IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 16, 2020, accepted September 7, 2020, date of publication September 23, 2020, date of current version October 14, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3026209*

# 2D-Key-Points-Localization-Driven 3D Aircraft Pose Estimation

**YIBO LI[1], RUIXING YU[1], AND BING ZHU[2]**
[1]School of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China
[2]School of Electronic Engineering, Xi'an Shiyou University, Xi'an 710065, China

Corresponding author: Ruixing Yu (yrxgigi@nwpu.edu.cn)

This work was supported in part by the National Nature Science Foundation of China under Grant 61101191, in part by the Shaanxi Key Research and Development Plan under Grant 2020GY-047, and in part by the Special Scientific Research Project of Shaanxi Provincial Department of Education under Grant 17JK0599.

**ABSTRACT** In this paper, we are interesting in inferring 3D pose estimation of aircraft object leveraging 2D key-points localization. Monocular vision based pose estimation for aircraft can be widely utilized in airspace tasks like flight control system, air traffic management, autonomous navigation and air defense system. Nonetheless, prior methods using directly regression or classification can not meet the requirements of high precision in aircraft pose estimation context, other approaches using PnP algorithms that need additional information such as template 3D model or depth as prior knowledge. These methods do not exploit to full advantage the correlation information between 2D key-points and 3D pose. In this paper, we present a multi-branch network, named AirPose network, using convolutional neural network to address 3D pose estimation based on 2D key-points information. In the meantime, a novel feature fusion method is explored to enable orientation estimation branch adequately exploit key-points information. Our feature fusion method significantly decreases 3D pose estimation error also avoids the involvement of RANSAC based PnP algorithms. To address the problem that there is no available dedicated aircraft 3D pose dataset for training and testing, we build a visual simulation platform on Unreal Engine 4 applying domain randomization (DR) skill, named AKO platform, which generates aircraft images automatically labeled with 3D orientation and key-points location. The dataset is called AKO dataset. We implement a series of ablation experiments to evaluate our framework for aircraft object detection, key-points localization and orientation estimation on AKO dataset. Experiments show that our proposed AirPose network leveraging AKO dataset can achieve convincing results for each of the tasks.

**INDEX TERMS** Object pose estimation, orientation estimation, keypoints localization, feature fusion, data generation.

## I. INTRODUCTION

3D Pose estimation of aircraft object is a challenging problem facilitated by the well-developed aircraft detection algorithms in very recent years [1]–[5]. As a higher level task based on aircraft detection, 3D aircraft pose estimation can be widely utilized in many airspace tasks, such as vision-based flight control system [6], [7], air traffic management [8], autonomous navigation and air defense system. Compared with infrared sensors and radar system, monocular camera based on visible light can capture images with more details and high-resolution. With the tremendous development of deep-learning based methods on visible light images in recent years, monocular visible light sensor becomes an effective supplement for airspace situational awareness. We break down the 3D aircraft pose estimation problem into three subtasks: aircraft object detection, 2D key-points localization and 3D aircraft orientation estimation. Hence, we propose a network including three branches based on Mask R-CNN [9] architecture to address the tasks, named as AirPose network.

The involvement of deep-learning methods has greatly promoted the development of object detection, 2D key-points localization and 3D object orientation estimation algorithms to the next level. However, as for 3D object pose estimation methods, the lack of accuracy is still the main challenge to be tackled [11]. Some works recently regress [39]

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

**FIGURE 1.** Examples from our AKO dataset. AKO dataset contains images and corresponding pose information of aircraft objects with great component variety.

or classify [10] the 3D pose directly from the images in an end-to-end manner. Some approaches utilize key-points location as an intermediate representation to improve the performance by smoothing the model training process followed by Random Sample Consensus (RANSAC) based Pespective-n-Point (PnP) algorithms which is not an end-to-end network. Besides, the demand for 3D object models restrict the network to specific objects, resulting in the weakness of general applicability.

In this paper, we propose a 3D aircraft orientation estimation pipeline taking the output of key-points localization network as prior knowledge. The network fuses the key-points localization information as geometry feature with the extracted color feature to provide more robust aircraft pose information which enables us to exploit the key-points feature whereas avoiding RANSAC based PnP algorithms and the necessity of 3D models of the objects.

Another problem for applying deep-learning based methods on uncommon situation is the difficulty of data collection. Acquisition of abundant high-quality images is extremely effort-consuming in aircraft pose estimation context. To address this problem, we build the AKO platform based on Unreal Engine 4 (UE4) to generate aircraft images automatically labeled with 3D orientation and key-points location. We use the AKO platform to construct a dataset, named AKO dataset,[1] containing 15000 synthetic images for training and testing. The synthetic images are generated by merging the aircraft images and background images or constructed scenes. We adopted domain randomization (DR) skill [12] in the AKO platform to strengthen the network general applicability.

Through experiments and ablation study, each of our three network branches shows competitive performance on AKO dataset in comparison with the state-of-the-art algorithm [13]. It can be concluded that our simple but effective feature fusion method has greatly improved the accuracy than directly inferring the 3D orientation of object using only feature maps extracted from the original image. Compared with methods in previous works like [33], [39] [34] that need the specific 3D model of objects for pose estimation, our network shows convincing generalization ability for different aircraft model.

*Contributions:* In the light of previous work, the contributions of our work are summarized as follows:

i. We propose a novel aircraft-oriented network, named as AirPose network to address the issue of aircraft 3D pose estimation. To the best of our knowledge, this is the first work to combine keypoints localization and 3D pose estimation in an single end-to-end architecture.
ii. We explore a feature fusion method to effectively fusing the key-points geometry feature and original color feature, which significantly improves the 3D pose estimation accuracy.
iii. We construct an image data generation platform for 3D aircraft pose estimation applying domain randomization skill, which enable us to build image dataset, *i.e.* AKO dataset, at a low cost. We make the AKO dataset publicly available at https://www.kaggle.com/portguss/ako-dataset.

