Unsupervised Graph Based Video Object Extraction

Submission Type: Full Papers

Kapil Gupta
Indian Institute of Technology,
Patna-800013, Bihar, India
kg3634@gmail.com

Maheshkumar H. Kolekar
Indian Institute of Technology,
Patna-800013, Bihar, India
mahesh@iitp.ac.in

Abstract

A method to extract the object containing regions in the ‘object proposal’ set in the video is employed. The segmented object containing areas are then employed to construct segmentation models for optimal video object extraction. First, an unsupervised graph based framework is used for detection and extraction of foreground in the video. We take into account the general properties (spatially cohesiveness, smooth motion trajectories, predicted-shape similarity, appearance and motion across frames) of object and use this to extract object from all available object proposals. Second, object proposal expansion is done by using motion based proposal predictions and unsupervised graph is constructed based on these proposals. Last, a discriminative function is introduced to differentiate between moving and background. This method is evaluated on segtrack dataset and it gives better result than the state-of-the-art.

Keywords

Object proposal, unsupervised graph, and discriminative function.

I. Introduction & Previous Work

Our objective is to extract the object from the video in unsupervised way. Video object extraction is a standard problem in computer vision. It is a primary step in various video understanding application such as summarization of video, retrieval of content based video, activity recognition in video and replacement of the target in particular video. For video object extraction supervised and unsupervised methods are used. In the latter class of methods, [2, 3, 4] annotation is for video object segments in key frames for initial step. Then optimal methods involving motion and appearance constraints are used for all frames. Some methods [5, 6] need annotations of object region for first frame, then tracking of object region is used for remaining frames. These supervised methods give good segmentation results. However, in general, video data is processed in big amount in many computer vision applications that makes supervised method costly. So, unsupervised techniques have also been developed for video object extraction. Some of these approaches use grouping of motion [7, 8, 9] to extract the object from video. Some techniques [10, 11, 12] use efficient optimization techniques for video object segmentation. However, all of these unsupervised techniques do not tell about the appearance and movement of object. Therefore, the extracted regions do not correspond to an exact object but only to coherent appearance or motion containing image regions.

Recently, several techniques [13, 14, 15] have been developed that tell about the appearance of a general object. Particularly, the technique [15] tells about the appearance and motion of the object and proposes the concept of object proposal. Lee et al. [1] and Ma and Latecki [16] employs this method to segment the object from video. Lee et al. [1] used the set of object proposals and then employs spectral graph clustering to get the several binary outlier/inlier partitions.
Unsupervised Graph Based Video Object Extraction

**Figure 1. Selection of Object Region from the Set of Object Proposals.** The first row indicates frames in the video. The second row displays object proposals corresponding to key frames (in red outlines) segmented by [1]. “?” shows the failure of object segmentation by the approach. The third row indicates object proposals selected by unsupervised graph approach. Observe that the unsupervised graph approach extracts object in all frames. The outcomes of row 2 and 3 are before to GMM and MRF optimization. In this paper we show that expansion of object proposals gives good improvement in object extraction performance. In Table 1, we have shown the quantitative results and comparisons to previous approaches.

Specific object containing regions correspond to each partitioned inlier. In order to measure the ‘objectness’ of proposal in the partition, motion and appearance based cues have been used. Highest average weighted cluster in terms of ‘objectness’ is corresponding to object in the video. There is one drawback of this method that it ignores the order of the proposals in the video, and therefore, does not have an explicit model of the shape and location of object with time. Ma and Latecki [16] resolves this issue by taking the object proposal of adjacent frames. Video object extraction problem is solved as a constrained Maximum Weight Cliques problem in order to find the exact object region from all the frames simultaneously. However, this problem is based on non-deterministic polynomial. So approximation has been taken to solve this problem. So methods [1, 16] are based on object proposal and both have limits compared to the proposed approach. In both methods, object proposal of specific frame is not directly related to the object proposal of neighboring frame. And both approaches could not tell the actual shape of the object in neighboring frames when calculating similarity between regions, which does not make it suitable for the segmentation of fast moving objects.

