Object Tracking in Unmanned Aerial Vehicle (UAV) Videos Using a Combined Approach

Shuqun Zhang

Department of Computer Science, College of Staten Island / City University of New York
2800 Victory Blvd, Staten Island, NY 10314
zhangs@mail.csi.cuny.edu

ABSTRACT

UAV videos are difficult to process because of fast abrupt motion, low resolution, noisy imagery, cluttered background, low contrast, and small target size. In particular, object size varies from several to thousands pixels in different videos, which is difficult to handle well using a single algorithm. This paper proposes a switching/combined approach for object tracking in UAV videos, where small objects are tracked using a spatio-temporal segmentation method and larger objects are tracked using a modified statistical deformable model. The proposed snake model addresses the problems in the existing statistical snakes that require a good initialization, manual parameter tuning and high computational complexity. It detects clutter, high gradient noise and partial occlusion, and corrects the object bounding box by automatically adjusting the snake parameters. The computational complexity is significantly reduced by performing the operations only on a small image region and using a fast snake deformation method. The effectiveness of the proposed method is demonstrated using real UAV video sequences.

1. INTRODUCTION

Object tracking is one of the most important and yet difficult problems in video processing. Most of existing tracking techniques achieve good tracking performance when deal with a stationary camera with good image quality. The fast moving camera in UAV often results in abrupt discontinuities in motion. This makes tracking moving or stationary targets in UAV imagery a very challenging subject since accurate motion computation has been and remains a challenging problem. Other factors that make tracking even harder may include low resolution, noisy imagery, cluttered background, occlusion, significant change of scale, low contrast and small size of the target.

Most of existing tracking algorithms are geared for large objects with clear features and boundaries. Small moving objects usually need to be processed using a different algorithm [1]. In our testing UAV videos, the object size varies from 5x5 to about 90x90, and there is no prior information about the videos or object sizes. In this paper, a combined approach is proposed for object tracking in UAV videos, where small objects are tracked using a spatio-temporal segmentation method and larger objects are tracked using a modified statistical deformable model. The size of the object is determined by the spatio-temporal segmentation method. In the modified snake model, several strategies are used to overcome the problems in the existing statistical snakes such as the requirements of an appropriate shape initialization, manual parameter tuning and high computational complexity. To deal with problems such as cluttered background, high gradient noise and partial occlusion, the proposed tracker monitors the size of object bounding box and then automatically corrects it by adding more forces to the snake model.

2. THE PROPOSED METHOD

Our tracking problem is to develop an algorithm that can track UAV objects in real-time, where no prior knowledge about the videos and objects is available and minimal human intervention (a single mouse click on or near the object) is required. The proposed method consists of the following steps. Once an object is clicked, two consecutive frames are extracted from the video and sent for spatio-temporal segmentation to determine the object size. If the object is determined to be small, temporal segmentation is used for tracking. Otherwise, a statistical snake algorithm is applied on the current video frame for segmentation. Then the object bounding box is calculated and its size is monitored to see if it suddenly increases too

much. Large increase in object bounding box size is normally due to strong noise, clutter, occlusion or another-object interference. If it does, a bounding box correction step is applied through either adding extra forces to the snake model or regularizing the object bounding box using the bounding boxes in the past frames. The final step of tracking is performed by segmenting the subsequent frames of video sequence and establishing a correspondence of objects between frames. It is noted that all the above operations except motion estimation are performed on a 101x101 region of interest (ROI) centered at the mouse click point to reduce computation time since objects in UAV videos are relatively small and can be enclosed in this ROI. The above steps are detailed below.

2.1. Spatio-temporal segmentation

A spatio-temporal segmentation method is used for determining the object size and tracking small objects [2]. Here is a brief review and the details can be seen in Reference 2. It consists of the following two steps: motion compensation and object detection.

