A Crowdsourcing Approach to Support Video Annotation

R. Di Salvo
Dep. Electrical, Electronics and Computer Engineering
University of Catania, Italy
roberto.disalvo@dieei.unict.it

D. Giordano
Dep. Electrical, Electronics and Computer Engineering
University of Catania, Italy
dgiordan@dieei.unict.it

I. Kavasidis
Dep. Electrical, Electronics and Computer Engineering
University of Catania, Italy
kavasidis@dieei.unict.it

ABSTRACT

In this paper we present an innovative approach to support efficient large scale video annotation by exploiting the crowdsourcing. In particular, we collect big noisy annotations by an on-line Flash game which aims at taking photos of objects appearing through the game levels. The data gathered (suitably processed) from the game is then used to drive image segmentation approaches, namely the *Region Growing* and *Grab Cut*, which allow us to derive meaningful annotations. A comparison against hand-labeled ground truth data showed that the proposed approach constitutes a valid alternative to the existing video annotation approaches and allow a reliable and fast collection of large scale ground truth data for performance evaluation in computer vision.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces; I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis

Keywords

Ground truth generation, Online game, Image Segmentation, Seed positioning

1. INTRODUCTION

In many computer vision and image processing applications, the assessment of the performance of an algorithm is generally a standard procedure: the algorithm is tested against a known and accepted standard dataset and the obtained results are then compared to annotations corresponding to the best obtainable results (ground truth). Computer vision scientists usually dedicate a large part of their time and effort to generate annotations needed to evaluate their algorithms. For this reason, much research has been devoted to find methods for acquiring ground truth data more efficiently.

VIGTA '13 July 15 2013, St. Petersburg, Russia

When semi-automatic methods for image and video annotation are concerned, several techniques aiming at minimizing the user intervention in gathering annotations on image and video sequences [6] [7] have been developed. Nevertheless, object annotation remains a time consuming, tedious and error-prone task, which makes the exploration and invention of more efficient techniques necessary to achieve better results.
Given the large number of users on the Web, crowdsourcing approaches are gaining more and more attention among the computer vision researchers [12, 17]. These methods typically rely on users' motivation and quality control for creating reliable image and video annotations.

In line with these recent trends, in this paper we present a crowdsourcing method, which exploits an on line game, for video annotation. By using the data gathered from this game, large scale video annotations can be generated. In order to play, the users need to take photos of objects through the game's levels, providing an increasing dataset of annotations which are then used for algorithm evaluation. In particular, in this work we demonstrate that starting from big noisy annotations and using simple image segmentation techniques, it is possible to generate reliable ground truth for object detection and classification. In order to assess the accuracy of our approach, we evaluated two classic techniques for image segmentation which require an initial labeling that may be either a point within the object (seed) or some region (or line) outside the object to be segmented. Needless to say, this initial labeling is the single most important parameter that influences the performance of image segmentation algorithms: if the initial labels are not positioned accurately either the result will contain undesirable information (a segment that contains the object and part of its surroundings) or it will omit desirable information (a partial result).

The main contributions of this work are:

- to show that reliable video annotations can be derived by using low quality and noisy data gathered quickly and easily;
- to show that the quality of such annotations increases as more users play with the game making it an effective and valid crowdsourcing application for the collection of ground truth data.

The remainder of this paper is as follows: Section 2 discusses about existing annotation tools and methods, Section 3 describes our approach in detail, while Section 4 discusses the performance obtained by comparing the results with a hand-labeled ground truth. Finally, in the last section, conclusions

are drawn and future works are given.

2. RELATED WORKS

Annotating videos is a tedious task and much effort has been made in order to devise more efficient methods. The ViPER-GT [3] tool is a baseline application for gathering ground truth data. It is stripped of any intelligent method for assisting the annotation task but it has been established in the scientific community not only because of its stability and of its standard file format, but mainly because of the lack of alternatives. A similar application is the GTTool [6] which enriches the annotation process with assisting tools such as automatic object detection, tracking and segmentation algorithms. The main limitation of the aforementioned applications is that they are stand-alone applications which do not provide any practical means to integrate different datasets and share them with other researchers. This encouraged scientists to search for other models for obtaining large scale ground truth.

