Moving object detection for unconstrained low-altitude aerial videos, a pose-independant detector based on Artificial Flow

Thomas Castelli[a,b]
[a]Survey Copter
Airbus Defense and Space
Pierrelatte, FRANCE

Alain Trémeau[b], Hubert Konik[b], Eric Dinet[b]
[b]Laboratoire Hubert Curien
Université Jean-Monnet
Saint-Etienne, FRANCE

***Abstract*—Automatic detection of moving objects is an important task for aerial surveillance. It has been a popular and well-studied subject for the computer vision community, but is still a challenge. The method we introduce targets surveillance low-altitude mini and micro-UAVs. We take advantage of the inherent image motion on footage captured by such aerial vehicles. Our method confronts Optical Flow vectors and an estimated Flow in order to detect independently moving pixels. This motion-based approach is robust to operational conditions and to the geometric properties of the scene. The efficiency of the method was computed on the VIVID database. The moving areas detected will make the tracking task more robust and efficient.**

Keywords—aerial; UAVs; Moving Object Detection; dense optical flow; artificial flow

I. INTRODUCTION

The use of Unmanned Aerial Vehicles (UAVs) is rapidly growing for civil applications, and quickly became a necessity for the military. Computer vision brings a significant advantage in terms of functionality and ease of use for the operators. A robust moving object detection algorithm is crucial for surveillance UAVs as it allows the operator to rely on an automated system that highlights interesting areas for him. Such a particular function has been widely studied in the research community. Many approaches were proposed to achieve it on mini-UAVs' images. The limitations for the computer vision algorithms are caused by the intrinsic properties of mini and micro-UAVs: low altitude, ego-motion, perspective, limited payload and also the vast variety of scenes it can encounter (urban areas, forest, desert etc…). The inherent problems we need to deal with in terms of image processing are: fast and unconstrained image motion, change of objects' appearance, partial or full occlusion, parallax, too many or not enough image features, and SWaP (Size, Weight and Power) constraints that limit the computational capabilities of embedded image processors.

The work presented in this paper aims to a specific application. Our method has been designed to be suitable for aerial images captured by low altitude mini-UAVs for surveillance and to tackle their aforementioned constraints.

Many different methods were proposed in the literature during the last 15 years. The background modelling approach uses a Gaussian mixture model for each pixel. It is widely used in video surveillance with fixed cameras, and has been adapted to aerial imagery by adding a step of image stabilization or image registration to cancel motion [1–4]. It is suitable for wide area or high-altitude images because there is a high overlap between images due to low image motion, but low-altitude UAVs' fast moving scene need a close to perfect ego-motion compensation, which is not realistic in operational conditions due to previously mentioned limitations. It also induces a latency of 5 to 20 frames, and is thus not applicable in the targeted application. In [5] and [6] Xiao *et al.* use reference images for geo-registration by using an external database such as TerraServer or GIS. In [5] they then segment the image and use monocular structure from motion to estimate the depth of the scene in order to identify buildings, roads and trees. It is an efficient way of preventing false detections due to parallax but it is also computationally intensive and requires a detailed database, thus limiting applications to high-resolution satellite reference images. A popular temporal approach is the track before detect approach [7–9]. The principle is to generate tracklets using interest points, those tracklets are then analyzed using temporal and spatial clustering scheme to output moving objects. This needs to buffer images and thus leads to a delay. In addition, as it relies entirely on automatic feature selection, it cannot be robust to the diversity of textureless scenes a UAV can encounter.

