Multiple Object Detection of Workpieces Based on Fusion of Deep Learning and Image Processing*

Yi Lei, Xifan Yao+, Wocheng Chen, Junming Zhang
School of Mechanical and Automotive Engineering
South China University of Technology
Guangzhou, Guangdong, China
+mexfyao@scut.edu.cn

Jorn Mehnen, Erfu Yang
Faculty of Engineering
University of Strathclyde
Glasgow, U.K
{jorn.mehnen, erfu.yang}@strath.ac.uk

Abstract—A workpiece detection method based on fusion of deep learning and image processing is proposed. Firstly, the workpiece bounding boxes are located in the workpiece images by YOLOv3, whose parameters are compressed by an improved convolutional neural network residual structure pruning strategy. Then, the workpiece images are cropped based on the bounding boxes with cropping biases. Finally, the contours and suitable gripping points of the workpieces are obtained through image processing. The experimental results show that mean Average Precision (mAP) is 98.60% for YOLOv3, and 99.38% for that one by pruning 50.89% of its parameters, and the inference time is shortened by 31.13%. Image processing effectively corrects the bounding boxes obtained by deep learning, and obtains workpiece contour and gripping point information.

Keywords—workpieces detection; deep learning; pruning filters; image processing

I. INTRODUCTION

As a classic task in industrial scenarios, workpiece detection is critical in many flexible assembly units, automated factories and other production environments. Under the background of Industry 4.0, the tasks of industrial production are gradually changing from large-scale production to personalized customization and single-piece/small-batch production [1]. As a result, the workpiece detection becomes a multi-target and irregular detection, which places higher requirements on the automated detection technology.

In terms of detection methods, machine vision has been widely used because of its non-contact and high efficiency. However, traditional machine vision algorithms have problems such as difficulty in feature definition and extraction, and inaccurate multi-target positioning. In recent years, the rapid development of artificial intelligence such as deep learning has largely solved these problems. With the increasing demand and complexity of tasks, pure deep learning methods tend to continuously expand the scale of the network. For instance, ExtremeNet [2] draws an octagon by detecting the four poles and center point of the target object, making the bounding box more fit the target object, but the price is that its parameter scale is significantly larger than that of rectangular bounding box network YOLOv3 [3]. Using pure deep learning methods to improve the network output form in one step often requires larger parameter scale and more complex backbone network. This not only requires better equipment to support its computing power, but also increases the inference time, making it difficult to achieve video-level detection requirements in industrial scenarios.

In view of the above problems, using YOLOv3 to automatically extract the features of workpieces and locate their rectangular bounding boxes for the multi-objective and personalized workpiece detection task. In order to avoid the problem of over-parameterization, based on basic convolutional neural network (CNN) pruning strategy [4], an improved pruning filter strategy is designed for the residual structure of Darknet-53, which is the backbone network of YOLOv3. For multiple consecutive residual identity blocks [5], the kernels of the last layer in each block need to be pruned in the same order. In this strategy, the hyperparameter α is set as the evaluation weight, and the ℓ_1-norm of the corresponding kernels are assigned and integrated according to the evaluation weights in order of hierarchy. The unified pruning order is determined after the integrated result is sorted. We use the pruning strategy to test Darknet-53 (tiny) on the CIFAR100 dataset [6] and test YOLOv3 on the workpieces dataset. Finally, based on the detected rectangular bounding boxes, a cropping bias is set according to the type of the workpieces to increase the pixel coverage area, and the original image is cropped to extract the contours and appropriate gripping points of the workpieces through traditional image processing.

II. RELATED WORK

Workpiece detection integrates the classification and positioning task of the workpieces. According to the algorithms used, it can be generally divided into two categories: template-based matching algorithm and learning-based classifier algorithm. The template matching uses traditional machine vision algorithms to process image pixels, and matches the template's edges or corners with the template to determine the positions and categories information of the workpieces. For example, based on Hu moment invariance extraction, the contours of the workpieces are extracted and matched with the template to identify the target workpieces [7]. Obtaining the contours information of the workpieces through the Canny algorithm, combining the probabilistic Huff transform and Freeman chain code to identify the shape and calculate the center space coordinates of the workpieces [8]. Since such algorithms mostly use global search, they are time-consuming [9], and are sensitive to factors such as shadows and noise [10, 11]. It is necessary to adjust the image preprocessing method according to the specific usage scenarios. Therefore, they are usually used in scenarios with fewer types of workpieces, simple features, high discrimination, and stable production environment.

