Received May 13, 2020, accepted June 15, 2020, date of publication June 22, 2020, date of current version July 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3003917

Syncretic-NMS: A Merging Non-Maximum Suppression Algorithm for Instance Segmentation

JUN CHU[1,2], YIQING ZHANG[1,2], SHAOMING LI[1,2], LU LENG[1,2,3], (Member, IEEE), AND JUN MIAO[1,4]

[1]Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, Nanchang Hangkong University, Nanchang 330063, China
[2]School of Software, Nanchang Hangkong University, Nanchang 330063, China
[3]School of Electrical and Electronic Engineering, College of Engineering, Yonsei University, Seoul 120749, South Korea
[4]School of Aeronautical Manufacturing Engineering, Nanchang Hangkong University, Nanchang 330063, China

Corresponding author: Lu Leng (leng@nchu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61663031, Grant 61866028, and Grant 61661036, in part by the Key Program Project of Research and Development (Jiangxi Provincial Department of Science and Technology) under Grant 20171ACE50024 and Grant 20192BBE50073, in part by the Foundation of China Scholarship Council under Grant CSC201908360075, and in part by the Open Foundation of Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition under Grant ET201680245 and Grant TX201604002.

ABSTRACT Instance segmentation is typically based on an object detection framework. Semantic segmentation is conducted on the bounding boxes that are returned by detectors. NMS (non-maximum suppression) is a common post-processing operation in instance segmentation and object detection tasks. It is typically used after bounding box regression to eliminate redundant bounding boxes. The evaluation criteria for object detection require that the bounding box be as close as possible to the ground truth, but they do not emphasize the integrity of the included object. However, sometimes the bounding boxes cannot contain the complete objects, and the parts beyond the bounding boxes cannot be correctly predicted in the subsequent semantic segmentation. To solve this problem, we propose the Syncretic-NMS algorithm. The algorithm takes traditional NMS as the first step and processes the bounding boxes obtained by traditional NMS, judges the neighboring bounding boxes of each bounding box, and combines the neighboring boxes that are strongly correlated with the corresponding bounding boxes. The coordinates of the merged box are the four coordinate extremes of the bounding box and the highly relevant neighboring box. The neighboring box with strong correlation is merged with the corresponding bounding box. Based on an analysis of the influences of corresponding factors, the criteria for correlation judgment are specified. Experimental results on the MS COCO dataset demonstrate that Syncretic-NMS can steadily increase the accuracy of instance segmentation, while experimental results on the Cityscapes dataset prove that the algorithm can adapt to application scenario changes. The computational complexity of Syncretic-NMS is the same as that of traditional NMS. Syncretic-NMS is easy to implement, requires no additional training, and can be easily integrated into the available instance segmentation framework.

INDEX TERMS Instance segmentation, non-maximum suppression, correlation judgment, object localization, object detection.

I. INTRODUCTION

Instance segmentation is a multi-mission learning task that consists of object detection and semantic segmentation. In the task, an algorithm generates bounding boxes for specified object categories in images and assigns classification scores to them. Then, the algorithm classifies the foreground objects in the bounding boxes at the pixel level [1]. A popular class of instance segmentation algorithms is based on object detection frameworks, such as Faster R-CNN (faster region-convolutional neural network) [2] and Cascade R-CNN (cascade region-convolutional neural network) [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Shiqi Wang.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

In such algorithms, NMS (non-maximum suppression) is a common post-processing operation. Its main objective is to eliminate redundant bounding boxes that are generated during the detection process, thereby substantially reducing the false detection rate.

Traditional NMS is the widely used post-processing algorithm in object detection, but it has some shortcomings in instance segmentation. For the localization task in object detection, the predicted bounding boxes must be as close as possible to the labeled ground truth; however, the integrity of the detected objects was not fully considered. When the complexity of image structure is normal, although the bounding boxes slightly larger than the ground truth has little effect on the predictive results (as shown in Figure 1, larger bounding boxes does not bring worse predicting result), those bounding boxes smaller than the true boundary boxes cannot fully contain the objects and will lead to serious problems. Instance segmentation requires semantic segmentation operations after localization. Objects that are not contained in bounding boxes cannot participate in the segmentation process, thereby resulting in a decrease in the segmentation accuracy. In addition, just as our brain uses the association between objects and the environment to promote visual perception and cognition [4], a moderate amount of contextual information helps increase the accuracy of predictions [5]. Therefore, expanding the network's receiving range or the size of the candidate regions can enhance the segmentation accuracy of deep learning networks [6]–[8].

FIGURE 1. When the complexity of the image structure is normal, the segmentation results of the smaller bounding box on the left and the larger bounding box on the right are almost the same.

Based on the above analysis, we propose Syncretic-NMS, which is a merging non-maximum suppression algorithm for instance segmentation that is based on traditional NMS. The algorithm judges the neighboring bounding boxes of proposed boxes, merges the bounding boxes that are strongly correlated to the proposed boxes, and generates bounding boxes that contain the complete objects.

The main innovations of the algorithm in this paper are as follows:

1) A general NMS algorithm is proposed, which can replace all traditional NMS algorithm modules that are based on greedy algorithms in instance segmentation. Its algorithm complexity is consistent with traditional NMS, and it needs not be added to the training phase; thus, it is very easy to implement.

2) Compared with traditional NMS, Syncretic-NMS merges neighboring bounding boxes such that the bounding boxes contain complete objects, which can provide more context information during segmentation and improve the segmentation accuracy.

3) Using a threshold self-test procedure, Syncretic-NMS can adapt to various application scenarios.

