Min-cut Max Flow Optimization in Markov Random Field for Automatic Primary and Unconstrained Video Object Segmentation

G. Nithiya and P. Vijayakumar

Abstract--- Automatic segmentation of the primary object in a video clip is a challenging problem as there is no prior knowledge of the primary object. Most existing techniques thus adapt an iterative approach for foreground and background appearance modeling. However, these approaches may rely on good initialization and can be easily trapped in local optimal. In addition, they are usually time consuming and difficult for analyzing videos. To solve this problem, recent work introduces a new approach for automatic primary video object segmentation. The input is a plain video clip without any annotations and the output is a pixel-wise spatio-temporal foreground vs. background segmentation of the entire sequence. However how to handling essentially unconstrained settings, becomes very difficult task by using automatic primary video object segmentation based on Markov Random Field (MRF). So in this work proposed new primary video object segmentation by following Min-Cut Max Flow in MRF (MCMF-MRF) .This work present a MCMF-MRF technique for separating foreground objects from the background in a video. MCMF-MRF method is fast, fully automatic, and makes minimal assumptions about the video. This enables handling essentially unconstrained settings, including rapidly moving background, arbitrary object motion and appearance, and non-rigid deformations and articulations. Similar too many existing image and video object segmentation approaches, we cast the segmentation to a two-class node labeling problem in a MCMF-MRF. Within the MRF graph, each node is modeled as a super pixel, and will be labeled as either foreground or background in the segmentation process. It embeds the appearance constraint as auxiliary nodes and edges in the MCMF-MRF structure, and can optimize both the segmentation and appearance model parameters simultaneously in one MCMF. The extensive experimental evaluations validate the superiority of the proposed MCMF-MRF structure over the state-of-the-art methods, in both efficiency and effectiveness.

Index Terms--- Automatic, Primary, Video, Object, Segmentation, Graph Cut, Appearance Modeling, Min-Cut Max Flow in MRF (MCMF-MRF).

I. INTRODUCTION

Video object segmentation is a well-researched problem in the computer vision community and is a prerequisite for a variety of high-level vision applications, including content based video retrieval, video summarization, activity understanding and targeted content replacement. Both fully automatic methods and methods requiring manual initialization have been proposed for video object segmentation. In the latter class of approaches, [1] need annotations of object segments in key frames for initialization.

The first row shows frames from a video. The second row shows key object proposals (in red boundaries) extracted by [2]. "?" indicates that no proposal related to the primary object was found by the method. The third row shows primary object proposals selected by the proposed method. Note that the proposed method was able to find primary object proposals in all frames. The results in row 2

G. Nithiya, Research Scholar, Department of Computer Science, Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science, Coimbatore. E-mail:nithiya.nithu1@gmail.com

P. Vijayakumar, Head, Department of Computer Application, Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science, Coimbatore. E-mail:vijayvigash@gmail.com

and 3 are prior to per-pixel segmentation. In this paper we demonstrate that temporally dense extraction of primary object proposals results in significant improvement in object segmentation performance.

Optimization techniques employing motion and appearance constraints are then used to propagate the segments to all frames. Other methods ([3]) only require accurate object region annotation for the first frame, then employ region tracking to segment the rest of frames into object and background regions. Note that, the aforementioned semi-automatic techniques generally give good segmentation results. However, most computer vision applications involve processing of large amounts of video data, which makes manual initialization cost prohibitive. Consequently, a large number of automatic methods have also been proposed for video object segmentation. A subset of these methods employs motion grouping [4] for object segmentation. Other methods [5] use appearance cues to segment each frame first and then use both appearance and motion constraints for a bottom-up final segmentation. Methods based on efficient optimization frameworks for spatiotemporal grouping of pixels for video segmentation. However, all of these automatic methods do not have an explicit model of how an object looks or moves, and therefore, the segments usually don't correspond to a particular object but only to image regions that exhibit coherent appearance or motion.

The Primary object in a video sequence can be defined as the object that is locally salient and present in most of the frames [6]. The target of automatic primary video object segmentation is to segment out the primary object in a video sequence without any human intervention. It has a wide range of applications including video object recognition, action recognition and video summarization. Some examples are shown in Figure 1. The existing works on video object segmentation has been addressed by methods requiring a user to annotate the object position in some frames [7], and by fully automatic methods [8], which input just the video.

Figure 1: Illustration of Primary Object Segmentation in Videos

The top row is the original video frames with the expected segmentation results rendered as red contours. The bottom row is the same segmentation results after removing the background. The method belongs to the latter and does not assume the object is present in all the frames. Following the outstanding performance of Markov Random Field (MRF) based methods in image object segmentation [9], many of the existing video object segmentation approaches also build spatio-temporal MRF graphs and show promising results [10]. These approaches build a spatio-temporal graph by connecting spatially or temporally connected regions, e.g., pixels [11] or superpixels [12], and cast the segmentation problem into a node labeling problem in a MRF. This process is illustrated graphically in Figure 2. Such automatic primary video object segmentation methods usually have three major steps: initial visual or motion saliency estimation, spatio-temporal graph connection and foreground/ background appearance modeling. Automatic foreground/background appearance modeling is important as the saliency estimation is usually noisy especially along object boundaries due to cluttered background or background motions. However, it is challenging because there is no prior knowledge about foreground and background regions. Formally, with the presence of appearance constraints, there are two groups of parameters in the optimization process, i.e., segmentation labels x and appearance model.

For many commonly used appearance models such as Gaussian Mixture Models (GMM) [13] or Multiple Instance Learning [14], it is intractable to solve both parameters simultaneously. Hence, many existing methods adapt an iterative approach. They use the segmentation result of the previous iteration to train foreground and background appearance models which are then used to refine the segmentation in the next iteration. However, these methods can be easily trapped in local optimal and are time consuming especially for video data. The latter scenario is more practically relevant, as a good solution would enable processing large amounts of video without human intervention. However, this task is very challenging, as the method is given no knowledge about the object appearance, scale or position. Moreover, the general unconstrained setting might include rapidly moving backgrounds and objects, non-rigid deformations and articulations

In the recent work some of the work follows a fully automatic methods discovers a set of key-segments to explicitly model likely foreground regions for video object segmentation. Idea is to leverage both static and dynamic cues to detect persistent object-like regions, and then estimate a complete segmentation of the video using those regions and a novel localization prior that uses their partial shape matches across the sequence. Most of the proposals do not correspond to an actual object. The goal of the proposed work is to generate an enhanced set of object proposals and extract the segments related to the primary object from the video. In this work propose a technique for fully automatic video object segmentation in unconstrained settings. Proposed method is computationally efficient and makes minimal assumptions about the video: the only requirement is for the object to move differently from its surrounding background in a good fraction of the video. The object can be static in a portion of the video and only part of it can be moving in some other portion (e.g. a cat starts running and then stops to lick its paws). This method does not require a static or slowly moving background (as opposed to classic background subtraction methods [25]). Moreover, it does not assume the object follows a particular motion model, or that all its points move homogeneously (as opposed to methods based on clustering point tracks [15]. This is especially important when segmenting non-rigid or articulated objects such as animals.