Learning-based classifier algorithms are representative of machine learning algorithms that have been applied in workpiece detection tasks, such as support vector machine (SVM), and AdaBoost [12-14]. Since such classifiers need to define features manually, they still cannot be used in the case

of complex features. In recent years, deep learning has emerged because of its obvious advantages in deep and abstract feature extraction of workpieces, without the need to manually define features, and being insensitive to environmental interference factors. For example, CNNs are used to extract the edges and corners of cube candidate objects, and a decision tree is established according to detection probability and geometric relationship to identify the type of workpieces [15]. Using ResNet-34 as a feature extractor, a deep transfer learning method named cold-to-hot training that transfers prior knowledge from labeled synthetic images to unlabeled real images for workpiece viewpoints estimation [16]. Taking the standardized depth map as input to train a fully CNN (convolution and deconvolution neural network, the convolution part is composed of VGG16 loaded with a pre-trained model), and using the segmentation mask as the output. Segmenting the workpiece area from the non-workpiece area in the point cloud data helps to increase the number of detected workpieces and estimate the correct object poses [17]. Using the position image of the workpieces to train the CNN to push the position deviation of the mechanical part for accurately placing the workpieces on the fixture [18]. Thanks to the powerful fitting capabilities of various deep learning network frameworks, although workpiece detection tasks are more diverse and complex, they can do tasks that traditional algorithms cannot.

However, the complexity of the task requires the corresponding network scale to match, which means that more computing power and memory are needed. At the same time, it has become a trend to deploy neural networks to hardware platforms with limited computing capabilities such as mobile terminals [19], so lightening of the networks has always been one of the important research directions for deep learning. At present, common network compression algorithms include network pruning, network distillation, and network decomposition [20]. Among them, pruning the convolution kernels of CNN is an effective method that can remove redundant parameters to avoid network performance overflow. It is true that network architecture is more important than inherited weights after pruned as pointed out by [21], but for industrial applications, loading pre-trained models from existing network structures for transfer learning [22] can make tasks converge with less data. After pruning and retraining the trained model, we can quickly find a network model with the appropriate structure and parameter scale for specific tasks. Since the residual structure [5] was proposed, it has been widely used in various CNNs. For the pruning of the residual structure, the main difficulty appears in the pruning standard of the last convolutional layer. The representative solution is to choose not to prune [23] or use the ℓ_1-norm ordering of the shortcut convolutional layer as the standard [4]. Darknet-53 only uses the identity block in series, and we have improved this pruning situation.

III. WORKPIECE DETECTION METHOD

As described above, in the multi-target detection of workpieces, it is necessary to determine the type and pose of the workpieces. This paper proposes a workpiece detection method that combines deep learning and traditional machine vision algorithms. It combines the advantages of deep learning in workpiece simple positioning, and image processing to detect workpiece contours in pixel processing.

The proposed workpiece detection is divided into three stages: training and pruning, detection and cropping, and traditional machine vision processing, as shown in Fig. 1.
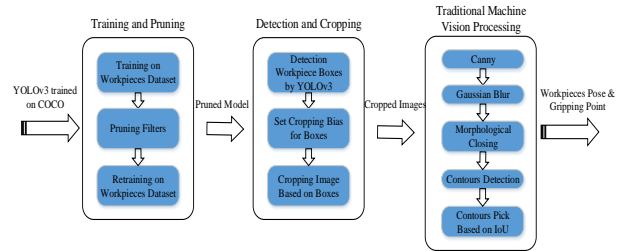

Fig. 1. Three stages of proposed workpiece detection

A. Training and Pruning

The YOLOv3 is loaded with the trained weights on the COCO dataset [24] and trained on the workpieces dataset. Transfer learning strategy enables YOLOv3 to quickly converge with fewer training samples on the workpiece dataset. Then, the network is pruned in kernel granularity by using the improved residual structure pruning strategy and retrains. With the accuracy of the network maintained, we find the optimal parameter scale for the workpiece dataset under the YOLOv3 network structure.