LabelMe [11] is an on-line multi-user environment for image annotation where the user is presented with still images and she has to annotate them manually by using simple drawing tools. Moreover, *LabelMe* offers annotation quality control by considering annotations that contain more points as more precise. However, the quality control technique is very simplistic, at best, and, additionally, it does not provide an efficient way to integrate annotations of the same object made by different users. Moreover, the lack of any assisting annotation tools does not make the annotation process any easier. Finally, *LabelMe* is designed specifically for still images and not videos, although a version for videos was also created [19] but with limited success.

Many of these shortcomings are overcome in *PerLa* [7], an on line web-based platform that features more tools in order to assist the user in the annotation process. In particular, it enables the user to apply contour extraction and image segmentation techniques to image sequences, speeding up the whole process (although the performance of these methods are still heavily influenced by the characteristics of the image). *PerLa* also features user quality assessment, annotation integration and sharing, but still, the majority of the work must be completed manually.

Moreover, most of the existing solutions rely on self motivated persons, and these are usually the researchers themselves. In order to provide further motivation, crowdsourcing methods that award the users with money on a per-annotation basis have been proposed. Examples of crowdsourcing platforms are Amazon's Mechanical Turk [12] and CrowdFlower[1] where the users are paid for their annotations, but yielding, often, poor results [2, 18].

Another alternative for users' motivation is personal amusement [15]. For example, the ESP [14], Peekaboom [16] and KissKissBan [5] games try to exploit players' agreement (making two players guess each other's labels) to gather ground truth. While fun, these games aim at producing high level labels which describe the contents of the image, providing no means to acquire lower level data (e.g. object contours). Moreover, these games do not offer any means of quality control and the annotation integration mechanisms adopted are rather primitive.

Unlike the methods described above, we propose a simple game-based approach for reliably annotating videos, that does not require any specific knowledge from the users in order to use it.

[1]http://crowdflower.com/

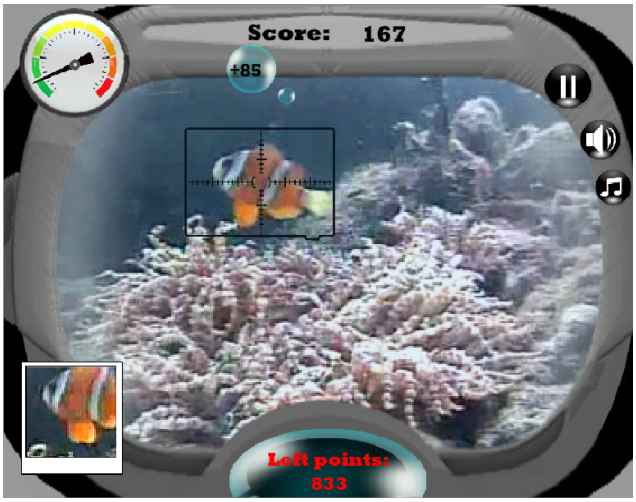

Figure 1: *Flash the Fish*. On the top left, the time remaining before the level ends is shown. On top, the achieved score and on the top right button the controls to pause the game, mute the music and the sound effects, respectively, can be found. On bottom left, the last taken photo is shown and on the bottom the points needed to advance to the next level are shown. Finally, the central area shows the video and the camera's shutter, which is centered on the mouse's pointer.

3. VIDEO ANNOTATION BY USING CROWDSOURCED DATA

In this section a brief description of the game *Flash the Fish*, used to gather the annotations, a discussion on how the produced data are processed to determine the initial labels and the procedure to build up the video annotations are given.

3.1 Flash the Fish

Flash the Fish[2] [8] is an on line game that enables an easy and fast acquisition of massive annotations on videos in the form of points.

