Proceedings of the 2007 American Control Conference
Marriott Marquis Hotel at Times Square
New York City, USA, July 11-13, 2007

Autonomous Searching and Tracking of a River using an UAV

Sivakumar Rathinam*, Pedro Almeida, ZuWhan Kim, Steven Jackson,
Andrew Tinka, William Grossman and Raja Sengupta
Center for Collaborative Control of Unmanned Aerial Vehicles, University of California, Berkeley.

***Abstract*— Surveillance operations include inspecting and monitoring river boundaries, bridges and coastlines. An autonomous Unmanned Aerial Vehicle (UAV) can decrease the operational costs, expedite the monitoring process and be used in situations where a manned inspection is not possible. This paper addresses the problem of searching and mapping such littoral boundaries using an autonomous UAV based on visual feedback. Specifically, this paper describes an exploration system that equips a fixed wing UAV to autonomously search a given area for a specified structure (could be a river, a coastal line etc.), identify the structure if present and map the coordinates of the structure based on the images from the onboard sensor(could be vision or near infra-red). Experimental results with a fixed wing UAV searching and mapping the coordinates of a 2 mile stretch of a river with a cross track error of around 9 meters are presented.**

I. INTRODUCTION

This paper describes an exploration system that equips a fixed wing UAV to autonomously search a given area for a specified structure such as river or coastal line, identify the structure if present and map the coordinates of the structure based on images from an onboard sensor(could be vision or near infra-red). We assume that the GPS position of the UAV is known whereas the GPS coordinates of the structure are unknown. Such a system could capture changing coastline or river boundaries, i.e., be helpful in situations where the GPS coordinates of the structure are not exactly known. Since the GPS coordinates of the river boundaries are assumed to be unknown, we first find the river in the given area and then track the river using the UAV. There are two main problems that arise while building such a mapping system:

- Real time detection of the river using images from an onboard camera.
- Control design for tracking the detected river using a fixed wing UAV.

Related to these problems, the following are the main contributions of this paper:

1) An off line learning algorithm and a real time detection algorithm that can identify the river in a given image. The real time algorithm runs at around 3 Hz and works on both color and near infra red images.
2) Given a search area specified by the user, the UAV exploration system can autonomously identify the river in the given area and map its coordinates along a specified river direction.
3) Experimental results with a fixed wing UAV searching for the river and mapping its coordinates for nearly 2 miles of a river with a cross track error of 9 meters in Camp Roberts, California.

II. RELATED VISION BASED FOLLOWING LITERATURE

Vision based following has engaged researchers for nearly two decades. Majority of the work done in this area relevant to this paper is on *ground vehicles* following roads or lanes. The two well know vision based tracking systems for ground vehicles that travel at large speeds are that of Dickmans et al. [1] and Taylor et al. [2]. Dickmans et al. provide a road detection algorithm based on extended kalman filter and employed a control strategy based on full state feedback. The vehicle was tested at speeds up to 100 m/s [1] in the race track(Autobahn). Taylor et al. [2] provide a simple edge detection algorithm and compare different control strategies (lead-lag control, full state feedback and input-output linearization) for the road following problem. The vehicle in the experiments [2] travelled at a speed of around 70 km/hr.

Though there have been many approaches to vision based landing [3] [4] and navigation [5] [6] of UAVs, structure following by small autonomous UAVs is a relatively new area. Our previous vision based road following work on a short runway (refer [7]) was the first contribution in this area without the use of any artificial markings. In [7], a binary classifier is applied on simple pixel properties (such as RGB color values) to discriminate the target (river) from the background. Then, a connected component analysis is applied to extract the target region. However, this approach is not flexible enough to apply to varying colors of river and the background.