Another category of methods relies solely on motion acquired from the video stream. The motion is either estimated by frame differencing [10], [11], motion layers [12] or Optical Flow [13],[14]. Two or three frame differencing is a concept that is not robust for our application; depending on the motion of the UAV and of the moving objects, the mask will not represent the entire object and will not have the same shape at all from one mask to the next. Optical flow also has limitations, but in its dense implementations [15],[16] it keeps the shape of a rigid moving object. Aerial images captured from a mini-UAV are very dynamic; motion is consequently inevitable and needs to be taken advantage of. Narayana *et al.* presented a motion segmentation method using the orientation of the flow's vectors [17]. They use a probabilistic model to estimate the number of regions in order to properly segment the image. Rodrigez-Canosa *et al.* published a similar approach in [14], they compare an Optical Flow to an Artificial Flow computed from FAST feature selection and a homography transformation

The research was supported by a DGA-MRIS scholarship

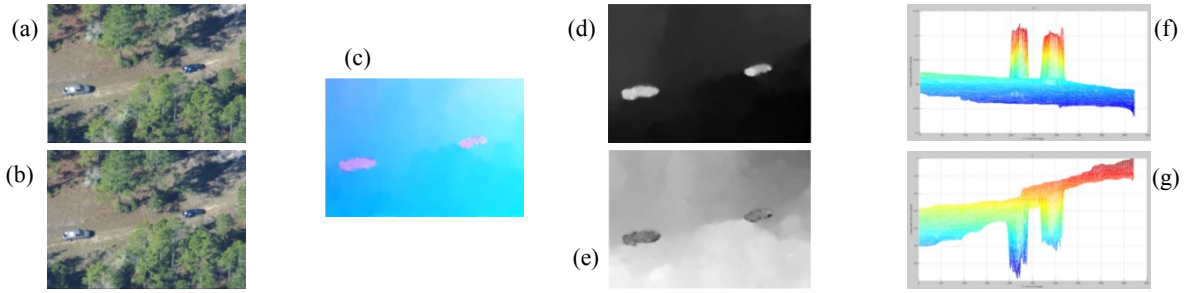

Fig. 1. (a) input image 1 (VIVID egtest05, frame00040), (b) input image 2 (frame00041), (c) dense Optical Flow, (d) orientation (in grey levels), (e) magnitude (in grey levels), (f) 3D orientation (in false colors) and (g) 3D magnitude (in false colors).

using Parallel Tracking and Mapping (PTAM) algorithm. They mentioned limitations such as the need to adapt algorithm parameters depending on the flight scenario, using a ground facing camera (in nadir configuration), the calibration phase before takeoff and the impossibility to recover from an abrupt displacement of the UAV leading to a PTAM failure.

Our method is motion based and also takes advantage of the confrontation between an Optical Flow and an Estimated Flow while being straight forward and simple to implement. Global image motion is inherent to the targeted application of low-altitude mini-UAVs; it is the result of the displacement of the UAV and the gimbal. Our approach uses this fact to efficiently segment independently moving objects from the global motion of the scene.

The paper is organized as follows. In section 2 we describe the method we use to detect moving objects. In sections 2.1 and 2.2 we detail first our general segmentation approach using the flow vectors' orientation, and next a derived Artificial Flow-based method made to improve robustness for difficult cases by using both the vectors' orientation and amplitude. Next in section 3 we present the obtained results on the well-known VIVID database, and lastly in section 4 we discuss future work.

II. MOVING OBJECT DETECTION

A. *Segmentation using the Optical Flow's orientation*

The principle of the proposed approach is to identify the areas of interest only using motion. The input for segmentation is a dense Optical Flow field made of vectors with orientation and magnitude [15]. Orientation is an efficient cue in our targeted application because it is robust to: 1) the lack of planarity of the scene and 2) the lack of perpendicularity between the scene and the camera's sensor. Magnitude is highly dependent on the angular relation between the camera and the captured scene. For example in the case of a sideways camera translation aimed downwards from the horizon to a planar and horizontal surface, the orientation of the flow field will be constant throughout the image, but the relative movement will not be the same. We can see in Fig. 1 that the orientation of the vectors describing the background, represented by grey values in (d) and shown in 3D in (f), are very similar in the whole image; but that the magnitude of these vectors, represented by grey levels in (e) and shown in 3D in (g), varies greatly (from bottom-left to top-right) due to the perspective. This effect is an issue for any thresholding method we could apply to the corresponding image due to the grayscale gradient.

