

A Conceptual Artificial Neural Network Model in Warehouse Receiving Management

Judy X Yang, Lily D Li, and Mohammad G. Rasul

Abstract—The purpose of this research is to explore a suitable Artificial Neural Network (ANN) method applying to warehouse receiving management. A conceptual ANN model is proposed to perform identification and counting of components. The proposed model consists of a standard image library, an ANN system to present objects for identification from the real-time images and to count the number of objects in the image. The authors adopted four basic mechanical design shapes as the attributes of images for shape analysis and pre-defined features; the joint probability from Bayes theorem and image pixel values for object counting is applied in this research. Compared to other ANNs, the proposed conceptual model is straightforward to perform classification and counting. The model is tested by employing a mini image dataset which is industrial enterprise relevant. The initial result shows that the proposed model has achieved an accuracy rate of 80% in classification and a 97% accuracy rate in counting. The development of the model is associated with a few challenges, including exploring algorithms to enhance the accuracy rate for component identification and testing the model in a larger dataset.

Index Terms—Classification, counting, contour, template matching, warehouse management.

I. INTRODUCTION

In today's highly competitive industrial environment characterised by high consumer's requirements for products with high quality, low-profit margins and short delivery times, the industry management team are forced to seek every opportunity to have their business processes at an optimisation level. However, information delays or data errors at the warehouse receiving stage have been problematic, which needs urgent management team attention [1]-[4].

The data at the receiving stage of the enterprise warehouse is the foundation of all supply chain management and company operations. The accuracy is considered one of the essential functions of manufacturing and business enterprises. The data often has a significant overall impact on the whole supply chain profit prediction. To resolve the inaccurate data, it depends on a suitable management model at the warehouse receiving stage [1]-[4].

Several Artificial Neural Network models can be learned to design a conceptual model application in industrial

warehouse management. The current state-of-the-art object detection system, Faster R-CNN hypothesises bounding boxes, resamples pixels or features for each box, and applies a high-quality classifier [5]. The datasets of the Faster R-CNN were all from PASCAL VOC, MS COCO and ILSVRC [1]-[3], [6], [7]. Although its accuracy is over 80%, it has been too computationally intensive for embedded systems, even with high-end hardware, and too slow for real-time application in warehouse management [1]-[3], [6]-[8].

This paper is to propose an Artificial Neural Network model, named the ANN-CIC model (Artificial Neural Network for Components Identification and Counting). The model can be applied to perform identification and counting of components at the warehouse receiving management stage. The OpenCV [1] API and the relevant functions were adopted to build the conceptual model to perform classification, identification and counting. The model has two features. The first one is that four basic geometric shapes are introduced as the attributes of the images. The second feature is building a template image library which has a similar function as a sliding window in the complicated CNN model. The image template library is used for classifying and identifying the objects in the inputting photos captured by the camera [1]-[3], [9].

With this minor modification, the proposed model can achieve a relatively higher accuracy, which meets the design goal.

The rest of the paper is organised as follows: Section II proposes the ANN-CIC model. Section III presents the experimental results. Section IV discusses the related work, and the final Section V concludes the paper and indicates the future work.

II. A CONCEPTUAL ANN-CIC MODEL FOR IDENTIFICATION AND COUNTING

A. Proposed Model Architecture

The model consists of an industrial camera and a workstation. The workstation includes four subsystems: an input images transforming system, a built-in template images library, an image feature extraction system, and an identification and counting system. The output of the ANN-CIC system can be interfaced with the enterprise management system.

Fig. 1 illustrates the proposed ANN-CIC model architecture.

B. Principle of the ANN-CIC Model

To improve the classification accuracy of objects, a standard image components database is introduced. The Bayes Classifier [10]-[12], a prevailing algorithm to provide

Manuscript received November 4, 2019; revised September 1, 2020.
The authors are with the School of Engineering and Technology, CQUUniversity, Rockhampton, QLD 4702, Australia (e-mail: j.yang@cqu.edu.au, l.li@cqu.edu.au, m.rasul@cqu.edu.au).

an objects classification function, is adopted to perform classification of images. The Bayes classifier analyses and compares a series of standard templates with the input image based on their basic shapes. It calculates the maximum probability of the input image. Then the classifier concludes the class of the input image, i.e., the category of the component in the image.