The remainder of this paper is organized as follows: Related works are reviewed in Section II. Section III introduces traditional NMS. Section IV introduces our method, including the details of the pipeline, the correlation judgment factors and criteria, and the threshold selection methods, and an algorithm complexity analysis is conducted. Section V introduces the experiments and analyzes the results. We present the conclusions of this study in Section VI.

II. RELATED WORKS

NMS is an important part of the detection algorithm. It was first used for edge detection [9] and, subsequently, for feature point detection [10], [11] and object detection [12]–[14]. Early NMS of object detection was not always an integrated component in the pipeline [15]. In the subsequent development, NMS was gradually integrated and differentiated into the following three methods: greedy NMS, bounding box aggregation and learning NMS.

Greedy NMS is a traditional and the most popular NMS method in object detection. The strategy of this method is simple and intuitive: For a set of overlapping bounding boxes, the bounding box with the maximum score is selected, and the neighboring bounding boxes are deleted according to specified rules, e.g., if they exceed the manually set IoU (intersection over union) threshold. In recent years, related algorithms have been improved on this basis. Soft-NMS [16] reduces the scores of (rather than directly deleting) neighboring bounding boxes that exceed the IoU threshold and improves the robustness to object occlusions. References [17] and [18] use the IoUs of the bounding boxes that are predicted by the network and the ground truth as the localization reliability parameter of the bounding boxes. They replace the classification score with the localization reliability parameter as the input of the NMS; thus, the bounding boxes do not deviate from the object during the iterative regression process. Weighing [19], [20] and fusion [21]–[23] can be used in NMS to further improve performance. In a recent study, Softer-NMS [24] was proposed, which uses a new loss function to train the bounding box regression model. After obtaining the standard deviation of the predicted localization, the bounding boxes are fused using the average weights. Fast-NMS [25] realizes speedup of the batch sorting algorithm and the IoU calculation and uses matrix operations and thresholds to identify the detection results that must be retained for each class. Greedy NMS remains the best choice [26], but this type of method has the disadvantages of requiring manual setting of the threshold and of yielding only a locally optimal solution.

Bounding box aggregation is another method for suppressing redundant bounding boxes. The core strategy of this type of method is the combination or clustering of bounding boxes, rather than using a greedy algorithm to obtain a locally optimal solution. For a series of bounding boxes that are returned by a classifier, the algorithm typically groups these bounding boxes according to specified rules. Then, for each set of bounding boxes, all the bounding boxes that satisfy the requirements are aggregated into a single bounding box. Representative methods of this type are the VJ detector (Viola-Jones face detector) [27] and Overfeat [28]. When processing many bounding boxes, the VJ detector cascades multiple classifiers, arranges the classifiers from front to back in order of increasing complexity, and aggregates the bounding boxes. Overfeat judges the correlation between the bounding boxes that are returned by the classifier. If the IoU between two bounding boxes is higher and the distance between their centers is shorter, the probability that they are regarded as the identical object is higher. Then, the average of the four coordinate extreme values of all relevant bounding boxes is calculated and returned to realize the aggregation of the bounding boxes. In addition, [13] and [29] also present effective methods for bounding box aggregation. However, methods of this type also require manual setting of the threshold and are relatively time-consuming.

Learning NMS is novel. The main strategy of methods of this type is to add NMS to the neural network in an end-to-end manner and to score and filter all the original detection results. The representative method is Learning-NMS [30]. For the network to obtain only one corresponding bounding box for each object, the method proposes a loss function. If the detector generates two or more bounding boxes for an object during training, it will be punished. Neighboring bounding boxes conduct joint processing so that the detector has sufficient information to determine whether a single object has been detected multiple times. References [31]–[33] also integrate NMS into deep learning networks. Compared with greedy NMS, this method performs better when dealing with occlusion and on dense detection problems, but it is outperformed overall by the latest greedy NMS algorithm. Although learning NMS does not require manual setting of the threshold, it is time-consuming.

III. TRADITIONAL GREEDY NMS

Traditional NMS is a greedy algorithm. The main characteristic of this algorithm is that it can only obtain a locally optimal solution and not a globally optimal solution. To fully explain the pipeline of the traditional NMS algorithm, the relevant definitions are presented here.

The classifier extracts several bounding boxes from the images and passes the first n bounding boxes with higher scores to the NMS algorithm. Define the bounding box list B as a tensor, which is expressed as $B = b_1, b_2, \ldots, b_n$. The classification score list S that is returned by the classifier is a one-dimensional array, which expressed as $S = s_1, s_2, \ldots, s_n$ and corresponds to the bounding box information in the bounding box list B element by element.

To facilitate description of the algorithm, a bounding box in the bounding box list is set to $b_i (i = 1, 2, \ldots, n)$, and its corresponding classification score is denoted as $s_i (i = 1, 2, \ldots, n)$. For this bounding box, assume that its area is $area(b_i)$. If $b_j (j = 1, 2, \ldots, n, j \neq i)$ is a neighboring box of b_i and the area of b_j is $area(b_j)$, their IoU can be expressed as:

$$iou\left(b_i, b_j\right) = \frac{area\left(b_i \cap b_j\right)}{area\left(b_i \cup b_j\right)} \tag{1}$$

According to the above definition, the traditional non-maximum suppression algorithm process is as follows:

Algorithm 1 Pipeline of Traditional Non-Maximum Suppression

Input: The bounding box list $B = b_1, b_2, \ldots, b_n$ of the top n bounding boxes with high scores that are returned by the classifier, the score list $S = s_1, s_2, \ldots, s_n$, and the threshold N_t.

Output: Bounding box list D and corresponding score list S.

