Tilt Correction Toward Building Detection of Remote Sensing Images

Kang Liu, Zhiyu Jiang, Mingliang Xu, Matjaž Perc, and Xuelong Li, *Fellow, IEEE*

***Abstract*—Building detection is a crucial task in the field of remote sensing, which can facilitate urban construction planning, disaster survey, and emergency landing. However, for large-size remote sensing images, the great majority of existing works have ignored the image tilt problem. This problem can result in partitioning buildings into separately oblique parts when the large-size images are partitioned. This is not beneficial to preserve semantic completeness of the building objects. Motivated by the above fact, we first propose a framework for detecting objects in a large-size image, particularly for building detection. The framework mainly consists of two phases. In the first phase, we particularly propose a tilt correction (TC) algorithm, which contains three steps: texture mapping, tilt angle assessment, and image rotation. In the second phase, building detection is performed with object detectors, especially deep-neural-network-based methods. Last but not least, the detection results will be inversely mapped to the original large-size image. Furthermore, a challenging dataset named Aerial Image Building Detection is contributed for the public research. To evaluate the TC method, we also define an evaluation metric to compute the cost of building partition. The experimental results demonstrate the effects of the proposed method for building detection.**

***Index Terms*—Building detection, cost of building partition (CoBP), deep neural network (DNN), remote sensing, tilt correction (TC).**

I. Introduction

BUILDING detection plays a crucial role in the field of remote sensing, such as urban planning, natural disaster survey, illegal construction surveillance, antiterrorism surveillance, and emergency landing [1]–[6]. From the perspective of information acquisition, the remote sensing images can be divided into synthetic aperture radar (SAR) images, light detection and ranging (LiDAR) images, and optical images.

Compared to the optical images, the signal-to-noise ratio (SNR) of SAR and LiDAR is relatively low. However, the SAR comparatively has much stronger ability of cloud and mist transmission. Hence, the SAR can complement the limits of optical sensors, especially when the optical sensors lose efficacy due to their daylight and weather dependence. In order to combine better results, Dubois *et al.* [7] utilized two types of detector to complement with each other for the SAR phase images. Based on the K-singular value decomposition method, Adelipour and Ghassemian [8] proposed a building detection method using sparse representation to learn dictionaries. To ensure a consistent and fast computation of the complex SAR information, Ferro *et al.* [9] adopted some low-level features and their composition features, such as length, width, and height, to the facade detection. Aiming at the problem of automatically detecting manmade structures in very high resolution SAR images, Shahzad *et al.* [10] first used advanced interferometric techniques to classify the spaceborne SAR tomography point clouds and then used the fully convolutional network (FCN) to detect the buildings.

In comparison to the SAR, the working frequency of LiDAR is much higher. The LiDAR also has the ability of accurate ranging and high resolution. For offering a high success rate for building detection, Cai *et al.* [11] proposed a coarse-to-fine strategy, which is based on semisuppressed fuzzy C-means and restricted region growing. Dey and Awrangjeb [12] applied a corner correspondence algorithm to give an evaluation metric on the extracted boundary. Chen *et al.* [13] adopted a multiscale grid method to reconstruct airborne LiDAR data and detect building roofs. Without filtering the ground points, a virtual first and last pulse method is proposed to detect the buildings [14]. To discriminate building regions from LiDAR data, a vegetation mask-based connected filter algorithm based on digital surface model data of LiDAR point cloud is contributed in [15].

The SNR of the optical images is relatively high, and the detailed information is much rich. Much more optics-based methods can be divided into traditional methods and deep neural network (DNN)-based methods. The traditional methods are usually based on hand-designed features and require much technical expertise [16]. Huang *et al.* [17] proposed a morphological building index (MBI) to describe the building characteristics. Based on the shape, spectral, geometric, and contextual information, the MBI is the comprehensive semantic

Manuscript received March 16, 2021; revised May 14, 2021; accepted May 21, 2021. Date of publication May 25, 2021; date of current version June 16, 2021. This work was supported in part by the Key Research Program of Frontier Sciences, Chinese Academy of Sciences under Grant QYZDY-SSW-JSC044, in part by the National Natural Science Foundation of China under Grant 61871470 and Grant 62001397, in part by the Natural Science Basic Research Program of Shaanxi under Grant 2020JQ-212, and in part by the Open-Ended Foundation of National Radar Signal Processing Laboratory under Grant 61424010207. *(Corresponding author: Xuelong Li.)*

Kang Liu is with the Shaanxi Key Laboratory of Ocean Optics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: liukang@opt.ac.cn).

Zhiyu Jiang and Xuelong Li are with the Key Laboratory of Intelligent Interaction and Applications (Ministry of Industry and Information Technology) and the School of Artificial Intelligence, Optics and Electronics, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: jiangzhiyu@nwpu.edu.cn; li@nwpu.edu.cn).

Mingliang Xu is with the School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: iexumingliang@zzu.edu.cn).

Matjaž Perc is with the Faculty of Natural Sciences and Mathematics, University of Maribor, 2000 Maribor, Slovenia (e-mail: matjaz.perc@gmail.com).

Digital Object Identifier 10.1109/JSTARS.2021.3083481

information of manmade objects. Li *et al.* [18] proposed a multiscale morphological attribute index to extract buildings. This method can overcome the inherent defects of the MBI to a certain degree. Inspired by the observed geometric features, Huang *et al.* [19] proposed a geometric building index for accurate building detection. Norman *et al.* [20] combined the plateau objective function and the statistical method to extract building footprint. This method is a space statistical optimization method. Agarwal and Rajan [21] extracted candidate building pixels to the maximum stable extremum region (MSER). Uniting with independent component analysis, these geometric features are then used to choose buildings. Karadag *et al.* [22] fused the related problem information of buildings to the segmentation step for building detection. However, the aforementioned methods are traditional methods, and the generalization and robustness of these methods are limited and unstable.

In order to compensate for the traditional building detection models, a large number of DNN-based works have emerged [23]–[25]. This trend is reaping huge fruits from the DNN, which has been widely used in the natural image applications. Alidoost and Arefi [26] used a single aerial image and a convolutional neural network (CNN) to verify the ability of building detection and roof identification. To overcome the diverse spatial resolutions, scales, and orientations, Hamaguchi *et al.* [27] combined a few CNN models. For the trained CNN model, Dong *et al.* [28] designed object features with suitable scales. To relieve the diversity problem of samples, Zhu *et al.* [29] presented the first generative adversarial network (GAN)-based data augmentation method. The method is based on a multibranch conditional GAN. Especially conducive to large-scale buildings, Ji *et al.* [30] contributed a weight-shared Siamese U-Net network. For arbitrary direction buildings, Yang *et al.* [31] made a detection network with U-rotation to find accurate bounding boxes. Based on near-infrared information, Fang *et al.* [32] utilized fine spatial resolutions to detect building shadow. Bai *et al.* [33] aligned the texture information with region of interest and density residual network, so that the regional mismatching problem can be solved. To detect dense and small buildings, Shu *et al.* [34] proposed an end-to-end model guided by the center point. Jiang *et al.* [35] used an encoder–decoder network and a residual refinement module to form a predictive architecture for the prediction. Yao *et al.* [36] applied the visual saliency and condition random field to train a coarse-to-fine model. The model is mainly aiming to detect airports. Xie *et al.* [37] improved the real-time YOLO [38] algorithm to be a new framework with local constraints. Reda and Kedzierski [39] proposed a faster edge region CNN algorithm for improving building detection. In order to automatically detect buildings, Shahzad *et al.* [10] utilized an FCN to train the model. To reduce the influence of complex backgrounds, Du *et al.* [40] integrated with the saliency map to design a single-shot method.

However, the most reviewed works have ignored the building tilt problem of large-size remote sensing images. Consequently, these tilted buildings are not beneficial to keep the semantic completeness of the buildings in the task of object detection, because the buildings are usually divided into separately oblique parts when the large-size images are partitioned, as shown in Fig. 2. On the one hand, we have observed that the distribution of buildings, especially in urban cities, is usually with certain relations to the environment or surrounding facilities. It can be found that the map of land planning and directions of roads have some relationships with the distribution of buildings. On the other hand, we have found that the outlook shapes of buildings are mostly rectangle-like and have obvious boundary lines. This clue is also rewarding to the prediction of building tilt angle. Thus, we can utilize these priors to predict the tilt angle for the correction of remote sensing images.