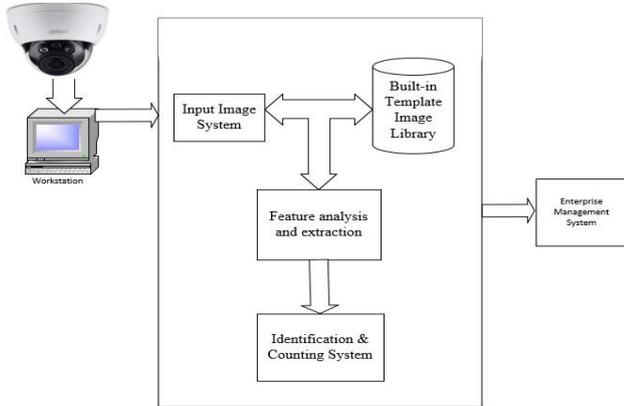


Fig. 1. Proposed ANN-CIC model.



Fig. 2. Basic shapes.

The contours of industry components usually consist of several basic shapes [13], e.g., rectangles, circles, triangles and polygons. In this research, these four primary attributes are defined to represent the components (refer to Fig. 2).

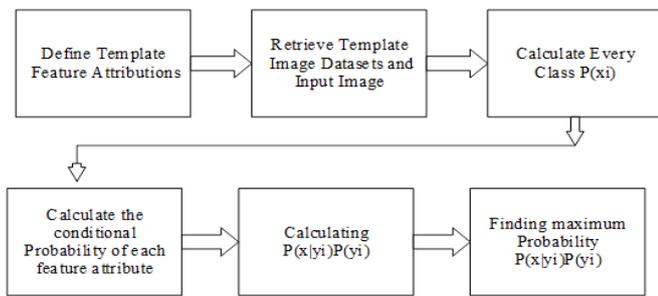


Fig. 3. Bayes theorem.

Fig. 3 illustrates the principle of the Bayes Classifier Process.

Depending on the number of basic shapes present in an image, any component can be represented by Equation (1).

$$X = \sum_{i=1}^4 x_i \quad (1)$$

where X represents the image component category, x illustrates the numbers of each attribute present, and i is from 1 to 4, x_1 is the number of rectangles, x_2 is the number of circles, x_3 is the number of triangles, x_4 is the number of polygons.

The probability of the input image and pre-template image in the library is calculated by using the Bayes theorem. Bayes theorem provides a way of calculating posterior probability $P(Y|X)$ from $P(Y)$ and $P(X|Y)$ where Y represents the test image.

The multiple variable Bayes equation is as below:

$$P(Y|X) = \frac{P(Y \cap X)}{P(X)} \quad (2)$$

$$P(Y \cap X) = P(X \cap Y) = P(X|Y)P(Y) \quad (3)$$

$P(Y|X)$ is the posterior probability of Y given X has occurred. $P(Y \cap X)$ is the probability of Y occurred, and X occurred. $P(X)$ is the probability of X .

The prior probability of each attribute $p(x_i)$ can be defined based on empirical and statistics.

Based on 20 stochastic industrial components, the prior probabilities of the four shapes $P(x_i)$ are calculated as below:

$$P(x_1=50) = 0.20$$

$$P(x_2=90) = 0.36$$

$$P(x_3=10) = 0.04$$

$$P(x_4=100) = 0.40$$

According to the inputted image shape analysis, the input image classification is calculated by using the Bayes theorem Equations (1), (2) and (3), which are adopted to calculate when the quantity of the four shapes is different from the prior probability. The joint probability calculation is as below:

$$P(Y = C_k | X) = P(Y = C_k | x_1, x_2, x_3, x_4) = \prod_{i=1}^4 P(Y = C_k | x_i) \quad (4)$$

where

$$C_k = \prod_{i=1}^4 x_i / \sum_{i=1}^4 X_i \quad (5)$$

Equation (5) is used for calculating the probability of the four shapes of the images, including both input images and template images.

