Autonomous Person Detection and Tracking Framework Using Unmanned Aerial Vehicles (UAVs)

Hajer Fradi, Lorenzo Bracco, Flavia Canino and Jean-Luc Dugelay
Digital Security Department, EURECOM, Sohpia Antipolis, France
Email: {hajer.fradi, jean-luc.dugelay}@eurecom.fr

***Abstract*—While person tracking has made significant progress over the last few years, most of the existing approaches are of limited success in real-time applications using moving sensors. In particular, we emphasize in this paper the need for a visual tracker that enables autonomous navigation functionality to drones (UAVs), mainly to follow a specific target. To achieve this goal, a color-based detection framework is proposed. The approach includes as well the execution of control commands, which are essential to switch from the detection to automatically follow the detected target. Our proposed approach is evaluated on videos recorded by drones. The obtained results demonstrate the effectiveness of the proposed approach to accurately follow a target in real-time, and despite different challenges such as lighting changes, speed, and occlusions.**

***Index Terms*—Detection, Tracking, Moving Sensors, Color, Drone Commands, Real-time**

I. Introduction

Automatic detection and tracking of people in video data is a common task in the research area of video analysis and its results lay the foundations of a wide range of applications such as visual surveillance, intelligent transportation, augmented reality, and traffic control. In this context, intensive study has been conducted by proposing different methods either for single object or multiple object tracking [1]–[5]. However, most of the existing methods are not applicable to moving sensors since there is no static background and due to multiple reasons such as changes in the target appearance, scene motion, noise in the video and rapidly changing lighting conditions. Therefore, accurate object tracking from a moving camera remains challenging [6]–[8].

For the aforementioned reasons, features based tracking methods [3] are mostly employed in the case of moving sensors. These methods aim at building an appearance model which can deal with changes in the dynamic environment conditions (illumination and background) and changes in the target appearance (pose variations, viewpoint, abrupt motion, occlusions and non-rigid deformation). Existing features based tracking methods generally make use of a pixel-wise or a spatial representation of the target from different features such as pixel intensity, color, texture and edges [6], [8].

In this paper, we particularly aim at proposing a visual tracking framework using drones. The technology of Unmanned Aerial Vehicles (UAVs), commonly known as drones, is becoming nowadays more and more widespread in various applications mainly for humanitarian missions [9], surveillance applications [10], [11] (to ensure safety control, and emergency contingency plan, or to follow a suspicious person) and guiding applications (guiding persons in large building and assisting disabled people). The increasing interest of drones emphasizes the need for autonomous navigation functionality [12], [13] instead of manual control. In the same context, we intend in this work to add autonomous functionality to mini-drones in order to enable fully automatic target tracking without human control.

In this paper, our primary objective is to build an accurate visual representation of the target that has to deal with both of the changing environment and the changing target appearance. The second objective is to apply this visual representation in order to enable following the target by mini-drones. This application will add other challenges in addition to the classical difficulties hampering any tracking algorithm mainly about real-time requirements.

The remainder of this paper is organized as follows: In the next Section II, we present our proposed approach for real-time detection and tracking using mini-drones. Detailed experimental results follow in Section III. Finally, we briefly conclude and give an outlook of potential future works.

II. Proposed Approach

In this section, we present our proposed approach for real-time single object detection and tracking using mini-drones. The approach essentially consists of two steps. First, person detection is performed using an appearance model based on a visual representation and statistical measurement. Second, the drone is automatically controlled through a set of commands in order to follow the detected target. The remainder of this section describes each of these system components.

A. Detection framework based on visual characteristics

Since our primary goal is to autonomously follow a specific target by drones, an accurate detection is essentially required. To achieve that, color cue is selected as an appropriate feature that could discriminate the target from the background and from other objects. This cue has the advantage to be computationally efficient, invariant to scaling, and to handle partial occlusions [14]. Also, to fit the application needs related to the interaction with the drone, the designed solution has to be

