Multi-Objects Segmentation and Tracking Based Graph-Cuts

Amira Soudani
RIADI Laboratory, Team of research SIIVA.
University of la Manouba, Higher Institute of Computer
2 Rue Abou Rayhane Bayrouni
2080 Ariana, Tunisia
amira.soudani@gmail.com

Ezzeddine Zagrouba
RIADI Laboratory, Team of research SIIVA
University of la Manouba, Higher Institute of Computer
2 Rue Abou Rayhane Bayrouni
2080 Ariana, Tunisia
ezzeddine.zagrouba@fsm.rnu.tn

ABSTRACT
In this paper we present an algorithm that joints segmentation and tracking of multiple objects in a video via graph-cuts optimization technique. The proposed approach is composed of two steps. First, we initialize tracked objects through an initialization step based on background subtraction algorithm. Hence we obtain initial observations that will be used to predict the location of target object in the next frame. Then, we process a tracking step based on an energy function associated to each predicted observation. The minimization of this energy via graph-cut allows us to yield a better segmentation that matches extracted observations with initial detected objects. Experimental validation of the proposed method is performed in several video sequences and provides us significant tracking results.

Keywords
Tracking, Segmentation, Graph-Cuts, Prediction.

1. INTRODUCTION
Object segmentation and tracking in video has been the focus of many researches. In fact, robust and accurate separation of background from foreground objects has been considered crucial in many applications. Several object tracking approaches have been proposed in this aim. In addition to the algorithm itself, the difference between those methods lies on the choice of the representation and shape of the tracked objects, on the property of the the image and on the nature of the estimated motion. This choice depends on the application and the processed video.

Tracking algorithm can be classified on three categories. The first is the the Point Tracking methods that aim to match detected objects between successive images. There are deterministic methods which associate observations to tracked objects by minimizing a distance computed based on some characteristics of the object (proximity and appearance) [1], [2], [3], [4]. We find also the probabilistic methods that deal with variations (noise, movement, appearance) by adding an incertitude to the models associated to the object and the observations [5]. There are also methods based on the minimizing of energy functions in order to follow a contour or a region taking into account the topology changes [6], [7], [8], [9], [10]. Secondly, there are Kernel Tracking methods which are based on the tracking of a predefined shape (rectangle or ellipse) around or inside the tracked objects. They are based on the conservation of the appearance (usually color or luminance) of the object for at least two consecutive frames. Those methods can be based on the differential tracking of object which assumes a conservation of the luminance of visible pixels between two consecutive frames [11], [12], [13] or based on the tracking of color distributions [14], [15]. Finally, there are the Silhouette Tracking methods which apply dynamic segmentation without prior knowledge about the object’s shape. They are based on successive segmentations and they generally evolve the edge of the object at the previous frame until its new position at the current one. Current approaches can be classified into methods based on state models that define a model for the object’s edge which will be considered as a state model for the filtering algorithm [16], [17], [18].

Each class of method cited above presents positive as negative points. The Point Tracking methods can treat the case of apparition of new objects in the scene whereas the quality of track is depending well on the external detected observations. Thereby, if they are not well detected the tracking process can fail. The Kernel Tracking methods allow robust tracking with low cost but can’t treat the entrance of new objects on the scene and they are not adapted for the tracking of small objects. Finally, the Silhouette Tracking approaches whose main advantages are their flexibility to handle a large variety of object shapes and their capability for dealing with object split and merge. However they not deal with apparition of new objects on the scene.

On the other hand, the graph-cuts optimization techniques has reached important results especially in image segmentation [19], [20]. Therefore, many works tried to adapt this technique on multi-objects tracking and it gives considerable results [21], [22], [23].

In this paper, we address the problem of multiple object segmentation and tracking. We will present an algorithm of
silhouette tracking based on min-cut/max-flow algorithms. Hence, we generalize the graph-cuts technique for segmentation and the tracking of multiple targets based on energy minimization. We begin with an initialization step in order to initialize moving objects. An improved background subtraction algorithm is applied. Each extracted observation will be modeled by a mask of pixels that will be propagated in the next frames based on the optical flow vectors associated to the considered mask. This process provides us a prediction which will be used to evaluate the pixels likelihood in current frame by computing their color distribution from the previous one. Finally, a segmentation is performed by minimizing an energy function computed for each object. In fact, it is a labeling function combining two terms: the data dependent term and the smoothness one. This function is evaluated for the assignment of each pixel to a given label. The determination of the energy function is constrained by a prediction step which allows us to lead segmentation to extract the corresponding matching observation.

The reminder of the paper is organized as follows. Section 2 outlines the proposed approach. Then, section 3 describes the energy function associated to each tracked object. In section 4, the experimental results are shown. Finally, in section 5, we summarize our work and we present possible future research directions.

2. OVERVIEW OF THE METHOD

As hypothesis, we assume that the background is static, so our goal is to track the dynamic objects detected in the scene. Hence, we will joint segmentation to tracking in order to extract observations that best matches with initial detected object. Thereby, the proposed method consists of two steps (Algorithm 1).