YOLOv3 is trained on the workpiece dataset as in [3] . For anchor prior mechanism, the k-means algorithm is used to obtain 9 bounding boxes prior on the training set of the workpiece dataset. During training and verification, the network input is images with size of 416×416. Since the type of the workpiece dataset is 10, the specifications of the three output layers (batch, height, width, channels) of YOLOv3 become (batch, 13, 13, 45), (batch, 26, 26, 45), and (batch , 52, 52, 45). Therefore, after loading the weights besides the output layers, we use he-normal [25] to initialize the output layers. The effect of YOLOv3 on workpiece detection after trained is shown in Fig. 2.
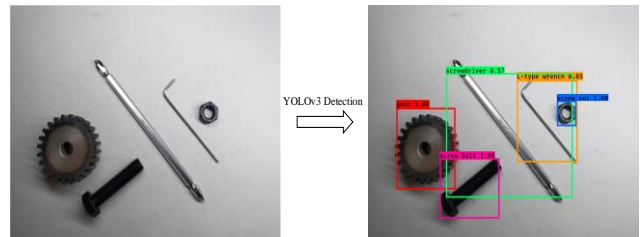

Fig. 2. YOLOv3 workpiece detection

YOLOv3 achieves excellent performance on the workpiece dataset, so we proceed to prune the network. In this paper, the sequential structure convolutional layer is pruned using the method in [4]. And a greedy pruning strategy is adopted in the continuous multi-layer pruning, that is, the influence of the convolutional pruning of the previous layer is eliminated in advance when the pruning order of a certain layer is calculated.

In Darknet-53, the meta-structure of serial identity blocks is shown in Fig. 3, where conv stands for Convolution-Batch Normalization [26]-Leaky Relu. This paper designs a pruning strategy for the meta-structure. The process of pruning n% is as follows:

1) For each block, calculate ℓ_1-norm of $conv_{i,1}$:

$$s_{i,1} = \|\mathcal{K}_{i,1}\|_1 \qquad (1)$$

Prune $conv_{i,1}$ kernels corresponding to the smallest n% of $s_{i,1}$ index, and then prune the corresponding dimensions of $conv_{i,2}$;

2) Calculate ℓ_1-norm of $conv_0$ and $conv_{i,2}$ for each block, and use (1) to obtain s_0 and $s_{i,2}$. Set the hyperparameter α to calculate:

$$\omega \leftarrow \alpha s_0 + (1-\alpha)s_{1,2} \qquad (2)$$

For $i = 2:k$ with a step of 1, calculate:

$$\omega \leftarrow \alpha\omega + (1-\alpha)s_{i,2} \qquad (3)$$

Prune $conv_0$ and $conv_{i,2}$ kernels corresponding to the smallest n% of ω index, and then prune the corresponding dimensions of $conv_{i,1}$ and next meta-structure $conv_0$;

3) Load the pruned meta-structure weights into the new model for retraining. The hyperparameter α is selected according to the training results to obtain a model with the best accuracy.
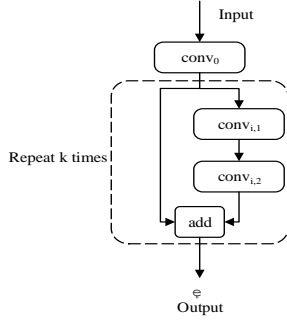

Fig. 3. Series connection of identity blocks

B. Detection and Cropping

Inputting the detection image into the pruned YOLOv3 can obtain the bounding boxes of the workpieces. According to the type of the workpieces, corresponding cropping biases are added to the four directions of the bounding boxes, and the cropped pixel coverage area is enlarged to avoid losing part of the workpiece information, as shown in Fig. 4. Therefore, the multi-target detection problem is approximately transformed into a single-target detection problem.

The cropped images of the workpiece test dataset are processed by traditional image processing, the mean Average Precision (mAP) is calculated from the bounding rectangle of the workpiece contours and the data labels to represent the accuracy of the detected contours. Under certain cropping bias, mAP reaches the optimal value, thereby determining the cropping bias of each type of workpieces.
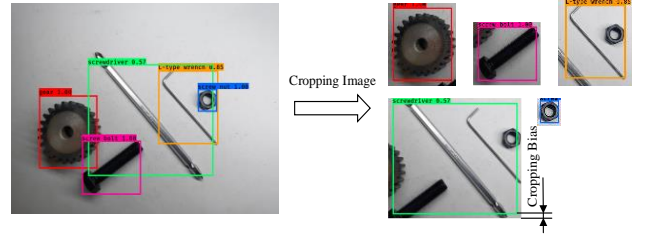
Fig. 4. Cropping image based on boxes with cropping bias