We therefore suggest using the vectors' orientation for our purpose. We start from the assumption that low-altitude aerial images have a global motion due to the movement of the UAV and its gimbal. We then need to estimate this motion to be able to segment independently moving objects. Usually, the targeted object represents a small part of the image; then we can threshold the orientation map from the average orientation of the dense Optical Flow. Therefore, we compute the mean and standard deviation of the orientation map, next defined two boundary values that will be used as thresholds. Those two values are defined by (1) and (2):

$$lowThreshold = \bar{x} - 3.5\,\sigma \quad (1)$$

$$hiThreshold = \bar{x} + 3.5\,\sigma \quad (2)$$

with (3) and (4) respectively the mean and standard deviation of the Optical Flow vectors' orientation:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (3)$$

$$\sigma = \left(\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^{\frac{1}{2}} \quad (4)$$

The value *3.5* has been determined empirically and was found to be efficient for different types of situations encountered in the VIVID dataset and other proprietary UAV footage; it is set and does not need to be changed to fit particular scenarios.

B. *Segmentation using Artificial Flow*

Aerial surveillance sometimes induces specific scenarios for which segmentation using the vectors' orientation may fail.

Fig. 2. Dense Optical Flow of images (a) 40-41 and (b) 58-59 of Egtest05 from VIVID. While image (a) is easy to process with the orientation-based segmentation method, image (b) is more challenging due to the coherency of vectors' orientations (represented here by the same hue).

Those scenarios arise when only vectors' magnitude provides the needed information. For example, the orientation-based method may fail when either the relative movement of a moving object inside a frame is close to zero, or when the orientation of the moving object and of the scene are the same, as shown in Fig. 2. Those two cases cannot be properly segment-

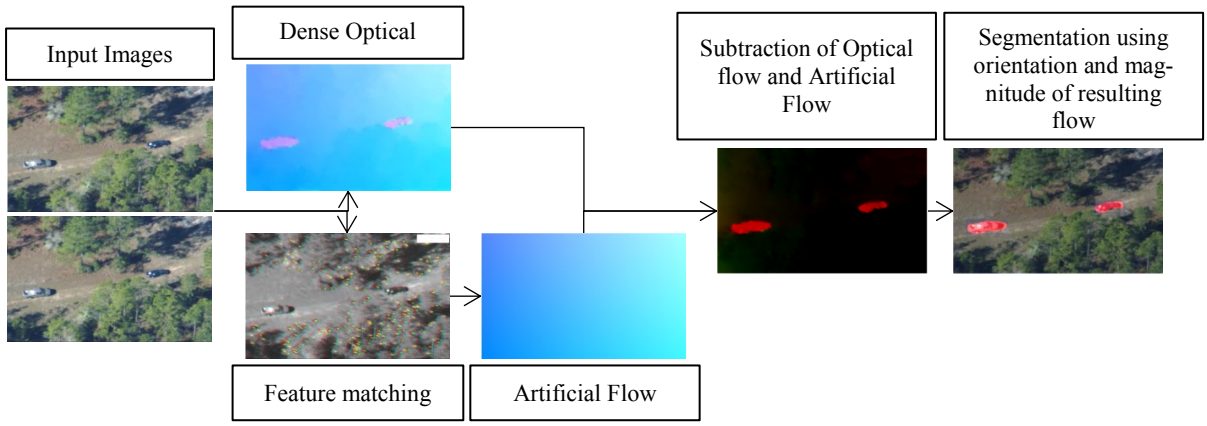

Fig. 3. Flowchart of the proposed algorithm.

ed using only orientation, thus a solution combining orientation and magnitude must be used.

