



A Real-Time Method to Remotely Detect a Target Based on Color Average and Deviation

Henry Cruz¹(✉), Juan Meneses², and Gustavo Andrade-Miranda³

¹ Universidad de las Fuerzas Armadas-ESPE, Sangolquí, Ecuador
hocruz@espe.edu.ec

² Research Center on Software Technologies and Multimedia Systems for Sustainability (CITSEM), Technical University of Madrid, Ctra. Valencia, Km. 7, 28031 Madrid, Spain

³ Facultad de Ingeniería Industrial. Av. Las Aguas, Universidad de Guayaquil, Guayaquil, Ecuador

Abstract. This paper presents a new semiautomatic method to remotely segment a target in real-time. The aim is to obtain a fast distinction and detection based on RGB color space analysis. Firstly, a pixel of the desired target manually is selected and evaluated based on weighing different surrounding areas of interest (A_i). Later, statistical measures, identification of deviation parameters and the subsequent assignation of identifiers (ID's) that are obtained from the color information of each region A_i . The performance of the algorithm is evaluated based on segmentation quality and computation time. These tests have been performed using databases as well in real-time and accessed in remote way (distance from the control-site 8.828.12 km) to prove the robustness of the algorithm. The results revealed that the proposed method performs efficiently in tasks as; objects detection in forested areas with high density (jungle images), segmentation in images with few color contrasts, segmentation in cases of partial occlusions, images with low light conditions and crowded scenes. Lastly, the results show a considerable decrease of the processing time and a more accurate detection of a specific target in relation with other methods proposed in literature.

Keywords: Segmentation · Average color · Identifiers · Specific target
Time processing

1 Introduction

Several methods for detecting a specific or multiple target have been proposed in the literature [1]. Most of them are focused on appearance modelling, segmentation-based techniques, localization strategies and classification-based techniques. As a clarification, the segmentation is the process of dividing an image into regions by classifying the pixels according to common attributes. Ideally, all regions found have physical interpretations and correspond to the objects in a scene. The segmentation algorithms are used to detect, identify, recognize and track a single or multiple objects in a scene.

Therefore, its applications are countless in different areas such as, e.g. remote sensing, recognition, traffic regulation, smart cities, social networks, augmented reality, smart cars, smartphones, computer vision, robotics, automation, medicine, games, biometric recognition, video analysis, annotation and tagging, content based on image retrieval, photo manipulation, video analysis, annotation and tagging, content based image retrieval, photo manipulation, image based rendering, intelligent video surveillance, scene understanding, among others [2–5]. On the other hand, the classification-based techniques seek to fulfil the key task of categorizing an image with robustness in terms of accurate detection [6, 7]. The classification-based techniques segment each of the objects that compose the image based on a set of features that better describe each of them. Some of the common features used are: texture, shape, brightness, contour energy, and curvilinear continuity, among others. According to Richards and Jia [8], the methods for supervised classification are mostly used for quantitative analysis (especially in remote sensing) based on different pixel-characteristics classifications.

One of the most challenging tasks in the detection of specific targets is the ability to efficiently extract their features. The literature shows the need to control variables depending on object or environment variations such as texture, colour, shape, dimensions, lighting variations, viewpoint, scale, occlusion and clutter [9, 10]. Another complex problem is the cluttering given the fact that the objects that surround the targets are very similar, causing false positives detections. It is a difficult task that still must be improved [11]. In order to reduce cluttering caused by colour variations and to improve the specific target detection with a minimum segmentation processing time, the Average and Deviation Segmentation Method (ADSM) has been proposed.

The ADSM starts with a manual pixel values acquisition of a specific target. This selection serves two purposes: the first is to get information about the average colour of the object of interest and to establish the desired region of interest (ROI). The second is to generate identifiers (ID's) in each ROI to localize the specific target. The RGB values have been widely used in the literature to obtain an accurate detection of a specific objective [12]. Other methods found in the literature include; light intensity variation [13], texture information [14], super-pixels resolution [15], Haar-like features [16] and saliency [17].