1) Develop tensor D as storage space and save the bounding box coordinate information;
2) WHILE $B \neq$ empty
3) Select the maximum s_M in S;
4) Add the corresponding bounding box b_M to D and delete it from B;
5) FOR b_i IN B:
6) IF $iou\left(b_M, b_i\right) \geq N_t$
7) Delete the corresponding information of b_i in B and S;
8) END IF
9) END FOR
10) END WHILE

According to the above algorithm pipeline, there is only one suppression condition for the bounding box. If $iou\left(b_M, b_i\right)$ exceeds the threshold N_t, the bounding box will be suppressed. Due to the conciseness of the judgment conditions, traditional NMS has extremely high efficiency, but some high-confidence bounding boxes may also be filtered out by mistake, thereby resulting in the obtained bounding boxes not including complete objects. Aiming at overcoming this problem, this paper proposes a new NMS algorithm that merges neighboring boxes: Syncretic-NMS.

IV. SYNCRETIC-NMS

A. SYNCRETIC-NMS PIPELINE

Similar to the traditional NMS algorithm, the Syncretic-NMS algorithm that is proposed in this paper accepts the bounding box list B and classification score S that are returned by the classifier as input, obtains the bounding box list D after one round of NMS, and conducts neighboring box correlation

judgment and merge operations. However, the coordinate information of the bounding box is added to the bounding box list B, which is used to obtain the combined border coordinates. A plane Cartesian coordinate system is established, where the positive directions of x-axis and y-axis are horizontally rightward and vertically upward, respectively. Consider a bounding box $b_i = \{i, (x_1^i, y_1^i, x_2^i, y_2^i)\}$ in B, where i is the label of each bounding box, which associates the bounding box with its corresponding classification score, and the quadruple $(x_1^i, y_1^i, x_2^i, y_2^i)$ represents the coordinates of each bounding box, where (x_1^i, y_1^i) and (x_2^i, y_2^i) represents the coordinates of upper-left corner and the lower-right corner of the bounding box, respectively.

The correlation judgment operation is performed on all adjacent boxes of the bounding box in the list in order. If the degree of correlation between the neighboring box and the candidate box exceeds a threshold N_c, it is added to the temporary bounding box storage list A that contains the candidate box. For all bounding boxes in A, find the minimum x_1^i, the maximum y_1^i, the maximum x_2^i, and the minimum y_2^i coordinate values among all the coordinates, and replace the four coordinates of the bounding box accordingly. Finally, repeat this for all boxes in D. After this operation, the area of the bounding box is enlarged, and other high-confidence bounding boxes are merged. By manually controlling the threshold N_c, the degree of enlargement of the bounding box area can be adjusted. The proposed algorithm expands the range of the bounding box by retaining and merging adjacent bounding boxes that are strongly related to the candidate bounding box, and it can completely contain the boundary of the object.

The algorithm pipeline is as follows:

Before conducting Step 15, to transform the bounding box coordinate information into a modifiable state, it is necessary to delete the constant mark ''const'' of the input bounding box list and to modify the corresponding head file.

Algorithm 2 Pipeline of Syncretic Non-Maximum Suppression

Input: The bounding box list $B = b_1, b_2, \ldots, b_n$ of the top n bounding boxes with high scores that were returned by the classifier (the bounding boxes in the list $b_i = \{i, (x_1^i, y_1^i, x_2^i, y_2^i)\}$), the score list $S = s_1, s_2, \ldots, s_n$, the threshold N_t, and the correlation judgment threshold N_c.

Output: Bounding box list D and the corresponding score list S.

1) Develop tensors D and A as storage spaces for saving the bounding box coordinate information for return and for correlation judgment, respectively;
2) WHILE $B \neq$ empty
3) Pick the maximum s_M in S;
4) Add the corresponding bounding box b_M to D and delete it from B;
5) FOR b_i IN B:
6) IF $iou\,(b_M, b_i) \geq N_t$
7) IF $iou\,(b_M, b_i) * s_i \geq N_c$
8) Add b_M and bounding box b_i to A;
9) END IF
10) Delete the corresponding information of b_i from B and S;
11) END IF
12) FOR a_i IN A:
13) Select the minimum x_1 coordinate, the maximum y_1 coordinate, the maximum x_2 coordinate and the minimum y_2 coordinate of all the bounding box coordinates of list A;
14) END FOR
15) Replace the coordinates of b_M in D with the four coordinates;
16) END FOR
17) END WHILE

B. CORRELATION JUDGEMENT FACTORS AND CRITERIA

Correlation judgment is a key mechanism for controlling whether neighboring boxes are retained. Neighboring boxes that pass the judgment will participate in the merging of bounding boxes. The bounding box classification score reflects the probability that the objects in the bounding box belong to a specified category. The higher the score, the higher the localization accuracy of the bounding box. The IoU between the bounding boxes reflects the degree of correlation between the bounding boxes. The closer the two boxes are, the higher the IoU between the bounding boxes. Therefore, the bounding box classification score and the IoU of the bounding boxes are positively correlated with the correlation between the bounding boxes. In Overfeat [23], the distance between bounding boxes is also an important factor for judging the correlation of bounding boxes. Under comprehensive consideration, we use the classification score, IoU, and adjacency as the factors for determining the correlation of neighboring boxes. The experimental results and analysis in Section V are used to evaluate a single model under various combinations of factors. The results demonstrate that the optimized result can be obtained by using the product of the classification score and IoU as the correlation judgment criterion.

