HAL
open science

On Spatio-Temporal Saliency Detection in Videos using Multilinear PCA

Désiré Sidibé, Mojdeh Rastgoo, Fabrice Mériaudeau

To cite this version:

Désiré Sidibé, Mojdeh Rastgoo, Fabrice Mériaudeau. On Spatio-Temporal Saliency Detection in Videos using Multilinear PCA. International Conference on Pattern Recognition, IEEE/IAPR, Dec 2016, CANCUN, Mexico. hal-01390675

HAL Id: hal-01390675

https://u-bourgogne.hal.science/hal-01390675v1

Submitted on 2 Nov 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On Spatio-Temporal Saliency Detection in Videos using Multilinear PCA

Désiré Sidibé, Mojdeh Rastgoo, and Fabrice Mériaudeau
LE2I UMR6306, CNRS, Arts et Métiers, Univ. Bourgogne Franche-Comté, F-21000 Dijon, France

***Abstract*—Visual saliency is an attention mechanism which helps to focus on regions of interest instead of processing the whole image or video data. Detecting salient objects in still images has been widely addressed in literature with several formulations and methods. However, visual saliency detection in videos has attracted little attention, although motion information is an important aspect of visual perception. A common approach for obtaining a spatio-temporal saliency map is to combine a static saliency map and a dynamic saliency map. In this paper, we extend a recent saliency detection approach based on principal component analysis (PCA) which have shwon good results when applied to static images. In particular, we explore different strategies to include temporal information into the PCA-based approach. The proposed models have been evaluated on a publicly available dataset which contain several videos of dynamic scenes with complex background, and the results show that processing the spatio-tempral data with multilinear PCA achieves competitive results against state-of-the-art methods.**

I. Introduction

Visual saliency is an attention mechanism which helps focusing on regions of interest rather than processing the whole visual data. It is a useful concept for humans in their daily life and it holds an important place in computer vision applications such as object detection [1], image segmentation [2], robots navigation and localization [3], object tracking [4], image retargeting [5] and image/video compression [6].

Visual attention is generally processed in two approaches which are bottom-up and top-down. Bottom-up attention is a stimulus driven approach derived solely from the conspicuousness of regions in a visual scene. Top-down attention approaches are goal driven and refer to voluntary allocation of attention to certain features, objects or regions in space [7]. Bottom-up approach is more thoroughly investigated than top-down attention approach because the data-driven stimuli are easier to control than cognitive factors such as knowledge and expectations [8].

While saliency detection is a widely studied problem, most of the existing techniques are limited to the analysis of static images, and these approaches cannot be directly extended to the analysis of videos sequences. In the following, we briefly introduce some of the spatio-temporal saliency detection methods described in literature. A recent survey of state-of-the-art methods can be found in [9].

A common approach for obtaining a spatio-temporal saliency map is to combine a static saliency map with a dynamic saliency map. For example, Marat *et al.* [10] proposed a saliency detection method which combines a static saliency map computed using color features, and a dynamic saliency map obtained from optical flow features. Chang et *al.* [1] proposed a spatio-temporal saliency model based on information theory. They used the self-information of local patches as a measure of saliency both in the spatial and temporal domain, and fused the two maps to get the final spatio-temporal saliency map. Kim *et al.* [11] presented a salient region detection method based on a center-surround hypothesis. They used edge and color orientations to compute the spatial saliency, and used temporal gradients to compute the dynamic map which is fused with the spatial one. Zhou *et al.* [12] proposed a dynamic saliency model based on the fact that the displacement of the foreground and the background can be represented by the phase change of the Fourier spectra, and the motion of background objects can be extracted by phase discrepancy in an efficient way. In [13], Seo and Milanfar proposed a space-time saliency detection method which is based on a bottom-up framework and uses local regression kernels as local features. A similar method is developed in [14], where the video patches are modeled using dynamic textures and saliency is computed based on discriminant center-surround. Mancas *et al.* [15] proposed a bottom-up saliency method based on global rarity quantification. The model is based on a multi-scale approach using features extracted from optical flow, and the final saliency map gives the rarity of the statistics of a given video volume at several scales. In [16], the authors proposed a method combining color features for static saliency computation, and texture features for dynamic saliency. The final spatio-temporal saliency map is obtained by fusion of both static and dynamic maps.

