VIDEO SALIENT OBJECT DETECTION VIA CROSS-FRAME CELLULAR AUTOMATA

Jingfan Guo[1], *Tongwei Ren*[1,*], *Lei Huang*[1], *Xingyu Liu*[1], *Ming-Ming Cheng*[2], *Gangshan Wu*[1]

[1] State Key Laboratory for Novel Software Technology, Nanjing University, China
[2] Media Computing Lab, CCCE & CS, Nankai University, China
guojf@smail.nju.edu.cn, {rentw, leihuang}@nju.edu.cn, liuxingyu@smail.nju.edu.cn,
cmm@nankai.edu.cn, gswu@nju.edu.cn

ABSTRACT

Salient object detection aims to detect the attractive objects on images and videos. In this paper, we propose a novel salient object detection method for videos based on cross-frame cellular automata. Given a video, we first represent the video frames with super-pixels, and construct a saliency propagation network among super-pixels within a frame and between adjacent frames based on their appearance similarities and temporal coherency. Second, we initialize the saliency map of each frame with the fusion of two saliency maps generated by appearance and motion features independently. Finally, we utilize cellular automata updating to propagate saliency among super-pixels iteratively and generate the coherent saliency maps with complete objects. The experimental results show that our method outperforms the state-of-the-art methods on different types of videos.

Index Terms— Salient object detection, video saliency, cross-frame cellular automata, saliency propagation network

1. INTRODUCTION

Served as a fundamental of various multimedia applications, salient object detection aims to detect the attractive objects on images and videos [1]. It is widely used in content-aware editing [2], information retrieval [3], social computing [4] and so on. In decades, many methods are proposed for detecting salient objects on images effectively [5–7], but the study of salient object detection on videos is still insufficient [8, 9].

Compared to salient object detection on images, video salient object detection faces two challenges. One challenge is that object motion usually plays an important role in salient object detection on videos, because it represents the temporal contrast of objects to background [10]. The other challenge is that the appearances of salient objects in videos are variable, which makes it difficult to obtain coherent salient objects over frames [11]. Hence, simply applying image salient object methods on video frames cannot obtain satisfactory performance.

To overcome the aforementioned challenges, we propose a novel video salient object detection method using cross-frame cellular automata. Cellular automata can update the state of each cell in terms of its current state and the states of the cells in its neighborhood [12], which has shown its effectiveness in saliency map refinement for image salient object detection [6]. In the proposed method, we apply cellular automata updating on a cross-frame saliency propagation network to refine the saliency map of each video frame and generate coherent saliency maps with complete objects. Figure 1 shows an overview of the proposed method. We first represent each video frame with super-pixels and construct a cross-frame saliency propagation network based on the appearance similarities and motion coherency among the super-pixels within a frame and between adjacent frames. Second, we calculate two saliency maps for each video frame based on its appearance and motion features, and fuse them to initialize saliency propagation network. Finally, we apply cellular automata updating on the initialized saliency propagation network to refine the saliency maps of all the frames iteratively until the final saliency maps are generated.

Our contributions mainly include:

- We apply cellular automata in video salient object detection for the first time by constructing a cross-frame saliency propagation network, which helps to generate coherent saliency maps for videos.
- We utilize both appearance and motion features followed by entropy-based adaptive fusion in saliency map initialization, which can handle both static and moving salient objects.
- We construct a dataset, named NoMot, containing 10 videos without obvious camera motion or object motion to complement evaluation, and validate the proposed method on it together with two public datasets. It shows that our method outperforms the state-of-the-art methods on different types of videos.

This work is supported by National Science Foundation of China (61321491, 61202320, 61572264, 61620106008), and Collaborative Innovation Center of Novel Software Technology and Industrialization.

