A PARTICLE FILTERING APPROACH TO SALIENT VIDEO OBJECT LOCALIZATION

Charles Gray, Stuart James and John Collomosse

Centre for Vision Speech and Signal Processing (CVSSP)
University of Surrey
Guildford, United Kingdom.
Email: {charles.gray | s.james | j.collomosse}@surrey.ac.uk

Paul Asente

Imagination Lab
Adobe Research
San Jose, USA.
Email: asente@adobe.com

ABSTRACT

We describe a novel fully automatic algorithm for identifying salient objects in video based on their motion. Spatially coherent clusters of optical flow vectors are sampled to generate estimates of affine motion parameters local to super-pixels identified within each frame. These estimates, combined with spatial data, form coherent point distributions in a 5D solution space corresponding to objects or parts there-of. These distributions are temporally denoised using a particle filtering approach, and clustered to estimate the position and motion parameters of salient moving objects in the clip. We demonstrate localization of salient object/s in a variety of clips exhibiting moving and cluttered backgrounds.

Index Terms— Video object localization, Moving object segmentation, Particle filter, Tracking.

1. INTRODUCTION

Digital video is ubiquitous in society; vast quantities of professional and user-generated content are generated daily, motivating new techniques to index and visually summarize video assets. Due to high data volumes inherent to video, it is often desirable to simplify these tasks by first identifying the salient objects within video clips. However such preprocessing often makes strong assumptions, e. g. on color, background or scene content, that prohibit their application to the diverse footage found in general purpose video repositories. This paper contributes a novel fully automatic algorithm for salient video object localization using motion cues, that can operate over content exhibiting multiple moving objects and diverse background textures and motions.

The proposed algorithm assumes salient objects to be large and in motion for sustained periods of time under stable i. e. slowly varying or constant motion parameters. Appearance information is also considered, as motion parameters are estimated within visually homogeneous super-pixels that are assumed to correspond to moving objects, or parts thereof. The algorithm measures object motion relative to global motion in a scene, and so compensates for the camera egomotion frequently observed in general footage. Although the contribution of this paper focuses upon the localization and tracking of salient objects, for visualization purposes a subsequent segmentation (e. g. Grab-Cut) may be used to isolate a refine matte of the video object.

Identification of salient objects proceeds as a two step-process. In the first pass, optical flow vectors $V(t)$ are calculated between each frame and its predecessor. To identify potential salient objects (or fragments thereof) present at time t, a subset of vectors $v \in V(t)$ are repeatedly sampled at random. The parameters of an constrained affine (Euclidean) motion model explaining v are inferred via a least squares process. Selection of v is subject to rules promoting the spatial coherence of vectors; specifically, they are selected local to a super-pixel present at t — also selected at random. This results in several sets of motion models each explaining a super-pixel's motion. These models form point clouds in a parameter space that are denoised via a particle filtering technique. The second pass of our process performs unsupervised clustering to group the denoised points, yielding a sequence of motion descriptions for each salient object in the clip.

We describe our algorithm in detail within Sec. 3, and in Sec. 4 apply it to a variety of diverse video footage exhibiting challenging foreground and background motion conditions in many cases containing multiple moving objects.

2. RELATED WORK

Salient video object extraction is a long-standing Computer Vision problem addressing both salient object localization and segmentation; we focus on the former task.

Salient object detection frequently draws upon visual attention heuristics to determine saliency from appearance information. Visual saliency detectors based on biologically inspired filters [1, 2] or computational models such as graphs [3] and sliding window detectors based on relative contrast [4, 5] and geometric cues [6, 7, 8] have been proposed to detect salient objects. Definitions of saliency are often task specific, and so trainable rather than prescribed heuristic measures have also been proposed [9, 5, 10].