The method we propose to overcome the problem described above is defined by the framework shown in Fig. 3. The magnitude cannot be used right away because of its dependency to scene structure and perspective; consequently it has to be compensated. Therefore, we propose to use the inlier keypoints matched in two consecutive frames to compute first an affine projective transformation. The resulting matrix is then used to compute an artificially corrected Optical Flow field (as in [14]). Next, before computing the final segmentation step, the two flows are subtracted to detect the moving objects from the background, thus getting rid of the slope effect on magnitude caused by the perspective, as shown in Fig. 4.

The affine transformation computed from keypoints, despite its limitation to represent only flat planes, is relevant in our case because the segmentation method applied in the last step is parameterized by an interval of values of $\pm 3.5\,\sigma$, thus making the whole method immune to small motion irregularities caused by noise or parallax.

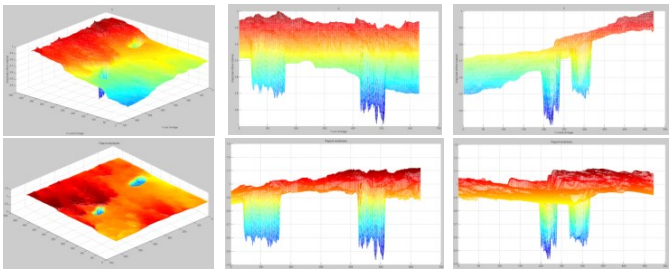

Fig. 4. Magnitude of the dense Optical flow (from left to right): 3D view, Y-Z view and X-Z view. 1st row: Optical flow's magnitude. 2nd row: Artificially corrected flow's magnitude.

III. Experimental Results

As mentioned before we primarily worked on the VIVID database [18] (Fig. 5) as it is the closest to our targeted application. The ground truth data is available, moreover we can target one moving object independently of the others, and resulting masks are available for every ten frames. Unfortunately, the evaluation website dedicated to VIVID [18] is not accessible. All papers using VIVID made therefore their own handmade ground truth [19],[13]. We will firstly present results alongside the state of the art methods. We computed the metrics as close as possible to the methods described in the different papers. In a second section, to be able to compare efficiently our detection method with future techniques and for better clarity, we decided nevertheless to use only the ground truth packaged with VIVID. It has the advantage to be publicly available and to be included with the dataset.

Fig. 5. Images from the VIVID Dataset. Top line images 0, 550 and 1820 from EgTest01. Middle line images 0, 760 and 1830 from EgTest04. Bottom line images 0, 830 and 1530 from EgTest05.

VIVID database is a challenging dataset for moving object detection and tracking, vehicles are moving along roads or open areas and are often occluded by each other or vegetation, change of illumination and camera viewpoint cause appearance and shape variation. Objects are usually 20 by 50 pixels in height and width. The ground truth has also limitations. The most challenging limitations of this dataset are: - it only describes one object per frame even if there is multiple moving objects; - it contains several masks with missing data; - often several consecutive frames are the same, which doesn't impact the tracking process but impacts the moving objects detection. Another important remark: masks do not include the object's shadow (Fig. 6).

Fig. 6. (at left) result of our moving object detection method, (at the center and at right) the red area represents the result of our method, the yellow area represents the ground truth, and the green area represents the ground truth pixels not detected by our method (the right image corresponds to a zoom on the car on the left part of the center image).

In the following section we will compare to the state of the art with different metrics, firstly with a *Correct detection Ratio* and *Miss Detection Ratio* and then the overlap area between the detected object(s) and the ground truth.

Hasan *et al.* present their technique in [20]. After a video stabilization step based on a homography, they use motion and appearance cues to detect moving regions. A tracking before detection framework is used to generate tracklets that will be afterwards merged using graphs. They compute metrics as follows:

$$Correct\ Detection\ Ratio = \frac{Nb.of\ Correct\ Detections}{Nb.of\ System\ Detections}$$

$$Miss\ Detection\ Ratio = \frac{Nb.of\ Miss\ Detections}{Nb.of\ Ground\ Truth\ Detections}$$

with a correct detection being a detection having at least a 50% overlap with the ground truth. We compare our method with Hasan's GMAC (Geometric, Motion and Appearance Constraints aerial video tracker) [20], and four other trackers: MIL (Multiple Instance Learning) [21], OAB1 (Online Adaboost) [22], OAB5 (modified Online Adaboost) [21], and MS+PF (Mean Shift Particle Tracker) implemented by [20] in Fig. 7 and 8.

