Spatiotemporal Saliency Estimation by Spectral Foreground Detection

Çağlar Aytekin, Horst Possegger, Thomas Mauthner, Serkan Kiranyaz, Horst Bischof, and Moncef Gabbouj

Abstract—**We present a novel approach for spatiotemporal saliency detection by optimizing a unified criterion of color contrast, motion contrast, appearance, and background cues. To this end, we first abstract the video by temporal superpixels. Second, we propose a novel graph structure exploiting the saliency cues to assign the edge weights. The salient segments are then extracted by applying a spectral foreground detection method, quantum cuts, on this graph. We evaluate our approach on several public datasets for video saliency and activity localization to demonstrate the favorable performance of the proposed video quantum cuts compared to the state of the art.**

Index Terms—**Salient object detection, foreground detection, spatiotemporal, saliency, spectral graph theory.**

I. Introduction

IMAGE and video saliency detection has gained much attention in the last two decades after the seminal work of Itti *et al.* [1]. Saliency maps allow us to filter irrelevant image and video regions which do not contain visually interesting information. Thus, saliency estimation is a valuable preprocessing step for a variety of applications such as image and video summarization [2], stereoscopic video coding [3], image and video retargeting [4]–[6], compression [7], object detection [8], surveillance [9], [10], and action recognition [11].

We focus on identifying a salient object region as a whole to enable further improvements for applications such as object detection or action recognition. Prior work in this research field is mostly based on three cues, namely contrast, appearance, and background. First, the contrast cue covers the assumption that a salient object is in contrast with its local surrounding [12] or the rest of the scene [13]. Second, the appearance cue is related to the expected shape and location properties of an object, such as the fact that an object has a well-defined closed boundary [14] or that it covers a large area of the image region [15], [16]. Finally, the background cue is used in order to define saliency as a dissimilarity measure from a set of expected background regions, such as image boundaries, e.g. [17], [18].

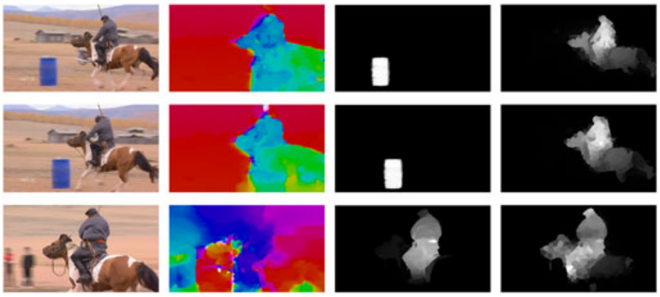


Fig. 1. Importance of motion for spatiotemporal saliency. From left to right: horse-riding frames, optical flow visualization, saliency result of [16], and the proposed VQCUT.

Recently, there have been successful approaches to visual saliency detection in images such as graph-based manifold ranking [19], absorbing Markov chain [20], geodesic saliency [17], saliency filters [21], robust background detection [18], and Quantum Cuts [15]. Although these methods achieve a considerable performance for still images, it is not a straightforward task to apply them directly on videos, as the saliency concept for videos can be much more complex than in still images. This is due to the additional motion information. For example, an object may not have a distinctive local or global color contrast; however, it can be very salient due to its motion (e.g. see Fig. 1). Such issues motivated research on specific methods for video saliency, i.e. detecting salient objects in videos. Moreover, video saliency detection adresses the shortcomings of some video segmentation methods that require manual annotation in the first frame [22], extracts a number of spatiotemporal tubes containing many irrelevant proposals [23] or that are designed to extract only one primary object from the video [24], [25].

Some of the recent approaches to video saliency are as follows. Guo and Zhang [26] represent each frame as a quaternion containing intensity, color, and motion information. Then, a multiresolution model is proposed to calculate the spatiotemporal saliency map of an image by this representation. Liu *et al.* [27] propose a superpixel based saliency model by exploiting motion and color histograms in both local and global manner. Mancas *et al.* [28] compute the saliency map by determining the global rarity in the optical flow using a multi-scale approach. Itti and Baldi [29] extract low-level information such as color,

Manuscript received March 18, 2016; revised October 22, 2016 and March 9, 2017; accepted June 2, 2017. Date of publication June 8, 2017; date of current version December 14, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Qi Tian. *(Corresponding author: Çağlar Aytekin.)*

Ç. Aytekin and M. Gabbouj are with the Department of Signal Processing, Tampere University of Technology, Tampere 33101, Finland (e-mail: caglar.aytekin@tut.fi; moncef.gabbouj@tut.fi).

H. Possegger, T. Mauthner, and H. Bischof are with the Institute for Computer Graphics and Vision, Graz University of Technology, Graz 8010, Austria (e-mail: possegger@icg.tugraz.at; mauthner@icg.tugraz.at; bischof@icg.tugraz.at).

S. Kiranyaz is with the Department of Electrical Engineering, Qatar University, Doha 2713, Qatar (e-mail: mkiranyaz@qu.edu.qa).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2017.2713982

motion, flicker, orientation and intensity with linear center-surround filters and obtain a master-saliency map. Rahtu *et al.* [30] use motion, illumination, and color contrast cues to obtain a saliency model, whereas Singh *et al.* [31] use color, motion, objectness, and boundary cues to extract different saliency maps. These saliency maps are then merged via weight learning by linear support vector machines. Fang *et al.* [32] compute temporal and spatial saliency separately and then merge them considering an uncertainity analysis regarding the confidence on each saliency map. Zhao *et al.* [33] learn a fixation bank including color, intensity, orientation and motion features and the human fixations around a given location. During testing, the feature maps are decomposed into blobs and local activation patterns are matched against the fixation bank to determine the saliency value of the blobs. Mauthner *et al.* [34] introduce a Bayesian saliency formulation via encoding-based joint distribution estimations for color and motion information separately, and then both local and global color and motion saliency estimations are merged in an adaptive manner.

Most of these video saliency approaches consider spatial and temporal saliency as separate problems and try to fuse them after separately calculating each, e.g. [27], [31], [32], [34], whereas others rely only on motion information, e.g. [28]. However, a salient object might be visually interesting either because of its motion or solely based on color contrast. Additionally, object saliency might result from both color and motion information combined, although separately they might not be in high contrast with the rest of the video. Therefore, relying solely on one of these measures or considering them separately and then merging them to estimate saliency would fail to detect the salient objects in such scenarios. Zhao *et al.* [33] adress this issue by learning saliency from combined features, however their method is supervised and aimed to detect human fixation maps, rather than salient objects. Finally, the background and appearance cues that proved useful in image saliency tasks have not been given much attention by the video saliency methods, although they may bear significant information for an accurate salient object detection.

In this paper, to address these issues we propose a unified method which simultaneously exploits: (i) the local color and motion contrast, (ii) the global motion contrast, (iii) the appearance cue modeled as an expectation of large area coverage, and (iv) the background cue via estimating the dominant motion at the video boundaries. To this end, we formulate an optimization criterion which combines all of these cues. We first represent the video by temporal superpixels [35] and form a graph using our novel edge weights which are based on these saliency cues. Next, we apply Quantum Cuts (QCUT), a recently proposed spectral foreground extraction algorithm [15] in order to find the saliency probability of regions, given a set of background regions. The main reason for selecting QCUT is the fact that it differs from other spectral graph algorithms, such as normalized cuts [36], by its ability to formulate an optimization problem solely based on the foreground and to incorporate prior background information. QCUT was shown to outperform many competing algorithms for salient object detection in still images, e.g. see [15], [16]. The graph construction of these works however, is designed for still images. The main difference of the proposed method compared to QCUT is the novel graph construction that is more suitable for salient object detection from videos. We show that this contribution is crucial as it leads to a 25 percent relative improvement on maximum F1 measure on the results obtained by applying QCUT to each video frame separately.

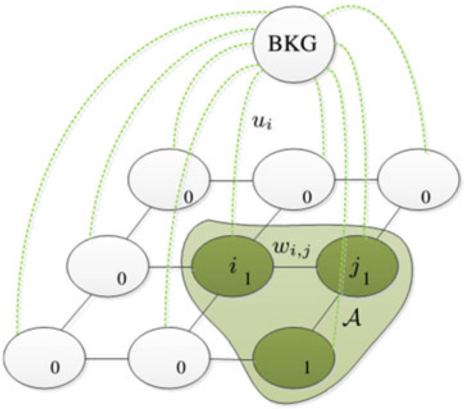


Fig. 2. Graph structure for QCUT. To separate foreground nodes (denoted by 1) from background nodes (denoted by 0), QCUT relies on the pairwise affinities $w_{i,j}$, as well as the unary background prior u_i (i.e., the connection to the auxiliary background node, denoted as BKG).

The rest of the paper is organized as follows: First, we briefly summarize QCUT in Section II. Then, we present the proposed spatiotemporal saliency detection method *Video Quantum Cuts* (VQCUT) in Section III. In Section IV, we demonstrate the performance of our unified method for both video saliency and activity localization tasks. Finally, Section V concludes the paper.

II. Quantum Cuts

Quantum Cuts (QCUT) is a spectral foreground detection method for graphs. Consider a graph with nodes to be labeled as foreground or background, augmented by an additional auxiliary node BKG to represent the background sink. Connecting every node to its neighbors and the auxiliary BKG node, we obtain a representation as illustrated in Fig. 2. For simplicity, this illustration defines nodes to correspond to image pixels with a 4-connected neighborhood. However, this representation can be easily generalized to any connected graph.

