Improved Situation Awareness for Autonomous Taxiing Through Self-Learning

Bowen Lu, Matthew Coombes, Baibing Li, *Member, IEEE*, and Wen-Hua Chen, *Senior Member, IEEE*

Abstract—As unmanned aerial vehicles (UAVs) become widely used in various civil applications, many civil aerodromes are being transformed into a hybrid environment for both manned and unmanned aircraft. In order to make these hybrid aerodromes operate safely and efficiently, the autonomous taxiing system of UAVs that adapts to the dynamic environment has now become increasingly important, particularly under poor visibility conditions. In this paper, we develop a probabilistic self-learning approach for the situation awareness of UAVs' autonomous taxiing. First, the probabilistic representation for a dynamic navigation map and camera images are developed at the pixel level to capture the taxiway markings and the other objects of interest (e.g., logistic vehicles and other aircraft). Then, we develop a self-learning approach so that the navigation map can be maintained online by continuously map-updating with the obtained camera observations via Bayesian learning. An indoor experiment was undertaken to evaluate the developed self-learning method for improved situation awareness. It shows that the developed approach is capable of improving the robustness of obstacle detection via updating the navigation map dynamically.

Index Terms—Autonomous taxiing, self-learning, situation awareness, unmanned aerial vehicle.

I. INTRODUCTION

AFTER several decades of research and development, unmanned aerial vehicles (UAVs) are now widely used in civil applications (e.g. monitoring gas pipelines [1] and surveillance of electrical power infrastructures [2]). Civil UAVs are unlikely to have their own specialized aerodromes for taking-off and landing due to the enormous initial investment and the following maintenance expenses. A feasible solution would be to share civil aerodromes with manned aircraft so that the existing facilities can be fully utilized during off-peak time. This paper considers an important safety issue, i.e. situation awareness of a UAV during its autonomous taxiing.

There is considerable attention paid to the research on improving the ground surface operation efficiency and safety in the recent years as aerodromes have become increasingly complex with heavier traffic flow. This includes investigations on aircraft departure (pushback) phase [3], surface traffic automation systems for taxiing phase [4], and optimization for taxiway routing and runway scheduling [5]. Apart from being more efficient, safety is another important research direction regarding aerodrome operation. For example, with a survey on surveillance technologies, [6] investigates the key separation parameters for a conflict detection and resolution system. Clearly a hybrid environment for both manned and unmanned aircraft has raised greater challenges to the operation and safety issues in the already very complicated aerodrome environment. Using vertical take-off and landing (VTOL) type of UAVs could be a solution for some applications for not introducing too much additional troubles to the aerodrome. However, they have limited payloads and flight endurance while comparing with fixed-wing aircraft. Therefore, this paper focuses on solving the problems that a fixed-wing UAV may cause or encounter.

A typical hybrid civil aerodrome for manned and unmanned aircraft is a highly dynamic environment, and autonomous taxiing is one of the major tasks of a UAV operating on the aerodrome surface. Hence an autonomous taxiing system is required to be adaptive and responsive to provide safety guarantee. A detailed discussion on safety issues can be found in [7]. In general, the autonomous taxiing system on a UAV is expected to have the same visual information acquiring capability as a human pilot, which includes obtaining information visually from signs, light signals, pavement markings, and obstacles. Therefore, vision-based approaches are preferred in applications (see, e.g. [8]).

While autonomous taxiing has been achieved previously on the Global Hawk aircraft using DGPS and highly accurate maps to guide the aircraft around a segregated air force controlled airport [9], this is totally inadequate for a civil aerodrome. Such a system would need constant supervision by the remote pilot as it will not react to obstacles, and will not function without DGPS corrections or in GPS denied environments. In contrast, by allowing a UAV to have a greater degree of autonomy in the taxiing phase of aerodrome operations, as well providing greater situation awareness, remote pilots can be tasked to handle multiple aircraft at once [10]. Such a system would significantly improve the utility of UAVs by reducing the workload of the remote unmanned aircraft pilot and by easing integration into the national airspace system (NAS) [11]. Because of the importance, autonomous taxiing and automated aerodrome operations are an identified research gap to bringing UAVs into the NAS, and to operate in non-segregated aerodromes [12].

B. Lu, M. Coombes, and W.-H. Chen are with the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough LE11 3TU, U.K. (e-mail: B.Lu@lboro.ac.uk; M.J.Coombes@lboro.ac.uk; W.Chen@lboro.ac.uk).

B. Li is with the School of Business and Economics, Loughborough University, Loughborough LE11 3TU, U.K. (e-mail: B.Li2@lboro.ac.uk).

So far there has been minimal research in the area of UAV autonomous taxiing. A closely related research area is lane detection for roadway traffic. Road surface is usually dark, whereas roadway markings are usually painted in white or yellow (light colors). Hence, in practice, light intensity and color-based approaches are commonly used for lane detection. For example, a color-based approach is used for tracking unmarked road lanes in [13] and [14] respectively. [15] investigates extracting lane marks by matching the dark-light-dark pattern from observations. In doing so, the RGB camera observations are converted into light intensity images and mapped into a ground plane for lane detection. Lane detection can also be carried out based on features other than color so errors associated with differing light levels are minimized. [16] presents a two-stage lane marking detection scheme, where the Hough transform is first of all used to detect the vanish point from an observed image and an inverse perspective mapping is applied according to the vanish point information; then a neural network is employed to recognize markers from the mapped image. [17] and [18] also use the Hough transform for lane detection.

Although great effort has been made to increase the reliability of lane detection, worn markings, low resolution and noisy input images set a ceiling to the performance of pure computer vision solutions. To further improve on the robustness of the vision approaches, domain knowledge needs to be incorporated. In the literature, maps are sometimes treated as prior knowledge and used as a reference for improving on the localization of a vehicle by comparing the prior information from the map with the current camera vision. Specifically, [19] aligns the camera vision from a vehicle with a sequence of coarse-to-fine birds-eye images to improve on the visual localization accuracy. A mutual information based matching between vehicle observations and a local orthographic imagery for localization is reported in [20]. [21] compares the real-time laser scan from vehicle with a pre-collected obstacle point cloud to achieve better localization. In the autonomous vehicle localization literature discussed, various sensors and observation-map matching strategies are used and promising results are obtained.

Recently, by combining the two lines of research on vehicle lane detection and vehicle localization, [22] has made the first attempt to use both the map information and camera images for a taxiway centerline extraction in an aerodrome scenario. The updating process in [22], however, is static and hence is unable to adapt to the dynamic environment of an aerodrome.

In this paper, we propose a self-learning probabilistic approach for maintaining and updating a dynamic navigation map for autonomous taxiing that includes an aerodrome map and an obstacle map. More specifically, the camera observations, aerodrome map and obstacle map are each represented as a probability distribution. With a given GPS measurement from the UAV, matching between the current camera observation and navigation map is carried out to increase the localization accuracy. Then Bayesian inference is drawn to combine the current camera observation with the map prior to calculate the taxiway's posterior distribution. The obtained posterior distribution of the dynamic navigation map at the current time step is further treated as prior knowledge at the next time step for

processing the next frame of image. In this way, the navigation map can be dynamically maintained and updated to adapt to the dynamic environment of an aerodrome.

This paper is organized as follows. Section II highlights the research challenges of autonomous taxiing and outlines a probabilistic learning framework to address these research challenges. In Section III we develop the probabilistic representations for camera observations, aerodrome map and obstacle map. Kullback-Leibler divergence based matching is discussed in Section IV for improved localization. In Section V we investigate the self-learning process of the dynamic navigation map. A practical indoor experiment is discussed in Section VI. Finally, concluding remarks are given in Section VII.

II. RESEARCH CHALLENGES AND FRAMEWORK

A. Research Challenges

Broadly speaking, there are two possible approaches to support autonomous taxiing for UAV operation in aerodromes. One is the combination of GPS measurements with a detailed aerodrome map. By using high accuracy GPS, an aircraft can be located on the aerodrome map so the local environment information can be extracted. This will make aerodrome navigation possible. However, although an aerodrome is usually well regulated, it is a dynamic environment with a high level of uncertainty. For example, a map might be out of date due to maintenance work or accidents, foreign objects may cause an obstruction, airport service vehicles could be driving around the aerodrome causing there to be an unmapped obstacle.

The other approach is, equivalent to a pilot, to observe real local environment information using the on-board cameras. Computer vision, however, is quite susceptible to errors caused by changing weather and light conditions. For example, the taxiway markings could be worn out and hence not clearly visible, false obstacle detections and other clutter might be generated from camera observations. As a safety critical system, neither approach would fulfill the integrity and safety requirements for an autonomous taxiing system of unmanned aircraft.