Many of these methods are based of the fusion on a static and a dynamic saliency map, often computed separately. However, as shown in [17], the fusion method must be carefully designed to obtain a good final spatio-temporal saliency map. Several fusion methods are evaluated in [17]. One main issue with the fusion approach is the fact that the spatial and temporal information are decorrelated during computation. Therefore, the strong spatio-temporal correlation between the regions of consecutive frames of a sequence is not taken into account. In this paper, we propose a method that explicitly consider a sequence of images as a 3D, space-time volume. Our approach is based on the static saliency detection method of Margolin *et al.* [18] which used PCA to compute the distinctiveness of local patches in a fast fashion. We extend their idea to compute spatio-temporal saliency in dynamic scenes. To this end, we explore different strategies to include temporal information into the PCA-based formulation, and finally adopt a multilinear PCA (MPCA) approach which computes 3D local patches saliency both in space and time. Experimental results on a public dataset show that the MPCA-based approach achieves competitive results when compared

with other proposed methods based on fusion of a static map with a dynamic map.

The rest of the paper is organized as follows. In Section II, we first briefly describe the PCA-based static saliency detection method. Then, in Section III, we describe the extension of this approach for spatio-temporal saliency detection. In particular, we discuss different strategies for considering spatial and temporal information jointly. Section IV, shows performance evaluation of our method and comparison with other approaches. Finally, Section V gives concluding remarks.

II. PCA-BASED SALIENCY DETECTION IN STATIC IMAGES

This section gives a brief description of the PCA-based method of Margolin *et al.* [18], which forms the basis of our approach. The key idea of this algorithm is to compare an image patch not only to its k-nearest neighbours, as done is many previous approaches, but to all other images patches. However, to avoid the computation burden of a comparison with all image patches, PCA is used to represent the patches and compute their distinctness. Another benefit of using PCA is that it reveals the internal statistics of the patches distribution within the image.

More specifically, given an image I, we first extract all local patches $p_{\mathbf{x}}$ centered at pixel $\mathbf{x} = (x, y)$, in the image. The local patch $p_{\mathbf{x}}$ is represented as a vector of size $d = W \times H$, where $W \times H$ is the size of the patch. By putting together all local patches from I, we form a matrix $\mathbf{X}_I = [p_1, \ldots, p_M]$, M being the number of patches. In the next step of the algorithm, we apply PCA to $\mathbf{X}_I$, and represent each local patch by its projection onto the principal axes $\widetilde{p}^k$, $k = 1, \ldots, K$: $p_{\mathbf{x}} = \sum_{k=1}^{K} \alpha_{\mathbf{x}}^k \widetilde{p}^k$. Finally, the pattern distinctness of patch $p_{\mathbf{x}}$ is defined as the L_1 norm of $p_{\mathbf{x}}$ in the PCA coordinates:

$$P(p_{\mathbf{x}}) = \sum_{k=1}^{K} |\alpha_{\mathbf{x}}^k|. \tag{1}$$

The pattern distinctness of Eq. (1) is complemented with a color distinctness computed as the sum of L_2 distances from all other patches in CIE LAB color space. The final saliency map is just the product of the color and pattern distinctness. Note that for further robustness, this procedure is applied in three different resolutions of the input image I: 100%, 50% and 25%, and the results are averaged. In [18], a final refinement of the saliency map is achieved by adding an organization prior defined by a Gaussian placed at the center of the image.

Despite its simplicity, this PCA-based algorithm works remarkably well for static saliency detection, while being very fast. The reader is referred to [18] for further details.

Figure 1 shows some pattern saliency maps produced by the PCA-based approach. As can be seen, the method detects the salient object in the static image of the first row. However, in the case of the image shown in second row, the method fails to highlight the salient objetcs which are the two cyclists. This is because this image is from a video sequence, and the motion information is key to identifying the cyclists as salient objects in the scene. It is therefore important to consider temporal information within this PCA-based framework when working with video sequences. The next section deals with this issue.

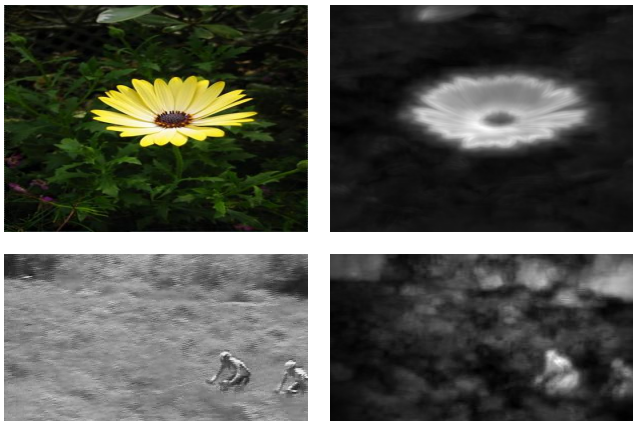


Fig. 1. Examples of saliency maps obtained with the PCA-based approach [18].