## II. RELATED WORK

### A. OBJECT DETECTION AND 2D KEY-POINTS LOCALIZATION

Briefly, object detection problem has been researched for many years. Recent approaches, such as R-CNN [14], Fast R-CNN [15], Faster R-CNN [16] and YOLO [17], show amazing performance on detection task. As for Key-points localization problem, it has attracted considerable study in recent years [9], [18]–[23] especially for human body pose estimation since Toshev and Szegedy utilized deep learning method to directly regress the key-points 2D coordinates with multi-stage network architecture [18]. The fully convolutional network proposed by Long *et al.* enables the network make dense predictions for per-pixel which greatly improves the key-points localization accuracy [19]. Based on FCN, Newell *et al.* proposed an hourglass architecture in which the features are processed across all scales to improve the performance [21]. He *et al.* adopt Faster R-CNN and FCN to propose Mash R-CNN architecture for object detection and instance segmentation, which can be easily generalized to key-points localization task [9].Different from the human body, the aircraft body struture is rigid. This inflexibility makes aircraft key-points relatively more detectable.

### B. 3D AIRCRAFT POSE ESTIMATION

Some of the previous aircraft pose estimation works are focused on handcrafted feature selection and aircraft geometry structure, such as line feature detection [3]–[5], [24], [25] and skeleton extraction algorithm [26]–[28].These methods have low computational complexity, but also limitations. When the certain parts of the object is self-occluded, the geometry features like line feature are not detectable which makes them vulnerable to occlusion. Moreover, these

[1] https://www.kaggle.com/portguss/ako-dataset

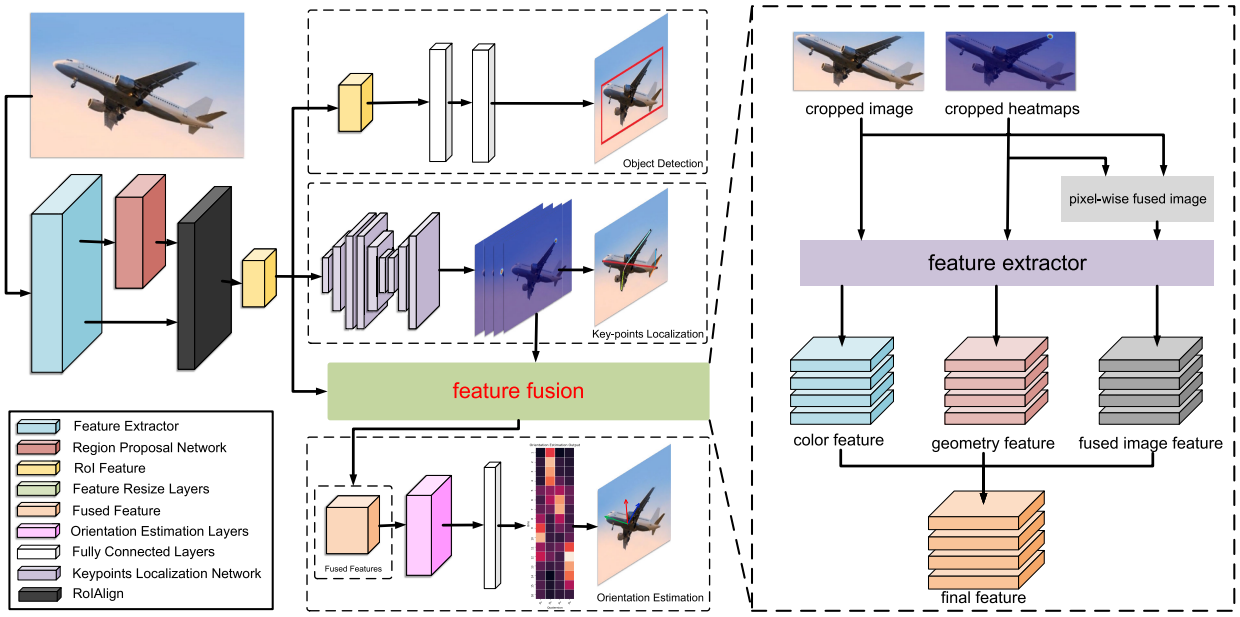


**FIGURE 2.** Overall Architecture: the the head network is followed by three branches. The upper branch is to detect a 2D bounding box of the object. The middle branch uses RoI feature to output the location of each key-points. Then the key-points information and the cropped bounding box of the object are fused in the feature fusion module which yields the fused features. The lower branch which utilizes the fused features to output the orientation estimation of the object. Feature Fusion:first the cropped image and the cropped heatmaps are fused in pixel-wise. Then the cropped image, the cropped heatmaps and the pixel-wise fused image are fed-in the feature extractor module to yield color feature, geometry feature and fused image feature. The three feature are then stacked to the final feature.

algorithms require at least two sensors to yield 3D pose information [3], [4].

Recently, driven by the effectiveness of convolutional network [19], [29], [30], plenty of approaches to 3D pose estimation are based on deep learning methods [31]. These algorithms can be divided into methods using 3D model during inferring [23], [32]–[35]and methods without 3D model matching that directly yield 3D pose information from 2D images [36]–[40]. Mahendran *et al.* use DL methods on 3D pose estimation by directly regressing the relative pose [39]. Su *et al.* train pose estimation network on the synthetic dataset, also use model augmentation to increase the general applicability of their network [41]. Xiang *et al.* introduce a multi-branch network separately outputs the relative rotation, semantic labels and center of the object [33]. Li *et al.* match the rendered image against the observed image iteratively to refine the pose [34]. Pavlakos *et al.* and Song *et al.* adopt semantic key-points localization step to improve the orientation estimation performance [23], [42].

Broadly, these methods can yield accurate orientation estimation, however, requiring strict prerequisites, such as depth information [10], [33], [43] and precise model of target to be detected [34], [35]. As for aircraft pose estimation situation, the depth information is hard to collect due to the long distance between the sensor and the target, also the model of the object is not available if it's non-cooperative object.

### C. SYNTHETIC IMAGE DATASET

One of the most significant problem for using deep-learning skill in monocular 3D pose estimation is the deficiency of image dataset with accurate annotations of 3D pose information. Recently, researchers start using synthetic images dataset to train deep learning network for object detection [44], [45], key-points localization [46], semantic segmentation [47] or 3D pose estimation [2], [13], [41], [48]. Su *et al.* first use synthetic images for 3D viewpoint estimation, also use 3D model deformation for dataset augmentation [41]. Tobin *et al.* and Tremblay *et al.* introduce the domain randomization (DR) technique bridging the reality gap for training models on simulated images [12], [49]. Sharma *et al.* and Proenca *et al.* respectively propose URSO visual simulator and SPEED dataset [2], [13] for flying machine pose estimation like ours, however, both their datasets are based on very few specific models with limited general applicability.