Motivated by the above fact, this article mainly focuses on the formulation of the tilt correction (TC) algorithm, which takes advantage of the edge information of the large-size remote sensing images. The *main contributions are fourfold* and summarized as follows.

1) We propose a new framework for detecting buildings in large-size images. First, the TC is performed based on tilt angle estimation. Second, building detection is performed with object detectors. Finally, the detection results are inversely mapped to the original large-size image. The results are more reasonable, because the buildings can avoid the oblique cutting of partition and the completeness of objects can be preserved to a certain extent.
2) A TC algorithm is especially proposed to solve the tilt problem of remote sensing images. This is a simple and effective method, which estimates tilt angles by linear edge detection and statistic histogram.
3) An evaluation metric named cost of building partition (CoBP) is defined. The CoBP is a quantitative indicator, which can evaluate the average CoBP for the whole image dataset.
4) A publicly available Aerial Image Building Detection (AIBD) is contributed. The box annotations of AIBD are converted from a publicly semantic segmentation dataset. The AIBD dataset contains totally 11 571 samples and annotated in the form of PASCAL VOC[1] and also converted into the form of COCO.[2]

The remainder of this article is organized as follows. Some related works are reviewed in Section II. The proposed methodology introduced in Section III. Section IV presents the dataset, evaluation metrics, experiment results, and their analysis. Finally, Section V concludes this article.

II. Related Works

In this section, we will briefly review several works, which are mostly related to this article, including object detection methods and TC. The object detection methods used in this article are mainly based on the DNNs.

A. Object Detection Methods

Object detection is a fundamental task in the field of computer vision, which mainly focuses on object positioning and object classification in an image or video sequence. It can be widely

[1] [Online]. Available: https://host.robots.ox.ac.uk/pascal/VOC/
[2] [Online]. Available: https://cocodataset.org/

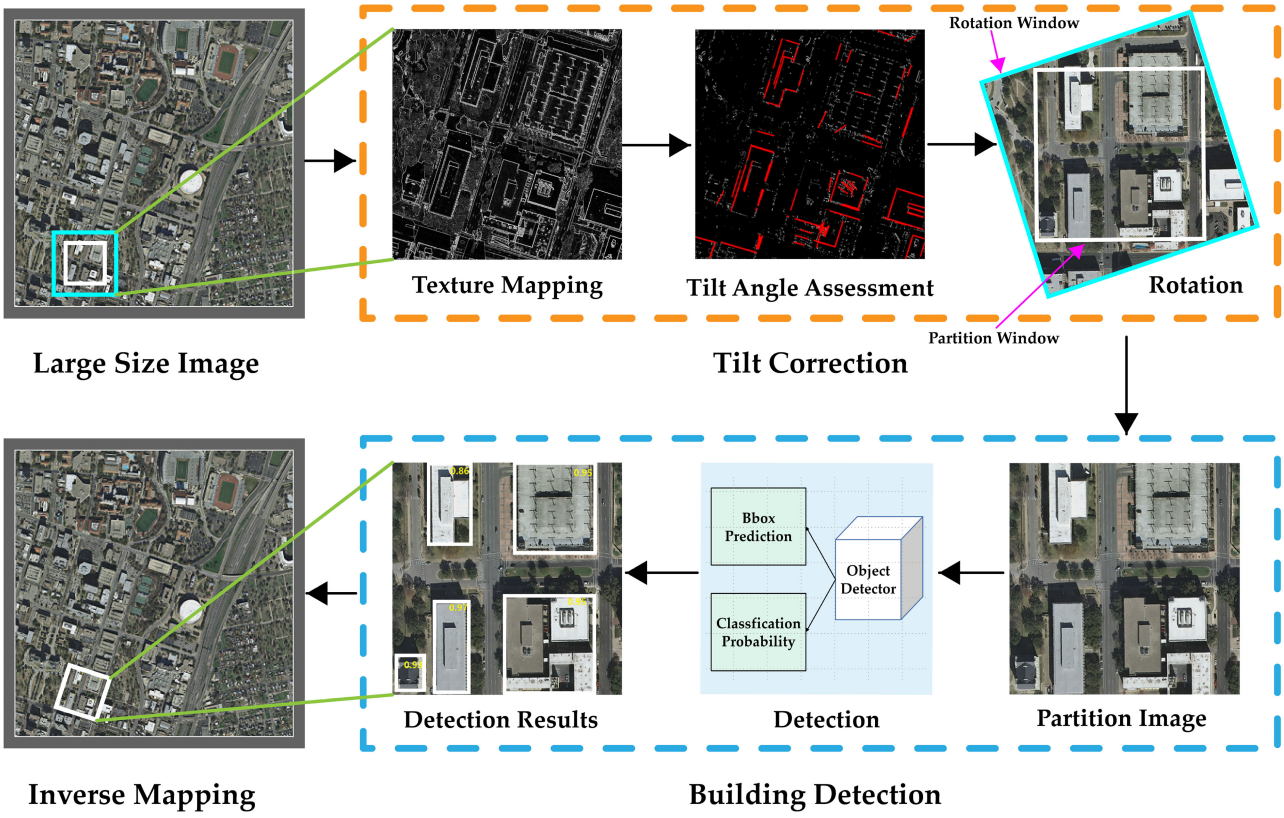

Fig. 1. Flow architecture of the proposed method. The flow architecture mainly contains two phases: TC and building detection. The TC includes three steps: separately texture mapping, tilt angle assessment, and image rotation and partition. The second phase performs building detection on the partitioned images by means of object detectors. The detection results will be eventually mapped into the original large-size image.

Fig. 2. Examples of TC. (a) Original examples without TC. (b) Processed examples with TC. It can be observed that the building instances without TC, particularly for the components surrounded by red ellipse, are partitioned into oblique parts. This phenomenon is not beneficial to the semantic completeness of the buildings. However, those with TC are mostly not partitioned into oblique parts.

used in robot navigation, intelligence video surveillance, industry detection, aeronautics and astronautics, and so on [41]. The DPM detector [42] is a representative peak of the traditional object detection algorithms, which achieved state-of-the-art results on the PASCAL VOC benchmarks in the years of 2007, 2008, and 2009. Based on the core idea of "divide and conquer," DPM is an evolution method from the HOG detector [43]. The final results of DPM are obtained by combining the inference of the different object components.

The DNN-based detection techniques have been proved to possess good excellent performance among natural image scenes. These methods perform much better than traditional detectors and can be mainly regarded as two-stage methods and one-stage methods. The two-stage methods usually perform region proposals first and then determine the category and location of the candidate objects. There are several representative methods. In 2014, the region-based convolutional neural network (R-CNN) [44] is proposed. The selective search method is utilized to generate region proposals, and then, the fix-resized images are input to AlexNet, where the map features are extracted. Finally, the classifier support vector machine is used to predict the category for each candidate. However, the R-CNN has much computation time for so many region proposals. Later on, Girshick *et al.* [45] proposed the Fast R-CNN, which supports the bounding box regression, but is still time-consuming. Combining with the region proposal network, Ren *et al.* [46] proposed the Faster R-CNN, which is the first

end-to-end deep detector [46]. Adding a segmentation branch for the Faster R-CNN, He *et al.* [47] proposed the Mask R-CNN as a generic instance segmentation architecture.

Some other researchers regarded the object detection as a regression problem. Most of the one-stage methods determine the category and location in a unified phase [38], [48], [49]. Compared with the two-stage methods, the running time of YOLO [38] is superfast, and the data volume is competitively small. Meanwhile, the SSD [48] combining the regression theory and the anchor mechanism can balance the precision and the inference speed. In order to solve the imbalanced problem of the one-stage methods between the positive samples and negative samples, a focal loss is designed in the RetinaNet [49]. The RetinaNet can achieve the mean average precision (mAP) with the two-stage methods.