Equation (6) is to calculate the joint probability variance between input images and the template images.

$$\text{Min}(A_j) = P(K_j) - P(T_j) \quad (6)$$

where $P(K_j)$ is defined as the input image's joint probability and $P(T_j)$ is defined as the template image's probability. The Coding procedure is as follows:

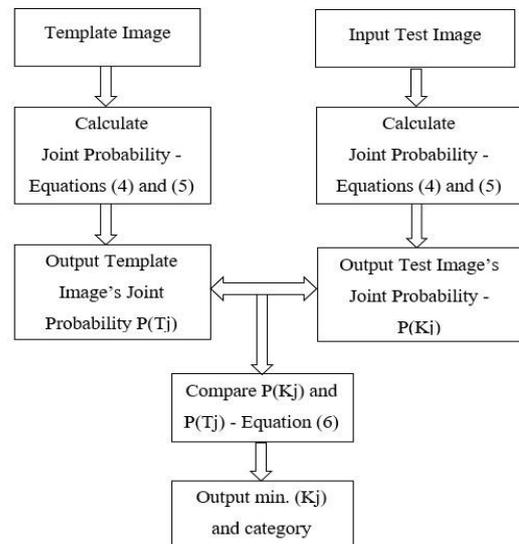


Fig. 4. Coding procedure.

If the output label from Equation (6) is the same as the input image label, the classifier is "True", and it is denoted 1;

if the output label is not the same, the classifier is “False”, and it is denoted as 0. The classifier prediction rate is defined as follows:

$$P = \text{Number of True} / (\text{Number of True} + \text{Number of False}) \quad (7)$$

After the classification system has judged the class of the input image, the subsequent counting system activates the below equations which undertake the pixel-based calculation:

$$Na = \frac{\text{Max.P} \cdot V}{\text{Max.pv}} \quad (8)$$

where Na is the actual number of objects.

$$Nb = \text{Round} \left(\frac{\text{Max.P}}{\text{Max.p}} \right) \quad (9)$$

where N_b is the rounded counting number of objects in the input image, $\text{Max. } P$ is the maximum white pixel value and $\text{Max. } p$ is the maximum white value of the template image.

$$Pc = (1 - \text{ABS}(Na - Nb) / N) * 100\% \quad (10)$$

where Pc is the counting accuracy rate, N is the actual existing number of the input image.

C. Modified ANN-CIC Model

The proposed ANN-CIC model is based on supervised machine learning algorithms[14]. According to mechanical design, most parts consist of basic shapes. Therefore, four basic shapes can be introduced to perform shape and contour analysis; then the process calculates the numbers of the four attributes (i.e., the shapes) in the images. It is similar to storing a series of different sized bounding boxes in advance. The pre-built template image library can be used as pre-defined criteria when the system performs identification and counting [15].

Five groups of images are used to illustrate the working process of the proposed model. The top line in Fig. 5 are the input images and the bottom line in Fig. 5 are the pre-template images in the library.

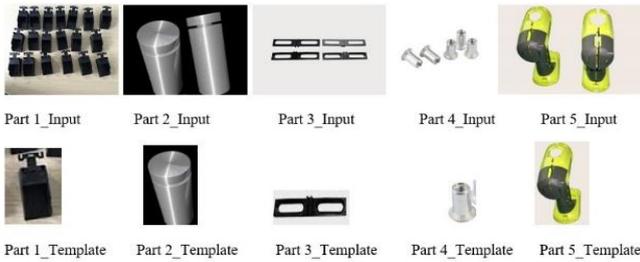


Fig. 5. Five groups of images (input & template).

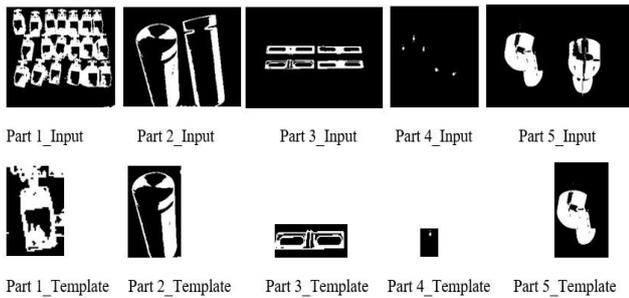


Fig. 6. Five groups of processed images (input & template).

The top line in Fig. 6 illustrates the processed input images and the bottom line in Fig. 6 are the processed template images. When an image is inputted, the system conducts

contour analysis based on the four basic shapes and compares with the built-in template image in the library. The image processing includes Grayscale, Binary, Gaussian Filter, Colour Threshold Value and Contour Feature Extraction.

The main programs for object detection and counting are described as below.

Identification and classification: The system performs contour analysis by using four shapes, compares the joint probability between the input image and the standard pre-template image library, assesses the comparison result, and generates the input image class as illustrated in Fig. 4. When the joint probability deviation is the minimum, the template image class is the input image’s target class. Then we reach the goal of classifying the category of the component [16], [17].

