Computer Vision and Image Understanding 131 (2015) 88–99

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

An interactive tool for manual, semi-automatic and automatic video annotation

Simone Bianco [1], Gianluigi Ciocca [1], Paolo Napoletano [*,1], Raimondo Schettini [1]

DISCo (Dipartimento di Informatica, Sistemistica e Comunicazione), Università degli Studi di Milano-Bicocca, Viale Sarca 336, 20126 Milano, Italy

ARTICLE INFO

Article history:
Received 2 December 2013
Accepted 4 June 2014

Keywords:
Interactive video annotation
Automatic annotation
Semi-automatic annotation
Incremental learning
Object detection

ABSTRACT

The annotation of image and video data of large datasets is a fundamental task in multimedia information retrieval and computer vision applications. The aim of annotation tools is to relieve the user from the burden of the *manual* annotation as much as possible. To achieve this ideal goal, many different functionalities are required in order to make the annotations process as automatic as possible. Motivated by the limitations of existing tools, we have developed the *i*VAT: an *interactive* Video Annotation Tool. It supports manual, semi-automatic and automatic annotations through the interaction of the user with various detection algorithms. To the best of our knowledge, it is the first tool that integrates several computer vision algorithms working in an *interactive* and *incremental learning* framework. This makes the tool flexible and suitable to be used in different application domains. A quantitative and qualitative evaluation of the proposed tool on a challenging case study domain is presented and discussed. Results demonstrate that the use of the semi-automatic, as well as the automatic, modality drastically reduces the human effort while preserving the quality of the annotations.

1. Introduction

In recent years several video annotation tools have been developed with a twofold aim of reducing the *human effort* necessary to generate ground truth of large scale visual datasets and improving the *annotations quality*. Most of the tools proposed in the literature include computer vision and machine learning methods that support humans to annotate more efficiently [1–8], while some promote the use of crowd-sourcing based platform to improve the quality of the annotations [9–11].

The different tools can be characterized depending on the functionalities they support. All the annotation tools allow the user to locate in a frame an object of interest by drawing a boundary around it. The most basic, and easily drawn, boundary shape is a rectangle but different tools support other shapes as well such as ellipses and polygons. The most advanced tools also allow the drawing of the boundaries with the aid of semi automatic algorithms such in the case of [4,10]. Although these boundaries should be, theoretically, drawn on every frame in the video sequence, it is often useful, in order to reduce the human efforts, to annotate only few frames (i.e. key frames) and then propagate the annotation by meaning of dedicated algorithms. Almost all the tools considered here incorporate a form of basic annotation propagation exploiting the visual coherence of neighbor frames. The most efficient (in terms of computation time) propagation strategy is based on a simple linear interpolation of the boundaries of an object between a starting position and ending one. More advanced strategies exploit tracking algorithms to explicitly locate instances of the same object across different frames (e.g. [4,5]). Tools that support the annotator with algorithms that accomplish these elementary computer vision tasks have demonstrated to be quite effective in terms of the number of user interactions, user experience, usability, accuracy and annotation time [9]. Since speed-up and simplify the annotation process is of paramount importance in these tools, they often include mechanisms to easily browse the video frames, shots, and make available different modes with which users can interact with the tool (i.e. a graphical user interface, short-cuts, mouse actions, etc.).

Notwithstanding these basic functionalities, the final objective of the annotation tools is to relieve the user from the burden of the *manual* annotation as much as possible. To achieve this ideal goal, computer vision methods are often included in existing tools to support automatic or semi-automatic video annotation. The most recent trend is the integration of algorithms that accomplish more complex computer vision tasks, such as supervised object

* Corresponding author.
E-mail addresses: bianco@disco.unimib.it (S. Bianco), ciocca@disco.unimib.it (G. Ciocca), napoletano@disco.unimib.it (P. Napoletano), schettini@disco.unimib.it (R. Schettini).

[1] The authors contributed equally to this work.

http://dx.doi.org/10.1016/j.cviu.2014.06.015
1077-3142/

detection, template matching, action recognition, event detection, and advanced object tracking [1,2,4,12]. In particular, the use of supervised object detection algorithms allows the automatic annotation the different objects of interest. However, one of the main drawbacks of these algorithms is that they are often domain specific, and must be heavily trained to have a robust detection. Template matching algorithms on the other hand, can be readily used to detect specific instances of the objects but may lack the robustness necessary to detect objects that often change their appearance. For all these reasons, a more efficient approach could be the integration of different annotation modalities to give the user a flexible tool able to work efficiently well across different application domains. At the same time the tools incorporating object detection algorithms, should provide a mechanism for expanding their knowledge of the domain (such as incremental learning algorithms), in order to iteratively increase their efficacy.