ADSM evaluates different areas of interest, which is a common practice in remote sensing where the spatial variability and texture of the image are considered [18]. Normally, rather than considering only the spectral characteristics of a given pixel, a group of pixels is used as literature suggests [19]. Lastly, the final segmentation is performed considering different areas of interest and different morphological operations. These kinds of detection method try to simulate the behaviour of a photo interpreter, permitting the recognition of homogeneous areas based on their spectral and spatial properties.

The paper is organized as follows. Section 2 details the ADSM methodology for detects a specific target. Section 3 shows the results obtained based on the Caltech 101 dataset [11] and Ecuadorian rainforest images obtained from an Unmanned Aerial Vehicle system (UAV's) of the Ecuadorian Air Force Research and Development Center (CIDFAE). Section 4 presents the discussion, and Sect. 5 the conclusions and future works.

2 Materials and Methods

This section provides a detailed explanation of the methodology carried out to detect a specific target. The object detection is based on properly ordered steps, which are described below (see Fig. 1 as an illustration):

1. Image acquisition
2. Selection of target coordinate (p_i) and evaluation areas (A_i)
3. Average and Deviation Segmentation Method (ASDM)
 - a. Color average
 - b. Deviation estimation
4. Post processing (acquisition of ROI's and ID's characterization)
5. Labeling (ID's contrasting)
6. Detected object.

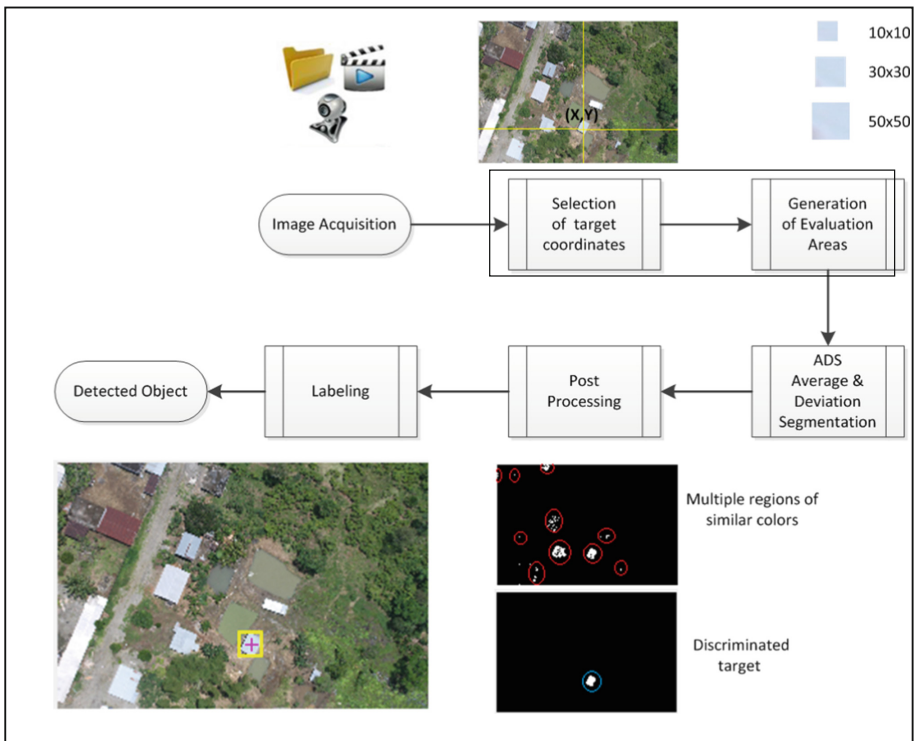


Fig. 1. Graphical representation of the steps followed to detect a target using ASDM.

2.1 Image Acquisition and Database

A total of 500 images are taken from the Caltech 101 [11] and CIDFAE database. The CIDFAE database has 1250 images of the Ecuadorian rainforest obtained from an Unmanned Aerial Vehicle system (UAV's) of the Ecuadorian Air Force Research and Development Centre (CIDFAE). The images resolutions are: RGB24 (320×240 pel), RGB24 (640×480 pel) and I240 (1280×1024 pel). This database is classified as confidential.