Although such measures may be trivially applied to independent video key-frames, pixel-wise image saliency has also been extended to video through spatio-temporal analysis, e. g. patch based rarity [11] was extended to video to

detect objects with unusual movement patterns [12]. Low-level spatio-temporal filtering has been post-processed in a bottom-up manner to develop more sophisticated salient object detectors, which simultaneously localize and estimate motion parameters. Tapu et al. [13] use RANSAC to recursively filter correspondences between sparsely detected SIFT keypoints, filtered to remove non-salient points under a visual salience measure, to identify coherently moving objects. RANSAC has also been used more generally to refine the accuracy of optical flow fields [14]. Our method also adopts a random sampling approach to derive rigid body motion estimates. However we sample dense motion vectors rather than sparse keypoint correspondences, and encourage spatial coherence by sampling within superpixel boundaries rather than hierarchically deriving coherent sub-regions using RANSAC. Motion vector analysis has been used elsewhere for grouping moving pixels into objects based on spatio-temporal parameters [15] or vector magnitude and phase [16]. Probabilistic frameworks for aggregating vectors in a Markov random field [17] and tracking these over time [18] have been explored. Aggregations of mid-level primitives to form coherent salient objects under an energy maximization scheme was proposed in [19]. In our work we analyze motion vectors to determine the motion of individual super-pixels and aggregate these in space-time using mean-shift [20].

3. SALIENT OBJECT EXTRACTION

Motion is the primary cue for identifying salient objects under our framework. We initially pre-process each video frame independently, computing a dense set of optical flow vectors $V(t)$ between the set of pixel locations $I(t)$ within each frame, and those in its immediate predecessor $I(t-1)$. Without loss of generality we use the dense optical flow estimation algorithm of Brox et al. [21].

3.1. Camera motion compensation

Video clips frequently contain camera movement that results in global motion within the frame. These must be compensated for, in order to analyze the local motion of objects. As with prior work seeking to compensate for such motion [13], we model inter-frame camera movement as a homography $H(t)$ which we solve for each frame by minimizing:

$$H(t) = \underset{H}{\operatorname{argmin}} \sum_{\forall\{a \in I(t), b_a \in V(t)\}} |Ha - (a + b_a)| - |H^{-1}(a + b_a)' - a|. \quad (1)$$

where a is a point in $I(t)$, and b_a is its corresponding optical flow vector in $V(t)$, s.t. $a + b_a \in I(t+1)$. The minimization is performed via a RANSAC process in which a subset of $V(t)$ are repeatedly selected at random and used to obtain a candidate $H(t)$, which is then tested against all $V(t)$ via (1). The process yields a set of camera-motion compensated flow vectors $V'(t) = HV(t)$ for subsequent processing. We process only significant vectors where $|V'(t)| < \epsilon$.

3.2. Inter-frame motion estimates

We estimate of a set of motion parameters for moving objects at each time-step. These parameters are later (Sec. 3.3) tracked over time to remove sporadic object detections, and so identify temporally significant objects. We have opted for independent processing of time-steps, followed by an tracking and integration step (i.e a $2D+t$ approach) over a full spatial-temporal ($3D$ volumetric) representation to reduce complexity when dealing with lengthy clips.

For a given t, we repeatedly sample (with replacement) a set of pixel locations $p \in I(t)$ and associated optical flow vectors $v_p \in V'(t)$ from which we infer a Euclidean transformation $A(p, v_p)$ that best explains the motion of set v_p:

$$A(p, v) = \underset{A}{\operatorname{argmin}} \sum_{p, v_p} \parallel Ap - v_p \parallel . \quad (2)$$

where A is a rotation and translation, and $\parallel . \parallel$ the L^2 norm:

$$A = \begin{bmatrix} \cos\theta & -\sin\theta & T_x \\ \sin\theta & \cos\theta & T_y \\ 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

The parameter tuple $\{\theta, T_x, T_y\}$ is computed from the input sets of 2D column vectors (p, v_p) as follows:

$$p' = p - \frac{1}{|p|} \sum_{i=1}^{|p|} p_i. \quad (4)$$

$$v'_p = v - \frac{1}{|v_p|} \sum_{i=1}^{|v_p|} v_{p_i}. \quad (5)$$

$$M = \sum_{i=1}^{|p|} p'_i {v_{p_i}}'^T. \quad (6)$$

$$R = M(M^T M)^{\frac{1}{2}}. \quad (7)$$

yielding R the 2×2 upper-left of A from which θ is readily obtained via arc-tangent, and

$$s = \sqrt{\frac{1}{|v_p|} \sum_{i=1}^{|v_p|} v'_{p_i} / \frac{1}{|p|} \sum_{i=1}^{|p|} p'_i}. \quad (8)$$

$$\begin{bmatrix} T_x \\ T_y \end{bmatrix} = \frac{1}{|v_p|} \sum_{i=1}^{|v_p|} v'_{p_i} - R\frac{s}{|p|} \sum_{i=1}^{|p|} p'_i. \quad (9)$$

Points p are chosen to lie within spatially coherent regions (super-pixels) obtained via [22], preventing the motion parameter estimate being drawn from multiple targets. The first point sampled for inclusion to p is drawn from $V'(t)$. Subsequent points are sampled from the subset of $V'(t)$ that

fall within the same super-pixel as the first point. Typically we work with fewer than 100 super-pixels per frame, each of variable size around 1000 pixels. Note p are drawn from all super-pixels within the frame with $|V'(t)| > 0$.