The aforementioned methods are mostly anchor-based methods. There are mainly three shortcomings of these methods:

1) imbalanced positive and negative samples;
2) hyperparameters are difficult to adjust;
3) intersection over union (IoU) matching is time-consuming.

Therefore, many anchor-free methods are proposed for the object detection in recent years [50]–[52]. These methods mainly focus on the key points of the objects. The CornerNet [50] assumes that the paired corner points have similar embedding vectors and forms a heatmap for the bottom-right corner. This hypothesis is not always effective because some different appearances have similar embedding vectors. Based on the CornerNet and adding a center point, the CenterNet [51] forms a triplet to denote keypoints. However, the CenterNet cannot perform well for the dense objects. Attempting to avoid above shortcomings, the CentripetalNet [52] utilized the centripetal shift of object keypoints.

B. Tilt Correction

TC is usually needed before the task of detection and recognition in many application scenes, such as face recognition, character recognition, license plate recognition, etc. This process can reduce the detection and recognition difficulty because of the image tilt. For the ship license number recognition, Liu *et al.* [53] utilized MSER-based center points and $L1-L2$ distance line fitting. To perform character recognition, Li *et al.* [54] used the Hough transform to calculate the angle of plate rotation. This method estimated the coordinates of four corners of the plate region as well as the angles of each character rotation. Based on the character median line, Yang *et al.* [55] adopted a corner detection and projection method for license plate TC. In the field of remote sensing, object detection is also possessing enormous demand. And it plays an important role in many application scenes, such as airplane detection, ship detection, airport detection, building detection, etc. These methods also give inspiration to our proposed method.

III. Proposed Methodology

In this section, we will introduce the proposed methodology in detail. The flow architecture of the proposed method is shown in Fig. 1. The flow architecture mainly contains two phases: TC and building detection. As the first phase, the TC algorithm includes three steps: texture mapping (see Section III-A), tilt angle assessment (see Section III-B), and image rotation and partition (see Section III-C). Building detection, as the second phase, is explained in Section III-D. Building detection will be performed in the partitioned images by means of object detectors. The detection results will be eventually mapped to the original large-size image.

A. Texture Mapping

In this subsection, we will present texture mapping of the TC algorithm. For large-size remote sensing images, we have observed that the geographical distribution of buildings is mostly coupled with road planning orientation. This fact can contribute a clue to relieve the partition problem. In order to extract the orientation information for the tilt angle assessment, texture mapping is calculated. At the beginning, a denoising filter is performed to keep the edge information. The denoising filter is formed as

$$I_f(i,j) = \frac{\sum_{m,l} I(m,l)w(i,j,m,l)}{\sum_{m,l} w(i,j,m,l)} \tag{1}$$

where $I_f(i,j)$ is the filtered intensity of pixel (i,j) and $I(m,l)$ is the intensity of the center pixel (m,l) of the filter window. The term $w(i,j,m,l)$ is the filter weight, calculated as follows:

$$w(i,j,m,l) = \exp\left(-\frac{(i-m)^2+(j-l)^2}{2\sigma_s^2} - \frac{(I(i,j)-I(m,l))^2}{2\sigma_c^2}\right) \tag{2}$$

where σ_s^2 and σ_c^2 are the space smooth parameter and the color intensity smooth parameter, respectively. σ_s and σ_c are both set as 2 in this article. Then, texture mapping is obtained by

$$G = \alpha \times |G_x| + (1-\alpha) \times |G_{\mathrm{y}}| \tag{3}$$

where $|G_x|$ and $|G_y|$ are the gray values obtained with the horizontal edge detector and the vertical edge detector for the filtered image, respectively, and the term α is a weighted factor, which is set as 0.5 in this article.

For ensuring that the rotated image patches do not contain black areas after rotation, we perform boundary padding for the white box referred to in Fig. 1. The cyan square box is the padding patch (rotation window), and the padding width is calculated as

$$w_p = \left(\frac{\sqrt{2}-1}{2}\right) w_i \tag{4}$$

where w_i is the width of the purpose image patch i and w_p is the padding width of i. The size of square patch is $w_i \times w_i$ (500×500 in this article), so the maximum black area is appeared when the rotation angle is $\pm 45^\circ$ with the rotation radius of $\frac{\sqrt{2}}{2} w_i$. For a large-size image, the padding area is filled with the mean color of the whole image.

B. Tilt Angle Assessment

The tilt angle assessment is performed in this subsection. First of all, linear edges are detected on texture mapping. The main direction of image appearance usually depends on long lines. Therefore, the tilt angle of these long detected lines is calculated. The progressive probabilistic Hough transform (PPHF) [56] is utilized to detect the lines. Compared to the classical Hough transform for line detection, the PPHF mainly reduces the computing costs. The cross points with over vote-threshold number in the polar coordinates will be selected as the candidates. The lines related to these cross points are found as the detected lines. The detected lines are filtered by the length threshold (set as 40 pixels in this article). This approach can avoid the negative influence of nonsequence or background points.

For urban rectangle buildings, unparallel lines are usually vertical to each other. Due to the problem of rotation cycle, the linear edges of one building may have arctan tilt angles with different plus-minus signs. In order to calculate the tilt angle histogram and get the final tilt angle of the rotation window, we need to unify the angles. The final assessment tilt angle is its complement angle when the tilt angle of some linear edges is negative, as follows:

$$d_j = \begin{cases} d_j, & \text{if } d_j > 0 \\ 90 - |d_j|, & \text{otherwise} \end{cases}. \tag{5}$$

The tilt angle histogram H is used to calculate the rotation angle degree of the rotation window. H_i is the ith bin of H, which is defined as

$$H_i = \sum_{j=1}^{M} F\left(i = \left\lfloor \frac{d_j}{d_s} \right\rfloor\right) \tag{6}$$

where M denotes the number of detected lines, d_j is the arctan value, and the term d_s is the angle degree step and set as 10° in this article; thus, the histogram has nine bins. In addition, the $F(\cdot)$ is the indicator function, which equals 1 if the condition is established.

Then, the rotated degree θ is obtained with weighted l_j as

$$\theta = \frac{1}{W} \sum_{j=1}^{K} d_j \times l_j \tag{7}$$

where K denotes the number of detected lines of the max peak bin in H, l_j is the length of line j, and $W = \sum_{j=1}^{M} l_j$ is the sum of the line length of the max peak bin. Thus, the tilt angle of the rotation window is calculated.

C. Building Image Rotation

Building image rotation is based on the theory of affine transformation. The affine transformation usually performs through translation, scale, rotation, flip, and shear in the coordinate system. In order to get better results, usually, the rotation center is the center of the building image. The rotation will be performed as follows:

$$\begin{aligned} x_c &= (w_i + 2 \times w_p)/2 \\ y_c &= (w_i + 2 \times w_p)/2 \end{aligned} \tag{8}$$

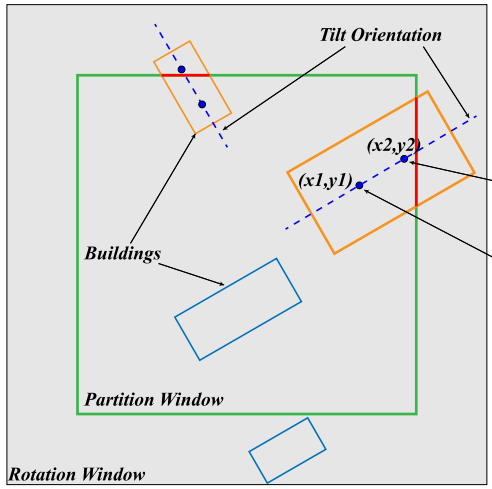

Fig. 3. Drawing of CoBP. The yellow rectangular boxes are contributed to the CoBP, while the blue rectangular boxes are not contributed to the CoBP. $A1$ and $A2$ are two points along the tilt orientation line of the partitioned building.

where (x_c, y_c) is the point coordinate of the rotation center, w_i is the width of the purpose image patch i, and w_p is the padding width of i. The rotated degree θ is obtained in Section III-B, and θ will be used to compute the point coordinate after rotation as follows:

$$\begin{aligned} x_2 &= (x_1 - x_c)\cos\theta - (y_1 - y_c)\sin\theta + x_c \\ y_2 &= (x_1 - x_c)\sin\theta + (y_1 - y_c)\cos\theta + y_c \end{aligned} \tag{9}$$

where (x_1, y_1) denotes the point coordinate before rotation and (x_2, y_2) denotes the point coordinate after rotation. The partition window by the white box with width w_i will be partitioned from the rotated window with width $w_i + w_p$. The rotated window does not contain the black area. Two examples of TC are shown in Fig. 2. The three subsections introduced above are the main steps of the proposed TC algorithm, which is also summarized in Algorithm 1.