The classification score and IoU have been introduced in the previous section, and their values range in [0,1]. The adjacency between the bounding boxes is defined as the Euclidean distance between the center point of b_M with the maximum classification score in the current bounding box list and the center point of the neighboring box. Let the coordinates of the upper-left corner and the lower-right corner of a bounding box b_i be (x_1^i, y_1^i) and (x_2^i, y_2^i), respectively, and the corresponding coordinates of the upper-left corner and the lower-right corner be (x_1^j, y_1^j) and (x_2^j, y_2^j), respectively. Then, the coordinates of the center point P_i of the bounding box b_i and the center point P_j of the neighboring box

are $(\frac{x_1^i+x_2^i}{2}, \frac{y_1^i+y_2^i}{2})$ and $(\frac{x_1^j+x_2^j}{2}, \frac{y_1^j+y_2^j}{2})$, respectively, and the Euclidean distance ρ between the two points is:

$$\rho = \sqrt{\left(\frac{x_1^j + x_2^j}{2} - \frac{x_1^i + x_2^i}{2}\right)^2 + \left(\frac{y_1^j + y_2^j}{2} - \frac{y_1^i + y_2^i}{2}\right)^2} \tag{2}$$

The adjacency should be normalized. The closer the two boxes are, the more likely they are to be correlated. The further away from the corresponding bounding box and the closer to the image boundary, the less likely a box is to be correlated to the corresponding bounding box. Based on the above reasons, we normalize the adjacency *adjc* between the bounding boxes to:

$$adjc = 1 - \frac{\rho}{\rho_0} \tag{3}$$

Here, ρ is the distance between the center points of the adjacent bounding boxes, and ρ_0 is the distance from the extension of P_iP_j to the image boundary. After P_iP_j is extended, it crosses the boundary with Q_j, and after P_jP_i is extended, it crosses the boundary with Q_i.

$$\rho_0 = \min\left(Q_iP_j, P_iQ_j\right) \tag{4}$$

If the center points of the two boxes are the same, *adjc* is 1. If the center point of the adjacent box is at the boundary of the image, *adjc* is 0. The range of *adjc* is [0,1]. After normalizing all features, the product of these features is also limited to [0,1], and all factors are positively related to the correlation between the bounding boxes. The algorithm synthesizes and integrates the correlation factors and conducts the correlation judgment.

The correlation judgment function is:

$$s_i = \begin{cases} s_i, & iou\left(b_i, b_j\right) * s_i \geq N_c \\ 0, & iou\left(b_i, b_j\right) * s_i < N_c \end{cases} \tag{5}$$

Here, s_i is the original classification score of the bounding box, $iou\left(b_i, b_j\right)$ is the IoU of the bounding box b_i and the neighboring box b_j, and N_c is the association judgment threshold.

C. THRESHOLD SELECTION

The traditional NMS algorithm regards a manually set threshold N_t as a constant, and any bounding box that is below the threshold N_t will be suppressed. The algorithm conducts an NMS operation, and the threshold N_t is also manually set. In the experiment, N_t was set to a constant value of 0.5. Similar to the traditional NMS algorithm, N_c is also a constant threshold that is set manually, and its value range is [0,1]. If N_c is too high, then Syncretic-NMS is equivalent to traditional NMS and will not merge any neighboring bounding boxes. However, if N_c is too low, too many neighboring bounding boxes will remain, which will also substantially affect the accuracy. In addition, for application scenarios, to obtain the optimal results, it is necessary to adjust the threshold N_c. Therefore, we design a method for optimizing the threshold automatically. The detailed data and analysis are presented in Section V.

D. ALGORITHM COMPLEXITY ANALYSIS

The time complexity of each step in Syncretic-NMS is $\mathcal{O}(n)$, where n is the number of bounding boxes. For Syncretic-NMS to process n bounding boxes, the computational time complexity is $\mathcal{O}(n^2)$, which is the same as the complexity of the traditional greedy NMS and that of the classic improved algorithm, namely, Soft-NMS. Syncretic-NMS adds additional traversal operations, although it will slightly affect the calculation speed, but the step will not increase the calculation complexity. It will not significantly affect the running speed of the detector that is applied to each detection network, and it can be easily added to the instance segmentation algorithm pipeline. Quantitative data of time cost are shown in Section V.

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. CORRELATION FACTOR ABLATION STUDY

Correlation judgment is a key mechanism for controlling whether adjacent boxes are retained. Neighboring boxes that pass the correlation judgment will participate in the merging of bounding boxes. To determine which form of the correlation judgment performs best, we design an ablation study among the correlation factors. The experiments are conducted on a classic instance segmentation framework: Mask R-CNN (mask region-convolutional neural network) [34]. The first stage of Mask R-CNN uses Faster R-CNN for bounding box regression and classification, and the second stage conducts semantic segmentation of the returned bounding boxes. The experiment uses the officially provided Mask R-CNN (ResNet-101 FPN) model. When the NMS threshold N_t is determined, merely the correlation judgment conditions are changed. The correlation judgment is made using various combinations of the three correlation factors that are specified above and uses the threshold self-test procedure that will be described later to dynamically determine the value of the threshold N_c. The final prediction results of traditional NMS and Syncretic-NMS that are obtained using several correlation judgment methods are presented in Table 1.

TABLE 1. Comparison of the prediction results that were obtained using various combinations of correlation factors under the same model. AP denotes the average precision.

Model	Syncretic -NMS	Correlation Factor			AP
		Classification score	IoU	Adjacency	
Mask R-CNN					35.7
	√	√	√		**37.8**
	√	√		√	37.2
	√		√	√	36.9

Compared with traditional NMS, the use of various combinations of the three correlation factors for correlation

judgment can increase the prediction accuracy of the model. According to the prediction results in Table 1, we finally select the classification score and the IoU as the correlation factors for correlation judgment.