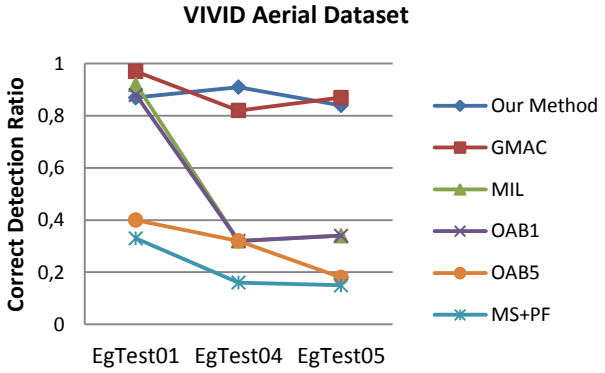

Fig. 7. Correct detection ratio computed according to [20] and compared with state of the art techniques : Hasan's GMAC [20], MIL [21], OAB1 [22], OAB5 [21], and MS+PF [20]

EgTest 01, 04 and 05 were chosen because they are challenging sequences with common operational scenarios. EgTest04 has object appearance and size change along with target occlusions. EgTest01 is favorable to tracking techniques as there are no occlusions, but appearance and size change are still present. And EgTest05 has a lot of targets in trees' shadows and parallax robustness can be tested with the tall trees and the lack of a flat ground plane.

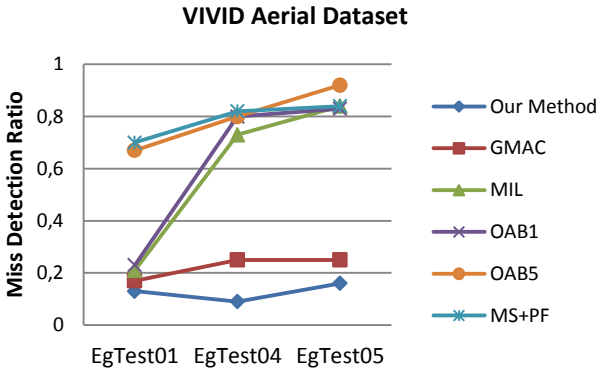

Fig. 8. Miss detection ratio computed according to [20] and compared with state of the art techniques : Hasan's GMAC [20], MIL [21], OAB1 [22], OAB5 [21], and MS+PF [20]

The *Correct Detection Ratios* obtained on the tested VIVID dataset sequences (Fig. 7) are comparable to the state of the art methods while having lower *Miss Detection Ratios* (Fig. 8), demonstrating the efficiency of our moving object detection method.

Siam *et al.*, in [19], [23] and [24], compared their methods with tracking techniques such as Mean Shift [25], Fg/Bg Ratio [25], [26], Variance ratio [27], Peak difference [27], and Adaptive tracker [28]. The computed average overlap is the average of the percentage of overlap between the bounding box of the ground truth and the bounding box of the tracked object for every 10^{th} frame.

In Fig. 9, despite our detection-only method without any tracking scheme, we perform better average overlaps compare to the other techniques.

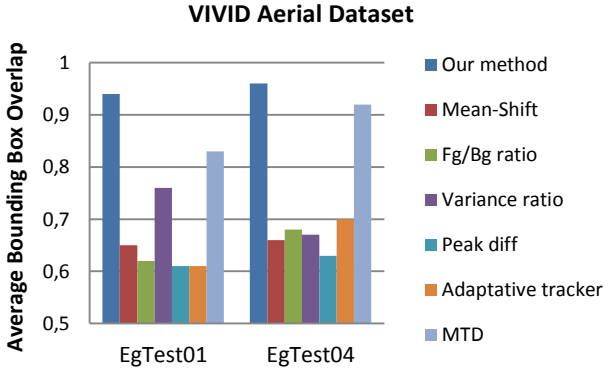

Fig. 9. Average overlap compared to six state of the art tracking techniques : Mean Shift **[25]**, Fg/Bg Ratio **[25], [26]**, Variance ratio **[27]**, Peak difference **[27]**, Adaptive tracker **[28]**, and Moving Target Detection [23].

