Research Article

The Aircraft Pose Estimation Based on a Convolutional Neural Network

Daoyong Fu ¹, Wei Li ¹, Songchen Han ^{1,2}, Xinyan Zhang ³, Zhaohuan Zhan ¹,
and Menglong Yang ¹

¹School of Aeronautics and Astronautics, Sichuan University, Chengdu, China

²Key Laboratory of Air Traffic Control Automation System, Sichuan University, Chengdu, China

³National Key Laboratory of Fundamental Synthetic Vision, Sichuan University, Chengdu, China

Correspondence should be addressed to Wei Li; li.wei@scu.edu.cn

Received 3 June 2019; Revised 13 September 2019; Accepted 25 October 2019; Published 19 December 2019

Academic Editor: Alessandro De Luca

Copyright © 2019 Daoyong Fu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The pose estimation of the aircraft in the airport plays an important role in preventing collisions and constructing the real-time scene of the airport. However, current airport target surveillance methods regard the aircraft as a point, neglecting the importance of pose estimation. Inspired by human pose estimation, this paper presents an aircraft pose estimation method based on a convolutional neural network through reconstructing the two-dimensional skeleton of an aircraft. Firstly, the key points of an aircraft and the matching relationship are defined to design a 2D skeleton of an aircraft. Secondly, a convolutional neural network is designed to predict all key points and components of the aircraft kept in the confidence maps and the Correlation Fields, respectively. Thirdly, all key points are coarsely matched based on the matching relationship and then refined through the Correlation Fields. Finally, the 2D skeleton of an aircraft is reconstructed. To overcome the lack of benchmark dataset, the airport surveillance video and Autodesk 3ds Max are utilized to build two datasets. Experiment results show that the proposed method get better performance in terms of accuracy and efficiency compared with other related methods.

1. Introduction

Real-time surveillance of airport surface is important for safeguarding the safety and efficiency of airport operation. Conventionally, Surface Movement Radar (SMR), Multilateration (MLAT) and Automatic Dependent Surveillance-Broadcast (ADS-B) are the main methods in the airport surface surveillance. Since the aircraft on airport surface are regarded as points in these methods, poses of the aircraft are ignored. However, the poses of the aircraft not only show the occupied spaces of the aircraft in airports, but also indicate the purpose of the aircraft's movement which plays important roles in preventing collisions. Therefore, not only the location information but also the poses of the aircraft should be estimated. Video surveillance can provide the fine-grained features of the aircraft, such as colors, shapes and sizes, which could help to estimate identities, positions, and poses of the aircraft. Some studies focus on the pose

estimation of humans for surveillance, activity recognition, gaming, etc. However, there is little attention to estimate the poses of other kinds of objects due to the insufficient samples of these objects. The pose estimation heavily relies on the key points recognition based on video processing. On the airport surface, the large sizes of the aircraft, the main objects of interest, lead to the difficulties in key points recognition and connection. Besides, since the movements of objects on the airport surface are always complex, the views of the aircraft confront some occlusions in cameras' images, which further increase the difficulties in pose estimation of the aircraft.

The aircraft attitude estimation methods mainly depend on parameter measurement by various sensors on the aircraft (such as gyroscope and GPS). Boedecker [1] used multiantennae Global Navigation Satellite System (GNSS) receivers to obtain an optimal configuration in economical, operational and accuracy aspects. To use all available information, Rhudy [2] combined the control inputs with the

sensor measurements through Kalman filtering technology. Han et al. [3] used central difference Kalman filter (CDKF) based on Stirling interpolation formulation to prevent the defects of the computational complexity and large linearization error caused by extended Kalman filter (EKF). Wind gusts had a great impact on the attitude estimation accuracy of the Small Unmanned Aircraft Systems (SUASs); Weibel et al. [4] utilized Global Positioning System (GPS) velocities to estimate attitude and heading reference systems that corrected accelerometer specific-force measurements. This method had a high level of performance with contemporary low-cost sensors in gusty conditions. The accuracy of attitude estimation degraded when the unmanned aerial vehicle (UAV) was under accelerative maneuver. In order to solve this problem, No et al. [5] fused the pseudo-attitude, magnetic-attitude and gyroscope measurement based on the Euler angle. This method maintained stable attitude accuracy even if the aircraft experienced sudden or continuous acceleration. In the event of gyroscopic failure, Kallapur et al. [6] utilized on-board accelerometers and a GPS receiver to update the errors in attitude propagation. However, it is obvious that the errors of the sensor will increase over time. And these methods cannot be used to estimate the poses of the noncooperative targets.

At the same time, there are some aircraft pose estimation methods based on computer vision. In order to avoid the increase of sensor error caused by fast movement, Zhao et al. [7] found the correspond points of the same aircraft in two images through the Speeded Up Robust Features (SURF) method. Tong et al. [8] presented a method of sequence screen-spot imaging based on the laser-aided cooperative target to amplify the motion attitude of the aircraft for improving the measurement accuracy. Carrio et al. [9] proposed a method based on thermal images to solve the effect of atmospheric and illumination conditions on visible light image sensors. Tehrani et al. [10] used panoramic images and optic flow to improve accurate attitude estimation in cluttered environments. In order not to rely on the control point, Zhang et al. [11] matched the simulated image with the real image. Zhao et al. [12] built the three-dimensional (3D) geometric model of the rigid object and the camera parameters. Lu et al. [13] used object space collinearity error to replace iterative optimization methods, which did not effectively account for the orthonormal structure of rotation matrices. Teng et al. [14] used line features of a straight wing aircraft's structure and geometry constraints when feature point matching was difficult and inaccurate in the large baseline and long-distance imaging. Luo et al. [15] used line clustering to improve the accuracy of the lines. Locally Linear Embedding (LLE) in [16] was used to preserve the intrinsic structure information. Ling et al. [17] used features of shapes and regions of the aircraft to reduce the complexity of 3D model matching. Wang et al. [18] presented a novel geometry structure feature to describe the objects' structure information. Fu and Sun [19] not only extracted the targets contours, but also computed their Pseudo-Zernike Moments (PZM). It had good performances on the aspect of adjusting to the 3D target with freely changed pose. However, these methods only use the shallow