Let $w_{i,j}$ denote the affinity between two nodes i and j of the graph, and u_i denote the background prior for node i (i.e. the unary potential of its connection to the BKG node). From a saliency perspective, we then seek the foreground partition $\mathcal{A}$ which is in contrast with the rest of the graph (contrast cue), has a large area (appearance cue), and is dissimilar from the background (background cue). All of these cues can be combined in a single optimization criterion as

$$\mathcal{A}^{\star} = \arg\min_{\mathcal{A}} \frac{\mathrm{cut}\left(\mathcal{A}, \bar{\mathcal{A}}\right)}{\mathrm{area}\left(\mathcal{A}\right)} \tag{1}$$

which can be rewritten as

$$\mathbf{y}^\star = \arg\min_{\mathbf{y}} \frac{\sum_{i,j} w_{i,j}\left(y_j - y_i y_j\right) + \sum_i u_i y_i}{\sum_i y_i} \tag{2}$$

where $\mathbf{y} = (y_1, \ldots, y_N)^\top$ is a binary label vector with $y_i = 0$ indicating that node i belongs to the background and $y_i = 1$ indicating foreground, respectively. The numerator in (2) consists of pairwise terms based on the connection to neighboring nodes, and unary terms which model the background prior. This joint optimization criterion simultaneously maximizes the area of $\mathcal{A}$, the local contrast of $\mathcal{A}$ via the pairwise terms, and the dissimilarity of $\mathcal{A}$ from the background via the unary terms.

Although this minimization is NP-hard, it was shown in [15] that a spectral approximation of the solution can be achieved by introducing a vector $\mathbf{z} = (z_1, \ldots, z_N)^\top$ satisfying $z_i^2 = y_i$, i.e. $z_i \in \{-1, 0, 1\}$. Without loss of generality, (2) can be expanded by adding the term $\sum_{i,j} w_{i,j}\left(z_i^2 z_j^2 - z_i z_j\right)$ to limit the solution set of $\mathbf{z}$. This term only penalizes the assignments $(z_i, z_j) = (1, -1)$ and $(z_i, z_j) = (-1, 1)$ and thus, leads to a tighter solution set for $\mathbf{z}$. For all other pairwise assignments, this term is ineffective and has no effect on the actual labeling $\mathbf{y}$. Hence, one can rewrite (2) as

$$\mathbf{y}^\star = \arg\min_{\mathbf{y}} \frac{\sum_{i,j} w_{i,j}\left(z_j^2 - z_i z_j\right) + \sum_i u_i z_i^2}{\sum_i z_i^2}. \tag{3}$$

In matrix form, this minimization problem corresponds to

$$\mathbf{z}^\star = \arg\min_{\mathbf{z}} \frac{\mathbf{z}^\top \mathbf{H}_\mathrm{M} \mathbf{z}}{\mathbf{z}^\top \mathbf{z}} \tag{4}$$

$$\mathbf{H}_\mathrm{M}(i,j) = \begin{cases} u_i + \sum_{k \in \mathcal{N}_i} w_{k,i} & \text{if } i = j \\ -w_{i,j} & \text{if } j \in \mathcal{N}_i \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where $\mathcal{N}_i$ is the set of neighbors of node i. If $\mathbf{z}$ is relaxed to have real values, the minimization in (4) has a spectral approximation as it turns into a Rayleigh Quotient and the solution $\mathbf{y}^\star$ is obtained as the eigenvector corresponding to the minimum eigenvalue λ of the eigenproblem

$$\mathbf{H}_\mathrm{M}\mathbf{z}^\star = \lambda \mathbf{z}^\star, \quad \mathbf{y}^\star = \mathbf{z}^\star \circ \mathbf{z}^\star \tag{6}$$

where $\circ$ is the Hadamard product. Note that as long as there exists a non-zero u_i, $\mathbf{H}_\mathrm{M}$ is positive definite and the minimum eigenvalue is greater than zero. Hence, the solution is obtained via the eigenvector corresponding to this minimum eigenvalue. Due to the similarity of (6) to discretized solutions of the time-independent Schrödingers Equation [37] in quantum mechanics, this method is called Quantum Cuts. In particular, the original Hamiltonian operator in quantum mechanics is a special case of the modified Hamiltonian $\mathbf{H}_\mathrm{M}$ in (5), where all the weights are constant.

QCUT was originally introduced as a salient object detection method for still images in [15], where each pixel of the input image corresponds to a separate node within the graph. The affinities between nodes of this 4-connected graph were selected as the inverse of their color distance. Aytekin *et al.* [16] proposed a multi-resolution extension of QCUT by using different granularity levels of a superpixel segmentation to form the graph. In both works, image boundaries were confidently connected to the background node, assuming that these regions are likely to be background.

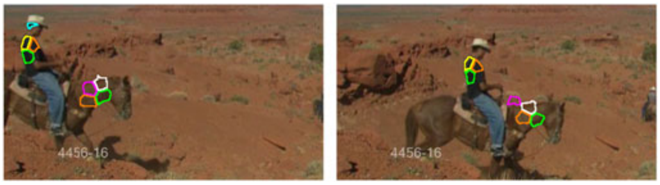


Fig. 3. Tracked temporal superpixels at different frames of a video. See text for details.

Although QCUT achieves notable performance for saliency detection in images, there are several limitations when applying it for video saliency tasks. First, the assumption that the salient object does not touch the boundary is often violated for videos. Second, applying QCUT on a per-frame level may not produce temporally consistent results throughout the video, i.e. an object which is salient in one frame may not be salient in another. Finally, both [15] and [16] only exploit color information to find an optimal cut. However, a salient object in a video may not necessarily be salient in color, but it can be salient due to its motion (recall Fig. 1). Therefore, we need a method to produce temporally consistent results which considers both color and motion saliency in a unified manner. In particular, we design an operator similar to $\mathbf{H}_\mathrm{M}$ in (5) for the whole video and apply Quantum Cuts once for this matrix to obtain the consistent saliency maps for the whole video.

III. Video Quantum Cuts

To apply Quantum Cuts for video saliency detection, we propose a novel spatiotemporal Hamiltonian operator matrix. First, a graph representation of the input video is formed as detailed in Section III-A. Second, we introduce our saliency based affinity measures in Section III-B and finally, we discuss how to model the background prior in Section III-C. Combining these terms, we can then define the spatiotemporal Hamiltonian operator matrix and solve the corresponding video saliency problem in Section III-D.

A. Video Graph Representation

To form a graph for the whole video, we first extract temporal superpixels (TSPs) [35] which are then used to represent the nodes, i.e. each superpixel S_i corresponds to a node of the graph. One drawback of the temporal superpixel extraction is that TSPs are not robust *w.r.t.* partial occlusions, sensitive to noisy optical flow estimation, and are likely to disappear frequently. Fig. 3 illustrates these limitations on a sample video frame. Note that some superpixels on the neck of the horse (magenta and white) shift their position and the superpixel on the rider's head (cyan) is lost after a few frames. Nevertheless, the TSP video abstraction preserves both the color and motion information which we can be useful to solve the video saliency problem in a unified

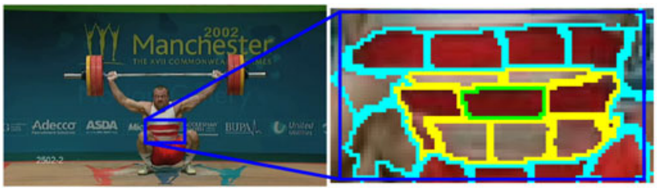


Fig. 4. First-order (yellow) and second-order (cyan) frame-wise neighbors $\mathcal{N}^F_{k,i}$ of a superpixel S_i (green) visualized on a close-up view of the blue rectangle (left).

manner. Moreover, at the end of this section, we will show that our proposed method provides robust solutions to handle the drawbacks of TSPs.

To define the edges of the graph, we first consider the set of frame-wise neighbors $\mathcal{N}^F_{k,i}$ of a superpixel S_i at frame k. We denote the superpixels that share a boundary with S_i as 1st-order neighbors. Similarly, 2nd-order neighbors are those superpixels that share a boundary with any 1st-order neighbor of S_i, see Fig. 4. Experimentally, we observed that including 2nd-order neighbors improves the results due to two valuable contributions, (i) they extend the local contrast information and (ii) they enable us to recover additional texture information. For example, the 2nd-order neighbors connect repeating homogeneous regions as in Fig. 4, which preserves the texture information.

Using the frame-wise neighborhood relation, we can define the set of video neighbors $\mathcal{N}^V_i$ for a superpixel S_i. These are the superpixels S_j which are frame-wise neighbors of S_i throughout the video until either S_i or S_j disappears. More formally, let $\mathcal{G}_k$ denote the graph formed from all superpixels that exist at frame k, where only those superpixels that share a boundary are connected with a weight of 1. Then, the video neighbors $\mathcal{N}^V_i$ of a superpixel S_i are given as

$$\mathcal{N}^V_i = \left\{S_j : S_j \in \mathcal{N}^F_{k,i}, \forall k : O_{k,i,j} = 1\right\} \quad (7)$$

$$\mathcal{N}^F_{k,i} = \{S_j : P_{k,i,j} \leq 2, O_{k,i,j} = 1\} \quad (8)$$

where $P_{k,i,j}$ denotes the length of the shortest path between S_i and S_j in the graph $\mathcal{G}_k$; and $O_{k,i,j}$ is an indicator of the co-occurence S_i and S_j, i.e. $O_{k,i,j} = 1$ if both superpixels are present at frame k. Having defined the edges of our video graph, we next introduce the weights of these edges, i.e. the affinity measure between nodes in this graph.