**Figure 2.** Some of the (6) proposals (in the right) generated by the previous technique [15]. No proposal is corresponding to desired object (person in the left frame). The aim of our
work is to expand the object proposal domain and select the proposal related to desired object only.

In this paper, we propose a method that tries to remove the drawback of the previous approaches. Note that, in general, the appearance and the shape of the object do not remain same in all frames, they change in every frame. So our main goal is to select object proposal in order of high ‘objectness’ and similarity across frames. Finally, we employ optical flow method to track the change in the shape of object, and calculate the change in actual and predicted appearance (along with shape) to compute the similarity between the object proposals of different frames. Appearance and motion based cues are used to calculate ‘objectness’ and these cues are based on high optical flow gradients at the outlines between the background and object proposals. Moreover, the selection of object proposal related to desired object is done through unsupervised and leveled graph for which (unlike[16]) there is an existence of optimum solution. Note that, method [1] has not used object proposal positions in temporal sequences, then it can generate proposals that do not correspond to desired object for many frames (Figure. 1). So unsupervised graph based method not only uses object proposals in efficient way but also enhances the object proposal domain using the object proposals of adjacent frames. So enhancement of object proposal and prediction of shape and appearance give accurate video object extraction. We have used this method on segtrack dataset and it gives better results than previous techniques.

In Section II, the leveled unsupervised graph based method is introduced, In Section III, comparative results are given for publicly available dataset (segtrack). The conclusion is given in Section IV.

II. Leveled Unsupervised Graph Based Video Object Extraction

A. The Layout

The layout of this approach has three steps (as shown in Figure 3); 1. Expansion and production of object proposal for each frame by using the object proposals of neighboring frames. 2. Leveled unsupervised graph generation from the object proposals of each frame. The highest weighted path in the graph maximizes similarity and objectness, and shows most probable object proposals related to desired object in the video. 3. The background and object models are made through the desired object proposals by using GMM (Gaussian Mixture Models) and MRF (Markov Random Field) based optimization for refinement of extracted results. Since this approach is basically based on leveled unsupervised graph for the desired object extraction from video, it is discussed first.

B. Leveled Unsupervised Graph Based Structure

Our main goal is to segment the object proposals with high shape (along with appearance) similarity, high ‘objectness’ and smooth changing shape from the object proposal domain acquired from the video. Since our objective is to segment the desired object only, our need is to segment only single object proposal from each frame. To fulfill these needs, the leveled unsupervised graph is formed as follows. We have denoted an object proposal by two nodes: an ‘enter node’ and an ‘exit node’. Unsupervised graph has two types of edges: single edges and poly edges. The objectness of an object proposal is measured by the weights of single edges. The weights of single edges related to measuring objectness are discussed in section II (C). All enter and exit nodes form a level in the same frame. A directed single edge is made from enter node to exit node in latter levels. The shape and appearance similarity between the object proposals of different frames is measured by the weights of poly edges. The function of weights assignment of poly edges are discussed in section II (D).
The example of graph structure is shown in Figure 4. It represents frame $a-1$, $a$ and $a+1$ of the graph, with corresponding levels of $2a-3$, $2a-2$, $2a-1$, $2a$, $2a+1$ and $2a+2$. Note that, we have represented only 3 proposals for every level, however, several (in the range of hundreds) object proposals are generated for every frame and there may be difference in the no. of proposals generated for different frames. The light...
Figure 4. Leveled Unsupervised Graph Structure. Node “T” and “S” are virtual sink and source nodes respectively. The edges in “T” and “S” do not contain any weight. The light blue nodes and the dark blue nodes are “enter nodes” and “exit nodes” respectively and each light blue-dark blue pair denotes an object proposal. All enter nodes and exit nodes in same frame are organized in a level. Dark blue edges are the single edges and green edges are the poly edges.

blue nodes are “enter nodes”, the dark blue nodes are “exit nodes”, the dark blue edges are single edges and the green edges are poly edges (in the graph we have shown only a few of poly edges for understanding purpose) Node S (source of left hand sided black edges) denotes a virtual source node and node T (sink of right hand side black edges) denotes virtual sink node. Both S and T have o weighted edges. Note that, we do not need to build poly edges from an exit node to all the enter nodes in the latter levels. In general, only taking poly edges to next three consecutive frame is sufficient for most of the detections.