features of the aircraft. They are not suitable to the complex airport scenes. Although shallow features contain more location information, they include few deep features that can improve the generalization ability of the method. Therefore, this paper uses a convolutional neural network (CNN) [20] to extract deep features of the aircraft to estimate aircraft pose.

The CNN is a deep learning model evolved from artificial neural networks. It mimics the working mechanism of the human brain. Because of its capacity to automatically extract features, the CNN has been employed in multiple fields. Abdel-Hamid et al. [21] used a CNN to improve speech recognition performance and proposed a limited-weight-sharing scheme that could better model speech features. Experiment results showed that the proposed method could reduce the error. Lin et al. [22] proposed an improved CNN which was utilized to improve the ability of the evaluation method of the distribution network. The operation state and network structure characteristics of the power network were analysed by this CNN. And the CNN gave an optimal evaluation result. In order to enhance the security and improve the detection ability of malicious intrusion behavior in a wireless network, Yang and Wang [23] designed an improved CNN. This CNN regarded low-level intrusion traffic data of wireless network as features that were used to detect the intrusion behavior of the network. Yang et al. [24] combined the hierarchical symbolic analysis and CNN to diagnose the faults of rotating machinery. This CNN used the features of the vibration signals to evaluate health conditions of rotating machinery. Due to a lack of understanding of the underlying atrial structures, the treatment for atrial fibrillation (AF) was suboptimal. Xiong et al. [25] proposed a CNN named "AtrialNet" that was able to successfully process each 3D late gadolinium-enhanced magnetic resonance imaging (LGE-MRI) within 1 min. Because of the strong adaptability of the CNN, using the CNN to estimate the aircraft pose is feasible.

In this paper, a CNN-based pose estimation method of the aircraft on the airport surface is proposed. A 2D skeleton model is established to represent an aircraft inspired by the estimation method of human pose which generally falls into two main categories: top-down methods and bottom-up methods. Top-down [26] methods first detect all people in the image and then estimate single person pose. Bottom-up methods [27] first detect all key points in the image and then use postprocessing to match these key points. A bottom-up method is applied in this paper, which contains key-point detection and matching. A CNN architecture is designed to detect the key points and components of the aircraft existing in the image. For the key-point matching, the matching relationship and the Correlation Fields (CFs) are proposed. The matching relationship is utilized for rough matching, which restricts the matching of key points. The CFs are utilized for refined matching, which contains the information of all detected components obtained through the CNN.

The main contributions of this work are summarized as follows. (1) A 2D skeleton is designed to represent an aircraft in images. (2) A method of aircraft pose estimation is

proposed, which includes two steps, *i.e.*, key-point detection and matching.

The rest of the paper is organized as follows. Section 2 shows the process of designing aircraft skeleton. Section 3 introduces the aircraft pose estimation process including the key-point detection and matching. Section 4 shows the experimental results and the network optimal process. Finally, Section 5 gives the conclusion.

2. Aircraft Skeleton Design

The cameras project the 3D objects into 2D images, which lead to the loss of the depth information. In order to design the aircraft 2D skeleton, the key points of an aircraft must be selected, which could provide the shape information as much as possible. The correlations between these key points are also important to indicate the occupied space of an aircraft. In this study, two principles to define the key points of an aircraft are proposed. (1) The selected key points should include main end points, junction points and inflection points of an aircraft. (2) The selected key points should be easily recognized.

It is obvious that the connection between the fuselage and the wing is surface contact rather than point contact. Thus, the center point of the connecting surface is used to denote this connecting surface. According to the first principle, an aircraft nose end point, two center points of the connecting surfaces between the fuselage and two wings (hereinafter point 2 and 4), two wing end points, two center points of the connecting surfaces between the fuselage and two horizontal empennages, two horizontal empennage end points, the center point of the connecting surface between the fuselage and the vertical empennage (hereinafter point 10), a vertical empennage end point and a tail end point are deemed as the key points of an aircraft as shown in Figure 1(a). It is shown that the key points of an aircraft tail have low recognition accuracy when the sizes of the aircraft in the image are small or the plane is facing the camera, *e.g.*, the tail end point, the vertical empennage end point and the point 10. According to the second principle, these points are ignored. The remaining key points of an aircraft are shown in Figure 1(b). In order to facilitate the description of the position of the aircraft, an aircraft center point has been added (shown as the green point in Figure 1(c)), which is also the midpoint of point 2 and 4. Thus, there are $N=10$ types of key points.

