Particle filter framework for salient object detection in videos

Karthik Muthuswamy, Deepu Rajan

Centre for Multimedia and Network Technology, School of Computer Engineering, Nanyang Technological University, 50 Nanyang Avenue, N4-02C-92 639798, Singapore
E-mail: kart0028@e.ntu.edu.sg

Abstract: Salient object detection in videos is challenging because of the competing motion in the background, resulting from camera tracking an object of interest, or motion of objects in the foreground. The authors present a fast method to detect salient video objects using particle filters, which are guided by spatio-temporal saliency maps and colour feature with the ability to quickly recover from false detections. The proposed method for generating spatial and motion saliency maps is based on comparing local features with dominant features present in the frame. A region is marked salient if there is a large difference between local and dominant features. For spatial saliency, hue and saturation features are used, while for motion saliency, optical flow vectors are used as features. Experimental results on standard datasets for video segmentation and for saliency detection show superior performance over state-of-the-art methods.

1 Introduction

The human visual system (HVS) does a remarkable job at separating foreground from background clutter even when the objects in the scene have never been encountered before. This capability is often attributed to its ability to quickly locate ‘attentive’ regions in the scene and subsequently, to understand the rest of the scene. In computer vision, the phenomenon of identifying attentive regions in image and videos is known as salient region detection, where the region may or may not represent a single object. The two main approaches to salient region detection are top-down and bottom-up. The former approach is an object or task-oriented process and incorporating a fair amount of training, while the bottom-up approach is driven by contrast between features leading to a measure of conspicuity of a region.

The distinguishing aspect between image and video saliency is the temporal motion information in the latter that introduces the notion of motion saliency in addition to spatial saliency of images. Bottom-up saliency for videos helps in foreground region detection by assigning a saliency measure to a pixel to be spatially and temporally salient. The most commonly used technique to easily segment the foreground from the background is by applying a threshold to the saliency map [1]. However, if the saliency map is not accurate, this might result in spurious regions being identified as foreground.

In this paper, we propose a method for detecting the most salient object in a video based on a computationally simple approach for the generation of spatio-temporal saliency maps. The most salient object is identified by localising a set of pixels using particle filters whose weights are computed from the saliency maps and colour features. The location of the particles is influenced by the saliency map as well as by the colour information in the frame. The spatio-temporal saliency map allows the particle filter to converge faster on the most salient foreground object by refining the weights of the particles while the colour map provides the robustness in matching the reference and target distributions. The spatio-temporal saliency map is generated at the pixel-level in the original resolution as opposed to other methods [2–4] that operate on a reduced resolution or on patches to reduce the computation time. In the latter case, the accuracy of the saliency maps is compromised when resized to the original resolution while our pixel-level computation to generate saliency maps has been shown to generate accurate saliency maps at an average processing time of eight frames per second on a frame resolution of $352 \times 288$.

2 Related work

In the context of videos, motion saliency plays a very important role in identifying the final saliency measure, in addition to the spatial saliency from individual frames. Here, we present the relevant work in motion saliency and refer the readers to [5] for a survey on spatial salient region detection. Stimuli-based saliency detection in videos was first presented by Itti and Baldi [6] where they propose that salient events are unexpected or surprising. In their model, they define surprise as a statistical deviation from a recent history of visual stimuli. Mahadevan and Vasconcelos [7] have extended the centre-surround hypothesis of saliency detection to videos, to measure the spatio-temporal saliency.
measure by modelling the motion present in the frames as a dynamic texture. Their method is largely dependent on the size of the patch used in the centre-surround formulation and is also computationally expensive, requiring more than half a minute per frame to calculate the spatio-temporal saliency measure. Gopalakrishnan et al. [8] model the motion present in the video using an autoregressive moving average model, identifying salient motion using control theory concept of observability. They show that salient foreground motion is more predictable than the dynamic background motion. In our previous work [9], we extended [8], to model the background motion using controllability. Similar to [8], we model the video as a linear dynamic system whose parameters are identified from the video.

Xia et al. [10] use a motion history map to mark pixels in a salient object using its self-information and global contrast information. Luo et al. [11] propose a spatio-temporal saliency measure where spatial and temporal consistency measures are included in the sparse feature selection model. Ren et al. [12] propose a saliency measure based on the sparse reconstruction of features where the spatial saliency is captured from high centre-surround contrast measures and the temporal saliency is measured from the local motion trajectory contrast obtained from spatio-temporal patches. Li et al. [13] propose a sparse centre-surround saliency model that detects patches that contain more information than its spatio-temporal surrounding patches.

