

# A monocular vision-based perception approach for unmanned aerial vehicle close proximity transmission tower inspection

Jiang Bian , Xiaolong Hui , Xiaoguang Zhao and Min Tan

## Abstract

Employing unmanned aerial vehicles to conduct close proximity inspection of transmission tower is becoming increasingly common. This article aims to solve the two key problems of close proximity navigation—localizing tower and simultaneously estimating the unmanned aerial vehicle positions. To this end, we propose a novel monocular vision-based environmental perception approach and implement it in a hierarchical embedded unmanned aerial vehicle system. The proposed framework comprises tower localization and an improved point-line-based simultaneous localization and mapping framework consisting of feature matching, frame tracking, local mapping, loop closure, and nonlinear optimization. To enhance frame association, the prominent line feature of tower is heuristically extracted and matched followed by the intersections of lines are processed as the point feature. Then, the bundle adjustment optimization leverages the intersections of lines and the point-to-line distance to improve the accuracy of unmanned aerial vehicle localization. For tower localization, a transmission tower data set is created and a concise deep learning-based neural network is designed to perform real-time and accurate tower detection. Then, it is in combination with a keyframe-based semi-dense mapping to locate the tower with a clear line-shaped structure in 3-D space. Additionally, two reasonable paths are planned for the refined inspection. In experiments, the whole unmanned aerial vehicle system developed on Robot Operating System framework is evaluated along the paths both in a synthetic scene and in a real-world inspection environment. The final results show that the accuracy of unmanned aerial vehicle localization is improved, and the tower reconstruction is fast and clear. Based on our approach, the safe and autonomous unmanned aerial vehicle close proximity inspection of transmission tower can be realized.

## Keywords

Close proximity inspection of transmission tower, tower localization, UAV self-positioning, monocular vision

Date received: 31 July 2018; accepted: 26 November 2018

Topic: Vision Systems

Topic Editor: Antonio Fernandez-Caballero

Associate Editor: Tiziana D'Orazio

## Introduction

The power transmission tower (PTT) provides a crucial foundation for economic development. The electrical devices for power delivery are mainly concentrated on the PTT. Both the equipments and PTT are exposed to the complex and diverse natural environment and lack the regular maintenance, which may encounter multiple types of

The State Key Laboratory of Management and Control for Complex System, Institute of Automation Chinese Academy of Sciences, Beijing, China

### Corresponding author:

Jiang Bian, The State Key Laboratory of Management and Control for Complex System, Institute of Automation Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing, China.

Email: bianjiang2015@ia.ac.cn



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License

(<http://www.creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

damages and makes power delivery a hidden danger. Autonomous transmission tower inspection has always been a hot issue in the field of robotics. In the last decades, related researches are mainly conducted based on the following two popular platforms<sup>1,2</sup>: unmanned aerial vehicles (UAVs)<sup>3-5</sup> and rolling on wires robots (RWR).<sup>6-8</sup>

In RWR inspection, the inspection is conducted by a robot climbing along transmission lines suspended by a pole tower. The main advantage is the inspection accuracy, because the equipped sensors are close to the PTT and its related components. However, passing across various obstacles on the lines has always been a major weakness for RWR.<sup>1,2</sup> LineScout,<sup>9</sup> a successful wire-climbing robot in recent years, was specifically designed with a LineArm to grasp on both sides of the obstacle and flipped its wheel frame to overcome obstacles. But it loses efficiency when encountering complicated conductor connection on pylon.

In UAV inspection, the inspection platforms mainly contain fixed-wing UAVs<sup>10</sup> and vertical takeoff and landing (VTOL) UAVs which comprise large helicopter<sup>11</sup> and multi-rotor aircraft. However, the first two are either too fast or too far to acquire detail information<sup>12</sup> of the PTT and always with a high inspection cost.<sup>1,2</sup> They are more suitable for a relatively long distance monitoring of power transmission lines (PTLs) along PTLs corridor. Only the multi-rotor UAVs are maneuverable enough to fly and hover quite close to the PTT and keep a high inspection accuracy.<sup>13</sup> Besides, it always has a low operation cost and is capable of accessing different locations for multi-type refined PTT inspection tasks.

Nowadays, based on the requirements of UAV refined PTT inspection, the faults that need to be inspected mainly include the tower deformation and inclination,<sup>14,15</sup> the insulator string broken and contamination,<sup>16,17</sup> and all other kinds of small component faults like damage or missing of shock hammers and wire clips. It requires the UAV to fly in close proximity to the tower and realize fixed-location hovering while maintaining a safe stand-off distance from the pole to take high-quality pictures. So, the UAV should be able to fly safely to avoid the pole and be capable of accurate self-positioning relative to the tower.

At present, the navigation for close proximity tower monitoring can be conducted in three ways consisting of manual operation, GPS-fixed-location navigation, and assisted-control semi-autonomous navigation. The professional manual operation requires that a highly skilled pilot controls the UAV to approach the tower and a co-pilot operates the equipped camera to take pictures. Pilots are required to be highly focused to control the UAV. It loses efficiency because of high operator workload and it has a risk of collision due to improper manipulation. As for the GPS-fixed-location navigation, the UAV is required to carry a camera that hovers at fixed-locations and follows a flight path, which are all preprogrammed by GPS-based geo-locator. Luque et al.<sup>18</sup> achieved navigation around PTTs by ground control station (GCS). The GCS transmits

control inputs to UAV and obtains information from the payload. However, the ability of self-positioning and tower-localizing cannot work under the condition of unstable GPS and this method lacks the consideration of surroundings. With regard to the semiautonomous navigation, its aim is to reduce the operator's cognitive load and level of skill. The methods mainly include using external force feedback through a haptic control device,<sup>19,20</sup> altering the magnitude and direction of the operator's input,<sup>21</sup> and reducing the degree of freedom (DoF) that the operator controls.<sup>22</sup> In essence, these assistances above are based on the UAV perception of the relationship between its own positions and surroundings. Mcfadyen et al.<sup>21</sup> presented a theoretical analysis of sensor performance to constrain the platform behavior by maintaining a safety buffer zone to the electrical pole. Moore et al.<sup>23</sup> developed a UAV system utilizing a lidar to percept polyhedron obstacle and conduct inspection of electrical transmission infrastructure. Sa et al.<sup>22</sup> developed an onboard flight controller using visual features for visual servoing to inspect pole-like structures. In conclusion, autonomous navigation of UAV-refined inspection around the tower is really challenging and it has not been fully implemented. The key problem of safety and autonomy is to give UAV the ability to determine the position of the tower and simultaneously be well aware of its own locations.

In the robotic navigation and infrastructure inspection literature of last decade, the Simultaneous Localization and Mapping (SLAM) is studied extensively and shows great prospect, since it can successfully perform simultaneous estimation of the state of a robot and the construction of a model (map) of the environment. Recently, visual SLAM systems have demonstrated that drift errors of trajectory estimation can be below 1% in real-world outdoor scenes.<sup>24-26</sup> Thus, lately, vision-based navigation is popular for robots like UAV. In addition, images collected by cameras are also ideal data for UAV navigation, because they provide rich information quickly and are easy to be obtained and analyzed. Voigt et al.<sup>27</sup> implemented an embedded egomotion estimation system based on stereo cameras for the inspection of boilers and common indoor scenarios. Burri et al.<sup>28</sup> and Nikolic et al.<sup>29</sup> used a stereo-visual-based quad-rotor platform to realize Visual Odometry (VO) to inspect a thermal power plant boiler system. Teng et al.<sup>30</sup> proposed a power line inspection system solution based on mini-UAV-borne LIDAR system which can extract pole point cloud and detect pole deformation. Cerón et al.<sup>31</sup> implemented a Visual SLAM process in an AR-DRone 2.0 platform and used SLAM for drone navigation in power line surrounding. The detailed studies of visual SLAM system are promising to realize UAV self-positioning and tower-localization for close proximity inspection of high-voltage electric tower.

Before 2010, filter-based visual SLAM was common. Subsequently, the keyframe-based visual solutions in combination with sparse nonlinear optimization were

demonstrated more efficient and more accurate than the filtering approaches.<sup>32</sup> Recent successful keyframe-based real-time SLAM algorithms can be divided into Dense-Direct-based SLAM and Sparse-Feature-based SLAM. The former is capable of reconstructing the environment with a dense or semi-dense map. Meanwhile, the camera motion is estimated by employing photometric errors derived from image pixel intensities. Literatures include LSD-SLAM,<sup>33</sup> DTAM,<sup>34</sup> and REMODE.<sup>35</sup> By comparison, the latter takes advantage of salient image features like keypoints to localize camera and performs a sparse point-based reconstruction of the environment. Examples contain monoSLAM,<sup>36</sup> PTAM,<sup>37</sup> and Oriented FAST and Rotated BRIEF (ORB)-SLAM.<sup>38,39</sup> Among them, the ORB-SLAM seems to be the state-of-the-art in public datasets, yielding better accuracy than direct methods. Currently, the existing wide variety of SLAM frameworks has not been analyzed and tested for refined high-voltage tower inspection.

Line features are very prominent in the PTT inspection environment. It provides abundant and useful visual structural information for UAV odometry especially in the poorly textured and illumination-changing scenes where feature points lose efficacy. In the recent literatures, the combination of point and line features has been employed for SLAM system. Lu et al.<sup>40</sup> presented a Red-Green-Blue-Depth (RGBD) visual odometry utilizing point and line features extracted from RGB-D data. Ruben<sup>41</sup> proposed a probabilistic approach to fuse points and line segments to form a stereo visual odometry. Zhang et al.<sup>42</sup> designed a graph-based visual SLAM system using straight lines with orthonormal representation and achieved better reconstruction performance. However, these studies cannot be directly applied to the transmission tower environments outdoors. The modifications of the SLAM details and the improvements based on the appearance characteristics of the tower are still necessary.

With regard to transmission tower localization, tower is expected to be extracted from the constructed environment map. Whereas, there are many noises in the sparse map and the key points reflect little information of a PTT. Besides, the dense or semi-dense map is too slow to be directly processed on the compact UAV platform with limited computing resources. To locate the PTT fast and accurately, suitable PTT detection algorithm in 2-D image can be fused into the SLAM framework. Martinez et al.<sup>43</sup> developed a machine learning-based approach combined with a tracking-by-registration strategy. But, traditional methods have to be faced with the complicated design of features and the choice of classifier. In recent years, deep learning (DL) technology has achieved great breakthroughs and reached the state-of-the-art in the field of 2-D object detection.<sup>44</sup> However, they quite consume the resources of graphic processing units (GPUs) and cannot yet achieve the real-time performance in the embedded system.

In this article, we propose an effective monocular vision-based environmental perception approach to realize

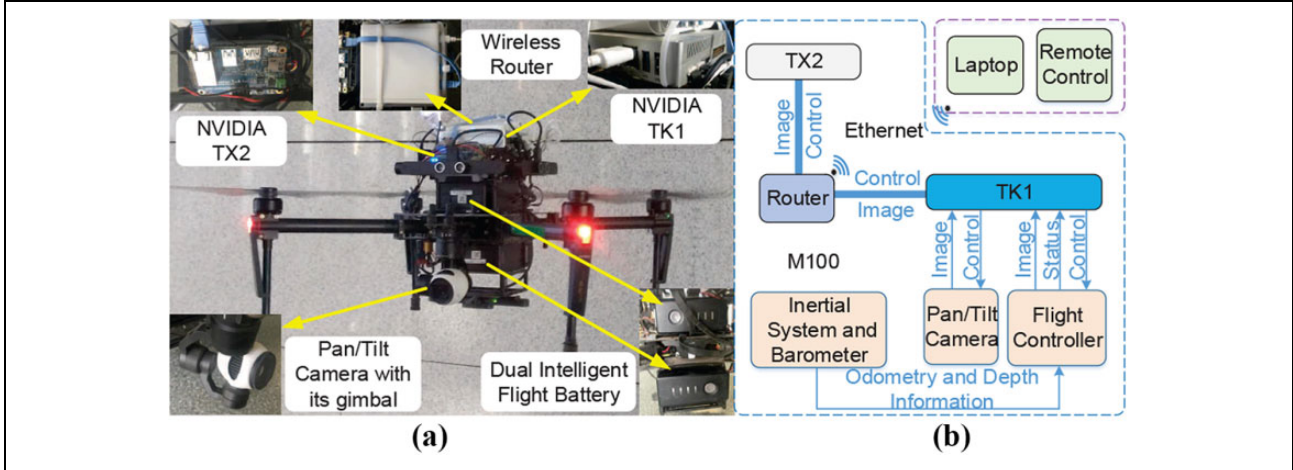
PTT localization and estimate the UAV self-positions for close proximity PTT inspection. We tackle these issues utilizing a point-line-based SLAM and tower detection method (PL-TD). It is implemented in a hierarchically developed embedded UAV system. PTT has the prominent line feature. Thus, an improved extraction and matching method of the line features is presented to enhance frame association. Building upon the ORB-SLAM, a suitable SLAM system leveraging both point and line features is designed followed by a semi-dense mapping which can reflect the contour and line-shaped structure of the PTT. In particular, the bundle adjustment (BA) optimization component is incorporated with the point-to-line distance and the intersections of lines to improve the accuracy of UAV localization. For PTT localization, we create a transmission tower data set (<https://drive.google.com/open?id=1UyP0fBNUqFeoW5nmPVGzyFG5IQZcqlc5>) and customize a fast neural network (Tower Region Convolutional Neural Network (R-CNN)) for PTT detection to address the real-time problem and improve the detection accuracy. The detection can be well fused into the SLAM framework to provide an accurate position of PTT in 3-D space. In addition, we designed two paths that allow the UAV's field of vision to cover most part of the PTT to realize safe close proximity inspection. Then, along the two paths, the whole UAV system built on the Robotic Operating System (ROS) framework is evaluated both in a synthetic scene and a real-world PTT inspection environment and achieves satisfactory results (<https://youtu.be/tF3hrZsBw7w>).