The outcome of the iterative sampling and Euclidean motion estimation process is a set of transformations $\{A(p_1, v_{p_1}), \ldots, A(p_n, v_{p_n})\}$ that describe each sampling. In practice we use $|p| = 20$ samples (i. e. $|p| \ll |V'(t)'|$) and $n = 100$ iterations. We augment the 3 parameters of $A(p_i, v_{p_i})$ with the centroid of p_i i. e. $(\mu_x, \mu_y) = \sum_1^{|p|} p_i$ yielding a point in 5D space $(\mu_x, \mu_y, \theta, T_x, T_y)$ that describes both the motion and position of p at time t.

Thus after sampling n iterations we obtain a set of 5D points, written $\mathcal{A}(t)$ that describe the motion and position of moving objects present at t. Fig. 1 illustrates a set of such estimates derived from a single frame. Obtaining a distribution of estimates for object motion is preferable to deriving a single estimate from all vectors, since optical flow generates frequent outliers in real-world data [14].

3.3. Particle Filtering of $\mathcal{A}(t)$

We refine the noisy set of motion models $\mathcal{A}(t)$, obtained on a per-frame basis, by filtering out those corresponding to short-lived or erratically moving objects which we assume to be non-salient. This is achieved by tracking the 5D cloud of motion estimates over time using a particle filter [23].

3.3.1. Framework

We define a set of m particles for each frame, written $X^t = \{x_t^1, x_t^2, ..., x_t^m\}$ with super-script indicating the index, within the 5D space $(\mu_x, \mu_y, \theta, T_x, T_y)$. The particles describe the spatio-temporal attributes of moving objects in the video. These are the hypotheses, and are computed progressively for each frame using hypotheses from the previous frame X_{t-1} and observed data from the video $\mathcal{A}(t)$. For convenience we use notation $\mathcal{A}(t) = \{z_t^1, z_t^2, ..., z_t^n\}$ to denote the latter. Note that X_t and $\mathcal{A}(t)$ are maintained separately despite being defined in the same 5D space. In our implementation we use $m = 500$ particles.

Each hypothesis has associated with it a prior probability $p(x_t^i)$ representing the likelihood that the hypothesis describes the motion of a salient object. At $t = 1$, X^1 are initialized at random within $\Re^5$ and $p(x_1^i) = \frac{1}{m}$ sets a uniform prior.

At each time-step, the posterior for each hypothesis is:

$$p(x_t^i|\mathcal{A}(t)) \quad \propto \quad p(x_{t-1}^i)p(\mathcal{A}(t)|x_t^i). \tag{10}$$

where,

$$p(\mathcal{A}(t)|x_t^i) \quad = \quad 1 - \frac{1}{|J|}\sum_{j \in J}\mathcal{N}(|x_t^i - z_t^j|; \Sigma). \tag{11}$$

and $J \subseteq \mathcal{A}(t)$ s.t. $|z_t^j - x_t^i| < T$, i. e. J indicates the subset of motion models local to hypothesis x_t^i. $\mathcal{N}$ indicates a normal

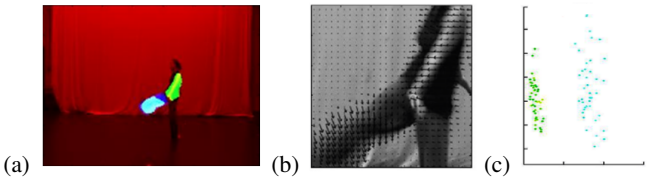

Fig. 1. Illustrating the clustering of different moving objects (or parts thereof) into temporally coherent groups in the 5D parameter space. (a) source; (b) optical flow; (c) 5D clusters visualized via PCA projection. Sequence: $DANCER$.