This paper aims to integrate these two approaches together to provide a much more reliable and safer method for the autonomous taxiing of UAVs. Technically, our hypothesis is that these two approaches are complementary with each other. The detailed aerodrome map provides the information such as what the camera system expects to see and where to focus the attention on. This helps to identify worn out markings, deal with poor visibility and reduce clutter. On the other hand, the camera system of a UAV is able to identify unexpected objects and their changes so that it is able to update the navigation map in real time and provide a better safety assessment. To realize this, there are a number of research challenges to address, including:

1) How to fuse these two different approaches together?
2) If there is any conflict between the information from these two sources, what is the strategy in resolving the conflict (e.g. which one shall be trusted more)?
3) How to make the overall system better than each individual part in terms of integrity, safety and accuracy?
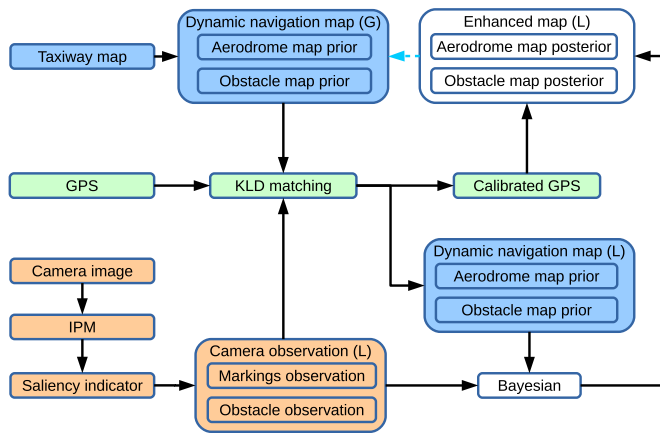
Fig. 1. Framework structure.

In this paper to address these issues, a probabilistic approach will be developed in a Bayesian learning framework. The proposed self-learning approach is able to take advantages of both approaches to provide robust and reliable situation awareness in dynamic and uncertain environments to support UAVs' autonomous taxiing. This is particularly important under poor visibility conditions such as night time, fog, haze, and heavy rain.

B. Research Framework Structure

In this section, we provide an overall framework structure for the proposed approach. Instead of considering a GPS-denied environment, we consider a more typical aerodrome environment where UAVs navigate with GPS measurements and an accurate aerodrome map. However, we don't assume the GPS measurements are highly reliable. Aerodromes are highly dynamic environments full of uncertainties such as the other aircraft, vehicles, etc.; it is practically impossible to include the information of these moving objects into a map in advance. Rather, the only way to address this issue is to employ an online detecting process with various on-board sensors, such as camera and LIDAR. In order to detect an object from noisy camera images as early and robustly as possible during the navigation, here we use a self-learning approach to maintain the detection results. To handle the dynamic information, we propose a concept of "dynamic navigation map" which is a knowledge base containing both the static and dynamic information, and we develop a self-learning approach for maintaining and updating it. The dynamic navigation map is a collection of two individual maps, i.e. an aerodrome map and an obstacle map. The aerodrome map provides the position information of all the taxiway/runway centerlines, stop bars, and other markings. This map can be treated as either static or dynamic depending on its accuracy. The main purpose of the aerodrome map is to provide a reference for visual localization, which will be further used as a starting point for updating either the aerodrome map itself or the obstacle map or both. The obstacle map, on the other hand, is a dynamic map for storing the previously learned obstacle distribution; it can be combined with the current observation to gain a more robust detection. The overall framework structure is given in Fig. 1.

As shown from the left column of Fig. 1, the framework includes two major input sources, i.e. the aerodrome map and camera images. In order to have a probabilistic representation, the prior distributions are initialized for the aerodrome map and obstacle map according to the given aerodrome map by setting an initial variance on each location. Due to the fact that each camera image is generated from a forward-facing camera and the aerodrome map is a two-dimensional top-view image, an inverse perspective mapping (IPM) pre-processing is required to map the camera image to be consistent with the top down view of the aerodrome map. A "camera observation" refers to this mapped image instead of the original untransformed camera image. Then an indicator is calculated to extract different objects (obstacles, the centerline and other traffic markings) from the mapped observation, and these objects are further represented as probability distributions by following the same process as the aerodrome and obstacle maps. The subsequent Bayesian analysis is undertaken that combines the prior distributions of aerodrome and obstacles, and the likelihood functions formed using the camera observation. It should be noted that the dynamic navigation map is a global map (labeled by "G") covering the whole aerodrome and the observation (labeled by "L") is a camera viewing area related local image.

The third input of the framework is GPS measurements. This information is required to link the camera observations and the global dynamic navigation map, indicating which area on the dynamic navigation map corresponds to the current camera observation. Due to the measurement noise of the GPS, a matching process is required to refine the measurement. This is undertaken by finding the optimal matching location by minimizing the Kullback-Leibler divergence (KLD) between the distributions of the navigation map and the camera observation. This matching process produces an optimal local navigation map together with a calibrated GPS measurement. With the introduced process on the three inputs, the local camera observation is aligned with the dynamic navigation map, and both of them are represented with an identical probabilistic representation format. Thus, the two sources are ready to be combined and research challenge 1 is addressed.

Once a matched local navigation map is found, the posterior distributions of the aerodrome and obstacles are produced by Bayesian inference, named as "enhanced map" in Fig. 1. This enhanced map is then updated back onto the global dynamic navigation map according to the calibrated GPS measurement, and will be used for matching the next frame of the camera observation in the next time step. By using a Bayesian framework, the "enhanced map" (posterior) is a weighted combination of the "dynamic navigation map (L)" (prior) and the "camera observation (L)" (likelihood). The weights are defined by the variances of the prior and likelihood distributions, which reflect the confidence in each of them. Therefore, Bayesian inference provides guidance for solving the information conflicts mentioned in challenge 2.

Regarding research challenge 3, the accuracy of the GPS measurements are refined in the proposed framework by comparing the camera observation with the aerodrome map. In addition, the robustness of obstacle detection is improved by combining previous observations into the learned obstacle map. Thus, the accuracy and robustness of the overall system become better.

III. PROBABILISTIC REPRESENTATIONS

In general, camera observations in poor visibility are quite noisy and object extraction is not very reliable. We therefore incorporate a probabilistic self-learning approach. To this end, we first of all discuss the probabilistic representations for the aerodrome map and camera observations.

A. Pre-Processing of Taxiway Map and Camera Images

In this paper, the initial navigation map $M = \{M_c, M_o\}$ is defined to be a collection of the aerodrome map M_c and the initial obstacle map M_o. The aerodrome map M_c is a binary image in which the pixels associated with the taxiway centerline and other traffic markings are "1"s and the rest pixels are "0"s. The initial obstacle map M_o is set as an image with all zeros, indicating that no obstacle is presumed on the map.

As explained earlier, camera observations are captured from a forward-facing camera mounted on the aircraft. The IPM is employed to produce a top view of the observation that is consistent to the aerodrome map. The perspective transform matrix that is required by the IPM can be computed from the camera's intrinsic and extrinsic parameters. The error of these parameters, however, accumulates. Therefore, instead we use a direct reconstruction with a set of measured control points to find the transformation matrix (see [23]).

The taxiway centerline is in a salient bright yellow color on the dark asphalt ground. We use the frequency-tuned salient region detection approach to characterize all the salient objects (both the centerline and obstacles) in the camera observations. Visual saliency is the perceptual quality that makes an object or person stand out relative to its neighbors and thus captures our attention. This new approach is suitable for the automatic detection of visually salient regions in images and has demonstrated both higher precision and better recall. In the recent years, it has attracted a lot of attention in the academic literature (see, e.g., [24], among many others).

The saliency indicator $\mathcal{S}$ is defined as

$$\mathcal{S}(X) = \|I_\mu - \boldsymbol{I}_G(X)\| \tag{1}$$

in which X denotes pixel locations of the image. $I_\mu = [L_\mu, a_\mu, b_\mu]$ is the mean image feature vector of the observation in $L*a*b*$ color space, and $\boldsymbol{I}_G = [\boldsymbol{L}_G, \boldsymbol{a}_G, \boldsymbol{b}_G]$ is the blurred observation image with a Gaussian filter. The blurry process removes fine texture details and high spatial-frequency noise. $\|\cdot\|$ is the L_2 norm (Euclidean distance). To ensure the navigation map has the same format, the Gaussian filter is also applied to M. The blurred navigation map is denoted by $\mathcal{M} = \{\mathcal{M}_c, \mathcal{M}_o\}$.

We note that the centerline and obstacles are mixed together in the saliency indicator $\mathcal{S} = \{\mathcal{S}_c, \mathcal{S}_o\}$. To distinguish them, the saliency indicator $\mathcal{S}$ is compared with $\mathcal{M}_c$ via the KLD matching, which will be discussed in Sections IV and V-B.