978-1-5090-6067-2/17/$31.00

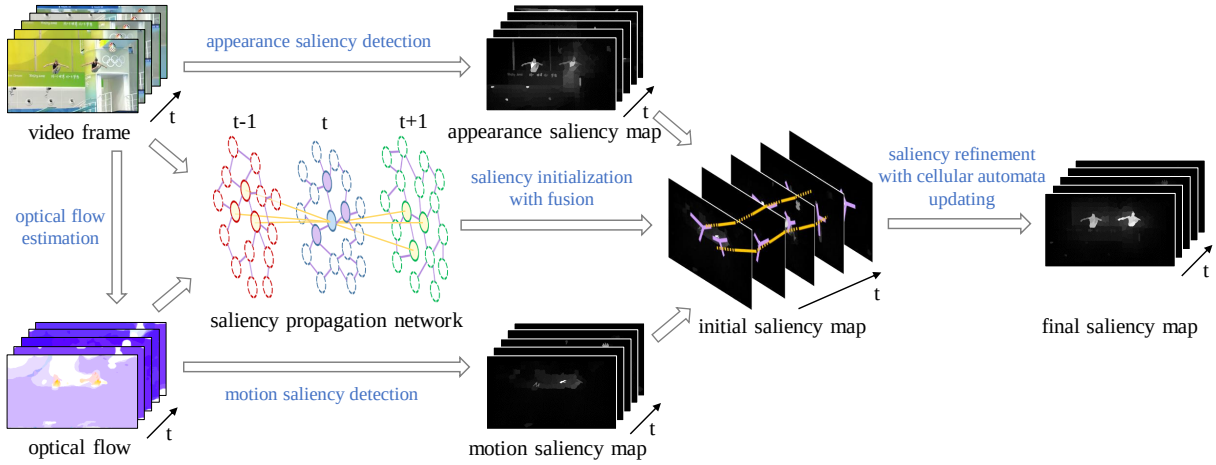

Fig. 1. An overview of the proposed video salient object detection method.

2. RELATED WORK

2.1. Image salient object detection

Salient object detection methods on images mainly depend on appearance features, such as color contrast. For example, Achanta *et al.* [13] detected salient objects based on the distance between the color of each pixel and the average color of the image. Cheng *et al.* [14] weighted global color contrast with spatial distance to emphasize saliency locality. Image content location is also considered to refine saliency detection performance, such as center bias [15] and boundary bias [5].

Besides color and location, depth is used in salient object detection on RGB-D images. For instance, Feng *et al.* [7] proposed a novel RGB-D saliency feature, named local background enclosure, to directly measure salient structure from depth. Guo *et al.* [16] initialized saliency maps with the fusion of color saliency and depth saliency, and propagated saliency among super-pixels for refinement.

Recently, graph-based models are widely used in salient object detection. Specifically, Yang *et al.* [17] generated a coarse saliency map by background modeling and propagated saliency by manifold ranking. Qin *et al.* [6] generated the initial saliency map by integrating color distinction and spatial distance against boundary, and refined saliency map with cellular automata. Zhang *et al.* [18] employed multiple graphs and modeled the visual rarity in the optimization framework to satisfy the requirement of saliency detection.

2.2. Video salient object detection

Different to fixation prediction on videos [19–21], video salient object detection focuses on extracting complete salient objects by exploring the spatio-temporal characteristics of video content. Seo *et al.* [22] measured the likeness of each pixel to its surroundings using local regression kernel and computed its saliency with self-resemblance. Liu *et al.* [23] measured the spatial and temporal saliency based on global contrast, spatial sparsity and motion distinctiveness, and fused them to generate the saliency maps. Huang *et al.* [9] classified the trajectories with SVM to remove the dominant camera motion, and calculated the saliency of each trajectory by diffusing it to its surrounding regions. Wang *et al.* [10] calculated the geodesic distance from each super-pixel to the frame boundaries to measure its object probability. Liu *et al.* [11] proposed a saliency model for unconstrained videos based on super-pixel level graph and spatiotemporal propagation.

3. OUR METHOD

3.1. Saliency propagation network construction

An effective video salient object detection method requires to provide coherent saliency maps on all the video frames, i.e., if a region on some video frame is indicated as a salient object, the corresponding regions on other video frames should also be indicated as a salient object with similar saliency. To generate such coherent saliency maps, we construct a saliency propagation network $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ crossing all video frames.