C. Single Edges

The objectness of object proposal is measured by single edges. To measure objectness, motion and appearance are two key factors. The equation for discriminative function based on motion and appearance for object proposals is given by:

\[ S_{single}(r) = M(r) + A(r), \]  

(1)

where \( r \) denotes object proposal. \( M(r) \) is motion score and \( A(r) \) is intra-frame appearance score. \( M(r) \) is defined in terms of average Frobenius norm. Forbenius norm is defined for optical flow variation around the outlines of object proposal \( r \). The Frobenius norm for optical flow variation is given by:

\[ \| V \|_F = \| \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} \|_F = \sqrt{u_x^2 + u_y^2 + v_x^2 + v_y^2}, \]  

(2)

where \( V = (u, v) \) denotes optical flow of the frame in forward direction, \( u_y, v_y \) and \( u_x, v_x \) denote optical flow gradients in y and x direction consecutively.

In general, the motions of background and foreground objects differ, outline of forward object shows discontinuity in motion. The reason behind the introduction of motion scoring function is this discontinuous motion. From Figure 5, it can be observed that the gradient of optical flow has high value around foreground object outline. The value of the gradient of optical flow is measured by using the Forbenius norm, which is given in equation (2). The desired object region is directly proportional to the Frobenius norm. Practically, we do not get exact desired object outline corresponding to the gradient of optical flow. So, we take the average of optical flow after dilating the object proposal outline, which is considered as motion score. The example is shown in Figure 5. \( A(r) \) (appearance score) is calculated by the method[15].
Figure 5. The Motion Scoring of the Magnitude of the Gradient of Optical Flow. One of the video frame from segtrack dataset is shown in row 1, column 1, one of the object proposals and its dilated outline are shown in row 1, column 2 and 3 respectively. Optical flow of the frame in forward direction and the scale of gradient of optical flow are represented in row 2, column 1 and 2 respectively. The output for the magnitude of the gradient of optical flow corresponding to the dilated outline of the object proposal is shown in row 2, column 3.

D. Poly Edges

Similarity between object proposals of different frames is acquired by poly edges. Color, shape, size and position are key factors to be considered for the computation of similarity between proposals. These factors are defined as the weight of poly edges as follows:

\[ P_{\text{poly}} = \lambda \cdot P_{\text{overlay}}(r_p, r_q) \cdot P_{\text{color}}(r_p, r_q), \]  

(3)

where \( r_p \) and \( r_q \) are object proposals from frame \( p \) and \( q \), \( \lambda \) is a constant value for correcting the ratio between single and poly edges. \( P_{\text{overlay}} \) denotes the overlap similarity between object proposals and \( P_{\text{color}} \) denotes the color histogram similarity:

\[ P_{\text{color}}(r_p, r_q) = \text{hist}(r_p) \cdot \text{hist}(r_q) ^ T, \]  

(4)

where \( \text{hist}(r) \) is the normalized color histogram for an object proposal \( r \).
Unsupervised Graph Based Video Object Extraction

$$P_{\text{overlay}}(r_p, r_q) = \frac{|r_p \cap \text{twist}_{pq}(r_q)|}{|r_p \cup \text{twist}_{pq}(r_q)|}$$

(5)

where \(\text{twist}_{pq}(r_q)\) is the twisted object proposal from \(r_q\) by optical flow to frame \(p\). \(P_{\text{color}}\) denotes the color similarity between object proposals and \(P_{\text{overlay}}\) denotes the size and position similarity between object proposals. If regions in the object proposals are close, and shapes and sizes are same, the magnitude would be larger, and contrariwise. In order to predict object proposal regions, we employ optical flow method unlike previous approaches [16, 1]. Therefore, our approach is better in the detection of fast moving objects.