In our second vision based following experiments [8], an autonomous UAV tracked an aqueduct in the Crowslanding Airbase, California for a distance of 700 meters based on vision. The vision algorithm proposed in that work had two main components: 1) A semi supervised learning algorithm that extracts the target structure from a single sample image and 2) a realtime detection algorithm that can detect the target structure. The learning algorithm automatically generated a *cross-section profile* of the target structure with the boundaries marked in it. The detection algorithm used this cross-section profile to fit a cubic-spline curve to the target structure. Though this vision algorithm worked on different structures such as roads, highways and canals, the algorithm did not work with the river that could have irregular boundaries. Also the vision algorithm in [8] assumed that the

*Corresponding author: rsiva@berkeley.edu

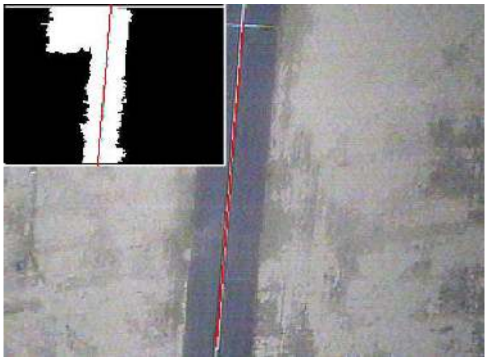


Fig. 1. Our first vision-based UAV navigation on outdoor environment without artificial markings [7]: The experiment was done on a short runway. The road detection was based on a off-line learning (supervised) of lane marking and road pixel colors.

Fig. 2. Our Second vision-based UAV navigation tracking a canal [8]: The experiment was done on a canal stretch of around 700 meters.

structure in each image is almost linear and the intensity profiles are vertically correlated which may not be the case.

A. Challenges and the current approach

The following characteristics of a river makes it difficult to detect relative to roads and aqueducts:

- The boundaries of most rivers are irregular.
- Rivers lack the directional texture which was a key property used in [8].
- In fact, most rivers do not have *any* texture (except an occasional texture generated by accidental sun reflection or churning water at an obstruction).

One might try to detect a river by assuming that that the river in a video image is mostly an uniform region. Then one could apply an image segmentation algorithm to find candidate regions and classify the river by looking at their shapes (thin region crossing the whole image). However, the image segmentation requires non-realtime computation and its results are noise sensitive.

Therefore, in this paper, the idea of [8] is applied where the time consuming segmentation algorithm is used only in a off-line learning procedure. This offline procedure is used to automatically extract the parameters for a pixel classification algorithm. For pixel classification, simple intensity and texture features are used. Texture features are used here

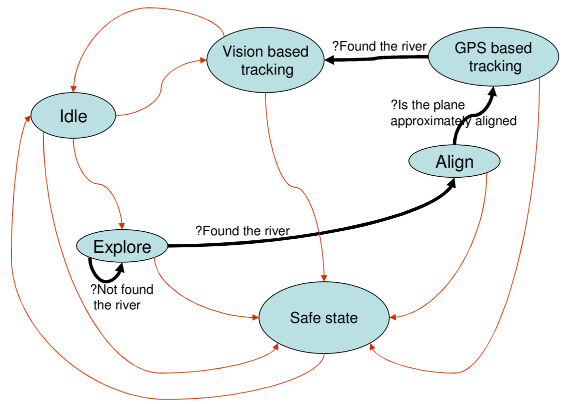


Fig. 3. State transition diagram of the exploration system.

because even though a river may not have any texture, the surrounding land does. Then, a real time algorithm uses the learnt parameters to find the river and non-river components.

Before presenting the detection and control algorithms, the overall design of the exploration system is discussed in the following section.