Guo et al. [14] utilise a phase spectrum instead of an amplitude spectrum to detect saliency in videos. Spectral-based techniques tend to detect regions with rich edges and their biological plausibility is often unclear. Zhai and Shah [15] propose a computationally expensive method for spatial colour contrast and motion contrast measurement based on point correspondences in consecutive frames. Seo and Milanfar [16] represent a pixel/voxel using local regression kernels and measure the centre-surround contrast based on the resemblance of a pixel/voxel to its surroundings. Li et al. [17] propose a motion saliency measure based on the consistency in motion calculated from the motion vectors in the frame. Marat et al. [18] propose spatial and temporal filters that are inspired by the retina model to estimate spatial and temporal saliency measures. Table 1 provides a summary of the related work on motion saliency, categorising them based on the approach followed.

Since our objective is salient object detection in videos, the underlying assumption is that there are one or more objects that capture the attention of the HVS. This is related to the task of object segmentation in videos. However, to the best of our knowledge, there is only one previous work that uses saliency for this task, viz. [28]. Their spatial saliency measure is obtained by segmenting each frame by identifying superpixels whose saliency values are calculated as the distance of the centroid of a superpixel to that of its surroundings. Their motion saliency map is calculated by performing a dense optical flow to identify the pixels that are in motion through a forward or backward optical flow propagation. It is known that both segmentation and dense optical-flow are expensive techniques to be performed on every frame on a video. They utilise a CRF to combine the spatial and temporal saliency maps which makes it even more expensive for the calculation of the spatio-temporal saliency map. In this paper, they report a computation time of 86 s per frame, corroborating our statement about the computationally expensive nature of their approach. Other methods for video object segmentation use appearance [29, 30], motion [31, 32] and a combination of the two cues [33, 34]. Graph-based models [30] and mean-shift techniques [35] are also commonly employed. Lin and Davis [36] and Wu and Nevatia [37] train object part detectors by composing an object shape model hierarchically utilising these detectors to describe all possible configurations of the object of interest. Yin et al. [38], Criminisi et al. [39] and Li et al. [28] propose object extraction techniques based on a conditional random field (CRF). While Yin et al. [38] and Criminisi et al. [39] employ CRF to maximise the joint probability of colour and motion models to predict the labels of each image pixel, Li et al. [28] employ a CRF to combine saliency induced features to extract objects from a video.

Unsupervised video segmentation approaches do not require off-line training of specific object detectors or classifiers. In such approaches, foreground object extraction can be solved by treating the problem as background subtraction. In other words, foreground objects are identified by subtracting the background information from the current frame information. These techniques are typically applied to videos that are shot using static cameras [40, 41]. In order to apply such methods to a wider variety of videos, background subtraction is performed by learning the background model from the input video, treating the foreground pixels as outliers that cannot be accommodated by the learnt background model. Staufer and Grimson [42]

Table 1 Summary of related work in motion saliency

<table>
<thead>
<tr>
<th>Approach to saliency</th>
<th>Summary of method</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>biologically inspired works [6, 18, 19]</td>
<td>biologically inspired models that try to emulate the HVS</td>
<td>patch based approaches resulting in lower resolution saliency maps</td>
</tr>
<tr>
<td>optical flow based methods [17, 20–24]</td>
<td>models salient motion based on optical flow vectors either by assuming straight line motion or through clustering optical flow vectors to label salient motion</td>
<td>optical flow based saliency detection techniques would fail when the video has global motion</td>
</tr>
<tr>
<td>background modelling based methods [7, 23, 25–27]</td>
<td>motion present in the video is modelled as an AR system by learning the parameters from a set of frames</td>
<td>require a training phase to learn the background statistics and, they fail to identify salient regions when the video is shot using a freely moving camera</td>
</tr>
<tr>
<td>autoregressive (AR) model-based methods [7–9]</td>
<td>such models represent spatial and temporal features by sparse feature reconstruction</td>
<td>videos shot with freely moving cameras pose a great challenge to such frameworks when learning the model parameters as these models are based mainly on centre-surround approach to saliency detection, saliency maps are still of inferior quality such techniques often tend to be expensive when calculated on a for every frame of a video</td>
</tr>
<tr>
<td>other methods</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
utilise a Gaussian mixture model as a probabilistic model to estimate the background while Elgammal et al. [25] utilise a non-parametric background model to estimate the background based on the history of intensity values. Bugeau and Pérez [21] compensate for the camera motion based on the history of intensity values.