C. Traditional Machine Vision Processing

After cropped, multiple workpiece images are obtained, and the images are sequentially subjected to Canny algorithm [27], Gaussian blur, morphological closing, and find contours. Then, the polygon fitting curves of the contour point sets (cylindrical workpieces such as gears, bearings and screw dies are circumscribed circles) are used as the workpiece contours. Finally, calculate the Intersection over Union (IoU) by detecting the bounding rectangles of the contours (Fig. 5(f) red rectangles, cylindrical workpieces directly use the contours) and the bounding boxes of deep learning (Fig. 5(f) green rectangles), and select the detection contour corresponding to the maximum IoU as the workpiece contour, as shown in Fig. 5.
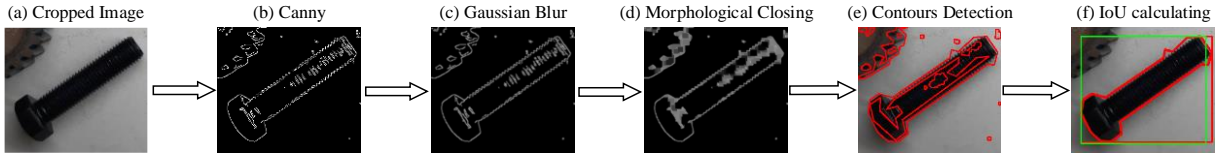

Fig. 5. The process of workpiece contour detection

Obtaining the contour of the workpiece determines the pose of the workpiece, and then finds suitable gripping points for the workpieces. For cylindrical workpieces, directly select two points in the diameter direction on the circular contour of the workpieces as gripping points. For non-cylindrical workpieces, the circumscribed oblique rectangle of the workpiece contour is first obtained, then the midpoints of the longer sides of the rectangle are connected, and the intersection points of the connecting line and the detected contour are used as the gripping points. As shown in Fig. 6, the red curves represent the detection contours, the blue rectangles represent the circumscribed oblique rectangles of the contours, and the purple arrows represent the gripping points of the workpieces.
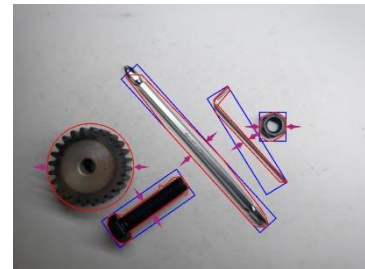

Fig. 6. Workpiece detection contours and grip points

IV. EMPIRICAL ANALYSIS

Based on the above methods, this study first tests the improved pruning strategy on the CIFAR100 dataset using the Darknet-53 (tiny), and then trains and prunes YOLOv3 on the workpieces dataset to obtain the network model with the best parameter amount. Finally, we analyze the accuracy of the workpiece contours obtained after processing the cropped images.

A. Pruning Strategy Test

Based on TensorFlow [28], the Darknet-53 (tiny) network framework shown in Table I was tested on CIFAR100 dataset. Convolutional represents the Conv-BN-Leaky Relu series structure. We set each layer of Conv's padding to SAME, a weight decay of 0.0005, momentum of 0.9, initialized weight with he-norm [25], and Leaky Relu with an alpha of 0.1. We trained it on a single GPU with batch size of 128 based on stochastic gradient descent (SGD). The initial learning rate was set to 0.01, changed to 0.1 after 400 iterations, then divided by 10 at 32k and 48k iterations, and terminated at 64k iterations. After pruned, the network iterated 40 epochs at a constant learning rate of 0.001. For the CIFAR100 dataset, we did not divide the validation set, and the image preprocessing was consistent with [5].

TABLE I. DARKNET-53 (TINY) FOR CIFAR100

	Type	Filters	Size	Output
	Convolutional	32	3×3	32×32
	Convolutional	64	3×3	32×32
8×	Convolutional	32	1×1	
	Convolutional	64	3×3	
	Residual			32×32
	Convolutional	128	3×3/2	16×16
8×	Convolutional	64	1×1	
	Convolutional	128	3×3	
	Residual			16×16
	Convolutional	256	3×3/2	8×8
8×	Convolutional	128	1×1	
	Convolutional	256	3×3	
	Residual			8×8
	Average Pool		Global	
	Connected		100	
	Softmax			