In order to evaluate the performance of the ADSM algorithm in the real-time over long distances and with bandwidth limitations, a remote experiment was done. The remote test has been performed between the CITSEM (Madrid-Spain $40^{\circ}23'22.6''N$ $3^{\circ}37'35.9''W$) as control site and CIDFAE (Ambato-Ecuador $1^{\circ}12'41.5''S$ $78^{\circ}34'28.1''W$) as terminal site. The approximate distance between both labs is around 8.828.12 km. The procedure was done using a P2P communication protocol, the resolutions of the images obtained remotely are: UYVY (352×210 pel), UYVY (352×480 pel) and UYVY (720×480 pel).

2.2 Selection of Target Coordinate (P_i) and Evaluation Areas

The ADSM is a semiautomatic method, where, is necessary to choose manually the pixel of interest (p_i) inside the specific target that we want to segment. The color information of the pixels that surround p_i is taken and three areas (A_i) of different size are defined as:

$$p_i \in A_1 \in A_2 \in A_3 \quad (1)$$

Where each area is included in the next larger one as it is shown in Fig. 2. The sizes of the areas were defined empirically based on the results obtained from different observation trials. The results that minimize the computational burden are obtained with areas of size 10×10 , 30×30 and 50×50 pixel. Therefore, the following relationships have been established:

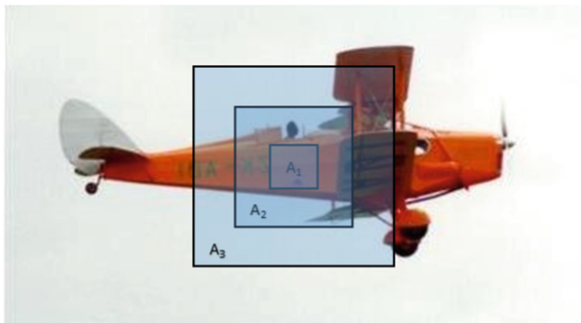


Fig. 2. Three areas of interest are defined around the pixel of interest p_i (center pixel of all areas).

$$\begin{aligned}
 & p_i(r, g, b) \in A_1, A_2, A_3, \\
 & \text{Where; } A_i = (A_i^r, A_i^g, A_i^b) \\
 & \text{Such that: } p_i^r \in A_i^r, p_i^g \in A_i^g, p_i^b \in A_i^b.
 \end{aligned}
 \tag{2}$$

The RGB components of the pixel p_i (r , g and b) are taken in each area, and they are weighted differently according to the size of the area.

2.3 Average and Deviation Segmentation Method (ADSM)

The segmentation by color average and deviation is divided in two steps. The first one computes the average value of the colors inside and around the target; and the second one computes the deviation of the average to find out the ROI. The complete procedure to detect and segment the targets is illustrated in Fig. 3.

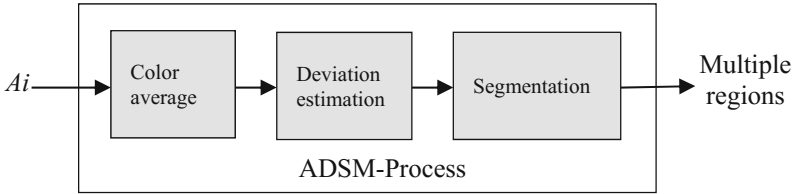


Fig. 3. Illustration of the ADSM algorithm.