## III. AIRPOSE NETWORK

### A. AIRCRAFT DETECTION

Our work introduces the AirPose network with three branches and a feature fusion module to detect the aircraft, locate the key-points and estimate the orientation in parallel. Fig.2 shows our 3D pose estimation overall pipeline. The input of the network is an RGB image $\mathbf{X}$ of single aircraft in the width of $w$ and height of $h$. Subsequently, a feature extractor based on ResNet-50, $f(*)$, with pre-trained weights is shared by the three branches as the network backbone:

$$f(\mathbf{X}) \rightarrow \mathbf{X}', \tag{1}$$

where $\mathbf{X}'$ refers to the extracted feature maps. Subsequently, $\mathbf{X}'$ is sent to the Region Proposal Network which outputs

s set of aircraft object proposals, each with an objectness score [16]. After the RPN, the feature of proposals are resized to a fixed size of $7 \times 7 \times 256$ by applying RoIAlign [9]. Mark the fixed size feature map as $\mathbf{M}_i$ corresponding to the $i^{th}$ proposal region. The first branch, $l(*)$, is to output the bounding box of aircraft object as follow:

$$l(\mathbf{M_0}, \mathbf{M_1}, \ldots, \mathbf{M_k}) \rightarrow \hat{\mathbf{v}} = (\hat{v}_x, \hat{v}_y, \hat{v}_w, \hat{v}_h), \tag{2}$$

where $\mathbf{v}$ is the prediction of object location. In this branch, the feature, $\mathbf{M}_i$, is sent to fully connected layers to output both the softmax probability of aircraft and bounding box regression offsets. The loss fuction of first branch can be written as:

$$L_1 = L_{cls} + \lambda L_{loc}(\mathbf{v}, \hat{\mathbf{v}}). \tag{3}$$

Mark the fixed size feature of detected object as $\mathbf{M}'$. We utilize $\mathbf{M}'$ to get the object key-points localization and 3D orientation estimation in the next two branches.

### B. KEY-POINTS LOCALIZATION

The second branch, $g(*)$, is used to estimate the $P$ key points locations, $L_p$, where $p \in \{1, \ldots, P\}$ for all $P$ parts:

$$g(\mathbf{M}') \rightarrow \{\hat{b}_p(Y_p = z)\}_{p \in \{1,\ldots,P\}}, \tag{4}$$

where $Y_p \in \mathcal{Z} \subset \mathbb{R}^2$, $b_p$ is the relative possibility of the $p^{th}$ part is at every location $z = (x_z, y_z)$ of $\mathcal{Z}$ predicted by the network. We merge all the $b_p(Y_p = z)$ for $P$ parts to generate the corresponding belief maps:

$$\hat{\mathbf{b}}_p[x_z, y_z] = \hat{b}_p(Y_p = z)_{p \in \{1,\ldots,P\}}. \tag{5}$$

In this work, we select eleven representative semantic key-points closely related to the aircraft structure from aircraft body (*i.e.* nose×1, tail×1, wingtip×2, wing root×2, horizontal tail tip×2, vertical fin tip×1 and engine×2). To yield precise aircraft parts locations, our key-points localization network adopts stacked hourglass architecture [21] based on successive steps of pooling layers and upsampling layers. Different from original hourglass network, our architecture starts from the upsampling stage. The result confidence maps are refined by the cascaded hourglass-like convolutional network. Every hourglass stage can be divided into two components. The first is an encoder stage, of which the convolutional layers and max pooling layers continuously reduce the resolution of the feature maps by half. After the resolution comes to the lowest, as shown as the smallest and layer in hourglass model in Fig.2, the second stage of upsampling begins. Instead of use transposed convolutional layer, this network takes bilinear interpolation as a simplified approach to continuously increase the feature resolution by a factor of two until reaching the output resolution. And the information of every upsampling layer and its corresponding same-scale downsampling layer are linked together to keep the representability of the features. After the final upsampling layer, the network produces the prediction in the form of confidence heatmaps. The ground truth heatmap of each part

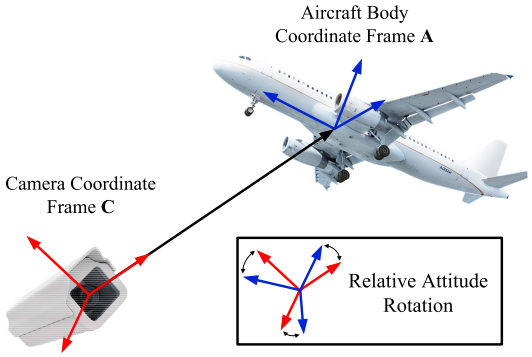


**FIGURE 3.** Definition of camera coordinate frame, C and aircraft body coordinate frame, A. The relative attitude rotation is associated to $\mathbf{q_{AC}}$.

is generated by applying a 2D Gaussian distribution centered at the labelled position $L_p = (x_p, y_p)$. The groundtruth associated with the $p^{th}$ part can be written as:

$$\mathbf{b}_p(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{-\left((x - x_p)^2 + (y - y_p)^2\right)}{2\sigma^2}\right) \tag{6}$$

Then a L2 loss function is applied to train the hourglass network comparing the prediction $\hat{\mathbf{b}}_p$ to the groundtruth $\mathbf{b}_p$:

$$L_2 = \frac{1}{P} \sum_{p=1}^{P} \|\hat{\mathbf{b}}_p - \mathbf{b}_p\|_2^2. \tag{7}$$

The loss function is minimized during intermediate supervision. After the supervision layer, there begins another technically same hourglass stage to refine the prediction produced by the first module. Then the refined prediction $\hat{L}_p$ can be inferred from the maximum response of the final output heatmaps $\hat{\mathbf{b}}_p$ as follow:

$$\hat{L}_p = \arg\max_{(x_z, y_z) \in \mathcal{Z}} \hat{b}_p[Y_p = (x_z, y_z)], \tag{8}$$

where $\{\hat{L}_0, \hat{L}_1, \hat{L}_2, \ldots, \hat{L}_{P-1}\}$ is the locations of $P$ individual parts. The results analysis and ablation study of key-points localization branch are shown in section V. In experiments, we find that the output heatmaps of first branch contains a wealth of aircraft structure information, which is significantly beneficial for orientation estimation branch.