The purpose of the game is to take photos of fish in underwater video segments, by clicking on them (Fig. 1) gaining as many points as possible. The user needs to reach a minimum score to advance to the successive game levels. Each click of the user contributes in estimating the presence of moving objects (in our case, fish) at the corresponding point in the video.

In order to complete the game, the user must pass 7 different levels. Every time a game session starts, a list with 7 video segments selected randomly from our repository, that contains more than 600.000 10-minute underwater videos, is generated. To make the game more competitive the difficulty of each level increases progressively. The first level has an initial frame rate of 5 FPS and the time available

[2]http://f4k-db.ing.unict.it

Figure 2: A heatmap produced by the game. Each colored area corresponds to a cluster.

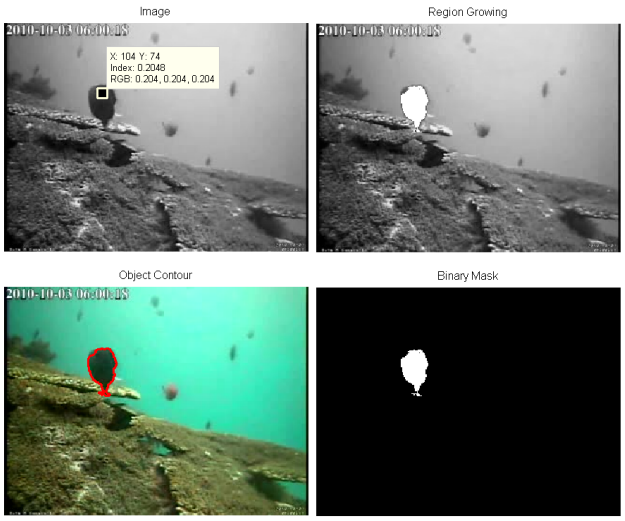

Figure 3: *Region Growing.* a)On the top left the seed derived from *Flash the Fish* game heatmaps. The remaining images show the result of the applied algorithm in terms of object contour and the corresponding binary mask.

for the user to complete it is 35 seconds. At each successive level the frame rate of the video segment is increased by one, while the time available is reduced by 2 seconds, to a maximum of 11 FPS and a minimum of 23 seconds at the seventh and last level.

The raw users' clicks do not hold any significant information because there exists no indication whether each one corresponds to an object in the video. For this reason, unsupervised K-Means cluster analysis [4] is performed on the players' clicks in order to extract the locations of the most clicked areas. Since the game's purpose relies on the belief that the most clicked areas represent actual objects, the resulting clusters will be devoid of the influence of noisy clicks, because clusters with low numbers of clicks are discarded.

The output of the game are the clusters with their associated points which can be also represented with heatmaps showing where the majority of the clicks are located. To generate a heatmap each point in a cluster is represented as a 3D Gaussian distribution. Summing all these distributions yields the colored shapes seen in Fig. 2. The characteristics of such maps are exploited in order to provide the input needed by the image segmentation algorithms.

3.2 Generating Object Annotations

Starting from the above clusters we resort to image segmentation approaches to generate annotations on moving objects. In detail, we used two approaches: the classic *region growing* that works by identifying the differences between objects in the image according to their color characteristics, and the *Grabcut* that performs image segmentation by means of a probabilistic approach.

3.2.1 Region Growing Based

Region growing [1] is a fairly common technique for image segmentation, which groups together the pixels or subregions in gradually larger regions according to a given criterion. The approach starts from a set of key points, also known as *seeds*, from which the regions grow. Afterwards, all the surrounding pixels that have similar properties to those of the starting seed are added to the region until a specific ending condition is satisfied.

Two issues must be addressed when dealing with the region growing approaches: a) the initial seed's position and b) the pixels' similarity policy. In our case, the initial seeds' position is determined by selecting the local maxima of the heatmaps as coming out from the previously calculated clusters.

Starting from these points, the region is iteratively grown by comparing all neighbouring pixels to the region by using the difference between a pixel's intensity value and the region's mean. Then, the pixel with the smallest measured difference in this way is allocated to the respective region. This process stops when the intensity difference between the region's mean and the new pixel becomes larger than a certain threshold.