### 2.1 Main contributions

We present a framework that is novel on two fronts. Firstly, we propose a novel low-cost method to calculate full-resolution saliency maps based on dominant features. We show that this measure could be utilised to measure spatial as well as temporal saliency and we propose a new technique to combine the two measures, based on the motion present on each frame, to generate the final spatio-temporal saliency map.

Secondly, we propose a new technique that utilises the spatio-temporal saliency map to identify and track the most salient object present in a given video. While particle filters have been utilised to perform object tracking, the proposed method provides an automated mechanism to initialise, track and self-correct itself to identify the ‘most salient’ object in the video.

### 3 Salient object detection

We employ a particle filter for its ability to approximate the posterior distribution of a system based on a finite set of weighted samples. Particle filters are also highly robust to partial occlusion and computationally inexpensive. The weight of each particle is initialised using a uniform distribution from a hypothetical state of the system with a corresponding weight.

The spatio-temporal saliency and the colour maps are used to calculate the weight of the samples, which allows subsequent iterations to move the particles closer to the most salient object. In the proposed framework, we detect only one object of interest. Fig. 1 illustrates a workflow of how the particle filter framework is used to detect the salient object. Colour versions of all the Figures used in this paper are available online.

#### 3.1 Particle filters

In this section, we provide a brief review to particle filters. Please refer to Thrun et al. [43] and Arulampalam et al. [44] for more details. A particle filter is a sequential Monte Carlo method that recursively approximates the posterior distribution from a finite set of weighted samples: \( \{x_i^t, w_i^t\}_{i=1,...,N} \), where each sample \( x_i^t \) represents a hypothetical state of the system with a corresponding weight \( w_i^t \).

Considering that we have the observations \( Y_t = y_0, ..., y_t \) of the system up to time \( t \), the goal is to estimate the state of the system \( x_t \) from the posterior distribution \( p(x_t|Y_t) \).

The state-space model of the system can be represented as

\[
x_t = A x_{t-1} + \eta_{t-1}
y_t = C x_t + \epsilon_t
\]

where \( A \) and \( C \) are the state transition and measurement matrices and \( \eta_{t-1} \) and \( \epsilon_t \) are the system and measurement noises. Similar to any linear Bayesian technique, a particle filter employs a predict and update approach. During the prediction step, the posterior probability density at time \( t \) is calculated using the state transition model

\[
p(x_t|Y_{t-1}) = \int p(x_t|x_{t-1}) p(x_{t-1}|Y_{t-1}) \, dx_{t-1}
\]

The correction step updates the posterior using Bayes’ rule as

\[
p(x_t|Y_t) = \frac{p(y_t|x_t)p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})}
\]

#### 3.2 Extracting feature distribution

The proposed approach utilises colour and spatio-temporal maps as features to detect the salient object. At this point, we assume that both feature maps are available to us. Our computationally efficient method to generate spatio-temporal maps is presented in Section 4. We employ colour for its robustness against structural changes in the object while the spatio-temporal saliency measure allows the filter to quickly converge on the salient object. We calculate the observation likelihood of colour as a feature and subsequently use it together with the spatio-temporal...
measure of the samples to refine the weights. The feature distribution \( p(x) = p_u(x) u = 1, 2, \ldots, m \) of a region centred at location \( x \) is given by Nummiaro et al. [45]

\[
p_u(x) = K_{\text{norm}} \sum_{r=1}^{N_p} k \left( \frac{|x - x_i|^2}{h} \right) \delta(b(x) - u)
\]

(4)

where \( K_{\text{norm}} \) is a normalising constant, \( \delta \) is the Kronecker function, \( k \) is a kernel with bandwidth \( h \), \( N_p \) is the number of pixels in the region and \( b(x) \) is a function that assigns the bin number to the feature obtained from location \( x \). The histograms are calculated from the HSV colour space to make the algorithm less sensitive to lighting conditions, wherein the number of bins \( m \) is set to \( 8 \times 8 \times 4 \). The kernel is a spatial weighting function that gives lower weight to pixels farther away from the centre \( x_i \) and is defined as [45]

\[
k(r) = \begin{cases} 1 - r^2, & \text{if } r < 1 \\ 0, & \text{otherwise} \end{cases}
\]

(5)