B. Representations of Navigation Map and Camera Observations

The modeling and analysis in this paper are carried out at the pixel level so that the objects of interest (e.g. obstacles) can
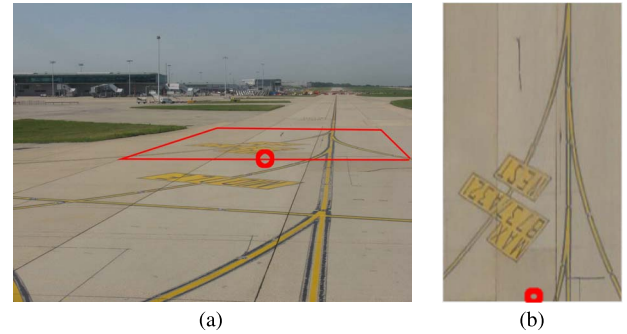

Fig. 2. A taxiway of Stansted Airport, U.K. (a) Front view of ROI. (b) Top view of ROI.

be detected as early and robustly as possible when the objects are very small; this can give a UAV sufficient time to react (collision avoidance, etc.) during autonomous taxiing.

Let $\mathbb{M} = \{\mathbb{M}_c, \mathbb{M}_o\}$ denote the parameter matrices corresponding to the ground truth of locations for the taxiway $\mathbb{M}_c$ and obstacles $\mathbb{M}_o$ respectively. Bayesian inference is used to infer this information from the navigation map (treated as prior knowledge) and camera observation (treated as likelihood function). It is assumed that both the navigation map and the observation at the pixel level follow the Gaussian distributions below:

$$\text{Navigation map}: q(\mathbb{M}) = \mathcal{N}\left(\mathbb{M}; \mathcal{M}, \sigma_{\text{Nav}}^2\right) \tag{2}$$

$$\text{Camera observation}: q(\mathcal{S}|\mathbb{M}) = \mathcal{N}\left(\mathcal{S}; \mathbb{M}, \sigma_{\text{Obs}}^2\right) \tag{3}$$

where σ_{Obs}^2 and σ_{Nav}^2 are the variances specifying the noise levels of the observation and navigation map. Note that as the navigation map contains both the aerodrome map and obstacle map, $q(\mathbb{M}) = \mathcal{N}(\mathbb{M}; \mathcal{M}, \sigma_{\text{Nav}}^2)$ in the above equation applies to both of the two maps, and hence it includes two separate distributions, i.e. the aerodrome map $q(\mathbb{M}_c) = \mathcal{N}(\mathbb{M}_c; \mathcal{M}_c, \sigma_{c,\text{Nav}}^2)$ and the obstacle map $q(\mathbb{M}_o) = \mathcal{N}(\mathbb{M}_o; \mathcal{M}_o, \sigma_{o,\text{Nav}}^2)$. Similarly, the camera observation distribution $q(\mathcal{S}|\mathbb{M})$ includes two components: one for the surface marking observation $q(\mathcal{S}_c|\mathbb{M}_c)$ and the other for obstacle observation $q(\mathcal{S}_o|\mathbb{M}_o)$. The relationship among them is illustrated in Fig. 1.

To demonstrate the representation process discussed in this section, a photograph[1] taken at Stansted Airport, UK, is used as a raw observation from a forward-facing camera as shown in Fig. 2(a), where a calibrated trapezoid area marks the region of interest (ROI). Fig. 2(b) is the mapped top view of the ROI after the IPM.

An aerodrome map $\mathcal{M}_c$ is generated from Google maps and blurred with a Gaussian filter, as shown in Fig. 3(a). The only information contained in this map is the layout of the taxiway centerline. Note that the aerodrome map is in the global (geographic) coordinate system, whereas the camera image is in the UAV's local coordinate system (heading to south-west in the global coordinate system).
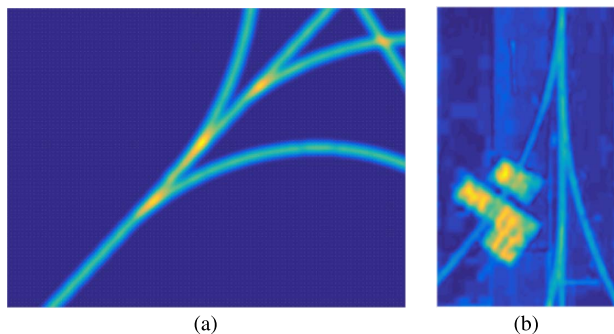
---

[1]http://www.geograph.org.uk/photo/2481393

Fig. 3. Filtered aerodrome map and saliency indicator. (a) Filtered aerodrome map. (b) Saliency indicator.

Eq. (1) is employed to further process the mapped observation to the saliency indicator $\mathcal{S}$, as shown in Fig. 3(b). Since no obstacle is present in this observation, we have $\mathcal{S}_c = \mathcal{S}$.

IV. MATCHING OF OBSERVATION WITH NAVIGATION MAP

Although a consistent representation format is constructed in Section III, a positional matching process is required to find the sub-area from the global navigation map $q(\mathbb{M})$ that matches to the local camera observation $q(\mathcal{S}|\mathbb{M})$. In addition, the matching process also allows us to separate obstacles from the centerline.

Specifically we find the optimal matching solution by minimizing the KLD between $q(\mathbb{M})$ and $q(\mathcal{S}|\mathbb{M})$. As a by-product, when the KLD is minimized, the KLD of each pixel can be used as a criterion for finding obstacles. In this section we consider the matching problem only; the obstacle problem will be dealt with in Section V-B.

A. Position Measurement Model

Let $\boldsymbol{p} = [x, y, \theta]^T$ denote the ROI position in an aerodrome coordinate system (two dimensional position with a yaw angle) which is the middle point on the nearest edge of the trapezoid ROI, as shown by the circle in Fig. 2(a). Although the position of the ROI cannot be measured directly, it can be obtained by simply applying a rigid transformation on the raw GPS measurement. To simplify notation, the transformation process is suppressed notationally and we consider the (transformed) GPS measurement on the ROI position in the rest of the paper. The GPS measurement model is given as

$$\hat{\boldsymbol{p}} = f(\boldsymbol{p}) + \boldsymbol{\varepsilon} \qquad (4)$$

where $f(\cdot)$ is a measurement function and $\boldsymbol{\varepsilon} = [\varepsilon_x, \varepsilon_y, \varepsilon_\theta]^T$ is a vector of additive Gaussian measurement noises.

B. Locational Matching With Symmetrized KLD

Given a measurement $\hat{\boldsymbol{p}}$, the objective of KLD matching is to estimate the ground truth $\boldsymbol{p}$ such that the corresponding sub-area on the global aerodrome map $q(\mathbb{M}_c)$ best matches to the

(a) Matched area in the global navigation map

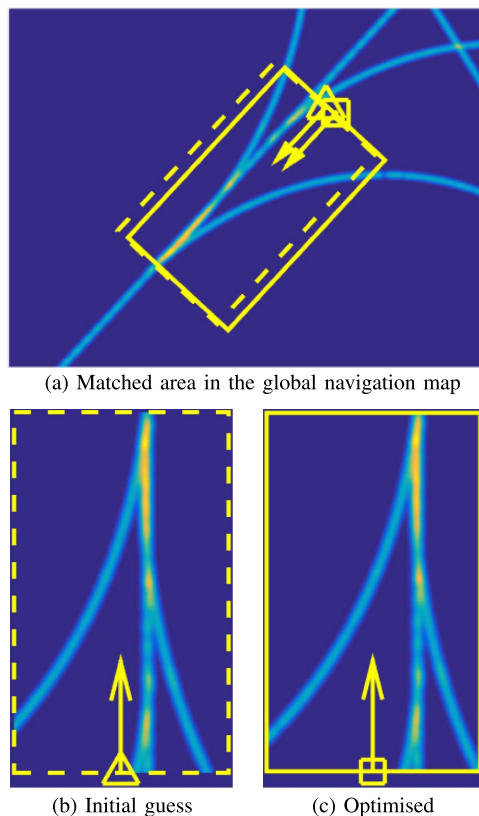(b) Initial guess   (c) Optimised

Fig. 4. Optimally matched area in the global navigation map. Comparing the cropped maps (b) and (c) with the camera observation in Fig. 3(b) [e.g., the bottom right corner of (b) and (c)], we can see that the centerline in (c) is better matched.

observed centerline $q(\mathcal{S}_c|\mathbb{M}_c)$. For this end, we minimize a symmetrized KLD between the two distributions

$$\boldsymbol{p}_{\text{opt}} = \arg\min_{\boldsymbol{p}_{\text{test}}} \sum_{X \in \mathbb{I}} [D_{\text{KL}}\left(q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X) \| q(\mathcal{S}_c|\mathbb{M}_c; X)\right)$$
$$+ D_{\text{KL}}\left(q(\mathcal{S}_c|\mathbb{M}_c; X) \| q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X)\right)] \quad (5)$$

where $\boldsymbol{p}_{\text{test}} = \hat{\boldsymbol{p}} + [dx, dy, d\theta]^T$ is a test location in the search space around the measurement $\hat{\boldsymbol{p}}$, and $\boldsymbol{p}_{\text{opt}}$ is the best guess of the ROI position $\boldsymbol{p}$. $q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X)$ denotes the distribution of the aerodrome map of each pixel X at the test location $\boldsymbol{p}_{\text{test}}$, and $q(\mathcal{S}_c|\mathbb{M}_c; X)$ is the observation distribution of each pixel X. $\mathbb{I}$ is a pixel collection of the local observation area. The operator $D_{\text{KL}}(P\|Q)$ computes the Kullback-Leibler divergence of Q from P.