D. Building Detection and Inverse Mapping

The partitioned image will be input into the object detector in the building detection phase. The object detector can be two-stage methods, one-stage methods, anchor-based, and anchor-free methods. Most of the recent object detection methods possess both predicted bounding boxes and classification probability scores. The detection results can be shown visually with bounding boxes and probability scores. Large-size remote sensing images usually contain geography and elevation. The detection results will be more reasonable and comprehensive if integrating this information. Therefore, the detection results are necessary to be inversely mapped into the large-size image. Inverse mapping of the detected bounding boxes will be performed for the detection results with inverse mapping degree $-\theta$ referred to

$$\begin{aligned} x_{p1} &= (x_{p2} - x_c)\cos(-\theta) - (y_{p2} - y_c)\sin(-\theta) + x_c \\ y_{p1} &= (x_{p2} - x_c)\sin(-\theta) + (y_{p2} - y_c)\cos(-\theta) + y_c \end{aligned} \tag{10}$$

Fig. 4. Examples of the AIBD. The geometric shapes, color characteristics, and scale variations are tremendously different.

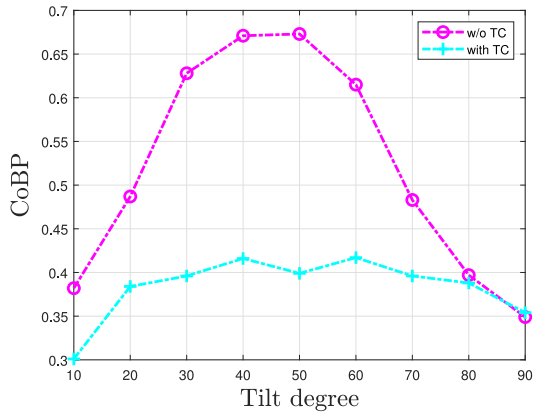

Fig. 5. Curves of CoBP.

where (x_{p1}, y_{p1}) is the point coordinate of the detected object box in the partition window and (x_{p2}, y_{p2}) is the point coordinate of the detected object box in the image after inverse mapping rotation. However, the relative position index should be taken into consideration for the final coordinate in the large-size image.

E. Cost of Building Partition

To quantitatively evaluate the effect of the TC method, we also define a metric named CoBP in (3). The metric CoBP can evaluate the average CoBP for the whole image dataset. It can give a quantitative indicator with the TC method. The drawing of CoBP is shown in Fig. 3

$$\text{CoBP} = \frac{1}{\text{N} \times \text{M}} \sum_{i=1}^{\text{N}} \sum_{j=1}^{\text{M}} \exp(-(1-V_T)^2) \tag{11}$$

where N denotes the number of sample images and M is the number of buildings, which are partitioned by the partition window in a rotation window. $\Delta x_{ij}{=}|x1_{ij} - x2_{_{ij}}|$ and $\Delta y_{ij}{=}|y1_{_{ij}} - y2_{_{ij}}|$ are, respectively, the horizon distance and the vertical distance between the two points $A1$ and $A2$ of the jth building in the ith rotation window. Because of the periodicity of the tilt degree, the tangent value V_T of the tilt building is calculated as

$$V_T = \frac{\min(\Delta x_{ij}, \Delta y_{ij})}{\max(\Delta x_{ij}, \Delta y_{ij})}. \tag{12}$$

In Fig. 3, the buildings on the boundary of the partition window are taken into consideration, while the others are not. The best situation is that no buildings are partitioned into tilted parts. However, it is unavoidable that some buildings are partitioned into parts. We want to know how much the TC algorithm can decrease the CoBP for the whole testing dataset. Therefore, the CoBP is calculated on the building instances, which are partitioned by the partition window. The maximum is obtained when V_T equals 0, while the minimum is obtained when V_T equals 1. Hence, the value range of the CoBP is 0.368–1.0.

Algorithm 1: TC for Building Detection.

Input: Large-size image I, patch width w_i, patch number N
Output: N corrected image patches

For image I:
for $i = 1$ to N **do**
  Make boundary padding as i_p with width w_p using (4) for patch i.
  Calculate weighted texture mapping T_i referring to (3).
  Perform edge detection on T_i and obtain image E_i.
  Find M lines of E_i utilizing line detector.
  for $j = 1$ to M **do**
    Calculate arctan value d_i and length l_j of line L_j as (5).
    Compute angle histogram H in (6).
  end for
  Assess tilt angle θ referring to (7).
  Perform image rotation on i_p adopting θ.
  Partition image patch i from i_p with width w_i.
end for

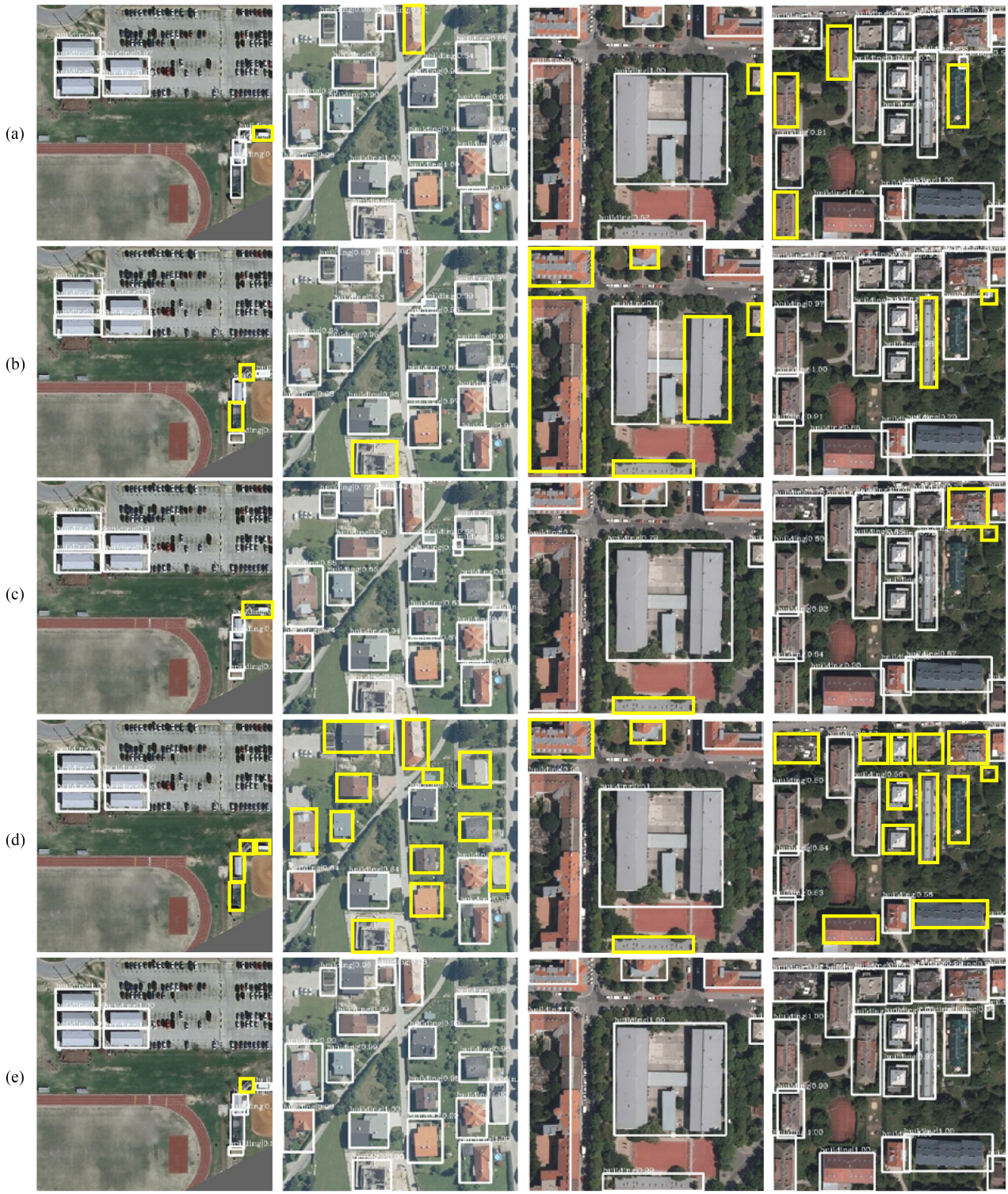

Fig. 6. Qualitative comparisons of (a) SSD-512, (b) YOLOv3-608, (c) RetinaNet, (d) CentripetalNet, and (e) Faster R-CNN (with resnet101). The Faster R-CNN with TC achieves the best results. The white bounding boxes are as TPs and the yellow bounding boxes are FNs.