The key new element in this approach is a rapid technique to produce a rough estimate of which pixels are inside the object based on motion boundaries in pairs of subsequent frames. This initial estimate is then refined by integrating information over the whole video with a spatiotemporal extension of Min-Cut Max Flow (MCMF) based GraphCut. This second stage automatically bootstraps an effective appearance modeling technique in the MRF based segmentation framework for primary video object segmentation on the initial foreground estimate, and uses it to refine the spatial accuracy of the segmentation and to also segment the object in frames where it does not move. It embeds the appearance constraint directly into the graph by adding auxiliary nodes/connections, and the resultant graph-partition problem can be solved efficiently by one graph cut. Although inspired by the idea of [15] made the non-trivial extension from static images to videos, and generalizes the framework in more complicated cases.

II. LITERATURE REVIEW

Finding “interesting” objects in image or video is a long-standing topic in vision, addressed in various forms including saliency detection, figure-ground segmentation, or object discovery. Whereas most saliency detectors rely on bottom-up image cues (e.g., [16]), recent work suggests that higher-level saliency may actually be learned from labeled data of segmented objects [17], drawing on classic Gestalt cues. In particular, interesting approaches to generate and rank an image’s multiple figure-ground segmentation hypotheses with results showing that higher ranked figure proposals are more likely to be objects in an image. Inspired by this premise, expand the notion of “object-like” regions to video, and introduce the requisite motion and persistence cues. Beyond single images, some

work considers discovering repeated patterns among pairs or groups of unlabeled images [18]. It is challenging since some unknown portion of any image may contain the repeated pattern, calling for iterative refinement techniques [18].

Video offers stronger temporal consistency constraints than assorted snapshots, which this approach aims to leverage. In video with a stationary background, moving foreground regions pop-out well with classic background subtraction algorithms. However, for generic videos with unknown camera motion, lighting changes, and poor resolution or interesting but static objects they are inadequate. Repeated features in video are extracted in [19]; however, the local feature approach means the objects are often not delineated well from background, whereas seek fully segmented regions. More importantly, the grouping objective does not explicitly target discovery of a salient object. To this knowledge, no prior work considers unconstraint category-independent "object-like" foreground regions in video; this is considered as major part of this work

There exist several datasets for video segmentation, but none of them has been specifically designed for video object segmentation, the task of pixel-accurate separation of foreground object(s) from the background regions. Despite being recently adopted by works focusing on video object segmentation [20], the dataset does not fulfill several important requirements. Most of the videos have low spatial resolution, segmentation is only provided on a sparse subset of the frames, and the content is not sufficiently diverse to provide a balanced distribution of challenging situations such as fast motion and occlusions. The Berkeley Video Segmentation Dataset (BVSD) [21] comprises a total 100, higher resolution sequences. It was originally meant to evaluate occlusions boundary detection and later extended to over- and motion-segmentation tasks (VSB100 [22]). However, several sequences do not contain a clear object. Furthermore, the ground-truth, available only for a subset of the frames, is fragmented, with most of the objects being covered by multiple manually annotated, disjoint segments, and therefore this dataset is not well suited for evaluating video object segmentation.

Common low level video segmentation methods include superpixel segmentation [23] and supervoxel segmentation [24]. Superpixel segmentation methods typically over-segment the entire frame into visually coherent groups or segments. Supervoxel segmentation is similar to superpixel segmentation but also groups pixels temporally and, hence, produces spatio-temporal segments. Computer vision applications have come to rely increasingly on superpixels in recent years, but it is not always clear what constitutes a good superpixel algorithm. In an effort to understand the benefits and drawbacks of existing methods, we empirically compare five state-of-the-art superpixel algorithms for their ability to adhere to image boundaries, speed, memory efficiency, and their impact on segmentation performance. The findings have led us to conclusive evidence that the hierarchical graph-based and segmentation by weighted aggregation methods perform best and almost equally-well on nearly all the metrics and are the methods of choice given this proposed assumptions.

The method in [25] produces multiple proposal chains by linking local segments using long-range temporal constraints. It then obtains the final segmentation result by pixel-wise per-frame MRF smoothing using the appearance and location priors learned from these initial chains. The method in [49] first segments the selected key frames into an over complete set of segments using image segmentation algorithms and then employs the cohesive sub-graph mining technique to find the salient segments with similar appearance and strong mutual affinity.

The method in [26] present a video co-segmentation method that uses category-independent object proposals as its basic element and can extract multiple foreground objects in a video set. The use of object elements overcomes limitations of low-level feature representations in separating complex foregrounds and backgrounds. They

formulate object-based co-segmentation as a co-selection graph in which regions with foreground-like characteristics are favored while also accounting for intra-video and inter-video foreground coherence. To handle multiple foreground objects, expand the co-selection graph model into a proposed Multi-state Selection Graph (MSG) model that optimizes the segmentations of different objects jointly. This extension into the MSG have been applied not only to the co-selection graph, but also can be used to turn any standard graph model into a multi-state selection solution that can be optimized directly by the existing energy minimization techniques. The experiments show that our object-based multiple foreground video co-segmentation method (ObMiC) compares well to related techniques on both single and multiple foreground cases. The method in [26] does not have an explicit global appearance model, and to solve this work we adapts the iterative appearance modeling approach using multiple instance learning.

III. PROPOSED METHODOLOGY

This work presents a MCMF-MRF technique for separating foreground objects from the background in a video. MCMF-MRF method is fast, fully automatic, and makes minimal assumptions about the video. This enables handling essentially unconstrained settings, including rapidly moving background, arbitrary object motion and appearance, and non-rigid deformations and articulations. Similar too many existing image and video object segmentation approaches, we cast the segmentation to a two-class node labeling problem in a MCMF-MRF. Within the MRF graph, each node is modeled as a super pixel, and will be labeled as either foreground or background in the segmentation process. It embeds the appearance constraint as auxiliary nodes and edges in the MCMF-MRF structure, and can optimize both the segmentation and appearance model parameters simultaneously in one MCMF. The goal of this work is to segment objects that move differently than their surroundings. The MCMF-MRF video segmentation method has two main stages: (1) efficient initial foreground estimation, (2) foreground-background labelling refinement. Now gives a brief overview of these two stages, and then presents them in more detail in the rest of the section.