Objects counting: After finishing the component identification, the model calculates the number of objects in the input image. This counting system is based on contour pixel values. Firstly, the colour of the template image and the input image are converted to a grey colour. Secondly, the GaussianBlur algorithm [1], [12], is used for setting an appropriate brightness value between 0 and 255 to get the expected white and black component image. The three functions adopted from OpenCV API are as follow [1]-[3], [13], [18]-[20].

$$\text{Image_Gray} = \text{cv2.cvtColor}(\text{image-input}, \text{cv2.COLOR_BGR2GRAY}) \quad (11)$$

$$\text{Image_Blur} = \text{cv2.GaussianBlur}(\text{Image_Gray}, (5,5), 0) \quad (12)$$

$$\text{Image_Results} = \text{cv2.threshold}(\text{Image_Blur}, 0, 255, \text{cv2.THRESH_BINARY} + \text{cv2.THRESH_OTSU}) \quad (13)$$

where cv2 is an object instance of the OpenCV API [1], [12].

III. EXPERIMENT

Experiments and verification of the conceptual model were performed through five groups of input images and their corresponding template images.

A. The Conceptual Model System Required

This conceptual model verification executes at the standard desktop computer, Intel(R) Core, TM i5-8500 CPU @ 3.00GHz, Installed Memory: 16GB.

Software tool and application packages are: Python 3.7 [21]; OpenCV 4.0; Numpy; Pandas; Matplotlib; TensorFlow [22].

B. Classification Subsystem Verification

Five template images in the library and the corresponding five input images are selected to compare their joint probability.

Table I illustrates the inputted test images and template images’ joint probabilities, which is based on Fig. 4 coding process and Equations (5), (6) and (7). Fig. 7 shows the two kinds of raw image probabilities comparison. The horizontal axis represents the five groups parts, and the vertical axis represents the joint probabilities.

When verifying the classification algorithm, the image noise was not removed for the first time, and the accuracy is only 60%.

TABLE I: RAW IMAGES JOINT PROBABILITY COMPARISON

Raw Image Joint Probability		Input Image Joint Probability				
		K-1	K-2	K-3	K-4	K-5
Template Image Joint Probability		0.003257	0.001289	0.000166	0.001582	0.000364
		Cross ABS Value				
T-1	0.007792	0.004535	0.006503	0.007626	0.006210	0.007428
T-2	0.001204	0.002053	0.000085	0.001038	0.000378	0.000840
T-3	0.000237	0.003020	0.001052	0.000071	0.001345	0.000127
T-4	0.001286	0.001971	0.000003	0.001120	0.000296	0.000922
T-5	0.000392	0.002865	0.000898	0.000225	0.001191	0.000922
Output Min. Value		0.001971	0.000003	0.000071	0.000296	0.000028
Prediction Result		0	0	1	1	1
Prediction Rate		60%				

Remarks: T-Template Image, K-Input Image. The bold indicates that the detection results are correct.

TABLE II: PROCESSED IMAGES JOINT PROBABILITY COMPARISON

Processed Canny Image Joint Probability		Input Image Joint Probability				
		K-1	K-2	K-3	K-4	K-5
Template Image Joint Probability		0.000677	0.013148	0.001521	0.001659	0.000924
		Cross ABS Value				
T-1	0.000772	0.000094	0.012376	0.000749	0.006210	0.007428
T-2	0.015780	0.015103	0.002632	0.014260	0.000378	0.000840
T-3	0.001229	0.000552	0.011919	0.000291	0.001345	0.000127
T-4	0.001524	0.000847	0.011624	0.000003	0.000296	0.000922
T-5	0.000977	0.000299	0.012171	0.000544	0.001191	0.000922
Output Min. Value		0.000094	0.002632	0.000071	0.000296	0.000028
Prediction Result		1	1	0	1	1
Prediction Rate		80%				

Remarks: T-Template Image, K-Input Image. The bold indicates that the detection results are correct

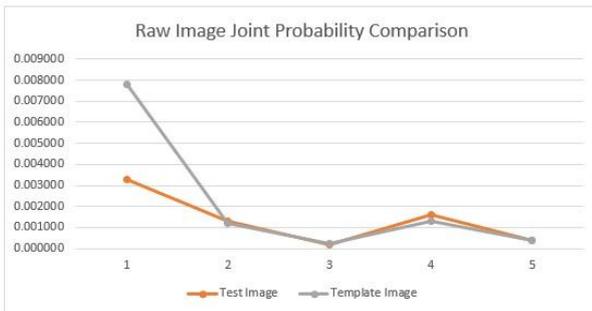


Fig. 7. Raw image's joint probability comparison chart.