Motion compensation and "moving" of static objects: The motion parameters of the moving camera are estimated using a 3-parameter affine model (translation + rotation). Two successive frames I_n and I_{n+1} of an image sequence are assumed to have the same luminance value at different times:

$$I_n(x,y) = I_{n+1}(x\cos\theta - y\sin\theta + t_x, y\cos\theta + x\sin\theta + t_y), \quad (1)$$

where t_x, t_y and θ are the horizontal translation, vertical translation and rotation angle, respectively. The three motion parameters can be obtained using the gradient-based method [3]. To improve the accuracy, they are estimated iteratively. Another operation in this step is to apply a 3x3 mean filter to make static objects "move". This is necessary because tracking of static objects is also required and the temporal change-detection in the next step is well known to be incapable of detecting static objects. The "moving" operation will allow static objects to be treated as moving objects, which is done by an image shrinking operation. In change detection, the image differencing typically results in broken object boundaries for a moving object. For a static object, it generates an all-zero image ideally. If the change detection also outputs boundaries for a static object, we can consider the stationary object "moved." A simple way to do this is to shrink the object first and then send it to the change detection along with the original object.

Object Detection: In this step, the standard change detection (an image differencing followed by a thresholding operation) is applied on the two aligned images, I_{n+1} and I_n, to segment the moving object temporally:

$$D(x,y) = T\left[\left|I_n(x,y) - I_{n+1}(x,y)\right|\right], \quad (2)$$

where the threshold for the threshold function T[.] is calculated automatically as 2.7σ, where σ is the standard deviation of the difference image. To separate the object regions from noise, a morphological closing followed by a size filtering are performed on the thresholded image, then the moving object is located by finding the region(s) closest to the mouse click point.

For small or point targets, the object region obtained from the above step is typically everything we need and no further operation is required. However, for larger objects, the region found may be just a small piece of the object and further spatial segmentation is needed. Therefore, Sobel edge detection is applied on the image, and then a morphological dilation is used for edge linking. Next, the edges having a non-empty intersection with the region closest to the mouse click point are identified as belonging to the object. The identified object edges are in turn used to merge object regions, in which those regions having overlap with the object edges are merged.

A threshold is used to separate small and large objects based on the area of the object region obtained above. To increase the robustness of classification, we count the number of times the object is classified as small or large in the first five frames. If the count is more than three times, the object size classification is very possible to be good.

2.2. Statistical snake-based segmentation

It was found that the segmentation results from the previous spatio-temporal method are sometimes not accurate enough due to the errors caused by edge-linking and/or region-merging. The edge-linking in the spatio-temporal segmentation cannot ensure to yield a closed object edge for every image. On the contrary, segmentation using snakes always gives connected edges. So here a region-based statistical snake [4] is adopted for the object detection of larger objects. The statistical snake allows one to take into account statistical properties of the input image and is thus possible to achieve optimal performance under known noise model.

Assume that the image $\mathbf{s} = s(x, y)$ is composed of two non-overlapping areas, the object region Ω_o and the background Ω_b. If the object's gray values $\mathbf{o} = o(x, y)$ and background noise $\mathbf{b} = b(x, y)$ are uncorrelated and independently distributed, both regions can be described

by a corresponding probability density function. We define a binary window function **w** = w(x, y) to separate the object from the background, where w(x, y) = 1 within the object region Ω_o and 0 in the background. The image can be written as the sum of two components:

$$s(x,y) = o(x,y)w(x,y) + b(x,y)[1 - w(x,y)]. \quad (3)$$

The image segmentation therefore is to estimate the most likely shape **w** for the object in the image. The parameter **w** can be estimated through either maximizing likelihood (ML) or maximizing a posteriori (MAP) method. Without any additional a priori knowledge, **w** is selected by maximizing the likelihood function. It has been shown [4] that for Gaussian distribution one can obtain the maximum likelihood segmentation by maximizing

$$E(s,w) = N_o(w)\log[\sigma_o^2(w)] + N_b(w)\log[\sigma_b^2(w)], \quad (4)$$

where σ_o^2 is the variance inside the window **w** and σ_b^2 is the variance outside it; $N_o(w)$ is the number of pixels inside the window and $N_b(w)$ is the number of pixels outside it. In our active contour model, we will minimize the above energy term and add some regularizing terms, like the area of the region inside **w**:

$$E'(s,w) = (1-\lambda)\cdot E(s,w) + \lambda \cdot Area, \quad (5)$$

where λ (<1) is set to be proportional to the size of object bounding box, which will be discussed in Section 2.3.