2.3.1 Color Average

The segmentation process starts by computing the statistical average of the RGB values inside each of the three areas of interest A_i . To ensure the detection and discrimination of a specific target, the pixels belonging to A_i are weighted taking into account the distance with the target coordinate p_i . The lower the distance, the higher the weight w_i :

$$w_1 > w_2 \geq w_3 \tag{3}$$

w_1 , w_2 , and w_3 correspond to the weights of the A_1 , A_2 , and A_3 , respectively, in this case the weights was calculated based on several empirical tests. Considering the relation (3), the total weight value, Q_k , over all areas A_i is computed through the following expression:

$$Q_k = \sum_{i=1}^3 \sum_{j=1}^{n_j} p_j^k * w_i, \tag{4}$$

Where n_j is the number of pixels in each A_i , p_j^k is the pixel value of the RGB component k and w_i are the weights given to each area. From (4), the weighted average value, Q_{mk} ,

is computed as the division among Q_k and the weighted sum of the whole evaluated pixels:

$$Q_{mk} = \frac{Q_k}{\sum_{i=1}^3 A_i * w_i}, \tag{5}$$

Being, Q_{m1} the average value of the red component, Q_{m2} the average value of the green component and Q_{m3} the average value of the blue component. Lastly, those values are used as a reference to search similar targets around the images.

2.3.2 Deviation Parameters

It is not straightforward to find objects using only Q_{mk} information as the targets are not necessarily uniform. Therefore, an extended search area is computed; specifically a deviation parameter to widen the valid range around the average color that is used.

To effectively evaluate the pixels that correspond to a specific target, deviation limit is defined. The deviation limit allows obtaining valid colors based on Q_{mk} value. This procedure permits to determine similar objectives in relation to the chosen pixel value. Figure 4 illustrates an example in which all color combinations within the deviation limit are accepted meanwhile the outliers are not accepted.

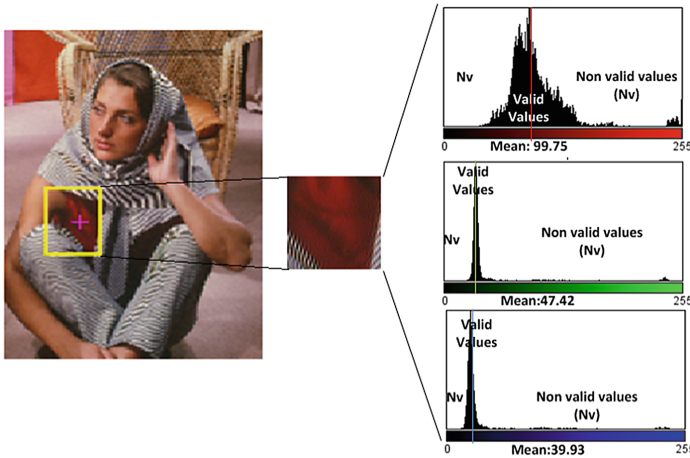


Fig. 4. Estimation of valid color ranges

To determine the deviation limits, three assumptions are made:

1. The maximum values of each color component are denoted as R_m, G_m, B_m and the most representative averages are named as R_{mx}, G_{mx}, B_{mx} , respectively.
2. The identification and detection of the target of interest depends on an area of maximum interest (Ai_m), the maximum value of each RGB component is extracted from the area of maximum interest and denoted as R_{max}, G_{max} and B_{max} .

3. The deviation restricts the detection error, therefore a tolerance interval (TI) is defined as $TI = \{a, b\}$:

$$a \leq D_{RGB} \leq b, \tag{6}$$

where D_{RGB} is the deviation for each RGB component. Considering the presented assumptions, the relation of difference values for $R_{max}, G_{max}, B_{max}$ and R_{mx}, G_{mx}, B_{mx} are computed as:

$$D_R = R_{max} - R_{mx}, \tag{7}$$

$$D_G = G_{max} - G_{mx}, \tag{8}$$

$$\text{and } D_B = B_{max} - B_{mx}. \tag{9}$$

D_R, D_G and D_B are the initial deviations for each RGB component. The obtained deviations are analyzed to obtain single reference value as $D_{Rmx} \sqrt{D_{Gmx}} \sqrt{D_{Bmx}}$, which lies inside TI such as:

$$\forall \{D_R, D_G, D_B\} \exists ! D_{Rmx} \sqrt{D_{Gmx}} \sqrt{D_{Bmx}} | a \geq D_{aRGB} \geq b \tag{10}$$