III. Overall Design of the Exploration System

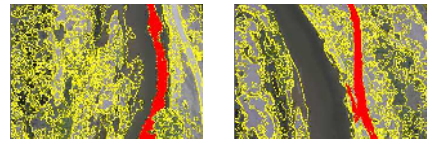
This section explains the different states of the UAV and how the states are switched from one to another. The UAV can be in any one of the following states:

- Idle
- Explore
- Vision-based tracking
- GPS-based tracking
- Align
- Safe state

Figure 3 shows the state transition diagram. The **Idle** state is the initial state of the UAV. In this state, the airplane can be either loitering around any GPS waypoint or tracking a fixed turn rate command. If one wants the UAV to start searching for the river, the human operator can change the state of the UAV from Idle to **Explore**. In the Explore state, the UAV is automatically guided to a circular search area where it should look for the river. The center and the radius of the search area are parameters that can be specified by the human operator (these parameters are specified before each flight and currently we do not change them on the fly). After the UAV reaches the circular search area, the UAV is made to track the circumference of the search area. The real time vision algorithm (section IV) onboard searches for the river in the images collected along the circumference of the river. Once a river segment is found, the GPS coordinates of the river segment are computed using the roll, pitch, yaw angles and the GPS coordinates of the UAV. Then, the UAV automatically transitions to the **Align** state based on the direction specified by the user. In the implementation, the user could choose the UAV direction to be either west to east or east to west. Figure 11 illustrates an example

(a) (b) (c)

Fig. 4. (a) Example image for learning; and (b) the image segmentation result. (c) The river is detected by finding the *longest* region across the whole image.



(a) (b)

Fig. 5. The semi-supervised learning procedure requires user's validation on the classification result due to possible false detections: (a) false detection due to segmentation error, and (b) false detection by a competing structure (nearby road). The use of infrared can reduce the false alarms a lot.

where the user has chosen the east to west direction. In the Align state, the UAV tries to orient itself along the calculated GPS coordinates of the river. Once the UAV has oriented itself, it first transitions to **GPS-based tracking** and then to **vision-based tracking**. In GPS-based tracking the UAV tracks the curve produced by the GPS coordinates of the river. In Vision based tracking, the UAV tracks the curve produced directly by the real time vision algorithm. Though GPS-based tracking may not required, adding this state improved the transition of the UAV from the Align state to the Vision-based tracking state. Both these tracking states involve the following basic control problem: Given a curve, generate turn rate commands such that the fixed wing UAV follows the curve. For this control problem, we use the strategy suggested in [9]. Please refer to [8] for further details. The UAV can be transitioned from any state at any time by the human operator to a state called the **Safe** state. As the name indicates, in the **Safe** state, the UAV is forced to loiter around a known GPS waypoint. In practice, this is very useful as the human operator has the ability to send the UAV to this **Safe** state if the UAV is not following the river well or in the case of other unfortunate events.

IV. River detection algorithm

The proposed algorithm works on both color and near-infrared images. We provide examples of both types. Figures 4-7 are color images and figures 15 are infrared. The near-infrared has a very nice property that water absorbs near-infrared wavelengths and appears black in the image. Therefore, for the field test, a near-infrared camera was used to maximize robustness.

A. Learning

The learning phase first applies the pyramid-linking image segmentation algorithm suggested by Burt et al. [10] [11]. An example image segmentation result is shown in Figure 4b.

In the majority of images, the river component happens to be the longest region crossing the entire image. Hence, the goal is to find such a region across the whole image. The principle axis of inertia is found first for each segmented region by using the image moments. The angle, θ_R, of the principle axis for a given segmented region (R) is defined as:

$$\theta_R = \frac{1}{2}\tan^{-1}\frac{2m_{11}}{m_{20}-m_{02}}, \tag{1}$$

where $m_{ij} = \sum_{q\in R}(x_q-\bar{x})^i(y_q-\bar{y})^j$, $\bar{x} = \sum_{q\in R}\frac{x_q}{|R|}$, $\bar{y} = \sum_{q\in R}\frac{y_q}{|R|}$ and (x_q, y_q) are the image coordinates of pixel q present in the segmented region (R). The longest region is found by examining the variance of the point distribution along the principal axis. The final river detection result is as shown in Figure 4c.