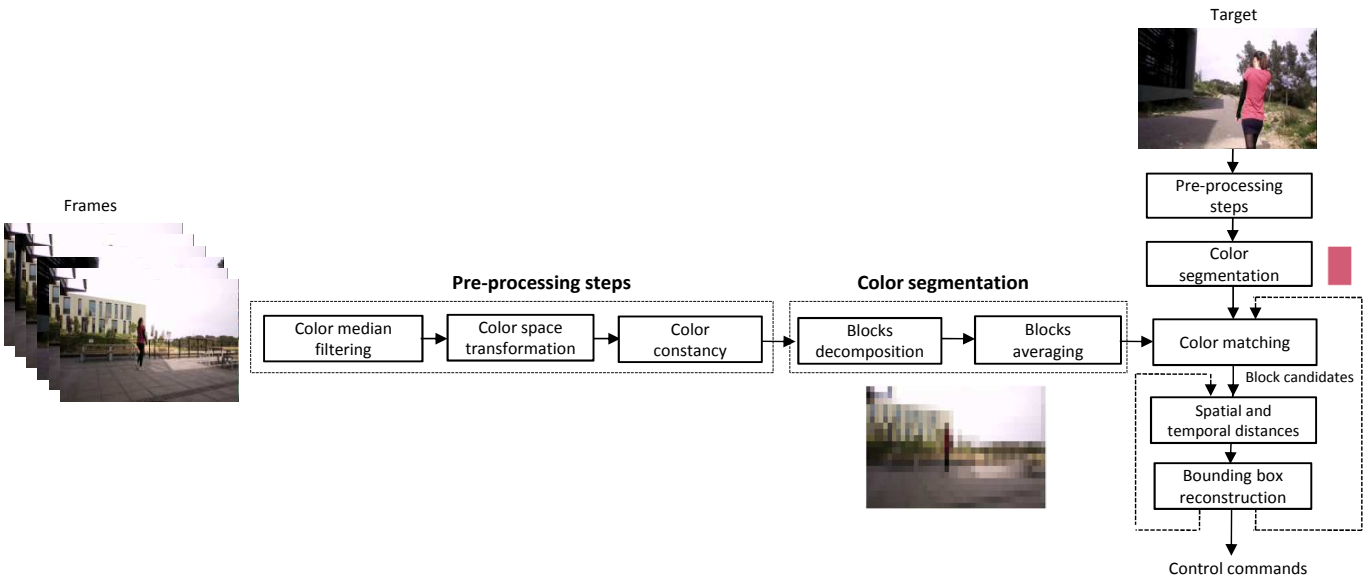


Fig. 1. Flowchart of the proposed color-based detection approach

fast enough in such a way that the drone can follow the target in real-time.

Toward the goal of being both robust enough to get accurate detections, and fast enough to be applied in real-time, person detection is performed through an appearance model using color information and some statistical measurements. This representation is basically inspired from [15] with some modifications and additional processing steps to fit the aforementioned application needs. The refered work [15] consists of identifying hooligans in a stadium based on the colors of the corresponding teams. The extracted color features have shown good identification results. These features are applied in our framework, where a target that appears in the field-of-view of the drone camera has to be followed.

For more details, our proposed detection framework is composed of two major steps: color patterns segmentation, see section II-A1 and color matching, see section II-A2. Also, some pre-processing steps are included, section II-A3. Finally, some post-processing steps are performed to filter out erroneous detected candidates, section II-A4. A flowchart illustrating the detection approach is shown in Figure 1.

1) Color patterns segmentation: Once the target appears in the field-of-view of the drone camera and is introduced to our system, color patterns from the torso are segmented. Precisely, a quadtree decomposition is applied to the target in order to extract different color patterns. This technique consists of dividing the bounding box of the target into four equal-sized blocks. Each block is recursively divided into 2 by 2 sub-blocks until it reaches a homogeneity criterion (which is the local variance). When the recursive decomposition process is finished, each resulting sub-block is replaced by its mean color. This averaging step speeds up the matching and reduces false detections due to some isolated pixels. Once the quadtree decomposition is performed, the different color patterns present in the upper part of the target are extracted.

2) Color matching: The following step consists of matching extracted color patterns in every next frame. To speed up the process, the quadtree decomposition is not applied in the next frames as proposed in [15]. Instead, each frame received by the drone camera is subdivided with a fixed grid into small blocks of N by N pixels; for each block the average color is computed. Then each color pattern extracted from the target is compared to the small blocks of the frame using euclidean distance. As a result of this matching process, a set of block candidates $\mathcal{B}_k = \{b_1^k, ..., b_{n_k}^k\}$ is obtained at a frame k. b_j^k denotes the j^{th} block candidate at this frame and is defined as $b_j^k = \{x_j^k, y_j^k, w_j^k, h_j^k\}$, where (x_j^k, y_j^k) is the upper left position of the block and w_j^k, h_j^k are the respective width and height. The blocks of the query colors are further filtered and recombined to build the final bounding box of the detected torso, see Section II-A4.