We first proceed to an initialization step. In fact, the user is required to roughly select an area belonging to the background at the first frame in order to define the color distribution associated to it, then we perform a background subtraction algorithm including a post-processing step providing us a set of observations denoted $Obs^i_0$ at time $t = 0$ by removing noise and small objects with $i = \{1..N\}$. Then, we proceed to the tracking process which includes the prediction of a mask followed by the determination of an energy function and its minimization based on graph-cuts optimization algorithm. This will be the subject of the next paragraph.

3. OBJECT SEGMENTATION AND TRACKING

Each extracted observation $Obs^i_t$ associated to an object $i$ at time $t$ is represented by a set of pixels forming a mask, $M^i_t$. This last will be used to compute a predicted one denoted $M^{i+1}_t$ associated to the frame at time $t + 1$. We translate pixels belonging to each mask $M^i_t$ by adding the average of the optical flow vectors at each frame at time $t$, it is defined as follows:

$$M^i_{t+1} = M^i_{t} + \bar{d}^i_{t-1}. \quad (1)$$

Algorithm 1 Overview of the proposed method

**Input:** Sequence Frames

**Output:** Tracked Observations

1. **Object initialization (at $t=0$)**
   - Background initialization
   - Background subtraction
   - Objects selection: $Obs^i_0$, $i = \{1..N\}$

2. **Object Segmentation and Tracking**
   For each frame $t$
   - For each observation $Obs^i_t$ do
     - Mask prediction:
       - Optical flow estimation
       - Mask prediction (equation 1)
       - Segmentation based Graph-cuts
     - Graph construction and estimation of edges costs (equation 3)
     - Energy minimization (equation 7)
     - Mask Updating (equation 8)
   done

done

$$d^i_{t} = \frac{\sum_{j \in M^{i-1}_t} \| \bar{V}_j \|}{|M^{i-1}_t|}. \quad (2)$$

where $j$ is a pixel belonging to the mask $M^{i}_t$, $\bar{V}_j$ is its associated optical flow vector.

Each object will be tracked independently, so an energy function $E^i_t$ associated to object $i$ at the frame of time $t$ will be defined which corresponds to a weighted graph. This energy will be minimized by the min-cut/max-flow algorithm [24]. The mask prediction step allows us to compute a color distribution for each observation that will be used in the definition of this function.

The energy function $E^i_t$ is determined for the assignment of each pixel $p$ to a label $l^i_t = \{\text{background} : \text{"bg"}, \text{foreground} : \text{"fg"}\}$ and will be minimized via graph-cuts optimization technique. It is composed of two terms: a Data Term, $D_{D^i(l^i)}$ that penalizes the assignment of a given pixel $p$ to a given label $l$ based on its color distribution and a Smoothness Term $S_{\{p,q\} \in N(l^i_p, l^i_q)}$ which penalizes labeling discontinuity of neighboring pixels $p$, $q$ even if they have similar intensities. In the next paragraphs the energy function model will be specified and the graph construction and minimization will be illustrated.

3.1 **Energy function model**

The energy function is computed for each observation $Obs^i_t$ at each frame at time $t$, it is defined as follows:
\[ E_i^t(l) = \sum_{p \in P} D_{i,p}^t(l_p) + \sum_{(p,q) \in V} S_{i,(p,q)}^t(l_p, l_q). \]  

(3)

where \( l \in \{fg, bg\} \), \( p \) and \( q \) two pixels belonging the set of all pixels \( P \) in the image and \( V \) represents the set of neighboring pixels.

The first term of this energy function is a data dependent term that evaluates the cost of the assignment of a pixel to a given label while the second one is a regularization term which penalizes the assignment of neighboring pixels to different labels (border discontinuity).

### 3.1.1 Data term

This term evaluates the penalty for assigning a pixel to a given label. In fact, a color distribution is computed for the background and the foreground based on the previous given label. In fact, a color distribution is computed for each pixel to a given label while the background color distribution is based on the pixels selected by user in the initialization step:

\[ D_{p,l}^t(l_p) = \sum_{p \in P} -\log P(I_p | l_p). \]  

(4)

where \( P(I_p | l_p) \) denotes the probability at pixel \( p \) in the reference image belonging to a label \( l_i \).

For a given label \( l_i \), the global component of color model is represented by a Gaussian Mixture Model (GMM):

\[ P(I_p | l_p = l_i) = \sum_{k=1}^{K} \omega_{ik} N(I_p | \mu_{ik}, \Sigma_{ik}). \]  

(5)

where \( \mathcal{N} \) is the Normal distribution and \( \omega_{ik}, \mu_{ik} \) and \( \Sigma_{ik} \) represent respectively the weight, the means and covariance matrix of the \( k^{th} \) component for label \( l_i \), \( K \) is number of the components associated to the mixture model of label \( l_i \).