2.3. Tracking

Segmentation-based object tracking is to separate the object from the background for each frame of a video sequence. For small object tracking, it is relatively easy since the temporal segmentation usually gives accurate object region. For tracking larger objects with the snake, we need to handle snake initialization and deformation, and clutter/noise detection and correction.

Snake initialization and deformation

Snake-based tracking needs to have a specific good initialization in the first video frame since it is very important for segmentation performance. Most of the snake algorithms assume that one can draw an initial polygonal contour near the object manually. However, this is unavailable for our case. On the other hand, for tracking purpose we are interested only in the object's bounding box and not in accurate extraction of object boundary. It is therefore the initial contour (window **w)** for deformation can be chosen as a rectangle. Furthermore, **w** keeps the rectangle shape during contour deformation, and the deformation is performed through moving the four lines of the rectangle window. This can significantly reduce the effects of initial contour as well as the computational cost since each time we move a whole line of pixels instead of a single pixel. For the proposed snake algorithm, a 25x25 square centered at the mouse click point is used as the initial contour. To segment relatively small objects, it is necessary to move the rectangle contour inwards first and calculate the energy term in Eq. (5). If the energy is reduced, then the rectangle lines continue to be moved inwards until the minimal energy is reached. Otherwise, the direction of movement is reversed and the same energy minimization procedure is applied. This segmentation produces a binary window that enables us to locate the target in the image. After the current video frame is segmented, its final object bounding box is used as the initial contour for the segmentation of the next frame. Since the snake-based tracking method does not need motion estimation/compensation step, it will be removed after the object is determined to be large.

Clutter/noise detection and correction

The tracking method described above assumes that there is only one object in the ROI. If another object/clutter enters the ROI or the tracked object is occluded by another object/background, the tracking could fail. Indeed, when the tracked object is close to another object, the snake will increase to enclose both objects together. When the objects move away one from another, the snake model is unable to separate them and continues to track both objects as a single one. Another limitation of the snake model is its sensitivity to high gradient values of background pixels (considered as noise), for example, edges of road and runway. It is observed that the object's bounding box tends to become much larger than the expected in the above mentioned cases. Therefore the sudden increase in object bounding box can be used as the detection condition for clutter, strong noise, occlusion and other problems. Our proposed solution is to automatically adjust the value of λ in Eq. (5) according to size of object bounding box. Larger value of λ means that more pressures are added to the snake so the bounding box can be pushed toward the true object boundary. By linking the size of object bounding box with the snake parameter, we can eliminate the problem of manual parameter tuning. This has been shown to work quite well.

3. TRACKING RESULTS

The proposed tracking algorithm is illustrated on several real image sequences. In all the experiments, the only human intervention is a mouse click. The threshold for separating small and large object is 80 pixels. The first tracking experiment is performed on an IR sequence with a point target. The result is shown in Fig. 1. The low resolution IR imagery has high ego motion and lacks

texture and shape information of the target. The low contrast of the object makes tracking even harder. The accuracy of motion estimation affects the tracking performance.

The second experiment is performed on those videos with larger objects to demonstrate the performance of the proposed snake algorithm. The tracking result is shown in Fig. 2, where the parameter λ is set to be 0.29. The segmentation of vehicles is very good because the rectangle nature of vehicles fits the proposed statistical model well. The tracking results from other videos (not shown) are also quite promising.