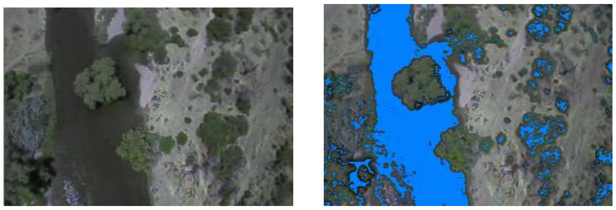
The segmentation-based algorithm is not accurate enough for fully automated off-line learning. False detections occur due to segmentation errors or by competing long regions (such as nearby roads) as shown in Figure 5. Therefore, the learning procedure involves manual validation where a human operator provides additional feedback on whether a result is correct or not. The user intervention is minimal because it requires only a few mouse clicks during the entire learning process even with a poor detection rate (say, 60-70%). The detection rate greatly improved when using the near-infrared sensor because the infrared intensity values of the river region are more uniform than those of the visible color sensor.

Once the result is validated, the pixel statistics of the river and the background are collected. Five features are used (three for near-infrared): RGB color values (total 3 or a single intensity value for the near-infrared) and X- and Y-gradient values for each pixel. To obtain the gradient values, 3×3 Sobel masks were applied. For each feature, its statistics are fitted into a parametric distribution such as a Gaussian or gamma distribution. Based on histograms of the collected experimental data, gamma distributions are used for color or intensity values and Gaussian distributions with zero-mean are used for gradient values.

B. Realtime detection

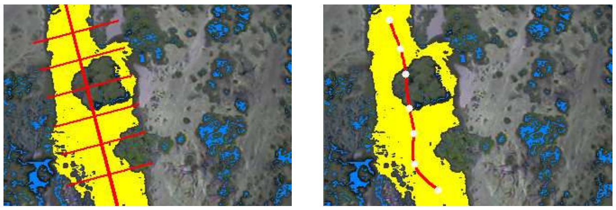
Based on the learnt distributions, Bayesian likelihood is used to classify whether a given pixel belongs to the river or not. For fast calculation, log-likelihood and pre-calculated table-lookup is used for the gamma distributions (RGB and near-infrared intensity can only have 256 values). An example pixel classification result is shown in Figure 6b. There are some false classifications because the color of the river is similar to that of trees. However, the classifier still rejected many of the tree pixels due to the use of texture features.

A connected component analysis is applied to the result of the pixel classification algorithm. The method in Section IV-A is applied to find the longest region. Once the river region is found, a curve is fitted along the center of the river region. The curve fitting procedure is illustrated by figure 7. To fit the river curve, the detected river region is divided into n



(a) (b)

Fig. 6. (a) An example target image and (b) the pixel classification result.



(a) (b)

Fig. 7. Curve fitting procedure: (a) the river region is divided into n ($n = 7$ in our example) regions along the principal axis, and (b) a spline curve is fitted onto the regions' centers of mass.

($n = 7$ in our example) regions along the principal axis of inertia. Then, a spline curve is fitted onto the regions' centers of mass. The final result is shown in Figure 7b.

These image coordinates on the curve are then converted to ground coordinates using the roll, pitch, yaw angles and the height measurements from the sensors onboard of the plane. Then using the GPS information of the UAV, the GPS coordinates of the river can be estimated.

V. Experimental results

The exploration system was tested using a Sig Rascal model aircraft (Figure 8). Low level aircraft control and stabilization was performed by a commercial Cloud Cap Piccolo avionics package (Figure 9) [12]. The camera was mounted at an angle of $\theta = 10$ degrees with respect to the yaw axis of the aircraft (figure 10). The vision and the control algorithms run on a PC104 onboard the UAV. The PC104 communicates directly with the Piccolo avionics through a serial port. The flight tests were conducted at the Camp Roberts, California. Figure 11 is a picture of the river over which the UAV was flown. Figure 11 also shows the safe waypoint where the UAV can be sent in the case of any unfortunate incidents. The circumference of the search area and the direction shown in figure 11 can be specified by the user.