In the construction of $\mathcal{G}$, we treat the super-pixels in video frames as the network nodes. Given a video frame F^t, we over-segment it into super-pixels using simple linear iterative clustering algorithm [24], and represent frame F^t with a super-pixel set $\mathcal{V}^t = \{p_i^t\}$, where p_i^t is the super-pixel in frame F^t and $\mathcal{V}^t$ is a subset of $\mathcal{V}$.

We use large displacement optical flow algorithm [25] to generate optical flow between adjacent frames, which describes the motion of each pixel from one frame to the subsequent one. Given the video frames and optical flows, we extract three features for each super-pixel: average color on $L^*a^*b^*$ color space, average motion vector, and 32-D motion histogram with 8 orientations and 4 velocities.

For each super-pixel p_i^t in frame F^t, we match it with the super-pixels in the adjacent frames. For the super-pixels in each frame are generated independently, one super-pixel may not be exactly matched to another super-pixel in its adjacent frame. To address the problem, we use soft-matching strategy. Here, we take matching p_i^t to the super-pixels in frame F^{t+1} as an example. For each pixel within p_i^t, we determine its corresponding position in frame F^{t+1} according to its optical flow, and find the super-pixel in F^{t+1} which the corresponding position belongs to. For each super-pixel p_j^{t+1} in frame F^{t+1}, we calculate the ratio of the pixels within p_i^t whose corresponding positions belong to p_j^{t+1}:

$$\rho_{i,j}^{t,t+1} = \frac{|\Phi(p_i^t, p_j^{t+1})|}{|p_i^t|}, \tag{1}$$

where $\Phi(p_i^t, p_j^{t+1})$ is a set of the pixels within p_i^t whose corresponding positions belong to p_j^{t+1}; $|\cdot|$ denotes the set cardinality.

Considering the inaccuracy of optical flow, we only retain the high-confidence matching relationships to avoid mismatching, i.e., the matching relationship from p_i^t to p_j^{t+1} is retained when $\rho_{i,j}^{t,t+1}$ is larger than a threshold τ_{int}, which equals 0.3 in our experiments. Finally, we normalize the matching relationship from p_i^t to p_j^{t+1}:

$$\gamma_{i,j}^{t,t+1} = \begin{cases} \frac{\rho_{i,j}^{t,t+1}}{\sum_{\Omega_i^{t,t+1}} \rho_{i,k}^{t,t+1}}, & \rho_{i,j}^{t,t+1} \in \Omega_i^{t,t+1} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\Omega_i^{t,t+1} = \{\rho_{i,k}^{t,t+1} | \rho_{i,k}^{t,t+1} > \tau_{int}\}$. Thus, we have $\mathcal{E}^{t,t+1} = \{(p_i^t, p_j^{t+1}) | \gamma_{i,j}^{t,t+1} > 0\}$, where $\mathcal{E}^{t,t+1}$ is a subset of $\mathcal{E}$ with the edges from F^t to F^{t+1}. Similarly, we match p_i^t to the super-pixels in F^{t-1} for bi-directional saliency propagation.

Inspired by [6], saliency propagation among super-pixels within a video frame is beneficial to refine its saliency map. Hence, we supplement the relationships between super-pixels within a frame to the saliency propagation network, $\mathcal{E}^t = \{(p_i^t, p_j^t) | p_i^t \text{ is adjacent to } p_j^t\}$. The weight of edge between p_i^t and p_j^t is:

$$\gamma_{i,j}^t = \Big(\frac{\boldsymbol{m}_i^t \cdot \boldsymbol{m}_j^t}{2\|\boldsymbol{m}_i^t\|\|\boldsymbol{m}_j^t\|} + \frac{1}{2}\Big) \cdot \exp\Big(-\lambda \|\boldsymbol{c}_i^t - \boldsymbol{c}_j^t\|_2\Big), \tag{3}$$

where $\boldsymbol{m}_i^t$ and $\boldsymbol{m}_j^t$ are the average motion vectors of p_i^t and p_j^t; $\boldsymbol{c}_i^t$ and $\boldsymbol{c}_j^t$ are the average $L^*a^*b^*$ color of p_i^t and p_j^t, respectively; λ equals 10 in our experiments.