The correlations between key points called the matching relationship are important for indicating the occupied space of an aircraft. The matching relationship is described by the connections of key points. There are some principles to define the matching relationship. (1) The connection of two key points is on the airframe and keeps away from the airframe contour. (2) Each key point is connected at least once. According to the principles above, the matching relationship that divides an aircraft into $M=9$ components as shown in Figure 1(d). Each connection represents one type of the components. The matching relationship indicates which two types of key points can match. It also indicates the direction that points

from one end of each component to the other. The matching relationship can be expressed as follows:

$$\mathbf{B} = \{a_{n_1} \longrightarrow a_{n_2} \mid n_1, n_2 \in \{1, 2, \dots, N\}\}, \quad (1)$$

where a_{n_1} and a_{n_2} denote, respectively, the n_1^{th} and n_2^{th} type of key points. If a_{n_1} has a matching relationship with a_{n_2} . At the same time, a_{n_1} and a_{n_2} are, respectively, the start and end of the component defined by these two types of key points, $a_{n_1} \longrightarrow a_{n_2} = 1$. Otherwise, $a_{n_1} \longrightarrow a_{n_2} = 0$.

3. Aircraft Pose Estimation

3.1. Overview. In order to estimate aircraft pose, an aircraft pose estimation method based on the CNN is proposed, which mainly contains two steps: the key points and CFs detection and the key-point matching. In the first step, a CNN is established to detect all key points and components of the aircraft in the image. This CNN simultaneously generates N detection confidence maps for the key points (Figure 2(b)) and M detection CFs for the components (Figure 2(c)). In the second step, the matching relationship is applied to match all key points in N confidence maps (Figure 2(d)), which is called rough matching. Then, the results of the rough matching are further refined through the CFs (Figure 2(e)), which is called refined matching. Finally, the 2D skeletons of all aircrafts are obtained from the refined results (Figure 2(f)).

3.2. Step 1: The Key Points and CFs Detection

3.2.1. Network Architecture. According to the goals of the detection, the CNN shown in Figure 3 contains two detection tasks: to detect the key points and CFs. Thus, the proposed network includes two branches: Branch 1, to predict the key points (the dark green area in Figure 3) and Branch 2, to generate the CFs (the yellow area in Figure 3).

The Visual Geometry Group Network (VGGNet) [28] is utilized as the backbone network. An input image is processed by Part 1 of the proposed network, which will generate a set of feature maps F as the input of Part 2. The two branches of Part 2 (shown in brown box in Figure 3) are for the prediction of the key points and CFs, respectively. Both two branches include three stages. Each stage consists of convolution layers. In the first stage of Branch 1, a set of key point confidence maps is predicted. In each subsequent stage of Branch 1, the predictions of Branch 1 in the previous stage, along with feature maps F , are used to produce refined predictions. Branch 2 is similar to Branch 1. The only difference between them is that Branch 1 is used to predict key points, while Branch 2 is used to predict the CFs.

3.2.2. Training and Testing

(1) Training Phase. In this section, the training method of the proposed CNN is introduced. The CNN consists of two

FIGURE 1: Simplification process of the aircraft 2D skeleton. (a) Candidate key points. (b) Ignore key points. (c) Add aircraft center point. (d) 2D skeleton.

FIGURE 2: Over pipeline.

FIGURE 3: Architecture of the CNN.

parts for extracting features and predicting the key points and CFs, respectively. In order to train the proposed CNN, the squared hinge loss (L_2 loss) function is utilized. Suppose there are $T=3$ stages in Part 2 of the proposed CNN. At each stage, the loss function is made up of two parts:

$$\begin{cases} f_D^t = \sum_{n=1}^N \sum \mathbf{W}(A) \cdot \|\mathbf{D}_n^t(A) - \mathbf{D}_n^*(A)\|_2^2, \\ f_S^t = \sum_{m=1}^M \sum \mathbf{W}(A) \cdot \|\mathbf{S}_m^t(A) - \mathbf{S}_m^*(A)\|_2^2, \end{cases} \quad (2)$$

where the loss f_D^t is in the end of the t^{th} stage of Branch 1 and the loss f_S^t is in the end of the t^{th} stage of Branch 2, $\mathbf{D}_n^t(A)$ denotes the value at location $A \in \{(x, y) | x \in [0, w], y \in [0, h], x \text{ and } y \in \mathbb{Z}\}$ in the detection confidence maps for the n^{th} type of key points, $\mathbf{S}_m^t(A)$ denotes the value at location A in the detection CFs for the m^{th} type of components, and $\mathbf{D}_n^*(.)$ and $\mathbf{S}_m^*(.)$ denote the groundtruth. $\mathbf{W}(A) = 0$ when the label is missing at location A in the image; otherwise, $\mathbf{W}(A) = 1$. The overall loss function is

$$f = \sum_{t=1}^T (f_D^t + f_S^t). \quad (3)$$

(2) *Testing Phase.* During testing, the proposed CNN simultaneously generates N detection confidence maps and M detection CFs. N detection confidence maps include type and location information of detected key points. M detection CFs include direction and location of detected components. The information of M detection CFs would be applied for key-point matching.