Test video of the river was collected first by flying a helicopter under manual control. The helicopter was flown at an altitude between 100 to 150 meters over the river. The recorded onboard video was used as an input to the learning phase of the river detection algorithm. For the experiments using the fixed wing UAV, the UAV was flown at 135 meters. This height was chosen to image the river with adequate resolution. The vehicle would be able to detect and follow the river if it was flying at a higher altitude but the data collected would not be as useful due to poor resolution. Moreover,

Fig. 8. Sig Rascal Model Aircraft used for flight tests.

Fig. 9. Piccolo avionics package that performs low level flight control and stabilization.

the vehicle was not flown at a lower altitude because the controller would not perform well due to an excessively myopic view of the river.

The GPS coordinates of points on the river as estimated using the vision algorithm during the exploration are shown along with GPS map of the Camp Roberts river in figure 12. We fitted a curve to these estimated GPS coordinates. Figure 13 shows this curve. The average cross track error of the estimated curve with respect to the centerline of the river is approximately 9 meters. The GPS coordinates of the UAV while tracking the river is shown in figure 14. The yellow square region in figure 13 indicates a part of the river where no data was collected. This is also the region where the curve fitting is poor. In this region, the fixed wing UAV was unable to follow sharp turns in the river. Figure 15 shows the processed, near infra red images collected when the UAV was tracking the river.

VI. Conclusions

This paper addressed the problem of searching and mapping a stretch of river using a fixed wing UAV. The assumptions are that the GPS coordinates of the river are not known whereas the GPS position of the UAV is known. A possible future direction would be to solve this problem

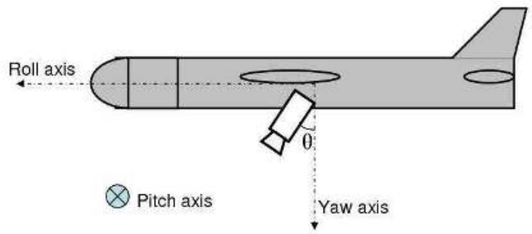


Fig. 10. Camera is oriented at an specified angle θ with respect to the yaw axis. Hence, if the aircraft is flying parallel to the ground plane, then $\theta = 0$ is the case where the camera is looking straight down.

Fig. 11. River in Camp Roberts miltary base, California.

Fig. 12. Coordinates of the river estimated from the vision algorithm during the exploration shown with the map of the Camp Roberts river.

Fig. 13. Estimated coordinates of the river after curve fitting.

Fig. 14. GPS coordinates of the UAV tracking the river.

with faster detection algorithms. Currently the detection algorithm processes the frames individually and does not use information from previous frames. By performing filtering across frames, one can increase the image processing speed and hence as a result the control would be better. If the GPS information of the UAV is also not known, then this searching and mapping problem becomes a variant of a Simultaneous Localization and Mapping (SLAM) problem using vision. This SLAM problem in these outdoor environments with the fixed wing UAVs travelling at 20 m/s would be a difficult and a useful problem to solve.

References

[1] E. D. Dickmanns and B. D. Mysliwetz, "Recursive 3-d road and relative ego-state recognition," *IEEE Trans. Patt. Analysis Machine Intell.*, vol. 14, no. 2, pp. 199–213, 1992.

[2] C. J. Taylor, J. Kosecka, R. Blasi, and J. Malik, "A comparative study of vision-based lateral control strategies for autonomous highway driving," *Int. J. Robotics Research*, vol. 18, no. 5, May 1999.

[3] S. Saripalli, J. Montgomery, and G. Sukhatme, "Vision-based autonomous landing of an unmanned aerial vehicle," *Proceedings of the 2002 International Conference on Robotics and Automation*, 2002.