B. Affinity Measures

As a first measure to decide on the affinity between two superpixels, we exploit their color information. We represent each superpixel by its mean L*a*b* color

$$\mu^{\mathrm{C}}_i = \frac{1}{|S_i|} \sum_{p \in S_i} LAB_p \quad (9)$$

where p denotes a pixel and $|S_i|$ denotes the number of pixels within S_i. The color distance between two superpixels is then given as

$$D^{\mathrm{C}}_{i,j} = \left\|\mu^{\mathrm{C}}_i - \mu^{\mathrm{C}}_j\right\|. \quad (10)$$

As objects may be salient due to their respective motion, we exploit their motion similarity as a second affinity cue. To achieve this, we define a pairwise motion trajectory $T_{i,j}$ between S_i and S_j by concatenating the centroids $C_{i,k}$ of S_i for each frame k in which it co-occurs with S_j as

$$T_{i,j} = \mathrm{cat}\left(C_{i,k}\right), \forall k : O_{k,i,j} = 1 \quad (11)$$

where $\mathrm{cat}(\cdot)$ is the concatenation operator. We shift the origins of the trajectories $T_{i,j}$ and $T_{j,i}$ to $(0, 0)$ to reduce the bias of intra-frame centroid differences, i.e. we focus on the distance of the trajectory shape. Then, we define the motion dissimilarity between two superpixels as

$$D^{\mathrm{M}}_{i,j} = \frac{\|T_{i,j} - T_{j,i}\|}{\sum_k O_{k,i,j}}. \quad (12)$$

Besides color and motion, we additionally exploit the joint lifespan length of two superpixels as an affinity measure. Normalizing this measure by the lifespan of the respective superpixel, we define the repetition affinity

$$R_{i,j} = \frac{\sum_k O_{k,i,j}}{\sum_k O_{k,i}} \quad (13)$$

where $O_{k,i}$ indicates if superpixel S_i exists at frame k (i.e. $O_{k,i} = 1$) or not (i.e. $O_{k,i} = 0$). Note that this affinity measure is only computed for video neighbors, as other superpixels will not be connected within the video graph. Thus, if both superpixels S_i and S_j appear and disappear at the same frame they are tightly connected. Also, if S_i disappears very early compared to S_j, it is still tightly connected to S_j but not vice-versa, i.e. the superpixel with the longer lifespan has a higher confidence of being part of the foreground region than the other one (which might be a dynamically changing object part or even noise).

Finally, we expect a salient region within a video to stand out in motion information in a global manner as well. To model this global motion saliency, we propose the following simple, yet efficient measure. Salient superpixels (superpixels belonging to salient regions) have a lot more frame-wise neighbors throughout the video than non-salient ones, as their neighbors change frequently due to the motion contrast. Hence, we consider the number of the total frame-wise neighbors of superpixel S_i throughout the video, normalized by the superpixel lifespan, as a measure of global motion saliency. The reason for the normalization is to prevent over-emphasizing superpixels with long lifespan or suppressing short ones, as some temporal superpixels within the objects may disappear due to sudden appearance changes. More formally, the global motion contrast measure is given as

$$G^{\mathrm{M}}_i = \frac{\left|\bigcup\limits_{k \in V} \mathcal{N}^F_{k,i}\right|}{\sum_k O_{k,i}}. \quad (14)$$

Consider two superpixels with the same lifespan, where one belongs to a static background region and the other is on the boundary of a moving object. The latter superpixel would have a higher number of unique neighbors throughout its lifespan as its immediate background keeps changing, whereas the former (static background) superpixel has only a few unique neighbors as it is within a static background. Therefore, (14) would yield a much higher value for the object boundary superpixel highlighting its motion distinctiveness which is desirable. Now consider two superpixels which are both on the boundary of the same moving object, with one having a shorter lifespan than the other, e.g. due to occlusion of that object part. These two superpixels should have the same global motion contrast measure since they are representing the boundary of the same object. Thus, we normalize the global motion contrast by the lifespan of the superpixels.

In summary, we have four different sources of information for graph affinities, namely the color difference $D^{\mathrm{C}}_{i,j}$, the motion difference $D^{\mathrm{M}}_{i,j}$, the repetition affinity $R_{i,j}$, and the global contrast measure G^{M}_i. Combining the repetition affinity with the motion and color distance can be achieved as follows. First, color and motion differences are normalized to have a mean value of 1 within a video to ensure equal contribution of both terms. Then, we linearly combine the color and motion distances, normalized *w.r.t.* the repetition affinity, i.e. the normalized distance $D^{\mathrm{N}}_{i,j}$ decreases with larger repetition affinity

$$D^{\mathrm{N}}_{i,j} = \frac{D^{\mathrm{C}}_{i,j} + D^{\mathrm{M}}_{i,j}}{R_{i,j}}. \tag{15}$$

Integrating the global motion contrast measure G^{M}_i with the constructed measure so far is however, rather more difficult. In particular, G^{M}_i will be high for superpixels on the border of an object that is salient in motion, as these will have many frame-wise neighbors throughout the video. However, superpixels within the salient object will not have as many changing frame-wise neighbors. Hence, G^{M}_i should be used to enhance the distance between superpixels between the object boundaries and the background only, i.e. those regions where the normalized color distance is already high. To control the effect of G^{M}_i, we introduce the confidence measure

$$\xi_i = 1 - \exp\left(\frac{-(D^{\mathrm{N}}_{i,j})^2}{\sigma^2}\right) \tag{16}$$

where σ is a pre-defined penalization constant. Note that this type of non-linear penalization is commonly used in cut-based methods, e.g. [36], [38]. Finally combining all measures and inverting the distance for conversion to affinities, we obtain the pairwise affinity scores

$$w_{i,j} = \left(\left(D^{\mathrm{N}}_{i,j}\right)^2 \left(1 + \xi_i \cdot G^{\mathrm{M}}_i\right) + \epsilon\right)^{-1} \tag{17}$$

where ϵ is a small constant to prevent division by zero. Equation (17) adopts the same distance to affinity conversion function with [15], the square of the pairwise distances between graph nodes are simply inverted. The final pairwise distance is obtained by weighting the normalized distance $D^{\mathrm{N}}_{i,j}$ to include the contribution of the global motion contrast measure G^{M}_i. The weighting depends on the confidence ξ_i where the lowest confidence leads to using $\left(D^{\mathrm{N}}_{i,j}\right)^2$ as is and the highest confidence leads to doubling it.

C. Video Background Prior

We now turn our attention to the diagonal potential matrix formed by the background unaries u_i. [15] assumed that the image boundaries are definitely background and thus, they are assigned high unary background potentials to pixels on the image border. The unary potentials for the rest of the image was left 0, meaning that no prior background information is available inside the image. This approach could result in failure for videos, since it is common that objects enter the video at a later time step, leave the field-of-view earlier or touch the frame boundaries throughout the video. Therefore, we propose a robust background prior to prevent inaccurate assignments of background to salient regions on the video boundaries. The background prior is only assigned to regions which contribute to the dominant motion on the video boundary. The regions that are in contrast with the dominant motion on the other hand are not given any prior as they might represent salient regions. This is achieved as follows. First, we extract an approximate motion vector for each superpixel as

$$M_i = \frac{C_{i,\max(\mathcal{K}_i)} - C_{i,\min(\mathcal{K}_i)}}{\sum_k O_{k,i}} \tag{18}$$

where $\mathcal{K}_i = \{k : O_{k,i} = 1\}$ is the set of frames in which S_i is present.

We assume that there is dominant background motion information in the motion vectors of superpixels on the frame boundaries. We represent this motion vector set by $\mathcal{M}$ as follows:

$$\mathcal{M} = \left\{ M_i : \sum_k O_{k,i} > \tau_M, \exists k : S_i \in B_k \right\}. \tag{19}$$

B_k is the set of superpixels at frame k which are located on the frame boundaries. Superpixels with a lifespan shorter than τ_M frames are not considered, since they do not provide reliable information.

Next, in order to find the dominant motion on the video boundary, we perform k-means with N_{C} clusters on the set of boundary motion vectors $\mathcal{M}$. Within the N_{C} clusters, we search for the salient clusters which are assumed to have significantly less samples than the remaining clusters. To this end, we first sort the clusters in descending order according to their cardinality. The largest cluster is always considered as background due to the assumption that the majority of the superpixels at the video boundaries belong to the background. Consecutive clusters are added to the background only if they contain more elements than half the size of the previously added cluster. The remaining clusters are not considered as background, as they have a sufficient motion contrast compared to the majority of the superpixels on the video boundaries. Algorithm 1 summarizes this selection of the background superpixel set $\mathcal{B}$.

D. Spatiotemporal Hamiltonian Matrix

Given both the pairwise affinities and the background prior, we can now form the spatiotemporal Hamiltonian matrix $\mathbf{H}_{\mathrm{ST}}$

Algorithm 1: Background superpixel selection

```
Input: Boundary motion vectors $\mathcal{M}$
Output: Set of background superpixels $\mathcal{B}$
Obtain clusters $\mathcal{C}_i, i = 1, \ldots, N_{\mathrm{C}}$ of $\mathcal{M}$ using k-means
  and sort these in descending order according to their
  cardinality.
$\mathcal{B} = \mathcal{C}_1$
for $c = 2$ to $N_{\mathrm{C}}$ do
    if $|\mathcal{C}_c| \geq 0.5\,|\mathcal{C}_{c-1}|$ then
        $\mathcal{B} = \mathcal{B} \cup \mathcal{C}_c$
    else
        Exit the loop
```

Algorithm 2: Single scale Video Quantum Cuts

```
Input: Input video (RGB frames), number of used
  eigenvectors $N_\lambda$, superpixel granularity $N_{\mathrm{S}}$
Output: Saliency map $\mathcal{S}$
Extract temporal superpixels [35] for the input video
  with granularity $N_{\mathrm{S}}$.
Construct the Hamiltonian matrix $\mathbf{H}_{\mathrm{ST}}$ as in Eq. (20).
Solve multispectral QCUT for $\mathbf{H}_{\mathrm{ST}}$:
• Compute the eigenvectors $\psi_i$ of $\mathbf{H}_{\mathrm{ST}}$ with the smallest
  $N_\lambda$ eigenvalues $\lambda_i$.
• Compute the saliency map $\mathcal{S}$ by Eq. (22).
```

for the video as

$$\mathbf{H}_{\mathrm{ST}}(i,j) = \begin{cases} u_i + \sum_{k \in \mathcal{N}_i^V} w_{k,i} & \text{if } S_i = S_j \\ -w_{i,j} & \text{if } j \in \mathcal{N}_i^V \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

where $w_{i,j}$ is given by (17) and u_i is defined as

$$u_i = \begin{cases} \infty & \text{if } S_i \in \mathcal{B} \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

Next, we solve QCUT once for $\mathbf{H}_{\mathrm{ST}}$ in order to find a saliency map for the whole video. Thus, we call this method Video Quantum Cuts (VQCUT), as summarized in Algorithm 2. Due to various changes in the video, some superpixels on the salient object might disappear prematurely and some are born at a later time step, thus causing a fragmented temporal superpixel extraction. In such scenarios, the first eigenvector of $\mathbf{H}_{\mathrm{ST}}$ might not completely cover the whole object movement throughout the video. To overcome this limitation, we exploit a larger spectrum of N_λ small eigenvalues. Therefore, we combine the eigenvectors ψ_i by a confidence measure that is inversely proportional to their eigenvalues λ_i, to obtain the saliency map $\mathcal{S}$ as

$$\mathcal{S} = \sum_{i=1}^{N_\lambda} \frac{\psi_i \circ \psi_i}{\lambda_i}. \tag{22}$$

Note that these are the saliency scores on the superpixel level. Thus, to get per-frame saliency maps, we propagate the saliency score of each superpixel back to the corresponding image regions.