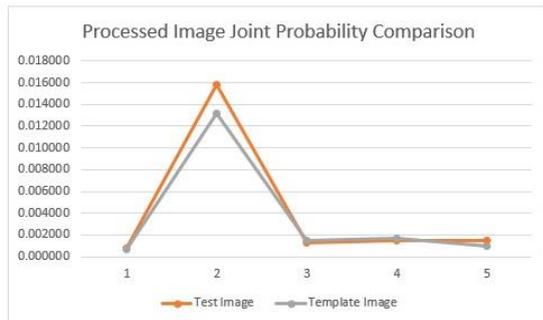


Fig. 8. Processed images joint probability comparison chart.

After the image was processed (Fig. 6), and the parameters are fine-tuned, the experiment data results are shown in Table II.

Fig. 8 shows the joint probability calculation results. The horizontal axis represents the five groups' parts, and the vertical axis represents the joint probabilities.

Based on Table II and Fig. 8, the prediction accuracy rate can reach 80%, which has demonstrated that the outcomes by using the processed images are better than a raw image test.

C. Counting Verification

The five group images in Fig. 5 and Fig. 6 are used to perform counting model verification. The top line of five

images are the test images to be identified (refer to Fig.5 and Fig. 6). The number of components in these images also need to be counted. The bottom five images are the corresponding templates in the standard image library. Two groups of images are selected to illustrate the object counting method by Python coding: they are Part 1_Input and Part 1_Template, Part 2_Input and Part 2_template shown in Fig. 6.

```
-----Hello Python-----
Processed Image Total White Hist Value: [12319.]
```

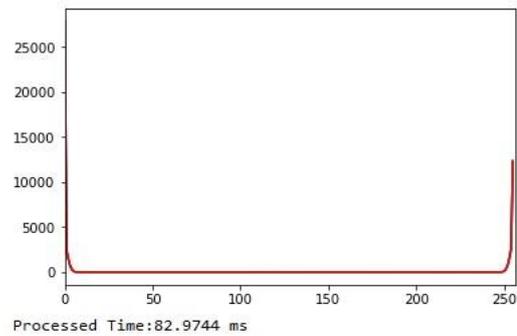


Fig. 9. Histogramy of Input image (part_1 input).

```
-----Hello Python-----
Processed Image Total White Hist Value: [696.]
```

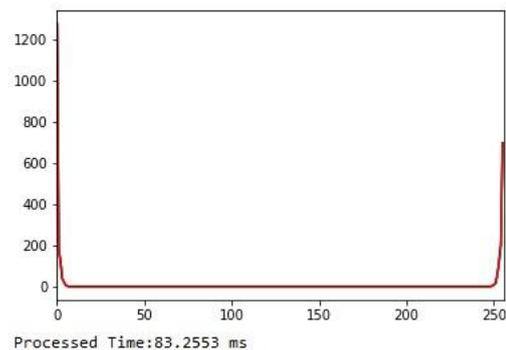


Fig. 10. Histogramy of template image (part_1_template).

Fig. 9 and Fig. 10 are histogram of Part 1_Input and Part 1_Template. Fig. 11 and Fig. 12 are histogram of Part 2_Input and Part 2_Template.

The next step is to calculate the number of components in the input images. According to the pixel values obtained for these images, the number of components can be calculated by using Equation (8) and Equation (9).

The accuracy is calculated by using Equation (10). Table III presents the experiment results. The experiment results demonstrate that the proposed ANN-CIC counting detector model can achieve an average accuracy of 96.8%.

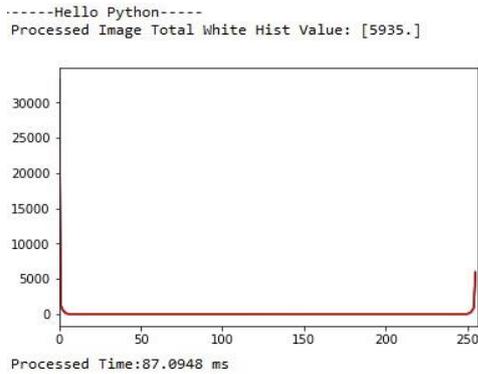


Fig. 11. Histogramy of input image (part 2_input).