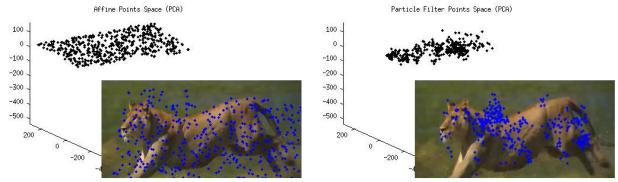

Fig. 2. Illustrating the improved spatio-temporal coherence of the 5D distribution derived from a frame before (left) and after (right) particle filtering. Blue points indicate spatial samples (left) and particle positions (right). 5D plots visualized via PCA projection. Sequence: $SAFARI$.

variate with a specified mean and a covariance Σ. Parameters T and Σ are set empirically to 10^5 and 10 respectively, encoding an assumption of expected change in 5D space-time motion parameters over one time step.

3.3.2. Iterative process

Under the above framework, particle filtering proceeds as follows. First, a population of hypotheses X_t is computed by sampling m hypotheses stochastically from X_{t-1} with a bias to $p(x_{t-1}^i)$. Under the above framework, particle filtering proceeds as follows. First, a population of hypotheses X_t is computed by sampling m hypotheses stochastically from X_{t-1} with a bias to $p(x_{t-1}^i)$.

Second, the 5D position of these hypotheses are updated through the addition of Gaussian noise to inject diversity:

$$x_t \leftarrow x_t + \mathcal{N}(0; \Sigma). \tag{12}$$

Third, the posterior probabilities for X_t are evaluated against the data $\mathcal{A}(t)$ for that frame via (10). The prior probabilities of X_t are then updated:

$$p(x_t^i) \leftarrow p(x_t^i|\mathcal{A}(t)). \tag{13}$$

The result is a set of filtered motion estimates X_t that tend to cluster around temporally stable estimates within $\mathcal{A}(t)$. Fig. 2 illustrates the signal of Sub-sec. 3.2 before (i. e. $\mathcal{A}(t)$) and after (i. e. X_t) filtering.

Note that for clarity we described particle filtering as a separate process following Sub-sec. 3.2. In practice both processes require data only from t and $t-1$ and so can be run in tandem, in a single pass as the video clip is processed.

Fig. 3. Representative results: salient objects identified in single and multi-object videos with moving backgrounds. Ground truth (green), proposed (red/yellow), Alexe et al. [8] (blue). Sequences: CAR (top); $HORSE$ (middle); $MULTI$ (bottom).

3.4. Object Clustering

The final stage of our process is to cluster the filtered motion estimates X into distinct salient objects. We do so by running the mean-shift [20] clustering algorithm over a $6D$ representation of hypotheses stored from all time instants, comprising the 5 dimensions of X_t plus time, i. e. $(\mu_x, \mu_y, \theta, T_x, T_y, t)$. Typically this results in a grouping that identifies independent salient objects within the sequence, however temporal oversegmentation due to long or complex trajectories can occur. This can be resolved by aggregating pairs of clusters where over half of the points in their distributions arise from the same tracked particle.

4. RESULTS AND DISCUSSION

We have evaluated over a wide range of sports and wildlife clips containing several hundred frames of single and multiple moving objects, with either a static or panning camera.

Fig. 3 presents representative visual results, comparing against manually generated ground-truth bounding boxes (BBs), and BBs returned by Alexe et al. [8]; a state-of-the-art salient object detector also designed for operation on diverse footage. Since Alexe et al. generates several BBs we use the most likely BB returned by the method. In all cases we are qualitatively closer to the ground-truth BB and retain a consistent lock on the object (or objects) whereas Alexe et al. sporadically changes lock to different objects in the scene including non-salient objects such as the bushes (CAR) or fence ($HORSE$) in the background. This is because Alexe et al. do not enforce temporal coherence.

Fig 4 (top) quantifies the performance of both methods against a groundtruth using the ratio $\frac{A \cap G}{A \cup G}$, where A is the BB returned by the algorithm being evaluted and G is the ground-truth BB. The relative performance improvement versus [8] for 4 single object clips is $\sim$52% (CAR) $\sim$84% ($DANCER$) $\sim$80% ($HORSE$) $\sim$26% ($SAFARI$). In the latter case, dust clouds cause occlusion and non-salient motion that confuse both methods. We do not compare multiple objects vs. [8] as the choice of comparison BB is subjective.