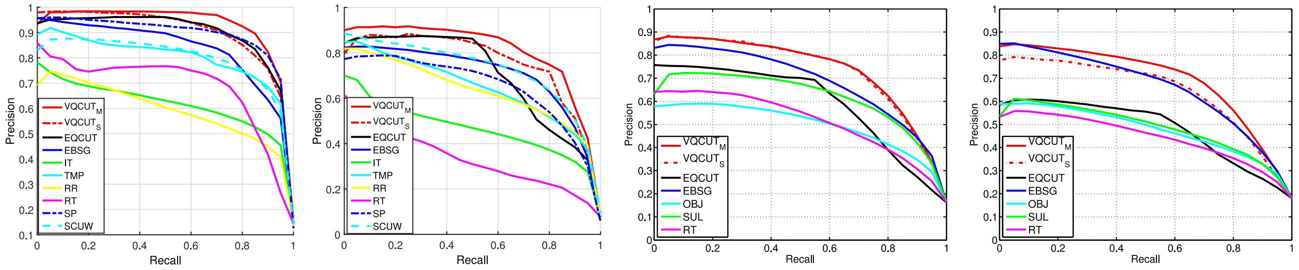
The proposed method not only provides a joint optimization of various saliency cues for salient object detection, it also addresses the shortcomings of temporal superpixel (TSP) extraction method as in the following cases. (i) TSPs are not robust to occlusions. This is particularly addressed by using several eigenvectors as explained in Section III-D. A first eigenvector may represent the salient object region before it gets occluded and another eigenvector may represent the same salient object after the occlusion. (ii) Due to noisy optical flow estimation, inaccurate shifts can occur in TSPs' positions. In our proposed graph construction, we only define TSPs as graph neighbors if they have stayed neighbors in each frame during their joint lifespan. This automatically disconnects the inaccurate TSPs whose locations are shifted incorrectly from the ones that are more reliable. (iii) In very dynamic scenes some TSPs may have extremely short lifespan and might be unreliable. The proposed asymmetric Repetition affinity introduced in Section III-B elegantly handles these unreliable TSPs by favoring the TSPs with longer lifespans as foreground regions. Moreover, the TSPs with short lifespans are also discarded during dominant boundary motion estimation as explained in Section III-C.

Finally, similar to [16], we observed that embedding QCUT in a multi-resolution framework yields improved results. Therefore, we extract the saliency maps for several scales (i.e. different levels of superpixel granularity) and combine the results by averaging the saliency maps. Our experiments demonstrate the performance gain from this multi-resolution approach, especially for scenarios where the object scale changes significantly. A higher superpixel resolution helps detecting relatively small objects, whereas a lower resolution proves useful to detect large ones.

IV. Experimental Results

We perform an extensive evaluation of our VQCUT on publicly available datasets. First, we provide a quantitative analysis on two video saliency datasets, namely Fukuchi [39] and Segtrack [40]. The Fukuchi dataset consists of 10 videos, capturing usually one salient object. Some videos are highly dynamic and contain both camera and object motion. The Segtrack dataset contains 14 videos and is more complex, as it captures more than one salient object within each video and includes severely cluttered backgrounds. We compare our approach against the state-of-the-art video saliency methods RT [30], RR [28], ITTI [29], TMP [31], SP [27], SCUW [32] and EBSGR [34].

Besides these standard video saliency tasks, we also apply VQCUT for activity localization where the goal is to localize activities within a video to enable unsupervised training of activity recognition algorithms. We follow the evaluation protocol of [34] and demonstrate the performance of VQCUT on two selective sports activity datasets: UCF Sports [41] and Olympic Sports [42]. The UCF Sports dataset contains 150 low-quality television broadcast videos of 10 different sports. This dataset depicts challenging scenarios, such as cluttered backgrounds, fast camera and object motion, and non-rigid object deformations. Additionally, we evaluate our approach on the Olympic



(a) (b) (c) (d)

Fig. 5. Recall-precision curves for VQCUT$_{\mathrm{M}}$ (multiscale), VQCUT$_{\mathrm{S}}$ (single scale), and competing methods on the video saliency datasets (a) Fukuchi [39] and (b) Segtrack [40] (compared to EQCUT [16], EBSGR [34], ITTI [29], TMP [31], RR [28], RT [30], SP [27], and SCUW [32]), as well as activity localization on (c) UCF Sports [41] and (d) Olympic Sports [42] (compared to EQCUT [16], EBSG [34], OBJ [43], SUL [44], and RT [30]).

TABLE I
Performance Metrics for the Video Saliency Task on (a) Fukuchi and (b) Segtrack, as Well as for the Activity Localization Task on (c) UCF Sports and (d) Olympic

	VQCUT$_{\mathrm{M}}$	VQCUT$_{\mathrm{S}}$	EQCUT	EBSG	ITTI	TMP	RR	RT	SP	SCUW
NSS	**2.47**	2.30	2.36	2.20	1.66	2.16	1.64	1.53	2.45	2.16
SIM	**0.75**	0.69	0.71	0.69	0.37	0.51	0.35	0.25	0.58	0.47
Mean F_1	**0.55**	0.54	0.54	0.52	0.45	0.52	0.44	0.47	**0.55**	0.52
Max F_1	**0.87**	0.83	0.84	0.78	0.65	0.78	0.62	0.72	0.86	0.78
$sAUC$	0.96	0.95	0.93	0.89	0.88	0.91	0.87	0.84	**0.97**	0.94

(a) Fukuchi.

	VQCUT$_{\mathrm{M}}$	VQCUT$_{\mathrm{S}}$	EQCUT	EBSG	ITTI	TMP	RR	RT	SP	SCUW
NSS	**3.36**	3.08	2.50	3.00	2.00	2.40	2.55	1.22	2.69	2.65
SIM	**0.59**	0.56	0.44	0.56	0.18	0.25	0.25	0.12	0.31	0.36
Mean F_1	**0.50**	0.49	0.45	0.47	0.37	0.45	0.44	0.29	0.46	0.48
Max F_1	**0.78**	0.76	0.66	0.71	0.52	0.64	0.63	0.38	0.67	0.71
$sAUC$	0.92	0.85	0.79	0.89	0.92	**0.94**	0.91	0.78	0.92	0.91

(b) Segtrack.

	VQCUT$_{\mathrm{M}}$	VQCUT$_{\mathrm{S}}$	EQCUT	EBSG	OBJ	SUL	RT
NSS	**1.45**	1.42	0.95	1.42	1.04	1.02	0.96
SIM	**0.42**	0.41	0.29	0.38	0.28	0.35	0.28
Mean F_1	**0.48**	**0.48**	0.42	0.46	0.41	0.45	0.40
Max F_1	**0.72**	0.71	0.62	0.66	0.56	0.65	0.55
$sAUC$	0.84	0.82	0.70	**0.85**	0.78	0.70	0.74

(c) UCF Sports.

	VQCUT$_{\mathrm{M}}$	VQCUT$_{\mathrm{S}}$	EQCUT	EBSG	OBJ	SUL	RT
NSS	**1.28**	1.13	0.69	**1.28**	0.92	0.77	0.71
SIM	**0.40**	0.37	0.26	**0.40**	0.28	0.27	0.26
Mean F_1	**0.47**	0.45	0.39	0.45	0.39	0.40	0.38
Max F_1	**0.69**	0.65	0.55	0.64	0.53	0.54	0.51
$sAUC$	**0.81**	0.74	0.67	0.76	0.77	0.69	0.72

(d) Olympic Sports.

Sports dataset[1] which contains 134 videos of 16 different sports. This dataset also covers several challenging scenarios, including large object scale variations. For both activity localization tasks, we compare VQCUT to the state-of-the-art EBSG [34], RT [30], SUL [44], and OBJ [43].

A. Parameter Settings

In all our experiments, we keep the parameters of VQCUT fixed. In particular, we use a temporal superpixel granularity of $N_{\mathrm{S}} = 800$ for our single scale experiments (denoted VQCUT$_{\mathrm{S}}$) and $N_{\mathrm{S}} = \{400, 600, 800\}$ for the different scales of our multi-resolution approach (denoted VQCUT$_{\mathrm{M}}$). The penalization constant is set to $\sigma = 0.1$. To estimate the background potentials, we use a minimum lifespan length for reliable boundary superpixels of $\tau_M = 10$ and obtain $N_{\mathrm{C}} = 5$ boundary motion vector clusters. To overcome the limitations due to fragmented temporal superpixels, we combine the saliency scores corresponding to the $N_\lambda = 10$ smallest eigenvectors. VQCUT has been shown to be quite robust to parameter selections. Detailed experiments for different parameter settings are provided in Appendix E.

B. Evaluation Protocol

We evaluate all approaches via recall-precision curves, mean and maximum F_1 score, normalized scanpath saliency (NSS) [45], the similarity metric (SIM) [46], and the shuffled AUC [47]. Details on the evaluation metrics are provided in Appendix A. Note that the video saliency datasets provide detailed binary segmentation masks as ground truth, whereas for the activity localization datasets only bounding box annotations are available. For the latter, we follow the box prior evaluation of [34] to allow for a fair comparison. To include OBJ [43], we follow the parametrization of [48] and take the top 100 proposals returned by the objectness detector to create a max-normalized saliency map per frame. Please note that VQCUT is completely unsupervised and requires no pre-training.