Next section concentrates of moving object detection without tracking. We compared two Optical Flow based methods with ours. Considering the limitations and the remark mentioned above about VIVID, we computed the recall and precision values as follows. As only one object is described in the ground truth, we only considered the detected region(s) corre-

sponding to this object. For example in Fig. 6, our method detects two objects. As the car on the right part is not described in the VIVID database's ground truth masks, we did not take it into account in the computation of the recall and precision (as defined in [19]). Results for three sequences of the VIVID dataset are presented in Fig. 10 and 11:

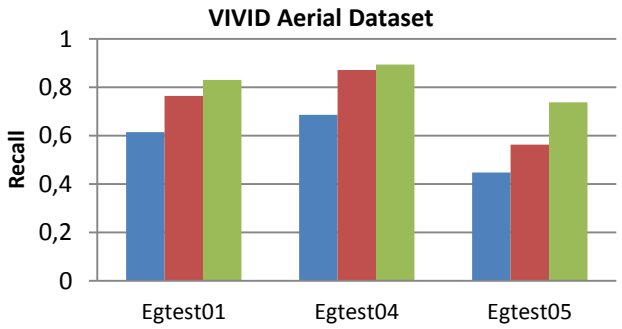

Fig. 10. Recall values obtained using the VIVID database and the ground truth. Blue: Vector's orientation, Red: Vector's orientation and magnitude and Green: Vector's orientation and magnitude using Artificial Flow.

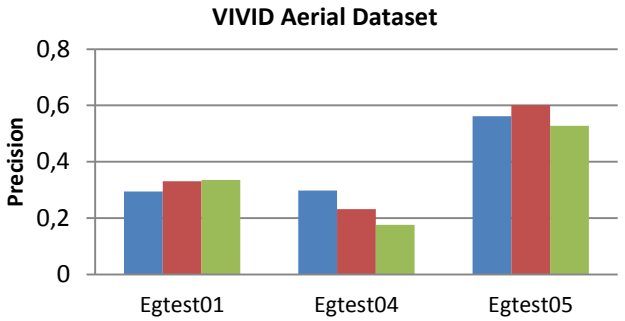

Fig. 11. Precision values obtained using the VIVID database and the ground truth. Blue: Vector's orientation, Red: Vector's orientation and magnitude and Green: Vector's orientation and magnitude using Artificial Flow.

Our results are comparable to the ones introduced in [19], [13] and [24], but unlike them, we used in our evaluation the regular ground truth provided with VIVID, which does not include the object's shadow. This impacts our results by a precision loss due to the shadow. In our motion detection based approach the shadow of objects are an integral part of the moving objects since it has the same behavior. If we remove the shadow from the objects as described in [29], the evaluation results are significantly better, gaining up to a factor 3 on precision, and exceed the performance of other methods introduced in [19], [13] and [24].

Fig. 12. Result of our moving object detection method, the red area represents the result of our method, the yellow area represents the ground truth. The center image corresponds to a zoom on the car corresponding to the ground truth. The right image corresponds to the same room but here before moving detection cast shadows were removed.

For example, in Fig. 12, before motion detection we applied a pre-processing step to remove the cast shadow related to cars in order to perform a more fair comparison in regards to the ground truth. The obtained results are now significantly higher in terms of precision and prove that our method, despite lower recall values, is efficient.