B. THRESHOLD SELF-TEST

Among scenarios, image datasets often differ in terms of their characteristics. For example, the image structure of the MS COCO [35] dataset is typically complicated, and there is often one or more large instances in the images. The images in the Cityscapes [36] dataset contain many small objects. The occlusions between objects are more severe, and the instances are more concentrated on two categories: person and car. The traditional NMS algorithm must adjust the threshold according to the application scenario. This problem is also encountered with Syncretic-NMS. When the application scenario is changed, the threshold must be adjusted to control the amplitude of the bounding box to avoid negative effects. This article utilizes two manually set thresholds: N_t, as in traditional NMS, must be manually adjusted to the optimal value according to the model, and N_c, which is used to control the correlation judgment, is set via a designed self-test procedure.

The experimental results on the MS COCO and Cityscapes datasets demonstrate that when N_c changes in the range [0,1], there is always a unique peak in the sensitivity of the network to Syncretic-NMS. As shown in Figure 2, in the range of [0.1,1], the prediction performance is evaluated from 0.1 in increments of 0.1. The threshold changes from small to large in the interval, and AP (average precision) increases and, subsequently, gradually decreases. As the threshold N_c approaches 1, the role of Syncretic-NMS weakens. When $N_c = 1$, no bounding box is added or merged, which is the same as traditional NMS. Therefore, the task of the threshold self-test can be simply transformed into a task of finding the peak in the interval [0,1].

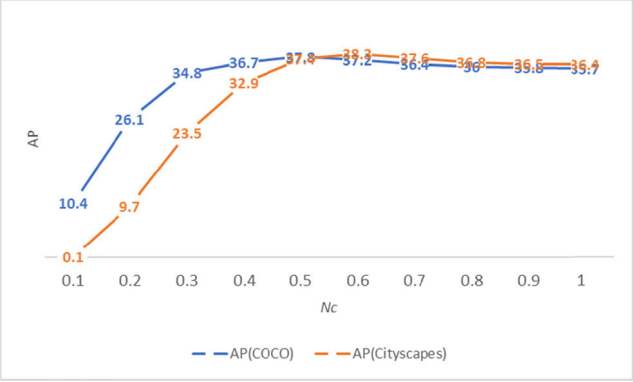

FIGURE 2. Sensitivity of Mask R-CNN to Syncretic-NMS on various datasets.

Based on this phenomenon, we design a threshold self-test procedure. On 500 specified dataset images (the images must use MS COCO-type annotations), the function first determines the prediction accuracy at the default threshold $N_c = 0.5$. Then, on these images, the prediction accuracy after the change of N_c is measured in increments and decrements of 0.1. If a higher prediction accuracy is realized, it continues to increase or decrease. When the accuracy reaches the peak, the final threshold will be determined. The final threshold is a constant value, and there is no need to readjust the threshold in the dataset. This method does not incur excessive time costs.

C. INSTANCE SEGMENTATION ON THE MS COCO DATASET

Syncretic-NMS is evaluated on the MS COCO dataset with 80 categories. The models we use are all publicly available official models, and they are trained on the union of 115k training images and 35k validation images (trainval 35k). After replacement with Syncretic-NMS, the available model was evaluated on a set of 5k validation images. To evaluate the performance of the model on instance segmentation, Syncretic-NMS is used to replace NMS on the classic instance segmentation network, namely, Mask R-CNN, and the current state-of-the-art instance segmentation network, namely, MS R-CNN (mask scoring region-convolutional neural network), for comparative quantitative experiments. In addition, the influence of the selection of the threshold N_c on the final result was evaluated. Finally, the effectiveness of the comparison model is visualized.

In the experiment, the AP of the mask is selected as the evaluation index, and the degree of approximation of the ground truth by the mask was compared. If the numbers of true-positive examples, true-negative examples, false-positive examples, and false-negative examples of the sample classification are defined as TP, TN, FP, and FN, respectively, the accuracy is:

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{6}$$

MS COCO's main evaluation index, namely, AP, refers to the average accuracy rates on 10 IoU levels and 80 categories. The IoU threshold is from 0.5 to 0.95, and the accuracy is evaluated once every step of 0.05. As the IoU threshold increases, the prediction result is closer to the ground truth, and the AP decreases. Then, the average of the 10 measurements is regarded as the final AP. On MS COCO, the average AP value of 80 categories is the final AP, also called mAP (mean average precision). The "AP" in all tables of the paper are mAP. AP_{50} and AP_{75} refer to the accuracies at IoU thresholds of 0.5 and 0.75, respectively, while AP_S, AP_M, and AP_L are average accuracies for small objects (area $\leq 32^2$), medium objects (32^2 < area $\leq 96^2$), and large objects (area $> 96^2$). The higher the AP is, the stronger the prediction ability is.

According to Table 2, Syncretic-NMS yields significantly higher values than NMS for each evaluation index and realizes approximately 2% improvement on each model. In addition, Syncretic-NMS realizes improvements on the small-, medium- and large-object evaluation indicators. To more clearly visualize the improved performance of

TABLE 2. Comparison of the results of instance segmentation networks using traditional NMS and Syncretic-NMS on the MS COCO dataset. The backbone networks are all ResNets [37] (residual networks). FPN [38] (feature pyramid network) and DCN [39] (deformable convolutional network) are utilized in the experiment.

Method	Syncretic-NMS	Backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN		ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	51.1
	√	ResNet-101-C4	34.9	57.9	37.6	14.8	37.4	52.0
		ResNet-101 FPN	35.7	58.0	37.8	15.5	38.1	52.4
	√	ResNet-101 FPN	37.8	61.7	40.8	17.9	39.9	53.2
MS R-CNN		ResNet-101 FPN	38.3	58.8	41.5	17.8	40.4	54.4
	√	ResNet-101 FPN	39.9	62.1	44.0	20.1	41.9	55.1
		ResNet-101 DCN-FPN	39.6	60.7	43.1	18.8	41.5	56.2
	√	ResNet-101 DCN-FPN	41.4	64.4	45.8	21.2	43.3	57.2

TABLE 3. Comparison results of Mask R-CNN using traditional NMS, Soft-NMS, Syncretic-NMS and the fusion version on the MS COCO dataset. All experiments in the table are performed on a single model (Mask R-CNN with backbone of ResNet-101 FPN). In the fusion version, Soft-NMS replaces the original NMS as the first step of Syncretic-NMS.