C. Comparison to the Baseline

The recall-precision curves for all evaluations are illustrated in Fig. 5, while the remaining performance metrics are summarized in Table I. Our results are denoted as VQCUT$_{\mathrm{S}}$ (single scale) and VQCUT$_{\mathrm{M}}$ (multi-resolution), respectively. As a first experiment, we compare VQCUT to the baseline, i.e. EQCUT [16], the multi-resolution extension of QCUT. From the results we clearly see the improvements over EQCUT by incorporating novel affinities covering spatiotemporal saliency cues.

[1] Bounding box annotations for the Olympic Sports dataset have been provided by the authors of [34].

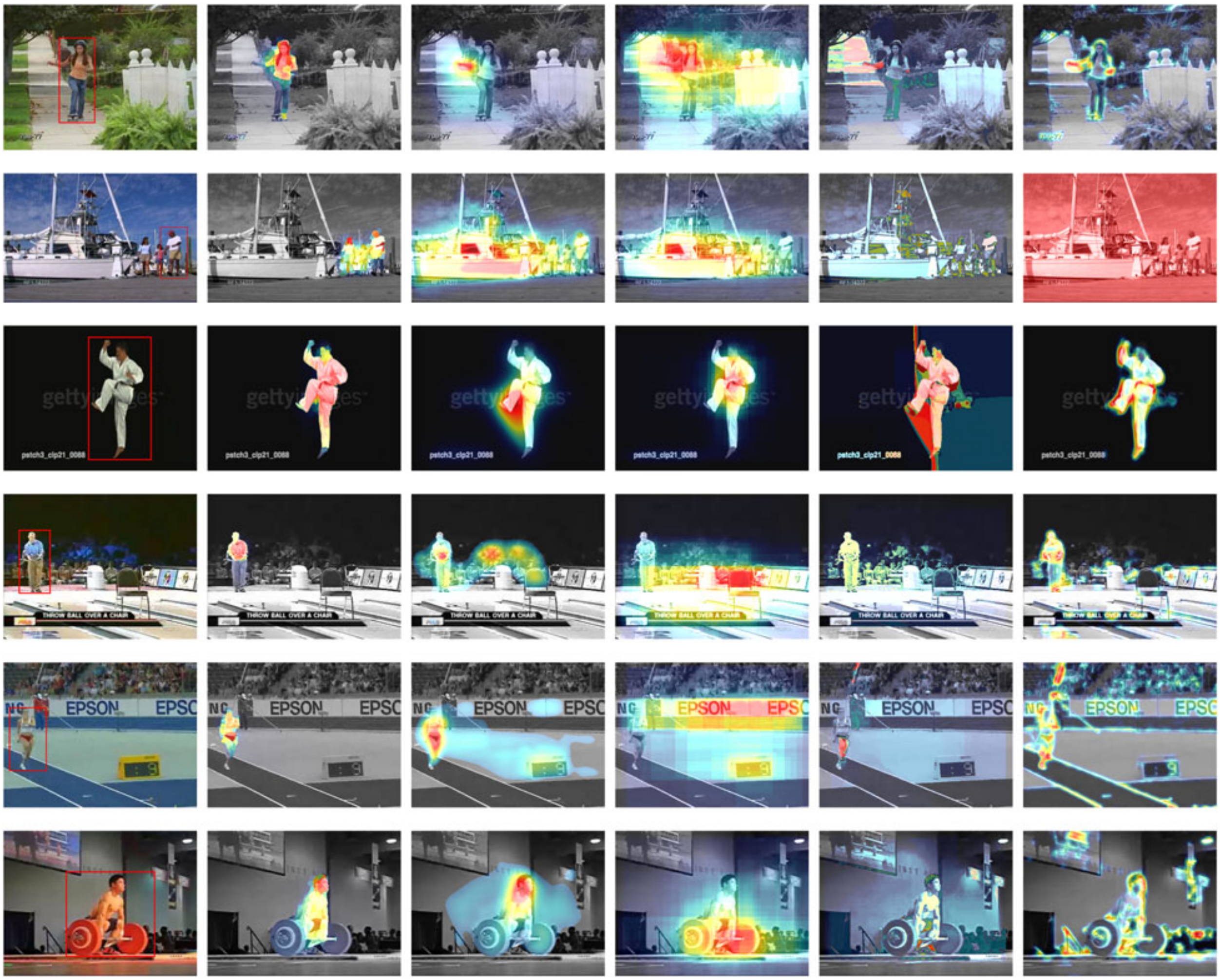


Fig. 6. Exemplary saliency results for activity localization on UCF Sports (top three rows) and Olympic Sports (bottom three rows). From left to right: input image with bounding-box ground truth annotation, saliency maps obtained by our VQCUT_{M}, EBSG [34], OBJ [43], RT [30], and SUL [44].

Furthermore, compared to [34] which separately fuses motion and appearance saliency maps, we see that our unified approach which simultaneously exploits motion and appearance in a joint optimization step achieves a notable performance improvement.

D. Multiresolution Benefits

Applying VQCUT on graphs formed for different levels of superpixel granularities and combining the saliency scores (VQCUT_{M}) also yields a notable improvement compared to the single scale version (VQCUT_{S}). The only exception is the UCF Sports dataset, where both VQCUT_{M} and VQCUT_{S} perform on par. This is due to the limited amount of scale variation within this dataset. For all other datasets, where the object scale varies much more, VQCUT_{M} yields significantly more robust results.

E. Comparison to the State-of-the-Art

Our evaluations demonstrate the favorable performance of both VQCUT variants (i.e. single scale and multi-resolution) compared to the state-of-the-art for both, standard video saliency and activity localization tasks. Except for the very similar performances of VQCUT_{M} and EBSG in terms of NSS and SIM measures on the Olympic Sports dataset, VQCUT_{M} achieves the top performance, usually with a notable gap to the competitors. Fig. 6 illustrates the performance of VQCUT_{M} and competitors for several different sports on the UCF Sports and Olympic Sports datasets. A visual comparison to the state-of-the-art for the salient object detection task on the Fuckuchi and Segtrack datasets is provided in Fig. 7.

F. Computational Complexity

An unoptimized Matlab code for the proposed method takes around 0.78 seconds per frame for the multiresolution variant and 0.35 seconds per frame for the single resolution variant on an Intel Core i7-3740QCM CPU@2.70 GHz. The computational complexity comparison in Table II illustrates that the proposed method is faster than the competing methods. Note that the reported run time does not include temporal superpixel extraction time. Temporal superpixel extraction takes around 11.21 seconds for the single resolution variant and 23.44 second for the multi-resolution variant. In future, similar faster methods can be employed in order to speed up the whole method.

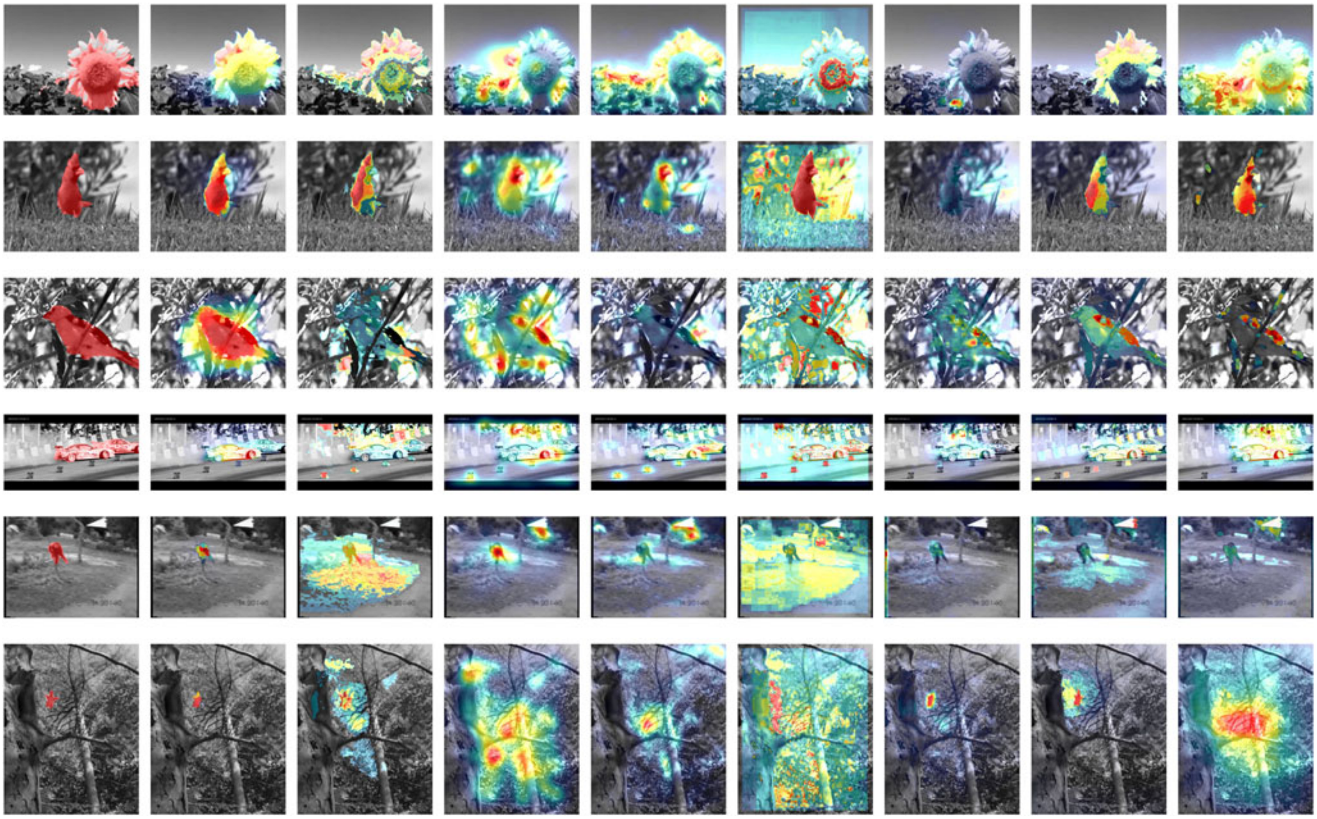


Fig. 7. Exemplary saliency results for spatiotemporal salient object detection on Fukuchi (top three rows) and Segtrack (bottom three rows). From left to right: input image with ground truth annotation, saliency maps obtained by our VQCUT_M, EBSG [34], ITTI [29], RR [28], RT [30], SCUW [32], SP [27], and TMP [31].