To illustrate the matching process, we return to the example that we examined earlier. Using Fig. 3(b) as the local markings observation and Fig. 3(a) as the global aerodrome map, a matching result is shown in Fig. 4(a). Given a measured camera state marked with a triangle and an arrow, an initial guess of the area corresponding to the current observation is obtained, as shown to be the area with a dashed rectangle in Fig. 4(a). A KLD cost is then computed in this neighboring area using Eq. (5). Considering the high accuracy and availability of local area augmentation system (LAAS) in many aerodromes, and the increasing use of satellite based augmentation system (SBAS), a 6 m × 6 m neighboring search region would be

Fig. 5. KLD cost mesh.

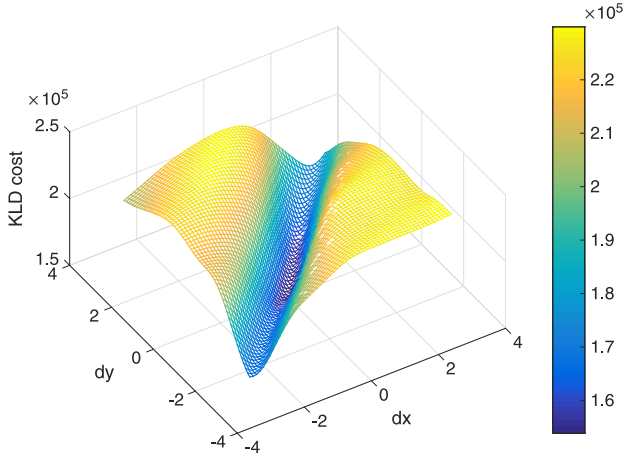Fig. 6. Observation frame sequence in 1 s. (a) time $= 0$ s. (b) time $= 0.2$ s. (c) time $= 0.4$ s. (d) time $= 0.6$ s. (e) time $= 0.8$ s. (f) time $= 1$ s.

sufficient for the matching. Fig. 5 displays the KLD cost in the neighboring area with $dx \in [-3, 3]$ meters and $dy \in [-3, 3]$ meters ($d\theta$ is optimized during the matching, but not shown in this figure), and $\boldsymbol{p}_{\text{opt}}$ is obtained with the minimal KLD cost. Corresponding to $\boldsymbol{p}_{\text{opt}}$, a solid rectangle as the optimally matched area is worked out. In addition, Fig. 4(b) and (c) show the extracted sub-areas from the global navigation map with the raw GPS measurement $\hat{\boldsymbol{p}}$ and with the optimal estimate $\boldsymbol{p}_{\text{opt}}$ respectively.

We note that an integral operation is involved in computing the KLD where the evaluation at each pixel is very computationally demanding. However, since both Eqs. (2) and (3) are Gaussian, its analytical solution can be obtained (see, e.g. [25])

$$D_{\text{KL}} \left(q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X) \| q(\mathcal{S}_c | \mathbb{M}_c; X) \right)$$
$$= \log \frac{\sigma_{\text{Obs}}}{\sigma_{\text{Nav}}} + \frac{\sigma_{\text{Nav}}^2 + (\mathcal{M}_{c,X} - \mathcal{S}_{c,X})^2}{2\sigma_{\text{Obs}}^2} - \frac{1}{2} \quad (6)$$

$$D_{\text{KL}} \left(q(\mathcal{S}_c | \mathbb{M}_c; X) \| q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X) \right)$$
$$= \log \frac{\sigma_{\text{Nav}}}{\sigma_{\text{Obs}}} + \frac{\sigma_{\text{Obs}}^2 + (\mathcal{M}_{c,X} - \mathcal{S}_{c,X})^2}{2\sigma_{\text{Nav}}^2} - \frac{1}{2}. \quad (7)$$

Consequently, Eq. (5) reduces to

$$\boldsymbol{p}_{\text{opt}} = \underset{\boldsymbol{p}_{\text{test}}}{\arg \min}$$
$$\sum_{X \in \mathbb{I}} \left[\frac{\sigma_{\text{Nav}}^4 + \sigma_{\text{Obs}}^4 + (\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2)(\mathcal{M}_{c,X} - \mathcal{S}_{c,X})^2}{2\sigma_{\text{Nav}}^2 \sigma_{\text{Obs}}^2} - 1 \right]$$
$$(8)$$

where $\mathcal{M}_{c,X}$ and $\mathcal{S}_{c,X}$ are the values associated with $q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}}, X)$ and $q(\mathcal{S}_c | \mathbb{M}_c; X)$ respectively at each pixel X.

Remarks:

(i) The element-wise matrix operation in Eq. (8) may incur a heavy computational cost. In order to increase the computation speed and meet the real-time process requirement, we note that only a sub-area $q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}})$ extraction from $q(\mathbb{M}_c)$ is needed in each of the optimization iterations. Hence, instead of extracting from the full aerodrome map $q(\mathbb{M}_c)$, a pre-cropped map can be used to speed up this operation. More specifically, the sub-area to be selected is a circumscribed rectangle of the estimated ROI region (defined by the estimated ROI
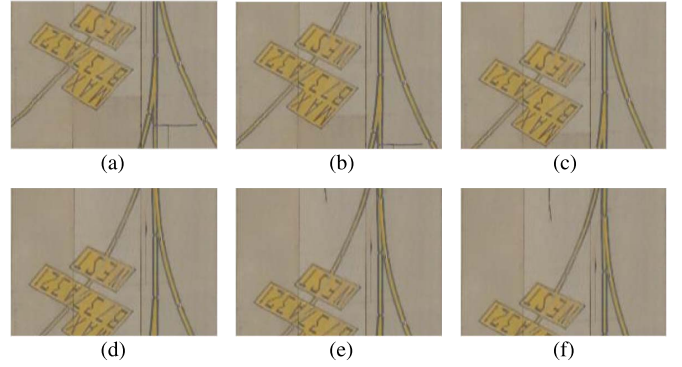
position $\boldsymbol{p}_{\text{opt}}$), and it's edges are parallel to the two axes of the aerodrome coordinate system.

(ii) Temporally reducing the resolution of $q(\mathbb{M}_c; \boldsymbol{p}_{\text{test}})$ and $q(\mathcal{S}_c | \mathbb{M}_c)$ can effectively control the number of pixels in $\mathbb{I}$. In doing so, the computational cost for calculating the KLD will be substantially reduced in each iteration of the optimization.

C. Calibrated GPS Measurements

As mentioned earlier, the GPS measurement $\hat{\boldsymbol{p}}$ in Eq. (4) is subject to some noise. The KLD matching provides the optimal estimate $\boldsymbol{p}_{\text{opt}}$ of the ROI position as it is worked out using two sources of information, i.e. the raw GPS measurements and camera observation. This estimated ROI position is then used to guide the self-learning process by indicating which area of the navigation map matches to the current camera observation and should be updated. As a by-product, the accuracy of GPS measurements is also improved.

To demonstrate the improved GPS accuracy, we consider a UAV moving forwards at a constant speed of 10 m/s with a given initial ROI position. An observation sequence is generated from Fig. 2(b) and used for the demonstration purposes. By setting the temporal interval between observations as 0.1 s, half of the sequence is given in Fig. 6 with a time interval of 0.2 s. Measurements with respect to each of the observations are simulated by adding a zero mean Gaussian noise to the given camera trajectory. Fig. 7 displays a comparison between the calibrated measurement $\boldsymbol{p}_{\text{opt}}$ with raw measurement $\hat{\boldsymbol{p}}$. Universal Transverse Mercator (UTM) coordinate system is used in this figure, which converts longitude and latitude into a two dimensional Cartesian coordinate system in meters. An offset vector $[x, y] = [311100, 5752700]$ is subtracted from the axes. The straight line with asterisks is the ground truth of the ROI positions, and a circle on it marks its initial position at $t = 0$ s. Distances between two neighboring asterisks are 1 m apart. Triangles are the simulated measurements, and boxes are the corrected measurements with the KLD matching. From this figure, it can be clearly seen that the calibrated measurements are much closer to the ground truth than the raw GPS measurements.