From (10) it can be inferred that:

$$D_{RGBf} < a; D_{RGBf} = a \wedge D_{RGBf} > b; D_{RGBf} = b. \tag{11}$$

D_{RGBf} is the final deviation that has to be applied to each RGB component to obtain valid values for carry out the segmentation. Lastly, each of the image pixels is analyzed considering the valid RGB range. Hence, every pixel with a color value inside this range is marked as a possible target pixel. The method works due that the pixel majority is classified in a threshold (deviation) where pixels are grouping with similar color range.

2.3.3 Post Processing (Acquisition of ROI's and ID's Characterization)

Within the post processing, different morphological operations are applied to keep the shape features of the target. Specifically, to distinguish the target region from the unwanted parts.

Firstly, dilation is used on the binary image obtained via segmentation. Later, the dilated image is subtracted from the former image to obtain the edges. In this way, a closed contour is obtained for each segmented object. Lastly, the detected contours are filled, and an erosion process is performed.

2.3.4 Labeling (ID's Contrasting)

Once all the objectives are obtained through segmentation, it is necessary to discriminate the specific target. For this reason, a special characterization through identifiers (ID's) and labeling has been developed.

Firstly, the coordinates of all detected regions are obtained and an identifier (ID) is assigned to each region generating a matrix of ID's. To discriminate the target region, the coordinates of the detected regions are compared with the coordinates of the initially selected pixel, and the ID of the coincident region is selected while the rest are discarded:

$$\forall \{ID_1, ID_2, ID_3, \dots, ID_n\} \exists ! O_i | ID_i \{x, y\} \cong IO_i \{x, y\} \tag{12}$$

where O_i is the object of interest, ID_n are the ID's of n regions, ID_i is the identifier of the region of interest and IO_i the identifier of the target region. From (12), the algorithm compares the matrix of ID's with the value of the IO_i , if it is a match with the value of the chosen pixel; it is chosen as the final identifier. Finally, a bounding box is used to better visualize the results, and the data of the location and center of mass is depicted. The complete process is illustrated in Fig. 5.

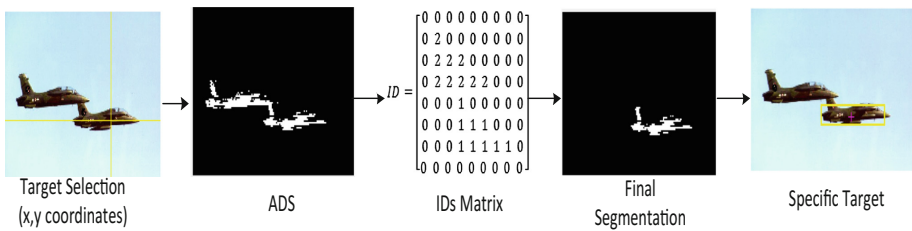


Fig. 5. Description of the characterization process through ID's.

2.4 ADSM Implementation

A graphical interface developed in Matlab R2012a and R2013b is implemented. Through the interface the detection and discrimination of a specific target can be visualized. This graphic interface additionally includes a classifier of successes and failures. Through this tool, it was possible to perform image assessments in the control site and terminal site. In the control site, a computer with an Intel (R) DualCore with 2.13 GHz processor, 2 GB RAM and 64 bit OS was used. For remote testing in CIDFAE, a computer with an Intel (R) Core (TM) i7 3.4 GHz processor, 7.49 GHz, RAM, 64 bit OS and electro-optical camera has been employed.

3 Experimental Results

Experimental tests have been performed within the research center CITSEM (Spain) and the CIDFAE (Ecuador). The experiments focused on proving two types of evidence: segmentation quality, and processing time.