The remainder of this article is organized as follows. The second section gives an overview of the UAV hardware, system architecture, and the two paths planned for refined inspection. The third section explains the DL-based TD and the reason to choose semi-dense mapping. Details of our improved PL-based visual SLAM system, comprising of the heuristic extraction and matching of the lines, the improvement of BA optimization, are described in the fourth section. The experimental results and analyses are shown in the fifth section. Finally, the conclusions are summarized in the sixth section.

## System description and inspection paths

### Hardware platform

We employ a refitted DJI Matrice 100 quad-rotor platform, as shown in Figure 1(a). For the sake of portability and endurance, we leave most of the UAV space for two intelligent flight batteries and concisely equip the UAV with a Pan/Tilt camera (PTC), two advanced low power consumption-embedded processors NVIDIA TK1 and NVIDIA TX2 and a light wireless router. The rewards of adopting a single PTC are great, since it is of low weight, is cheap, consumes low power, and occupies a small mounting space. Besides, the PTC can rotate to



**Figure 1.** (a) The prototype of the refitted DJI Matrice 100 inspection UAV. (b) The hierarchical system architecture of the inspection UAV. UAV: unmanned aerial vehicle.

**Table 1.** Specifications and performance of the inspection UAV.

UAV performance	Concrete parameters
UAV (including TB48D battery)	2431 g
TB48D battery	676 g
Pan/tilt camera	247 g
Wireless router	100 g
Symmetrical motor wheelbase	650 mm
Propeller length	345 mm
UAV height	310 mm
Max. speed	17 m/s
Max. pitch angle	35°
Max. angular velocity	Yaw: 150°/s
Max. speed of ascent	4 m/s
Max. speed of descent	5 m/s
Max. wind resistance	10 m/s
Vertical hovering accuracy	0.5 m
Horizontal hovering accuracy	2.5 m
Hovering time (TB48D*1)	28 min
Hovering time (TB48D*2)	40 min
CE certificate standard	3.5 km
FCC certificate standard	5 km

UAV: unmanned aerial vehicle; CE: Conformité Européenne; FCC: Federal Communications Commission.

provide flexible tower observation perspectives. The main specifications and related performances of the inspection UAV are listed in Table 1.

### System architecture

Taking the system stability and ease of operation into account, the inspection UAV adopts a three hierarchical system architecture based on ROS network, as shown in Figure 1(b). TK1, as an underlying controller, communicates with the PTC and a flight controller. Meanwhile, TX2, with more computing power, is used as an onboard central processor and is primarily responsible for running algorithms. The laptop, for supervising and remote control, works in the

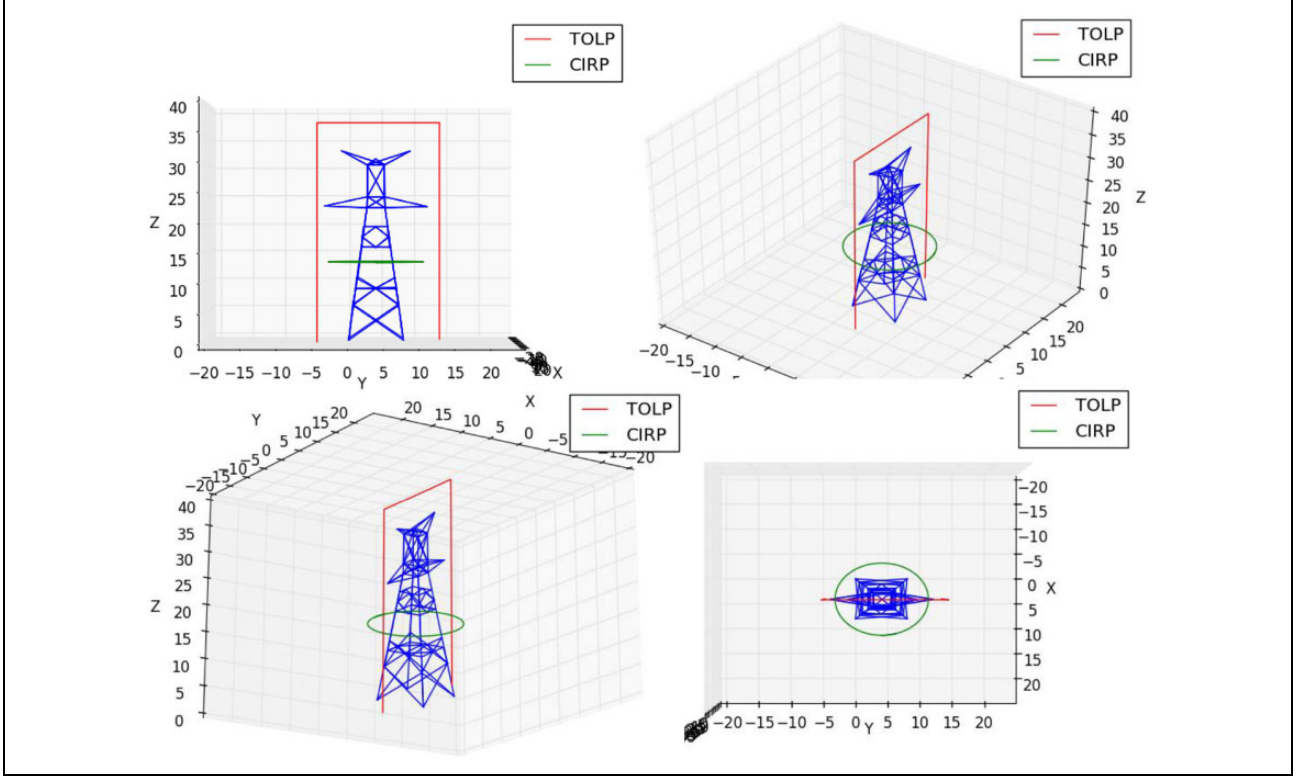
upper level. These three scattered subsystems are connected by the wireless router and an image transmission module.

### Paths for inspection

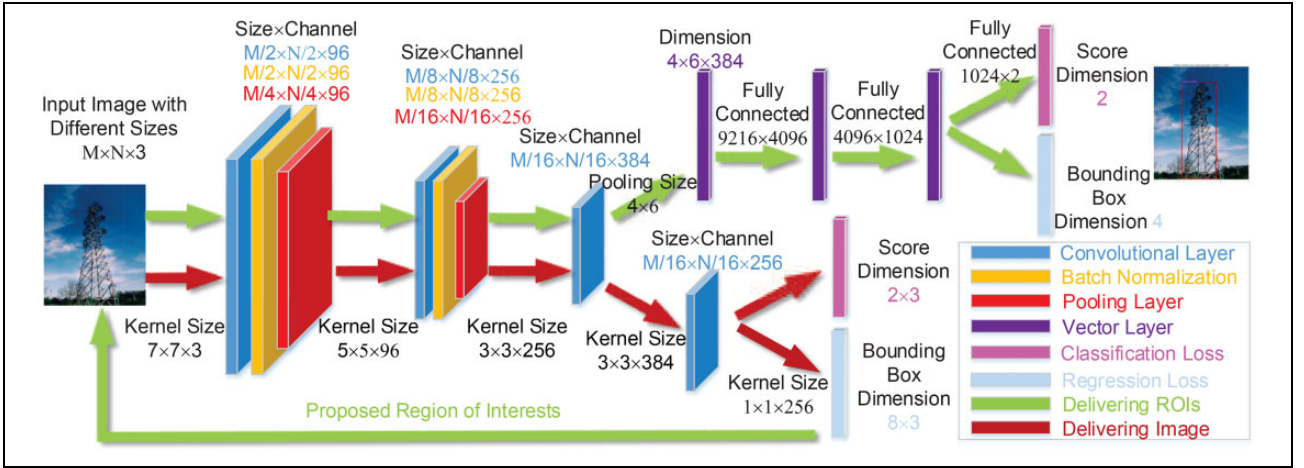
We created a synthetic scene which contains a PTT model. As illustrated in Figure 2, it has the same size as the actual tower and the units are in meters. Then, according to the characteristics of close proximity PTT inspection, two paths are proposed, named “circumvolant path” (CIRP) and “takeoff and land vertically path” (TOLP). As shown in Figure 2, they are denoted by the red and green lines in four different viewpoints, respectively. It can be seen that the two routes can effectively avoid the collision with thin PTLs and make it easy for a UAV’s field of vision to cover most of the electric devices.

### Transmission tower localization

The current DL-based object detection algorithms, which have an outstanding precision rate, can be trained in advance by a large number of tower samples. Faster R-CNN<sup>45</sup> framework is an algorithm that achieves rare false detections in our transmission tower data set among all tested algorithms and can be faster if we limit the number of region proposals. So, we customize the Tower R-CNN based on the tower characteristics and the Faster R-CNN structure to meet the detection requirements of speed and accuracy. As illustrated in Figure 3, the special designs of Tower R-CNN are set as follows: (1) Transmission tower is an structural object that is abundant in low-level edge features and doesn’t need to be described by deeper abstract features. Therefore, Tower R-CNN has fewer convolutional layers, which are capable of shallow feature extraction, to realize detection. Simultaneously, the detection speed is significantly increased. (2) According to the prior information of tower appearance, the anchor boxes<sup>45</sup> only with aspect ratio 2:1 are selected as proposals. It improves region proposal quality and obtains higher detection



**Figure 2.** Synthetic PTT environment and two safe close proximity navigation paths in four views. PTT: power transmission tower.



**Figure 3.** Schematic diagram of the Tower R-CNN.

accuracy. (3) The number of region proposals is usually large and their feature maps all need to be classified by the fully connected layers, which accounts for the amount of time. Due to the more distinguishable tower edge features, it is possible to reduce the parameters and simplify the structure of fully connected layers to improve detection efficiency. Additionally, to avoid overfitting problem and weight contamination, the layer-by-layer training is adopted.

Semi-dense method recovers object contours and textured surfaces. It exploits the information from every

pixel at which the gradient of image intensity is significant. This exactly accords with the line structure of tower appearance. Moreover, it is more useful than sparse point map in navigation due to much more point cloud information. Dense reconstructions<sup>34,35</sup> need GPU acceleration because of the high computational cost involved. While semi-dense only needs multi-threading optimization. The mapping algorithm is implemented according to Raul's work<sup>46</sup> and is built upon our PL-based visual SLAM.

## PL-based visual SLAM

### Heuristic extraction and matching of lines

PTT inspection environment is full of line structures. We compared commonly used line detection methods in the real-world environment and finally chose line segment detector (LSD)<sup>47</sup> due to its good performance in SLAM system. This is demonstrated in the experiment section. LSD, an  $O(n)$  line detector, is able to adapt to a certain degree of environmental change without parameter tuning and provides sub-pixel accuracy. However, lines detected by LSD on the PTT usually have unstable end points. Besides, LSD often divides a line into several segments. This intrinsic problem becomes more serious especially in the wild due to illumination variation. It causes failures for line extraction, matching, and tracking. Considering the fact that line structures of PTT are intersected at different corners brings abundant and remarkable intersections in the images. So we take advantages of the intersections to improve the performance of line detection in a heuristic way shown as follows:

- 1) The segments which should belong to one straight line are merged based on their differences in direction and distance. Let  $d_1$  indicates the distance between the two midpoints and  $d_2$  represents the minimum distance between the end points. If  $d_1$  and  $d_2$  are smaller than the given threshold that is experientially determined by the minimum length value between the shortest line and a tenth of the bounding box longitudinal edge of TD, and the direction difference is smaller than  $5^\circ$ , then the two segments are fused since they are probably the two candidates of one line.
- 2) For further fusing, the Euclidean distance between Line Band descriptors (LBDs) of two segments can be used. For each segment  $l_x$ , it has the smallest distance  $\varepsilon_1$  with segment  $l_1$  and the second smallest distance  $\varepsilon_2$  with segment  $l_2$ . If  $\varepsilon_1/\varepsilon_2 < 0.2$ , then the  $l_x$  and  $l_1$  can be fused. Additionally, the lines, that are close to but do not meet the fusion conditions, are adjusted to be parallel, since most of them are two sides of the same PTT linear structure.
- 3) For stable end points, the intersection points of two line segments are determined. The intersection can be on the extension line. It must be inside the tower area and the distances from the intersection point to the nearest end points of the two line segments must be less than the minimum length of the two segments. So that the extension line doesn't exceed the original lines. Besides, to reduce noise, the lengths of the two segments need to be empirically greater than one-fifth of the transverse edge of the TD bounding box.