[4] O. Shakernia, R. Vidal, C. Sharp, Y. Ma, and S. Sastry, "Multiple view motion estimation and control for landing an unmanned aerial vehicle,"

Proceedings of the 2002 International Conference on Robotics and Automation, 2002.
[5] S. Furst and E. Dickmanns, "A vision based navigation system for autonomous aircraft," *Intelligent Autonomous Systems*, vol. 5, pp. 765–774, 1998.
[6] B. Sinopoli, M. Micheli, G. Donato, and T. Koo, "Vision based navigation for an unmanned air vehicle," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1757–1765, 2001.
[7] E. Frew, T. McGee, Z. Kim, X. Xiao, S. Jackson, M. Morimoto, S. Rathinam, J. Padial, and R. Sengupta, "Vision-based road following using a small automonous aircraft," *Proc. IEEE Aerospace Conference*, 2004.
[8] S. Rathinam, Z. Kim, A. Soghikian, and R. Sengupta, "Vision based following of locally linear structures using an unmanned aerial vehicle," *44th IEEE Conference on Decision and Control and European Control Conference*, December 2005.
[9] R. Frezza, G. Picci, and S. Soatto, "A lagrangian formulation of nonholonomic path following," *The Confluence of Vision and Control, A. S. Morse et al. (eds.), Springer Verlag*, 1998.
[10] T. H. H. P. J. Burt and A. Rosenfeld, "Segmentation and estimation of image region properties through cooperative hierarchical computation," *IEEE Transactions of Systems, Man and Cybernetics.*, pp. 802–809, 1981.
[11] I. Corporation, "Open source computer vision library reference manual," 2001.
[12] "Cloud cap technology, inc, piccolo user's guide."

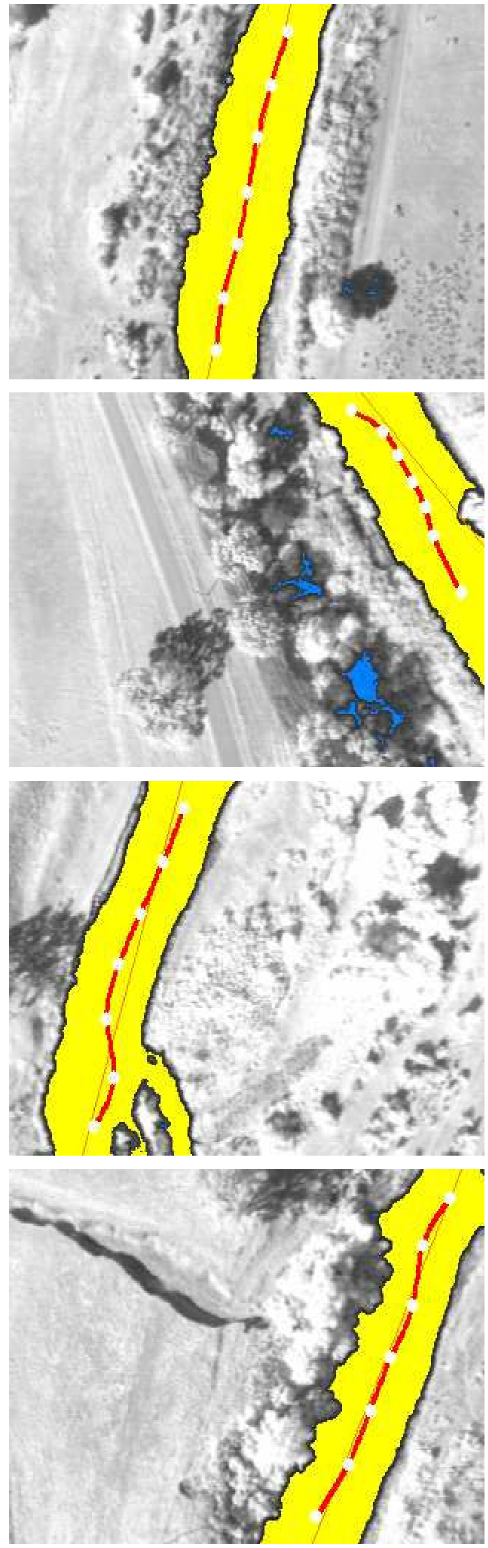


Fig. 15. Few sample images of the onboard processed video.