Method	Syncretic-NMS	Soft-NMS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN (ResNet-101 FPN)			35.7	58.0	37.8	15.5	38.1	52.4
		√	36.8	59.7	38.8	16.2	39.3	53.0
	√		37.8	61.7	40.8	17.9	39.9	53.2
	√	√	38.0	62.1	40.9	17.7	40.2	53.4

FIGURE 3. Visual comparison of Mask R-CNN using traditional NMS and Syncretic-NMS. From left to right are the original image, the prediction result that was obtained using NMS and the prediction result that was obtained using Syncretic-NMS. Syncretic-NMS can effectively improve the prediction result, the bounding boxes can more completely contain the entire object, and the objects that are outside the bounding box can now be correctly predicted.

Syncretic-NMS, we present a comparison effect chart in Figure 3. According to Figure 3, in the prediction result that is obtained using traditional NMS, the wheels and handlebars of the bicycle are partially outside the detected bounding box; hence, all pixels outside the bounding box fail to be predicted during segmentation. The bounding box that is obtained using Syncretic-NMS is larger, the bicycle is completely enclosed within the bounding box, and the pixels at the wheels and handlebars can be successfully predicted. Additional visual comparisons are presented in the appendix at the end of the paper.

In order to further prove the efficiency of Syncretic-NMS, we also conducted the comparative experiments with Soft-NMS and the fusion version (Syncretic-NMS built on Soft-NMS). As shown in Table 3, on the single model Mask R-CNN (ResNet-101 FPN), just the original NMS method is replaced for evaluation, both Soft-NMS and Syncretic-NMS

TABLE 4. The efficiency of Syncretic-NMS against traditional NMS.

Model	Syncretic-NMS	Average Prediction Time/ms
Mask R-CNN		556
	√	581

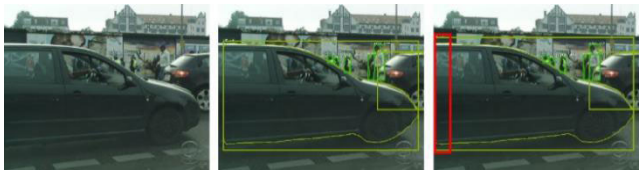

FIGURE 4. Examples of visual comparison on the Cityscapes dataset. To show the effect of Syncretic-NMS more clearly, the example image is a screenshot of the original image. The three columns of the image from left to right are a screenshot of the original image, the prediction result of Mask R-CNN, and the prediction result after using Syncretic-NMS. The red area in the figure is the enlarged range of the bounding box, which was manually labeled.

can improve the efficiency of the model. The performance of Syncretic-NMS is even better than that of Soft-NMS. In addition, when Soft-NMS replaced the original NMS as the first step of Syncretic-NMS, the efficiency of the fusion version is slightly better than that of the original Syncretic-NMS. Therefore, Syncretic-NMS is an effective and generalized NMS method for instance segmentation.

According to the efficiency results shown in Table 4, the evaluating results on an NVIDIA GTX 1080 Ti using