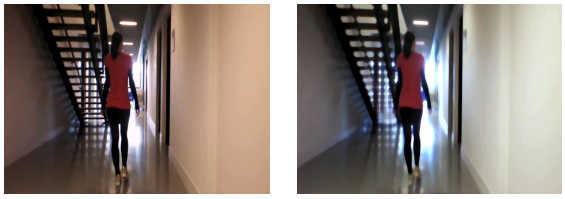
3) Pre-processing steps: Color segmentation and matching depict the key steps of the proposed detection framework. However, some additional enhancements are incorporated as pre-processing steps to deal with color changes due to the light variations in the scene and appearance variability of the target:

- Color median filtering: it is performed to smooth images by removing noise and flattening colors. It is important to improve the image quality before any further processing.
- Color space transformation: we propose to transform RGB to CIELab color space. CIELab has the particularity to be one of the color spaces where the distance between two colors is proportional to the difference that human eye can perceive [15]. This property is tremendously useful for color detection and comparison.
- Luminance component discarding: using CIELab color space enables separating the luminance component from the chrominance. This step is important since the color

must be recognized whether it is dark or bright.

- Color constancy: Grey World [16] is a color constancy algorithm which aims at estimating the ambient light in order to remove its effect on the colors in the scene. This could be simply done by computing the average color of the whole image since, statistically, the object colors would cancel each other. Then, colors at each frame are normalized using the estimated light color.

In Figure 2, the effects of median filtering and Grey World are shown on an exemplary frame where walls that are originally white appear slightly pink/red.



(a) Original image (b) Resulting image

Fig. 2. Effects of median filtering and Grey World using an exemplary frame

4) Post-processing steps: After applying the proposed detection framework, a set of block candidates which correspond to the extracted color patterns from the target is obtained at each frame. Our objective at this stage is to emphasize appropriate detections and to filter out erroneous ones. This is achieved by two filtering steps that enforce spatial and temporal constraints into the detected blocks. In a first step, detected block candidates with large displacements regarding the previous detected bounding box are removed. Since the motion of the target has to be compensated by the drone movement, the target is almost centered in the frame. Hence, large displacements in the target position might be a wrong detection. The second filtering step is based on the spatial distribution of detections by discarding blocks for which the spatial adjacency of colors is not respected. These two filtering steps add spatial and temporal coherence to the block candidates. After applying these filters, a new set of blocks $\mathcal{B'}_k = \{b_1^k, ..., b_{m_k}^k\}$ is obtained with $m_k \leq n_k$. The filtered blocks are consequently recombined to build the final bounding box of the target that fits a rectangular shape and is defined as:

$$\mathcal{D}_k = \{X_k, Y_k, W_k, H_k\} \tag{1}$$

where (X_k, Y_k) denotes the center position of the resulting bounding box:

$$X_k = \left(\min\left\{x_j^k\right\}_{1\leq j\leq m_k} + \max\left\{x_j^k\right\}_{1\leq j\leq m_k}\right)/2$$

$$Y_k = \left(\min\left\{y_j^k\right\}_{1\leq j\leq m_k} + \max\left\{y_j^k\right\}_{1\leq j\leq m_k}\right)/2$$

And W_k, H_k are the respective width and height defined by:

$$W_k = \max\left\{x_j^k\right\}_{1\leq j\leq m_k} - \min\left\{x_j^k\right\}_{1\leq j\leq m_k}$$

$$H_k = \max\left\{y_j^k\right\}_{1\leq j\leq m_k} - \min\left\{y_j^k\right\}_{1\leq j\leq m_k}$$

Furthermore, since the target appearance can be changed over time, this observation has to be accordingly updated. For this goal, we integrate learning capabilities to the proposed approach; each color pattern extracted from the target has to be updated according to the current detection using a learning rate α. More formally, each color pattern c_l at frame k is defined as:

$$c_l(k) = (1-\alpha)\ c_l(k-1) + \alpha\ c_l(k) \tag{2}$$

B. Control commands

Once the target is detected, the quadcopter is controlled through a set of commands to follow the person. vb Before detailing the tracking process, it is important to highlight that the drone used for tests is the Parrot AR.Drone 2.0 [17], whose standard quadcopter design enables flight with four degrees of freedom: vertical motion and Yaw rotation are independent, while forward/backward and sideways motions are performed via Pitch and Roll rotations, respectively, see Figure 3.

