

Analysis of Videos using Tile Mining

Toon Calders, Elisa Fromont, Baptiste Jeudy, Hoang Thanh Lam

▶ **To cite this version:**

Toon Calders, Elisa Fromont, Baptiste Jeudy, Hoang Thanh Lam. Analysis of Videos using Tile Mining. Real-World Challenges for Data Stream Mining Workshop-Discussion, Sep 2013, Czech Republic. hal-00871874

HAL Id: hal-00871874
https://hal.science/hal-00871874

Submitted on 10 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Analysis of Videos using Tile Mining

Toon Calders[2], Elisa Fromont[1], Baptiste Jeudy[1], Hoang Thanh Lam[3]

[1] Université de Lyon, Université Jean Monnet de Saint-Etienne
Laboratoire Hubert Curien, UMR CNRS 5516, Saint-Etienne, France
[2] Université Libre de Bruxelles (ULB), Bruxelles, Belgium
[3] TU Eindhoven,Department of Maths and Computer Science
Eindhoven, Netherlands

Abstract. We investigate how mining top-k largest tiles in a data stream under the sliding window model can be useful for (real-time) analysis of videos and, in particular, for tracking. We first explain how a tracking problem can be cast into a stream pattern mining problem. We then show some preliminary results on tracking in the particular context where both the objects and the camera are moving and where the user does not specify the regions of interest in the first frames of the videos.

1 Introduction

Pattern mining techniques are more and more often used in computer vision [13, 8, 12] to obtain features that are more discriminative than those extracted using computer vision algorithms. This is true for example in content-based images/videos retrieval, indexing, classification, tracking, etc. However, the main drawback of using traditional pattern mining techniques is their inefficiency when dealing with huge set of data (for example provided by Google image or Youtube for videos) or when trying to tackle real-time analysis problems. The data mining community has been working on the "Big Data" problem for many years coming up with promising solutions such as stream mining [5, 11]. The aim of this paper is to explain how stream mining could be used for (real-time) analysis of videos and, in particular, for tracking.

Many ongoing research works concerning object tracking in videos [1, 9] make strong assumptions about the objects to track (people, car, etc.) which can be modelled in advance, or about the tracking context (stable background, object moving in a single direction, stable lighting conditions, etc.) to perform an efficient tracking. Being able to introduce an unsupervised method which would i) track objects in difficult conditions (moving cameras, multiple objects) and, ii) automatically detect the interesting objects to track (assuming that they are the main focus of the video) could be of great interest for the vision community.

2 Transforming a Video Into a Stream Mining Problem

The first problem when trying to cast a video analysis problem into a pattern mining problem is to find the relevant features to mine (items, sequences, graphs) and the method to construct them.

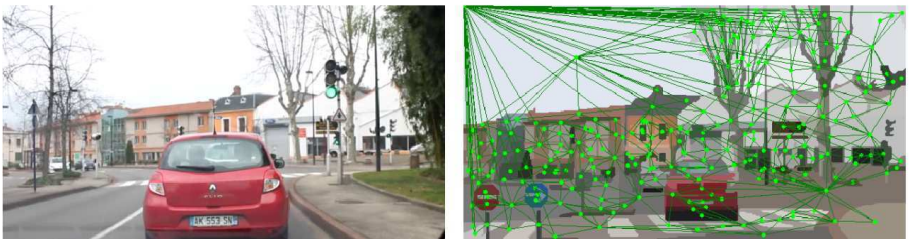


Fig. 1. Frame of a video and its corresponding RAG from a (static) segmentation.

2.1 Describing a Video

The first step is either to segment the frames and to consider only description of regions or, to find interest points in the frames and rely on geometrical structure to connect them [3, 2]. We will focus on the first solution. We propose two types of frame segmentations. The first segmentation (static) is done independently on each frame using the algorithm[4] presented in [7]. This algorithm has three parameters for which we use the default values. The second segmentation is the (dynamic) video segmentation algorithm[5] presented in [10]. This algorithm outputs regions that are identified through time, i.e, it provides a correspondence between regions in different frames.

2.2 Constraint-based Itemset Mining

We associate an item to each segmented region. The second type of segmentation seems more appropriate since the matching between regions of two consecutive frames ensures that a given item will be as stable as possible over time (of course, if the region completely disappears or changes too much, the item will also disappear). A transaction is thus the set of items that appear in a given frame. We assume that the object to track is frequent in the videos (i.e. that the video particularly focuses on it). An interesting problem to solve is thus to find the set of items that appear frequently over a video as fast as possible. This can be seen as a stream mining problem (possibly using a sliding windows model). To obtain meaningful patterns, a number of constraints can be added. Firstly, to avoid redundancy between patterns, we only mine closed patterns. Secondly, since small regions are often frequent, we impose to mine large frequent closed itemsets, i.e, large tiles in the stream. Finally, to follow a particular object, we can impose spatio-temporal constraints on the items which compose the tiles.

2.3 Constraint-based Graph Mining

In this case, for all segmented frame of the video, we create a region adjacency graph (RAG) [4]. In RAGs, the barycenters of the different regions in a frame

[4] http://www.cs.brown.edu/∼pff/segment/
[5] http://www.videosegmentation.com



Fig. 2. top-1 tile returned for a given frame by a stream tile mining algorithm for a window size of 500 frames.