3.1 Segmentation Quality

The present method focuses on better response times and more accurate detections; segmentation is designed to achieve both purposes. To verify the quality of the segmentations achieved with ADSM, the framework proposed by Martin [20], Creiver [21] and Wang et al. [22] have been used. The tests are based on the comparison between expert segmentation and automatic segmentation. In particular Global Consistency Error (GCE), Local Consistency Error (LCE) as well as Object Level Consistency Error (OCE) developed by Polak [23], are used. GCE and LCE are defined as:

$$GCE = \frac{1}{n} \min \left\{ \sum_i E(S_h, S_a, p_i), \sum_i E(S_a, S_h, p_i) \right\} \quad (13)$$

$$LCE = \frac{1}{n} \sum_i \min \{ E(S_h, S_a, p_i), E(S_a, S_h, p_i) \} \quad \forall GCE \leq LCE, \quad (14)$$

E represents the local refinement error, S_h is the expert image segmentation, S_a is the ADSM segmentation, p_i interest pixels relationship and n the number of pixels. The OCE is defined as follows:

$$OCE(S_h, S_a) = \min(E_{h,a}, E_{a,h}) \quad (15)$$

OCE defines an accuracy error, given the measured errors $E_{h,a}$ and $E_{a,h}$ based on penalization for over-segmentation and under-segmentation. Different trials are carried on between ADSM segmentation and expert segmentation using the database detailed in Sect. 2.1. As a matter of illustration, a sample of ten cases is presented and the results are depicted in Table 1.

Table 1. Results of the segmentation quality for LCE, GCE and OCE

Img.	1	2	3	4	5	6	7	8	9	10	Avg.
LCE	0.04	0.01	0.01	0.01	0.06	0.01	0.04	0.01	0.02	0.03	0.03
GCE	0.04	0.01	0.01	0.01	0.06	0.01	0.04	0.01	0.02	0.03	0.03
OCE	0.08	0.02	0.06	0.04	0.07	0.05	0.06	0.02	0.03	0.05	0.05

The results show an average value of 0.03 with LCE and GCE and 0.05 with OCE. Figure 6 illustrates some examples of the final segmentations obtained using ADSM.

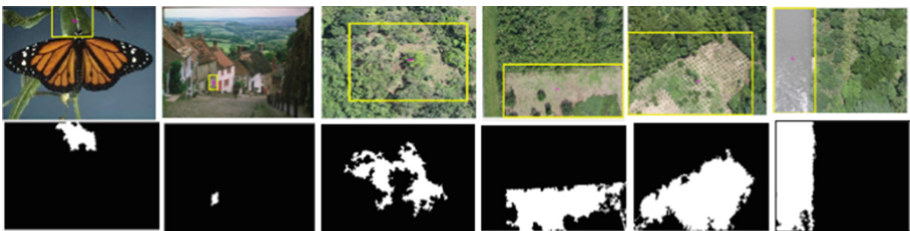


Fig. 6. Illustration of six images with its respective ADSM segmentation.

Additionally, the segmentation performance of ADSM is compared with the Region Growing based technique [24] and Chan-Vese algorithm [25] using the GCE as a metric. These methods were considered because show high performance for segmentation of the ROI's [12] [24, 25]. The results are shown in Table 2.

Table 2. GCE results of three different segmentation methods

Method	Average GCE
Region growing	0.0522
Chan-Vese	0.0167
ADSM	0.030

ADSM performs well (0.03 GCE average) in comparison with the other techniques. However, the aim of ADSM is to ensure fast target detection rather than a perfect segmentation.

3.2 Computational Cost

The computational cost of ADSM, Region Growing and Chan-Vese is computed for the whole database. The average values of the execution time (te), as well as the execution time plus processing time (tp) of each method, are compared. Te and tp are computed by the following relations (16) and (17):

$$te_T = \frac{1}{n} \sum_{i=1}^n te_i \tag{16}$$

$$tp_T = \frac{1}{n} \sum_{i=1}^n te_i + tp_i. \tag{17}$$

Where, te_i is the average execution time and represents the total time spent in the segmentation, n is the number of tests performed, and tp_T is the average execution time plus processing time. tp_T includes the time for labeling (made through ID's) and target discrimination until the target object is identified. Table 3 shows the tp values obtained with the electro-optical camera through remote access (in special external environmental conditions).