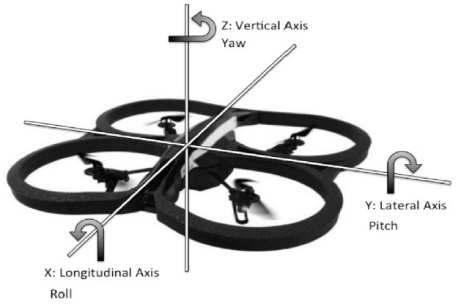


Fig. 3. Control commands of AR.Drone by setting Roll, Yaw, and Pitch

The tracking process depends on two criteria: the position (X_k, Y_k) of the detected bounding box $\mathcal{D}_k$ in the frame and its size $(H_k * W_k)$. For the first criterion, the distance between the center of the frame and the center of the detected bounding box $\mathcal{D}_k$ is considered to determine the corresponding movements. These distances are noted by Δ_x and Δ_y along the horizontal and vertical axes, respectively.

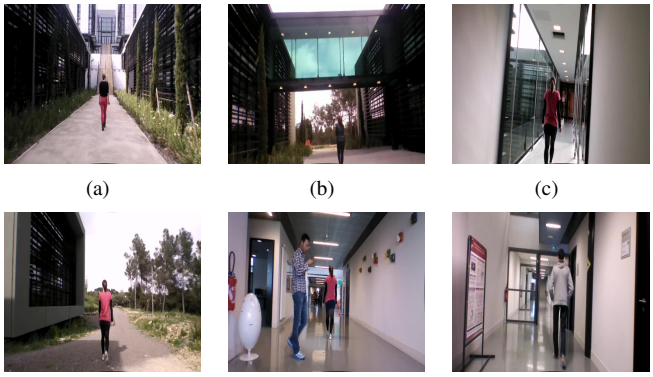
Precisely, the horizontal distance Δ_x is employed to control the sideways movement (Roll), while the vertical distance Δ_y is used to control the altitude. For the second criterion; the size of $\mathcal{D}_k$ regarding to the frame size is used to control the forward and backward movements (Pitch) based on the fact that the detected bounding box appears bigger when the person is closer.

Figure 4 illustrates some results of drone commands showing a real-time stream from the drone camera with the detected bounding box and the corresponding commands.

For all above strategies, some thresholds are empirically defined. If the target slightly moves the drone remains static, whereas if it moves more or less faster the drone moves with different speeds, well adapted to the person movements. An exception is noted for the backward speed which is low to avoid crashes, since the drone is not equipped with any camera on the back. For turning, a particular command strategy is

(a) horizontal displacement (b) forward movement

Fig. 4. Two sample frames showing the detected bounding box and the corresponding drone command



(a) (b) (c) (d) (e) (f)

Fig. 5. Sample frames of the recorded videos showing different scenes and scenarios (indoor/outdoor, occlusions, illumination changes, stairs, etc.)

introduced. With the strategies described above only, the drone would move forward and rotate at the same time. The problem is that the forward force is always applied to the tangent direction and thus the drone could drift out of its trajectory. To solve this problem, the forward speed is reduced while turning and a very small speed along the y-axis (Roll) is used.

III. Experimental Results

A. Drone Setup

The UAV used in the experiments is the Parrot AR.Drone 2.0 [17], a quadcopter, with the following specifications:

- Frontal camera: 720p at 30 fps
- Bottom camera: QVGA at 60 fps
- 1GHz 32 bit ARM Cortex A8 on board processor
- Linux OS based on kernel 2.6.32
- Wi-Fi b/g/n
- 3 axis gyroscope
- 3 axis accelerometer
- 3 axis magnetometer
- Pressure sensor
- Ultrasound sensors for ground altitude measurement

The application for autonomous navigation following a specific target is developed on GNU/Linux system. Precisely, the application is connected to the Wi-Fi hotspot created by the drone, receives the frames recorded by the drone cameras, processes them and sends back commands to the drone. It is important to mention that on-board processing is not constantly feasible since the computational power of the SOC is not sufficient. The communication with the drone has been done using ROS (Robot Operating System), which is a set tools developed for robot applications, and drone autonomy.