1. Efficient initial foreground estimation: The goal of the first stage is to rapidly produce an initial estimate of which pixels might be inside the object based purely on motion. Then compute the optical flow between pairs of subsequent frames and detect motion boundaries. Ideally, the motion boundaries will form a complete closed curve coinciding with the object boundaries. However, due to inaccuracies in the flow estimation, the motion boundaries are typically incomplete and do not align perfectly with object boundaries.

   Also, occasionally false positive boundaries might be detected. Propose a novel, computationally efficient algorithm to robustly determine which pixels reside inside the moving object, taking into account all these sources of error
2. Foreground-background labeling refinement: As they are purely based on motion boundaries, the inside-outside maps produced by the first stage typically only approximately indicate where the object is. They do not accurately delineate object outlines. Furthermore, (parts of) the object might be static in some frames, or the inside-outside maps may miss it due to incorrect optical flow estimation. The goal of the second stage is to refine the spatial accuracy of the inside-outside maps and to segment the whole object in all frames. To achieve this, it integrates the information from the inside-outside maps over all frames by (1) encouraging the spatio-temporal smoothness of the output segmentation over the whole video; (2) building dynamic appearance models of the object and background under the assumption that they change smoothly over time. Incorporating appearance cues is key to achieving a finer level of detail, compared to using only motion. Moreover, after learning the object

appearance in the frames where the inside-outside maps found it, the second stage uses it to segment the object in frames where it was initially missed (e.g. because it is static).

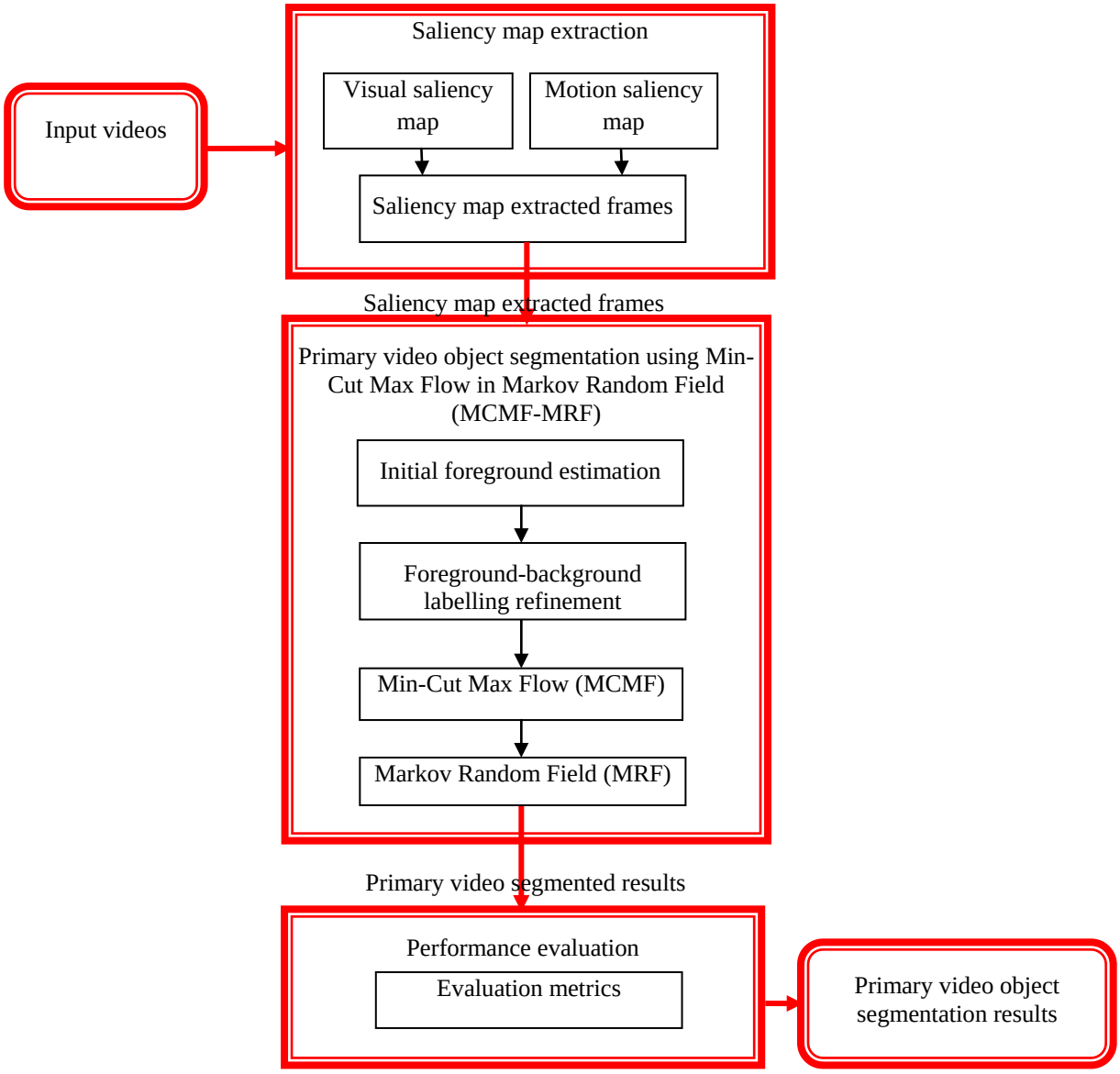

Figure 2: Overall Architecture Diagram

A. Efficient Initial Foreground Estimation

Optical flow begin by computing optical flow between pairs of subsequent frames (t; t + 1) using the state of-the-art algorithm [57]. It supports large displacements between frames and has a computationally very efficient GPU implementation [57]. Motion boundaries approach on motion boundaries, i.e. image points where the optical flow field changes abruptly. Motion boundaries reveal the location of occlusion boundaries, which very often correspond to physical object boundaries [58]. Let $\vec{f}_p$ be the optical flow vector at pixel p. The simplest way to estimate motion boundaries is by computing the magnitude of the gradient of the optical flow field:

$$b_p^m = 1 - \exp(-\lambda^m |\nabla \vec{f}_p|) \quad (1)$$

Where $b_p^m \in [0,1]$ is the strength of the motion boundary at pixel p; λ^m is a parameter controlling the steepness of the function. While this measure correctly detects boundaries at rapidly moving pixels, where b_p^m is close to 1, it is unreliable for pixels with intermediate b_p^m values around 0:5, which

could be explained either as boundaries or errors due to inaccuracies in the optical flow. To disambiguate between those two cases, compute a second estimator $b_p^\theta \in [0,1]$, based on the difference in direction between the motion of pixel p and its neighbours N:

$$b_p^\theta = 1 - \exp\left(-\lambda^\theta \max_{q \in N}(\delta\theta_{p,q}^2)\right) \quad (2)$$

where $\delta\theta_{p,q}^2$ denotes the angle between $\overrightarrow{f_p}$ and $\overrightarrow{f_q}$. The idea is that if n is moving in a different direction than all its neighbours, it is likely to be a motion boundary. This estimator can correctly detect boundaries even when the object is moving at a modest velocity, as long as it goes in a different direction than the background. However, it tends to produce false-positives in static image regions, as the direction of the optical flow is noisy at points with little or no motion . As the two measures above have complementary failure modes, combine them into a measure that is more reliable than either alone :

$$b_p = \begin{cases} b_p^m \ , \text{if } b_p^m > T \\ b_p^m . b_p^\theta \ , \text{if } b_p^m \leq T \end{cases} \quad (3)$$

where T is a high threshold, above which b_p^m is considered reliable on its own. As a last step threshold $b_p$at 0:5 to produce a binary motion boundary labeling.