IV. Experiments and Results

A. Dataset

For the evaluation requirement of the TC, a specialized dataset is needed. However, most of the existing datasets for building detection are already image patches, which cannot meet this requirement. Therefore, a publicly available AIBD is annotated in the form of PASCAL VOC[3] and also converted into the form of COCO.[4] The original images of AIBD are based on the Inria Aerial Image Data,[5] which are mainly used for the semantic

[3][Online]. Available: http://host.robots.ox.ac.uk/pascal/VOC/
[4][Online]. Available: https://cocodataset.org/
[5][Online]. Available: https://project.inria.fr/aerialimagelabeling/

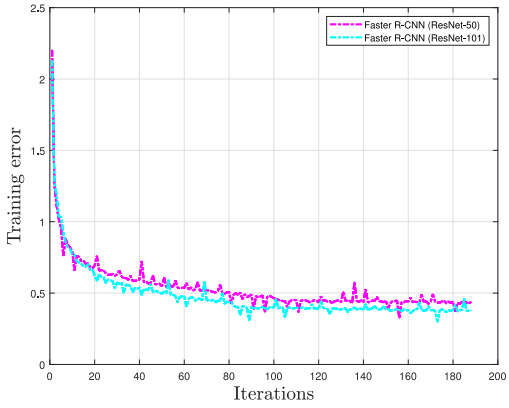

Fig. 7. Curves of training error of Faster R-CNN [46] with TC.

TABLE I
BUILDING STATISTICS OF AIBD

Scale	Instance Number	Percentage	Pixel Number
small	51977	0.273	$< 32^2$
medium	121515	0.638	$32^2 \sim 96^2$
large	16824	0.089	$> 96^2$

segmentation [57]. The building annotated labels of the AIBD are converted from the semantic annotations of Inria Aerial Image Data.

The Inria Aerial Image Data include a train subset and a test subset, which both have two semantic classes: building and not building. The train subset and the test subset both have five urban cities, and each city covers 81 km^2 with 36 image tiles, so the total number of tiles for both the train subset and the test subset is 180 and covers 405 km^2. The tile size is 5000 $\times$ 5000 pixels with 0.3-m resolution. The train subset has its reference data.

The AIBD can be accessed and publicly downloaded as soon as possible from Baidu Netdisk.[6] The AIBD includes two semantic classes: building and not building. The large-size images are partitioned into 500 $\times$ 500 image patches with the TC algorithm. The AIBD dataset totally contains 11 571 image samples and the same number of annotation files. Some statistics of AIBD are shown in Table I. The instance number, percentage, and pixel number of the small, medium, and large building instances are reported. The number of building instances is 51 977, 121 515, and 16 824 for the small instances, medium instances, and large instances, respectively. Correspondingly, the percentage of the small, medium, and large building instances is separately 0.273%, 0.638%, and 0.089%. The division criterion is based on the COCO metric. The AIBD is a challenging dataset for the task of building detection. Fig. 4 shows some examples of the AIBD. The geometric shapes, color characteristics, and scale variations are tremendously different. The geometric shapes, color characteristics, and scale variations of the building instances are also summarized as follows.

[6]Online. [Available]: https://pan.baidu.com/s/1u8XKQnADIa2pNTvRVnSntA (password:v09f)

TABLE II
EXPLANATIONS OF THE STANDARD COCO METRICS [46]

Metric	Explanation
AP_s	AP for small objects (areas are smaller than 32^2)
AP_m	AP for medium objects (areas are between 32^2 and 96^2)
AP_l	AP for large objects (areas are bigger than 96^2)
AP	AP at IoU = .50:.05:.95 (average over IoU thresholds)
AP_{50}	AP at IoU = .50 (equally to PASCAL VOC metric)
AP_{75}	AP at IoU = .75 (much strict metric)

1) *Geometric shapes:* The geometric shapes of AIBD are variable, such as rectangular, L-shape, U-shape, T-shape, and other irregular shape with many right angles.

2) *Color characteristics:* The color characteristics are distinct from each other among tremendously different backgrounds.

3) *Scale variations:* The pixel number of building objects ranges from tens to hundreds of thousands.

B. Evaluation Metrics

The average precision (AP) and its derivative metrics are adopted to quantitatively evaluate the proposed method for the task of building detection. The standard COCO metrics, including AP, AP_{50}, AP_{75}, AP_s, AP_m, and AP_l, are briefly reported in Table II. The metric AP is a comprehensive indicator in the field of object detection, comprehensively considering the metric *precision* and *recall* as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{13}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{14}$$

where the term TP is true positives, FP is false positives, and FN is false negatives. These terms are calculated from the IoU between the predicted bounding box and the ground truth bounding box

$$\text{IoU} = \frac{B_{\text{predict}} \cap B_{\text{gt}}}{B_{\text{predict}} \cup B_{\text{gt}}} \tag{15}$$

where B_{predict} denotes the area of the predicted bounding box, and B_{gt} denotes the area of the ground truth bounding box. For the multiclass object detection, the mAP is also calculated, and it is the average AP of all different object classes.

C. Experiment Setups

The dataset applied in the experiments is divided into the train subset, the validation subset, and the test subset as the percentage 0.25%, 0.25%, and 0.5%, respectively. The train subset and the validation subset are together used for training in our experiments. In order to keep the generalization ability of the model with TC, training samples are with their own TC samples together to train the detection model. Eight typical and representative methods are selected as the competitive methods, such as DPM [42], Faster R-CNN [46], YOLO [38],