B. Experiments

The performance of the proposed autonomous detection and tracking framework is evaluated within challenging videos recorded by drones. The tested videos comprise different scenes and different scenarios as well. The recorded dataset is outstandingly interesting because it includes real-life scenarios of various conditions (indoor/outdoor, and different lighting conditions) and of various scenarios (person walking, running, turning, walking up stairs, etc.). We show some sample frames of the recorded videos in Figure 5.

To evaluate the proposed approach and to enable comparisons with other methods, we generate the ground truth by annotating the recorded videos using Viper [18]. Our proposed approach is compared to a state-of-the-art method that combines Histogram of Oriented Gradients (HOG) for detection and particle filter for tracking [19]. For quantitative evaluations, we employed ATA and ATE metrics, which are commonly used to assess single object tracking [20].

1) Average Tracking Accuracy:

$$ATA = 1/N_{frames} \sum_{t=1}^{N_{frames}} \frac{|G^t \cap D^t|}{|G^t \cup D^t|} \quad (3)$$

2) Average Tracking Error:

$$ATE = 1/N_{frames} \sum_{t=1}^{N_{frames}} \frac{|G^t \backslash D^t|}{|D^t|} \quad (4)$$

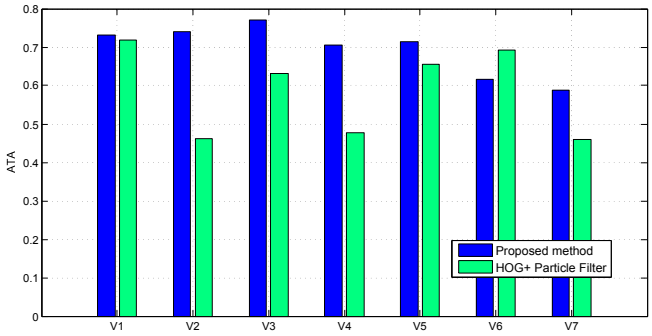
where G^t is the ground truth, D^t is the detected object in the frame t and N_{frames} is the total number of frames in the video. As defined, ATA computes the average ratio of the spatial intersection and the union of the ground truth and the tracked object over all frames. ATE computes the average tracking error. Both metrics vary from 0 to 1, but the closer ATA is to 1 the better is the result, whereas ATE is better when it is rather close to 0.

C. Results and Analysis

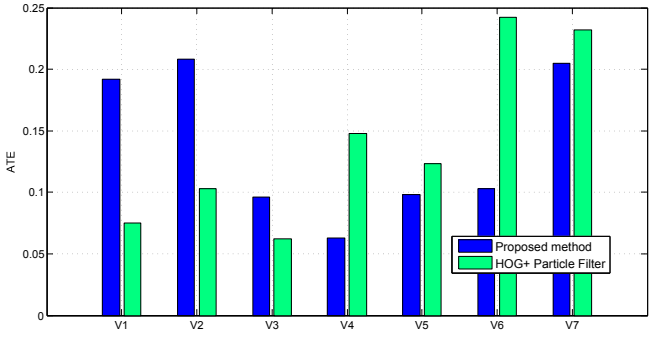
In Figure 6, the obtained results using our proposed detection and tracking approach on 7 recorded videos by drones are reported in terms of ATA and ATE. As shown, the proposed approach achieves accurate detection results; ATA reaches more than 0.7 for most of the videos. Slightly lower results are reported for the two last videos because of motion blur in the acquisition by the drone camera. In terms of ATE, the reported errors are often less than 0.2. These results demonstrate the relevance of the color based representation and the tracking process despite different challenges (such as illumination changes, different speeds, occlusions, indoor/outdoor, stairs, sharps turns, etc.). Also, in this figure, the obtained results are compared to the state-of-the-art method (HOG+Particle Filter)

[19] in terms of ATA and ATE. From this comparison, we notice that the proposed approach outperforms the compared method with a significant margin for most of the videos. It achieves 0.58 in terms of accuracy as the average result of the 7 videos compared to 0.69 achieved by our proposed approach. Using ATE, the compared method has an average error of 0.14 over all videos compared to 0.13 for our proposed approach.

In addition, it is important to highlight that the proposed method is fast enough to be applied in real-time, performing 25 fps using an Intel Core i5-2500 CPU, 8 Go of RAM running PC, compared to HOG+Particle Filter which is much more time consuming (around 5 fps).



(a) Evaluation in terms of ATA



(b) Evaluation in terms of ATE

Fig. 6. Comparison of the proposed method to HOG+Particle filter method in terms of ATA and ATE on 7 videos recorded by a mini-drone.