In order for the particles to be temporally coherent, the feature distribution sampled from frame \( t \) must be similar to the distribution sampled from frame \( t-1 \). Although a number of methods such as histogram intersection, entropy and so on have been used to compute the similarity between features, we adopt the Bhattacharyya distance [45] as it is computationally inexpensive. Let us define \( p' = \{p(x)\}_{t-1}^{t-1, \ldots, t} \) as the distribution sampled from the reference model and \( p(x_t) \) as a candidate model sampled from the current frame. The distance between the distributions \( p' \) and \( p(x_t) \) is calculated as

\[
p[p', p(x_t)] = \left[ 1 - \sum_{i=1}^{m} \sqrt{p'_u(x) p_u(x)} \right]^{1/2}
\]

(6)

where \( m \) is the number of bins in the histogram.

3.3 Particle initialisation and weight computation

Unlike a tracking framework, where the filter is initialised with a bounding box containing the object to be tracked, our method initialises the first set of particles at the centre of the frame, with each particle carrying the same weight. The initial samples are obtained within an ellipse whose two axes are set to half the width and height of the frame, respectively. Fig. 2 shows the particle initialisation and the final result of salient object detection. Fig. 2a shows the frame in which the particles are initialised. Once the particles are initialised, the subsequent steps iteratively update the weights of the particles based on the colour and spatio-temporal maps to detect the salient object. Fig. 2b shows the result of the object detection (as a red ellipse).

The weights of the particles are computed using spatio-temporal saliency and the colour maps as features. We obtain \( N \) samples \( x_i, t = 1, \ldots, N \) from the frame \( t \). Let the distance between \( p_c \) and the \( ith \) candidate colour distribution be \( \phi(c) \) (subscript \( c \) is for the colour). We assign to each sample \( x_i \) an importance weight given by

\[
w_i = \phi(c) e^{-\lambda \phi_i^2}
\]

(7)

where \( \phi_i \) is the mean value of the intensities within the elliptical region in the spatio-temporal saliency map. (Note that the saliency maps in this paper are represented as hot spots for better visualisation, but in the computations, they are represented as grey-scale images.) The parameter \( \lambda \) is set to 20 in our experiments similar to [46]. According to (7), if the Bhattacharya distance of the candidate and the reference distribution is low, implying high colour similarity and the mean of the spatio-temporal samples are high, then the corresponding sample gets a high weight. This ensures that new particles are obtained based on high colour similarity and a high spatio-temporal saliency, thus removing particles that have low spatio-temporal saliency. This also ensures that the particles move closer to the most salient object over a period of time through particle removal and addition of new ones.

3.4 Particle filter implementation

A particle filter is driven by the state vector and the dynamic model of the system. Similar to [45], we sample an ellipse whose state is defined as \( x = [x, y, \dot{x}, \dot{y}, H_x, H_y, \dot{a}] \), where \( x, y \) provides the location of the ellipse, \( \dot{x}, \dot{y} \) the velocity components, \( H_x, H_y \) are the length of the half axes of the ellipse and \( \dot{a} \) indicates the scale change. The state propagation is done using first-order auto-regressive (AR) process as \( x_t = Ax_{t-1} + \eta_{t-1} \). The state transition matrix used in the dynamic model is a constant velocity and scale model. The observation likelihood for each sample is updated using (7). Once the weights of the samples are updated, the mean state or the location of the salient object is calculated as

\[
E[w] = \sum_{i=1}^{N} w_i^t x_t^u
\]

(8)

The location of the ellipse in frame \( t \) is updated using the state information calculated from (8). Fig. 3 shows the results of object detection on two different videos. The top row of Fig. 3 shows a tracking shot of a lynx running on a snow covered region. Even though the background consists of large motion, the spatio-temporal saliency map (to be described in the next section) is able to clearly demarcate the salient object allowing the particles to converge and track the lynx reliably. The bottom row is also an example of a tracking shot of a skier in action. The shot demonstrates the ability of the framework to detect different salient objects at different times. Initially, the pine trees in the background are deemed salient as shown in Fig. 3b. However, the skier is detected and tracked successfully in the subsequent frames as shown in Figs. 3d and f.

The weights of the particles are reinitialised every \( \tau \) frames in order to avoid fixating on a particular object for the entire

Fig. 2 Particle initialisation and the final result of the salient object detection

a Particle initialisation
b Salient object detection


www.ietdl.org


© The Institution of Engineering and Technology 2015
duration of the video sequence. In our experiments, we set \( \tau = F/2 \), where \( F \) is the frame rate of the video. We described how the proposed framework avoids fixation on a particular object through an example shown in Fig. 3 second row. We provide another example in Section 5 describing how the proposed method might erroneously detect the wrong object as salient, but with particle weight reinitialisation, there is quick recovery from the error.