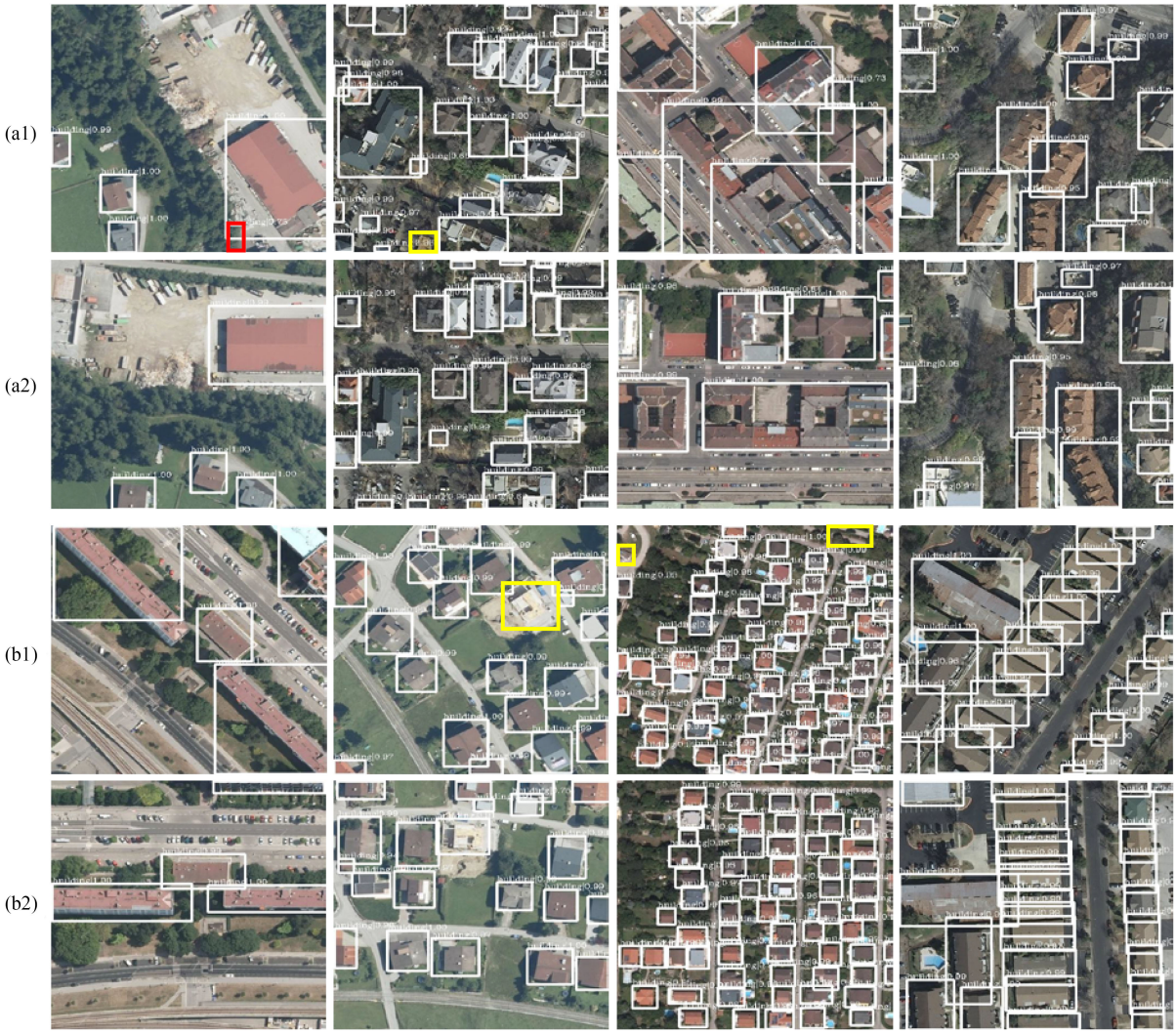

Fig. 8. Typical detection results of different scales, colors, and architectures. The results without TC are on rows (a1) and (b1), and the results with TC are on rows (a2) and (b2). The white bounding boxes are TPs, the yellow bounding boxes are FNs, and the red bounding boxes are FPs.

TABLE III
HYPERPARAMETER SETUPS OF THE TRAINING MODELS

Method	LR	Decay	Resized Scale	BS	Optimizer
Faster-RCNN [47]	0.02	0.0001	1333 × 800	16	SGD
YOLOv3-320 [38]	0.001	0.0005	320 × 320	16	SGD
YOLOv3-608 [38]	0.001	0.0005	608 × 608	16	SGD
SSD-300 [49]	0.002	0.0005	300 × 300	16	SGD
SSD-512 [49]	0.002	0.0005	512 × 512	16	SGD
RetinaNet [50]	0.01	0.0001	1333 × 800	16	SGD
CornerNet [51]	0.0005	—	511 × 511	8	Adam
CentripetalNet [53]	0.0005	—	511 × 511	8	Adam

SSD [48], CornerNet [50], CenterNet [51], FoveaBox [58], and CentripetalNet [52].

The competitive experiments are mainly performed on the platform mmdetection[7] [59]. The form of AIBD is converted into the form of COCO. The platform GPUs are with four Nvidia GeForce GTX 1080. The hyperparameter setups of the training models for the competitive methods are reported in Table III. The hyperparameter setups follow the presets of the platform mmdetection. The available and executable code version of

[7][Online]. Available: https://github.com/open-mmlab/mmdetection

TABLE IV
BUILDING DETECTION RESULTS OF COMPETITIVE METHODS USING THE COCO METRIC

Method	Backbone	TC	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
DPM [43]	—	✗	—	0.11e-02	—	—	—	—
		✓	—	0.15e-02	—	—	—	—
SSD-300 [49]	VGG-16	✗	0.423	0.803	0.408	0.309	0.462	0.487
		✓	0.435	0.823	0.418	0.324	0.466	0.515
SSD-512 [49]	VGG-16	✗	0.484	0.843	0.509	0.360	0.525	0.529
		✓	0.502	0.859	0.535	0.375	0.538	0.571
YOLOv3-608 [38]	Darknet53	✗	0.443	0.822	0.437	0.328	0.485	0.432
		✓	0.474	0.851	0.489	0.354	0.508	0.507
YOLOv3-320 [38]	Darknet53	✗	0.412	0.811	0.376	0.301	0.449	0.460
		✓	0.439	0.840	0.415	0.321	0.469	0.487
RetinaNet [50]	Resnet50	✗	0.477	0.839	0.484	0.362	0.516	0.507
		✓	0.499	0.858	0.519	0.377	0.533	0.563
CornerNet [51]	Hourgl.-104	✗	0.234	0.420	0.237	0.106	0.360	0.133
		✓	0.340	0.556	0.364	0.149	0.457	0.328
CentripetalNet [53]	Hourgl.-104	✗	0.451	0.794	0.459	0.322	0.493	0.509
		✓	0.479	0.818	0.504	0.36	0.515	0.553
Faster R-CNN [47]	Resnet50	✗	0.500	0.845	0.535	0.368	0.535	0.569
		✓	**0.515**	**0.861**	**0.548**	**0.383**	**0.545**	**0.602**
Faster R-CNN [47]	Resnet101	✗	0.502	0.846	0.539	0.364	0.539	0.585
		✓	**0.517**	**0.862**	**0.555**	**0.382**	**0.546**	**0.613**

TABLE V
RUNNING TIME ON SINGLE IMAGE OF THE COMPETITIVE METHODS

Time (sec.)	DPM [43]	SSD-300 [49]	SSD-512 [49]	YOLOv3-608 [38]	YOLOv3-320 [38]	RetinaNet [50]	CornerNet [51]	CentripetalNet [53]	Faster R-CNN [47]	Faster R-CNN [47]
Backbone	—	VGG-16	VGG-16	Darknet53	Darknet53	Resnet50	Hourgl.-104	Hourgl.-104	Resnet50	Resnet101
w/o TC	0.059	0.127	0.134	0.235	0.333	0.113	0.110	0.158	0.302	0.305
with TC	0.145	0.213	0.225	0.342	0.434	0.151	0.210	0.248	0.389	0.398

The time duration is measured in seconds.

DPM [42] is voc-release3[8] in MATLAB. Therefore, the platform of DPM is Windows 10 with 16-GB RAM and eight i7-9800 Intel Core CPUs. The available evaluation metric of DPM is only AP_{50}.

D. Experiment Results and Analysis

CoBP evaluation: In order to evaluate the effect of the CoBP for the whole dataset, a metric named CoBP is defined in Section III-E. The CoBP is calculated for both the samples with the TC algorithm and those without the TC algorithm. The tilt angle histogram is set as nine bins. The CoBP curves are shown in Fig. 5. The magenta curve is the average CoBP without the TC algorithm as the tilt angle bins on the AIBD. The cyan curve is the average CoBP with the TC algorithm as the tilt angle bins on the AIBD. We can find that the CoBP reduces a lot as the tilt degree of rotation window if the TC algorithm is applied. As the tilt angular periodicity, the tangent value V_T of the tilt building is calculated as (12). The CoBP reaches maximum when the tilt is 45°, while the CoBP reaches minimum when the tilt is 0° and 90°. The purpose of the proposed method is to reduce the CoBP so as to effectively perform building detection. It is revealed that the TC algorithm can relieve the problem of the building's oblique cutting.

However, not all of the buildings can be corrected to purposed orientation because the tilt angles are statistical values estimated by the detected lines of the images. The TC cannot be estimated if buildings do not exist in the partition boundary. Besides, the proposed method can be used to deal with the buildings of irregular shape with distinctive linear features. However, when the buildings do not have distinctive linear features, the proposed method cannot be effectively performed. For most cases, the TC can be utilized in the experiments.