For example, a set of detection confidence maps $\mathbf{D}$ and a set of detection CFs $\mathbf{S}$ are obtained after an input image is analysed by the network. The set $\mathbf{D} = (\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_N)$ has N detection confidence maps, each one for the corresponding type of the key points, where $\mathbf{D}_n \in \mathbb{R}^{w \times h}$, $n \in \{1, 2, \dots, N\}$. Each element in $\mathbf{D}_n$ represents the extent of confidence that the key points occurs at a certain designated location (Figure 2(b)). The set $\mathbf{S} = (\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_M)$ has M detection CFs, each one for the corresponding type of the component, where $\mathbf{S}_m \in \mathbb{R}^{w \times h \times 2}$, $m \in \{1, 2, \dots, M\}$. Each element in $\mathbf{S}_m$ encodes a 2D unit vector or a zero vector (Figure 2(c)). The 2D unit vector indicates that this element belongs to the m^{th} type of the components of the aircraft. At the same time, the direction of 2D unit vector encodes the direction that points from one end of the components to the other end. The zero vector indicates that this element belongs to the background. For detection confidence maps, the peak values higher than the threshold in $\mathbf{D}_n$ are chosen, and the locations of peak values are the positions of corresponding key points. Many peak values in one detection confidence map represent that there are multiple key points of the same type existing in the image. Finally, a set of key point detection candidates $A_N = \{a_n^j : \text{for } j \in \{1, \dots, J_n\}, n \in \{1, \dots, N\}\}$ could be obtained, where J_n represents the number of the detected n^{th} type of key points and a_n^j represents the j^{th} candidate of the detected n^{th} type of the key points. For the application of detection CFs, see Section 3.3.

3.2.3. *The Groundtruth.* As for $\mathbf{D}_{n,k}^*$, individual confidence map $\mathbf{D}_{n,k}^*$ for the n^{th} type of the key points of the k^{th} aircraft is generated. $G_{n,k} \in \{(x, y) | x \in [0, w], y \in [0, h], x \text{ and } y \in \mathbb{Z}\}$ denotes the groundtruth position of the n^{th} type of key point of the k^{th} aircraft in the image. Therefore, the value at location A in $\mathbf{D}_{n,k}^*$ is defined as follows:

$$\mathbf{D}_{n,k}^*(A) = \exp\left(\frac{-\|A - G_{n,k}\|_2^2}{\sigma^2}\right), \quad (4)$$

where σ is the radius of the key points. The higher the value of $\mathbf{D}_{n,k}^*$ is, the closer the point A is to the groundtruth position. The value at location A in groundtruth confidence map is the maximum of an aggregation of the values at the same location in all individual confidence maps:

$$\mathbf{D}_n^*(A) = \max_k \mathbf{D}_{n,k}^*(A). \quad (5)$$

For $\mathbf{S}_m^*$, the correlation vector fields $\mathbf{S}_{m,k}^*$ is defined as follows:

$$\mathbf{S}_{m,k}^*(A) = \begin{cases} \mathbf{v}, & \text{if } A \text{ is on the } m^{\text{th}} \text{ component of the } k^{\text{th}} \text{ aircraft,} \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Consider the right wing of one aircraft (Figure 4). Points $P_{n_1,k}$ and $P_{n_2,k}$, respectively, represent the groundtruth location of the n_1^{th} and n_2^{th} type of key points of the k^{th} aircraft in the image with a matching relationship between each other. $\mathbf{v} = (P_{n_2,k} - P_{n_1,k}) / \|P_{n_1,k} - P_{n_2,k}\|_2$ (equation (6)) is a unit vector whose direction is from $P_{n_1,k}$ to $P_{n_2,k}$. If point A satisfies equation (7), the point A belongs to this component and $\mathbf{S}_{m,k}^*(A) = \mathbf{v}$; otherwise, $\mathbf{S}_{m,k}^*(A) = 0$:

$$\begin{cases} 0 \leq \mathbf{v} \cdot (A - P_{n_1,k}) \leq l_{n_2,k}, \\ 0 \leq |\mathbf{v}_\perp \cdot (P_{n_2,k} - A)| - \delta_{l_1} \leq |\tan \theta \cdot \mathbf{v} \cdot (P_{n_2,k} - A)| \text{ or} \\ |\mathbf{v}_\perp \cdot (P_{n_2,k} - A)| \leq \delta_{l_2}, \end{cases} \quad (7)$$

where $l_{n_2,k} = \|P_{n_1,k} - P_{n_2,k}\|_2$ is the distance between points $P_{n_1,k}$ and $P_{n_2,k}$, $\mathbf{v}_\perp$ is a vector perpendicular to $\mathbf{v}$, $\theta = \tan^{-1}((\delta_{l_1} - \delta_{l_2}) / l_{n_2,k})$, and δ_{l_1} and δ_{l_2} are the thresholds. During the training, $\delta_{l_1} = 1$ and $\delta_{l_2} = 0.5$.

The groundtruth CFs $\mathbf{S}_m^*$ is the average value of all $\mathbf{S}_{m,k}^*$. The average operation in order to solve the aircraft overlap caused by shooting angle is

$$\mathbf{S}_m^*(A) = \frac{1}{\text{num}_1(A)} \sum_k \mathbf{S}_{m,k}^*(A), \quad (8)$$

where $\text{num}_1(A)$ is the number of nonzero vectors of all aircrafts at location A .

Equation (7) can be applied to the other components of the aircraft.