E. Maximum Weighted Path Solution

We have made the leveled unsupervised graph and the goal is clear: to get the maximum weighted path in the unsupervised graph. This graph is unsupervised because we do not need initial annotations in the graph for detection. Assume the graph has \(2f + 2\) levels (\(f\) is the no. of frames), the virtual source node \(S\) is in level 0 and the virtual sink node \(T\) is in level \(2f + 2\). Let \(K_{ab}\) represents the \(b\)th node in \(a\)th level and \(E(K_{ab}, K_{ij})\) represents the edge from \(K_{ab}\) to \(K_{ij}\). Let \(W = (w_1, w_2, \ldots, w_{h+1}) = (N_{01}, N_{10}, N_{20}, \ldots, N_{(2h+2)1})\) be a path from virtual source to virtual sink node. Hence,

$$W_{\text{max}} = \arg \max_w \sum_{a=1}^{h} E(p_a, p_{a+1})$$

(6)

Let \(OPT(a, b)\) be the maximum path value for \(N_{ab}\) from virtual source node. \(W_{\text{max}}\) builds a maximum (simple) weighted path problem for unsupervised graph. The maximum weighted path value satisfies the following recurrence for \(a \geq 1\) and \(b \geq 1\):

$$OPT(a, b) = \max_{i=0...a-1, j=1...M} [OPT(i, j) + E(N_{ij}, N_{ab})]$$

(7)

This maximum weighted path based problem could be easily solved by recursive and sub-divisional based programming, which is given in the book [12]. There is \(O(g + h)\) computational complication for the algorithm, in which \(g\) is the no. of nodes and \(h\) is the number of edges. In practice the value of \(\lambda\) is taken 1.

F. GMM and MRF Based Optimization

As we get the desired object proposals in a video, the outcomes are further filtered by a graph-based approach to get per-pixel extraction outcomes. We construct a graph (space temporal) by joining frames corresponding to time axis with optical flow. Each pixel in a frame is represented by each node of the graph and edges are represented by the 8-neighbour with in one frame and the backward-forward 18 neighbors in adjacent frames. Energy function is defined for labeling \(f = [f_1, f_2, \ldots, f_n]\) of \(n\) pixels with prior knowledge of \(h\):

$$E(f, h) = \sum_{i \in S} D_i^h(f_i) + \lambda \sum_{(i, j) \in h} V_{ij}(f_i, f_j)$$

(8)
where \( N \) contains adjacent pixels, \( S = \{p_i, \ldots, p_n\} \) is the set of \( n \) pixels in the video, and \( i,j \) index the pixels, \( p \) could be set to 0 or 1 which denotes background or foreground correspondingly. The unary term \( D_i^h \) describes the cost of labeling pixel \( i \) with label \( f_i \) which we find from the Gaussian Mixture Models (GMM) for both color and position.

\[
D_i^h(f_i) = -\log(\alpha U_i^c(f_i, h) + (1 - \alpha) U_i^l(f_i, h))
\]

where \( U_i^c(f_i, h) \) is the color-induced cost and \( U_i^l(f_i, h) \) is the position cost. Binary term \( V_{i,j}(f_i, f_j) \) is computed by the definition given in [17]. Equation (7) is optimized by graph cut based minimization approach [18], and after that we get the final extracted outcomes.

### G. Expansion and Generation of Object Proposal

Our goal is to select object proposals related to desired object only. In order to do that we expand the object proposal domain. So only optical flow or shape (along with appearance) based cues are not used for expansion purpose. The example is shown in Figure 6. Several object proposals were produced for frame \( a \) using approach in [15].