Inside-outside maps: The produced motion boundaries typically do not completely cover the whole object boundary. Moreover, there might be false positive boundaries, due to inaccuracy of the optical flow estimation. Present here a computationally efficient algorithm to robustly estimate which pixels are inside the object while taking into account these sources of error. The algorithm estimates whether a pixel is inside the object based on the point-in-polygon from computational geometry. The key observation is that any ray starting from a point inside the polygon (or any closed curve) will intersect the boundary of the polygon an odd number of times. Instead, a ray starting from a point outside the polygon will intersect it an even number of times. Since the motion boundaries are typically incomplete, a single ray is not sufficient to determine whether a pixel lies inside the object. Instead, we get a robust estimate by shooting 8 rays spaced by 45 degrees. Each ray casts a vote on whether the pixel is inside or outside. The final inside-outside decision is taken by majority rule, i.e. a pixel with 5 or more rays intersecting the boundaries an odd number of times are deemed inside.

Propose an efficient algorithm which we call integral intersections, inspired by the use of integral images. The key idea is to create a special data structure that enables very fast inside-outside evaluation by massively reusing the computational effort that went into creating the datastructure. For each direction (horizontal, vertical and the two diagonals) we create a matrix S of the same size W ×H as the image. An entry S(x; y) of this matrix indicates the number of boundary intersections along the line going from the image border up to pixel (x; y). For simplicity, we explain here how to build S for the horizontal direction.

The algorithm for the other directions is analogous. The algorithm builds S one line y at a time. The first pixel (1; y), at the left image border, has value S(1; y) = 0. Then move rightwards one pixel at a time and increment S(x; y) by 1 each time we transition from a non-boundary pixel to a boundary pixel. This results in a line S(:; y) whose entries count the number of boundary intersections . After computing S for all horizontal lines, the data structure is ready. Can now determine the number of intersections X for both horizontal rays (left→right, right→left) emanating from a pixel (x; y) in constant time by

$$X_{\text{left}}(x,y) = S(x-1,y) \quad (4)$$

$$X_{\text{Right}}(x,y) = S(W,y) - S(x,y) \quad (5)$$

Where W is the width of the image, i.e. the rightmost pixel in a line. This algorithm visits each pixel exactly once per direction while building S, and once to compute its vote, and is therefore linear in the number of pixels in the image. For each video frame t, apply the algorithm on all 8 directions and use majority voting to decide which pixels is inside, resulting is an inside-outside map Mt.

B. Foreground-background Labelling Refinement

Formulate video segmentation as a pixel labeling problem with two labels (foreground and background). Over segment each frame into superpixels S^t which greatly reduces computational efficiency and memory usage, enabling to segment much longer videos. Each superpixel $s_i^t \in S^t$ can take a label $l_i^t \in \mathcal{L}^t$. A labelling $\mathcal{L} = \{l_i^t\}_{t,i}$ of all superpixels in all frames represents a segmentation of the video. Similarly to other segmentation works define an energy function to evaluate a labeling. Markov Random Field (MRF) considered as a useful framework for characterizing the contextual information and widely used to image segmentation and restoration problems and etc. The statistical dependence between pixels is defined based on their neighbourhood system. In other words, pixels are considered to have a relationship with their neighbours using the concept of context. This brings the concept of smoothness prior model which allows producing smooth image classification pattern. MRF theory including its formulation described with set of random variables $d = \{d_1, d_2, \ldots, d_m\}$ is defined on the set S containing in number of sites in which each random variable in)takes a label from label set L. The family d is called random field. The set S is equivalent to an image containing in pixels; d is a set of pixel DN values; and the label set L depends upon the application. The label set L is equivalent to a set of the user-defined information classes. There are many kinds of random field models describing ways of labelling the random variables". MRF as one of a special type of random fields is described in the next paragraph. Based on the definition of random field, the configuration w for the set S as $w = \{d_1 = w_1, \ldots d_m = w_m\}$). For convenience, the notation of w can be simplified to = $\{w_1, w_2, \ldots w_m\}$. A random field with respect to the neighbourhood system is a MRF if its probability density function satisfies the following three properties;

Positivity: P(w) > 0, for all possible configurations of w , it has non -zero probability and P(w) is the probability of given dataset w.

Markovianity: $P(w_r|w_{s-r}) = P(w_r|w_{xr})$, this defines the neighbourhood system which can be interpreted as follow, membership value of pixel r is strongly dependent on it neighbouring pixels.

Homogeneity: $P(w_r|w_{Nr})$ is the same for all site r, for all pixels probability is dependent on neighbourhood pixels regardless of the pixel location. The neighbourhood system used in image analysis defines the first-order neighbours of a pixel as the four pixels sharing a side with the given pixel, as shown in Figure la. Second -order neighbours the four pixels having the corner boundaries with the pixel of interest. Similarly, higher- order neighbours can be extended same way. Energy minimization is used to solve the pixel labelling problem in different applications such as image restoration and segmentation etc. MAP solution can be obtained only by minimizing the global posterior energy. The posterior energy itself consists of prior and conditional energy function. In accordance to the Bayesian formulae, the MAP solution can be represented following:

$$p(w|d) = \frac{p(d|w)p(w)}{p(d)} \tag{6}$$

P(d) where, w is the membership value and d is a given dataset. The posterior probability can be maximized as follows:

$$w = \arg\max\{p(w|d)\} \tag{7}$$

Equation (6) shows that the MAP estimate is equivalent to the minimization of global energy function and can be expressed as:

$$\widehat{w} = \arg\min\{U(w|d) + U(w)\} \tag{8}$$

Where, $\widehat{w}$ is the optimal class membership value after minimiAng the global energy function, $U(w|d)$ is the conditional energy and U (w) prior energy function and the global posterior energy function can be defined as follow:

$$U(x|d) = U(d|w) + U(w) \tag{9}$$

An additional parameter of A, is added to equation (9) which controls the balance between the two energy functions and value of A ranges between 0 and 1.