In this way, we construct a saliency propagation network which can propagate saliency among super-pixels within a frame and between adjacent frames.

3.2. Saliency map initialization

We initialize the saliency propagation network by independently generating a saliency map for each video frame based on the intra-frame sub-network $\mathcal{G}^t = \{\mathcal{V}^t, \mathcal{E}^t\}$. In saliency map initialization, we utilize appearance and motion features separately to generate saliency maps and fuse them together, which can handle both static salient objects and moving salient objects.

For appearance saliency, we utilize a saliency detection method based on boundary connectivity and color contrast [5]. The appearance saliency of p_i^t is denoted as $S_{a,i}^t$.

For motion saliency, we consider that the super-pixels $\mathcal{V}_b^t = \{p_b^t | p_b^t \text{ is on the boundary of } F^t\}$ are more likely to be background, so their motion feature could approximately represent the background motion. To each super-pixel p_i^t, we calculate the geodesic distance, i.e., the accumulated edge weights along its shortest path, to super-pixels in $\mathcal{V}_b^t$, and choose the minimum geodesic distance as its motion saliency:

$$S_{m,i}^t = \min_{p_b^t \in \mathcal{V}_b^t} \Big(\min_{v_1 = p_i^t, v_2, \ldots, v_n = p_b^t} \sum\nolimits_{k=1}^{n-1} \omega(v_k, v_{k+1}) \Big), \quad s.t. (v_k, v_{k+1}) \in \mathcal{E}^t \tag{4}$$

where $\omega(\cdot,\cdot)$ is defined as:

$$\omega(p_i^t, p_j^t) = \exp\Big(\lambda \cdot \chi^2(\boldsymbol{h}_i^t, \boldsymbol{h}_j^t)\Big) \tag{5}$$

in which $\boldsymbol{h}_i^t$ and $\boldsymbol{h}_j^t$ are the 32-D motion histograms of p_i^t and p_j^t, respectively; $\chi^2(\cdot,\cdot)$ is chi-squared distance between two histograms; λ equals 10 in our experiments.

To fuse the two saliency maps generated by appearance and motion features, we propose an entropy-based adaptive fusion strategy, which conforms to the intuition that a saliency map with smaller entropy and average value tends to have higher confidence and hence deserves larger fusion weights:

$$S_i^t = \begin{cases} \delta_a^2 \cdot S_{a,i}^t + (1 - \delta_a^2) \cdot S_{m,i}^t, & \delta_a \le \tau_l \\ S_{a,i}^t \cdot S_{m,i}^t, & \tau_l < \delta_a < \tau_h \\ (1 - \delta_m^2) \cdot S_{a,i}^t + \delta_m^2 \cdot S_{m,i}^t, & \delta_a \ge \tau_h \end{cases} \tag{6}$$

where $\delta_a = \frac{\overline{S}_m \cdot E(S_m)}{\overline{S}_m \cdot E(S_m) + \overline{S}_a \cdot E(S_a)}$, in which S_s and S_m are pixel-level saliency maps with average value $\overline{S}_s$ and $\overline{S}_m$, respectively; $\delta_m = 1 - \delta_a$; $\tau_l = 0.4$ and $\tau_h = 0.6$ in our experiments. The entropy of a saliency map S is :

$$E(S) = -\sum\nolimits_{k=1}^{K} \log_2 n_k \tag{7}$$

where n_k is the number of pixels with saliency level k; K is the total number of saliency levels, which equals 256.

3.3. Saliency refinement with cellular automata updating

Based on the initialized saliency propagation network, we utilize cellular automata updating [12] to refine the saliency maps by propagating saliency among super-pixels. Inspired by [6], we treat our saliency propagation network as a cellular automata, in which the states of cells correspond to the

saliency values of super-pixels. Different from conventional cellular automata whose cells have finite number of states, saliency in our network is continuous-valued. Our network is iteratively updated to make each super-pixel have similar saliency value to its similar neighbors.