Table 3. Time processing average of 500 images (s) using electro-optical camera

Image size	ADSM	Chan-Vese	Region growing
352 × 210	1.2083	2.3276	2.0647
352 × 480	1.5215	3.2769	2.9549
720 × 480	2.5022	5.8171	4.9409

Comparing the times in Table 3, it can be observed that the computational time of ADSM is the lowest. In Table 4, the percentages of processing time reduction, ADSM in comparison to region growing and Chan-Vese also are shown.

Table 4. Percentages of processing time reduction, ADSM in comparison with region growing and Chan-Vese

Method	ADSM with Electro-Optical camera		
	352×210	352×480	720×480
Chan-Vese	48.09%	53.57%	56.99%
Region growing	41.48%	46.84%	49.36%
Average	44.79%	50.01%	53.18%

4 Discussions

An essential part of this work is the efficiency and effectiveness of detections in areas of dense vegetation, especially images of the Ecuadorian jungle, precisely other methods describe the difficult to get precision in these environments [26, 27]. The algorithm developed through ADSM, detect and discriminate a specific target despite the difficulty to differentiate objectives with similar color.

In relation to the segmentation quality test, ADSM performs well in general but it is overcome by the Chan-Vese method. ADSM is perfect for applications as surveillance, reconnaissance as well as remote sensing applications, where the response time and precise detection is primary that a perfect segmentation.

The results obtained remotely via electro-optical camera (Tables 3 and 4) show an overall reduction in tp when ADSM is used. This reduction is proportional to the resolution of the image and the speed of data transmission for remote access. ADSM presents a significant improvement in the tp parameter when is compared against region growing and Chan-Vese method. The worst tp are the ones obtained with Chan-Vese followed by region growing method, 5.81 seg, 4.94 seg, respectively.

5 Conclusions and Future Works

ADSM has been developed to obtain specific target detection and discrimination within multiple objects of similar colors. This method is based on the proposal to combine a calculation of an average color value together with deviation parameters that generate a novel method for real-time detection. ADSM has also contributed to solve some problems related with false positives in the detection of specific targets, especially in scenarios with color saturation such as jungle or forest. The proposal allows an efficient detection and discrimination of specific targets with low computational costs. ADSM do not use other attributes such as texture or morphology however show high performance.

The remote access tests have enabled to execute the graphical interface based on the developed method and the implemented algorithm, checking the robustness of the

application in real environmental conditions. A natural extension of this work is to develop a specific target tracking within multiple objects through ADSM, this must ensure an accurate tracking into real environmental conditions and ensure low processing rates.

Acknowledgments. Henry Cruz Carrillo gives thanks the Technological Scientific Research Center of the Ecuadorian Army (CICTE) for the collaboration obtained.

This work was sponsored by Spanish National Plan for Scientific and Technical Research and Innovation, project number TEC2013-48453-C2-2-R.

Henry Cruz Carrillo gives thanks Ecuadorian Air Force Research and Development Center (CIDFAE) for the collaboration obtained.