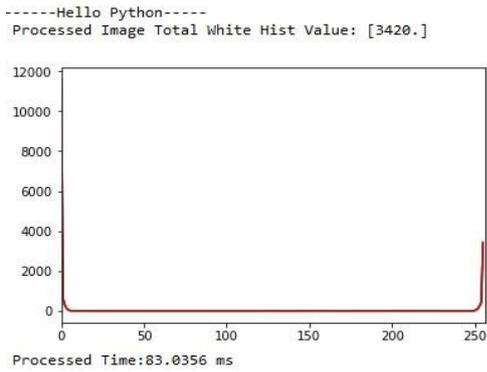


Fig. 12. Histogramy of template image (part 2_template).

TABLE III: FIVE SAMPLE COMPONENTS COUNTING ACCURACY

Name	Max. P/p (Pix.)	Actual No.	Calculate No.	Round No.	Processed Accuracy
T-1	696	1			
K-1	12319	19	17.7	18	84%
T-2	3420	1			
K-2	5935	2	1.74	2	100%
T-3	573	1			
K-3	2651	4	4.63	5	100%
T-4	23	1			
K-4	117	5	5.09	5	100%
T-5	13865	1			
K-5	25252	2	1.82	2	100%
Average Accuracy Rate					96.80%

Remarks: T-Template Image, K-Input Image

D. Morphological Transformation Verification

To verify the proposed model, a morphological image transformation is conducted. Three types of morphological change, namely Resize, Rotation and Warp, are used in the verification experiment. Fig. 13 shows the morphological images of Part 1. Fig. 14 presents the morphological images of Part 3.

Table IV shows the verification results for the morphological images.

According to Table IV, the accuracy rates for rotation and warp morphological images are 81.62% and 90.26% respectively, which are moderately acceptable. However, the model did not achieve a satisfactory accuracy for resized images. Therefore, for this conceptual model, it is suggested that the input images and the template images should take the same camera parameters to achieve better accuracy.

TABLE IV: TWO MORPHOLOGICAL IMAGES COUNTING ACCURACY

Image Name	Max Pixels	Actual No.	Calculation No.	Accuracy Rate
T-1	696	1		
K-1	12319	19	17.7	93.16%
T-1-Resize	9820	19	14.1	0%
T-1-Rotation	3918	17	5.62	88.24%
T-1-Warp	2525	12	3.63	83.33%
T-3-TM	573	1		
T-3_input	2651	4	4.63	90.5%
T-3-Resize	3759	4	6.56	36%
T-3-Rotation	1624	4	2.83	75%
T-3-Warp	1235	4	2.16	97.18%
Average Resize Accuracy Rate				0%
Average Rotation Accuracy Rate				81.62%
Average Warp Accuracy Rate				90.26%

Remarks: Remarks: T-Template Image, K-Input Image

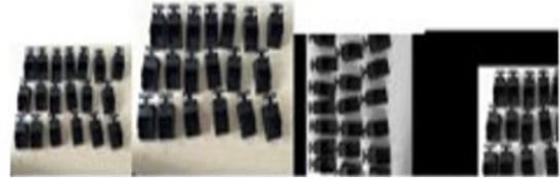


Fig. 13. Part 1 morphological images. (From L-R: Original, resized, rotated and warped)



Fig. 14. Part 3 morphological images. (From L-R: original, resized, rotated and warped)

IV. RELATED WORK

The current data collection at the warehouse receiving stage is still controlled manually in the industrial field. It is worthy for researchers to develop a new model to improve data collection.

There are several established methods for detection of multiple objects in images: OpenCV provides a series of API and mathematics to perform tracking and detecting objects in pictures, such as Template Matching and Shape Matching [1], [2], [14]-[18]. Harris Corner Detection [1], [2], [14]-[18], and Shi-Tomasi Corner Detector [1], [2], [14]-[18]. The state-of-the-art model Faster R-CNN [7] proposed MultiBox using sliding windows to perform tracking and detecting of objects. Moreover, SSD [3] is based on the Single box to perform tracking and detecting.