For three consecutive frames, we have the sub-network $\mathcal{G}^{t-1,t,t+1} = \{\mathcal{V}^{t-1} \bigcup \mathcal{V}^{t} \bigcup \mathcal{V}^{t+1}, \mathcal{E}^{t-1,t} \bigcup \mathcal{E}^{t} \bigcup \mathcal{E}^{t,t+1}\}$. The saliency value of p_i^t on the middle frame after an iteration is updated to:

$$S_i^{t*} = \eta_i S_i^t + \frac{1-\eta_i}{\sum_j \gamma_{i,j}^{t,t-1} + \sum_k \gamma_{i,k}^{t} + \sum_l \gamma_{i,l}^{t,t+1}} \cdot \left(\sum_j \gamma_{i,j}^{t,t-1} S_j^{t-1} + \sum_k \gamma_{i,k}^{t} S_k^{t} + \sum_l \gamma_{i,l}^{t,t+1} S_l^{t+1} \right), \quad (8)$$

where S_i^t is the saliency value of p_i^t before this iteration; S_j^{t-1}, S_k^t and S_l^{t+1} are the saliency values of p_i^t's neighbors in the previous, current and next frames, respectively; $\gamma_{i,j}^{t,t-1}$ and $\gamma_{i,l}^{t,t+1}$ are given by Eq. (2); $\gamma_{i,k}^{t}$ is defined in Eq. (3); η_i is a parameter to balance the influence of p_i^t's own saliency value and the saliency values of its neighbors, which is defined as:

$$\eta_i = \alpha \left(\max \left\{ \max_j \gamma_{i,j}^{t,t-1}, \max_k \gamma_{i,k}^{t}, \max_l \gamma_{i,l}^{t,t+1} \right\} \right)^{-1} + \beta, \quad (9)$$

where α and β are parameters to retain propagation stability, which equal 0.6 and 0.2 in our experiments.

We propagate the saliency value among super-pixels iteratively until obtaining stable saliency maps for all the video frames or reaching a predefined iteration number, which equals 10. For the existence of incomplete and omitted salient objects in initialization, the saliency of salient objects may be not high in value. Similarly, the highlighted background in initialization may cause the residual saliency in background, which may reduce the saliency difference between salient object and background. Hence, We globally normalize the saliency maps of all the frames to generate the final salient object detection result.

4. EXPERIMENTS

4.1. Datasets and experiment settings

Two public datasets, SegTrackV2 [26] and UVSD [11], are used in the performance validation. SegTrackV2 contains 14 videos with various scenes and diverse motion activities, and UVSD contains 18 unconstrained videos with complicated motion and complex scenes. Considering these two datasets emphasize motion cues more than appearance cues in saliency detection, we construct a dataset with 10 videos without obvious camera motion or object motion, named NoMot, for comprehensive evaluation. The groundtruth on each video frame is manually labelled using Adobe Photoshop.

All the experiments are carried out on a computer with i7 3.5GHz CPU and 32GB memory.

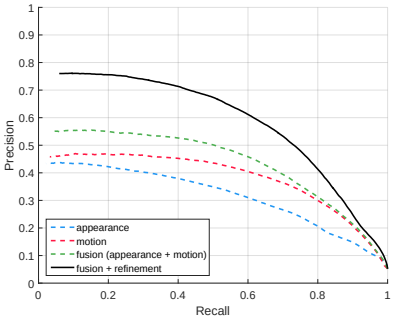

Fig. 2. Validation of each step in our method on UVSD.

4.2. Ablation study

We first validate the effectiveness of each step in our method: appearance saliency detection, motion saliency detection, saliency fusion and saliency refinement. Figure 2 shows the validation result on UVSD. We can see that the fused saliency obtains better performance than both appearance saliency and motion saliency, and saliency refinement further improves the performance. It shows that each step in our method is effective for generating the final saliency maps.