We suppose that shape (along with appearance) and position of object varies from one frame to another. In order to expand the object proposal domain for a frame, object proposals of neighboring frames are used. The approach of expansion starts with optical flow guidance (Figure 6). In order to expand object proposal in forward direction, each object proposal \( r_{a-1}^i \) in frame \( a-1 \) is twisted by the optical flow in the forward direction to frame \( a \), then a test is performed is if any proposal \( r_a^b \) in frame \( a \) has a high overlay ratio with the twisted object proposal, i.e.

\[
\alpha = \frac{|\text{twist}_{a-1,a}(r_{a-1}^i) \cap r_a^b|}{|r_a^b|}
\]

The adjacent overlaid regions, for areas in \( a +1 \) with \( \alpha \) larger than .5 are combined into a single area, are taken as extra proposals. Note that, this is an enhancement of proposal domain and not a substitution, so old proposals would also be there. This is an iterative process, although some useful proposals are missing in adjacent frames, they could be generated by this enhancement approach.
III. Results and Discussions

This approach is evaluated using Segtrack dataset [6]. Comparative results are shown in Table 1.

A. Experimental Results

There are 6 videos in this dataset. Ground-truth for each video is also available. We have evaluated this approach on 5 (parachute, monkeydog, girl, cheetah and birdfall) of the videos. Optical flow based model selection approach is used to infer the camera motion. The comparison is given in Table 1. This approach is an unsupervised approach except for parachute video.

The average per-frame pixel rate with respect to ground-truth is calculated by the approach [6].

\[
\text{Error} = \sum_{i=1}^{F} \sum_{k=1}^{\text{width}} \sum_{j=1}^{\text{height}} \frac{\text{XOR}(f_i(k,j), GT(k,j))}{F}
\]

in which \(f\) = extraction labeling outcomes, \(GT\) = ground-truth labeling of the video, \(F\) = no. of frames in the video. Fig. 7 represents the outcomes for the video of Segtrack Dataset.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>supervised?</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>parachute</td>
<td>220</td>
<td>502</td>
<td>235</td>
<td>201</td>
<td>221</td>
<td>222</td>
</tr>
<tr>
<td>girl</td>
<td>1488</td>
<td>1755</td>
<td>1304</td>
<td>1785</td>
<td>1698</td>
<td>1498</td>
</tr>
<tr>
<td>monkeydog</td>
<td>365</td>
<td>683</td>
<td>563</td>
<td>521</td>
<td>472</td>
<td>330</td>
</tr>
<tr>
<td>cheetah</td>
<td>633</td>
<td>1217</td>
<td>1142</td>
<td>905</td>
<td>806</td>
<td>620</td>
</tr>
<tr>
<td>birdfall</td>
<td>155</td>
<td>454</td>
<td>252</td>
<td>288</td>
<td>189</td>
<td>150</td>
</tr>
<tr>
<td>Avg.</td>
<td>452</td>
<td>791</td>
<td>594</td>
<td>592</td>
<td>542</td>
<td>446</td>
</tr>
</tbody>
</table>

Table 1. Comparative Results for Seg Track dataset
Figure 6 Enhancement of Object Proposal. For an object proposal, twisted with each optical flow in frame $a-1$, a check is made if any proposal in frame $a$ has large overlay ratio.
with the twisted one, they are combined into a single desired proposal. This approach produces desired object proposal that could not be generated by [15] by using the proposals of adjacent frame.

Figure 7. Segtrack datasets outcomes. The regions within the red outlines are the extracted desired objects.
IV. CONCLUSIONS

We have used a novel and effective leveled unsupervised graph (UG) based method (Figure 4) to extract the desired object in videos. For evaluation the ‘objectness’ of a proposal and calculation of similarity between object proposals across frames, novel techniques (Section II(C,D)) are used. Comparative results for SegTrack dataset are given in Table 1. The value in the Table 1, average per-frame pixel rate with respect to ground-truth (Avg.), is calculated by the error formula given in equation 11. This approach gives better results compared to previous approaches on SegTrack dataset.

If there are more than one moving objects in the video, then error may be increased corresponding to desired object. This is the limitation of the technique.

REFERENCES