V. Enhanced Navigation Map via Self-Learning

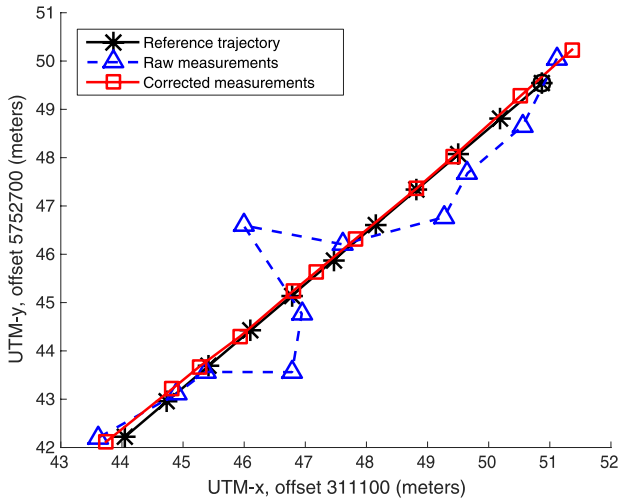This section investigates a self-learning process by pooling various sources of information together.

Fig. 7. Calibrated GPS measurements via the KLD matching.

Fig. 8. Illustration of a classification problem with and without the Bayesian learning process.

In the literature there are many different methods for fusing different sources of information. In this paper we use a Bayesian approach because: (a) we incorporate a probabilistic representation for each information source (the aerodrome map, camera images, etc.); (b) recursive Bayes' inference provides an error tolerance mechanism for dealing with various measurement noise; and (c) according to statistical theory, the posterior estimate is optimal with the minimum Bayes' risk. See [26] for an overview of machine learning with a probabilistic perspective.

A. Bayesian Updating

Given a probabilistic representation of a camera observation $q(\mathcal{S}|\mathbb{M})$ and the probabilistic representation of its corresponding local navigation map $q(\mathbb{M}; \boldsymbol{p}_{\text{opt}})$, the enhanced map is defined to be the posterior distribution obtained via Bayes' rule

$$\begin{aligned} q(\mathbb{M}|\mathcal{S}) &\propto q(\mathcal{S}|\mathbb{M})q(\mathbb{M}; \boldsymbol{p}_{\text{opt}}) \\ &\propto \mathcal{N}\left(\mathbb{M}; \mathcal{M}_{\text{post}}, \sigma_{\text{post}}^2\right) \end{aligned} \quad (9)$$

where

$$\mathcal{M}_{\text{post}} = \frac{\sigma_{\text{Obs}}^2}{\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2}\mathcal{M} + \frac{\sigma_{\text{Nav}}^2}{\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2}\mathcal{S} \quad (10)$$

$$\sigma_{\text{post}}^2 = \frac{\sigma_{\text{Nav}}^2 \sigma_{\text{Obs}}^2}{\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2}. \quad (11)$$

The above Bayesian inference applies to the aerodrome map and obstacle map respectively. Clearly the enhanced map $q(\mathbb{M}|\mathcal{S})$ at each time step keeps the navigation map $q(\mathbb{M})$ up to date. From Eq. (10), it can be observed that the mean of the enhanced map is the sum of the means of the current camera observation and navigation map, weighted by their precision (the reciprocals of their variances). The ratio between σ_{Obs}^2 and σ_{Nav}^2 determines which is the more dominant component in the learning process. For example, if camera observations have a higher level of uncertainty comparing with the aerodrome map (i.e. the prior knowledge is considered to be more reliable than the current observation for this type of application), a higher weight is allocated to the aerodrome map.
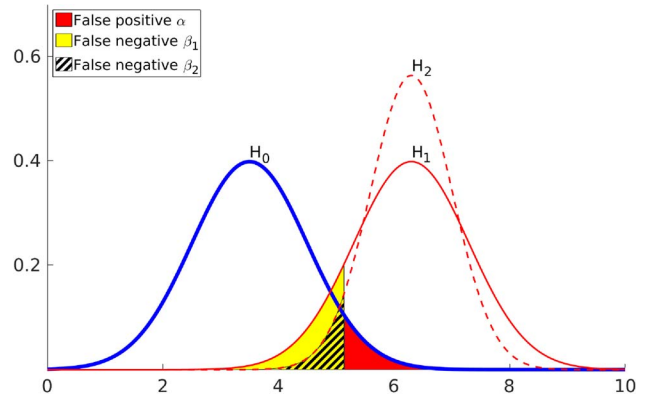
Eq. (11) gives the variance of the enhanced map. Note that $0 < \sigma_{\text{Nav}}^2/(\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2) < 1$ and $0 < \sigma_{\text{Obs}}^2/(\sigma_{\text{Nav}}^2 + \sigma_{\text{Obs}}^2) < 1$. Hence, we obtain from Eq. (11) that

$$\sigma_{\text{post}}^2 < \sigma_{\text{Nav}}^2 \quad \text{and} \quad \sigma_{\text{post}}^2 < \sigma_{\text{Obs}}^2$$

i.e. σ_{post}^2 is smaller than both σ_{Obs}^2 and σ_{Nav}^2. Consequently the uncertainty of the navigation map is reduced via Bayesian inference.

The enhanced navigation map with reduced posterior variance σ_{post}^2 increases the reliability of obstacle detection. For the same threshold used for obstacle detection, it can reduce false positive and false negative rates, and hence increase the accuracy of the detection.

To illustrate how the self-learning process improves obstacle detection, we consider a simplified classification problem with distributions $H_0 \sim \mathcal{N}(3.5, 1)$ and $H_1 \sim \mathcal{N}(6.3, 1)$, where H_0 and H_1 represent the probability distributions that a pixel of the camera observation comes from free space and obstacles respectively (see Fig. 8). With a given false positive rate $\alpha = 0.050$, the corresponding threshold is obtained as 5.145, thus leading to a false negative rate of $\beta_1 = 0.124$. Now, suppose that a posterior distribution is obtained, $H_2 \sim \mathcal{N}(6.3, 0.5)$, that updates the distribution H_1 with the same mean but a smaller variance. With the same false positive rate α, the false negative rate of H_2 is now reduced to $\beta_2 = 0.051$. Therefore the false negative is dramatically reduced after the self-learning.

Now to demonstrate the effect of the proposed self-learning process, we return to the example that we considered earlier and investigate the enhanced navigation map with the proposed self-learning method. Given a centerline indicator $q(\mathcal{S}_c|\mathbb{M}_c)$ in Fig. 3(b) and its optimally matched local centerline map $q(\mathbb{M}_c; \boldsymbol{p}_{\text{opt}})$ in Fig. 4(c), the enhanced local map is obtained in Fig. 9(a) using Eqs. (10) and (11). An updated centerline map $q(\mathbb{M}_c|\mathcal{S})$ is given in Fig. 9(b) by updating the corresponding area defined by $\boldsymbol{p}_{\text{opt}}$. It can be seen from the enhanced map that richer features from the camera observation are now included in the aerodrome map, such as the yellow label on the centerline.

B. Self-Learning Process

The aim of this paper is to develop a self-learning approach using the information collected from the camera of a UAV
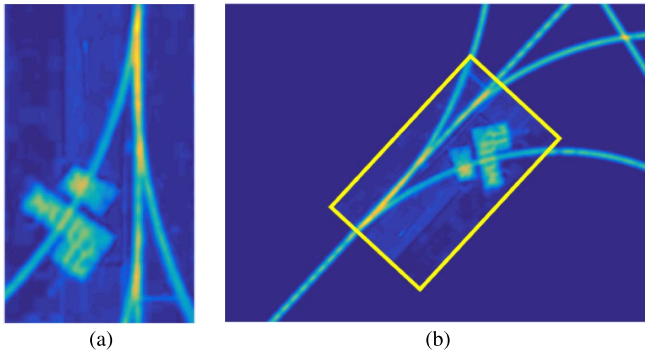
Fig. 9. Enhanced map. (a) Enhanced local map. (b) Updated aerodrome map.

to update the navigation map. In this subsection, we focus on two important issues of the self-learning process: (a) how knowledge is accumulated over time; and (b) how to deal with a dynamic environment with obstacles.

First, we consider a dynamic environment where the Bayesian updating process evolves from a time step t to the next time step $t + 1$.

In order to fully utilize the information collected in the previous time steps, the autonomous taxiing system needs to have a self-learning capability so that knowledge is accumulated over time. For this purpose, we treat the enhanced navigation map obtained at time t [i.e. the posterior distribution given in Eqs. (10) and (11)] as the prior knowledge in time step $t + 1$. The algorithm for the proposed navigation-map updating process is described as Algorithm 1.