LBD is an effective and robust local appearance-based method to find correspondences between lines. However, the appearance of the PTT has some similarity that leads to wrong matching. Therefore, we introduce a geometric matching criterion (GMC) of adjacent frames to effectively improve accuracy of LBD-based line matching. The GMC retains line matches which satisfy the following conditions:

- 1) The segments should have similar length.
- 2) The angle between two lines is less than a threshold.
- 3) The distance between the end points shouldn't exceed a certain threshold. In distance measurement, the two end points should be distinguished into different points according to the line direction that is determined by which side of the line segment is thicker.

### PL-based BA optimization

The keyframe-based SLAM architecture relies heavily on sparse nonlinear optimization (BA), since it is of vital importance to precision of motion and structure. After heuristic extraction and matching of PTT lines, we obtain accurate line matches and stable line end points. To integrate line features within the BA to further improve optimization accuracy in the PTT environment, we next describe the line parameterization, utilization of intersections, and the proposed error function.

Unlike the reprojection errors used in ORB point features, the distance between the projected end point and detected end point cannot be directly used since the 3-D lines may not be fully detected in the image or they are partially occluded. These situations possibly occur in a harsh wild inspection environment. So we use the point-to-line distance which can be divided into projected-point-to-detected-line distance and detected-point-to-projected-line distance. In the first case, the line measurement error  $El_{ik}$  for the  $k$  th line in the  $i$  th keyframe is represented by

$$\begin{aligned}
 \exp(\xi_{iw}) &= \exp(\xi_{i,i-1}) \cdot \exp(\xi_{i-1,w}) \\
 \xi_{iw} &= \begin{bmatrix} \rho_{iw} \\ \phi_{iw} \end{bmatrix}, \rho \in \mathbb{R}^3; \phi \in so(3) \\
 El_{ik} &= [l'_{ik} \cdot K \exp(\hat{\xi}_{iw}) P'_{wk} l'_{ik} \cdot K \exp(\hat{\xi}_{iw}) Q'_{wk}]^T \\
 \hat{\xi}_{iw} &= \begin{bmatrix} [\phi_{iw}]_{\times} & \rho_{iw} \\ 0^T & 1 \end{bmatrix} \\
 [\phi_{iw}]_{\times} &= \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}
 \end{aligned} \tag{1}$$

where  $\phi_{iw} = [\phi_1, \phi_2, \phi_3]$  represents the  $se(3)$  rotation of the  $i$  th PTC pose in the world frame,  $\rho$  represents the  $se(3)$  translation of the  $i$  th PTC pose in the world frame,  $K$  is the PTC intrinsic matrix,  $l'_{ik}$  denotes the homogeneous



representation of the  $k$  th infinite line in the  $i$  th keyframe,  $\xi_{iw}$  refers to the  $se(3)$  pose of  $i$  th keyframe, and  $\xi_{i,i-1}$  refers to the  $se(3)$  pose transformation from  $i-1$  to  $i$ .  $P'_{wk}$  and  $Q'_{wk}$  are the homogeneous coordinates of the 3-D end points of  $k$  th 3-D line in the world coordinate system. However, using the two end points to represent a line is a non-minimal line parametrization, which doesn't have good performance in terms of optimization accuracy and convergence.<sup>42</sup> Therefore, we define the  $El_{ik}$  as the distance from the 2-D detected end points to the projected line, in which case the line in 3-D space is treated as an infinite line with four DoFs, and it can be parameterized by the four-parameter-based orthonormal form<sup>48</sup> compactly. The  $El_{ik}$  is shown as equation (2)

$$\begin{aligned}
 \exp(\xi_{iw}) &= \exp(\xi_{i,i-1}) \cdot \exp(\xi_{i-1,w}) = \begin{bmatrix} R_{iw} & t_{iw} \\ 0 & 1 \end{bmatrix} \\
 L_{ck} &= \begin{bmatrix} R_{iw} [t_{iw}]_{\times} R_{iw} \\ 0 & R_{iw} \end{bmatrix} L_{wk} \\
 L_{ck} &= \begin{bmatrix} n_{ck} \\ d_{ck} \end{bmatrix} \\
 [t_{iw}]_{\times} &= \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix} \\
 l'_{ik} &= \begin{bmatrix} f_y & 0 & 0 \\ 0 & f_x & 0 \\ -f_y c_x & -f_y c_y & f_x f_y \end{bmatrix} n_{ck} \\
 K &= \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \\
 El_{ik} &= [p'_{ik} \cdot l'_{ik} \quad q'_{ik} \cdot l'_{ik}]^T
 \end{aligned} \tag{2}$$

where  $t_{iw} = [t_1, t_2, t_3]$  represents the translation of the  $i$  th PTC pose in the world frame,  $R_{iw}$  represents the rotation of the  $i$  th PTC pose in the world frame,  $K$  is the PTC intrinsic matrix,  $n_{ck}$  and  $d_{ck}$  separately denote the normal and orientation vectors of the  $k$  th 3-D line in the camera frame,  $p'_{ik}$  and  $q'_{ik}$  denote the homogeneous coordinates of the end points of the  $k$  th line segments in the  $i$  th keyframe, and  $L_{wk}$  and  $L_{ck}$  are the  $k$  th 3-D line in the world and camera coordinate frames, respectively. The state  $\xi_{iw}$  and orthonormal parameters of the 3-D line can be optimized by minimizing the  $El_{ik}$ . For derivation details, readers can refer to He's work.<sup>49</sup> However, the BA optimization doesn't take effect on the end points of 3-D lines, because lines are regarded infinitely long. So the 2-D end points matched in different keyframes are back-projected and fused to trim the corresponding 3-D line. The fusion strategy of Zhang's work,<sup>42</sup> which can further alleviate the occlusion problem and improve map accuracy, is applied in our approach.

There are a good deal of corner points in the PTT structure. Most of them are picked out by the heuristic line extraction method so as to stabilize the end points. The corner points are PTT salient features and can be quickly extracted based on the line detection results. These advantages make the corner an excellent feature for our framework. Therefore, we try to add them within our SLAM system as if they were ORB features. So that the corners can adapt to most of the SLAM architectures and the number of feature points can increase a lot to improve algorithm robustness without losing runtime efficiency. ORB feature is designed by adding orientation and multi-scale information on the basis of FAST<sup>50</sup> corners and it has a 256-bit rotated BRIEF<sup>51</sup> descriptor according to its direction. However, the BRIEF descriptor of corner point can't be directly computed since the extraction process of the corner is different from that of the ORB. Therefore, we design a simple and efficient method leveraging local image patch of the corner point to provide the necessary orientation and pyramid scale information for corner BRIEF construction. In our approach, the acute angular bisector of the corner point is adopted to represent the direction information. Different sizes of rectangular blocks, which take the corner as the center, are used to describe the multi-scale information. This is able to simplify the complex scale operations. After that, there is no distinction between the ORB features and the corner features. The point measurement error  $Ep_{ij}$  for both features can be uniformly expressed as equation (2)

$$Ep_{ij} = x_{ij} - K \exp(\hat{\xi}_{iw}) X_{wj} \tag{3}$$

where  $X_{wj}$  is the  $j$  th 3-D point in world coordinates and  $x_{ij}$  represents the  $j$  th 2-D point observation in the  $i$  th keyframe. Then, the final optimization cost function  $C$  in combination with points and lines can be obtained as equation (4)

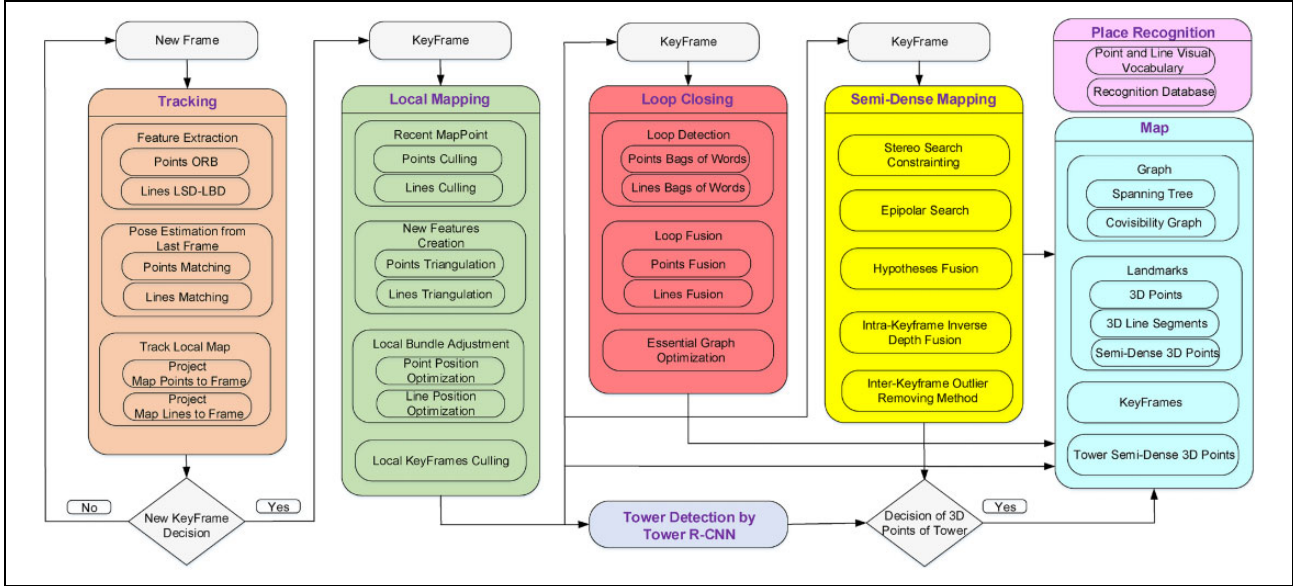
$$C = \sum_{ij} \lambda_p (Ep_{ij}^T \Sigma p_{ij}^{-1} Ep_{ij}) + \sum_{i,k} \lambda_l (El_{ik}^T \Sigma l_{ik}^{-1} El_{ik}) \tag{4}$$

where  $\lambda_p$  and  $\lambda_l$  denote the Huber loss functions, and  $Ep_{ij}^{-1}$  and  $El_{ik}^{-1}$  represent the inverse covariance matrices, which account for uncertainties of points and lines, respectively. They are computed by the Jacobians of the error functions ( $Ep_{ij}$  and  $El_{ik}$ ) with respect to the observations which include points  $p_{ij}$  and line segments  $l_{ik}$  in the  $i$  th keyframe. The computation process is shown in equations (5) and (6)

$$\Sigma p_{ij} \approx \frac{\partial Ep_{ij}}{\partial p_{ij}} \Sigma_o \frac{\partial Ep_{ij}}{\partial p_{ij}}^T \tag{5}$$

$$\Sigma l_{ik} \approx \frac{\partial El_{ik}}{\partial l_{ik}} \Sigma_o \frac{\partial El_{ik}}{\partial l_{ik}}^T \tag{6}$$

In the image, the uncertainties  $\Sigma_o$  are assumed to obey 2-D Gaussian distribution with standard deviations



**Figure 4.** Schematic diagram of proposed PL-TD framework. PL: point-line; TD: tower detection.

$\sigma_x = \sigma_y = 1$  pixel for both the points and the line segments.

### Overview of PL-TD

In this part, we briefly summarize our proposed PL-TD framework. As shown in Figure 4, PL-TD is composed of transmission tower localization and PL-based visual SLAM which is an extension of the ORB-SLAM. The framework contains five main threads: tracking thread, local mapping thread, loop closing thread, semi-dense mapping thread, and TD thread.

**The tracking thread.** The tracking thread is the visual odometry which estimates the poses of PTC. Besides, it determines when to add new keyframes. Firstly, a constant motion model and a window search strategy are used to guess the current PTC pose and initialize a coarse matching, respectively. Based on the matching, a reference keyframe which shares most features with the current frame is selected followed by a covisibility map of the keyframe is retrieved. Then, lines and points in the local map are projected to current frame to build more feature associations. Finally, the PTC poses are optimized by the proposed PL-based motion-only BA. For close proximity inspection, the policy of keyframe insertion is designed very generously so that the tracking is more robust. Redundant keyframes can be discarded subsequently in the local mapping thread. In addition, the relocalization module of ORB-SLAM is abandoned and once the tracking is lost, the UAV will hover for safety.