Table 1 summarizes and compares recent video annotation tools found in the literature. We have identified some properties that we consider very important in a video annotation tool. The properties refer to the tool's design, user interactions, and basic and advanced annotation functionalities. These properties are:

- Platform: the tool is a Web-based or Desktop application?
- Programming language: what programming languages have been used in the development of the tool?
- Cross-platform: can the tool be used on different platforms?
- Object's boundary shape: what kind of shapes can have a boundary?
- List of objects: the tool allows the annotation of a given list of objects?
- Object's attributes: other information are associated to an object's identity?
- Time-line of objects: does the tool support time-line visualization?
- Temporal reference: are the annotations temporally referenced in the visualization?
- Frames navigation: does the tool supports a navigation within frames?
- Shots navigation: does the tool supports the extraction and navigation of video shots?
- Range-based operations: does the tool support annotation operations on a range of frames/objects?
- Key Frames: the tool supports the annotation of key frames?
- Template matching: does the tool support semi-automatic annotation through the use of template matching algorithms?
- Annotation propagation: what kind of annotation propagation mechanism is included in the tool?
- Supervised object detection: does the tool support automatic annotation through supervised object detection algorithms?
- Incremental learning: does the tool support an incremental learning mechanism?
- Cooperative annotation: is cooperative/crowd sourcing annotation supported?
- Cross domain: can the tool be extended to work on different domains?
- Evaluation tool: does the tool includes an evaluation module to assess annotation quality?

As it can be seen, each tool possesses a set of important functionalities and properties but lacks others also important for the annotation task. For this reason, we developed *i*VAT, an interactive annotation tool that supports the user during the annotation of videos, and that integrates different computer vision modules for object detection and tracking. Among its main features there is the support of three different annotation modalities: manual, semi-automatic and automatic. It also integrates an incremental learning mechanism. To the best of our knowledge, it is the first tool that integrates several computer vision algorithms working in an *interactive* and *incremental learning* framework. This makes the tool flexible and suitable to be used in different application domains.

Previous versions of this tool have been presented in [7,13]. With respect to previous works the main contributions of this paper are:

- an in-depth state of the art analysis is presented and discussed;

Table 1
Existing video annotation tools comparison. With the bullet we indicate that the given tool owns the specific functionality, with the circle we indicate the opposite, while with the minus we indicate that information on that functionality is not provided. The half filled circle stands for not completely integrated functionalities (see Section 4.3).

	VATIC [9]	ViPER-GT [1]	FLOWBOOST [2]	LabeLME V [3]	GTTOOL [4]	GTVT [5]	GTTOOL-W [10]	Inter OD [6]	iVAT (current version)
Platform	Web based	Desktop	–	Web based	Desktop	Desktop	Web based	Desktop	Desktop
Programming language	Html/JS/ Python	Java	–	–	–	VS.NET C#	VS.NET C#	–	C/C++/Qt
Cross-platform	•	•	–	•	–	○	•	–	•
Object's boundary shape	Rect	Rect/Ellip/ Poly	Rect	Poly	Poly/Active Cont.	Rect	Poly/Active Cont.	Rect	Rect/Ellip/ Poly
List of objects	•	○	–	○	○	•	○	○	•
Object's attributes	•	•	○	•	•	○	•	○	•
Time-line of objects	○	•	○	○	○	○	○	○	•
Temporal reference	○	•	•	•	•	○	○	○	•
Frames navigation	•	•	–	•	•	○	•	•	•
Shots navigation	○	○	–	○	○	○	○	○	•
Range-based operations	○	•	–	○	○	○	○	○	•
Key Frames	•	•	•	•	○	•	○	•	•
Template matching	○	○	○	○	•	○	○	○	•
Annotation propagation	Lin. Interp.	Lin. Interp.	Time based reg.	Homogr. Pres.	Tracking	Tracking	○	○	Lin. Interp.
Supervised object detection	○	•	•	○	○	○	○	•	•
Incremental learning	○	○	•	○	○	○	○	•	•
Cooperative annotation	•	○	○	○	○	○	•	○	◐
Cross-domain	•	•	–	•	•	○	•	•	•
Evaluation tool	○	•	○	○	○	•	•	•	•