4.3. Comparison

To illustrate the effectiveness of our method, we compare it with six state-of-the-art salient object detection methods on videos: CE [8], DCMR [9], GD [10], SGSP [11], SP [23] and SR [22]. For all the compared methods, we use the default settings suggested by the authors.

Figure 3 shows three examples of salient object detection results generated by different methods. The top, middle and bottom sample videos are from NoMot, SegTrackV2 and UVSD, respectively. It shows that our method can obtain good saliency detection performance on different types of videos. In contrast, the compared methods may fail in some situations. For instance, SGSP may obtain random saliency map on the videos without obvious object motion (top example), GD may mix a salient object and background when they have similar colors (middle example), and DCMR may omit a salient object completely on some video frame (bottom example).

Figure 4 shows the comparison results of our method and other methods with PR curve on three datasets, and Table 1 shows the comparison of weighed F_β-measure F_β^ω ($\beta^2 = 0.3$ to follow common practice) [27]. It shows that our method outperforms other methods on all the datasets in most cases, except that GD performs better than us on SegTrackV2. It is mainly caused by the defect of our method that residual saliency may appear in background regions after propagation.

As shown in Table 1, we compare the efficiency of our method with other methods on running time per frame. Our efficiency is similar to that of SGSP and GD. The reason is that SGSP, GD and our method all rely on LDOF

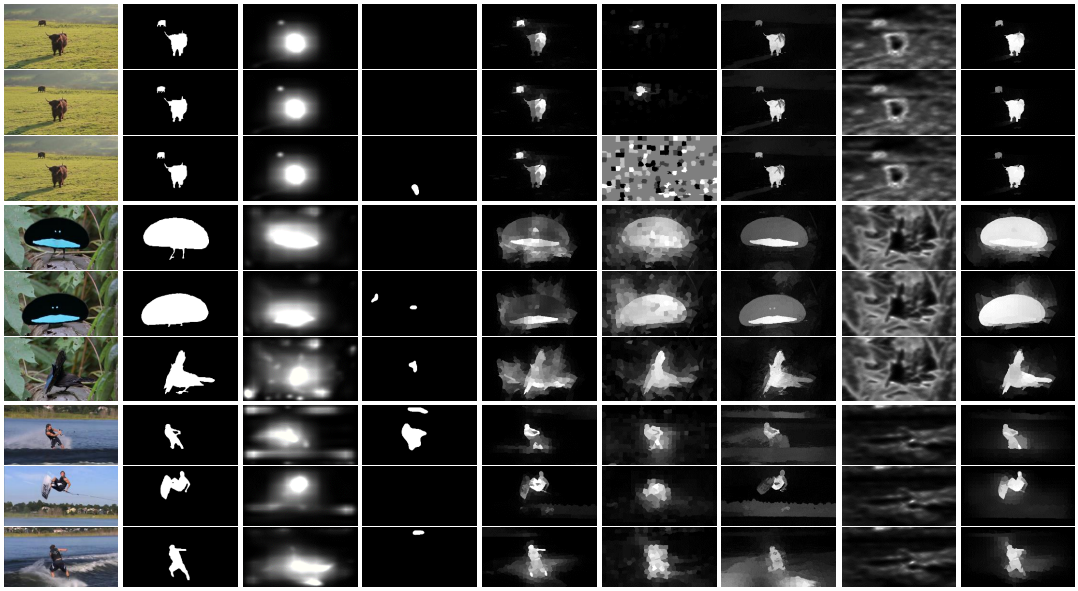

Fig. 3. Examples of the salient object detection results of different methods. (a) Video frames. (b) Manually labelled groundtruth. (c)-(h) Results of CE [8], DCMR [9], GD [10], SGSP [11], SP [23] and SR [22], respectively. (i) Our results.