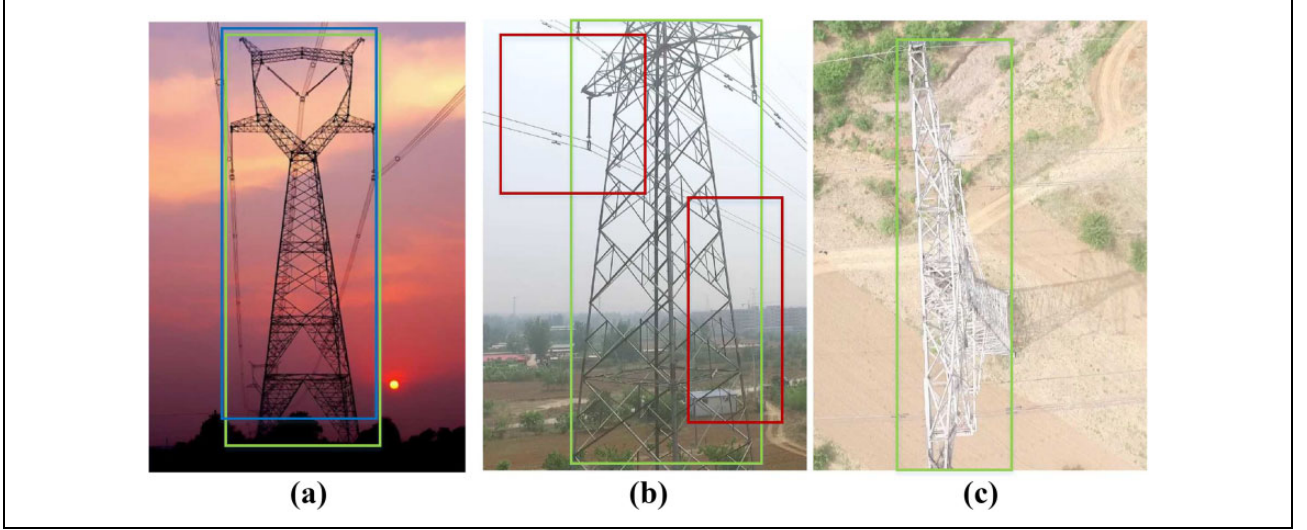
**The local mapping thread.** The local mapping thread performs PL-based local BA to optimize the local map that is related to the newly added keyframe. If a keyframe is

determined, the local mapping thread triangulates map points and lines according to the matching information collected in the tracking thread. Then, the points and lines, seen from less than three keyframes or in less than 25% of the frames from which they are expected to be seen, are discarded. Additionally, based on the number of co-visible points and lines, the local mapping is also responsible for removing redundant keyframes.

**The loop closing thread.** The loop closing thread checks whether the loops are detected and is in charge of correcting the drift errors. Based on the ORB and LBD features extracted from a large set of inspection pictures, the visual vocabulary of points and lines is trained off-line by Distributed Bag-of-Words (DBOW),<sup>52</sup> respectively. An online database-reserving bag of words vector of keyframes is established for loop candidates detection. For each detected loop candidate, based on the corresponding points, a RANSAC<sup>53</sup> scheme is performed to find a relative Sim(3) transformation<sup>54</sup>  $S$  using the method of Horn.<sup>55</sup> If  $S$  is found, the  $S$  will be optimized by minimizing the reprojection errors in both keyframes to find more correspondences. If there are enough correspondences, the loop can be accepted. Then, both sides of the loop are aligned and duplicated points and lines are fused. Finally, a PL-based pose graph optimization is performed globally.

**The semi-dense mapping thread.** The semi-dense mapping thread searches correspondences of pixels in high-gradient areas of keyframes. Due to a wide baseline between keyframes, the search of pixel correspondences is improved by an intra-keyframe inverse depth fusion and an inter-keyframe outlier removing method, which finally bring an accurate reconstruction with few outliers. For more details, readers can refer to Raul's work.<sup>46</sup>





**Figure 5.** Success and error judgement for detection. The green bounding boxes represent ground truth. (a) The blue bounding box represents TP, (b) the red bounding boxes represent FP, and (c) no detection bounding box (missed detection) represents FN. TP: true positive; FP: false positive.

*The TD thread.* The TD thread detects the PTT or part of the PTT fast and accurately by the proposed Tower R-CNN. In combination with the correspondences between the 2-D pixels and the 3-D semi-dense points, the PTT can be localized well and represented by enough point clouds in 3-D space for refined inspection.

## Experiments and analyses

### Transmission TD experiment

*Experiment setup.* For this experiment, 1300 sheets of transmission tower pictures were collected from refined inspection videos and annotated manually. The data set considers different backgrounds, illumination, image resolutions, observation viewpoints, and occlusion conditions. To verify the validity of our algorithm, we conducted comparisons of TD between the proposed Tower R-CNN and the three state-of-the-art DL-based detection frameworks: Faster R-CNN,<sup>45</sup> single multibox detector (SSD),<sup>56</sup> and YOLOv2.<sup>57</sup> We adopted 10-fold cross-validation<sup>58</sup> to find the best models. Following this scheme, the data set is randomly partitioned into 10 subsets with equal size, then the training and validation are conducted for 10 times. Each time, a different subset is taken out for validation while the remaining union of nine folds is used for training. We used the Caffe framework<sup>59</sup> to implement the training process on a GTX TitanX GPU and the validation process on TX2.

*Quantitative evaluation methodology.* For quantitative evaluation of the detection task, we adopted the intersection over union, which is the evaluation standard of the PASCAL Visual Object Classes challenge.<sup>60</sup> A detection is to be considered correct when the bounding box overlap ratio  $r$

between the ground truth  $B_{gt}$  and the predicted  $B_p$  exceeds 50%, in which  $r$  is defined by the following formula

$$r = \frac{\text{area}(B_{gt} \cap B_p)}{\text{area}(B_{gt} \cup B_p)} \quad (7)$$

where  $\text{area}(B_{gt} \cup B_p)$  represents the union of the ground truth bounding box, and the predicted bounding box and  $\text{area}(B_{gt} \cap B_p)$  denotes their intersection. Therefore, according to  $r$ , detections can be divided into three types: true positive (TP) (tower is correctly detected), false positive (FP) (background is mistaken as tower), and false negative (FN) (tower is not detected). The three different cases are illustrated in Figure 5.

Further, the precision and recall are employed as follows

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (8)$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (9)$$

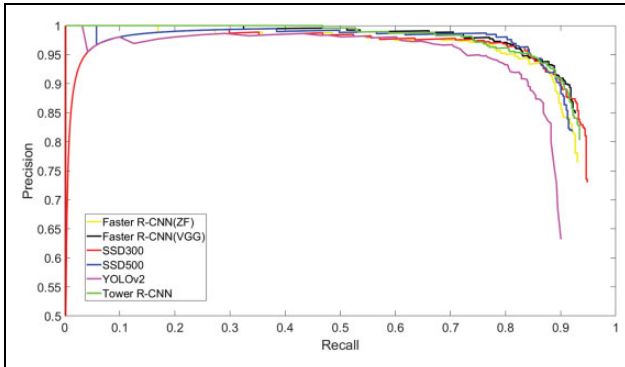
Each predicted bounding box has a confidence value between 0 and 1 to describe the degree of certainty. If the detection confidence is higher than a given threshold, it can be classified as TP. Otherwise, it is FP. So based on the different confidence thresholds, we can obtain many value pairs of the precision and recall. Subsequently, the precision–recall curve can be plotted. The average precision (AP), summarizing the shape of the precision–recall curve, is defined as the mean of 11 equally spaced recall levels  $[0, 0.1, \dots, 1]$

$$\text{AP} = \frac{1}{11} \sum_{r \in \{0, 0.1, \dots, 1\}} f(r) \quad (10)$$

**Table 2.** Performance of the inspection UAV.

Detection method	AP (%)	FPS
Faster R-CNN (VGG16)	89.8	0.8
Faster R-CNN (ZF)	88.6	2
SSD300	87.5	6
SSD512	88.1	2
YOLOv2	86.8	5.6
Tower R-CNN	89.8	5

FPS: frames per second; AP: average precision.

**Figure 6.** Precision–recall curve for TD. TD: tower detection.

where  $f(r)$  represents the precision at recall  $r$ , and AP is approximately equal to the area size under the precision–recall curve.

**Experimental results.** The comparison was made from the following three aspects: runtime, AP, and the false detection rate (precision–recall curve). All results are from the best models after 10-fold cross-validation. As shown at Table 2, SSD300 has the fastest runtime and YOLOv2 has a speed of 5.6 frames per second (FPS), but their AP is relatively low. Thus, they may have a low overlap ratio  $r$  and unstable bounding boxes due to environmental interference.

With respect to the precision–recall curve, as illustrated in Figure 6, Tower R-CNN denoted by the green line maintains a 100% precision over a fairly wide range of recall, which clearly surpasses Faster R-CNN, SSD, and YOLOv2. At this point, Faster R-CNN, SSD, and YOLOv2 encounter different degrees of false detection, even at a low level of recall. The high precision, namely no false detection, brings significant safety to close proximity navigation around PTT. Therefore, the proposed Tower R-CNN can provide reliable and real-time TD results for inspection task.

### Line extraction and matching experiment

For line extraction, we evaluated four effective line detection methods which are commonly used in the literatures: Progressive Probabilistic Hough Transformation (PPHT),<sup>61</sup> LSD,<sup>47</sup> EDLine,<sup>62</sup> and fast line detector (FLD).<sup>63</sup> To

evaluate the detection performance, we tested 100 images with  $640 \times 480$  resolutions in real-world close proximity PTT inspection environments and statistically analyzed the results in detail.

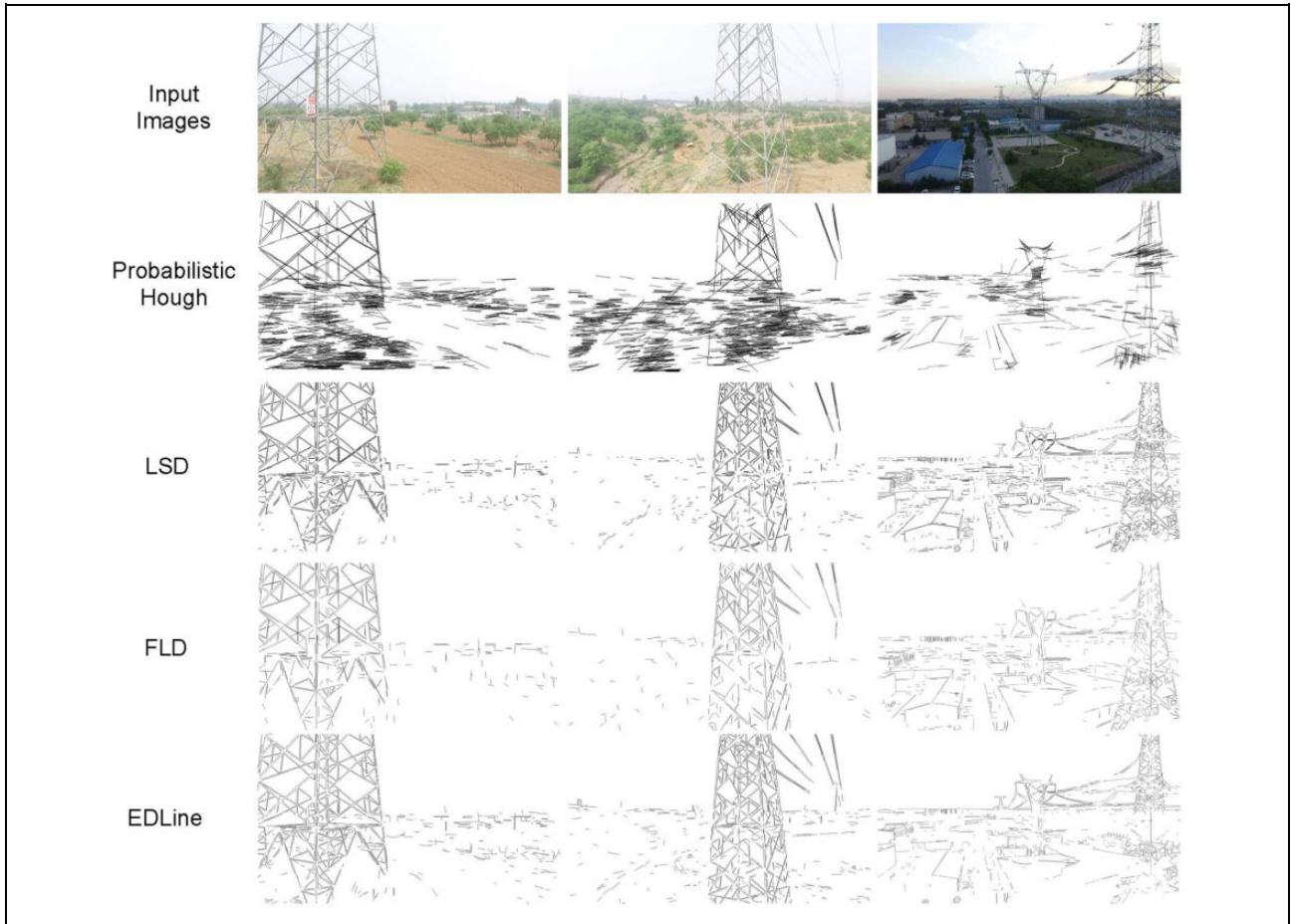
The evaluation of line detection is usually based on the extraction speed and line quality which consists of quantity, length, and repetition. All four line detection algorithms are written in the C++ language and run on the Nvidia TX2. As can be seen from Figure 7, it is difficult to detect the complete linear structure of a tower by applying PPHT. The straight lines are often contaminated by line-like noise. Besides, to some extent, FLD and EDLine are more susceptible to the environmental influences than LSD, and LSD has more obvious line detection results. As illustrated in Table 3, EDLine detects the largest number of lines. However, the lines detected have many repeated results that influence the line matching. LSD has few repeated results and detects more line segments than FLD but is slower than FLD and EDLine. Whereas, the accurate straight line detection of LSD can provide great safety, which is most important for close proximity navigation. Furthermore, the UAV will not fly too fast during close proximity PTT inspection, the frame rate of the camera doesn't have to be very high. Summing up the above, we choose LSD as the line detection method.