3.3. *Step 2: The Key-Point Matching.* A set of key-point detection candidates A_N is obtained through the CNN (e.g., the points in Figure 5(a)). However, it is a problem to quickly and accurately match them. Matching any two key-point candidates will end up with $C_L^2 = L! / [2!(L-2)!]$ possible results (e.g., the black lines in Figure 5(b)), where L denotes the number of all key-point candidates in A_N . If all C_L^2 possible results are needed to determine whether each one is correct, this will cause a significant computational cost. In order to decrease the runtime of postprocessing, the matching process is divided into two steps: rough matching and refined matching. The matching relationship is used in rough matching to decrease the number of possible results to

FIGURE 4: Judgment of the correlation fields.

achieve fast refined matching. Based on the results of rough matching, the CFs are applied in refined matching to get the best key-point matching results.

3.3.1. Rough Matching Using the Matching Relationship.

In the rough matching step, the matching relationship determines which two types of key points can be matched. For example, the aircraft nose end point could match aircraft center point, but it could not match any horizontal empennage end point. Figure 5(c) shows the results of rough matching. Compared to Figure 5(b), the number of possible results is significantly reduced, achieving the goal of decreasing the runtime. After rough matching, all possible results are regarded as rough component candidates.

3.3.2. Refined Matching Using the CFs.

In this section, the method of removing the wrong components from rough component candidates is introduced. Figure 6 shows the rough matching results of one detected aircraft center point (the green point) and three detected aircraft nose end points (the red points). It is obvious that the rough component candidate represented by the yellow arrow is correct and two blue arrows are wrong. The reason for this is that the boundary isolation of an aircraft is not considered. As for this question, the CFs is proposed, which utilizes the pixel continuity of the same aircraft and the boundary isolation between the aircrafts and the airport facilities. In other words, if rough matching candidates do not overlap completely with the aircraft in the picture (e.g., the blue arrows in Figure 6), they are wrong. The CFs include all information of components of the aircraft in the image. In this case, determine whether the component is correct by calculating the degree of overlap between the rough component candidates and the CFs. Refined matching denotes this process, which includes two steps: scoring and removing.

(1) *Scoring for Rough Component Candidates.* In order to remove wrong component candidates, all rough component candidates are scored by judging how much they overlap with the CFs. For example, there are two key-point candidate locations x_{n_1} and x_{n_2} , giving a rough component candidate for the m^{th} type of the components. And the S_m is offered. The correlation confidence E for this rough component candidate is measured by computing the line integral (equation (9)) along the line segment:

$$E = \int_0^1 S_m(P(u)) \cdot \frac{x_{n_2} - x_{n_1}}{\|x_{n_2} - x_{n_1}\|_2} du, \quad (9)$$

where $P(u) = (1-u)x_{n_1} + ux_{n_2}$ interpolates the position of x_{n_1} and x_{n_2} , $S_m(P(u))$ which along line segments $x_{n_1}x_{n_2}$ is the vector at location $P(u)$ in S_m . The higher the confidence, the more likely this rough component candidate is correct.

(2) *Removing Wrong Component Candidates.* After scoring for each rough component candidate, a maximum weight bipartite graph matching method [29] is performed to remove all wrong component candidates. It is obvious that the key-point matching can be divided into multiple bipartite graph matching steps (Figure 5(d)). Specifically, only the matching of the n_1^{th} and n_2^{th} type of the key points would be considered. A_{n_1} and A_{n_2} are, respectively, the sets of two types of key-point detection candidates. In this bipartite graph matching problem, key-point detection candidates are the nodes of the graph and all rough component candidates are the edges. Additionally, the edges are weighted by equation (9). In order to guarantee that all edges cannot share the same node, $Z = \{z_{n_1 n_2}^{kl} \mid k \in \{1, \dots, J_{n_1}\}, l \in \{1, \dots, J_{n_2}\}, n_1, n_2 \in \{1, \dots, N\}\}$ is defined, where $z_{n_1 n_2}^{kl} \in \{0, 1\}$ that determines whether $a_{n_1}^k$ and $a_{n_2}^l$ have been matched satisfies equation (10). The goal is to find the maximum weigh matching (equation (11)):

$$\begin{cases} \sum_{a_{n_2}^l \in A_{n_2}} z_{n_1 n_2}^{kl} \leq 1, & \forall a_{n_1}^k \in A_{n_1}, \\ \sum_{a_{n_1}^k \in A_{n_1}} z_{n_1 n_2}^{kl} \leq 1, & \forall a_{n_2}^l \in A_{n_2}, \end{cases} \quad (10)$$

$$\max_{Z_m} E_m = \max_{Z_m} \sum_{k \in \{1, \dots, J_{n_1}\}} \sum_{l \in \{1, \dots, J_{n_2}\}} E_{n_1 n_2}^{kl} z_{n_1 n_2}^{kl}, \quad (11)$$

where Z_m is the subset of Z for the m^{th} type of the components, $E_{n_1 n_2}^{kl}$ is the weight between $a_{n_1}^k$ and $a_{n_2}^l$, and E_m is the overall weight for the m^{th} type of the components. The Hungarian algorithm [30] is utilized to obtain the optimal matching. With these steps, the inaccurate component candidates could be removed and the refined component candidates are obtained. If two refined component candidates share the same key point, they would belong to the same aircraft. The airframe structure would be reconstructed through the refined component candidates that share the same key point.

4. Experiments

4.1. Experimental Setting

4.1.1. *Dataset.* Since there is no available dataset for the pose estimation of the aircraft, Autodesk 3ds Max is used to simulate the airport environment to obtain enough

FIGURE 5: Key-point matching strategies.