The Faster R-CNN approach, first introduced in MultiBox for learning objectives, has improved in both accuracy and execution time [7]. However, it needs a fine-tune at all layers from end to end for minimising the loss of both confidence and bounding box regression. Moreover, it requires thousands of images to train the model, which is very computationally expensive and time-consuming [1], [2], [7], [16], [17], [20],

[23], [24]. SSD [3] is based on Faster R-CNN but changed MultiBox into a single box to track and detect objects; the training and testing efficiency have much improvement.

SSD adopted a single box and based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes. The core of the SSD approach is predicting category scores and box offsets for a fixed set of default bounding boxes [3]. SSD single box provided a foundation of the proposed ANN-CIC model for the building of a classifier and counting detector. The pre-built template image library of the ANN-CIC model is similar to the default bounding box of the SSD.

The other sets of methods, namely Template Matching, Shape Matching, Contour Extraction, and Corner Detection, are also related to the model proposed in this paper [1]-[3], [7], [14]-[17]. That is, the image feature extraction process is involved. There are a series of algorithms in OpenCV API that can be used to extract, describe and match the features. The ANN-CIC classification method falls in this category because it proposed to build a standard image library as the default box to match and compare it with the input image. However, our approach is focusing on shape, contour and pixel bins analysis to reach classification and counting of objects.

V. CONCLUSION AND RECOMMENDATIONS

This paper proposed a model for the classification and counting of components for warehouse management at the receiving stage. A vital feature of the model is introducing four basic shapes and building the pre-built image library. The architecture allows us to perform straightforward input image feature extraction effectively and efficiently. Although it is a laboratory model, it demonstrated a high counting accuracy rate and is worthy of further exploration by using a larger industrial image dataset.

The proposed ANN-CIC conceptual classification and counting model has achieved an 80% accuracy rate in classification and a 97% accuracy rate in counting. The average accuracy rate is 78%, which outperformed SSD with an accuracy of 75.1% mAP.

Regarding this warehouse model, a standard desk PC can perform the test and satisfy the real-time requirement. It is believed that this model can be used as a block of the larger warehouse management systems. This model is still associated with a few challenges. It is necessary to study suitable algorithms to improve the accuracy rate of object classification and counting. Another challenge is to detect, classify and count the objects in the images that are occluded by other objects. Furthermore, the model needs to be trained and verified by a larger industry-related image dataset.

The contributions of this study can be summarised as below:

- Four basic shapes are introduced to perform shape analysis.
- A standard template image library is proposed to replace the bounding box, which has improved both the training time and the detection accuracy of components.
- The classification is predicted by comparing the correlation coefficient between template parts in the image library and input images.

- The counting detector is implemented by the colour pixel value calculation of the images.

CONFLICT OF INTEREST

The authors declare that there are no conflicts of interest regarding the publication of this paper.

AUTHOR CONTRIBUTIONS

Judy X Yang conducted the research, analysed the data and coded the program, and wrote the first version of the paper. Dr Li and Professor Rasul guided analysis, revised the article, and approved the final version.