Our method has also shown to be robust to parallax, we can see in Fig. 6 that the tall trees are not detected as moving objects, same remark in Fig. 10, the poles are not detected either. The best example is shown in Fig. 13 on images from Seq1 of VIVID. We can distinctly see the water tower structure detected as moving differently from the background on the Optical Flow, but our method still manages to correct it with the Artificial Flow in order to detect only the proper objects. We achieve 73% of average overlap on Seq1.

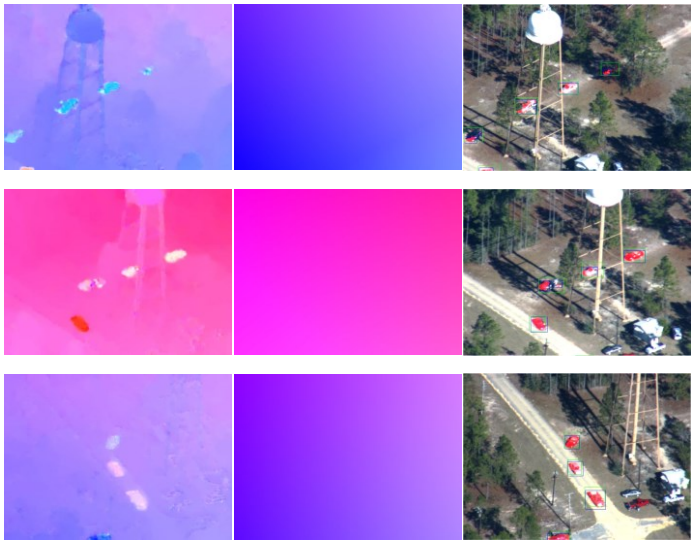

Fig. 13. Images 900 (top line), 992 (middle line) and 1257 (bottom line) from the VIVID Dataset Seq 1. Left column images are Optical Flows, middle columns are Artificial Flow and right columns are the Moving Object Detection result. Green boxes represent the ground truth and blue bounding boxes represent our result, red highlighted areas are pixel-wise result.

If we keep in mind that we do not have any tracking scheme, and that the computation is performed pixelwise and does not include higher level interpretations, we can claim that the proposed method is efficient. In some applications it does not matter if the detection method is sensitive to cast shadows, in most cases the objective is only to localize moving objects in a sequence and not to segment each object perfectly.

IV. Conclusion and Future Work

We developed a method capable of detecting independently moving objects from aerial images. We took advantage of the constant motion on such aerial footage. The confrontation between an Optical Flow and an estimated Flow makes our approach efficient, untroubled by scene geometry such as perspective, and also demonstrated robustness to parallax. Our results presented on the VIVID database are comparable to the state of the art methods despite the fact that we achieved a pixelwise detection and did not benefit from any kind of tracking scheme. This method has also been tested on proprietary UAV footage and may be used for monitoring purpose.

The work presented here is not yet running in real time, but can be if adapted to run on GPU. We have planned to add a tracking scheme to our detection method that will take the detected objects as input to improve the monitoring task. We have also planned to work with an estimated Flow method based not on image keypoints but estimated from the onboard sensors and a digital elevation map to further improve the robustness of the detection and make it faster.

ACKNOWLEDGMENT

The research was supported by a DGA-MRIS scholarship, and Survey Copter.

REFERENCES

[1] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, 1999, vol. 2.

[2] S. Ali and M. Shah, "COCOA - tracking in aerial imagery," *Proc Int Conf Comput. Vis.*, 2005.

[3] W. Yu, X. Yu, P. Zhang, and J. Zhou, *A New Framework of Moving Target Detection and Tracking for UAV Video Application*, vol. 37. Part B3b. 2008.

[4] Y. Lin, Q. Yu, and G. Medioni, "Efficient detection and tracking of moving objects in geo-coordinates," *Mach. Vis. Appl.*, 2011.

[5] J. Xiao, H. Cheng, F. Han, and H. Sawhney, "Geo-spatial aerial video processing for scene understanding and object tracking," presented at the Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[6] J. Xiao, H. Cheng, H. Sawhney, and F. Han, "Vehicle detection and tracking in wide field-of-view aerial video," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 679–684.