FIGURE 6: The rough matching results of one aircraft center point and three aircraft nose end points.

training and testing samples. To avoid manual annotation errors, ten types of key points on the 3D model of the aircraft were marked and the positions of them in the picture were derived. All images makeup a set called 3dmax dataset.

The images extracted from the airport surveillance videos are labelled manually to build the video dataset that is used to prove the reliability of the proposed method. The video dataset mainly includes the clips such as the aircraft entering and leaving the terminal building or the apron.

4.1.2. Evaluation Metric. The Microsoft Common Objects in Context (MS COCO) key-point evaluation method [31] are utilized to evaluate the proposed method. The key-point evaluation method describes the Object Key-point Similarity (OKS) that is defined by equation (12) and uses the mean Average Precision (AP) over 10 OKS thresholds as *main competition metric*:

$$\text{OKS} = \frac{\sum_i \exp\left(-\left(d_i^2 / (2s^2 k_i^2)\right)\right) \delta(v_i > 0)}{\sum_i \delta(v_i > 0)}, \quad (12)$$

where d_i is Euclidean distance between the predicted key points and the groundtruth, $s k_i$ is the standard deviation, and δ indicates that whether the key points are labelled.

4.2. Comparison to the State of the Art. In order to compare the performance of the proposed method, OpenPose [32] is utilized as the experimental comparison.

TABLE 1: The results on 3dmax dataset.

Methods	AP	AP ⁵⁰	AP ⁷⁵	AP ^M	AP ^L	FPS
The proposed method	58.5	70.1	59.0	11.8	82.1	18.5
OpenPose	57.0	68.6	55.6	8.7	81.6	10.9

TABLE 2: The results on video dataset.

Methods	AP	AP ⁵⁰	AP ⁷⁵	AP ^M	AP ^L	FPS
The proposed method	53.1	88.3	50.7	34.6	91.5	17.2
OpenPose	49.8	72.0	47.1	28.9	91.3	10.4

The overall performance is shown in Tables 1 and 2, where AP is the mean average precision (0.50:0.05:0.95AP), AP⁵⁰ is for OKS = 0.5, AP^M is for medium scale aircrafts, and AP^L is for large scale aircrafts. It can be seen from the experimental results on 3dmax dataset, the AP of the proposed method has 1.5% higher than that of OpenPose. Meanwhile, the frames per second (FPS) of the proposed network is also faster. It can also be observed from Table 2 that the proposed method has a 3.3% higher AP value tested on video dataset. For smaller targets (AP^M), it has a higher recognition rate. The proposed method outperforms OpenPose. The reason is that the network in OpenPose is optimized. Some experimental results of the proposed method are shown in Figure 7.

4.3. Ablation Study. In this section, evaluation of the method's performance is conducted by progressively integrating network architecture and hyperparameter optimization.

4.3.1. Network Architecture Optimization. The proposed network is inspired by OpenPose. In order to improve efficiency of the proposed method when keeping the similar accuracy, the network in OpenPose is optimized. Network optimization includes two parts, i.e., network simplification and network input strategy optimization. Network simplification means that a shrunk and simplified network compared with OpenPose is applied to predict key points, and network input optimization means that the input features are reduced at each stage (Figure 8).

FIGURE 7: Results in 3dmax dataset and video dataset.

FIGURE 8: The optimization of the input strategy.

Firstly, experiments on three network optimization architecture are conducted, i.e., OpenPose_3, architecture variation_1, and architecture variation_2. Compared with OpenPose, OpenPose_3 reduces the number of stages from six to three. Architecture variation_1 adds extra convolution layers and shallow features based on OpenPose_3 inspired by [33]. In the architecture variation_2, one extra stage is added to use shallow features, which is in parallel with OpenPose_3. According to Table 3, OpenPose_3 has the similar AP to OpenPose, but faster. It illustrates that a shallower network is more suitable to aircraft pose estimation since aircraft pose estimation is changeless than human pose estimation, and 3dmax dataset is smaller than the COCO dataset. If a network is too deep and trained on a small dataset, there will be a degradation problem [34]. With the network depth increasing, accuracy gets saturated and then degrades rapidly due to the problem of vanishing/exploding gradients. The architecture variation_1 and variation_2 have lower AP and slower OpenPose_3 when combine with shallow features. This is mainly because shallow features of conv3_4 are too low to be used to predict.

TABLE 3: The results of different architectures on 3dmax dataset.

Methods	AP	AP ⁵⁰	AP ⁷⁵	AP ^M	AP ^L	FPS
OpenPose	57.0	68.6	55.6	8.7	81.6	10.9
OpenPose_3	57.1	65.1	59.2	9.1	81.1	18.1
Architecture variation_1	53.7	64.9	53.8	8.5	76.7	16.7
Architecture variation_2	53.4	62.0	54.1	4.7	78.2	11.3
Architecture variation_3	57.0	66.1	58.0	8.3	81.5	18.5

Secondly, the network input optimization is applied to OpenPose_3. This experiment is called architecture variation_3. According to Table 3, architecture variation_3 has similar AP to OpenPose_3, but faster. This is because input optimization reduces the number of convolution layer channels, i.e., the number of the input features.

An interesting aspect in Table 3 is that architecture variation_3 (the proposed architecture) performs worse than OpenPose_3 for AP⁷⁵ and performs for AP⁵⁰. To further explore this phenomenon, the average precision is plotted as a function of OKS values in Figure 9. And

FIGURE 9: Average precision at different OKS values.