Based on the proposed heuristic line extraction method, the accuracy of tower line detection is further improved after LSD. Figure 8(b) and (d) are the heuristic line detection results of Figure 8(a) and (c), respectively. Compared with Figure 8(a) and (c), the red circles in Figure 8(b) and (d) show more stable end points, reflecting the intersections of the tower. Moreover, the lines circled in green are more discriminative and parallel. It is consistent with the actual appearance of the tower. In further, the lines circled in blue are elongated, overcoming the disadvantages that LSD often divides a line into several segments. The heuristic method reduces the risk of failure for line feature matching and tracking and provides stable observations for optimization.

Figure 9 shows several line matching examples in local maps. The same numbers marked on lines in different key-frames indicate that they are matched. The matching results are completed by the GMC of adjacent frames and the LBD descriptors. In further, the matching criterion can be accelerated by a guided search (GS) which is based on a predicted velocity motion model. A quantitative evaluation of adding the GMC and GS for matching was carried out in a video sequence. The image resolution is  $640 \times 480$ . As illustrated in Table 4, the proposed matching approach makes data association significantly robust with few wrong matches and a little time increase.

### Experiments in synthetic scene for UAV and PTT localization

This experiment was conducted to verify the accuracy of UAV self-localization which takes advantage of the



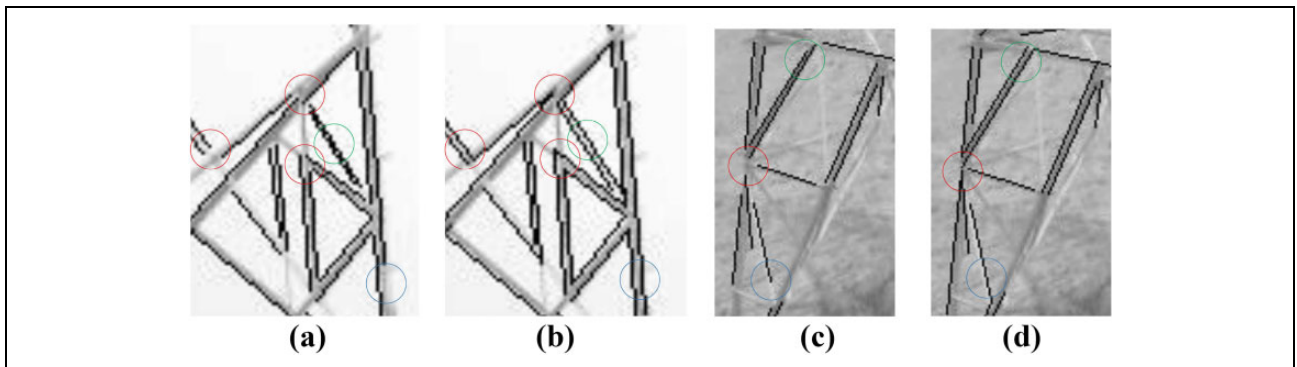
**Figure 7.** Results of different line detection in typical close proximity inspection situations.

**Table 3.** The results of line detection algorithms.

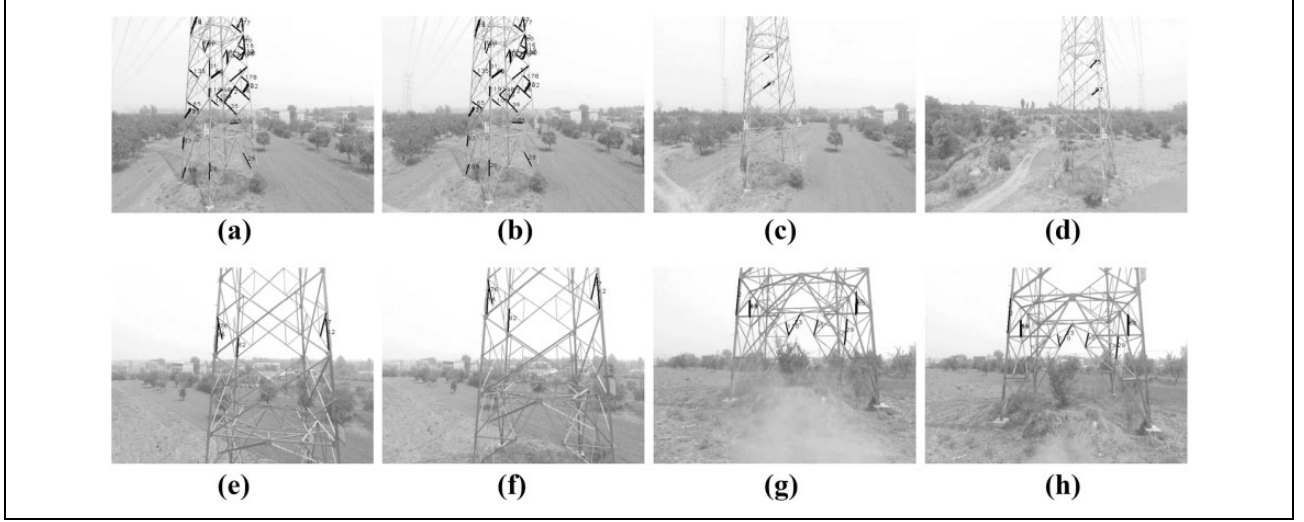
	LSD	FLD	EDLine	PPHT
Average run time (ms/image)	18.6	13.5	15.5	16.4
Average line numbers	206.5	111.6	255.2	235.7
Average line length (pixels)	56.3	86.4	92.6	108.3
Repeat lines	few	few	many	many

LSD: line segment detector; PPHT: Progressive Probabilistic Hough Transformation.

proposed point-to-line distance and the intersections of lines. Two-hundred camera positions were sampled at a fixed time interval from the proposed two paths. At each position, the camera can see part of the PTT according to the camera projection model. The observation of the line segment on the image plane can be obtained by projecting the PTT line structure to the camera plane. Besides more 3-D points are added around the tower corners and projected as the ORB point features. Then, we added Gaussian



**Figure 8.** Results of heuristic line extraction of PTT. PTT: power transmission tower.



**Figure 9.** Results of line matching in local maps. (b), (d), (f), (h) are the pictures with different viewpoints in the covisibility graph<sup>38</sup> of (a), (c), (e), (g), respectively. The matched lines are denoted with same numbers.

**Table 4.** The results of line matching approaches.

	LBD	Adding GMC	Adding GMC and GS
Average run time (ms/image)	21.2	24.7	22.4
Average wrong matching	5.3	1.6	1.2

LBD: line band descriptor; GMC: geometric matching criterion; GS: guided search.

white noise with a variance of 10 pixels to the point and end points of lines in the image. In addition, two other Gaussian white noise models with a variance of 3 m and 5° are separately imposed on the translation and rotation of camera poses. The collection process of simulation data is shown in Figure 10. Actually, we conduct the pose graph optimization, and the 3-D corner points of PTT are fixed in this experiment.

In the experiment, we adopted Levenberg–Marquardt algorithm in the Ceres optimization library,<sup>64</sup> which is developed by Google, as an optimization solution tool. For a fair comparison, the optimizer iterates the same number of steps. The optimized positions of 200 cameras are shown in Figure 11. In the synthetic scene, our approach based on fusion of points and lines makes the UAV self-positioning more robust to noises. It outperforms than the method based on ORB point feature only and the method based on line and intersection features only. The point-to-line distance and the intersections of lines provide extra useful constraints for UAV odometry.

With respect to the quantitative evaluation metric, we employ the Root Mean Square Error (RMSE) of Relative Pose Error (RPE) to evaluate the performance of our approach. The RPE measures the trajectory accuracy over a constant time interval  $\Delta$  and reflects the drift of the trajectory. It is defined as equation (11) at time step  $i$

$$E_i = (Q_i^{-1}Q_{i+\Delta})^{-1}(P_i^{-1}P_{i+\Delta}) \quad (11)$$

where  $P_1, \dots, P_n$  is a pose sequence from the estimated trajectory and  $Q_1, \dots, Q_n$  is another pose sequence from the ground truth trajectory. Assume we have  $n$  camera poses, then  $m = n - \Delta$  individual RPE can be obtained. From these errors, the RMSE over all time indices is defined as equations (12) and (13) for translational and rotational components, respectively

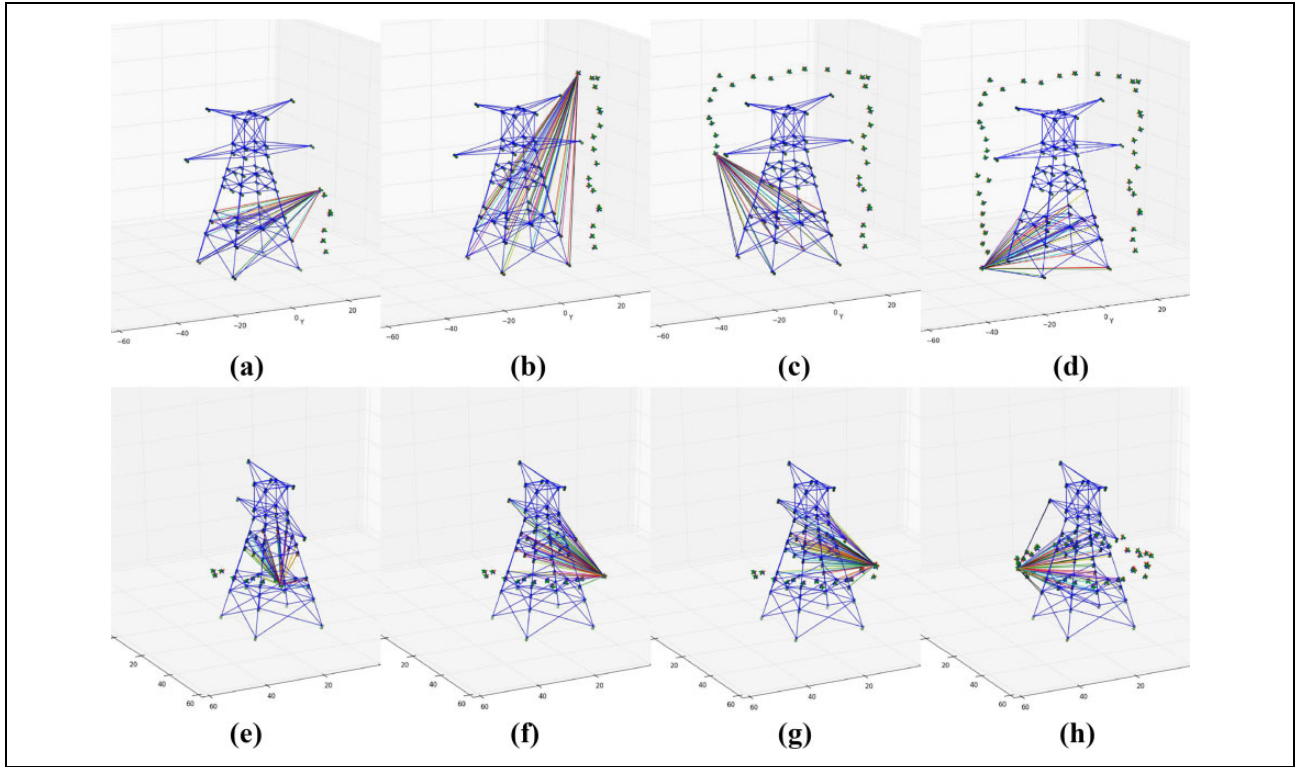
$$\text{RMSE}_{\text{translation}} = \left( \frac{1}{m} \sum_{i=1}^m \|\text{trans}(E_i)\|^2 \right)^{1/2} \quad (12)$$

$$\text{RMSE}_{\text{rotation}} = \left( \frac{1}{m} \sum_{i=1}^m \|\text{rota}(E_i)\|^2 \right)^{1/2} \quad (13)$$