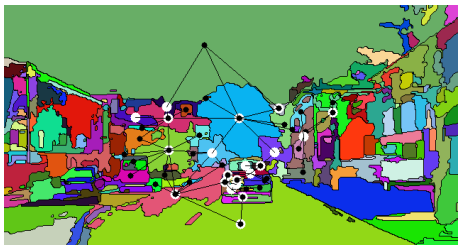


Fig. 3. top-30 star-shaped tile returned for a given frame by a stream tile mining algorithm for a window size of 300 frames.

are the nodes of the graph, and an edge exists between two nodes if the regions are adjacent in the frame. Fig.1 shows a RAG constructed from the regions of a frame of our example video. A video is thus represented as a sequence of graphs (or a dynamic graph). An interesting object might be a frequent subgraph in this sequence. Again, spatio-temporal constraints can be added to link graphs from one frame to another. This problem has been tackled in [6] but the proposed solution is still far from real-time. Casting this problem into an efficient graph stream mining problem would be more valuable for the computer vision community.

3 Preliminary Results

We worked on a real video made of 5619 frames ($\tilde{4}$ minutes). This video is shot from a car while following another car (the main object). In this video the main object goes out of the field of view, its scale and the global illumination change over the video which often confuses the segmentation algorithm.

We applied a stream top-k tile mining algorithm with a sliding window model on the segmented video (using the "dynamic" segmentation) and using the constraint-based itemset mining setting presented above. We printed for each transaction T_t, happening at time t, the top-1 tile of the window W_t (the windows that ends in transaction T_t). This top-1 tile might change at each transaction. The top-1 tile, as shown in Fig. 2 (each region part of this tile contains a white point) often contains regions on the car but also some regions on the background especially the sky and the road which are very frequent. Note that under this setting, the items which belong to a tile are not necessarily connected. Besides, when the car is slowing down, all the regions become frequent and points appear everywhere in the frames. It means that the recall of this algorithm is high but the precision is low. A possible solution to increase this precision would be to encode the adjacency information encoded in the RAG. Without any structural limitations, our problem would be equivalent to mine general graphs which is computationally expensive. To ensure this connectivity at a lower computational cost, we restrict, in a second experiment, the graph to be star-shaped. Moreover,

the tiles in the top-k are often very similar and only differ by one or two items, thus, when mining for the top-k largest star-shaped tile, we impose each top-k tiles to have a different center. Experiments as shown in Fig. 3 shows that we get visually more relevant tiles but this structure is also unsatisfactory as it favors large regions with many adjacent ones (such as the sky and the road again).

Both these experiments show that this method is promising (although we did not yet have a quantitative evaluation of the extracted tiles quality w.r.t object tracking) but a number of relevant constraints should be added to the stream tile mining algorithm to be able to separate the main objects from persistent background even if the background is slowly changing over time. Besides, without focusing much on the algorithm, these mining experiments are still far from done in real time (for this video, they both need more than an hour to be completed). All these problems will be tackled in future work.

References

1. A. Yilmaz, O.J., Mubarak, S.: Object tracking: A survey. ACM Computing Surveys 38(4), 13+ (2006)
2. Borzeshi, E.Z., Piccardi, M., Riesen, K., Bunke, H.: Discriminative prototype selection methods for graph embedding. Pattern Recognition 46(6), 1648–1657 (2013)
3. Çeliktutan, O., Wolf, C., Sankur, B., Lombardi, E.: Real-time exact graph matching with application in human action recognition. In: Human Behavior Understanding - Third International Workshop, HBU 2012, Vilamoura, Portugal, October 7, 2012. Proceedings. pp. 17–28. Lecture Notes in Computer Science, Springer (2012)
4. Chang, R.F., Chen, C.J., Liao, C.H.: Region-based image retrieval using edgeflow segmentation and region adjacency graph. In: IEEE International Conference on Multimedia and Expo. pp. 1883–1886 (2004)
5. Chi, Y., Wang, H., Yu, P.S., Muntz, R.R.: Moment: Maintaining closed frequent itemsets over a stream sliding window. Data Mining, IEEE International Conference on 0, 59–66 (2004)
6. Diot, F., Fromont, É., Jeudy, B., Marilly, E., Martinot, O.: Graph mining for object tracking in videos. In: ECML/PKDD (1). LNCS, vol. 7523, pp. 394–409 (2012)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vision 59(2), 167–181 (Sep 2004)
8. Fernando, B., Fromont, E., Tuytelaars, T.: Effective use of frequent itemset mining for image classification. In: Europ. Conf. on Computer Vision. pp. 214–227 (2012)
9. Goszczynska, H.: Object Tracking. InTech (2011)
10. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. CVPR (2010)
11. Jiang, N., Gruenwald, L.: Cfi-stream: mining closed frequent itemsets in data streams. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 592–597. KDD '06, ACM, New York, NY, USA (2006)
12. Quack, T., Ferrari, V., Leibe, B., Van Gool, L.: Efficient mining of frequent and distinctive feature configurations. In: Proceedings 11th IEEE international conference on computer vision, ICCV 2007 (2007)
13. Yuan, J., Yang, M., Wu, Y.: Mining discriminative co-occurrence patterns for visual recognition. In: CVPR. pp. 2777–2784 (2011)