Min - cut max - flow Algorithm

Graph cuts algorithms have been studied in computer vision in the last years and they still remain an active research area in this field. Research has been done to develop and improve methods for energy minimization in vision. Graph theory together with a number of optimization methods are provided in this charter. The min-cut/max-flow algorithms from combinatorial optimization that can be used minimization number of energy functions in vision. These energies as well as some graph based methods can be represented as (10).

$$E(L) = \sum_{p \in P} D_p(L_p) + \sum_{(p,q) \in N} V_{p,q}(L_p, L_q) \qquad (10)$$

where $L \in \{L_p | p \in P\}$ is called a labelling of image P, $D_p(.)$ is a data penalty function, $V_{p,q}$ is called an interaction potential, which encourage spatial coherence by penalizing discontinuities between neighbouring pixels, and N is a set of pairs of neighbours pixels . Figure 3 shows example of image labelling problem.

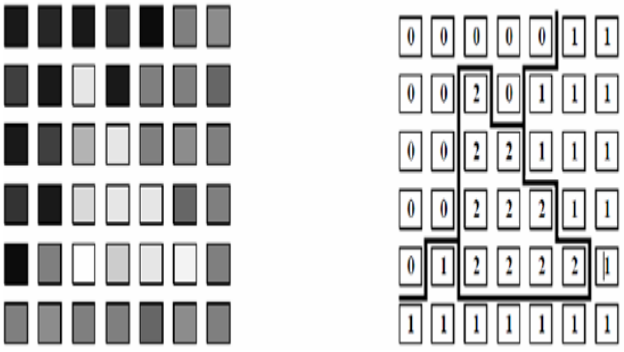

Figure 3: An Example of Image Labelling

An image in Figure.3.(a) is a set of pixels P with observed intensities for each $p \in P$. In the case of Figure 3(b), L assigns label $L_p \in \{0,1,2\}$ to each pixel p P. Such labels can represent object index in segmentation. In case of graph based methods it is assumed that a set of feasible labels at each pixel is finite. Thick lines in Figure 4.(b) show labelling discontinuities between neighbouring pixels. However the minimum cost of the graph produce a globally optimal binary labelling L in the case of Potts model of interaction in (10). A directed weighted graph $G = (V, E)$, which consist of a set of nodes V and a set of directed edges E connecting them .The nodes correspond to pixels in this case.

The graph contains of additional special nodes that are called terminals. These terminals correspond to the set of labels that are assigned to pixels. They are source, s and sink t and they correspond to the set of labels that can be assigned to pixels. Figure 4 shows an example of a two terminal graph that can be used to minimal the Potts case of energy (10) on 3x3 image with two labels. All edges in the graph are assigned some weight or cost. A cost of a directed edge(p,q)may differ from the cost of the reverse edge(q, p). There are two types of edges in the graph: n -links and t -links. N -links connect pairs of neighbouring pixels which represent a neighbourhood system in the image. Cost of n - links corresponds to a penalty for discontinuity between the pixels which are derived from the pixel interaction term $V_{p,q}$ in energy (10) links connect pixels with terminals (labels). The cost of at t-link connecting pixel and a terminal corresponds to a penalty for assigning the corresponding label the pixel which is derived from the data term D_p in the energy (10). Figure 4 shows that t -links re shown in red and blue, but n -are in yellow

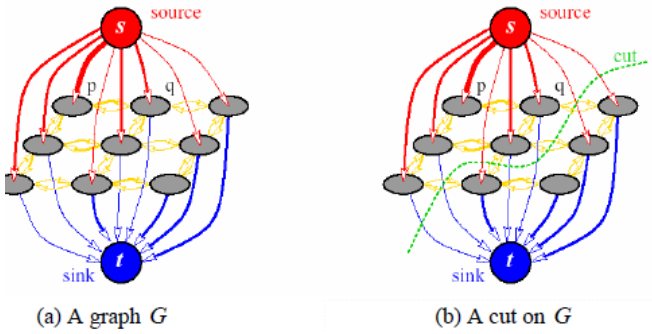

Figure 4: Example a Capacitated Graph

computational time. The graph cuts algorithms such as swap - move and expansion-move are based on min- cut algorithm. The mathematical background of min cut and max -flow provided below is described in An s/t cutC (can be called as cut) is a partitioning of nodes in graph into two disjoint subset T such that the source s is in S and the sink t is in T. An example of is shown in figure 4(b). The cost of cut is C ={S, T} is the sum of cost of boundary edges (p,q)such that $p \in S$ and $q \in T$. The cost is "directed" as it sums up weights of directed edges specifically from T . The minimum cut problem on a graph is to find the cut with minimum cost among all other cuts . One of the results in combinatorial optimization is that the minimum s/t cut problem can be solved by finding the maximum flow from source s to sink t. theorem of Ford states that a maximum flow from s to t saturates a set of edges in the graph dividing the nodes into two disjoint parts{S, T} corresponding to a minimum cut. Thus, min -cut and max-flow problems are equivalent. In fact, the maximum flow value is equal to the cost of minimum cut.

Algorithm Steps

Algorithm 1: Min-Cut Max Flow Optimization in Markov Random Field (MCMF-MRF) video segmentation

Step 1: Efficient initial foreground estimation

1.1. Optical flow begins by computing optical flow between pairs of subsequent frames (t; t + 1).

1.2. Estimate motion boundaries is by computing the magnitude of the gradient in equation (1)

1.3. Determine the number of intersections X for both horizontal rays (left→right, right→left)emanating from a pixel (x; y)

Step 2: Foreground-background labelling refinement

2.1. Formulate video segmentation as a pixel labeling problem with two labels

2.2. A labelling $\mathcal{L} = \{l_i^t\}_{t,i}$ of all superpixels in all frames represents a segmentation of the video.

2.3. Perform 4 MRF theory that satisfies three properties;

2.3.1. Positivity: P(w) > 0, for all possible configurations of w , it has non -zero probability and P(w) is the probability of given dataset w

2.3.2. Markovianity: $P(w_r|w_{s-r}) = P(w_r|w_{xr})$, this defines the neighbourhood system with membership value of pixel r is strongly dependent on it neighbouring pixels.

2.3.3. Homogeneity: $P(w_r|w_{Nr})$ is the same for all site r, for all pixels probability is dependent on neighbourhood pixels regardless of the pixel location.