Algorithm 1 The self-learning process

Let $t = 0$.
REPEAT

1) Process the camera image taken at time $t + 1$ and transform it into its probabilistic representation Eq. (3);
2) Obtain the GPS measurement Eq. (4) at $t + 1$;
3) Calculate the optimized measurement $\boldsymbol{p}_{\text{opt}}$ using the KLD matching method, Eq. (8);
4) Crop the posterior distribution obtained in time step t, i.e. $q(\mathbb{M}; \boldsymbol{p}_{\text{opt}})$, which is treated as the prior for time $t + 1$;
5) Draw Bayesian inference, Eq. (10);
6) Let $t = t + 1$;

UNTIL the aircraft has completed its taxi.

The full learning process is illustrated by Fig. 1. It is clear that as time increases during the taxiing, more and more camera observations will be incorporated into the dynamic navigation map. Hence, the weight of the navigation map gradually becomes dominant in the learning process. Intuitively, this means that if an object repeatedly appears in the observations, it will be confirmed and learned with an increasing level of confidence. In contrast, the information on confidence levels is rarely available in other detection approaches that fully rely on computer vision in the literature.

Next, we turn to investigate obstacles. The obstacle distribution is one of the most important pieces of the dynamic information. The saliency indicator $\mathcal{S}$ introduced in Eq. (1) contains both the centerline and obstacle information, whereas their counterparts in the navigation map are stored separately.

In the literature, there are many sophisticated methods for separating an object from its background. For example, [27] investigates a method that finds the obstacle-free-space in inverse perspective mapped images captured from a pair of front facing stereo cameras. On the other hand, [28] fuses motion information (optical flow) with stereo vision. In addition, to detect specific types of obstacles, [29] combines the active contour model with stereo vision and a dimension-ratio based object classifier is used to distinguish pedestrians, vehicles and other objects.

As the research focus of this paper is on improved situation awareness for UAVs' autonomous taxiing, we will not explore any technical details on obstacle filtering. Rather, we use a simple and straightforward approach based on the residual KLD cost map between the saliency indicator $\mathcal{S}$ and the taxiway centerline map $\mathcal{M}_c$ after the KLD matching. We note that each pixel value in the residual KLD cost map indicates how much the saliency map $\mathcal{S}$ is different from the centerline map $\mathcal{M}_c$ at each pixel. Assuming the centerline map is more accurate, a higher residual means the higher possibility of being an obstacle at the pixel. Therefore, we set a threshold of this residual map to generate two complementary indicators: a centerline indicator $\mathcal{S}_c$ with the obstacles being masked, and an obstacle indicator $\mathcal{S}_o$ with the centerline and the other traffic markings being masked. Each of them is then used as a likelihood function to compute the posteriors of the aerodrome map and obstacle map accordingly.

The variances of the aerodrome and obstacle maps, however, need to be updated differently. This is because an obstacle is assumed to be dynamic and its position may vary from time to time, whereas the centerline and the other traffic markings are assumed to be static. Hence the obstacle posterior variance obtained in Eq. (11) needs to be inflated before being employed as the prior distribution at the next time step.

Specifically, we follow [30] and suppose that the posterior variance for the obstacle map at time step t is σ^2_{post} (the subscript t is suppressed here for notation simplicity). In time step $t + 1$, the posterior distribution at time t is treated as the prior distribution at $t + 1$, i.e.

$$q(\mathbb{M}_o) = \mathcal{N}\left(\mathbb{M}_o; \mathcal{M}_{\text{post}}, \sigma^2_{o,\text{Nav}}\right)$$

where the prior variance $\sigma^2_{o,\text{Nav}}$ at $t + 1$ is an inflated posterior variance at t, i.e.,

$$\sigma^2_{o,\text{Nav}} = \frac{\sigma^2_{\text{post}}}{\lambda}$$

where $0 < \lambda < 1$ is usually termed a forgetting factor. λ can be used as a tuning parameter in practice. This variance-inflating method is commonly used in the literature (see, e.g. [30]). The above equation simply says that the obstacles are potentially moving objects and their positions may vary from time to time; when it evolves from t to $t + 1$, we become less confident in the posterior obstacle map obtained at time t.
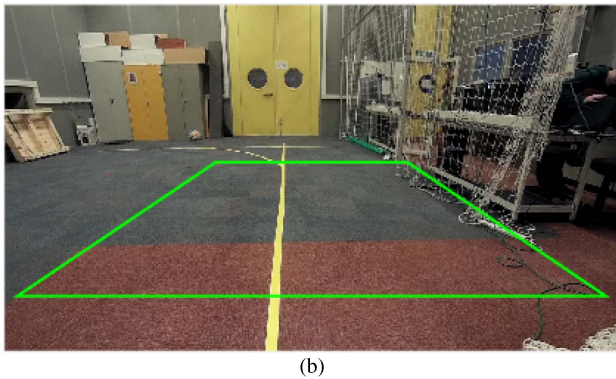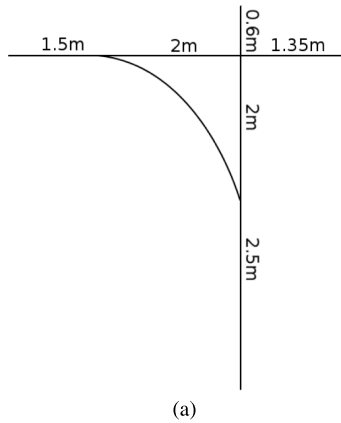
(a)

(b)

Fig. 10. Experiment environment. (a) Centerline layout. (b) Camera observation.

Fig. 11. Camera observation at 50% brightness.

(a)   (b)

Fig. 12. Camera observation and mismatching indicator at the 17th frame. (a) Camera observation. (b) Mismatching indicator.

VI. A PRACTICAL EXPERIMENT

To illustrate the proposed framework, a practical experiment was carried out in an indoor environment.

A. Settings for the Experiment

The purpose of this experiment is to demonstrate the ability of continuously self-learning when the proposed framework processes the collected information and deals with obstacles.

The environment was designed to be a taxiway junction with the centerline being marked with yellow tape in the experiment. The centerline layout with scale information is displayed in Fig. 10(a). A taxiway centerline map was generated based on this layout.

A differential drive unmanned ground vehicle (UGV) from Clearpath Robotics called the Husky was used as the vehicle platform. Highly accurate position measurements at each time step were obtained from a simultaneous location and mapping (SLAM) software, which utilizes the onboard LIDAR. A forward-facing GoPro camera was mounted on the top of Husky to produce camera images. Fig. 10(b) gives an experiment environment observation taken from the camera, in which the trapezoid marks the ROI that IPM was applied to.

B. Self-Learning for Obstacle Detection

The example given in previous sections updates the aerodrome map without considering obstacles. In this practical experiment, we assume the aerodrome map was accurate and focus on testing the self-learning method for obstacle detection. We compare obstacle detection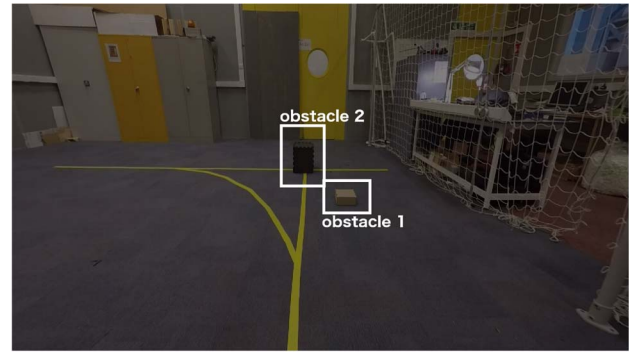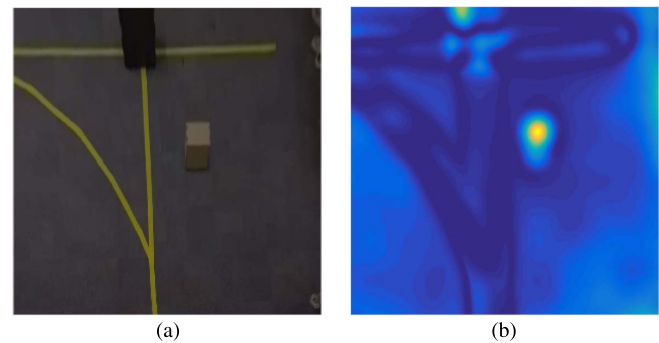 by using two different approaches: with and without the self-learning. To have a fair comparison, the whole experiment parameters were set the same for both approaches.

To demonstrate the performance in as many aspects as possible, we deliberately reduced the brightness of the observation to 50%. Two obstacles were placed in the environment. A small bright object (obstacle 1) was placed beside the centerline, and a big dark object (obstacle 2) was placed on the centerline crossing point, as shown in Fig. 11. During the experiment, the vehicle followed the centerline to the left and video was recorded at 30 frames per second. We evenly extracted 10% of the video frames for the experiment, i.e. 3 frames per second. The total number of extracted frames for the experiment was 40.