As shown in Fig. 7(a), we give the average value of 5 experiments at α = 0: 0.1: 1, with the same proportions of each layer of the network pruned. It can be seen that under different α's, the accuracy retention effects of the network are different. Among them, under different pruning ratios, α achieves the optimal solution at 0.6, 0.8, and 0.9, as shown in Figure 7(b). According to the average and standard errors, compared with α = 1.0 (the pruning orders of $conv_0$ and $conv_{i,2}$ in the meta-structure are determined by $conv_0$) , the ability of the network to retain performance has significant advantages when the α gets the best accuracy.
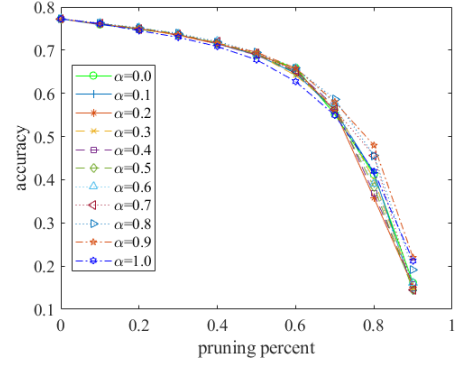
(a) Pruning filters at different α's
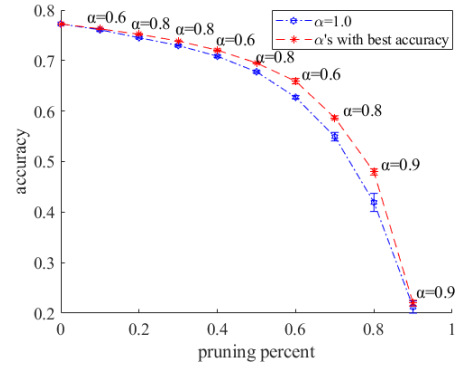
(b) Comparison of α's with best accuracy and α=1.0

Fig. 7. Darknet-53 (tiny) pruning analysis

B. Training and Pruning YOLOv3

The workpiece dataset created in this study consists of 10 types of workpieces, including screw bolt, screw nut, screwdriver, screw die, drill, wrench, L-type wrench, bearing, tap, and gear. We divided the 1488 workpiece images into three sets: 8391 for training, 420 for validation, and 1677 for test. Each image contains a maximum of 7 categories or 9 instances. Used the k-means algorithm to obtain 9 bounding boxes prior to the training as [(92,311), (117,116), (175,363), (191,191), (260,246), (279,286), (312,83), (359,154), (403,393)].

For both unpruned training and pruned retraining for YOLOv3, we set the mini-batch size to 4 and train 3 epochs with a constant learning rate of 0.0001 and a weight decay of 0.0015, and the remaining parameters were consistent with the model given in [3]. After the trained weights were loaded on the COCO dataset, YOLOv3 's $mAP^{IoU=0.5}$ on the workpiece test set reached 98.60%.

Since the optimal solutions for pruning appeared in $\alpha \in [0.6, 0.9]$ in the darknet-53(tiny) experiment, therefore we set $\alpha \in [0.6,1]$ with a step of 0.1 for pruning YOLOv3. The batch size and the number of samples were so small that the network was liable to fall into a local optimal solution, so the optimal solutions of 5 experiments were taken under different α's. This paper pruned each layer of YOLOv3 in the same proportions by 30%, 40%, and 50% respectively, and $mAP^{IoU=0.5}$ changed on the workpiece test set was shown in Fig. 8. It was obvious that YOLOv3 had the highest convergence accuracy when α = 0.8, which retained the performance of the network best. Fig. 9 showed the

$mAP^{IoU=0.5}$, parameter scale, and inference time(tested on Intel (R) Core (TM) i7-8750H CPU, NVIDIA GeForce GTX 1070 device) of the network at α = 0.8. As the pruning ratio increased, the scale of the network parameters gradually shrunk and could run with shorter inference time. In the case of pruning the convolution kernels of 30%, $mAP^{IoU=0.5}$ reached 99.38%, which was even higher than the unpruned model; while the parameter amount was reduced by 50.89%, and the inference time was shortened by 31.13%.
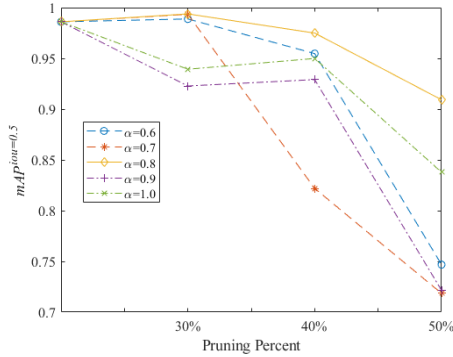