2.3.4. Compute MAP solution by equation (6)

2.3.5. The posterior probability can be maximized by equation (7)

2.3.6. The global posterior energy function can be defined by equation (9)

2.3.7. Minimization number of energy functions in min-cut/max-flow by equation (10)

Step 3: end

IV. SIMULATION RESULTS

In order to evaluate the effectiveness of the proposed appearance modelling technique, run experiments on several benchmark datasets including the SegTrack v21 and 10-video-clip dataset [13]. The videos in these two datasets are quite challenging. Many of the videos contain cluttered background and dynamic scenes due to camera motion or moving background objects. Some videos even contain fast motions such as the girl, person sequences in the SegTrack v2 dataset and the VWC102T, DO02_001 and DO01_055 sequences in ten video clip dataset. Some videos also contain cluttered background motions such as the swaying tree leaves and grass in the BR128T, BR130T and DO01_030 sequences in the ten video clip dataset. In some videos, the primary objects are visually very similar to the background, i.e., low contrast along object boundaries, such as the birdfall, frog and worm sequences in the SegTrack v2 dataset. Evaluate the proposed

approach against several state-of the- art methods including both MRF based method and non-MRF based methods.

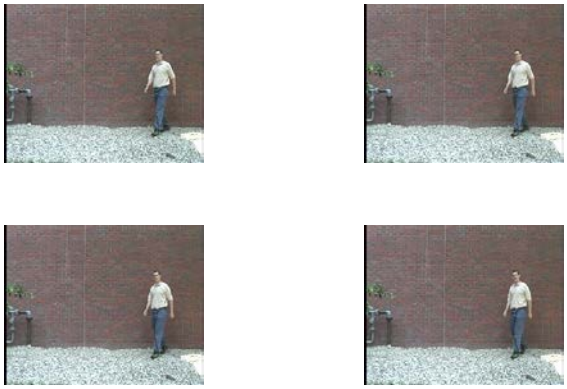

Figure 5: Input Video Frames

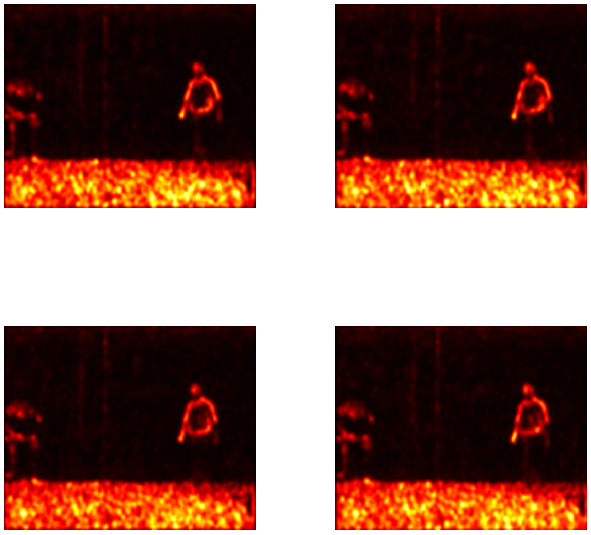

Figure 6: Visual Saliency Map

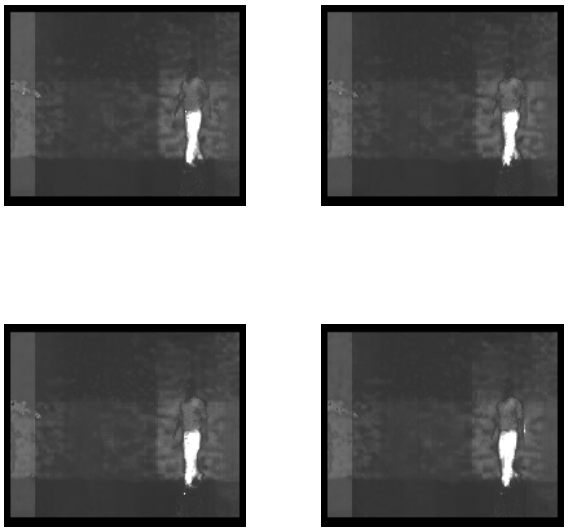

Figure 7: Motion Saliency Map

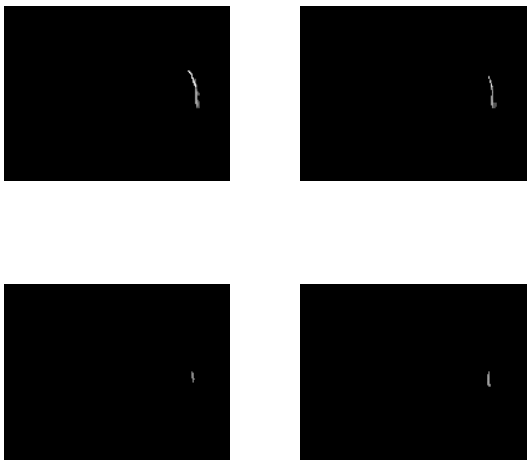

Figure 8: Primary Video Object Segmentation Results

Performance Evaluation

The results are computed and the performance is evaluated based on the parameters like Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), Normalized Correlation Coefficient (NC) and Structural Similarity index (SSIM).

Moreover, it is proposed to use the objects spatial transformation parameters to automatically model and predict the evolution of intrinsic camera parameters and accordingly tune the detector for better performance. The results illustrates that the process is a proposed MCMF-MRF based video object segmentation technique and is highly detection against the different and also for different noisy environments. The mathematical equations for these performance parameters are as given in equation (11) to equation (15).

Peak Signal to Noise Ratio (PSNR)

The PSNR (τ_x) in dB is given as,

$$\tau_x = 10 \log_{10} \frac{R^2}{\mu_x} \tag{11}$$

Where, R is the maximum possible value in the corresponding data and μ_x is Mean Square Error (MSE).

Mean Square Error (MSE)

Mean Square Error (MSE) is defined as

$$\mu_x = \frac{1}{T} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(I_x(i,j) - I_x'(i,j) \right)^2 \tag{12}$$

Where $I_x(i,j)$is the original data, $I_x^{'}(i,j)$is the watermarked data, and M and N are data height and width such that T=M×N. In this work, for video, PSNR is calculated by taking average of PSNR values of all the tracking of corresponding frames of the video. The average PSNR is computed as,

$$\bar{\tau} = \frac{1}{F}\sum_{x=1}^{F}\tau_x \tag{13}$$

Normalized Correlation Coefficient (NC)

The third parameter is Normalized Correlation coefficient (NC) used as reference for finding the similarities between original and extracted video. Since the NC is correlation coefficient, the value of NC as '1' indicates that the multi object extracted is highly correlated to that of the original one and the value of NC as '0' indicates that the multi object extracted is highly uncorrelated to that of the original. For the general correlation, the NC value ranges between 0 and 1. It is obvious from the correlation coefficient that more the value of NC, then the extracted watermark is closer towards the original. For each watermark (corresponding to each frame of the video), the correlation coefficient (NC) is computed using correlation coefficient expression as specified in equation (14).

$$NC_x = \frac{1}{T}\sum_{i=1}^{M}\sum_{j=1}^{N}\overline{I(i,j)\oplus I'(i,j)} \tag{14}$$

Where, T=M×N represents the total number of pixels of object extracted image for x[th] frame. x varies from 1 to F. Exclusive- NOR operation is performed to get the NC value.