Fig. 3 shows the results before and after snake parameter adjustment when the object is interfered by another object or clutter. The parameter λ is adjusted to be 0.55 automatically after clutter/noise detection. In Fig. 3(a), a cross is moved toward the object and then moves away. The experimental result shows that the tracker can separate the tracked object from the cross after automatic parameter adjustment. Fig. 3(b) shows that a vehicle is approaching two big clutters, the object's bounding box will increase to enclose one of the clutter. Again, the object's bounding box is reduced to fit the object after more weights are given to the parameter λ.

4. CONCLUSION

In summary, a combined method has been proposed for detecting and tracking both stationary and moving objects in UAV videos by integrating a spatio-temporal segmentation and a modified statistical snake algorithm. In particular, the original statistical snake algorithm has been improved to make it suitable for our tracking application, which includes minimizing the effect of initial contour, eliminating manual parameter tuning, and reducing computational complexity. By linking the size of object bounding box with snake parameters, the proposed snake algorithm can handle problems such as clutter, strong noise and partial occlusion.

This work was supported by Air Force Research Laboratory in Rome, New York. The author would like to thank Todd Howlett for his support.

REFERENCES

[1] D. Davies, P. L. Palmer, M. Mirmehdi, "Detection and tracking of very small low contrast objects," Proceedings of the 9th British Machine Vision Conference, Sept. 1998.

[2] S. Zhang and M. A. Karim, "Automatic target tracking for video annotation," *Op. Eng.* 43, 1867-1873, 2004.

[3] M. Irani and S. Peleg, "Improving resolution by image registration," CVGIP: Graph. Models and Image Process. 53, 231-239, 1991.

[4] C. Chesnaud, P. Refegier, and V. Boulet, "Statistical region snake-based segmentation adapted to different physical noise models," IEEE Trans. Patt. Anal. Mach. Intell. 21, 1145-1157, 1999.

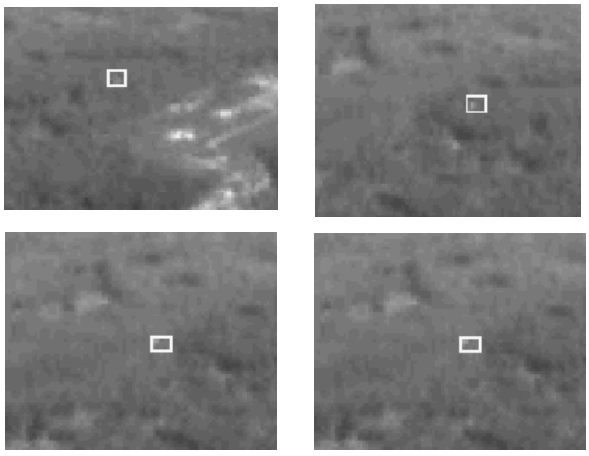

Fig. 1. Tracking results of a small point target, where the interval between two images is about 30 frames.

Fig. 2. Tracking results of a larger vehicle, where the interval between two images is about 20 frames.

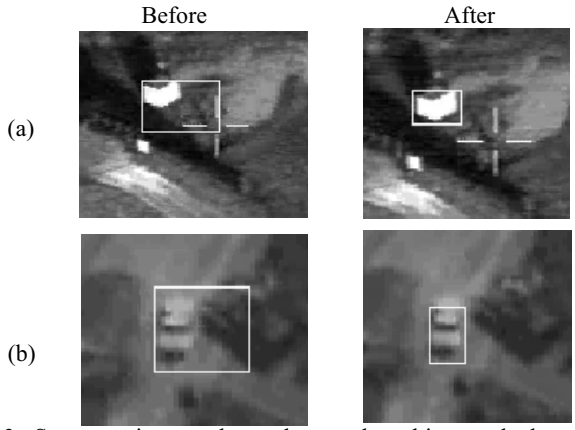

Fig. 3. Segmentation results under another-object and clutter interference, before and after automatic parameter adjustment.