### C. ORIENTATION ESTIMATION

The third branch, $h(*)$, is used to estimate the relative orientation of the target aircraft. As shown in Fig.3, the orientation can be represented by the rotation between the aircraft body coordinate frame, $A$, and the camera coordinate frame, $C$. In consideration of applying smooth interpolation and avoiding the Gimbal Lock problem, we adopt the quaternion, $\mathbf{q_{AC}}$, to represent this rotation, note that $\hat{\mathbf{q}}_{\hat{AC}}$ is marked as $\hat{\mathbf{q}}$.

Before directly use extracted feature $\mathbf{M}'$ to produce the estimation of quaternion, we analyse the correlation between keypoints localization error and orientation error. As shown in Fig.4, the orientation accuracy is closely related to key-points localization accuracy. Thus, this paper proposes a

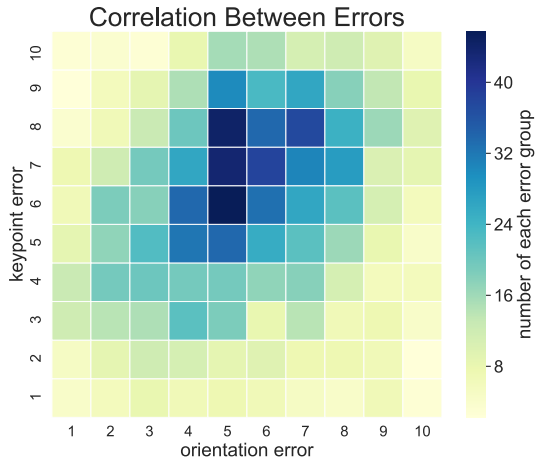


**FIGURE 4. Relativeness Analysis: We discretize both the normalized keypoints localization error and orientation error to ten subsets from group 1 to 10. With a certain fixed key-points localization error, the distribution of orientation error can be seemed as an approximate normal distribution. Along with keypoints localization error grows, the peak of the orientation error distribution shifts up to higher mean error.**

feature fusion method shown as the green block and in Fig.2 that fuse the key points location information $\hat{\mathbf{b}}_p[x_z, y_z]$ with the original feature maps $\mathbf{M}'$ for the improvement of pose estimation accuracy.

So the orientation estimation branch, $h(*)$, is to produce the quaternion as follow:

$$h(\mathbf{M}', \hat{\mathbf{b}}_0, \hat{\mathbf{b}}_1, \ldots, \hat{\mathbf{b}}_{P-1}) \rightarrow \hat{\mathbf{q}}. \tag{9}$$

Firstly, we discretize the $SO(3)$ space by uniformly sampling 32 bins from each orientation dimension. Then each of the $32 \times 32 \times 32$ Euler angles, $(\phi_i, \theta_i, \psi_i)$, is converted to corresponding quaternion $\mathbf{q}_i$, where the $\phi_i, \theta_i, \psi_i$ is yaw, pitch and roll of $i^{th}$ bin. Our goal is to produce the estimation, $\hat{\mathbf{q}}$, as close to the groundtruth $\mathbf{q}_{gt}$.

To yield precise orientation estimation for aircraft object, we propose the feature fusion module, shown in Fig.2, utilizing neural network to take in to consideration both the original image information and located key points information simultaneously.

After the key points location branch outputs the confidence maps of size $w' \times h' \times P$, we fuse the confidence maps with the original image by weighted averaging. Then we extract the feature of the heatmaps and the fused image

by using RoIAlign layer from [9] and conv layers to resize the features to fixed size $7 \times 7 \times 128$.

The features from three sources are then stacked together to a final feature providing more abundant aircraft pose information of size $7 \times 7 \times 512$.

Instead of directly regress the relative attitude, $\hat{\mathbf{q}}$, from the stacked features [39] nor do hard viewpoint classification [41], this paper addresses the pose estimation in a soft-classification manner enabling the network outputs more accurate results. Unlike One-Hot coding using in other classification tasks, we introduce a soft-classification coding which can be written as follow:

$$w_i = \frac{\exp\left(-\alpha_i^2/2\sigma^2\right)}{\sum_i \exp\left(-\alpha_i^2/2\sigma^2\right)}, \quad \forall i \in \Omega, \tag{10}$$

where $\Omega$ includes the indices refer to the $K$ nearest quaternions to the ground truth quaternion, $w_i$ is the confidence value assigned to the $i^{th}$ bin, $\sigma$ is a parameter that controls the Gaussian width and $\alpha_i$ is the angular distance between $\mathbf{q}_i$ and $\mathbf{q}_{gt}$:

$$\alpha_i = \arccos\left(\left|\mathbf{q}_{gt}^T \cdot \mathbf{q}_i\right|\right), \quad \forall i \in \Omega. \tag{11}$$

Then the total orientation estimation loss function, $L_3$, can be written as follow:

$$L_3 = -\sum_{j=1}^{K} w_j \log\left(\frac{e^{\hat{w}_j}}{\sum_{j=1}^{K} e^{\hat{w}_j}}\right) + \lambda L_{reg}, \tag{12}$$

where $L_{reg}$ represents the L2 regularization loss preventing overfitting and penalizing the large weights. We train the network using $L_3$ to output the estimation weights $\hat{w}_j$, then the final estimation $\hat{\mathbf{q}}$ can be inferred by minimizing the weighted least squares as follow:

$$\hat{\mathbf{q}} = \arg\min_{\mathbf{q}} \sum_{i=1}^{K} w_i (1 - |\mathbf{q} \cdot \mathbf{q}_i|)^2 \tag{13}$$

## IV. IMAGE DATA GENERATION

Our neural network pipeline contains millions of parameters to train, which necessitates a large annotated image dataset. In the aircraft pose estimation context, actual camera image dataset especially the relative orientation information is extremely hard to be obtained from non-cooperative aircraft using monocular camera. Therefore, we build an platform on UE4 to generate aircraft 3D pose dataset, named AKO dataset, containing 15000 synthetic images for training and testing. The synthetic images are rendered on UE4 by merging the 3D aircraft model and background images or constructed scenes. The images are automatically labeled with object bounding-box, key-points location and orientation information. We adopted domain randomization (DR) [12] in the AKO dataset skill to strengthen the network general applicability. To better learn the general aircraft structure, we apply random structure deformation on twelve different types of aircraft model as data augmentation.