Structural Similarity Index (SSIM)

Final parameter computed is SSIM. It is used to measure and evaluate the similarity between two data sets

$$SSIM(I,I^{'},x) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{15}$$

Where, I is the original input image and I' is the object extracted video from x[th] video frame, μ_m is the mean of the intensities available in the original input image of x[th] frame, μ_n is the mean of the intensities available in the object extracted video from x[th] video frame. On the similar grounds, $\sigma_x^2$is variance of original input image I, $\sigma_y^2$is the variance of object extracted input image I' and σ_{xy} is covariance of original and object extracted image of x[th] frame. The graphs shown in Figure 9 to Figure 12, represents the Structural Similarity index (SSIM), Normalized Correlation Coefficient (NCC), Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) of the video and object extracted video under the condition is applied on the video. The graphs shown in all methods represents that the proposed schema performs better for all parameters is applied on the video.

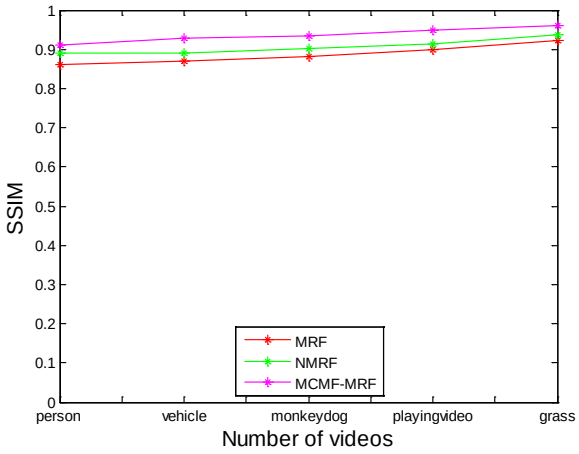

Figure 9: SSIM Comparison of Videos

The table 1 to table 4, represents the Structural Similarity index (SSIM), Normalized Correlation Coefficient (NCC), Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) of the video and object tracked video under the condition is applied on the video. The graphs shown in all methods represents that the proposed MCMF-MRF schema performs better for all parameters is applied on the video.

Table 1: SSIM Comparison of Videos

S.No	*Videos*	*SSIM*		
		MRF	*NMRF*	*MCMF-MRF*
1	Person	0.860	0.889	0.912
2	Vehicle	0.8712	0.8902	0. 9272
3	Monkeydog	0.8817	0.9018	0.9351
4	Playing	0.8982	0.9128	0.9478
5	grass	0.9228	0.9382	0.9593

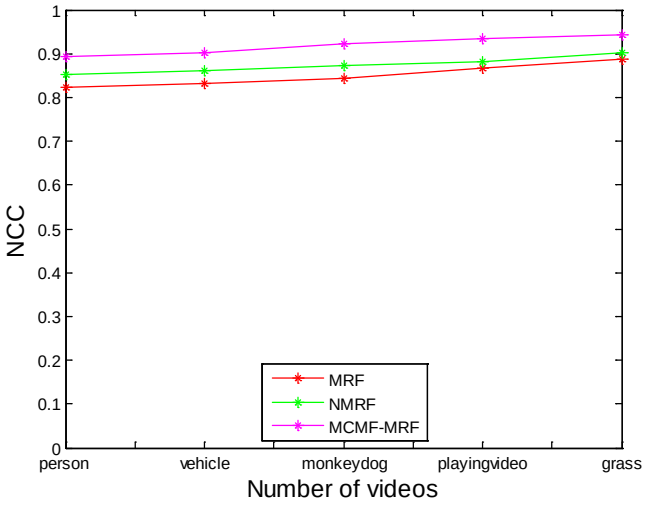

Figure 10: NCC Comparison of Videos

Table 2: NCC Comparison of Videos

S.No	*Videos*	*NCC*		
		MRF	*NMRF*	*MCMF-MRF*
1	Person	0.8230	0.8514	0.8936
2	Vehicle	0.8318	0.8621	0.9016
3	Monkeydog	0.8425	0.8718	0.9236
4	Playing	0.8663	0.8815	0.9352
5	grass	0.8878	0.9012	0.9436

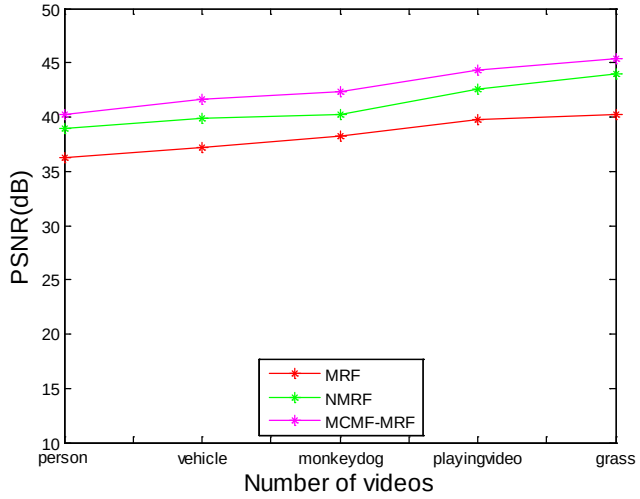

Figure 11: PSNR Comparison of Videos

Table 3: PSNR Comparison of Videos

S.No	*Videos*	*PSNR(dB)*		
		MRF	*NMRF*	*MCMF-MRF*
1	Person	36.25	38.98	40.23
2	Vehicle	37.18	39.86	41.63
3	Monkeydog	38.26	40.29	42.39
4	Playing	39.81	42.58	44.28
5	grass	40.25	43.93	45.37

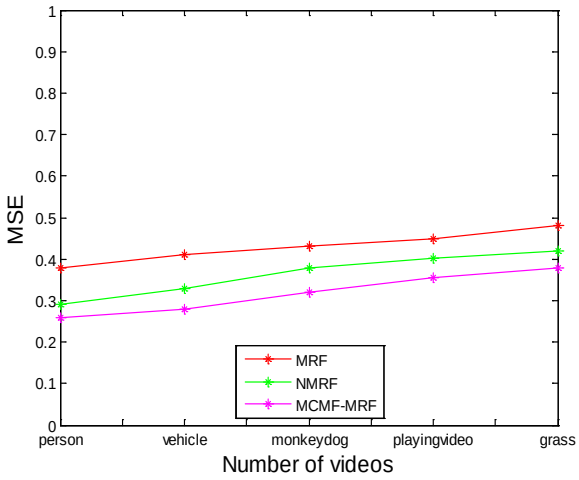

Figure 12: MSE Comparison of Videos

Table 4: MSE Comparison of Videos

S.No	*Videos*	*MSE*		
		MRF	*NMRF*	*MCMF-MRF*
1	Person	0.38	0.29	0.26
2	Vehicle	0.41	0.33	0.28
3	Monkeydog	0.43	0.38	0.32
4	Playing	0.4500	0.4012	0.3563
5	grass	0.4820	0.4186	0.3782