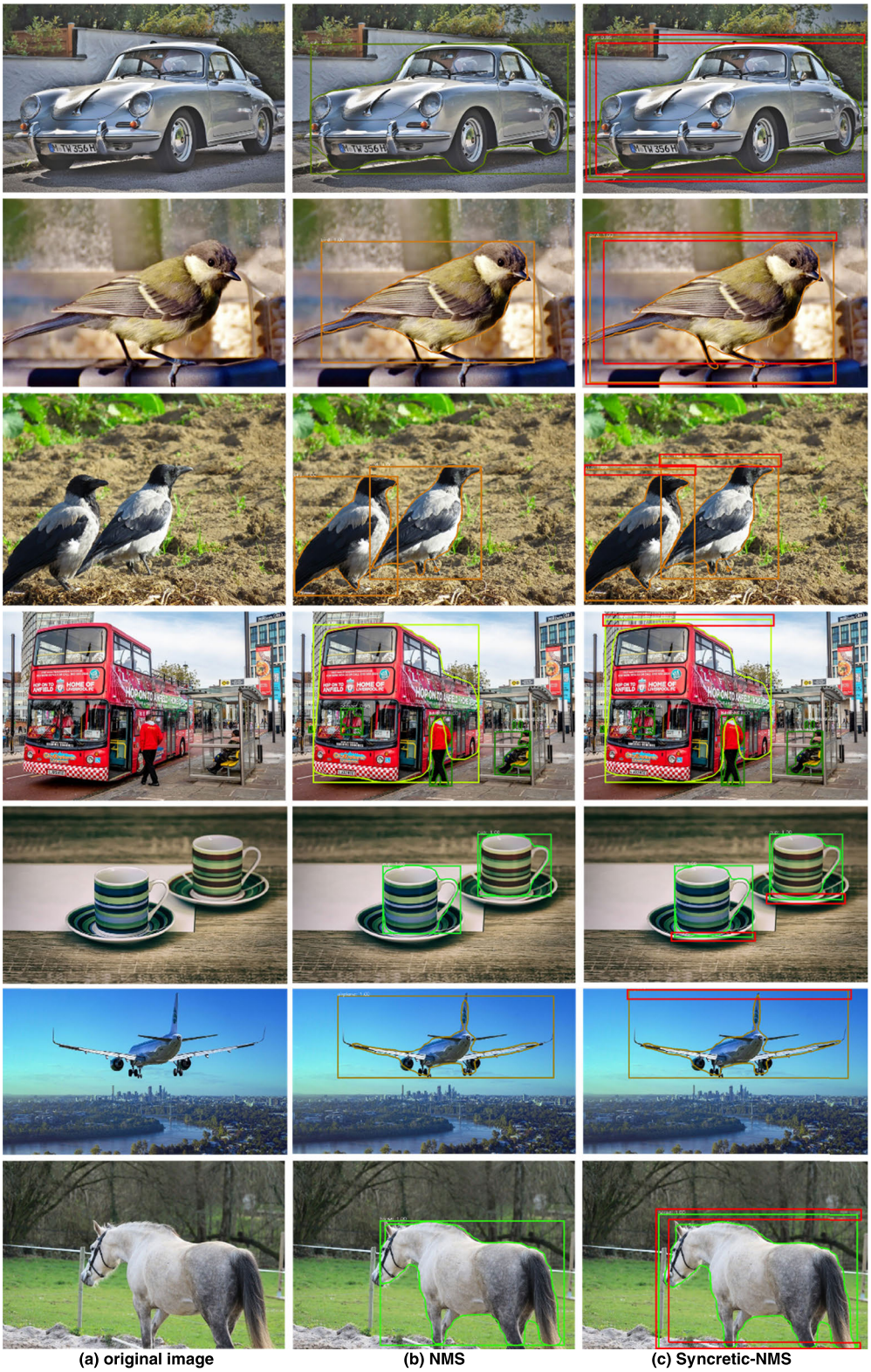

FIGURE 5. More visual comparison of Mask R-CNN using traditional NMS and Syncretic-NMS. Each column from left to right is the original image, the prediction result using NMS and the prediction result using Syncretic-NMS. The red area in the figure is the enlarged range of the bounding box manually labeled.

TABLE 5. Comparison of the results of instance segmentation networks using traditional NMS and syncretic-NMS on the cityscapes dataset.

Method	Syncretic -NMS	AP	person	rider	car	truck	bus	train	mcycle	bicycle
Mask R-CNN		32.0	34.8	27.0	49.1	30.1	40.9	30.9	24.1	18.7
	√	34.4	36.4	28.3	49.9	32.0	44.2	35.9	25.8	19.6

500 validation images demonstrate that the average prediction time per image is slightly increased. Syncretic-NMS does not substantially increase the calculational burden.

D. INSTANCE SEGMENTATION ON THE CITYSCAPES DATASET

Syncretic-NMS is also tested on the Cityscapes dataset. In contrast to the MS COCO dataset, the images of the Cityscapes dataset are from traffic scenes. There are more small objects in the images, and the occlusions between objects are more severe. We use the Mask R-CNN model that was trained on the MS COCO dataset to test whether Syncretic-NMS can adapt to other application scenarios on the Cityscapes dataset.

The Cityscapes dataset contains 2975 finely labeled training images, 500 validation images, and 1525 test images, all of which are the same pixel size. The core evaluation index, namely, AP, of Cityscapes dataset is consistent with that of the MS COCO dataset. The experiments also analyze the prediction results of various types of segmentation. We use Mask R-CNN (ResNet-50 FPN) that was trained on the MS COCO dataset as a benchmark. A threshold self-test procedure is used prior to testing. Table 5 presents the original results and the prediction results of the model after using Syncretic-NMS. Figure 4 presents a visualization example of this experiment. The vehicle part near the left boundary of the image can be correctly included in the bounding box under the action of Syncretic-NMS, and the improved part has been marked with a red box.

The experimental results demonstrate that Syncretic-NMS can satisfactorily adapt to changes in application scenarios, and when changing application scenarios, the threshold self-test procedure performs effectively.

VI. CONCLUSIONS

Syncretic-NMS is proposed in this paper, which is suitable for instance segmentation. It is used to obtain a bounding box that can well contain the complete object of interest and to obtain the relevant context information. Through correlation judgment and the corresponding coordinate mapping, the qualified neighboring boxes are merged using the traditional greedy NMS algorithm such that the returned bounding box is more suitable for subsequent semantic segmentation tasks. Through correlation judgment analysis, the most suitable correlation judgment method is identified, and a self-test procedure for the correlation judgment threshold is proposed accordingly so that Syncretic-NMS can be applied to various scenarios. Syncretic-NMS is easy to implement, does not require substantial additional computational complexity. In future research, we will develop a superior method for determining the threshold of association judgment to further improve the performance of the algorithm.

APPENDIX

See Figure 5.

REFERENCES

[1] Y. Zhang, J. Chu, L. Leng, and J. Miao, "Mask-refined R-CNN: A network for refining object details in instance segmentation," *Sensors*, vol. 20, no. 4, p. 1010, Feb. 2020.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 91–99.

[3] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[4] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert, "An empirical study of context in object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1271–1278.

[5] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3150–3158.

[6] H. Zhang, K. Wang, and F. Wang, "Advances and perspectives on applications of deep learning in visual object detection," *Acta Automatica Sinica*, vol. 43, no. 8, pp. 1289–1305, 2017.

[7] X. Zeng, W. Ouyang, B. Yang, J. Yan, and X. Wang, "Gated bi-directional CNN for object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 354–369.

[8] X. Zeng, W. Ouyang, J. Yan, H. Li, T. Xiao, K. Wang, Y. Liu, Y. Zhou, B. Yang, Z. Wang, H. Zhou, and X. Wang, "Crafting GBD-net for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 9, pp. 2109–2123, Sep. 2018.