Fig. 3 shows an example of segmentation by using the region growing technique described above. As we can see, starting from a single seed derived from the clustered users' clicks, good results are obtained in terms of object contour. The main drawbacks of the region growing segmentation algorithm are encountered a) when the background has a similar texture and color to the object of interest and b) when the seed is not positioned accurately inside the object.

Fig. 4 shows what might happen in the above cases: the area of the region grows beyond the object boundaries including parts of the image that are not logically connected to object or the object is not completely segmented, and misses desirable information.

In the next section the use of *Grab Cut* approach will be discuss which aim at overcoming the limitation of the just discussed method.

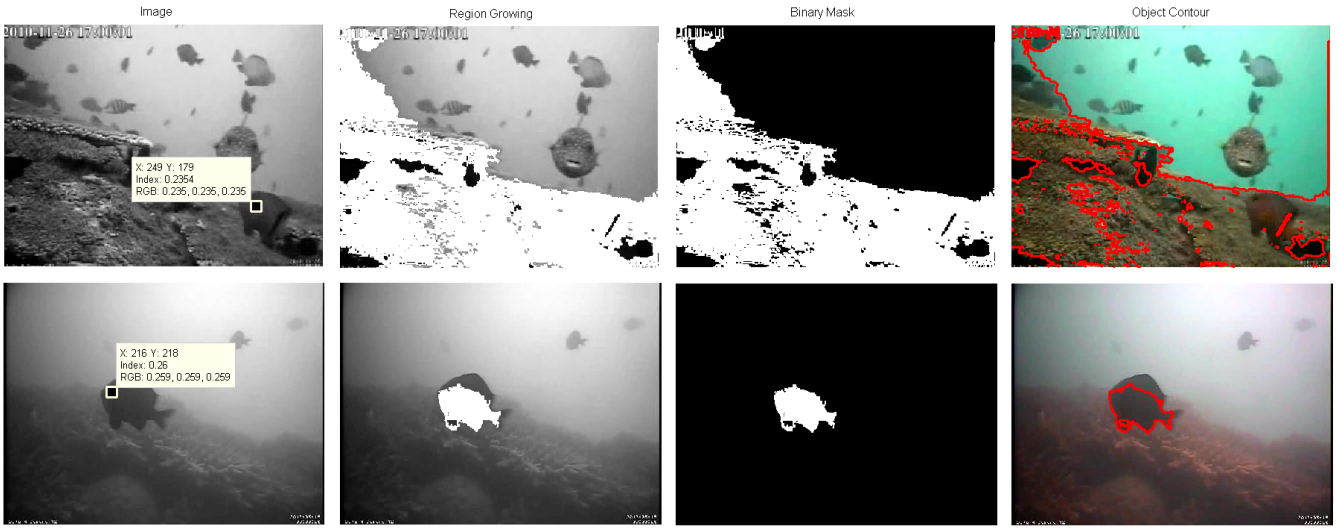

Figure 4: *Region Growing drawbacks. Top:* Background and object have similar colors and texture. *Bottom:* Inaccurate positioning of the seed.

3.2.2 Grab Cut

Grabcut[10] is a dynamic image segmentation algorithm that applies graph cuts [9] iteratively: each successive iteration aims at minimizing further the energy of the result of the previous ones. In contrast to the region growing algorithm, *Grabcut* operates in a different manner: instead of using a single pixel of the image as seed for determining the part of it that belongs to the desired segment or not, *Grabcut* uses an area where the object should be located.

Describing the exact theory behind *Grabcut* is not in the scope of this paper, but the reader can find more information in [10]. What this work addresses, instead, is the definition of the initial labeling for *Grabcut* to start the segmentation process.

In detail, by using the game data, the initial labeling is derived by processing the players' clicks, in order to define a region large enough to contain the whole object, but also small enough in order not to include unnecessary information. In our case, this region is computed as the convex hull containing all the points belonging to the same cluster. A labeling mask is then created, where white points (*foreground*) are all the points inside the convex hull, and the black ones (*background*) are the points outside. This mask constitutes the initial labeling for *Grabcut*.