V. CONCLUSION AND FUTURE WORK

In this work propose an efficient and effective appearance modeling technique in the MCMF-MRF framework for automatic primary video object segmentation. The goal of this work is to segment objects that move differently than their surroundings. The MCMF-MRF video segmentation method has two main stages: (1)efficient initial foreground estimation, (2) foreground-background labelling refinement. Efficient initial foreground estimation goal of the first stage is to rapidly produce an initial estimate of which pixels might be inside the object based purely on motion. Then compute the optical flow between pairs of subsequent frames and detect motion boundaries. Ideally, the motion boundaries will form a complete closed curve coinciding with the object boundaries. Foreground-background labeling refinement they are purely based on motion boundaries, the inside-outside maps produced by the first stage typically only approximately indicate where the object is. They do not accurately delineate object outlines. Furthermore, (parts of) the object might be

static in some frames, or the inside-outside maps may miss it due to incorrect optical flow estimation. The proposed method uses histogram features to characterize the local regions and embed the global appearance constraint into the graph by auxiliary nodes and connections. Compared with many existing appearance models, the optimization process of proposed method is non-iterative. Experimental evaluations show that proposed method is faster than many of the alternatives and the segmentation accuracy is also better than or comparable with the state-of-the-art methods. Currently, running time efficiency and memory requirements are a major bottleneck for the usability of several video segmentation algorithms. In these experiments observed that a substantial amount of time is spent preprocessing images to extract boundary preserving regions, saliency based feature extraction and motion estimates. Encourage future research to carefully select those components bearing in mind they could compromise the practical utility of their work. Efficient algorithms will be able to take advantage of the Full videos and accurate segmentation masks made available with this dataset. Leveraging high resolution might not produce better results in terms of region-similarity, but it is essential to improve the segmentation of complex object contours and tiny object region.

REFERENCES

[1] X. Bai, J. Wang, D. Simons and G. Sapiro, "Video snapcut: robust video object cutout using localized classifiers", ACM Transactions on Graphics, Vol. 28, No. 3, 2009.

[2] B. Price, B. Morse and S. Cohen, "Livecut: Learningbased interactive video segmentation by evaluation of multiple propagated cues", 12th International Conference on Computer Vision, Pp.779–786, 2009.

[3] X. Ren and J. Malik, "Tracking as repeated figure/ ground segmentation", Conference on Computer Vision and Pattern Recognition, Pp. 1–8, 2007,

[4] Y. Sheikh, O. Javed and T. Kanade, "Background subtraction for freely moving cameras", International Conference on Computer Vision, Pp. 1219–1225, 2009.

[5] A. Vazquez-Reina, S. Avidan, H. Pfister and E. Miller, "Multiple hypothesis video segmentation from superpixel flows", European conference on Computer vision, Pp. 268–281, 2010

[6] Y. Lee, J. Kim and K. Grauman. Key-segments for video object segmentation. International Conference on Computer Vision, Pp. 1995–2002, 2011.

[7] D. Zhang, O. Javed and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions", Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Pp. 628–635, 2013.

[8] T. Wang and J. Collomosse, "Probabilistic motion diffusion of labeling priors for coherent video segmentation", IEEE Trans. Multimedia, 2012

[9] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories", European conference on computer vision, Pp. 282-295, 2010.

[10] P. Chockalingam, S.N. Pradeep and S. Birchfield, "Adaptive fragments-based tracking of non-rigid objects using level sets", 12th international conference on computer vision, Pp. 1530-1537, 2009.

[11] M. Tang, L. Gorelick, O. Veksler and Y. Boykov, "GrabCut in one cut", Proc. IEEE Int. Conf. Comput. Vis., Pp. 1769–1776, 2013.

[12] S.D. Jain and K. Grauman, "Supervoxel-consistent foreground propagation in video", Proc. Eur. Conf. Comput. Vis., Pp. 656–671, 2014.

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

[14] L. Wang, G. Hua, R. Sukthankar, J. Xue and N. Zheng, "Video object discovery and co-segmentation with extremely weak supervision", Proc. 13th Eur. Conf. Comput. Vis., Pp. 640–655, 2014.

[15] O. Barnichm and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences", IEEE Trans. Image Processing, 2011.

[16] L. Itti and P.F. Baldi, "A Principled Approach to Detecting Surprising Events in Video", Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, Pp. 631-637, 2005.

[17] T.L.J. Sun, N.N. Zheng, X. Tang, H.Y. Shum and P.R. Xi'an, "Learning to detect a salient object", Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, Pp. 1-8, 2007.

[18] G. Kim and A. Torralba, "Unsupervised detection of regions of interest using iterative link analysis", Advances in neural information processing systems, Pp. 961-969, 2009.

[19] T. Quack, V. Ferrari and L.V. Gool, "Video Mining with Frequent Itemset Configurations", International Conference on Image and Video Retrieval, Pp. 360-369, 2006.

[20] F. Perazzi, O. Wang, M. Gross and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation", IEEE International Conference on Computer Vision, Pp. 3227-3234, 2015.

[21] P. Sundberg, T. Brox, M. Maire, P. Arbelaez and J. Malik, "Occlusion boundary detection and figure/

ground assignment from optical flow", IEEE Computer Vision and Pattern Recognition (CVPR), Pp. 2233-2240, 2011.

[22] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani and L. Van Gool, "Seeds: Superpixels extracted via energy-driven sampling", Proc. 12th Eur. Conf. Comput. Vis., Pp. 13–26, 2012.

[23] C. Xu and J.J. Corso, "Evaluation of super-voxel methods for early video processing", IEEE Conf. Comput. Vis. Pattern Recognit., Pp. 1202–1209, 2012.

[24] D. Banica, A. Agape, A. Ion and C. Sminchisescu, "Video object segmentation by salient segment chain composition", Proc. IEEE Int. Conf. Comput. Vis. Workshops, Pp. 283–290, 2013.

[25] Y. Luo, G. Zhao and J. Yuan, "Thematic saliency detection using spatial-temporal context", Proc. IEEE Int. Conf. Comput. Vis. Workshops, Pp. 347–353, 2013.

[26] H. Fu, D. Xu, B. Zhang and S. Lin, "Object-based multiple foreground video co-segmentation", IEEE Conference on Computer Vision and Pattern Recognition, Pp. 3166–3173, 2014.