IV. Conclusion

In this paper, we presented our proposed approach for autonomous detection and tracking using mini-drones. The detection step is performed through an appearance model based on colors which is representative enough to distinguish the target from the background and from other objects. Then, the mini-drone is controlled through a set of commands in order to follow the detected target. The experimental results highlight the relevance of the proposed color appearance representation and the efficiency of the tracking process. In addition, we demonstrate that the proposed approach is robust enough to perform well in different situations. Also, by means of comparison to a state-of-the-art method, our approach has been experimentally validated showing more accurate results in real-time applications. The promising achieved results prove that our approach can be employed and extended to more complex future applications. Better performance is excepted by using more powerful drones and by considering other features (e.g., texture) as well.

Acknowledgment

This work was supported by the European project EIT Digital DRONES112 and the French National project FUI COOPOL.

References

[1] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[2] G. S. Walia and R. Kapoor, "Recent advances on multicue object tracking: a survey," *Artif. Intell. Rev.*, vol. 46, no. 1, pp. 1–39, 2016.

[3] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823 – 3831, 2011.

[4] V. Eiselein, H. Fradi, I. Keller, T. Sikora, and J. Dugelay, "Enhancing human detection using crowd density measures and an adaptive correction filter," in *10th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS*, 2013, pp. 19–24.

[5] H. Fradi, V. Eiselein, J.-L. Dugelay, I. Keller, and T. Sikora, "Spatio-temporal crowd density model in a human detection and tracking framework," *Signal Processing: Image Communication*, vol. 31, no. C, pp. 100–111, Feb. 2015.

[6] W.-C. Hu, C.-H. Chen, T.-Y. Chen, D.-Y. Huang, and Z.-C. Wu, "Moving object detection and tracking from video captured by moving camera," *J. Vis. Comun. Image Represent.*, vol. 30, no. C, pp. 164–180, Jul. 2015.

[7] Y. Wu, X. He, and T. Q. Nguyen, "Moving object detection with a freely moving camera via background motion subtraction," *IEEE Trans. Cir. and Sys. for Video Technol.*, vol. 27, no. 2, pp. 236–248, Feb. 2017.

[8] W. Choi, C. Pantofaru, and S. Savarese, "A general framework for tracking multiple people from a moving camera," *Pattern Analysis and Machine Intelligence (PAMI)*, 2013.

[9] T. Tanzi, L. Apvrille, J. L. Dugelay, and Y. Roudier, "Uavs for humanitarian missions: Autonomy and reliability," in *IEEE Global Humanitarian Technology Conference*, Oct 2014, pp. 271–278.

[10] A. Mashood, A. Dirir, M. Hussein, H. Noura, and F. Awwad, "Quadrotor object tracking using real-time motion sensing," in *2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, Dec 2016, pp. 1–4.

[11] T. Wang, R. Qin, Y. Chen, H. Snoussi, and C. Choi, "A reinforcement learning approach for uav target searching and tracking," *Multimedia Tools and Applications*, Feb 2018.

[12] J. Jimenez Lugo and A. Zell, "Framework for autonomous on-board navigation with the ar.drone," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1, pp. 401–412, Jan 2014.

[13] A. Burkle, F. Segor, and M. Kollmann, "Towards autonomous micro uav swarms," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1, pp. 339–353, Jan 2011.

[14] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 4, pp. 58:1–58:48, Oct. 2013.

[15] A. D'angelo and J.-L. Dugelay, "Color based soft biometry for hooligans detection," in *ISCAS 2010, IEEE International Symposium on Circuits and Systems*, 05 2010.

[16] D. Hilbert, "Color constancy and the complexity of color," *Philosophical Topics*, vol. 33, no. 1, p. 141, Jul. 2005.

[17] "Parrot sa. ar.drone 2.0." ardrone2.parrot.com/ardrone-2/specifications, 2013.

[18] V. Mariano, J. Min, J.-H. Park, R. Kasturi, D. Mihalcik, D. Doermann, and T. Drayer, "Performance Evaluation of Object Detection Algorithms," in *ICPR*, 2002, pp. 965–969.

[19] S. Rosa, M. Paleari, P. Ariano, and B. Bona, "Object tracking with adaptive hog detector and adaptive rao-blackwellised particle filter," *Proc. SPIE*, vol. 8301, 2012.

[20] D. M. Chu and A. W. M. Smeulders, "Thirteen hard cases in visual tracking," in *IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, 2010.