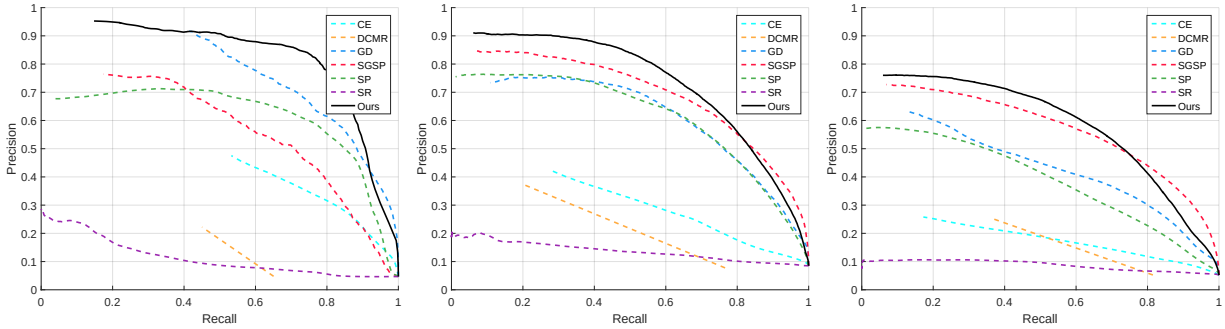

Fig. 4. Comparison with the state-of-the-art methods on PR curves. (a) NoMot. (b) SegTrackV2. (c) UVSD.

algorithm [25] to compute optical flow, which occupies most of the computation cost.

4.4. Discussion

In the experiments, we also find some limitations of the proposed method. Figure 5 shows a failure example. Small size of the salient object, complex scenes, combining with unconstrained camera motion, together lead to the result that our method cannot effectively distinguish the salient object from the background.

5. CONCLUSION

In this paper, we propose a video salient object detection method via cross-frame cellular automata, which can obtain coherent saliency maps over all the video frames. Specifically, we constructed a saliency propagation network to represent the similarities among super-pixels, initialized it with the fusion of appearance saliency map and motion saliency map, and refined the saliency maps iteratively by propagating saliency among super-pixels with cellular automata updating. The experimental results demonstrated that the effectiveness of our method in detecting both static and moving salient objects on different types of videos.

6. REFERENCES

[1] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *TIP*, vol. 24, no. 12, pp. 5706–5722, 2015.

[2] T. Ren, Y. Liu, and G. Wu, "Image retargeting based on global energy optimization," in *ICME*, 2009, pp. 406–409.

Table 1. Comparison with the state-of-the-art methods on F_β^ω and running time per frame.

Method	Code	NoMot		SegTrackV2		UVSD		average	
		F_β^ω	Time (s)	F_β^ω	Time (s)	F_β^ω	Time (s)	F_β^ω	Time (s)
CE	Matlab	0.1652	3.9397	0.1826	3.4351	0.1237	2.9538	0.1572	3.4429
DCMR	C++	0.1363	0.0465	0.1470	0.0413	0.1480	0.0426	0.1438	0.0435
GD	Matlab	0.3243	7.2564	**0.3557**	8.7907	0.2338	11.2767	0.3046	9.1079
SGSP	Matlab	0.1994	6.9027	0.3045	11.2483	0.2021	10.6294	0.2353	9.5935
SP	Matlab	0.2700	7.5342	0.2662	11.6843	0.1771	11.0561	0.2358	10.0915
SR	Matlab	0.0549	0.1811	0.0883	0.1576	0.0662	0.1524	0.0698	0.1637
Ours	Matlab	**0.4832**	7.1523	0.3434	9.4792	**0.2631**	9.1834	**0.3632**	8.6050

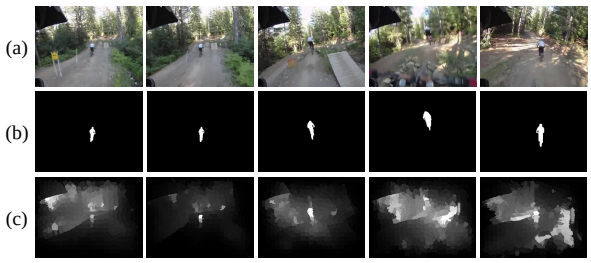

Fig. 5. A failure example of our method. (a) Video frames. (b) Groundtruth. (c) Our results.