[7] Kimura, M., Shibasaki, R., Shao, X., and Nagai, M., "Automatic extraction of moving objects from UAV-borne monocular images using multi-view geometric constraints," *IMAV 2014 Int. Micro Air Veh. Conf. Compet.*, Aug. 2014.

[8] X. Tong, Y. Zhang, T. Yang, and W. Ma, "Automatic Object Tracking in Aerial Videos via Spatial-temporal Feature Clustering," in *Intelligence Science and Big Data Engineering*, Springer, 2013, pp. 78–85.

[9] A. Kundu, K. M. Krishna, and C. Jawahar, "Realtime motion segmentation based multibody visual SLAM," in *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*, 2010, pp. 251–258.

[10] B. Jung and G. S. Sukhatme, "Detecting moving objects using a single camera on a mobile robot in an outdoor environment," in *International Conference on Intelligent Autonomous Systems*, 2004, pp. 980–987.

[11] Z. Yin and R. Collins, "Moving object localization in thermal imagery by forward-backward MHI," in *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, 2006, pp. 133–133.

[12] X. Cao, J. Lan, P. Yan, and X. Li, "Vehicle detection and tracking in airborne videos by multi-motion layer analysis," *Mach. Vis. Appl.*, vol. 23, no. 5, pp. 921–935, Sep. 2012.

[13] S. Dey, V. Reilly, I. Saleemi, and M. Shah, "Detection of independently moving objects in non-planar scenes via multi-frame monocular epipolar constraint," in *Computer Vision–ECCV 2012*, Springer, 2012, pp. 860–873.

[14] G. R. Rodríguez-Canosa, S. Thomas, J. del Cerro, A. Barrientos, and B. MacDonald, "A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera," *Remote Sens.*, vol. 4, no. 4, pp. 1090–1111, 2012.

[15] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Computer Vision-ECCV 2004*, Springer, 2004, pp. 25–36.

[16] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L 1 optical flow," in *Pattern Recognition*, Springer, 2007, pp. 214–223.

[17] M. Narayana, A. Hanson, and E. Learned-Miller, "Coherent Motion Segmentation in Moving Camera Videos using Optical Flow Orientations," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, 2013, pp. 1577–1584.

[18] R. Collins, X. Zhou, and S. K. Teh, "An open source tracking testbed and evaluation web site," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2005, pp. 17–24.

[19] M. Siam and M. ElHelw, "Robust autonomous visual detection and tracking of moving targets in UAV imagery," in *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*, 2012, vol. 2, pp. 1060–1066.

[20] M. Hasan, "Integrating Geometric, Motion and Appearance Constraints for Robust Tracking in Aerial Videos," 2013.

[21] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 983–990.

[22] H. Grabner, M. Grabner, and H. Bischof, "Real-Time Tracking via On-line Boosting.," in *BMVC*, 2006, vol. 1, p. 6.

[23] M. Siam, R. ElSayed, and M. ElHelw, "On-board multiple target detection and tracking on camera-equipped aerial vehicles," in *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, 2012, pp. 2399–2405.

[24] M. Siam and M. Elhelw, "Enhanced Target Tracking in UAV Imagery with PN Learning and Structural Constraints," in *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*, 2013, pp. 586–593.

[25] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *Pattern Anal. Mach. Intell. IEEE Trans. On*, vol. 25, no. 5, pp. 564–577, 2003.

[26] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," 1998.

[27] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *Pattern Anal. Mach. Intell. IEEE Trans. On*, vol. 27, no. 10, pp. 1631–1643, 2005.

[28] J. Wang and Y. Yagi, "Integrating color and shape-texture features for adaptive real-time object tracking," *IEEE Trans. Image Process.*, vol. 17, no. 2, pp. 235–240, 2008.

[29] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti, "Improving shadow suppression in moving object detection with HSV color information," in *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, 2001, pp. 334–339.