The dataset includes 12 types of aircraft models, besides, DR skill have been adopted from [49] to better learn the aircraft structure knowledge. The AKO dataset makes training and testing the AirPose network a feasible task.

We apply DR skills in following aspects:

- **Camera Parameters:** focal distance, aperture size;
- **Light Conditions:** location and intensity of the sun light, number of point light sources(from 1 to 8);

- **Camera Placement:** location, distance, angle of the camera with respect to the scene, note that the camera location is related to the variety of specific scene (*e.g.*, the location is more likely to be under the object when the background is sky);
- **Background Images:** two sources: background photograph and rendered scene;
- **Image Noise:** random Gaussian noise and random Gaussian edge blurring to the object;
- **Model Augmentation:** random texture and painting on model surface, and model stretch(range from 0% to 10%).

DR skill makes our images generated in a non-photorealistic way. However, this non-photorealistic manner do not down our model precision after the fine-tune on real images. On the contrary, our images include more variations and the generation process is far faster due to the DR technique. Note that our data generation pipeline outputs and resizes the images to resolution of 720 × 480.

## V. EVALUATION

### A. DATA AUGMENTATION

Our approach is evaluated on the AKO dataset. As for data augmentation, different from previous works, we apply data augmentation in a relatively prudent way since the classic data augmentation method such as spinning and cropping can cause the camera parameters change and the ground truth label error. Instead, as an offset, we generate more images to substitute data augmentation process by applying domain randomization and other process that won't change image resolution and spatial features still remain such as adding noise.

### B. IMPLEMENTATION DETAILS

We implement our pipeline on a single NVIDIA RTX2070S GPU with pytorch 1.0. In training period, we use Stochastic Gradient Descent (SGD) with a momentum of 0.9, a mini-batch size of 4 images from AKO dataset and a weight decay of 0.0001. The learning-rate is set to 0.001 at the beginning and decreased by ten respectively after the 5 epochs and 10 epochs. The feature extractor backbone is ResNet of depth 50.

Unless otherwise specified, the synthetic dataset is applied full domain randomization, the model is trained on synthetic images and then fine-tuned on real images, the test set contains only real images. The depth of feature from ROI is 256, and the fused key-points feature depth is 128. The key-points network including 4 stages to refine the results, and $SO(3)$ space is discretized to 32 bins for each dimension. The feature fusion process is applied by default. All training and testing processing are based on our AKO dataset.

### C. PERFORMANCE METRICS

To measure the performance of aircraft detection branch, we use the Intersection-Over-Union (IoU) metric as follow:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}. \tag{14}$$

**TABLE 1. Overall Performance.**

| Model | Testset | Detection | 2D Key-points | 3D Orientation | 3D Orientation |
|---|---|---|---|---|---|
| | | Mean IoU | PCK | Mean $E_{ori}$ | Mean $E_{ori}$ |
| | | ours | ours | modified SPN | ours |
| $M_s$ | $T_s$ | 75.6 | 87.9 | 11.3° | **8.4°** |
| $M_s$ | $T_r$ | 69.3 | 76.9 | 19.6° | **14.6°** |
| $M_{s+r}$ | $T_s$ | 76.1 | 88.3 | 10.2° | **8.0°** |
| $M_{s+r}$ | $T_r$ | 75.9 | 84.1 | 13.7° | **9.7°** |

As for keypoints localization, we use modified Percentage Correct Keypoints (PCK) metric named as aPCK that calculate the percentage of joints with predicted locations which are no further than a normalized distance from the ground truth on AKO dataset. This normalized distance associated with the airframe size is calculated by:

$$l_{norm} = k \times (l_x + \alpha \times l_y + \beta \times l_z), \tag{15}$$

where $l_x$, $l_y$, $l_z$ is the distances on 2D image respectively from nose to tail, left wing-tip to right wing-tip and fintip to the midpoint of horizontal stabilizer. We determine $k = 0.05$, $\beta = 2.5$ and $\alpha = 1.2$.

The angular distance between the estimated quaternion and the ground truth quaternion to evaluate our orientation estimation branch can be calculated as follow:

$$E_{ori} = \arccos\left(\left|\hat{\mathbf{q}} \cdot \mathbf{q_{gt}}\right|\right) \tag{16}$$

### D. RESULTS

First, Table 1 shows the oveall results of our three-branches pipeline. Where $M_s$ is the model trained only on synthetic images, and $M_{s+r}$ is $M_s$ then fine-tuned on real images. $T_s$ and $T_r$ are testsets including synthetic images and real images respectively. We evaluate the performance for $M_s$ and $M_{s+r}$ on $T_s$ and $T_r$. The model without fine-tuning on real image testset, $M_s$, yields high-precision results on $T_s$, but the accuracy decreases on real image testset $T_r$ rapidly. We alleviate this overfitting problem using fine-tuning metric on $M_{s+r}$ and achieve a significant improvement of compared with $M_s$. To compare the performance of orientation estimation with the performance of the state-of-the-art method, we implement and slightly modify the SPN proposed by [13]. The results shows that the orientation error of our AirPose network is significantly smaller than SPN's due to our deeper network. Some examples of our results are shown in Fig.5.

Table 2 shows how the feature fusion module affect the 3D pose estimation by switching the combinations of the three feature sources. The results shows that our feature fusion processing significantly improves the 3D pose estimation performance by up to 1.3°compared to only using the color feature. Adding keypoints feature and fused image feature could also reduce the Mean $E_{ori}$ by 0.9°and 0.6°.