The centerline map was initialized with the layout given in Fig. 10(a), whereas the obstacle map was initialized with an empty map (all-zero matrix). These two maps described initial knowledge of the environment and were to be dynamically updated using the camera observations during the taxiing.

In the experiment, whenever a new camera observation was obtained as shown in Fig. 12(a), the latest GPS measurement was simulated by the SLAM software and retrieved as an initial point to minimize the KLD between the observation and the aerodrome map. With respect to obstacle detection and the navigation map updating, we considered two approaches: the approach without the self-learning where the obstacle detection was based on the saliency threshold only, and the approach with the self-learning enhanced obstacle detection where additionally the probability distribution incorporated the prior knowledge. In the experiment, the threshold for each of the two approaches was chosen independently to optimize their individual performances. In practice, they need to be carefully calibrated.
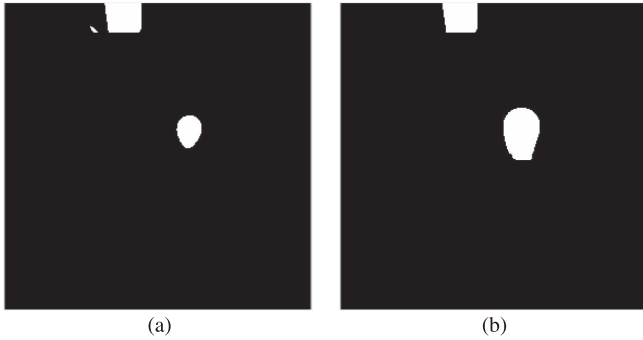
Fig. 13. Detection results at the 17th frame. (a) Without self-learning. (b) With self-learning.

Fig. 14. Performance comparison. (a) Recall. (b) F_1-score.

Here we take the 17th frame as an example: for the optimal matching point found, the mismatching level between the two is shown as Fig. 12(b). By setting a suitable threshold for each of the two approaches, the detection results are shown in Fig. 13, where the white areas denote the detected obstacle obscured regions in the camera view. Comparing the two results in Fig. 13 with the camera observation in Fig. 12(a), it can be seen that: (a) when an obstacle (e.g. obstacle 2) was relatively large, both approaches worked well and they were able to detect the obstacle easily. However, when an obstacle was relatively small (e.g. obstacle 1), the obstacle was detected more accurate with the self-learning enhanced approach than with the pure saliency based one; (b) due to the light condition, the side faced to the vehicle was darker than the top side of object 1. This side was almost not detected by the pure salience approach and consequently only the top side is shown in Fig. 13(a). In contrast, both the top side and the darker side of object 1 faced to the camera were detected in Fig. 13(b). This demonstrates that the self-learning enhanced approach is more reliable and is not very sensitive to the light condition; (c) the detection result without the self-learning shows an isolated false positive area [the small blob left to the detected obstacle 2 in Fig. 13(a)]. The self-learning approach, on the other hand, did not have such a problem because its detection was based on cumulative evidence rather than a single observation which was subject to the quality of that particular image.

To quantitatively evaluate the performance of the proposed self-learning approach, we consider two commonly used criteria: recall ratio and F_β-score. Recall (also termed sensitivity or true positive rate) measures the proportion of positives that are correctly identified as such. F_β-score is a widely used indicator in object detection (see, e.g., [31], [32]) to evaluate the overall performance of both recall and precision which is defined as

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

where β is the parameter for weighting the importance between precision and recall. Here we use F_1-score by setting $\beta = 1$, i.e. the same weight is applied to precision and recall. Clearly, F_1-score is the harmonic average of precision and recall.

The recall ratios of the two approaches (with and without the self-learning) for detecting the two obstacles are displayed in Fig. 14(a). The first and second obstacles moved into the
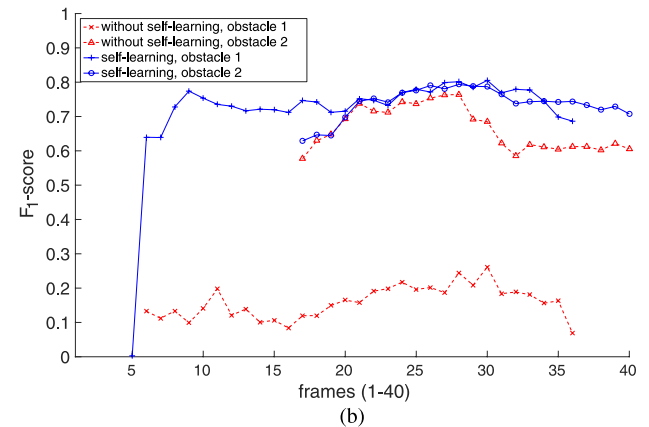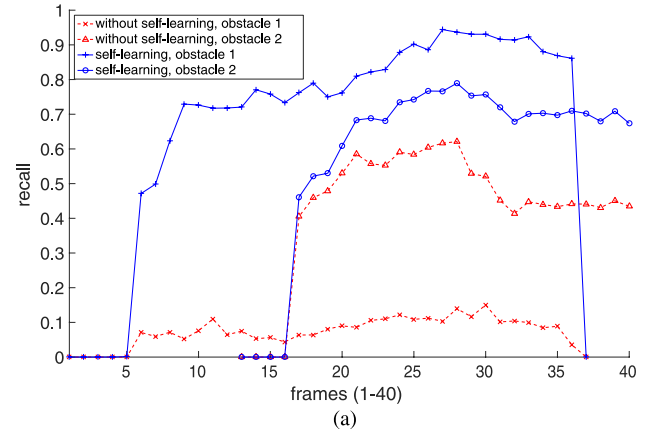
camera view region at the 1st and 13th frames respectively. It can be seen that the recall ratio of the self-learning enhanced approach increased much faster as the obstacle moved into the camera view. This is due to the fact that the posterior variance is dramatically reduced at each pixel when more evidence on the obstacles is accumulated via the Bayesian self-learning. As the consequence, the corresponding posterior probability increases rapidly and exceeds the threshold. After the vehicle moved forward and turned left, the first obstacle disappeared from the camera view in the 37th frame. Thus the recall of obstacle 1 drops to zero in Fig. 14(a).

In contrast, the performance of recall ratio for the approach without the self-learning varies: the recall ratio is high (low) when detecting a large (small) obstacle.

Next, we turn to consider the F_1-score displayed in Fig. 14(b). From the figure we can see that without the self-learning, the detection performance on the two obstacles are very different (the two dashed curves). Due to the bigger size of obstacle 2, the saliency detection without the self-learning gives a comparable F_1-score as the one with the self-learning. However, for a smaller object (obstacle 1), the F_1-score is significantly reduced without the self-learning, whereas the self-learning enhanced approach provides a better and consistent performance for both the smaller and larger obstacles.

In addition, without the use of the self-learning approach, in order to achieve a consistent detection performance, the algorithm requires a careful selection of different thresholds for each of the target obstacles according to their size, color, etc.

Since obstacles are unpredictable, it is not feasible to set multiple thresholds in real applications. In contrast, when the self-learning approach is used, the detection performance becomes more robust and consistent: the smaller (1st) and larger (2nd) obstacles can both be detected with similar performances when a single threshold was set. Such robustness demonstrated by the proposed approach is very important in practical applications.

Before concluding this section, we offer a couple of final comments. As a three-dimensional obstacle is projected onto a two-dimensional map, the shape of the obstacle after the projection varies from different observing angles. Therefore, the detected obstacle on the obstacle map does not reflect its real size from the top view. However, with mounting the camera at an appropriate height, the ground touching point of the obstacles can always be observed. Since this point is located on the ground, the localization of the obstacle based on this point will be accurate. Additionally, while the obstacle map is updated with a forward-facing camera observation, any area outside the current camera view region is uncertain. Thus, this global obstacle map should only be used as a temporary database for providing a continuous enhancement to the current camera-view region, but not for any global path planning. Due to the dynamic nature of an aerodrome environment, no obstacle is assumed to be permanent, and hence the navigation map should be reinitialized before the next taxiing task.

VII. Conclusion

This paper proposes a knowledge enhancing framework via a self-learning approach so that a vision input is reinforced by prior knowledge, and dynamic environment obstacles can be updated into the knowledge base during the taxiing process of a UAV.

To demonstrate the overall capability of the framework, a practical experiment was conducted in an indoor environment which shows that the performance of the developed approach meets our expectation.