References

1. Xin, Z., Yee-Hong, Y., Zhiguang, H., Hui, W., Chao, G.: Object class detection: a survey. *J. ACM Comput. Surv. (CSUR)* **46**(1), 101–151 (2013)
2. Tsai, M.K.: Automatically determining accidental falls in field surveying: a case study of integrating accelerometer determination and image recognition. *Safety Sci. J.* **66**, 19–26 (2014)
3. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. *J. Robot. Auton. Syst.* **61**(12), 1258–1276 (2013)
4. Cabrera, R., Tuytelaars, T.: Boosting masked dominant orientation templates for efficient object detection. *Computer Vis. Image Und. J.* **120**, 103–116 (2014)
5. Andrade-Miranda, G., Godino-Llorente, J.I.: Glottal gap tracking by a continuous background modeling using inpainting. *Med. Biol. Eng. Comput.* **55**, 2123–2141 (2017)
6. Dong, J., Xia, W., Chen, Q., et al.: Subcategory-aware object classification. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 827–834 (2013)
7. Chia, A., et al.: Structural descriptors for category level object detection. *IEEE Trans. Multimedia* **11**(8), 1407–1421 (2009)
8. Richards, J., Xiuping, J.: *Remote sensing Digital Image Analysis: An Introduction*, 4th Edition, Chap. 8, pp. 193–338. Springer, Heidelberg (2005). <https://doi.org/10.1007/3-540-29711-1>
9. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **88**(2), 303–338 (2010)
10. Maggio, E., Cavallaro, A.: *Video Tracking Theory and Practice*, 3rd edn, pp. 3–120. Wiley, Hoboken (2011)
11. Lei, F., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. *Computer Vis. Image Underst.* **106**(1), 59–70 (2007)
12. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color based in probabilistic tracking. In: *Proceedings of 7th European Conference Computer Vision*, pp. 661–675 (2002)
13. Ross, D., Lim, J., Lin, R., Yang, M.: Incremental learning for robust visual tracking. *Int. J. Comput. Vis.* **77**, 125–141 (2008)
14. Avidan, S.: Ensemble tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(2), 261–271 (2007)
15. Wang, S., Lu, H., Yang, F., Yang, M.: Superpixel tracking. In: *Proceedings of IEEE International Conference Computer Vision*, 1323–1330 (2011)

16. Grabner, H., Bischof, H.: On-line boosting and vision. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 260–267 (2006). <https://doi.org/10.1109/cvpr.2006.215>
17. Lui, Z., Zou, W., Le Meur, O.: Saliency tree: a novel saliency detection framework. *IEEE Trans. Image Process.* **23**(5), 1932–1952 (2014)
18. Seo, Y., Lee, K.: Category classification of multispectral image data using spatial information in the small image region. *IEEE Geosci. Remote Sens. Symp.* **4**, 1978–1980 (1993)
19. Jianghong, S., Zhongming, Z., Qingye, Z., Yanfeng, W.: An algorithm for eliminating the isolated regions based on connected area in image classification. *IEEE Geosci. Remote Sens. Symp.* **5**, 3058–3061 (2004)
20. Martin, D., Fowlkes, C., Tal, D., et al.: A database of human segmented natural images and its applications to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of International Conference Computer Vision, Vancouver, pp. 416–425 (2001)
21. Crevier, D.: Image segmentation algorithm development using ground truth image data sets. *Comput. Vis. Image Understand.* **112**(2), 143–159 (2008). <https://doi.org/10.1016/j.cviu.2008.02.002>
22. Wang, M., Li, R.: Segmentation of high spatial resolution remote sensing imagery based on hard-boundary constraint and two-stage merging. *IEEE Trans. Geosci. Remote Sens.* **52**(9), 5712–5725 (2014)
23. Polak, M., Zhang, H., Pi, M.: An evaluation metric for image segmentation of multiple objects. *J. Image Vis. Comput.* **27**(8), 1223–1227 (2009)
24. Herwitz, S.R., et al.: Imaging from an unmanned aerial vehicle: Agricultural surveillance and decision support. *Comput. Electron. Agricult.* **44**(1), 49–61 (2004)
25. Chan, T.F., Vese, L.A.: Active contours without edges. *IEEE Trans. Image Process.* **10**(2), 266–277 (2001)
26. Cruz, H., Eckert, M., Meneses, J., Martínez, J.F.: Precise real-time detection of nonforested areas with UAVs. *IEEE Trans. Geosci. Remote Sens.* **55**(2), 632–644 (2017)
27. Cruz, H., Eckert, M., Meneses, J., Martínez, J.F.: Efficient forest fire detection index for application in unmanned aerial systems (UASs). *Sensors* **16**(6), 893, 1–15 (2016). <https://doi.org/10.3390/s16060893>