It can be noted that the proposed model can be extended to detect multiple objects of interest by clustering and segmenting the video frame based on the spatio-temporal saliency values. This would allow pixels with similar saliency measures and those that are spatially close to be clustered together. The cluster, which represents a region/object is associated with a rank based on the average saliency value of the cluster. Multiple particle trackers can be initialised around each of the cluster centroids as seed points, thus, building a framework that would consider multiple hypothesis based on different colour models that are obtained from each cluster and whose weights are calculated based on the colour composition of the segment. The salient object detection approach presented assumed the availability of a spatio-temporal saliency map. In the next section, we describe how such a map can be obtained at the pixel level with computational efficiency.

## 4 Spatio-temporal saliency

Salient region detection approaches in [2–4, 16] measure saliency at the patch-level resulting in saliency maps at reduced resolutions. The saliency of a pixel in the original resolution is obtained by interpolation, which introduces spurious regions in the saliency map. Our approach calculates saliency at pixel-level without significant computational overhead. The conspicuity of a region is measured by computing the contrast difference present in the region to the rest of the frame [2]. Saliency at a pixel location can be measured as the accumulated distance of the feature at the pixel to the features extracted from the rest of the pixel locations in the frame. For a frame consisting of \( n \) pixels, this requires the calculation of the differences for \( n-1 \) pixels for every pixel, requiring a total of \( n(n-1) \) calculations per frame. This would require calculations of the order of \( O(n^2) \) for every frame of a video, which is inefficient especially for videos.

In the proposed framework, we reduce the number of computations required for calculating saliency, by measuring the contrast of a feature at a given pixel location to that of its set of dominant values. Thus, we measure the conspicuity of a pixel by accumulating the distance of the feature value \( f_i \) at pixel \( i \) to the dominant feature values \( d \in D \), weighted by the probability of occurrence of the dominant feature value. The conspicuity of \( f_i \) is hence formulated as

\[
\text{Sal}(f_i) = \sum_{d \in D} p(d)||f_i - d||_1
\]

where \( p(d) \) denotes the probability of occurrence of the dominant feature \( d \) and \( ||.||_1 \) is the L1 norm. The dominant feature values are calculated by constructing a 50-bin histogram from the set of feature values. This histogram is smoothed to avoid secondary peaks, using a moving average filter with a 5-bin window. The local maxima values present in the smoothed histogram form the set of dominant feature values. The proposed measure reduces the maximum number of computations required per pixel to the number of dominant feature values. As the proposed conspicuity measure is able to efficiently measure the contrast of features with the least number of computations, we employ it to calculate the temporal saliency measure, discussed in Section 4.1 and the spatial saliency measure, discussed in Section 4.2. Subsequently, we combine the two measures to calculate the spatio-temporal saliency measure in Section 4.3.

### 4.1 Motion saliency map

Studies have shown that motion plays a major role in garnering attention and to outweigh low-level features such as orientation, intensity and texture in videos [47]. Yantis et al. [48] have shown that the HVS is particularly sensitive to isolated abrupt stimulus and relative movement of objects which refers to the contrast in motion within a spatial neighbourhood. In a similar spirit, we identify salient motion from the contrast or the relative movement of objects in the scene calculated as

\[
\text{SMot}(m_k) = \sum_{m \in M} p(m)||m_k - m||_1
\]

where \( M \) is the set of dominant motion magnitudes computed from optical flow vector velocities that are obtained according to [49] and \( m_k \) is the motion magnitude at pixel \( k \). The dominant motion magnitudes in each frame are calculated...
using the technique discussed in Section 4. The local maxima values form the set $M$.

4.2 Spatial saliency map

Spatial attention cues influence the attention process when similar motion dominates the frame, for example, swaying of trees in the background or when the foreground objects have similar motion, for example, a crowd walking in the same direction. The spatial saliency map of a frame is calculated based on the colour contrast according to the formulation in (9). Unlike motion, colour contrast is influenced by the compactness of similar colour pixels [50]. We adopt the method proposed by Morse et al. [51] to calculate the dominant colours using hue and saturation maps from the HSV image. The effect of the saturation values are included in the hue histogram which is hence defined as $H(b) = \sum_{(x,y) \in I_b} S(x,y)$, where $H(b)$ is the saturation value at location $(x, y)$ and $I_b$ is the set of image coordinates having hues corresponding to the $b$th bin. The dominant hues are calculated in a similar manner as dominant motion by obtaining the local maxima values from a smoothed histogram $H(b)$. The spatial saliency measure of a pixel $k$ is

$$SSp(c_k) = \sum_{c \in C} \rho(c) ||c_k - c||_1$$  \hspace{1cm} (11)

where $C$ is the set of dominant hues and $c_k$ is the hue value at pixel $k$.

