X., Han, T.X., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 32–39 (2009)

Wang, Y., Wang, C., Zhang, H., Dong, Y., Wei, S.: Automatic ship detection based on retinanet using multi-resolution gaofen-3 imagery. Remote Sensing **11**(5) (2019)

Yin, D., Tang, W., Chen, P., Yang, B.: An improved algorithm for target detection in low light conditions. J. Phys. Conf. Ser. **2203**(1), 012045 (2022)

Yu, D., Yan, H.: An efficient algorithm for smoothing, linearization and detection of structural feature points of binary image contours. Pattern Recogn. **30**(1), 57–69 (1997)

Zou, B., De Koster, R., Gong, Y., Xu, X., Shen, G.: Robotic sorting systems: Performance estimation and operating policies analysis. Transp. Sci. **55**(6), 1430–1455 (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Wujie Ge received the B.E. degree from the North China Electric Power University, Beijing, China, in 2018. He is currently pursuing the M. E. degree with the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. His research interests include object detection and robotic control.

Silu Chen received his B.Eng. and Ph.D. degrees in Electrical Engineering, both from the National University of Singapore, in 2005 and 2010 respectively. From 2010 to 2011, he was with Manufacturing Integration Technology Ltd, a Singapore-based semiconductor machine designer, as a senior engineer on motion control. From 2011 to 2017, he was a scientist in Mechatronics Group, Singapore Institute of Manufacturing Technology, Agency for Science Technology and Research. Since 2017, he has been with Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China, as a professor. His current research interests include high-speed, high-precision motion control and industrial automation. Dr Chen is a technical editor of IEEE/ASME Transactions on Mechatronics and an associate editor of International Journal of Intelligent Robotics and Applications.

Hua Hu received the B.E. degree from the Hebei University of Technology, TianJin, China, in 2020. He is currently pursuing the M. E. degree in mechanical engineering with Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. His research interests include adaptive control and neural network control.

Tianjiang Zheng received the B.S. degree in automation and the M.S. degree in control theory and control engineering from Shenyang Ligong University, Shenyang, China, in 2006 and 2009, respectively, and the Ph.D. degree in robotics, cognition, and interaction technology from Genova University. From 2013 to 2015, he was a Research Assistant with the Precision Drive and Advanced Robot Group, Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences. Since 2016, he has been a Senior Engineer with the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, and the Zhejiang Key Lab-oratory of Robotics and Intelligent Manufacturing Equipment Technology and Engineering, Ningbo, China. He is also a Supervisor of the University of Chinese Academy of Sciences, Beijing, China. He has been published more than 20 articles and papers. He has appliedmore than ten patents. His research interests include the control of mobile robots, mobile manipulations, continuum manipulations, and parallel robots.

Zaojun Fang received his B. Eng degree from University of Science and Technology, Anshan, Liaoning, China, in 2005, and his M. Eng and Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2008 and 2011, respectively. He is currently a professor at the Zhejiang Key Laboratory of Robotics and Intelligent Manufacturing Equipment Technology, Ningbo Institute of MaterialsTechnology and Engineering, Chinese Academy of Sciences, Ningbo, China. His current research interests include robot vision, robot control and automation.

Chi Zhang received the B.E. degree and M.E. degree from Xi'an Jiaotong University in 1999 and 2002 respectively, and the Ph.D. degree from Nanyang Technological University, Singapore in 2007, all in Electrical and Electronic Engineering. He is currently a professor and the director of the Zhejiang Key Laboratory of Robotics and Intelligent Manufacturing Equipment Technology, and the director of Institute of Advanced Manufacturing Technology, both being subsidiaries of Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. His current research interests include precise actuator design, precision motion control, permanent magnet motor and its control, advanced robotics and intelligent manufacturing equipment.

Guilin Yang received the B.E. and M.E. degrees from the Jilin University of Technology, Jilin, China, in 1985 and 1988, respectively, and the Ph.D. degree from Nanyang Technological University, Singapore, in 1999, all in mechanical engineering. He is currently a professor and the vice president of Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. He was the founding director of the Zhejiang Key Laboratory of Robotics and Intelligent Manufacturing Equipment Technology, Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China. His research interests include precision actuators, parallel-kinematics mechanisms, modular robotic systems, and industrial robotics. He has published over 350 technical papers in refereed journals and conference proceedings. Prof. Yang was a recipient of the R&D 100 Awards in 2014.