Fig. 8. YOLOv3's $mAP^{IoU=0.5}$ curves after pruned

Fig. 9. Unpruned and pruned data comparison at α = 0.8

C. Workpiece Contour and Gripping Points Analysis

First, according to the method described in Section III, we determined the cropping bias of each type of workpieces as [20,10,20,10,10,10,30,10,10,20](pixels). Based on Open Source Computer Vision Library (OpenCV) [29], we set the upper and lower thresholds of Canny as [50, 100], the k-size of the Gaussian blur filter as 5×5, and the dilate and erode of the morphological operation were iterated twice. In order to obtain the workpiece contour from the contour points set, we divide it into three processing situations according to the type of workpieces. For cylindrical workpieces such as gears, bearings and screw dies, using circumscribed circle of contour points set as workpiece contour directly; For the taps, screwdrivers and nuts, using the convex hull polygon curve as workpiece contour, which is a convex boundary that most tightly encloses contour points set; For wrenches, L-type wrenches, screw bolts and drills, approximating the polygon curve of contour points set with a specified accuracy of 0.008 times the perimeter of the contour points set, thus the smooth multi-point curve is transformed into a polygon. In order to analyze whether the workpiece contours after image processing keep the accuracy of bounding boxes from deep learning, we use mAP as the test index by the bounding rectangle of the workpiece contours and the labels, and choose two models: unpruned and pruned off 30% convolution kernels with α = 0.8 for workpiece detection. And then traditional image processing performs. The

changes in mAP before and after pruning are shown in Fig. 10.

The detection effect of the YOLOv3 was similar before and after pruning, and the pruned model was even slightly better than the unpruned at lower IoU nodes. After image processing, without losing the deep learning bounding boxes, part of the bounding boxes were corrected, and the bounding rectangles of the workpiece contours coincided with the labels higher, which proved the effectiveness of the method.
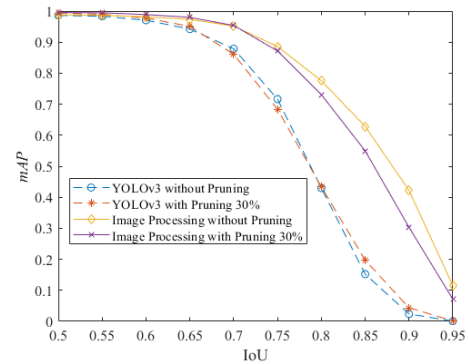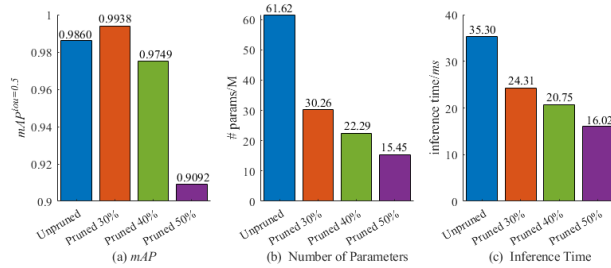

Fig. 10. mAP trends under different IoU's

V. CONCLUSION

This paper has proposed a workpiece detection method that combines deep learning with traditional image processing. According to the structural characteristics of YOLOv3, pruning filter strategy is improved, and experiments show that the improved strategy retains network performance more effectively. The original image is cropped according to the deep learning bounding boxes with cropping biases, and the multi-target detection problem is approximately transformed into a single-target one. A series of image processing algorithms are used to obtain the workpiece contours and appropriate gripping points. The experiment has proved that the image processing corrects the workpiece positioning information obtained by deep learning to a certain extent, and the degree of coincidence between the detected and the actual workpiece contours is higher.

REFERENCES

[1] X. Yao, J. Zhou, J. Zhang and C. R. Boër, "From Intelligent Manufacturing to Smart Manufacturing for Industry 4.0 Driven by Next Generation Artificial Intelligence and Further On," 2017 5th International Conference on Enterprise Systems (ES), Beijing, 2017, pp. 311-318.