[9] A. Rosenfeld and M. Thurston, "Edge and curve detection for visual scene analysis," *IEEE Trans. Comput.*, vol. C-20, no. 5, pp. 562–569, May 1971.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[11] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, 2004.

[12] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2015, pp. 1440–1448.

[13] R. Rothe, M. Guillaumin, and L. Van Gool, "Non-maximum suppression for object detection by passing messages between Windows," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 290–306.

[14] G. Zhang, S. Zhang, and J. Chu, "A new object detection algorithm using local contour features," *Acta Automatica Sinica*, vol. 40, no. 10, pp. 2346–2355, 2014.

[15] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *Int. J. Comput. Vis.*, vol. 38, no. 1, pp. 15–33, 2000.

[16] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS—Improving object detection with one line of code," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5561–5569.

[17] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, "Acquisition of localization confidence for accurate object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 784–799.

[18] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. ACM Multimedia Conf. (MM)*, 2016, pp. 516–520.

[19] L. Leng, J. Zhang, M. K. Khan, X. Chen, and K. Alghathbar, "Dynamic weighted discrimination power analysis: A novel approach for face and palmprint recognition in DCT domain," *Int. J. Phys. Sci.*, vol. 5, no. 17, pp. 2543–2554, Dec. 2010.
[20] L. Leng, J. Zhang, J. Xu, M. K. Khan, and K. Alghathbar, "Dynamic weighted discrimination power analysis in DCT domain for face and palmprint recognition," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Nov. 2010, pp. 467–471.
[21] L. Leng, M. Li, C. Kim, and X. Bi, "Dual-source discrimination power analysis for multi-instance contactless palmprint recognition," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 333–354, Jan. 2017.
[22] L. Leng and A. B. J. Teoh, "Alignment-free row-co-occurrence cancelable palmprint fuzzy vault," *Pattern Recognit.*, vol. 48, no. 7, pp. 2290–2303, Jul. 2015.
[23] L. Leng and J. Zhang, "PalmHash code vs. PalmPhasor code," *Neurocomputing*, vol. 108, pp. 1–12, May 2013.
[24] Y. He, C. Zhu, J. Wang, M. Savvides, and X. Zhang, "Bounding box regression with uncertainty for accurate object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2888–2897.
[25] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9157–9166.
[26] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," 2019, *arXiv:1905.05055*. [Online]. Available: http://arxiv.org/abs/1905.05055
[27] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2001, pp. 511–518.
[28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*. [Online]. Available: http://arxiv.org/abs/1312.6229
[29] D. Mrowca, M. Rohrbach, J. Hoffman, R. Hu, K. Saenko, and T. Darrell, "Spatial semantic regularisation for large scale object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2003–2011.
[30] J. Hosang, R. Benenson, and B. Schiele, "Learning non-maximum suppression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4507–4515.
[31] L. Wan, D. Eigen, and R. Fergus, "End-to-end integration of a convolutional network, deformable parts model and non-maximum suppression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 851–859.
[32] C. Desai, D. Ramanan, and C. C. Fowlkes, "Discriminative models for multi-class object layout," *Int. J. Comput. Vis.*, vol. 95, no. 1, pp. 1–12, Oct. 2011.
[33] P. Henderson and V. Ferrari, "End-to-end training of object class detectors for mean average precision," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 198–213.
[34] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.
[35] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[36] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.
[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[38] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[39] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.
[40] M. A. Rahman and Y. Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *Proc. Int. Symp. Vis. Comput.* Cham, Switzerland: Springer, 2016, pp. 234–244.

JUN CHU received the Ph.D. degree from Northwestern Polytechnic University, Xi'an, China, in 2005.

She was a Postdoctoral Researcher with the Exploration Center of Lunar and Deep Space, National Astronomical Observatory of Chinese Academy of Sciences, from 2005 to 2008. She was a Visiting Scholar with the University of California at Merced, USA. She is currently the Director of the Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, and a Full Professor with the School of Software, Nanchang Hangkong University. Her research interests include computer vision and pattern recognition. She was also a member of the Computer Vision Special Committee and China Computer Federation.

YIQING ZHANG received the master's degree from Nanchang Hangkong University, Nanchang, China, in 2020. His research interests include instance segmentation and object detection.

SHAOMING LI received the B.S. degree from Sichuan Agricultural University, Ya'an, China, in 2018. He is currently pursuing the M.S. degree with Nanchang Hangkong University, Nanchang, China. His research interests include computer vision, image processing, and machine learning.

LU LENG (Member, IEEE) received the Ph.D. degree from Southwest Jiaotong University, Chengdu, China, in 2012.

He performed his Postdoctoral Research with Yonsei University, Seoul, South Korea, and the Nanjing University of Aeronautics and Astronautics, Nanjing, China. He was a Visiting Scholar with West Virginia University, USA. He is currently an Associate Professor with Nanchang Hangkong University and a Visiting Scholar with Yonsei University. He has published more than 70 international journal and conference papers. He has been granted several scholarships and funding projects for his academic research. He is a Reviewer of several international journals and conferences. His research interests include image processing, biometric template protection, and biometric recognition.

Dr. Leng is a member of the Association for Computing Machinery (ACM), the China Society of Image and Graphics (CSIG), and the China Computer Federation (CCF).

JUN MIAO received the Ph.D. degree from Nanchang University, Nanchang, China, in 2015. He is currently a Researcher with the Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, and an Associate Professor with the School of Aeronautical Manufacturing Engineering, Nanchang Hangkong University. His research interests include computer vision, 3D reconstruction, and pattern recognition.