Table IV shows the building detection results of competitive methods using the COCO metric. For building detection, all of the representative methods adopted frequently used backbones, such as VGG-16, Darknet-53, Resnet50, Resnet101, and Hourglass-104. Each of the methods was tested on the condition with and without the TC algorithm. We can see that all of the APs with the TC algorithm are better than those without the TC algorithm. The best APs are achieved by the Faster R-CNN [46], and the results are optimal when the backbone is Resnet101. The result values of the metric AP_{50} achieve the

[8][Online]. Available: http://www.rossgirshick.info/latent/

highest scores, among which the results of CornerNet [50] are the worst. Considering the building instance scale, the results of the small instances are as good as the medium instances and the large instances on the whole. However, the AP_{50} results of the DPM [42] are the worst. The main reason is that the robustness and generalization of the traditional DPM is limited. Therefore, the DPM cannot effectively manage the large building dataset with complex backgrounds and variable building objects. The experiment results demonstrate that the TC algorithm is beneficial to the APs of building detection.

The average running time for each image of the competitive methods is reported in Table V. The running time of the methods with TC is more than those without TC. The running results of YOLOv3-320 [38] are the slowest because the resizing process of large-size variation needs much more time. On the whole, all the running time results of the competitive methods are less than 0.5 s.

Example results of the proposed method are presented in Fig. 6. The white bounding boxes are as TPs and the yellow bounding boxes are FNs. The Faster R-CNN with resnet101 achieves the best results. The Faster R-CNN has the least FNs among its competitors. Take the irregular image in the third column as example; the bounding boxes of examples are more accurate. There are nearly no FNs of the three examples. The curves of the training error of Faster R-CNN [46] with TC as the iterations are shown in Fig. 7.

Here, we would like to discuss some unsatisfactory results. Some building examples are unfortunately marked in yellow bounding boxes as FNs and red bounding boxes as FPs, especially for YOLOv3-608 [38] and CentripetalNet [52]. There are two reasons that can explain this phenomenon. On the one hand, the scales, colors, and architectures of the building objects are extreme variations, so the building dataset possesses huge within-cluster variation. On the other hand, quite a few building boundaries are not clear in intricate backgrounds. Therefore, it is difficult for the one-stage methods and anchor-free methods to obtain superior detection results.

Some more detection results of the Faster R-CNN, whose backbone is resnet101, are shown in Fig. 8. The detection results without TC are at rows (a1) and (b1), and those results with TC are at rows (a2) and (b2). The detection rectangles of rows (a1) and (b1) have many overlaps and contain much backgrounds inner them, whereas the detection rectangles of rows (a2) and (b2) are much compact to the real building objects. It is revealed that a more robust model with less difficult testing samples can perform better and the bounding boxes of the samples without TC contain much more backgrounds and are not fully semantic building objects.

V. Conclusion

In this article, we first propose a framework for detecting objects in a large-size image instead of small-size patch, particularly for building detection. The framework mainly consists of two phases. In the first phase, a simple and effective TC algorithm is proposed to solve the problem of oblique cutting, and this method can preserve the completeness of building objects. In another aspect, the detection results will be more reasonable and comprehensive when inversely mapped to the large-size image. Building detection is performed in the second phase with object detectors, especially DNN-based methods. Besides, a new evaluation metric named CoBP is defined to evaluate the CoBP for the dataset. Moreover, a challenging AIBD is annotated for the public research. The experimental results manifest the effects of the proposed method both qualitatively and quantitatively.

References

[1] W. Liu *et al.*, "Building footprint extraction from unmanned aerial vehicle images via PRU-Net: Application to change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 14, pp. 2236–2248, 2021.

[2] S. Karimzadeh and M. Matsuoka, "Building damage characterization for the 2016 Amatrice earthquake using ascending-descending COSMO-skyMed data and topographic position index," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 11, no. 8, pp. 2668–2682, Aug. 2018.

[3] S. M. Mousavi and G. C. Beroza, "Bayesian-deep-learning estimation of earthquake location from single-station observations," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 8211–8224, Nov. 2020.

[4] Q. Wang, S. Liu, J. Chanussot, and X. Li, "Scene classification with recurrent attention of VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1155–1167, Feb. 2019.

[5] Z. Y. Lv, T. F. Liu, and J. A. Benediktsson, "Object-oriented key point vector distance for binary land cover change detection using VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6524–6533, Sep. 2020.

[6] X. Li, M. Chen, F. Nie, and Q. Wang, "A multiview-based parameter free framework for group detection," in *Proc. Conf. Artif. Intell.*, 2017, pp. 4147–4153.

[7] C. Dubois, A. Thiele, and S. Hinz, "Building detection and building parameter retrieval in InSAR phase images," *Photogramm. Eng. Remote Sens.*, vol. 114, pp. 228–241, 2016.

[8] S. Adelipour and H. Ghassemian, "Building detection in very high resolution SAR images via sparse representation over learned dictionaries," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 11, no. 12, pp. 4808–4817, Dec. 2018.

[9] A. Ferro, D. Brunner, and L. Bruzzone, "Automatic detection and reconstruction of building radar footprints from single VHR SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 935–952, Feb. 2013.

[10] M. Shahzad, M. Maurer, F. Fraundorfer, Y. Wang, and X. X. Zhu, "Buildings detection in VHR SAR images using fully convolution neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1100–1116, Feb. 2019.

[11] Z. Cai, H. C. Ma, and L. Zhang, "A building detection method based on semi-suppressed fuzzy C-means and restricted region growing using airborne LiDAR," *Remote Sens.*, vol. 11, no. 7, 2019, Art. no. 848.

[12] E. K. Dey and M. Awrangjeb, "A robust performance evaluation metric for extracted building boundaries from remote sensing data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 13, pp. 4030–4043, Jul. 2020.

[13] Y. Chen, L. Cheng, M. Li, J. Wang, L. Tong, and K. Yang, "Multiscale grid method for detection and reconstruction of building roofs from airborne LiDAR data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 7, no. 10, pp. 4081–4094, Oct. 2014.

[14] A. Mahphood and H. Arefi, "Virtual first and last pulse method for building detection from dense LiDAR point clouds," *Int. J. Remote Sens.*, vol. 41, no. 3, pp. 1067–1092, 2020.

[15] Z. Z. Zhao, H. T. Wang, C. Wang, S. T. Wang, and Y. Q. Li, "Fusing LiDAR data and aerial imagery for building detection using a vegetation-mask-based connected filter," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1299–1303, Aug. 2019.

[16] X. Li, M. Chen, F. Nie, and Q. Wang, "Locality adaptive discriminant analysis," in *Proc. Int. Joint Conf. Artif. Intell.*, 2017, pp. 2201–2207.

[17] X. Huang, W. Yuan, J. Li, and L. Zhang, "A new building extraction postprocessing framework for high-spatial-resolution remote-sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 10, no. 2, pp. 654–668, Feb. 2017.

[18] J. Li, J. Cao, M. Feyissa, and X. Yang, "Automatic building detection from very high-resolution images using multiscale morphological attribute profiles," *Remote Sens. Lett.*, vol. 11, no. 7, pp. 640–649, 2020.