TABLE II
COMPUTATIONAL COMPLEXITY OF COMPETING METHODS

	VQCUT_M	VQCUT_S	EQCUT	EBSG	ITTI	TMP	RR	RT	SP	SCUW	OBJ	SUL
Time (sec)	0.78	0.35	0.85	2.06	NA	NA	0.94	8.01	11.85	26.90	3.14	0.41

V. Conclusion

We proposed a spatiotemporal saliency detection method for videos by combining local color and motion contrast cues, a global motion contrast cue, as well as shape and background cues all within a single joint optimization problem. Our experiments on several public datasets demonstrate the benefits of this joint optimization of the saliency cues. Moreover, both single scale and multi-resolution variants of Video Quantum Cuts perform favorably to the state-of-the-art, even for complex tasks, such as activity localization. As a future work, we plan to run the proposed algorithm on video partitions separately rather than the whole video at once, in order to handle failure cases due to change of the concept of saliency with changing video content.

Appendix

In the following, we provide additional explanations and visualizations. Appendix A summarizes the applied evaluation metrics. Appendix B visualizes the cut cost distributions resulting from the proposed affinity measures. Appendix C analyzes the effect of each part of the proposed spatiotemporal affinity measure. Finally, Appendix D presents a baseline experiment against Normalized Cuts and Appendix E demonstrates the robustness to varying parameter settings.

A. Evaluation Metrics

As stated in the main paper, we evaluate all approaches via recall-precision curves, mean and maximum F_1 score, normalized scanpath saliency (NSS) [45], the similarity metric (SIM) [46], and the shuffled AUC [47]. The precision, recall, and F_1 measures are defined as

$$\text{pre}(\tau) = \frac{|\mathcal{G} \cap \mathcal{S}_\tau|}{|\mathcal{S}_\tau|}, \qquad \text{rec}(\tau) = \frac{|\mathcal{G} \cap \mathcal{S}_\tau|}{|\mathcal{G}|} \tag{23}$$

$$F_1(\tau) = 2\frac{\text{pre}(\tau)\,\text{rec}(\tau)}{\text{pre}(\tau) + \text{rec}(\tau)} \tag{24}$$

respectively, where $\mathcal{G}$ is the binary ground truth mask and $\mathcal{S}_\tau$ is the binary segmentation obtained via thresholding the saliency mask $\mathcal{S}$ at confidence τ.

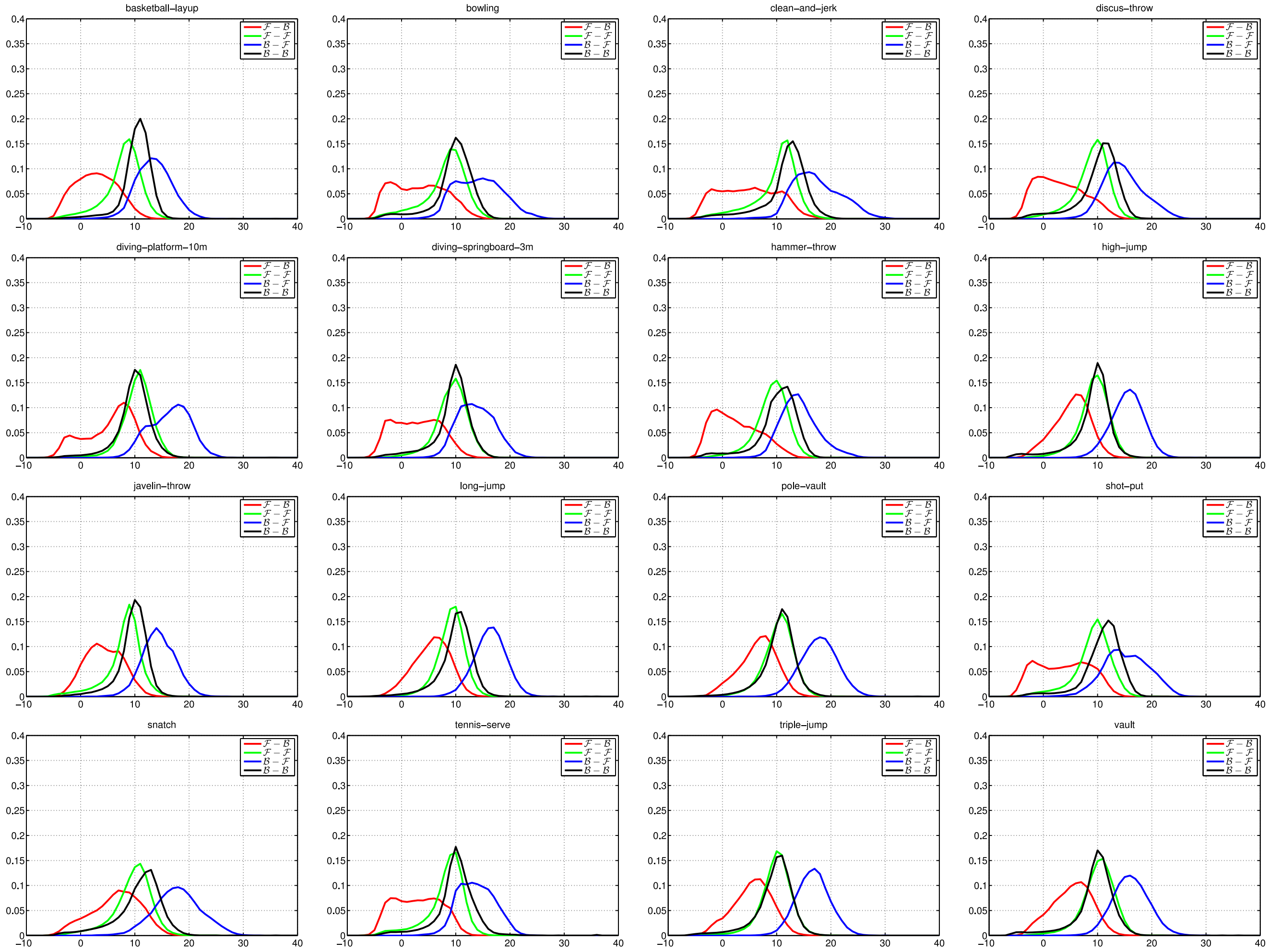


Fig. 8. Log-affinity histogram plots of the cut costs on the 16 sports categories of the Olympic Sports Dataset.

The normalized scanpath saliency is defined as

$$\text{NSS} = \frac{1}{N_{\text{G}}}\sum_{p=1}^{N_{\text{G}}}\frac{\mathcal{S}(p)-\mu_{\mathcal{S}}}{\sigma_{\mathcal{S}}} \tag{25}$$

where $\mathcal{S}(p)$ is the value of the evaluated saliency map at the ground truth point p and N_{G} is the total number of ground truth points. For this, the saliency map is normalized to have zero mean and unit standard deviation.

The similarity metric is defined as

$$\text{SIM} = \sum_{x=1}^{X}\min\left(\mathcal{S}(x),\mathcal{G}(x)\right) \tag{26}$$

where $\mathcal{S}(x)$ is the normalized probability distribution of the saliency map at point x and $\mathcal{G}(x)$ is the normalized probability distribution of the ground truth at point x.

The area under the curve (AUC) metric is defined as area under the receiver operating characteristics curve. The shuffled AUC [47] (sAUC) uses samples from other saliency maps instead of the tested one in order to select the random threshold values to form the AUC curve.

These measures can be readily applied for video saliency datasets, where binary ground truth segmentations are available. For the activity localization datasets, however, only coarse bounding box annotations are available. To allow for a fair comparison, we follow the box prior evaluation in [34], i.e. the thresholded saliency masks $\mathcal{S}_\tau$ are filled with spanning bounding boxes before computing the recall and precision values. For more details, please refer to [34].

B. Cut Cost Distributions

The proposed spatiotemporal Hamiltonian matrix is assymmetric, i.e. $w_{i,j}$ is the cut cost if S_j is assigned foreground and S_i is assigned background, whereas $w_{j,i}$ is the cut-cost if S_i is assigned foreground and S_j is assigned background. Fig. 8 illustrates the cut cost distribution for the cases:

1) Both S_j and S_i are foreground (denoted as $\mathcal{F}$–$\mathcal{F}$).
2) Both S_j and S_i are background ($\mathcal{B}$–$\mathcal{B}$).
3) S_j is foreground and S_i is background ($\mathcal{F}$–$\mathcal{B}$).
4 S_i is foreground and S_j is background ($\mathcal{B}$–$\mathcal{F}$).