Table 3 shows how increase of feature depth improve the performance of overall pipeline. The performance changes significantly from the depth of 32 to 256 for all the three tasks. But from 256 to 512, compared to the increase of parameters, the growth of accuracy slows down.

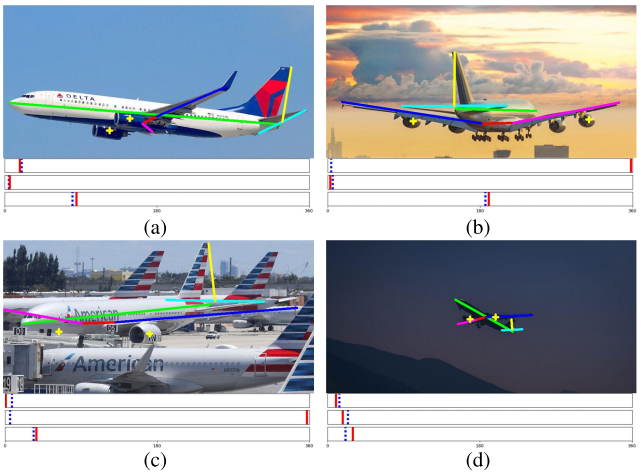


**FIGURE 5.** Some examples from our network with predicted key-points location and 3D orientation estimation. The prediction and ground-truth of 3D orientation are shown as blue dotted lines and red lines respectively.

**TABLE 2.** Impact of Feature Fusion Module to 3D Pose Estimation.

| color feature | keypoints feature | fused image feature | Mean $E_{ori}$ |
|---|---|---|---|
| ✔ | ✘ | ✘ | 11.0° |
| ✔ | ✔ | ✘ | 10.1° |
| ✔ | ✘ | ✔ | 10.4° |
| ✔ | ✔ | ✔ | 9.7° |

**TABLE 3.** Impact of Feature Depth.

| Feature Depth | Detection | key-points | Orientation |
|---|---|---|---|
| | Mean IoU | PCK | Mean $E_{ori}$ |
| 32 | 67.5 | 74.3 | 16.7° |
| 128 | 73.1 | 81.3 | 11.4° |
| 256 | 75.9 | 84.1 | 9.7° |
| 512 | **76.3** | **84.7** | **9.2°** |

The impact of our multi-stage architecture for key-points localization is shown in Table 4. We compare aPCK of each key-points on model architecture of 1 to 8 stages. Parts with distinct edges, such as nose, wing tail (WT), horizontal tail tip (HTT), vertical fin tip (VFT) and tail, are easier to be localized and get more accurate results compared to parts with less texture and less significant margin. As the number of stages increases, the performance gets more accurate. Note that all of the stages share the total same structure. The effect of multi-stage architecture is notable at 1- to 2- and 2- to 4-stage, the accuracy of which are 76.9%, 83.1% and 88.2%. The modest improvement is from 4- to 8-stage: from 88.2% to 88.8%. key-points localization performance respect to the normalized distance $k$ on each part are shown in Fig.6.

We also propose an ablation study to discuss the effect of domain randomization applied in our AKO dataset by take one of the domain randomization parameters out at a time. As shown in Fig.7, both the keypoints localization and orientation estimation accuracy change with coherence of

**TABLE 4.** Impact of Stage Number On Key-points Localization.

| Model | Nose | WT | WR | HTT | VFT | Tai | Eng | Mean |
|---|---|---|---|---|---|---|---|---|
| 1-stage | 83.3 | 77.1 | 70.9 | 78.5 | 77.6 | 82.2 | 75.1 | 76.9 |
| 2-stage | 86.9 | 84.0 | 73.2 | 84.2 | 81.7 | 86.1 | 80.0 | 81.6 |
| 4-stage | 89.4 | 88.2 | 74.7 | 84.5 | 85.9 | 88.9 | 83.0 | 84.1 |
| 8-stage | 90.2 | 89.3 | 75.8 | 84.5 | 86.4 | 90.6 | 83.76 | 84.9 |

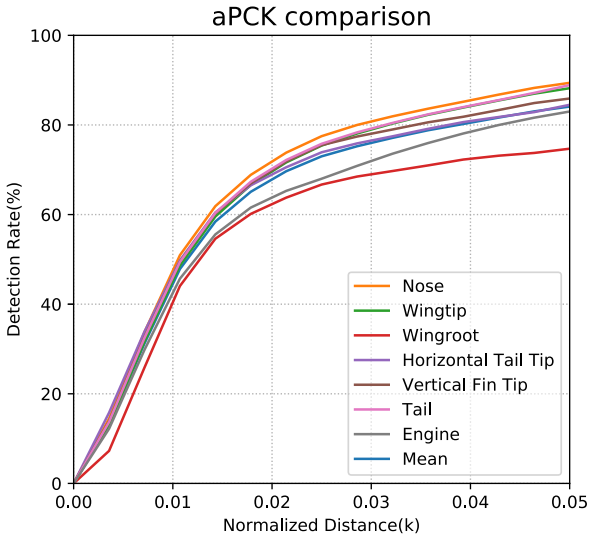


**FIGURE 6.** aPCK comparison for each part.

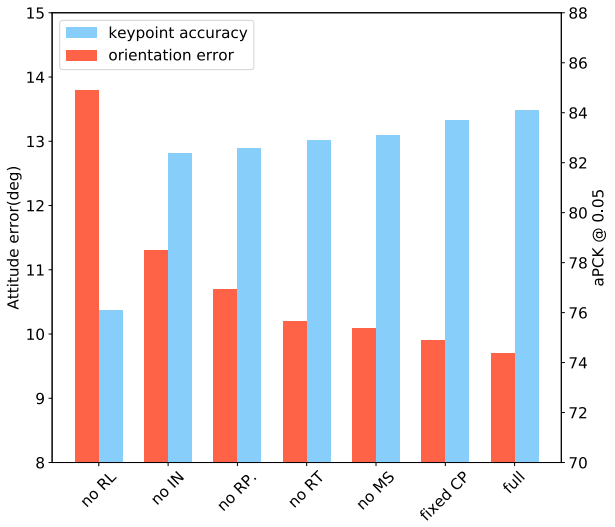


**FIGURE 7.** Impact on performance of both the keypoints localization and the orientation estimation by omitting each of the DR randomized conditions. RL, IN, RP, RT, MS and CP represent randomized light, image noise, randomized painting, randomized texture, model stretch and camera parameters respectively.

changing trend. As we can see, the absence of randomized light most hurts the performance. Without light randomization skills, the aPCK drops to 76.1 and the attitude error increase to 13.8. Different from [49], our results shows one unexpected point that the missing of random aircraft surface texture did not decay the accuracy as much in [49], which can be explained by that the distinct structure of objects and the lower-complexity background reduced the necessity of random texture.