REFERENCES

- [1] J. Minichino, *Learning OpenCV 3 Computer Vision with Python*, 2nd ed. Birmingham: Packt Publishing, 2015.
- [2] X.-Y. Gong, H. Su, D. Xu, Z.-T. Zhang, F. Shen, and H.-B. Yang, "An overview of contour detection approaches," *International Journal of Automation and Computing*, vol. 15, no. 6, p. 656, 2018.
- [3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer Vision-ECCV, Lecture Notes in Computer Science*, B. Leibe et al., Eds., vol. 9905, pp. 21-37, 2016.
- [4] B. Emami-Mehrgani, W. P. Neumann, S. Nadeau, and M. Bazrafshan, "Considering human error in optimizing production and corrective and preventive maintenance policies for manufacturing systems," *Applied Mathematical Modelling*, vol. 40, no. 3, pp. 2056-2074, 2016.
- [5] A. Ghazvini, S. N. H. S. Abdullah, and M. Ayob, "A Recent trend in individual counting approach using deep network," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 5, no. 5, p. 7, 2019.
- [6] G. Ross, "Fast R-CNN," in *Proc. 2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440-1448.
- [7] R. Shaoqing, H. Kaiming, R. Girshick, and S. Jian, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2017.
- [8] J. Dou, C. Chen, and P. Yang, "Genetic scheduling and reinforcement learning in multirobot systems for intelligent warehouses," *Mathematical Problems in Engineering*, pp. 1-10, 2015.
- [9] F. Lecellier, S. Jehan-Besson, and J. Fadili, "Statistical region-based active contours for segmentation: An overview," *IRBM*, vol. 35, no. 1, pp. 3-10, 2014.
- [10] T. C. Henderson and N. Boonsirisumpun, "Issues related to parameter estimation in model accuracy assessment," *Procedia Computer Science*, vol. 18, no. C, pp. 1969-1978, 2013.
- [11] M. Martinez-Arroyo and L. E. Sucar, *Learning an Optimal Naive Bayes Classifier*, vol. 4, pp. 958-958, 2006.
- [12] Z. Yang and P. I. Rockett, "The Bayesian operating point of the canny edge detector," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3409-3416, 2006.
- [13] U. Grenander, "Geometrics of knowledge," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 94, no. 3, p. 783, 1997.
- [14] Patent issued for computer vision-based object tracking system (USPTO 9964624), *Journal of Engineering*, p. 3327, 2018.
- [15] P. Hidayatullah and M. Zuhdi, "Color-texture based object tracking using HSV color space and local binary pattern," *International Journal on Electrical Engineering and Informatics*, vol. 7, no. 2, pp. 161-174, 2015.
- [16] J. Yasmin and M. Sathik, "An improved iterative segmentation algorithm using canny edge detector for skin lesion border detection," *International Arab Journal of Information Technology*, vol. 12, no. 4, pp. 325-332, 2015.
- [17] X. Lu and K. Nishiyama, "Low-resolution colour-based visual tracking with state-space model identification," *Computer Vision and Image Understanding*, vol. 114, no. 9, pp. 1045-1054, 2010.
- [18] Industrial components. [Online]. Available: <https://www.google.com/imghp?hl=en>
- [19] T. Islam, M. A. Rico-Ramirez, D. Han, and P. K. Srivastava, "Artificial intelligence techniques for clutter identification with polarimetric radar signatures," *Atmospheric Research*, vol. 109-110, pp. 95-113, 2012.
- [20] M. B. Steven, *Inventory Best Practices*, 2nd ed. Hoboken: John Wiley & Sons, 2011.

- [21] B. I. Den, Python Deep Learning Cookbook: Over 75 Practical Recipes on Neural Network Modeling, Reinforcement Learning, and Transfer Learning Using Python, 2017.
- [22] A. Géron, "Hands-on machine learning with scikit-learn and tensorflow," *O'Reilly Media*, 2013.
- [23] OpenCV-Python Tutorial. [Online]. Available: <https://opencv-python-tutorials.readthedocs.io/>
- [24] R. Medina-Carnice, S. Muñoz, B. Yeguas, and M. Diaz, "A novel method to look for the hysteresis thresholds for the canny edge detector," *Pattern Recognition*, vol. 44, no. 6, pp. 1201-211, 2011.



Lily Li received a B Eng. from Xi'an Jiaotong University (China), a MIT from the University of Newcastle (Australia), a PhD and a Grad. Cert. in Tertiary Education from CQUniversity (Australia). Dr Li has been a full-time academic at CQUniversity since 2002. Her experiences consist of course coordination, teaching and learning management, curriculum design and research. Dr Li's research interests include computational intelligence, evolutionary optimisation and software engineering. She is a member of the Australia Computer Society.

Copyright © 2020 by the authors. This is an open-access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



Judy X. Yang received her bachelor of agricultural machinery degree from Henan Agricultural University (China). She studies Master of Informatics by research at CQUniversity (Australia) since June 2019. Her research focuses on machine learning application in supply chain management. She has worked in manufacturing engineering and supply chain management fields over 18 years. Her work history includes in an America manufacturing company Penn engineering in Shanghai as a project mechanical engineer and MIS electrical company as an OEM project engineer. She is a member of the Engineers Australia since 2015.



Mohammad Rasul has obtained his PhD from The University of Queensland (Australia). Currently, he is a professor of mechanical engineering at the School of Engineering and Technology, Central Queensland University, Australia. He is specialised and experienced in research and teaching in clean energy and thermodynamics. He has made significant contributions in research with over 450 publications, 4700 citations in Scopus and \$3.4 million research grants. He has supervised 30 PhD/master of engineering research students to completion. He is involved with professional communities through his varied roles such as membership of engineers Australia, various conference committees, editorship and grant assessors.