Figure 9 shows that architecture variation_3 is about the average of the OpenPose and OpenPose_3. Compared with OpenPose_3, OpenPose predicts more true positive key points at low OKS values (≤ 0.65). The reason is that the impact of the degradation problem is not obvious at low OKS values. However, at high OKS values (≥ 0.7), the impact of the degradation problem is gradually reflected, which leads to lower AP values than OpenPose_3. As a result, OpenPose has a similar AP to OpenPose_3. Compared to OpenPose_3, architecture variation_3 has higher AP at low OKS values (≤ 0.65) when the features are reduced. The reason is that OpenPose_3 combines the feature maps F with the prediction of Branch 1 and 2 as the input of next stage. However, Branch 1 and 2 have different tasks, indicating that the feature fusion of Branch 1 and 2 not only does not necessarily contribute to their tasks, but also has the opposite effect. Architecture variation_3 avoids this problem. However, architecture variation_3 has lower AP at high OKS (≥ 0.7) than OpenPose_3. This is because more features in OpenPose_3 are helpful to improve the accuracy at high OKS, but not obvious at low OKS.

4.3.2. Optimal Hyperparameters. Based on architecture variation_3, the results of the hyperparameter tuning experiments are presented in this section to study the effect the radius of the key points σ . Through the analysis, it is found that hyperparameter σ makes an important role in aircraft pose estimation. Thus, several experiments are carried out to measure the AP with different radii, ranging from $\sigma = 3.5$ to $\sigma = 21$. Table 4 shows results. The best results are obtained for $\sigma = 12$, and using smaller or larger values seems to decrease performance. The reason is that a smaller value provides less feature information to predict the key points. On the other hand, using a bigger value may increase the chance for mistakes in prediction around ground facilities, such as the air bridge.

Table 5 shows the whole ablation study. First, the number of the stages is reduced from six to three. As a result, runtime decreases significantly while the AP increases 0.1%. Secondly, the new input strategy is applied, therefore making

TABLE 4: Average precision at different values.

σ	3.5	7.0	10.0	12.0	14.0	17.5	21
AP	36.8	57.0	57.6	58.5	56.8	54.8	52.6

TABLE 5: The effectiveness of various designs.

Methods	AP	Stage number	Input strategy	σ	FPS
OpenPose	57.0	6		7.0	10.9
OpenPose_3	57.1	3		7.0	18.1
3 stages + input strategy	57.0	3	✓	7.0	18.5
The proposed method	58.5	3	✓	12.0	18.5

runtime further reduced and AP decreased only by 0.1%. Finally, the AP improves 1.5% because of the hyperparameter optimization. Through network optimization, the proposed method gets better performance on accuracy and efficiency.

5. Conclusion

In this paper, a CNN-based aircraft pose estimation method is proposed. This method exploits aircraft key points to generate the predesigned aircraft 2D skeleton, which include two steps, i.e., key-point detection and matching. A CNN is designed to produce the confidence maps and the CFs which can provide the information of the key points and components existing in the image, respectively. The matching relationship and the CFs are proposed to match key points quickly and accurately. Reconstructing the aircraft 2D skeleton is finished through linking the components sharing the same key-point. Two datasets are built to evaluate the proposed method. Several experiments are conducted to validate the effect of network optimization including network architecture and hyperparameters optimization. Compared to OpenPose, the proposed method gets higher accuracy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

DYF made the main contributions to the conception and algorithm's design, as well as drafting the article. WL provided significant revising for important intellectual content and gave final approval of the current version to be submitted. SCH conceived the study, supervised the work, and helped to draft the manuscript. ZHZ, XYZ, and MLY provided the technical advices and checked the manuscript. All authors read and approved the final manuscript.

Acknowledgments

This work was supported by the National Key R&D Program of China (grant no. 2018YFC0809500) and the National Natural Science Foundation of China (grant no. U1933134).