**TABLE 5. General Applicability of AirPose Network.**

| tasks | known model | unknown model |
|---|---|---|
| key-points | 83.9 | 84.2 |
| 3D attitude | 9.5° | 9.8° |

The aircraft models in the testing set can be divided into known model and unknown model. The unknown model means the aircraft model set that has no intersection with the training data model set, and vice versa. Table 5 shows the results are not sensitive to the prior of aircraft model. Both experiments on known model and unknown model can achieve high-accuracy at the almost same level which shows the convincing generalization ability of our network for different aircraft model without the prerequisite of the specific 3D model.

## VI. CONCLUSION

This paper proposes the AirPose network for aircraft object detection, 2D key-points localization and 3D orientation estimation with our AKO dataset. We show how our feature fusion method and domain randomization skill benefit the overall performance. As future work, further research is required in following directions. First, we will evaluate and improve the computational runtime and memory usage of our AirPose network for embedding it in hardware. Second, we will apply our method to video sequences. Lastly, we envision 3D pose estimation using external monocular sensor an promising direction in the field of airspace situational awareness.

## REFERENCES

[1] S. Sharma, C. Beierle, and S. D'Amico, "Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks," in *Proc. IEEE Aerosp. Conf.*, Big Sky, MT, USA, Mar. 2018, pp. 1–12. [Online]. Available: https://ieeexplore.ieee.org/document/8396425/

[2] P. F. Proenca and Y. Gao, "Deep learning for spacecraft pose estimation from photorealistic rendering," Jul. 2019, *arXiv:1907.04298*. [Online]. Available: http://arxiv.org/abs/1907.04298

[3] X. Teng, Q. Yu, J. Luo, G. Wang, and X. Zhang, "Aircraft pose estimation based on geometry structure features and line correspondences," *Sensors*, vol. 19, no. 9, p. 2165, May 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/9/2165

[4] 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, Jan. 2019. [Online]. Available: http://www.mdpi.com/1424-8220/19/2/342

[5] L. Chen, B. Guo, and W. Sun, "Relative pose measurement algorithm of non-cooperative target based on stereo vision and RANSAC," *Int. J. Soft Comput. Softw. Eng.*, vol. 2, no. 4, pp. 26–35, Apr. 2012. [Online]. Available: http://www.jscse.com/papers/?vol=2&no=4&n=3

[6] D. G. Ward, J. F. Monaco, and M. Bodson, "Development and flight testing of a parameter identification algorithm for reconfigurable control," *J. Guid., Control, Dyn.*, vol. 21, no. 6, pp. 948–956, Nov. 1998, doi: 10.2514/2.4329.

[7] M. Pachter, J. J. D'Azzo, and A. W. Proud, "Tight formation flight control," *J. Guid., Control, Dyn.*, vol. 24, no. 2, pp. 246–254, 2001, doi: 10.2514/2.4735.

[8] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Trans. Autom. Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," Jan. 2017, *arXiv:1703.06870*. [Online]. Available: http://arxiv.org/abs/1703.06870

[10] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "DenseFusion: 6D object pose estimation by iterative dense fusion," Jan. 2019, *arXiv:1901.04780*. [Online]. Available: http://arxiv.org/abs/1901.04780

[11] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. G. Buch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, C. Sahin, F. Manhardt, F. Tombari, T. Kim, J. Matas, and C. Rother, "BOP: Benchmark for 6D object pose estimation," in *Proc. 15th Eur. Conf.*, Munich, Germany, Sep. 2018, pp. 19–35.

[12] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," Mar. 2017, *arXiv:1703.06907*. [Online]. Available: http://arxiv.org/abs/1703.06907

[13] S. Sharma, C. Beierle, and S. D'Amico, "Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks," in *Proc. IEEE Aerosp. Conf.*, Mar. 2018, pp. 1–12.

[14] 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.*, Columbus, OH, USA, vol. 1, Jun. 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.

[15] R. Girshick, "Fast R-CNN," Sep. 2015, *arXiv:1504.08083*. [Online]. Available: http://arxiv.org/abs/1504.08083

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

[17] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, Jun. 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[18] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1653–1660, [Online]. Available: http://arxiv.org/abs/1312.4659

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," Nov. 2014, *arXiv:1411.4038*. [Online]. Available: http://arxiv.org/abs/1411.4038

[20] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," Jan. 2016, *arXiv:1602.00134*. [Online]. Available: http://arxiv.org/abs/1602.00134

[21] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," Mar. 2016, *arXiv:1603.06937*. [Online]. Available: http://arxiv.org/abs/1603.06937

[22] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," Nov. 2017, *arXiv:1711.07319*. [Online]. Available: http://arxiv.org/abs/1711.07319

[23] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis, "6-DoF object pose from semantic keypoints," Mar. 2017, *arXiv:1703.04670*. [Online]. Available: http://arxiv.org/abs/1703.04670

[24] H. Hmam and J. Kim, *Aircraft Recognition and Pose Estimation*, K. N. Ngan, T. Sikora, M.-T. Sun, Eds. Bellingham, WA, USA: SPIE, May 2000, p. 1198. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/1_2.386709

[25] W. Ling, X. Chao, and Y. Jie, "Aircraft pose estimation based on mathematical morphological algorithm and Radon transform," in *Proc. 8th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Shanghai, China, Jul. 2011, pp. 1920–1924. [Online]. Available: http://ieeexplore.ieee.org/document/6019888/

[26] C. Gold and J. Snoeyink, "A one-step crust and skeleton extraction algorithm," *Algorithmica*, vol. 30, no. 2, pp. 144–163, Jun. 2001. [Online]. Available: http://link.springer.com/10.1007/s00453-001-0014-x

[27] S. Schaefer and C. Yuksel, "Example-based skeleton extraction," in *Proc. SGP*, Jan. 2007, pp. 153–162.