Fig. 4 shows frames from three different videos and their corresponding spatial, motion and spatio-temporal saliency maps. The top row shows a frame from a video of a sauntering rhinoceros, shot with a stationary camera while the middle and bottom rows are videos of a wolf roaming in the forest and a skier performing a stunt, respectively. The last two videos are of tracking shots.

In the top row, the movement of the rhinoceros is large when compared to the background flutter in the grass. Hence, the animal’s motion is assigned a larger saliency than that of the swaying grass in the background. In the video with the wolf (middle row), the pixels present in the background have large motion primarily owing to the camera tracking the wolf while the motion magnitudes of the pixels on the wolf are low. The motion saliency map assigns a high confidence to the motion of the wolf owing to the large difference in the magnitudes between the tracked wolf and the dominant background motion. The frame shown in the bottom row of Fig. 4 is from a video of a camera tracking a skier performing a stunt. Similar to the case above, the skier is executing a somersault when his hand moves faster than the rest of his body, generating a large saliency measure along his arm. Thus, it can be seen that the proposed motion saliency measure is able to successfully identify the salient motion present from videos shot with a stationary camera or a tracking camera.

4.3 Spatio-temporal saliency map

The spatio-temporal saliency map combines the spatial and motion saliency maps calculated for each frame in such a way that the motion map gets a larger weight if there is high motion contrast in the sequence while the spatial saliency map gets a larger weight if the motion contrast is low. This is formulated as

$$STSal(I) = \alpha \times SMot(I) + (1 - \alpha) \times SSp(I)$$ \hspace{1cm} (12)

where $\alpha$ is an adaptive weight given by

$$\alpha = \frac{\text{median}(SMot(I))}{\text{max}(SMot(I))}$$ \hspace{1cm} (13)

If a large number of pixels have high motion saliency, then the median is closer to the maximum. On the other hand, if
there are fewer pixels that are closer to the maximum motion saliency measure, the median returns a lower value and \( \alpha \) evaluates to a lower value indicating that the influence of motion is not large enough for the motion saliency to dominate the attention process.

The spatio-temporal saliency map generated for the three video sequences discussed earlier are shown in Fig. 4d. The motion adaptive weights allow for a better estimate of the spatio-temporal saliency measure when the video is influenced by a large motion contrast. The top row in Fig. 4d is an example where the motion cues play a major role in the spatio-temporal saliency measure as the dominant hues are not very different for the foreground salient object and background. The frames shown in the middle and bottom row of Fig. 4d are examples where the camera tracks an object of interest. In this case, the dominant motion pixels are present in the background although the motion saliency is not large enough until the object had its own local motion. In these two cases, the spatial saliency map takes over the majority of the spatio-temporal saliency measure as the motion contrast is not high enough for the motion map to get control.

5 Experiments and results

In this section, we evaluate the proposed salient object detection approach together with the efficacy of the spatio-temporal saliency maps. The most appropriate way to evaluate a salient object detection algorithm is to compare the extracted object region with a ground truth segmentation of the object. To this end, we use the segmentation dataset described in [28], which consists of eight videos, mostly containing a single salient object. There are shots taken from a freely moving camera (‘Girl’), tracking shots (‘Fox’ and ‘Parachute’), and shots from a stationary camera (‘Penguin’ and ‘Redbird’). Their respective ground-truth segmentation results have been obtained from a number of video segmentation datasets provided by the authors [30, 52, 53]. A second dataset that is used to evaluate segmentation of salient object is the one provided by Fukuchi et al. [53], available at [http://www.brl.ntt.co.jp/people/akisato/saliency3.html](http://www.brl.ntt.co.jp/people/akisato/saliency3.html). The segmentation dataset has eight challenging videos shot in diverse background settings along with the ground truth segmented foreground information for each frame. It also has stationary as well as tracking shots.