Fig. 4. Top: Quantitative performance vs. Alexe et al. [8]. Bottom: Mattes pulled from video using GrabCut [24] over BBs returned by the proposed method.

Fig 4 (bottom) illustrates an application of our method to video summarization, in which GrabCut [24] is used to generate temporally coherent video mattes of salient objects.

5. CONCLUSION

We have presented a novel unsupervised algorithm for simultaneously detecting and localizing salient moving objects within a video, and estimating their motion parameters. We are able to identify single or multiple objects per clip and track these over several hundred frames. Such tracks may be used to pull mattes from the video e. g. for the purposes of summarization, which forms our main direction for future work. A current weakness is that stationary objects with respect to the background are not considered. We will explore appearance cues to address this in future work.

6. REFERENCES

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

[2] Y. Hu, X. Xie, W-Y. Ma, L-T. Chia, and D. Rajan, "Salient region detection using weighted feature maps based on the human visual attention model," in *Proc. Pacific Rim Conf. on Multimedia*, 2004, pp. 993–1000.

[3] J. Harel, C. Koch, and P. Perona, "Graph based visual saliency," *Advances in Neural Information Systems*, pp. 545–554, 2007.

[4] Y. Ma and H. Zhang, "Contrast based image attention analysis by using fuzzy growing," in *Proc. ACM Multimedia*, 2003, pp. 384–381.

[5] T. Liu, J. Sun, N. Zheng, X. Tang, and H. Shum, "Learning to detect a salient object," in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[6] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[7] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proc. Intl. Conf. on Computer Vision (ICCV)*, 2011.

[8] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[9] P. Hall, M. Owen, and J. Collomosse, "A trainable low-level feature detector," in *Proc. Intl. Conference on Pattern Recognition (ICPR)*, 2004, vol. 1, pp. 708–711.

[10] T. Liu, N. Zheng, W. Ding, and Z. Yuan, "Video attention: Learning to detect a salient object sequence," in *Proc. Intl. Conference on Pattern Recognition (ICPR)*, 2008, vol. 1, pp. 1–4.

[11] K. Walker, T. Cootes, and C. Taylore, "Locating salient object features," in *Proc. British Machine Vision Conf. (BMVC)*, 1998, pp. 557–566.

[12] O. Boiman and M. Irani, "Detecting irregularities in images and in video," *Intl. Journal of Computer Vision*, vol. 1, no. 74, pp. 17–31, 2007.

[13] R. Tapu, B. Mocanu, and E. Tapu, "Salient object detection in video streams," in *Proc. Intl. Symp. on Elec. and Telecom. (ISETC)*, Nov. 2012, pp. 275–278.

[14] Z. Lin S. Cohen Y. Wu Z. Chen, H. Jin, "Large displacement optical flow from nearest neighbor fields," in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[15] J. Wang and E. Adelson, "Representing moving images with layers," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 625–638, 1994.

[16] A. Belardinelli, F. Pirri, and A. Carbone, "Motion saliency maps from spatiotemporal filtering," *Proc. Attention in Cog. Sys. In Lecture Notes in Artificial Intelligence (Spring LNAI)*, pp. 112–123, 2009.

[17] W.-T. Li, H.-S. Chang, K.-C. Lien, H.-T. Chang, and Y.-F. Wang, "Exploring visual and motion saliency for automatic video object extraction," *IEEE Trans. on Image Processing*, vol. 22, no. 7, pp. 2600–2610, July 2013.

[18] K. Fukuchi, K. Miyazata, A. Kimura, and S. Takagi, "Saliency-based video segmentaiton with graph cuts and sequentially updated priors," in *Proc. Intl. Conf. on Multimedia and Expo (ICME)*, 2009.

[19] F. Guraya, F. Cheikh, A. Tremeau, Y. Tong, and H. Konik, "Predictive saliency maps for surveillance videos," in *Proc. Conf. on Distributed comp. and App. to Business Engineering and Science*, 2010, pp. 508–513.

[20] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002.

[21] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. European Conference on Computer Vision (ECCV)*, May 2004.

[22] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 11, pp. 2274–2282, 2012.

[23] Michael Isard and Andrew Blake, "CONDENSATION - conditional density propagation for visual tracking," *Intl. Journal of Computer Vision*, vol. 29, pp. 5–28, 1998.

[24] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, Aug. 2004.