One would expect the $\mathcal{F}$-$\mathcal{B}$ edges to have the minimum affinities since these are the desired object boundaries to be cut. On the other hand, we should avoid to cut $\mathcal{B}$-$\mathcal{F}$ edges, which means assigning a foreground node to the wrong partition. Additionally, we would expect high affinities for both $\mathcal{F}$–$\mathcal{F}$ and $\mathcal{B}$–$\mathcal{B}$

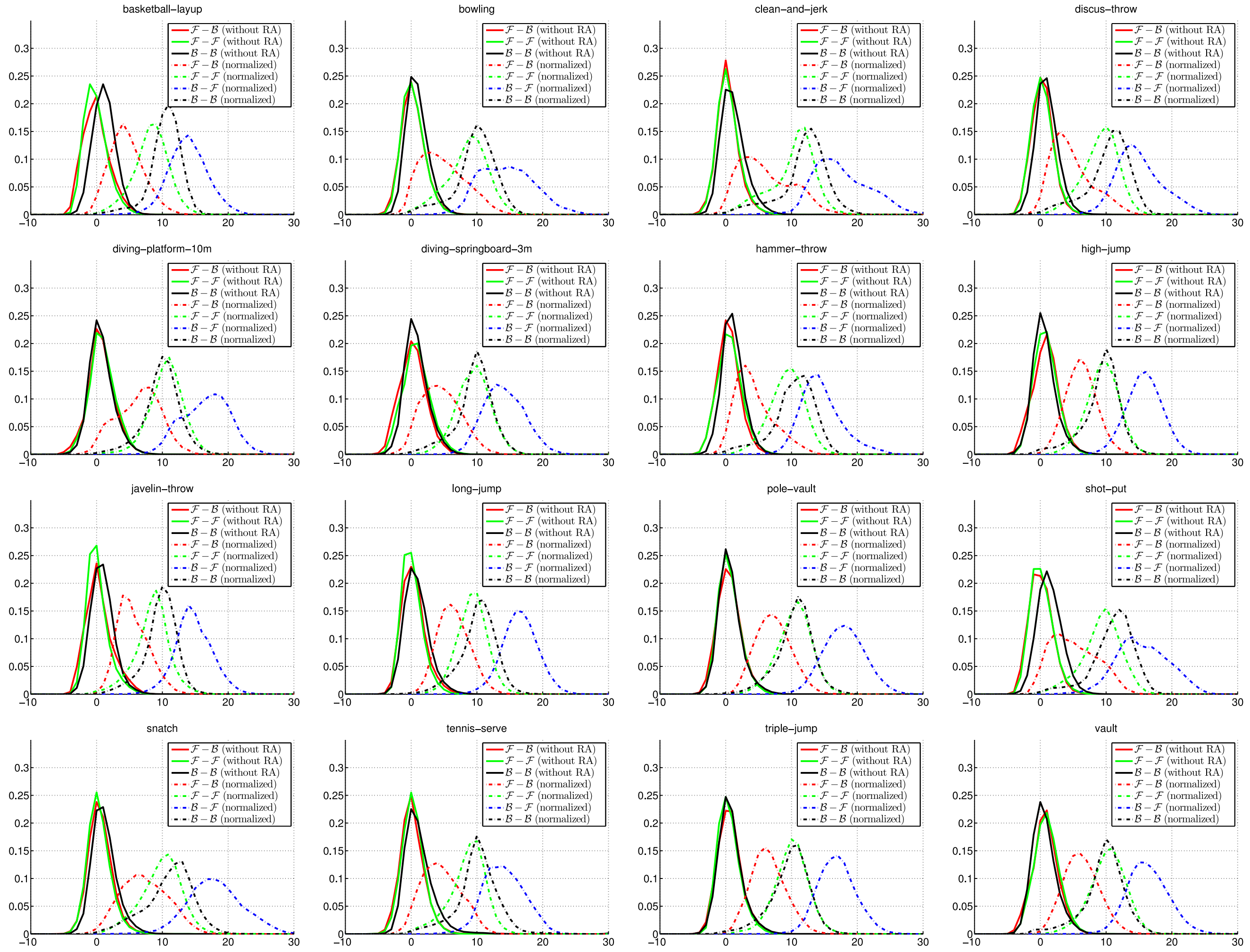


Fig. 9. Log-affinity histogram plots to demonstrate the effect of normalizing the combined color and motion affinity measure by the repetition affinity. See text for details.

edges. However, due to color differences within the salient objects and noisy superpixel segmentations these affinities can get damped. Nevertheless, as long as these affinities are higher than for $\mathcal{F}$–$\mathcal{B}$ edges, we find a correct cut. From Fig. 8 we can see that the proposed spatiotemporal affinity measure can seperate $\mathcal{F}$–$\mathcal{B}$ edges quite nicely from $\mathcal{B}$–$\mathcal{F}$ edges while preserving high values for both $\mathcal{F}$–$\mathcal{F}$ and $\mathcal{B}$–$\mathcal{B}$ edges.

C. Repetition Affinity and Global Motion Contrast

Next, we analyze the effect of each part of the proposed spatiotemporal affinity measure. To this end, we first show the benefits of normalizing color and motion cues by the corresponding repetition affinity in Fig. 9. Note that before normalizing the color and motion cues by the repetition affinity, the affinity assignment is symmetric, i.e. there is no difference between $\mathcal{F}$–$\mathcal{B}$ and $\mathcal{B}$–$\mathcal{F}$ connections (and hence, $\mathcal{F}$–$\mathcal{B}$ = $\mathcal{B}$–$\mathcal{F}$ in Fig. 9). For a complex dataset, such as Olympic Sports, color and motion cues are not discriminative enough to enhance the boundaries on the salient object border. Using the repetition affinity to normalize both color and motion cues, we can distinguish between object, background, and border regions. Similarly, by adding the global motion contrast cue (see Fig. 10) we can suppress edge weights on the object borders even more, while both object and background affinities stay high.

D. Quantum Cuts Versus Normalized Cuts

In this additional experiment, we compare Quantum Cuts (QCUT) [15] with Normalized Cuts (NCUT) [36]. NCUT clusters the image into two segments without providing any information to distinguish background from foreground. Thus, we assume that there is an oracle which correctly predicts which segment should be selected as foreground. We perform the comparison for both a *frame-wise oracle* (i.e. deciding for each frame separately which partition is foreground) and a *video oracle* (i.e. deciding once for the whole video). The results of both oracles are very similar, since saliency scores are backprojected to the corresponding superpixel regions and thus, no significant label change occurs throughout the whole video.

We evaluate both cut methods on the Segtrack dataset using a single superpixel granularity of $N_\mathrm{S} = 800$ and consider

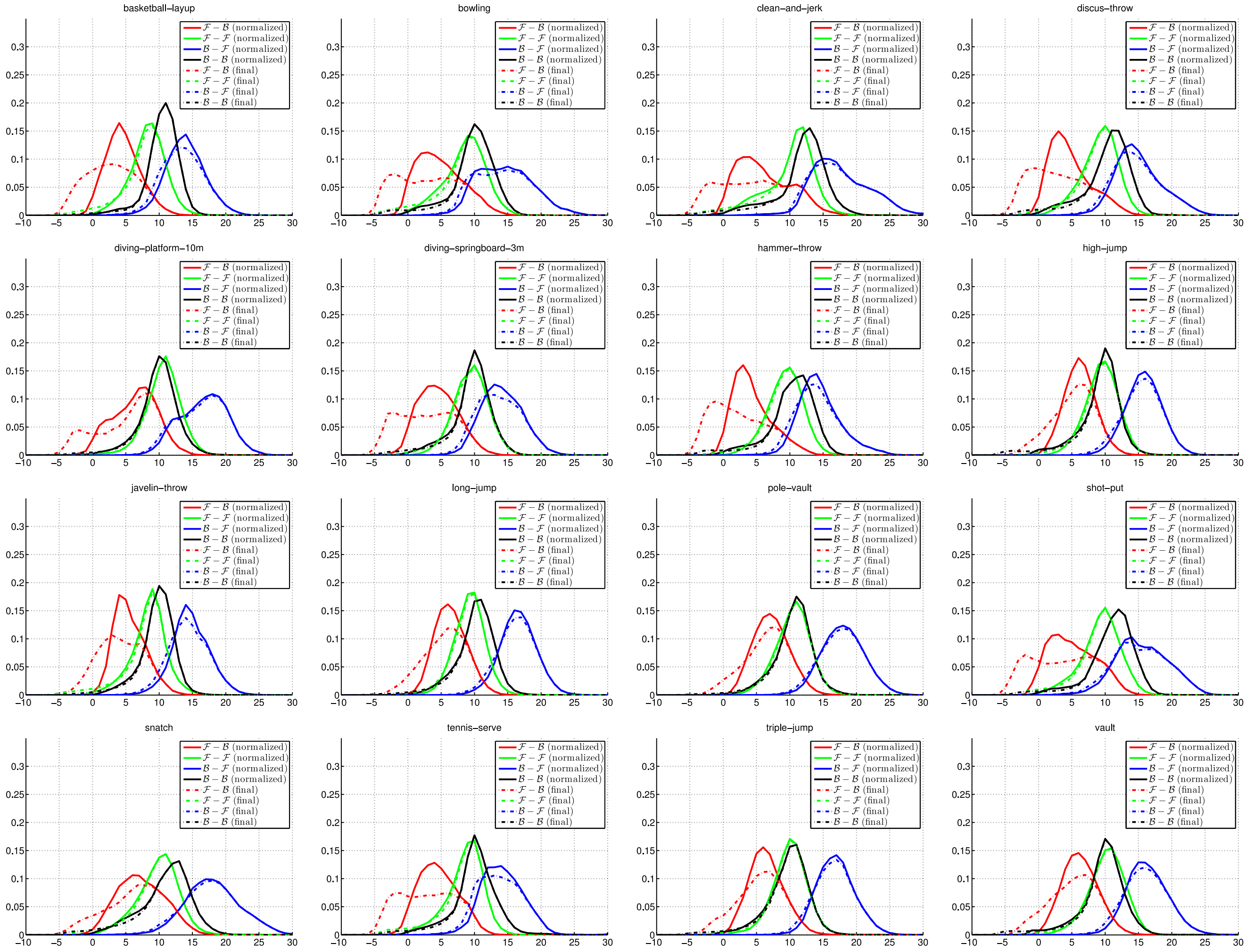


Fig. 10. Log-affinity histogram plots to demonstrate the effect of the global motion contrast cue. See text for details.

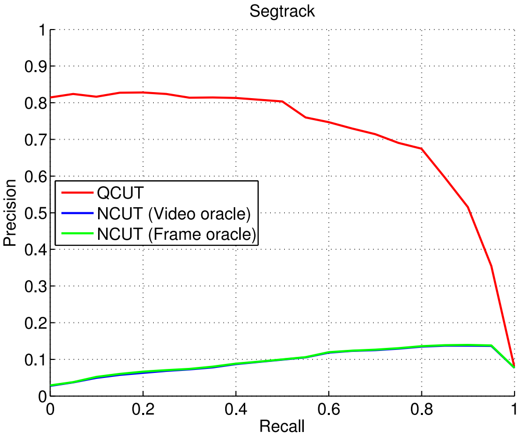


Fig. 11. Comparison of QCUT and NCUT on the same Hamiltonian $\mathbf{H}_{\mathrm{ST}}$.

only the first eigenvector to reconstruct the saliency maps. Both QCUT and NCUT are given the same spatiotemporal affinity matrix $\mathbf{H}_{\mathrm{ST}}$. From the results (see Fig. 11 and Table III) we can see that NCUT is not suitable for saliency estimation. This is mainly due to the fact that NCUT was designed as an image segmentation approach and is only applicable for related tasks.