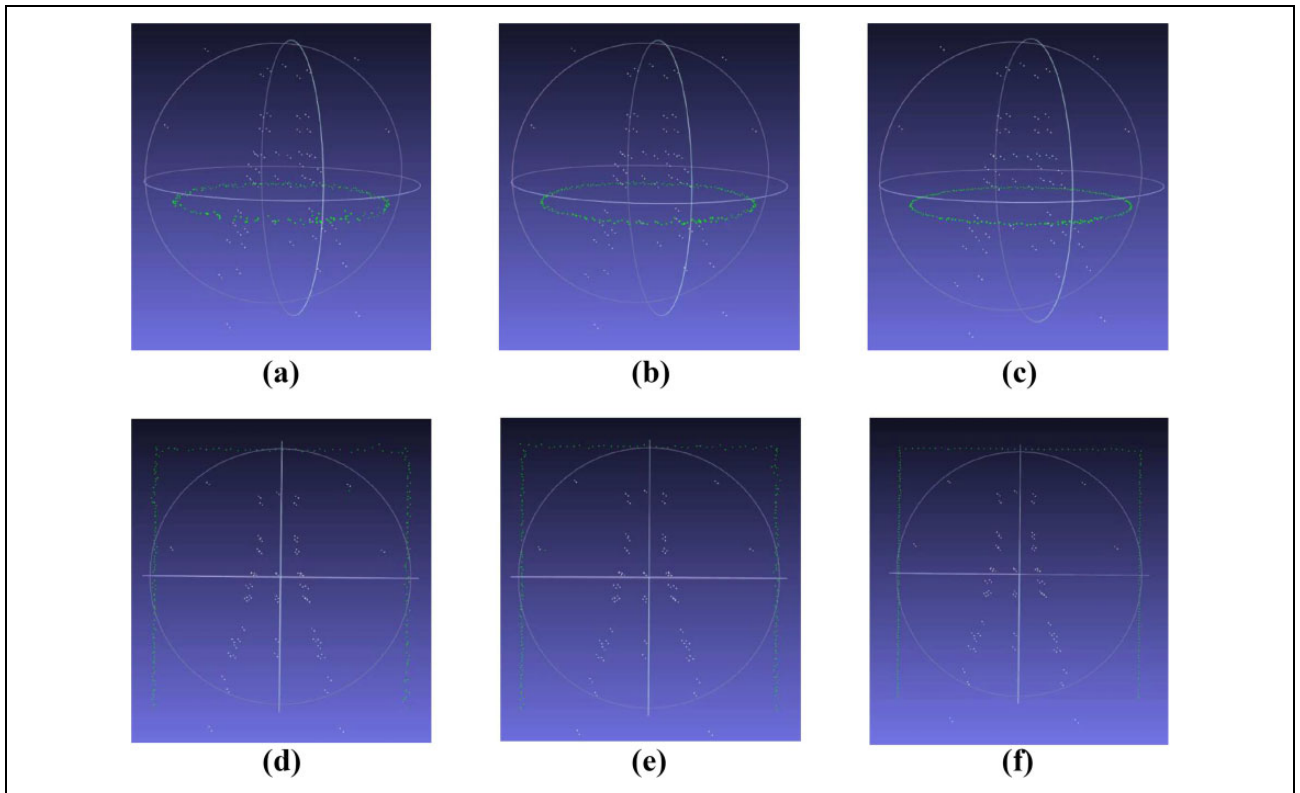
where  $\text{trans}(E_i)$  and  $\text{rota}(E_i)$  denote the displacement and rotation angle decomposed from the attitude matrix  $E_i$ . The first two rows of Table 5 show the average error statistics of 20 simulation experiments. From the experimental results, it can be seen that the translational and rotational errors of point-based optimization are reduced after adding constraints of the lines and their intersections to the cost function. Our method improves the UAV localization precision. After 20 simulation experiments, the third row of Table 5 shows the average run time of the motion-only BA for 200 camera positions. The run time increases 5 ms after adding optimization for lines and corners.

In the close proximity navigation around PTT, the precision of each reconstructed 3-D feature point of the tower is not needed. The accuracy is mainly determined by the 2-D TD in the first experiment. Based on the experimental setup above, we added different levels of noises to the 3-D tower corners and tested the robustness of tower





**Figure 10.** (a), (b), (c), (d) and (e), (f), (g), (h) show the generation of simulation data with Gaussian white noises along the CIRP and the TOLP, respectively. A PTT corner point is in the camera's field of view if there is a color line connecting it and the camera. PTT: power transmission tower; CIRP: circumvolant path; and TOLP: takeoff and land vertically path.



**Figure 11.** The results of motion-only BA. (a) and (d) The optimization results of using ORB point feature only. (b) and (e) The results of using point-to-line distance and the intersections of lines. (c) and (f) The results of using ORB points, lines, and intersections of lines. The green points denote the camera positions. The white points represent the PTT corner points. The white points make up the basic shape of the tower. BA: bundle adjustment; PTT: power transmission tower.

**Table 5.** The first two rows are the RMSE of RPE after minimizing different losses and the last row records the maximum variances of Gaussian white noise that the optimization can tolerate.

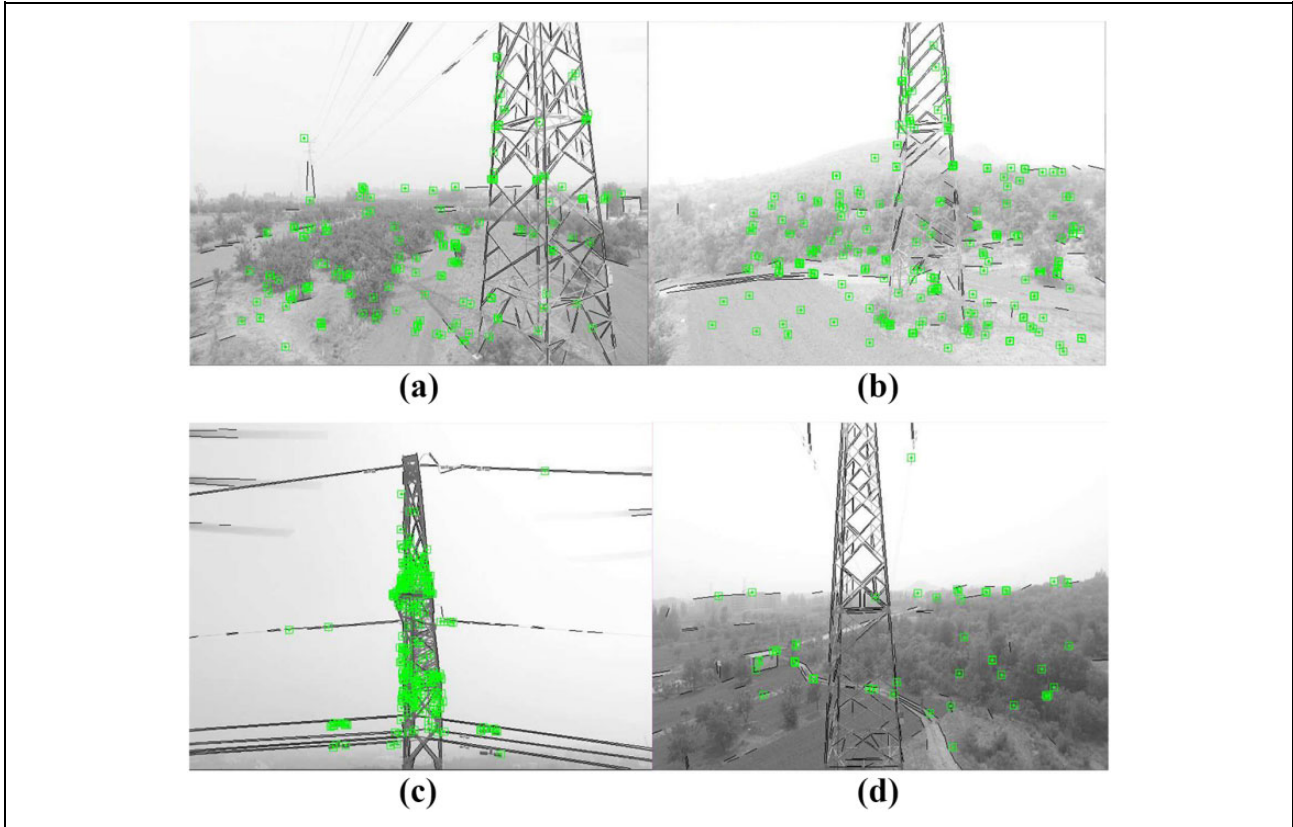
	BA with points	BA with lines and corners	BA with points, lines, and corners
Translational RMSE (m)	0.092	0.083	0.070
Rotational RMSE (rad)	0.047	0.055	0.034
Average run time (ms)	21.7	23.1	26.8
Maximum tolerable noise variance (m)	6.5	7.1	8.8

RMSE: root mean square error; RPE: relative pose error.

reconstruction. The tower corner points are restored by global BA. We define that the optimization converges if the average distance between optimized PTT corners and the ground truth corners is less than 0.1 m and the gradient variation is smaller than a threshold. The last row of Table 5 records the maximum variance of Gaussian white noise that the three methods can tolerate for successful convergence. It can be seen that our proposed method performs best in terms of reconstruction robustness.

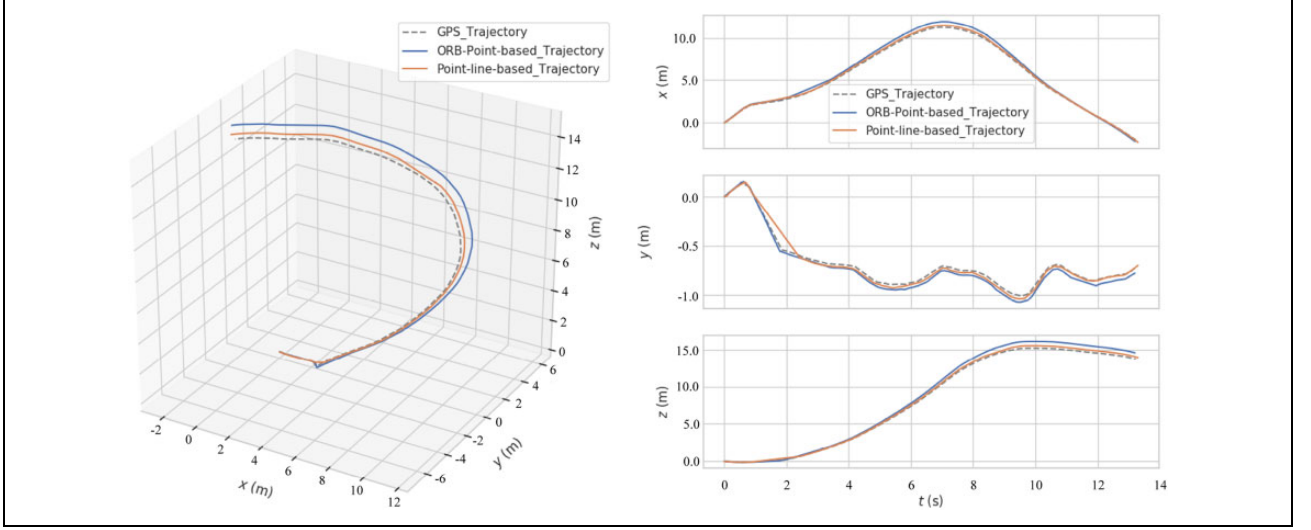
### Experiments in real-world scenes

We conducted several experiments in the field to validate our proposed monocular vision-based environmental perception approach. A transmission tower which has a typical 220-kV double circuit lattice steel structure is selected as an inspection target. It has a height of 35 m and a base width of 6 m and can be approximately enveloped by a  $8 \times 8 \times 35 \text{ m}^3$  (length, width, and height) cuboid. We operated the UAV to fly along our proposed CIRP and TOLP. Besides, we deliberately operated the camera to make the tower appear in the field of camera view and have a random position in image. In further, the trajectory of UAV can be recorded based on an accurate differential GPS system of the UAV. The GPS data of transmission tower is provided by a power company and based on WGS84 (World Geodetic Coordinate System 1984). It is in principle possible to evaluate the accuracy of the experimental trajectory. However, the accurate time alignment between the ground truth and the estimation is difficult to obtain in the field environment. The time deviation mainly comes from the out-of-sync transmission of GPS data and image data in the system. Considering the fact that the speed of UAV is slow during refined inspection, the deviation of time caused by system is still within the acceptable range in this large inspection scenario.



**Figure 12.** Sample images in real-world refined PTT inspection. (a) and (b) are pictures captured along the CIRP path; (c) and (d) are pictures captured along the TOLP path. The green points denote the detected keypoint and the black lines denote the detected line feature. PTT: power transmission tower; CIRP: circumvolant path; and TOLP: takeoff and land vertically path.





**Figure 13.** Comparison of the planned trajectory (recorded by the differential GPS system), the ORB keypoint-based trajectory and our PL-based trajectory. The three trajectories are around the tower and contain the viewpoint of Figure 12(d) which has fewer ORB keypoints. GPS: global positioning system. PL: point-line.

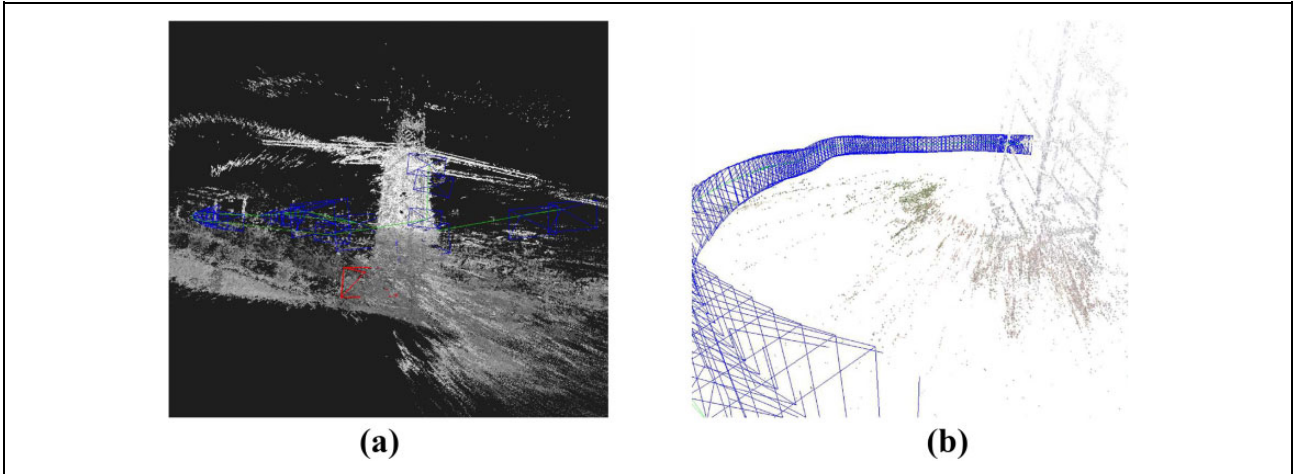
Figure 12 shows the results of image processing of the SLAM framework during navigation. As illustrated in Figure 12 (d), when the number of detected ORB keypoints is small, the number of detected PTT lines is prominent. To further demonstrate the effects of PTT lines for UAV self-localization, the UAV was operated to fly around the PTT and we recorded this planned trajectory by the differential GPS system. The ORB keypoint-based trajectory and our PL-based trajectory are calculated from the collected video. The three trajectories are aligned by correcting the scales and they are compared in Figure 13. It can be seen that our approach combining line features are closer to the ground truth. After 10 experiments, Table 6 shows the average RMSE errors of the two vision-based trajectories with respect to the planned trajectory, respectively.