Performance of the spatio-temporal saliency maps is evaluated using a human fixation video dataset [19] that is made available online [http://www.iab.usc.edu/CRCNS-DataShare](http://www.iab.usc.edu/CRCNS-DataShare). The dataset has a number of challenging video sequences that are shot under a wide variety of scenarios. Videos include sequences that are shot in natural settings such as children playing in a park, kids enjoying a game of soccer, synthetic in-game videos of games such as Super-Mario, and TV advertisements and clips from news casts. A few of the videos are combinations of randomly interlaced scenes like sports, news and game videos to study the HVS response to such rapid changes in motion. In total, the dataset contains 50 videos shot with a resolution of 640 × 480, with the number of frames ranging from 164 to 2814 frames. The total duration of the videos in the dataset is about 25 min of with a total of 46 495 frames. The study conducted by Itti [19] consisted of eight distinct subjects who view the clips while an eye-tracker tracks their eye movement. We provide the results of testing the current framework with challenging videos on our website [http://www3.ntu.edu.sg/home2009/KART0028/supplementary2.html](http://www3.ntu.edu.sg/home2009/KART0028/supplementary2.html), as supplementary material where we compared the performance of the proposed salient object detection method with that of colour-based particle filter only. We also provide results of testing the spatio-temporal saliency measure with challenging sequences.

5.1 Evaluation of salient object detection

Salient object detection is evaluated by segmenting the region where the particles are clustered and comparing with a ground-truth segmentation. The location of the centre of the ellipse is chosen as the seed point for region growing segmentation with the mean of the pixel intensities present in the ellipse as the threshold. As described in [28], the quantitative evaluation of segmentation uses a rate of mis-segmentation computed as \( \varepsilon(S) = |\text{GT} \cap S - S| / |\text{GT} \cup S| \), where \( S \) is the segmentation output obtained from the salient object extraction, GT is the ground-truth, \( F \) is the number of frames in the video and \( P \) is the number of pixels in each frame.

Table 2 compares the mis-segmentation rates of the proposed method with other well-known video object segmentation methods. The proposed method, denoted as DF, outperforms other methods in terms of average mis-segmentation rates even with a very simple region growing segmentation technique because of the power of particle filter being guided by the spatio-temporal saliency map and the colour maps. In CL [54], colour homogeneity in the background results in poor segmentation in the ‘Waterski’ and ‘Surfer’ videos. The performance of GC [55] in the ‘Penguin’ sequence is poor as their framework depends largely on the global colour contrast, which, in this case, would yield high saliency measures across the sequence. KS [34] identifies candidate pixels based on persistent motion and appearance cues. Hence, it fails to identify the object of interest in the ‘Fox’ and ‘Waterski’ sequences as these are tracking shots where the objects of interest moves relative to the background, resulting in an image-like appearance and zero motion. Note that the GT segmentation for ‘Penguin’ sequence used in [28] is different from the GT provided in the original segmentation dataset [30].

Fig. 5 compares the segmentation results with the GT of the object extracted from four videos in the dataset [53]. Figs. 5b, d and e are the GT segmentation results provided in the dataset while Figs. 5a, c and f are the results of the
segmentation obtained by using a region growing algorithm [56]. The location of the centre of the ellipse is utilised as the seed point for the region growing algorithm for which the mean of the pixel intensities present in the identified ellipse is utilised as a threshold. The f-measure provides with a quantitative evaluation of the proposed salient object detection method. The precision and recall measures are calculated from the segmented region and the GT information available in the dataset. Precision is calculated as the ratio of the number of foreground pixels that match with the GT foreground segmentation image to the number of pixels in the foreground image while recall is calculated as the ratio of the number of foreground pixels that match with the GT foreground segmentation image to the number of pixels in the GT segmentation result. Our method achieved an average precision of 0.977, recall of 0.804 and f-measure of 0.878. The high f-measure indicates that even with a computationally simple segmentation framework, the proposed framework is able to achieve highly reliable salient extraction while the high average precision indicates its ability to consistently detect the foreground object irrespective of the background clutter.

The proposed salient object detection method can process an average of 53 frames per second for a video with a frame resolution of $352 \times 288$, assuming that the spatio-temporal saliency maps have been pre-computed. This computation speed is achieved by utilising 200 particles for initialising the particle filter. The algorithm is implemented in MATLAB 2012b on a Windows 8 operating system with 4GB RAM and a 2.53 GHz Intel Core2 Duo processor.