[2] X. Zhou, J. Zhuo and P. Krahenbuhl, "Bottom-up Object Detection by Grouping Extreme and Center Points," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 850-859.

[3] J. Redmon, and A. Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.

[4] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf. Pruning filters for efficient convnets. arXiv preprint arXiv:1608.08710, 2016.

[5] K. He , X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2016, pp. 770-778.

[6] A. Krizhevsky, G. Hinton, "Learning Multiple Layers of Features from Tiny Images," Technical Report TR-2009, University of Toronto, Toronto, 2009.

[7] C. Huang, D. Chen and X. Tang, "Implementation of Workpiece Recognition and Location Based on Opencv," 2015 8th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, 2015, pp. 228-232.

[8] K. Xia and Z. Weng, "Workpieces sorting system based on industrial robot of machine vision," 2016 3rd International Conference on Systems and Informatics (ICSAI), Shanghai, 2016, pp. 422-426.

[9] W. Yan, F. Weiping, Z. Chensheng and W. Hongtao, "Image Matching for Workpiece Based on Genetic Algorithm," 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, 2009, pp. 152-157.

[10] D. Li, X. Zhang, T. Jiang and M. Zhang, "Application of improved wavelet threshold de-noising algorithm in work piece inspection," 2013 6th International Congress on Image and Signal Processing (CISP), Hangzhou, 2013, pp. 256-260.

[11] W. D. Yang, W. Guo , K. Peng and L. B. Liu, "Research on Removing Shadow in Workpiece Image Based on Homomorphic Filtering." Procedia Engineering, vol. 29, pp. 2360-2364, 2012.

[12] L. Yang, M. Chong, C. Bai and J. Li, "A multi-workpieces recognition algorithm based on shape-SVM learning model," Journal of Physics: Conference Series, vol. 1087, (2), pp. 022025, 2018.

[13] L. Yang, Y. Dong, J. Zhuang and J. Li, "A Recognition Algorithm for Workpieces Based on the Machine Learning," 2018 11th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 2018, pp. 371-375.

[14] S. Astanin , D. Antonelli , P. Chiabert and C. Alletto. "Reflective workpiece detection and localization for flexible robotic cells," Robotics and Computer-Integrated Manufacturing, vol. 44, pp. 190-198, 2017.

[15] K. Tang, F. Hu, W. Liu, Y. Deng, X. Wu and D. Luo, "Corner detection based real-time workpiece recognition for robot manipulation," 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau, 2017, pp. 2185-2190.

[16] C. Lu, H. Wang, C. Gu, K. Wu and X. Guan, "Viewpoint estimation for workpieces with deep transfer learning from cold to hot," International Conference on Neural Information Processing, Springer, Cham, 2018, pp. 21-32.

[17] M. U. Khalid, J. M. Hager, W. Kraus, M. F. Huber and M. Toussaint, "Deep Workpiece Region Segmentation for Bin Picking," 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), Vancouver, BC, Canada, 2019, pp. 1138-1144.

[18] C. G. Li and Y. Chang, "Automated visual positioning and precision placement of a workpiece using deep learning," The International Journal of Advanced Manufacturing Technology, vol. 104, (9), pp. 4527-4538, 2019.

[19] A. G. Howard, M. Zhu , B. Chen, and D. Kalenichenko. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv preprint arXiv: 1704.04861, 2017.

[20] J. Lei, X. Gao, J. Song, X. Wang and M. Song, "Survey of deep neural network model compression," Journal of Software, vol. 29, (2), pp. 251-266, 2018.

[21] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell. Rethinking the value of network pruning. arXiv preprint arXiv:1810.05270, 2018.

[22] M. Oquab, L. Bottou, I. Laptev and J. Sivic, "Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 1717-1724.

[23] J. Luo, J. Wu and W. Lin, "ThiNet: A Filter Level Pruning Method for Deep Neural Network Compression," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 5068-5076.

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, et al. Microsoft COCO: Common Objects in Context. arXiv preprint arXiv: 1405.0312, 2014.

[25] K. He, X. Zhang, S. Ren and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 1026-1034.

[26] Ioffe S, and Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.

[27] John Canny, "A computational approach to edge detection," IEEE Transactions on pattern analysis and machine intelligence, vol. 8, (6), pp. 679-698, Nov. 1986.

[28] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467, 2016.

[29] G. Bradski and A. Kaehler, "Learning OpenCV," OReilly Publications, 2008.