[3] H. Zhang, Z.-J. Zha, Y. Yang, S. Yan, Y. Gao, and T.-S. Chua, "Attribute-augmented semantic hierarchy: Towards a unified framework for content-based image retrieval," *TOMM*, vol. 11, no. 1s, pp. 21, 2014.

[4] J. Tang, M. Li, Z. Li, and C. Zhao, "Tag ranking based on salient region graph propagation," *MMSJ*, vol. 21, no. 3, pp. 267–275, 2015.

[5] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *CVPR*, 2014, pp. 2814–2821.

[6] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *CVPR*, 2015, pp. 110–119.

[7] D. Feng, N. Barnes, S. You, and C. McCarthy, "Local background enclosure for rgb-d salient object detection," in *CVPR*, 2016, pp. 2343–2350.

[8] Y. Li, Y. Zhou, J. Yan, Z. Niu, and J. Yang, "Visual saliency based on conditional entropy," in *ACCV*, 2009, pp. 246–257.

[9] C.-R. Huang, Y.-J. Chang, Z.-X. Yang, and Y.-Y. Lin, "Video saliency map detection by dominant camera motion removal," *TCSVT*, vol. 24, no. 8, pp. 1336–1349, 2014.

[10] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *CVPR*, 2015, pp. 3395–3402.

[11] Z. Liu, J. Li, L. Ye, G. Sun, and L. Shen, "Saliency detection for unconstrained videos using superpixel-level graph and spatiotemporal propagation," *TCSVT*, 2016.

[12] J. Von Neumann, "The general and logical theory of automata," *Cereb. Mech. Behav.*, pp. 1–41, 1951.

[13] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *CVPR*, 2009, pp. 1597–1604.

[14] M.-M. Cheng, N.J. Mitra, X. Huang, P.H.S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *TPAMI*, vol. 37, no. 3, pp. 569–582, 2015.

[15] T. Ren, Y. Liu, R. Ju, and G. Wu, "How important is location information in saliency detection of natural images," *MTAP*, vol. 75, no. 5, pp. 2543–2564, 2016.

[16] J. Guo, T. Ren, and J. Bei, "Salient object detection for RGB-D image via saliency evolution," in *ICME*, 2016.

[17] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, 2013, pp. 3166–3173.

[18] J. Zhang, K.A. Ehinger, H. Wei, K. Zhang, and J. Yang, "A novel graph-based optimization framework for salient object detection," *PR*, vol. 64, pp. 39–50, 2017.

[19] S.-H Zhong, Y. Liu, F. Ren, J. Zhang, and T. Ren, "Video saliency detection via dynamic consistent spatio-temporal attention modelling.," in *AAAI*, 2013.

[20] S.-H. Lee, J.-H. Kim, K.P. Choi, J.-Y. Sim, and C.-S. Kim, "Video saliency detection based on spatiotemporal feature learning," in *ICIP*. IEEE, 2014, pp. 1120–1124.

[21] H. Kim, Y. Kim, J.-Y. Sim, and C.-S. Kim, "Spatiotemporal saliency detection for video sequences based on random walk with restart," *TIP*, vol. 24, no. 8, pp. 2552–2564, 2015.

[22] H.J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *J Vision*, vol. 9, no. 12, pp. 15–15, 2009.

[23] Z. Liu, X. Zhang, S. Luo, and O. Le Meur, "Superpixel-based spatiotemporal saliency detection," *TCSVT*, vol. 24, no. 9, pp. 1522–1540, 2014.

[24] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *TPAMI*, vol. 34, no. 11, pp. 2274–2282, 2012.

[25] T. Brox and J. Malik, "Large displacement optical flow: descriptor matching in variational motion estimation," *TPAMI*, vol. 33, no. 3, pp. 500–513, 2011.

[26] F. Li, T. Kim, A. Humayun, D. Tsai, and J.M. Rehg, "Video segmentation by tracking many figure-ground segments," in *ICCV*, 2013, pp. 2192–2199.

[27] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?," in *CVPR*, 2014, pp. 248–255.