The application of the *Grabcut* algorithm is shown in Fig.5. The same figure also shows that *Grabcut* performs well even when both the background and the object have similar colour and texture characteristics.

Figure 5: Example of the application of the *Grabcut* segmentation algorithm. On the *top left* the original image can be seen, while in the *top right* the points that belong to an identified cluster are shown in yellow. The convex hull of these points is calculated (*bottom left*, in red) and *Grabcut* is applied on it *(bottom right)*.

4. PERFORMANCE EVALUATION

In order to evaluate the performance of both approaches we compared the obtained results against hand-drawn ground truth. This ground truth contained 4140 objects and it was generated with PerLa [7]. In order to motivate users to play with the game we organized a Facebook event and offered a prize for the player that would achieve the highest score. For the event's duration (4 days), more than 80 users participated in about 1300 game sessions providing more than 260.000 clicks.

Quantitative performance analysis was carried out on a pixel basis by comparing the results obtained by the segmentation algorithms against the hand-labeled ground truth and by computing the following metrics:

$$Precision = \frac{TP}{TP + FP}, \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Level	Annotated objects	Acquired clicks	Region Growing			Grabcut		
			Precision	Recall	F_1	Precision	Recall	F_1
1	722	71105	0.19	0.30	0.23	0.78	0.81	0.79
2	1847	70406	0.31	0.41	0.35	0.79	0.74	0.76
3	593	58528	0.44	0.47	0.45	0.63	0.68	0.65
4	251	47137	0.43	0.40	0.41	0.67	0.64	0.65
5	446	16276	0.51	0.41	0.45	0.39	0.42	0.40
6	104	522	0.63	0.61	0.62	0.29	0.28	0.28
7	177	342	0.65	0.67	0.66	0.22	0.24	0.23
Total	4140	264316	0.45	0.47	0.45	0.54	0.54	0.54

Table 1: Performance evaluation of the segmentation algorithms.

and

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \quad (3)$$

A TP (True Positive) was defined as a pixel that was contained both in the ground truth and in the segmentation result, while a FP (False Positive) was defined as a pixel contained in the resulting segment but not in the ground truth. Finally, a FN (False Negative) was a pixel that was contained in the ground truth but not in the resulting segmentation.

The obtained results, together with the dataset used for testing the application, are shown in Table 1. These results take into account only the annotations that had a PASCAL score of at least 0.7 with respect to an object in the ground truth. The PASCAL score is given by:

$$P_{score} = \frac{area(Annotation \cap GT)}{area(Annotation \cup GT)} \quad (4)$$

where *Annotation* is the resulting annotation and GT is the corresponding ground truth object. From the same figure, it is also possible to notice how the performance of the segmentation algorithms show a different behaviour. In particular, region growing performed better when the number of clicks available was low. In fact, in the lower levels (levels 1 and 2) the region growing based approach achieved, on average, 25% in precision and 36% in recall. The precision score was so low because of the large number of inaccurate clicks which, therefore, resulted in clusters whose centroids were outside the object's boundaries. On the contrary, in the highst levels (i.e. 6 and 7), where only few motivated and reliable users were able to get score, both precision and recall achieved, on avarage, 64%. In this case, even if the number of clicks was considerably lower than those of the first levels, they were extremely accurate as obtained by the best performing users.
Grabcut, instead, achieved better performance in the lower levels. In the first 4 levels, the precision and recall values were, on average, 71%. These scores reflect the much better capacity of *Grabcut* to handle complex backgrounds and its ability in the choice of the initial labeling.

Finally, the performance of Grabcut in the last two levels was very low because an accurate initial labeling could not be determined due to the lower number of clicks.

5. CONCLUSIONS

In this paper we presented a work that exploits crowd sourcing mechanisms in order to support the annotation of objects in a video by playing an online game. The obtained results (*F measure* of about 80% in the best case) when compared against a hand labeled ground truth dataset, showed that the proposed approach is able to generate reliable annotations providing a valid alternative to the existing groung truth generation methods.