TABLE III
COMPARISON OF QCUT AGAINST NCUT

	Max F_1	Mean F_1	NSS	SIM
QCUT	**0.73**	**0.48**	**3.07**	**0.52**
NCUT (Frame Oracle)	0.24	0.15	0.48	0.09
NCUT (Video Oracle)	0.24	0.15	0.44	0.09

We can see this from the minimization criterion of NCUT. Although it reduces the cut cost, it also tries to increase the color constancy within each segment via an association term. While this is useful for general image segmentation, it fails for salient object segmentation since object segments often have varying colors (e.g. different trouser and jacket color). Additionally, it is not possible to assign a background prior within NCUT, as there is no such definition as “background”.

In contrast to NCUT, QCUT can assign a background prior, which can be obtained in an unsupervised manner. Furthermore, the minimization criteria of QCUT focuses solely on the foreground, whereas NCUT is tailored for image segmentation. Thus, QCUT should be favored for saliency estimation when compared to NCUT or Graph Cut.

TABLE IV
PARAMETER ANALYSIS

N_λ	NSS	SIM	Max F_1	Mean F_1	sAUC
5	3.37	0.60	0.79	0.51	0.92
10	3.36	0.59	0.78	0.50	0.92
15	3.36	0.58	0.78	0.50	0.92

(a) Varying the number of eigenvectors N_λ.

σ	NSS	SIM	Max F_1	Mean F_1	sAUC
0.05	3.35	0.59	0.78	0.50	0.92
0.10	3.36	0.59	0.78	0.50	0.92
0.20	3.36	0.59	0.78	0.50	0.92

(b) Varying the penalization constant σ.

τ_M	NSS	SIM	Max F_1	Mean F_1	sAUC
5	3.36	0.59	0.78	0.50	0.92
10	3.36	0.59	0.78	0.50	0.92
15	3.37	0.59	0.78	0.50	0.92

(c) Varying the lifespan threshold τ_M.

N_C	NSS	SIM	Max F_1	Mean F_1	sAUC
3	3.36	0.59	0.78	0.50	0.92
5	3.36	0.59	0.78	0.50	0.92
7	3.35	0.59	0.78	0.50	0.92

(d) Varying the number of k-means clusters N_C.

Note that we cannot conduct an experiment comparing QCUT to Graph Cut [38], as the latter is an interactive method which requires both foreground and background seeds. This is due to the fact that Graph Cut minimizes the cut only and if there is no unary cost of not assigning a region as foreground (i.e. if no foreground seed is given) the best cut is no cut at all. Instead, everything is labelled as background. Our experimental results confirmed that without foreground seeds, Graph-Cut labels every region as background. QCUT on the other hand, always produces a foreground segment (i.e. performs a cut), due to the additional object term which maximizes the foreground segment area. Hence, QCUT can be used to extract foreground regions in an unsupervised manner.

E. Robustness to Parameters

We analyzed the robustness of our method *w.r.t.* its parameters (i.e. σ, τ_M, N_λ and N_C) on the Segtrack dataset. For each experiment, one parameters was changed and the remaining parameters were fixed to the values stated in Section IV. As can be seen from Table IV, VQCUT is quite robust to variations of its parameters. Thus, VQCUT can be applied to a variety of tasks without tedious parameter fine tuning.

REFERENCES

[1] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[2] W.-H. Chen, C.-W. Wang, and J.-L. Wu, "Video adaptation for small display based on content recomposition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 1, pp. 43–58, Jan. 2007.

[3] E. Ekmekcioglu, V. V. D. Silva, P. T. Pesch, and A. Kondoz, "Visual attention model aided non-uniform asymmetric coding of stereoscopic video," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 3, pp. 402–414, Jun. 2014.

[4] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," in *Proc. ACM SIGGRAPH*, 2008, Art. no. 16.

[5] T. Nguyen *et al.*, "Image re-attentionizing," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1910–1919, Dec. 2013.

[6] L. Zhang *et al.*, "Retargeting semantically-rich photos," *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1538–1549, Sep. 2015.

[7] H. Hadizadeh and I. V. Bajic, "Saliency-aware video compression," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 19–33, Jan. 2014.

[8] D. Gao and N. Vasconcelos, "Integrated learning of saliency, complex features, and object detectors from cluttered scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 2, pp. 282–287.

[9] D. Rudoy, D. B. Goldman, E. Shechtman, and L. Zelnik-Manor, "Learning video saliency from human gaze using candidate selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 1147–1154.

[10] Y. Tong, F. A. Cheikh, F. F. E. Guraya, H. Konik, and A. Tremeau, "A spatiotemporal saliency model for video surveillance," *Cogn. Comput.*, vol. 3, no. 1, pp. 241–263, 2011.

[11] T. V. Nguyen, Z. Song, and S. Yan, "STAP: Spatial-temporal attention-aware pooling for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 77–86, Jan. 2015.

[12] D. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2214–2219.

[13] M.-M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang, and S. M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 409–416.

[14] H. Jiang, J. Wang, Z. Yuan, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior," in *Proc. Brit. Mach. Vis. Conf.*, 2011, pp. 1–12.

[15] Ç. Aytekin, S. Kiranyaz, and M. Gabbouj, "Automatic object segmentation by quantum cuts," in *Proc. IEEE Int. Conf. Pattern Recog.*, Aug. 2014, pp. 112–117.

[16] Ç. Aytekin, E. Ozan, S. Kiranyaz, and M. Gabbouj, "Visual saliency by extended quantum cuts," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2015, pp. 1692–1696.

[17] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 29–42.

[18] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 2814–2821.

[19] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 3166–3173.

[20] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing Markov chain," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 1665–1672.

[21] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 733–740.

[22] L. Wen, D. Du, Z. Lei, S. Z. Li, and M.-H. Yang, "JOTS: Joint online tracking and segmentation," in *Proc. IEEE Conf. Conf. Comput. Vis. Pattern Recog.*, Jun. 2015, pp. 2226–2234.

[23] F. Xiao and Y. J. Lee, "Track and segment: An iterative unsupervised approach for video object proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2016, pp. 933–942.

[24] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 1777–1784.

[25] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 628–635.

[26] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185–198, Jan. 2010.

[27] Z. Liu, X. Zhang, S. Luo, and O. L. Meur, "Superpixel-based spatiotemporal saliency detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1522–1540, Sep. 2014.

[28] M. Mancas, N. Riche, J. Leroy, and B. Gosselin, "Abnormal motion selection in crowds using bottom-up saliency," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 229–232.

[29] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," in *Proc. IEEE Comput. Sci. Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 1, pp. 631–637.

[30] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä, "Segmenting salient objects from images and videos," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 366–379.

[31] A. Singh, C.-H. H. Chu, and M. A. Pratt, “Learning to predict video saliency using temporal superpixels,” in *Proc. Int. Conf. Pattern Recog. Appl. Methods*, 2015, pp. 201–209.

[32] Y. Fang, Z. Wang, W. Lin, and Z. Fang, “Video saliency incorporating spatiotemporal cues and uncertainity weighting,” *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3910–3921, Sep. 2014.

[33] J. Zhao, C. Siagian, and L. Itti, “Fixation bank: Learning to reweight fixation candidates,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2015, pp. 3174–3182.

[34] T. Mauthner, H. Possegger, G. Waltner, and H. Bischof, “Encoding based saliency detection for videos and images,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2015, pp. 2494–2502.

[35] J. Chang, D. Wei, and J. W. Fisher, “A video representation using temporal superpixels,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2051–2058.

[36] J. Shi and J. Malik, “Normalized cuts and image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[37] R. L. Liboff, *Introductory Quantum Mechanics*, 4th ed. Reading, MA, USA: Addison-Wesley, 2013.

[38] Y. Y. Boykov and M.-P. Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images,” in *Proc. Int. Conf. Comput. Vis.*, 2001, pp. 105–112.

[39] K. Fukuchi, K. Miyazato, A. Kimura, S. Takagi, and J. Yamato, “Saliency-based video segmentation with graph cuts and sequentially updated priors,” in *Proc. IEEE Int. Conf. Multimedia Expo*, Jun.–Jul. 2009, pp. 638–641.

[40] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg, “Motion coherent tracking using multi-label MRF optimization,” *Int. J. Comput. Vis.*, vol. 100, no. 2, pp. 190–202, 2012.

[41] M. D. Rodriguez, J. Ahmed, and M. Shah, “Action MACH—A spatio-temporal maximum average correlation height filter for action recognition,” in *Proc. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[42] J. C. Niebles, C.-W. Chen, and L. Fei-Fei, “Modeling temporal structure of decomposable motion segments for activity classification,” in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 392–405.

[43] B. Alexe, T. Deselaers, and V. Ferrari, “What is an object?” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 73–80.

[44] W. Sultani and I. Saleemi, “Human action recognition across datasets by foreground-weighted histogram decomposition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 764–771.

[45] R. J. Peters, A. Iyer, L. Itti, and C. Koch, “Components of bottom-up gaze allocation in natural images,” *Vis. Res.*, vol. 45, no. 18, pp. 2397–2416, 2005.

[46] T. Judd, F. Durand, and A. Torralba, “A benchmark of computational models of saliency to predict human fixations,” MIT Comput. Sci. Artif. Intell. Lab., Cambridge, MA, USA, Tech. Rep. MIT-CSAIL-TR-2012–001, 2012.

[47] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, “SUN: A Bayesian framework for saliency using natural statistics,” *J. Vis.*, vol. 8, no. 7, pp. 1–20, 2008.

[48] N. Shapovalova, A. Vahdat, K. Cannons, T. Lan, and G. Mori, “Similarity constrained latent support vector machine: An application to weakly supervised action classification,” in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 55–68.

Authors’ photographs and biographies not available at the time of publication.