[28] D. Cai, "Study on aircraft recognition in high spatial resolution remote sensing image based on skeleton characteristics analysis," *Adv. Mater. Res.*, vols. 268–270, pp. 1982–1985, Jul. 2011. [Online]. Available: https://www.scientific.net/AMR.268-270.1982

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778. [Online]. Available: http://ieeexplore.ieee.org/document/7780459/

[30] X. Wang, A. Shrivastava, and H. Mulam, "A-fast-RCNN: Hard positive generation via adversary for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3039–3048.

[31] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. G. Buch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, C. Sahin, F. Manhardt, F. Tombari, T.-K. Kim, J. Matas, and C. Rother, "BOP: Benchmark for 6D object pose estimation," Aug. 2018, *arXiv:1808.08319*. [Online]. Available: http://arxiv.org/abs/1808.08319
[32] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Proc. Comput. Vis. ECCV*, vol. 8690, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 536–551. [Online]. Available: http://link.springer.com/10.1007/978-3-319-10605-2_35
[33] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes," Nov. 2017, *arXiv:1711.00199*. [Online]. Available: http://arxiv.org/abs/1711.00199
[34] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep iterative matching for 6D pose estimation," Mar. 2018, *arXiv:1804.00175*. [Online]. Available: http://arxiv.org/abs/1804.00175
[35] Y. Hu, J. Hugonot, P. Fua, and M. Salzmann, "Segmentation-driven 6D object pose estimation," Dec. 2018, *arXiv:1812.02541*. [Online]. Available: http://arxiv.org/abs/1812.02541
[36] F. Michel, A. Kirillov, E. Brachmann, A. Krull, S. Gumhold, B. Savchynskyy, and C. Rother, "Global hypothesis generation for 6D object pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 115–124. [Online]. Available: http://ieeexplore.ieee.org/document/8099503/
[37] K. Hara, R. Vemulapalli, and R. Chellappa, "Designing deep convolutional neural networks for continuous object orientation estimation," Feb. 2017, *arXiv:1702.01499*. [Online]. Available: http://arxiv.org/abs/1702.01499
[38] M. Rad and V. Lepetit, "BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth," Mar. 2017, *arXiv:1703.10896*. [Online]. Available: http://arxiv.org/abs/1703.10896
[39] S. Mahendran, H. Ali, and R. Vidal, "3D pose regression using convolutional neural networks," Aug. 2017, *arXiv:1708.05628*. [Online]. Available: http://arxiv.org/abs/1708.05628
[40] T.-T. Do, M. Cai, T. Pham, and I. Reid, "Deep-6DPose: Recovering 6D object pose from a single RGB image," Feb. 2018, *arXiv:1802.10367*. [Online]. Available: http://arxiv.org/abs/1802.10367
[41] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 2686–2694. [Online]. Available: http://ieeexplore.ieee.org/document/7410665/
[42] X. Song, P. Wang, D. Zhou, R. Zhu, C. Guan, Y. Dai, H. Su, H. Li, and R. Yang, "ApolloCar3D: A large 3D car instance understanding benchmark for autonomous driving," Nov. 2018, *arXiv:1811.12222*. [Online]. Available: http://arxiv.org/abs/1811.12222
[43] J. Sock, S. H. Kasaei, L. S. Lopes, and T.-K. Kim, "Multi-view 6D object pose estimation and camera motion planning using RGBD images," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Venice, Italy, Oct. 2017, pp. 2228–2235. [Online]. Available: http://ieeexplore.ieee.org/document/8265470/
[44] X. Peng, B. Sun, K. Ali, and K. Saenko, "Learning deep object detectors from 3D models," Oct. 2015, *arXiv:1412.7122*. [Online]. Available: http://arxiv.org/abs/1412.7122
[45] T. Hodan, V. Vineet, R. Gal, E. Shalev, J. Hanzelka, T. Connell, P. Urbina, S. N. Sinha, and B. Guenter, "Photorealistic image synthesis for object instance detection," Feb. 2019, *arXiv:1902.03334*. [Online]. Available: http://arxiv.org/abs/1902.03334
[46] D. T. Hoffmann, D. Tzionas, M. J. Black, and S. Tang, "Learning to train with synthetic humans," Aug. 2019, *arXiv:1908.00967*. [Online]. Available: http://arxiv.org/abs/1908.00967
[47] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 3234–3243. [Online]. Available: http://ieeexplore.ieee.org/document/7780721/
[48] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield, "Deep object pose estimation for semantic robotic grasping of household objects," Sep. 2018, *arXiv:1809.10790*. [Online]. Available: http://arxiv.org/abs/1809.10790
[49] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," Apr. 2018, *arXiv:1804.06516*. [Online]. Available: http://arxiv.org/abs/1804.06516

**YIBO LI** was born in Xi'an, China, in 1996. He received the B.S. degree in navigation, guidance, and control from Northwestern Polytechnical University (NWPU), Xi'an, in 2018, where he is currently pursuing the M.S. degree in control engineering. His research interests include object pose estimation, key-points localization, and object detection based on deep learning methods.

**RUIXING YU** was born in Xi'an, China, in 1978. She received the M.S. and Ph.D. degrees in navigation, guidance, and control from Northwestern Polytechnical University (NWPU), Xi'an, in 2003 and 2006, respectively. She was sponsored by the Chinese Scholarship Council to work for one year with the Multimedia Research Group, University of Alberta. She is currently an Associate Professor with the School of Astronautics, NWPU. Her research interests include video tracking, image recognition, and feature detection and analysis.

**BING ZHU** was born in Xi'an, China, in 1977. He received the M.S. and Ph.D. degrees in navigation, guidance, and control from Northwestern Polytechnical University (NWPU), Xi'an, in 2006 and 2012, respectively. He is currently an Associate Professor with the School of Electronic Engineering, Xi'an Shiyou University. His research interests include non-destructive testing and image analysis and processing.