**Table 6.** Comparison of RMSE errors of the RPE.

	ORB keypoint-based trajectory	PL-based trajectory
Translational RMSE of RPE (m)	0.938	0.514
Rotational RMSE of RPE (rad)	0.213	0.152

RMSE: root mean square error; RPE: relative pose error; PL: point-line.

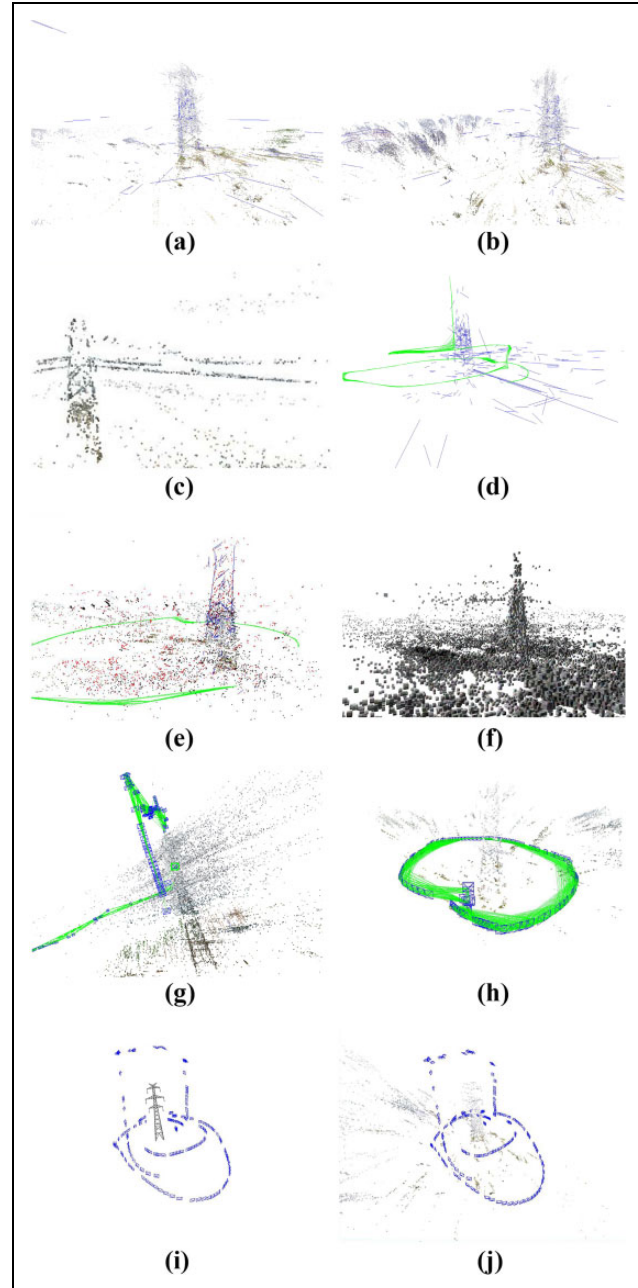
In terms of the reconstruction results of the PTT and environment, we compare the typical semi-dense reconstruction algorithms of LSD-SLAM and the keyframe-based semi-dense mapping.<sup>46</sup> LSD-SLAM is open source but the other is not. We implemented the semi-dense algorithm in C++ language and integrated it into our framework. As shown in Figure 14 (a), the map built by



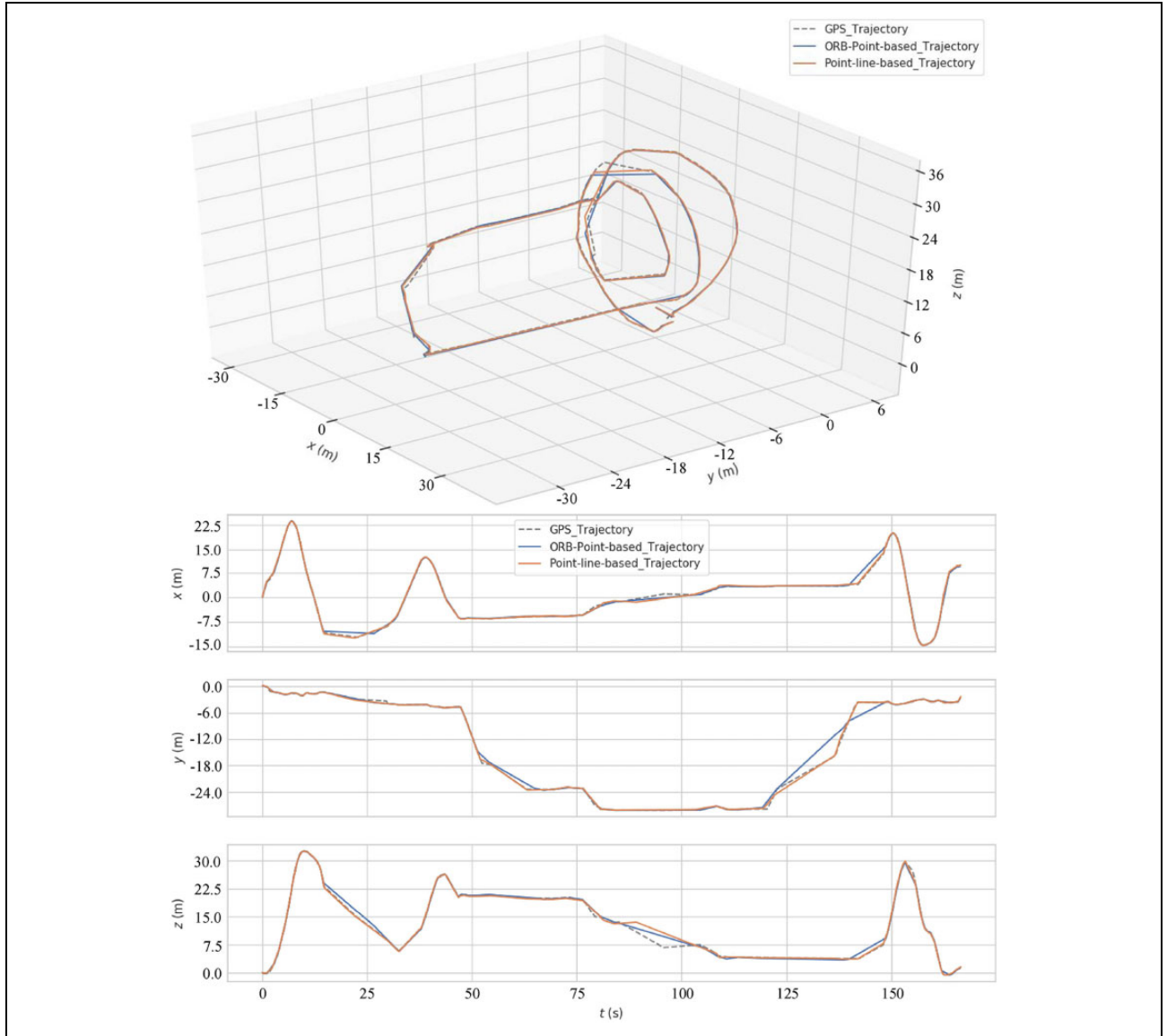
**Figure 14.** The comparison of reconstruction algorithm. (a) The result of LSD-SLAM; (b) the result of keyframe-based semi-dense mapping. The blue rectangles represent the camera poses. The green line represents the camera's trajectory. LSD: line segment detector.

LSD-SLAM contains a lot of noises and there are large deviation and jitter in UAV position estimation. LSD-SLAM is based on the photometric consistency hypothesis and localize the camera by optimizing directly over image pixel intensities. Therefore, LSD-SLAM is sensitive to illumination changes and the reconstruction accuracy is greatly degraded in the real-world inspection scene. In contrast, feature-based methods are able to match features with a wide baseline due to their good invariance to viewpoint and illumination changes. Camera poses are well optimized by BA over features. This allows to further obtain high quality and accurate reconstructions. As shown in Figure 14 (b), the keyframe-based semi-dense method recovers the PTT contours and reconstructs high-gradient areas which can reflect the line structures of PTT. This rich PTT representation is useful for UAV fixed-location inspection. The map requires no GPU and we implemented it only by adding a new thread.

Figure 15 shows the several complete UAV trajectories and PTT reconstructions in the real-world experiments. It can be seen that the triangulated 3-D lines and the semi-dense point cloud can accord with the PTT structures. To a certain extent, the accuracy of reconstructions of the environment is determined by the accuracy of estimation of camera poses. So, our approach can successfully estimate the UAV camera poses. Furthermore, we compared the ORB-SLAM scheme and our PL-based approach on a recorded video. The video completely covers the two paths of CIRP and TOLP. For a fair comparison, the parameters for point feature extraction were kept same. At each  $640 \times 480$  image, 2000 point features at 8 scale levels with a scale factor of 1.2 were extracted. Figure 15(i) shows the ground truth of PTT location and camera poses. Figure 15(j) shows the environment reconstruction and estimation of camera poses of our methods along the same path in Figure 15(i). We enlarged or reduced the camera positions by multiplying a suitable scale to minimize the mean square error between the sampled camera positions and the ground truth values. Figure 16 shows the comparison between the ORB-SLAM scheme and our PL-based approach along the CIRP and TOLP. It can be seen that our approach has a smaller error in more positions with respect to the GPS ground truth. Table 7 shows the average errors of 10 experiments. The translational RMSE of RPE of our approach is 0.393 m and meets the requirements of actual inspection. The PTT center is estimated by 3-D point clustering. The distance between the ground truth of PTT center and the detected center is 0.72 m. In terms of runtime, since the line features are extracted in parallel threads, the execution time on TX2 will not increase much. The UAV requires 105.1 ms per image, which satisfies the inspection requirements.



**Figure 15.** (a) and (b) The 3-D line feature and semi-dense mapping result; (c) the semi-dense mapping result when the UAV hovers near the top of the pole tower; (d) the 3-D line feature and the inspection paths around the tower; (e) a map in combination with the 3-D sparse keypoints, 3-D lines, and 3-D semi-dense points; (f) an octomap<sup>65</sup> which is built from the 3-D semi-dense point clouds; (g) the pose estimation of the UAV cameras along CIRP, the blue rectangles denote the camera positions and orientations; (h) the pose estimation of the UAV cameras along TOLP; (i) the camera trajectory and tower location recorded by GPS system; (j) the experimental results generated from our method on the images recorded from (i). UAV: unmanned aerial vehicle; CIRP: circumvolant path; and TOLP: takeoff and land vertically path.



**Figure 16.** The three trajectories cover the two paths of CIRP and TOLP. The trajectory denoted by the dashed line is recorded by the differential GPS system. The blue trajectory is calculated from the ORB keypoint-based scheme. The red trajectory is computed from our PL-based approach. CIRP: circumvolant path; and TOLP: takeoff and land vertically path; GPS: global positioning system; PL: point-line.

**Table 7.** Comparison of drift errors and runtime along the CIRP and TOLP.

	ORB-SLAM	Our approach
Translational RMSE of RPE (m)	0.485	0.393
Rotational RMSE of RPE (rad)	0.276	0.194
Distance error of PTT center (m)	0.96	0.72
Average run time (ms/image)	98.8	105.1

CIRP: circumvolant path; TOLP: takeoff and land vertically path; SLAM: simultaneous localization and mapping; PTT: power transmission tower.

## Conclusion and future work

In this article, a perception approach combining PL-based visual SLAM and TD is proposed for safe and autonomous

close proximity PTT inspection. The UAV takes advantage of enough perspective information provided by a monocular PTC to realize the reliable self-positioning and tower localization. All schemes are well implemented in an hierarchical embedded system. To make full use of the abundant line information in the PTT inspection environment, line extraction and matching are improved by a heuristic method, making them more suitable for tower linear structures. Besides, the intersections of lines are processed as ORB feature to increase algorithm robustness. To further improve accuracy of SLAM system, the cost function of BA optimization is proposed to combine ORB point feature with point-to-line distance and the intersections of lines. The loss function has more stable point-to-line distance

constraints and more point feature reprojection errors, making the framework more robust. To construct a useful map for navigation and simultaneously consider the real-time performance, the keyframe-based semi-dense mapping algorithm is implemented. To localize tower fast and accurately in 3-D space, a DL-based neural network is customized (Tower R-CNN) to detect part of or complete transmission tower in different viewpoints. Then, the contour and line-shaped structure of PTT can be reflected in map forming a rich representation. In addition, two safe paths, which can avoid collision with transmission lines and allow the UAV's PTC to observe the PTT, and electrical equipments, comprehensively, are proposed for refined inspection. Along the two paths, the whole perception strategy is validated in a synthetic scene. Finally, the designed inspection platform is tested in a real-world field environment, which achieves a satisfactory result.