While a more accurate version for segmenting objects in videos is underway, the data gathered by the game can also be used in order to derive other types of annotations. For example, object-tracking ground truth by grouping the user's clicks in the frame sequences and by applying object tracking methods, like [13], in order to find the trajectories' limits.

6. ACKNOWLEDGMENTS

We would like to thank Marco Rapisarda and Laura Pafumi for the implementation of the game. This research was funded by European Commission FP7 grant 257024, in the *Fish4Knowledge* project[3].

7. REFERENCES

[1] Rolf Adams and Leanne Bischof. Seeded region growing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(6):641–647, 1994.

[2] Michael Buhrmester, Tracy Kwang, and Samuel D Gosling. Amazon's mechanical turk a new source of inexpensive, yet high-quality, data? *Perspectives on Psychological Science*, 6(1):3–5, 2011.

[3] David Doermann and David Mihalcik. Tools and techniques for video performance evaluation. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 4, pages 167–170. IEEE, 2000.

[4] Richard Dubes and Anil K. Jain. Clustering techniques: The user's dilemma. *Pattern Recognition*, 8(4):247 – 260, 1976.

[5] Chien-Ju Ho, Tao-Hsuan Chang, Jong-Chuan Lee, Jane Yung-jen Hsu, and Kuan-Ta Chen. Kisskissban: a competitive human computation game for image annotation. In *Proceedings of the ACM SIGKDD*

[3] www.fish4knowledge.eu

Workshop on Human Computation, pages 11–14. ACM, 2009.

[6] I. Kavasidis, S. Palazzo, R. Di Salvo, D. Giordano, and C. Spampinato. A semi-automatic tool for detection and tracking ground truth generation in videos. In *Proceedings of the 1st International Workshop on Visual Interfaces for Ground Truth Collection in Computer Vision Applications*, VIGTA '12, pages 6:1–6:5, New York, NY, USA, 2012. ACM.

[7] Isaak Kavasidis, Simone Palazzo, Roberto Di Salvo, Daniela Giordano, and Concetto Spampinato. An innovative web-based collaborative platform for video annotation. *Multimedia Tools and Applications*, pages 1–20, 2013.

[8] Isaak Kavasidis, Concetto Spampinato, and Daniela Giordano. Generation of ground truth for object detection while playing an online game: Productive gaming or recreational working? *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013.

[9] P. Kohli and P. H S Torr. Efficiently solving dynamic markov random fields using graph cuts. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 922–929 Vol. 2, 2005.

[10] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[11] BryanC. Russell, Antonio Torralba, KevinP. Murphy, and WilliamT. Freeman. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008.

[12] A. Sorokin and D. Forsyth. Utility data annotation with amazon mechanical turk. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, pages 1–8, 2008.

[13] Concetto Spampinato, Simone Palazzo, Daniela Giordano, Isaak Kavasidis, Fang-Pang Lin, and Yun-Te Lin. Covariance based fish tracking in real-life underwater environment. In *VISAPP (2)*, pages 409–414, 2012.

[14] Luis Von Ahn. Games with a purpose. *Computer*, 39(6):92–94, 2006.

[15] Luis Von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326. ACM, 2004.

[16] Luis Von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 55–64. ACM, 2006.

[17] Carl Vondrick, Donald Patterson, and Deva Ramanan. Efficiently scaling up crowdsourced video annotation. *International Journal of Computer Vision*, 101(1):184–204, 2013.

[18] Carl Vondrick, Deva Ramanan, and Donald Patterson. Efficiently scaling up video annotation with crowdsourced marketplaces. In *Computer Vision–ECCV 2010*, pages 610–623. Springer, 2010.

[19] Jenny Yuen, Bryan C. Russell, Ce Liu, and Antonio Torralba. Labelme video: Building a video database with human annotations. In *ICCV'09*, pages 1451–1458, 2009.