### 3.1.2 Smoothness term

This term is determined between all pairs of neighboring pixels. It is based on color gradient and its goal is to smooth the segmentation of tracked object by penalizing the assignment of each pair of neighboring pixels to different labels:

\[ S_{(p,q) \in N}(l_p, l_q) = \text{dist}(p, q)^{-1} \exp \left( -\frac{\| I_p - I_q \|^2}{2\sigma^2} \right). \]  

(6)

where \( \text{dist} \) is the standard L2 Euclidean norm computing pixel distance in the image between neighboring pixels \( p \) and \( q \) and \( \sigma^2 = \frac{1}{|N|} \sum_{(p,q) \in N} \| I_p - I_q \|^2 \) is the average squared norm.

This term penalizes labeling discontinuity of neighboring pixels even if they have similar intensity.

### 3.2 Energy minimization based graph-cuts

The energy function cited above will be modeled by a graph \( G \) composed by a set of edges \( E \) and nodes \( V \), \( G = \{E, V\} \). Each pixel is considered as a graph node in addition to two extra nodes for label region (background or foreground). The data dependent term \( D_{p,l} \) is implemented by connecting each pixel \( p \) to these extra nodes. In fact, the edge weights represent the penalty of assigning each pixel \( p \) to label \( l \).

The smoothness term is computed by connecting each pairwise combination of neighboring pixels \((p,q)\) with an edge weight representing the penalty for assigning those pixels to different labels. The min-cut of this weighted graph provides us the the segmentation that best separates regions.

The final labeling of pixels is obtained by the minimization of the energy function defined below. This provides us the better segmentation of object \( i \) at time \( t \). Successive segmentations obtained by graph cut which take into account the prediction step let us to track each detected observation in the video.

\[ l_i^t = \arg\min E_i^t(l). \]  

(7)

In order to yield the tracking process more efficient, we update the predicted mask provided by the prediction step. In fact, we use the segmented object obtained by the energy minimization step to rectify the predicted mask associated to this object. This allows us to minimize the error propagation in the predicted mask.

\[ M_i^t = l_i^t. \]  

(8)

### 4. EXPERIMENTAL RESULTS

To improve the efficiency of the proposed algorithm, we perform the tracking on color sequence from PETS2006 data corpus (sequence1 camera 4), frame’s size is relatively big (720 × 576) pixels with a rate of 2.5 frames/second. It shows pedestrians who are walked. It is a relatively simple sequence with static background. The parameters are tuned as follow, we used a 8-neighborhood system, in fact the choice of of neighborhood affects the smoothness of the segmentation, so smaller neighborhood introduces irregular segmentation. The number of classes of the Gaussian Mixture Models was set to 5 and the experimentation are done on a standard workstation. First, we extract initial observations by subtracting the reference frame (figure 1(a)) to the current one (figure 1(b)). Extracted observations are surrounded in red on figure 1(c). The initialized objects will be tracked via successive segmentations based on graph-cuts optimization technique.

As shown on figure 2, the segmentation of the pedestrians is well estimated along the sequence even if detected objects have opposite motion direction (figure 2(k)) and detected silhouettes fit well with the extracted objects and they are automatically adjusted while objects are moving. Thereby, an experimental prototype is proposed to evaluate the efficiency of the proposed algorithm. There are four basic types of errors that this system can make. First, the system may
indicate the presence of an object which does not exist. A second type of error may occur when an object exists, but the system does not recognize it. The third type of error occurs when one object is tracked by multiple estimates. The last type of error occurs when multiple objects are merged on one detected observation. As ground truth objects, we were based on extracted observations defined manually by the user. The defined metrics are as follow:

- True Positive (TP): A reference object matching detected observation.
- False Positive (FP): An observation is not associated with a reference object.
- False Negative (FN): A reference object is not associated with an observation.
- Precision: TP / Number of all detected observations.
- Sensitivity: TP / Number of all reference object.

We apply the proposed approach on four sets of consecutive frames from PETS 2006 data corpus in order to reduce the computational time and we compare our method to the one presented by [25]. We deduce that our algorithm is better in term of extracted silhouettes which fit well with the target object in all the frames while they are moving. Figure 2 presents sample frames extracted from each set on which we show the results of the proposed algorithm. We compute the average of each metric for each set (table1). In fact, the best results are obtained on the first set, while the other results are relatively good. The decrease of the precision and sensitivity is due to the merging of detected objects in some frames even if there are spatially nearest to each other and can so affect the tracking results in successive frames.

5. CONCLUSION
In this paper we presented a multi-tracking approach based on graph-cuts optimizations. It is composed of an initialization step based on background subtraction which provides initial observations to be tracked in the video sequence.
Then, we perform successive segmentations based on the minimization of an energy function via graph-cuts optimization technique. This energy function includes a data dependent term based on color distributions deduced from predicted mask. The second term is the smoothness one based on color gradient which penalizes labeling discontinuity of neighboring pixels even if they have similar intensity. Experiments are performed and provided us good tracking results.

As future directions, we aim to improve the tracking process by including additional constraints to treat objects overlapping and merging. Therefore, the computational time can be improved by using multi-label graph method and by updating the graph at each frame. Also, we will try to joint segmentation, tracking and reconstruction based on graph-cuts minimization [26], [27].

6. REFERENCES