In the future, we will investigate how to integrate a high precision inertial sensor with point and line features into the UAV system. In addition, our algorithm can be easily migrated into stereo vision inspection system to get the absolute distance information. Based on the above works, an online fault diagnose system will come true.

### Declaration of conflicting interests


The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China under grants 61271432, 61673378, and 61421004.

### ORCID iD

Jiang Bian  <https://orcid.org/0000-0002-5125-4882>

Xiaolong Hui  <https://orcid.org/0000-0003-0737-952X>

### Supplemental material

Supplemental video for this article is available online.

### References

1. Pagnano A, Höpf M, and Teti R. A roadmap for automated power line inspection, maintenance and repair. *Proc Cirp* 2013; 12: 234–239.
2. Katrasnik J, Pernus F, and Likar B. A survey of mobile robots for distribution power line inspection. *IEEE Trans Power Deliver* 2010; 25(1): 485–493.
3. Zhou Z, Zhang C, Xu C, et al. Energy-efficient industrial internet of uavs for power line inspection in smart grid. *IEEE Trans Ind Inform* 2018; 14(6): 2705–2714.
4. Li Z, Liu Y, Walker R, et al. Towards automatic power line detection for a uav surveillance system using pulse coupled neural filter and an improved hough transform. *Mach Vision Appl* 2010; 21(5): 677–686.
5. Mejias L, Correa JF, Mondragón I, et al. Colibri: a vision-guided uav for surveillance and visual inspection. In: *IEEE international conference on robotics and automation (ICRA)*, Rome, Italy, 10–14 April 2007, pp. 2760–2761. IEEE.
6. Ju HY, ChangHwan K, and Dong HK. Mono-camera based simultaneous obstacle recognition and distance estimation for obstacle avoidance of power transmission lines inspection robot. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Vancouver, Canada, 24–28 September 2017, pp. 6902–6907. IEEE.
7. Seok KH and Kim YS. A state of the art of power transmission line maintenance robots. *J Electr Eng Technol* 2016; 9: 1412–1422.
8. Wang L, Liu F, Wang Z, et al. Development of a practical power transmission line inspection robot based on a novel line walking mechanism. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Taipei, Taiwan, 18–22 October 2010, pp. 222–227. IEEE.
9. Pouliot N, Richard PL, and Montambault S. Linescout technology opens the way to robotic inspection and maintenance of high-voltage power lines. *IEEE Power Energy Technol Syst J* 2015; 2(1): 1–11.
10. Mills S, Aouf N, and Mejias L. Image based visual servo control for fixed wing uavs tracking linear infrastructure in wind. In: *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 6–10 May 2013, pp. 5769–5774. IEEE.
11. Zhou X, Jia Y, Zhao Q, et al. Experimental validation of a compound control scheme for a two-axis inertially stabilized platform with multi-sensors in an unmanned helicopter-based airborne power line inspection system. *Sensors* 2016; 16(3): 366.
12. Matikainen L, Lehtomäki M, Ahokas E, et al. Remote sensing methods for power line corridor surveys. *ISPRS J Photogramm* 2016; 119: 10–31.
13. Nguyen VN, Jenssen R, and Roverso D. Automatic autonomous vision-based power line inspection: a review of current status and the potential role of deep learning. *Int J Elec Power* 2018; 99: 107–120.
14. Li C. Research on electric power tower deformation monitoring technique using real time kinematic (rtk) method based on beidou navigation satellite system. *Elect Power Inform Communicat Technol* 2015; 13(12): 24.
15. Liu Y, Cai L, Wang S, et al. High-voltage line tower inclination monitoring based on tls. *Eng Survey Map* 2016; 8: 014.
16. Anjum S, Jayaram S, El-Hag A, et al. Detection and classification of defects in ceramic insulators using rf antenna. *IEEE Trans Dielect El In* 2017; 24(1): 183–190.
17. Zhai Y, Wang D, Zhang M, et al. Fault detection of insulator based on saliency and adaptive morphology. *Multimed Tools Appl* 2017; 76(9): 12051–12064.
18. Luque-Vega LF, Castillo-Toledo B, Loukianov A, et al. Power line inspection via an unmanned aerial system based on the quadrotor helicopter. In: *Mediterranean*



- Electrotechnical Conference*, Beirut, Lebanon, 13–16 April 2014, pp. 393–397.
19. Lam TM, Boschloo HW, Mulder M, et al. Artificial force field for haptic feedback in uav teleoperation. *IEEE Trans Syst Man Cybern A Syst Human* 2009; 39(6): 1316–1330.
  20. Hou X, Mahony R, and Schill F. Representation of vehicle dynamics in haptic teleoperation of aerial robots. In: *2013 IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 6–10 May 2013, pp. 1485–1491. IEEE.
  21. McFadyen A, Dayoub F, Martin S, et al. Assisted control for semi-autonomous power infrastructure inspection using aerial vehicles. *arXiv preprint arXiv:180402154* 2018.
  22. Sa I, Hrabar S, and Corke P. Inspection of pole-like structures using a visual-inertial aided VTOL platform with shared autonomy. *Sensors* 2015; 15(9): 22003–22048.
  23. Moore AJ, Schubert M, Rymer N, et al. *Uav inspection of electrical transmission infrastructure with path conformance autonomy and lidar-based geofences*. NASA Report on UTM Reference Mission Flights at Southern Company Flights, <https://ntrs.nasa.gov/search.jsp?R=20170011048> (accessed November 2016).
  24. Scaramuzza D, Fraundorfer F, and Siegwart R. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. In: *IEEE International Conference on Robotics and Automation, 2009. ICRA '09*, Kobe, Japan, 12–17 May 2009, pp. 4293–4299. IEEE.
  25. Cummins M. Highly scalable appearance-only slam-fab-map 2.0. *Proc Robot Sci Syst (RSS)* 2009.
  26. Konolige K, Agrawal M, and Sola J. Large-scale visual odometry for rough terrain. In: *Robotics research*. Berlin: Springer, 2010, pp. 201–212.
  27. Voigt R, Nikolic J, Hürzeler C, et al. Robust embedded ego-motion estimation. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, America, 25–30 September 2011, pp. 2694–2699. IEEE.
  28. Burri M, Nikolic J, Hürzeler C, et al. Aerial service robots for visual inspection of thermal power plant boiler systems. In: *2012 2nd International Conference on Applied Robotics for the Power Industry (CARPI)*, Zurich, Switzerland, 11–13 September 2012, pp. 70–75. IEEE.
  29. Nikolic J, Burri M, Rehder J, et al. A uav system for inspection of industrial facilities. In: *IEEE Aerospace Conference (AERO 2013)*, Montana, America, 2–9 March 2013, pp. 1–8. IEEE.
  30. Teng G, Zhou M, Li C, et al. Mini-uav lidar for power line inspection. *ISPRS Int Arch Photogramm Remote Sens Spatial Inform Sci* 2017; XLII-2/W7: 297–300.
  31. Cerón A, Mondragón I, and Prieto F. Research on power line inspection by visual based navigation. In: *Proceedings of SPIE – The International Society for Optical Engineering*, San Francisco, America, 8–12 February 2015.
  32. Strasdat H, Montiel J, and Davison AJ. Real-time monocular slam: why filter? In: *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, America, 3–8 May 2010, pp. 2657–2664. IEEE.
  33. Engel J, Schöps T, and Cremers D. LSD-SLAM: large-scale direct monocular slam. In: *European Conference on Computer Vision*, Zurich, Switzerland, 6–12 September 2014, pp. 834–849. Springer.
  34. Newcombe RA, Lovegrove SJ, and Davison AJ. Dtam: dense tracking and mapping in real-time. In: *2011 IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 6–13 November 2011, pp. 2320–2327. IEEE.
  35. Pizzoli M, Forster C, and Scaramuzza D. Remode: probabilistic, monocular dense reconstruction in real time. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 31 May–7 June 2014, pp. 2609–2616. IEEE.
  36. Davison AJ, Reid ID, Molton ND, et al. Monoslam: real-time single camera slam. *IEEE Trans Pattern Anal Mach Intell* 2007; 29(6): 1052–1067.
  37. Klein G and Murray D. Parallel tracking and mapping for small ar workspaces. In: *6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007. ISMAR 2007*, Nara, Japan, 13–16 November 2007, pp. 225–234. IEEE.
  38. Mur-Artal R, Montiel JMM, and Tardos JD. ORB-SLAM: a versatile and accurate monocular slam system. *IEEE Trans Robot* 2015; 31(5): 1147–1163.
  39. Mur-Artal R and Tardós JD. ORB-SLAM: an open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans Robot* 2017; 33(5): 1255–1262.
  40. Lu Y and Song D. Robust RGB-D odometry using point and line features. In: *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, 13–16 December 2015, pp. 3934–3942.
  41. Gomez-Ojeda R and Gonzalez-Jimenez J. Robust stereo visual odometry through a probabilistic combination of points and line segments. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp. 2521–2526. IEEE.
  42. Zhang G, Lee JH, Lim J, et al. Building a 3-D line-based map using stereo slam. *IEEE Trans Robot* 2015; 31(6): 1364–1377.
  43. Martinez C, Sampedro C, Chauhan A, et al. Towards autonomous detection and tracking of electric towers for aerial power line inspection. In: *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, Orlando, America, 27–30 May 2014, pp. 284–295. IEEE.
  44. Huang J, Rathod V, Sun C, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In: *IEEE CVPR*, Honolulu, Hawaii, 21–26 July 2017, pp. 3296–3297.
  45. Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, Montreal, Canada, 7–12 December 2015, pp. 91–99.
  46. Mur-Artal R and Tardós JD. Probabilistic semi-dense mapping from highly accurate feature-based monocular slam. *Robot Sci Syst* 2015.

47. Von Gioi RG, Jakubowicz J, Morel JM, et al. LSD: a fast line segment detector with a false detection control. *IEEE Trans Pattern Anal* 2010; 32(4): 722–732.
48. Bartoli A and Sturm P. Structure-from-motion using lines: representation, triangulation, and bundle adjustment. *Comput Vis Image Und* 2005; 100(3): 416–441.
49. He Y, Zhao J, Guo Y, et al. Pl-vio: tightly-coupled monocular visual-inertial odometry using point and line features. *Sensors* 2018; 18(4): 1159.
50. Rosten E and Drummond T. Machine learning for high-speed corner detection. In: *European conference on computer vision*, Graz, Austria, 7–13 May 2006, pp. 430–443. Springer.
51. Calonder M, Lepetit V, Strecha C, et al. Brief: binary robust independent elementary features. In: *European conference on computer vision*, Crete, Greece, 5–11 September 2010, pp. 778–792. Springer.
52. Gálvez-López D and Tardos JD. Bags of binary words for fast place recognition in image sequences. *IEEE Trans Robot* 2012; 28(5): 1188–1197.
53. Fischler MA and Bolles RC. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun ACM* 1981; 24(6): 381–395.
54. Strasdat H, Montiel J, and Davison AJ. Scale drift-aware large scale monocular slam. *Robot Sci Syst VI* 2010; 2.
55. Horn BK. Closed-form solution of absolute orientation using unit quaternions. *JOSA A* 1987; 4(4): 629–642.
56. Liu W, Anguelov D, Erhan D, et al. SSD: single shot multi-box detector. In: *European Conference on Computer Vision*, Amsterdam, Netherlands, 8–16 October 2016, pp. 21–37. Springer.
57. Redmon J and Farhadi A. Yolo9000: better, faster, stronger. In: *IEEE CVPR*, Honolulu, Hawaii, 21–26 July 2017, pp. 6517–6525.
58. Refaellizadeh P, Tang L, and Liu H. Cross-validation. In: *Encyclopedia of database systems*. Berlin: Springer, 2009, pp. 532–538.
59. Jia Y, Shelhamer E, Donahue J, et al. Caffe: convolutional architecture for fast feature embedding. In: *Proceedings of the 22nd ACM international conference on Multimedia*, Orlando, America, 3–7 November 2014, pp. 675–678. ACM.
60. Everingham M, Van Gool L, Williams CK, et al. The pascal visual object classes (VOC) challenge. *Int J Comput Vision* 2010; 88(2): 303–338.
61. Matas J, Galambos C, and Kittler J. Robust detection of lines using the progressive probabilistic hough transform. *Comput Vis Image Und* 2000; 78(1): 119–137.
62. Akinlar C and Topal C. Edlines: a real-time line segment detector with a false detection control. *Pattern Recogn Lett* 2011; 32(13): 1633–1642.
63. Lee JH, Lee S, Zhang G, et al. Outdoor place recognition in urban environments using straight lines. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 31 May–7 June 2014, pp. 5550–5557. IEEE.
64. Agarwal S and Mierle K. *Ceres solver: Tutorial & reference*. Google Incorporated, 2012, 2: 72.
65. Schauwecker K and Zell A. Robust and efficient volumetric occupancy mapping with an application to stereo vision. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 31 May–7 June 2014, pp. 6102–6107. IEEE.