[19] J. Huang, G. Xia, F. Hu, and L. Zhang, "Accurate building detection in VHR remote sensing images using geometric saliency," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 3991–3994.
[20] M. Norman, H. Shafri, M. Idrees, S. Mansor, and B. Yusuf, "Spatio-statistical optimization of image segmentation process for building footprint extraction using very high-resolution worldview 3 satellite data," *Geocarto Int.*, vol. 35, no. 10, pp. 1124–1147, 2020.
[21] L. Agarwal and K. S. Rajan, "Integrating MSER into a fast ICA approach for improving building detection accuracy," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 4831–4834.
[22] Z. Z. Karadag, A. Senaras, and F. T. Yarman-Vural, "Segmentation fusion for building detection using domain-specific information," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 8, no. 7, pp. 3305–3315, Jul. 2015.
[23] P. J. Wang, X. Sun, W. H. Diao, and K. Fu, "FMSSD: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3377–3390, May 2020.
[24] Y. Liu *et al.*, "Multilevel building detection framework in remote sensing images based on convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 11, no. 10, pp. 3688–3700, Oct. 2018.
[25] C. Y. Xu, C. Z. Li, Z. Cui, T. Zhang, and J. Yang, "Hierarchical semantic propagation for object detection in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4353–4364, Jun. 2020.
[26] F. Alidoost and H. Arefi, "A CNN-based approach for automatic building detection and recognition of roof types using a single aerial image," *J. Photogrammetry Remote Sens. Geoinf. Sci.*, vol. 86, pp. 235–248, Jan. 2018.
[27] R. Hamaguchi, K. Nemoto, T. Imaizumi, and S. Hikosaka, "Detecting buildings of any size using integration of CNN models," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 1280–1283.
[28] Z. P. Dong, M. Wang, Y. L. Wang, Y. Zhu, and Z. Q. Zhang, "Object detection in high resolution remote sensing imagery based on convolutional neural networks with suitable object scale features," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2104–2114, Mar. 2020.
[29] D. Zhu *et al.*, "Diverse sample generation with multi-branch conditional generative adversarial network for remote sensing objects detection," *Neurocomputing*, vol. 381, pp. 40–51, 2020.
[30] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.
[31] J. Yang, L. Ji, X. Geng, X. Yang, and Y. Zhao, "Building detection in high spatial resolution remote sensing imagery with the U-rotation detection network," *Int. J. Remote Sens.*, vol. 40, no. 15, pp. 6036–6058, 2019.
[32] H. Fang, Y. Wei, H. Luo, and Q. Hu, "Detection of building shadow in remote sensing imagery of urban areas with fine spatial resolution based on saturation and near-infrared information," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, vol. 12, no. 8, pp. 2695–2706, Aug. 2019.
[33] T. Bai *et al.*, "An optimized faster R-CNN method based on DRNet and RoI align for building detection in remote sensing images," *Remote Sens.*, vol. 12, no. 5, 2020, Art. no. 762.
[34] Z. Shu, X. Hu, and J. Sun, "Center-point-guided proposal generation for detection of small and dense buildings in aerial imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 7, pp. 1100–1104, Jul. 2018.
[35] X. Jiang, X. Zhang, Q. Xin, X. Xi, and P. Zhang, "Arbitrary-shaped building boundary-aware detection with pixel aggregation network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote. Sens.*, to be published, doi: 10.1109/JSTARS.2020.3017934.
[36] X. Yao, J. Han, L. Guo, S. Bu, and Z. Liu, "A coarse-to-fine model for airport detection from remote sensing images using target-oriented visual saliency and CRF," *Neurocomputing*, vol. 164, pp. 162–172, 2015.
[37] Y. Xie, J. Cai, R. Bhojwani, S. Shekhar, and J. Knight, "A locally-constrained YOLO framework for detecting small and densely-distributed building footprints," *Int. J. Geograph. Inf. Sci.*, vol. 34, no. 4, pp. 777–801, 2020.
[38] J. Redmon, S. K. Divvala, R. B. Girshick, A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 27–30, 2016, pp. 779–788.
[39] K. Reda and M. Kedzierski, "Detection, classification and boundary regularization of buildings in satellite imagery using faster edge region convolutional neural networks," *Remote Sens.*, vol. 12, no. 14, 2020, Art. no. 2240.
[40] L. Du, L. Li, D. Wei, and J. S. Mao, "Saliency-guided single shot multibox detector for target detection in SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3366–3376, May 2020.
[41] X. Li and B. Zhao, "Video distillation," *Sci. China Inf. Sci.*, 2021, doi: 10.1360/SSI-2020-0165.
[42] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[43] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 20–26, 2005, pp. 886–893.
[44] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 23–28, 2014, pp. 580–587.
[45] R. B. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 7–13, 2015, pp. 1440–1448.
[46] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[47] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
[48] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, vol. 9905, Oct. 11–14, 2016, pp. 21–37.
[49] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[50] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis.,* vol. 11218, Sep. 8–14, 2018, pp. 765–781.
[51] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 27–Nov. 2, 2019, pp. 6568–6577.
[52] Z. Dong, G. Li, Y. Liao, F. Wang, P. Ren, and C. Qian, "CentripetalNet: Pursuing high-quality keypoint pairs for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 13–19, 2020, pp. 10516–10525.
[53] B. Liu *et al.*, "A horizontal tilt correction method for ship license numbers recognition," *J. Phys.: Conf. Ser.*, vol. 976, 2018, Art. no. 012013.
[54] P. Li, M. Nguyen, and W. Q. Yan, "Rotation correction for license plate recognition," in *Proc. 4th Int. Conf. Control, Autom. Robot.*, 2018, pp. 400–404.
[55] D. Yang, H. Zhou, L. Tang, S. Chen, and S. Liu, "A license plate tilt correction algorithm based on the character median line," *Can. J. Electron. Comput. Eng.*, vol. 41, no. 3, pp. 145–150, 2018.
[56] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic Hough transform," *Comput. Vis. Image Understanding*, vol. 78, no. 1, pp. 119–137, 2000.
[57] E. Maggiori, Y. Tarablka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The Inria aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2017, pp. 3226–3229.
[58] T. Kong, F. Sun, H. Liu, Y. Jiang, and J. Shi, "FOVEABOX: Beyond anchor-based object detector," *IEEE Trans. Image Process.*, vol. 29, pp. 7389–7398, Jun. 2020.
[59] K. Chen *et al.*, "MMDetection: Open MMLAB detection toolbox and benchmark," *CoRR*, vol. abs/1906.07155, 2019.

Kang Liu received the bachelor's degree in computer science and technology from Xi'an Jiaotong University, Xi'an, China, in 2013, and the master's degree in signal and information processing in 2016 from the University of Chinese Academy of Sciences, Beijing, China, where he is currently working toward the Ph.D. degree.

He is also an Assistant Research Fellow with the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an. His research interests include machine learning, computer vision, and remote sensing.

Zhiyu Jiang received the B.E. degree in intelligent science and technology from Xidian University, Xi'an, China, in 2013, and the Ph.D. degree in signal and information processing from the University of Chinese Academy of Sciences, Beijing, China, in 2018.

He is currently an Associate Professor with the School of Artificial Intelligence, Optics and Electronics, Northwestern Polytechnical University, Xi'an, China. His current research interests include remote sensing and computer vision.

Mingliang Xu received the B.E. and M.E. degrees from Zhengzhou University, Zhengzhou, China, in 2005 and 2008, respectively, and the Ph.D. degree from the State Key Laboratory of Computer-Aided Design and Computer Graphics, Zhejiang University, Hangzhou, China, in 2012, all in computer science.

He is currently an Associate Professor with the School of Information Engineering, Zhengzhou University. His research interests include computer graphics and computer vision.

Matjaž Perc is currently a Professor of physics and Director of the Complexity Science Laboratory, University of Maribor. He is among top 1% most cited physicists according to Clarivate Analytics.

Dr. Perc is the 2015 recipient of the Young Scientist Award for Socio and Econophysics from the German Physical Society and the 2017 USERN Laureate. In 2018, he received the Zois Award, which is the highest national research award in Slovenia. He is an Editor for *Physics Letters A* and *Chaos, Solitons & Fractals*. He is on the editorial board of *New Journal of Physics*, *Proceedings of the Royal Society A*, *Journal of Complex Networks*, *Europhysics Letters*, *European Physical Journal B*, *Scientific Reports*, *Royal Society Open Science*, *Applied Mathematics and Computation*, and *Frontiers in Physics*. In 2019, he became a Fellow of the American Physical Society. He is a member of the Academia Europaea and the European Academy of Sciences and Arts.

Xuelong Li (Fellow, IEEE) is a Full Professor with the School of Artificial Intelligence, Optics and Electronics, Northwestern Polytechnical University, Xi'an, China. He was the Founder of the Shaanxi Key Laboratory of Ocean Optics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an.