References

- [1] G. Boedecker, "Precision aircraft attitude determination with multi-antennae GPS receivers," *Gyroscopy and Navigation*, vol. 1, no. 4, pp. 285–290, 2010.
- [2] M. Rhudy, "A dynamic model-aided sensor fusion approach to aircraft attitude estimation," in *Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1406–1409, Boston, MA, USA, August 2017.
- [3] P. Han, H. Gan, W. He et al., "Aircraft attitude estimation based on central difference Kalman filter," in *Proceedings of the 2012 IEEE 11th International Conference on Signal Processing*, vol. 1, pp. 294–298, Beijing, China, October 2012.
- [4] D. Weibel, D. Lawrence, and S. Palo, "Small unmanned aerial system attitude estimation for flight in wind," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 7, pp. 1300–1305, 2015.
- [5] H. No, A. Cho, and C. Kee, "Attitude estimation method for small uav under accelerative environment," *GPS Solutions*, vol. 19, no. 3, pp. 343–355, 2015.
- [6] A. G. Kallapur, I. R. Petersen, and S. G. Anavatti, "Robust gyro-free attitude estimation for a small fixed-wing unmanned aerial Vehicle," *Asian Journal of Control*, vol. 14, no. 6, pp. 1484–1495, 2012.
- [7] Y. Zhao, G. Zhang, B. Hu, and B. Peng, "Aircraft relative attitude measurement based on binocular vision," in *Proceedings of the AOPC 2017: Optical Sensing and Imaging Technology and Applications*, vol. 10462, Beijing, China, October 2017.
- [8] Q. Tong, Q. Yuan, Y. Zhao et al., "A method for measurement of aircraft attitude parameters based on sequence screen-spot imaging," *IEEE Access*, vol. 6, pp. 41566–41577, 2018.
- [9] A. Carrio, H. Bavle, and P. Campoy, "Attitude estimation using horizon detection in thermal images," *International Journal of Micro Air Vehicles*, vol. 10, no. 4, pp. 352–361, 2018.
- [10] M. H. Tehrani, M. A. Garratt, and S. G. Anavatti, "Low-altitude horizon-based aircraft attitude estimation using UV-filtered panoramic images and optic flow," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 5, pp. 2362–2375, 2016.
- [11] Z. Zhang, G. Su, J. Zhang, and S. Zheng, "Airplane pose measurement from image sequences," *Editorial Board of Geomatics and Information Science of Wuhan University*, vol. 29, no. 4, pp. 287–291, 2004.
- [12] L. Zhao, S. Zheng, X. Wang, and X. Huang, "Rigid object position and orientation measurement based on monocular sequence," *Journal of Zhejiang University (Engineering Science)*, vol. 52, no. 12, pp. 2372–2381, 2018.
- [13] C.-P. Lu, G. D. Hager, and E. Mjolsness, "Fast and globally convergent pose estimation from video images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 610–622, 2000.
- [14] X. Teng, Q. Yu, J. Luo, X. Zhang, and G. Wang, "Pose estimation for straight wing aircraft based on consistent line clustering and planes intersection," *Sensors*, vol. 19, no. 2, p. 342, 2019.
- [15] J. Luo, X. Teng, X. Zhang, and L. Zhong, "Structure extraction of straight wing aircraft using consistent line clustering," in *Proceedings Of the 2017 2nd International Conference On Image, Vision And Computing (ICIVC)*, pp. 168–172, Chengdu, China, June 2017.
- [16] W. Yuan, P. Jia, L. Wang, and L. Shao, "Aircraft pose recognition using locally linear embedding," in *Proceedings of the 2009 International Conference on Measuring Technology and Mechatronics Automation*, vol. 3, pp. 454–457, Hunan, China, April 2009.
- [17] W. Ling, X. Chao, and Y. Jie, "Aircraft pose estimation based on mathematical morphological algorithm and Radon transform," in *Proceedings of the 2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, vol. 3, pp. 1920–1924, Shanghai, China, July 2011.
- [18] X. Wang, H. Yu, and D. Feng, "Pose estimation in runway end safety area using geometry structure features," *The Aeronautical Journal*, vol. 120, no. 1226, pp. 675–691, 2016.
- [19] T. Fu and X. Sun, "The relative pose estimation of aircraft based on contour model," in *Proceedings of the International Conference on Optical and Photonics Engineering (icOPEN 2016)*, vol. 10250, Chengdu, China, February 2017.
- [20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 2, pp. 1097–1105, 2012.
- [21] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.
- [22] J. Liu, Y. Zhang, T. Zhao et al., "Structure strength assessment method of distribution network based on improved convolution neural network and network topology feature mining," *Proceedings of the Chinese Society of Electrical Engineering*, vol. 39, no. 1, pp. 84–96, 2019.
- [23] H. Yang and F. Wang, "Wireless network intrusion detection based on improved convolutional neural network," *IEEE Access*, vol. 7, pp. 64366–64374, 2019.
- [24] Y. Yang, H. Zheng, Y. Li et al., "A fault diagnosis scheme for rotating machinery using hierarchical symbolic analysis and convolutional neural network," *ISA Transactions*, vol. 91, pp. 235–252, 2019.
- [25] Z. Xu, V. V. Fedorov, X. Fu, E. Cheng, R. Macleod, and J. Zhao, "Fully automatic left atrium segmentation from late gadolinium enhanced magnetic resonance imaging using a dual fully convolutional neural network," *IEEE Transactions on Medical Imaging*, vol. 38, no. 2, pp. 515–524, 2019.
- [26] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik, "Using k-poselets for detecting people and localizing their keypoints," in *Proceedings of the 27th IEEE Conference on Computer Vision And Pattern Recognition, CVPR 2014*, pp. 3582–3589, Columbus, OH, USA, June 2014.
- [27] M. Zhao, T. Li, M. A. Alsheikh, Mohammad, Y. Tian et al., "Through-wall human pose estimation using radio signals," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2018*, pp. 7356–7365, Salt Lake City, UT, USA, June 2018.
- [28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, <https://arxiv.org/abs/1409.1556>.
- [29] D. B. West, *Introduction to graph theory*, Vol. 2, Prentice hall, Upper Saddle River, NJ, USA, 2001.

- [30] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
- [31] T. Y. Lin, M. Maire, S. Belongie et al., "Microsoft COCO: Common objects in context," in *Proceedings of the 13th European Conference on Computer Vision, ECCV 2014*, pp. 740–755, Zurich, Switzerland, September 2014.
- [32] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 1302–1310, Honolulu, HI, USA, July 2017.
- [33] P. Qin, C. Li, J. Chen, and R. Chai, "Research on improved algorithm of object detection based on feature pyramid," *Multimedia Tools and Applications*, vol. 78, no. 78, pp. 913–927, 2018.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, Las Vegas, Nev, USA, July 2016.