5.1.1 Recovering from failed detection: A known limitation of particle filters is the dependence on the size of the sampling window. Objects that move very fast will be missed when their displacement is larger than the windows sampled around the reference particle location. Another limiting case of the proposed framework is shown in Fig. 6 where the objects stop momentarily and subsequently resume their motion. As the spatio-temporal saliency measure employs motion present in the scene to decide on the weights for the temporal saliency and spatial saliency maps, the spatial saliency measure outweighs the motion when the objects in the scene come to a standstill.

However, as described earlier, we avoid fixation on a location by forcing the particles to reinitialise every $\tau$ frames (Table 3). As a result of the reinitialisation, particles that are falsely detected will remain at that position only for a maximum of $\tau$ frames, in the worst case. As we keep the value of $\tau$ to be half the frame rate, an erroneous detection does not remain on the frame for more than half a second,
reducing error propagation. This can be seen in Fig. 6a, shows two frames from a video sequence and the particles corresponding to the most salient object. Fig. 6b shows the spatio-temporal saliency maps corresponding to the frames in Fig. 6a. Even though the object detection is incorrect in the top row owing to the poor saliency measure, it is not allowed to propagate any further through the reinitialisation of the particles as seen in the bottom row.

5.2 Evaluation of spatio-temporal saliency measure

Similar to [6], we adopt the Kullback–Leibler (KL) divergence and area under receiver operating characteristic (ROC) curve to evaluate the performance of the spatio-temporal saliency maps. KL divergence measures the correspondence of the salient regions to human saccade positions; larger the measure, the closer is the proposed saliency model to the human attention mechanism. We adopt the method followed in [6] wherein the saliency value that corresponds to a human saccade position is computed as the maximum of the human saccade positions over a circle of diameter 128 pixels, centred at the human saccade position. The saliency values collected over the entire database are discretised into 10 bins which is subsequently normalised to obtain the probability distribution $P$. The distribution for $Q$ is calculated in a similar manner from spatio-temporal saliency maps. As the positions are sampled randomly, we repeat the experiment 100 times to obtain a fair evaluation of the performance. The KL-divergence measure is the average symmetric KL-divergence measures of the 100 trials. The symmetric KL divergence measure is calculated as

$$
\text{KLDiv}_{\text{sym}}(P, Q) = \frac{1}{2} \text{KLDiv}(P||Q) + \frac{1}{2} \text{KLDiv}(Q||P)
$$

where $\text{KLDiv}(P||Q) = \sum_k P(k) \log(P(k)/Q(k))$, $P$ and $Q$ are the distributions of the human saccade points available from the dataset and the spatio-temporal saliency maps, respectively. Fig. 7 compares the KL-divergence measures calculated for six state-of-the-art spatio-temporal techniques. From Fig. 7, it is evident that the proposed spatio-temporal framework performs better than the best performing state-of-the-art method [12].

It can be seen that the simple algorithm for computing spatio-temporal saliency maps based on distance to dominant features performs better than state-of-the-art saliency measure, RF, both in terms of the area under curve (AUC) and KL-divergence measures. We show the saliency maps generated by different methods and compare it with

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC</td>
<td>0.659</td>
<td>0.633</td>
<td>0.64</td>
<td>0.686</td>
<td>0.636</td>
<td>0.708</td>
<td>0.787</td>
</tr>
</tbody>
</table>

Fig. 7 Illustration of KL divergence results

Model with a larger KL divergence measure indicates that it can better distinguish these two distributions.

www.ietdl.org


© The Institution of Engineering and Technology 2015
the saliency map computed using the proposed spatio-temporal saliency measure in Fig. 8.

For a qualitative evaluation, Fig. 8 compares spatio-temporal saliency maps as well as the salient object detection vis-a-vis human saccade points. In the videos, there is no single salient object. Yet, the location of human saccade points shown as ellipses in the frames in the first column are in agreement with the salient region detected as shown in the last column. Also, the pixel-based salient region detection method (DF) uses a more accurate saliency map as seen in the column marked ‘DF’.

We compare the speed of the proposed spatio-temporal saliency framework with that of the saliency frameworks compared above, based on the number of frames that are processed per second, in Table 4. Although the proposed method is not as fast as PS, the quality of the saliency map is much higher.

6 Conclusion

We have presented an algorithm for salient object detection in videos based on particle filters that uses spatio-temporal saliency maps and colour as cues. The performance is evaluated on segmentation datasets. We also develop a simple algorithm to generate spatio-temporal saliency map that outperforms many state-of-the-art methods. As a future work, we extend the results of the salient object detection framework to intelligently resize frames of a video.

7 References