The current framework improves the robustness of the saliency map based obstacle detection. The next step will be to extend the framework by incorporating other color and texture features to help UAVs gain a better understanding of the environment. In addition, taxiways usually have some informative marking/words. As markings/words recognition is a relatively mature research area, in our future work we will incorporate this additional information into the framework of this paper. Although in this paper we consider a dynamic environment of aerodromes where people and vehicles come and go, no dynamic equations are introduced for these moving obstacles; this will be another important future research issue.

Theoretically, once a proper probabilistic representation is worked out for these features, the methodology that we have developed in this paper can be used to combine all sources of information together to achieve an integrated situation awareness output. By taking the results from this paper and by utilizing knowledge pertaining to the highly controlled environment of an aerodrome, related decision making will be explored in our future research. Also, the framework will be tested with the data from a real taxiway and runway.

References

[1] D. Hausamann, W. Zirnig, G. Schreier, and P. Strobl, "Monitoring of gas pipelines a civil UAV application," *Aircr. Eng. Aerosp. Technol.*, vol. 77, no. 5, pp. 352–360, Oct. 2005.

[2] Z. Li, Y. Liu, R. Hayward, J. Zhang, and J. Cai, "Knowledge-based power line detection for UAV surveillance and inspection systems," in *Proc. IEEE 23rd Int. Conf. Image Vis. Comput. New Zealand*, Nov. 2008, pp. 1–6.

[3] P. Burgain, O. J. Pinon, E. Feron, J. P. Clarke, and D. N. Mavris, "Optimizing pushback decisions to valuate airport surface surveillance information," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 180–192, Mar. 2012.

[4] V. H. L. Cheng, V. Sharma, and D. C. Foyle, "A study of aircraft taxi performance for enhancing airport surface traffic control," *IEEE Trans. Intell. Transp. Syst.*, vol. 2, no. 2, pp. 39–53, Jun. 2001.

[5] G. Clare and A. G. Richards, "Optimization of taxiway routing and runway scheduling," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1000–1013, Dec. 2011.

[6] C. Mario and J. Rife, "Integrity and continuity for automated surface conflict-detection monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1179–1187, Sep. 2012.

[7] R. Loh and T. Roe, "UAVs in civil airspace: Safety requirements," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 24, no. 1, pp. 5–17, Jan. 2009.

[8] W. Eaton and W.-H. Chen, "Image segmentation for automated taxiing of unmanned aircraft," in *Proc. IEEE ICUAS*, Jun. 2015, pp. 1–8.

[9] G. Loegering and S. Harris, "Landing dispersion results—Global Hawk auto-land system," presented at the AIAAs 1st Technical Conference Workshop Unmanned Aerial Vehicles, Reston, VA, USA, 2002.

[10] "Unmanned aircraft systems roadmap 2005-2030," US Government, United States Dept. Def., Washington, DC, USA, 2005.

[11] E. W. Frew *et al.*, "Vision-based navigation for airfield surface operation," in *Proc. AIAA Guid., Navigat., Control Conf.*, 2008, pp. 4189–4194.

[12] "Directorate-general for enterprise and industry and European RPAS steering group and others," Roadmap for the Integration of Civil Remotely-Piloted Aircraft Systems into the European Aviation System, Eur. Commiss., OIB, Brussels, Belgium, 2013.

[13] M. A. Sotelo, F. J. Rodriguez, L. Magdalena, L. M. Bergasa, and L. Boquete, "A color vision-based lane tracking system for autonomous driving on unmarked roads," *Auton. Robots*, vol. 16, no. 1, pp. 95–116, 2004.

[14] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, and K.-C. Fan, "Lane detection with moving vehicles in the traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 571–582, Dec. 2006.

[15] R. K. Satzoda and M. M. Trivedi, "Drive analysis using vehicle dynamics and vision-based lane semantics," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 9–18, Feb. 2015.

[16] A. Kheyrollahi and T. P. Breckon, "Automatic real-time road marking recognition using a feature driven approach," *Mach. Vis. Appl.*, vol. 23, no. 1, pp. 123–133, Aug. 2010.

[17] V. Gaikwad and S. Lokhande, "Lane departure identification for advanced driver assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 910–918, Apr. 2015.

[18] Y. Zhang, J. Gong, and J. Tian, "Robust lane detection and tracking using improved Hough transform and Gaussian mixture model," in *Proc. 7th Int. Symp. MIPPR Int. Soc. Opt. Photon.*, T. Zhang and N. Sang, Eds. Nov. 2011, vol. 8003, pp. 1–8.

[19] A. Napier, G. Sibley, and P. Newman, "Real-time bounded-error pose estimation for road vehicles using vision," in *Proc. 13th Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2010, pp. 1141–1146.

[20] A. Napier and P. Newman, "Generation and exploitation of local orthographic imagery for road vehicle localisation," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2012, pp. 590–596.

[21] I. Baldwin and P. Newman, "Road vehicle localization with 2D pushbroom LIDAR and 3D priors," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 2611–2617.

[22] B. Lu, B. Li, and W. Chen, "Map enhanced visual taxiway extraction for autonomous taxiing of UAVs," in *Proc. IFAC Workshop Adv. Control Navigat. Auton. Aerosp. Veh.*, Seville, Spain, 2015, pp. 49–54.

[23] H. Opower, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[24] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, 2009, pp. 1597–1604.

[25] A. Ullah, "Entropy, divergence and distance measures with econometric applications," *J. Statist. Plan. Inference*, vol. 49, no. 1, pp. 137–162, 1996.

[26] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[27] A. Broggi, M. Bertozzi, A. Fascioli, C. Guarino Lo Bianco, and A. Piazzi, "Visual perception of obstacles and vehicles for platooning," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 3, pp. 164–176, Sep. 2000.

[28] U. Franke and S. Heinrich, "Fast obstacle detection for urban traffic situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 3, pp. 173–181, Sep. 2002.

[29] S. Liu, Y. Huang, and R. Zhang, "Obstacle recognition for ADAS using stereovision and snake models," in *Proc. 17th Int. IEEE ITSC*, Oct. 2014, pp. 99–104.

[30] B. Li, "On the recursive estimation of vehicular speed using data from a single inductance loop detector: A Bayesian approach," *Transp. Res. B, Methodol.*, vol. 43, no. 4, pp. 391–402, 2009.

[31] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, May 2004.

[32] T. Liu *et al.*, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, May 2011.

Bowen Lu received the M.Sc. and Ph.D. degrees from University of Essex, Colchester, U.K., in 2009 and 2014, respectively.

He is currently a Research Associate with the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, U.K. His research interests include Bayesian statistical modeling, data fusion, situation awareness, and machine learning.

Matthew Coombes received the M.Eng. degree in aeronautical engineering from Loughborough University, Loughborough, U.K., in 2010.

He is currently a Research Associate with the Department of Aeronautical and Automotive Engineering, Loughborough University. He has held a private pilot's license for three years. His research interests include situation awareness for unmanned aerial vehicles (UASs) and UAS forced landings.

Baibing Li (M'13) received the B.Sc. degree from Yunnan University, Kunming, China; the M.Sc. degree from Shanghai Jiao Tong University, Shanghai, China; the M.Sc. degree from Vrije Universiteit Brussel, Brussels, Belgium; and the Ph.D. degree from Shanghai Jiao Tong University, in 1991.

He was a Postdoctoral Research Fellow with Katholieke Universiteit Leuven, Leuven, Belgium, and a Research Associate with Newcastle University, Newcastle upon Tyne, U.K. In 2001, he was appointed as a Lecturer at Newcastle University. In 2004, he moved to the School of Business and Economics, Loughborough University, Loughborough, U.K., as a Lecturer, where he was subsequently appointed as a Reader in 2007 and a Professor in 2011. His research interests cover Bayesian statistical modeling and forecasting for Gaussian and non-Gaussian dynamic problems in various management areas. In recent years, much of his work has also involved transport and traffic management such as transportation demand analysis, travel behavior modeling, and intelligent transportation systems.

Prof. Li is a member of the Royal Statistical Society.

Wen-Hua Chen (M'00–SM'06) received the M.Sc. and Ph.D. degrees from Northeast University, Shenyang, China, in 1989 and 1991, respectively.

From 1991 to 1996, he was a Lecturer and then an Associate Professor with the Department of Automatic Control, Nanjing University of Aeronautics and Astronautics, Nanjing, China. From 1997 to 2000, he held a research position and then became a Lecturer of control engineering with the Centre for Systems and Control, University of Glasgow, Glasgow, U.K. In 2000, he moved to the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, U.K., as a Lecturer, where he was appointed as a Professor in 2012. As a Professor of autonomous vehicles, he is currently mainly working on the development of unmanned autonomous systems. His research interests include the development of advanced control, signal processing and decision-making methods, and their applications in aerospace engineering.

Prof. Chen is a Chartered Engineer in the U.K., a Fellow of The Institution of Engineering and Technology, and a Fellow of the Institution of Mechanical Engineers.