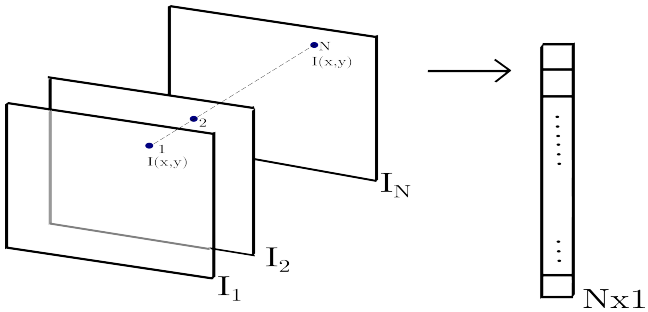


Fig. 2. Extracting pixels intensities along temporal axis.

III. PCA-BASED SPATIO-TEMPORAL SALIENCY DETECTION

In this section, we extend the PCA-based method for spatio-temporal saliency in dynamic scenes. In particular, we explore different strategies to include temporal information into the framework in order to select the best way of computing saliency both in space and time. The different representation are presented in the following sub-sections.

A. PCA-based saliency along temporal axis

One first idea to extend the PCA-based method for processing a video sequence, is to adopt a two-step saliency detection approach, i.e. to compute a static and a dynamic saliency maps for each frame, and then fuse both maps [17]. The static saliency map S_s is computed as described in Section II by considering local patches centered at each pixel location in the image. For dynamic map computation, we consider a set of N consecutive frames, and extract for each pixel location $\mathbf{x}$ the set of its intensity values in the N frames: $p(\mathbf{x}) = \{I_1(x, y), \ldots, I_N(x, y)\}$ as shown in Fig. 2. All vectors $p(\mathbf{x}) \in \mathbb{R}^N$ are put as columns of a data matrix X, and we compute a saliency map S_d following the approach described in Section II. The final spatio-temporal saliency map is obtained by fusion of S_s and S_d as $S = \alpha S_d + (1 - \alpha) S_s$, where $\alpha = \overline{S_d}/(\overline{S_d} + \overline{S_s})$, and $\overline{S_*}$ is the mean value of S_*.

B. PCA-based saliency with 3D local patches

The second option consists in computing a spatio-temporal saliency map by considering local 3D patches, or local 3D sub-

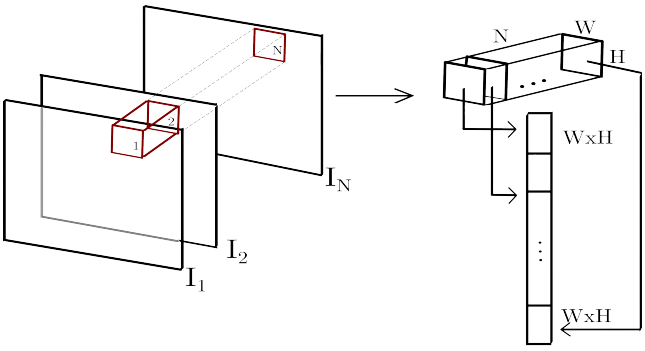


Fig. 3. Extracting local 3D patches and representing them as vectors.

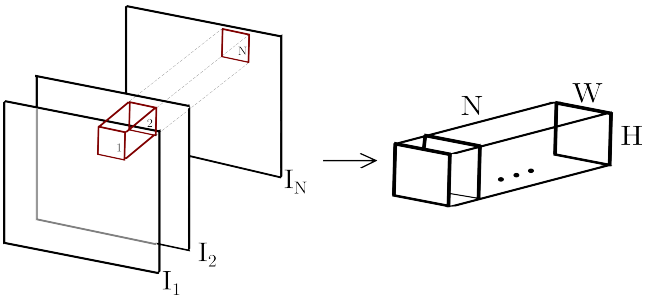


Fig. 4. Extracting 3D local patches and using a 3D tensor representation.

volumes, extracted in a set of N consecutive frames. More specifically, each local patch of size $W \times H$ centered at pixel location $\mathbf{x}$, in each frame, is represented as a column vector in $\mathbb{R}^{WH}$, and we concatenate the vectors from all N frames to form a final representation $\mathbf{x} \in \mathbb{R}^{NWH}$ as shown in Fig. 3. The representation includes both spatial and temporal information, and the saliency map is obtained applying PCA as described in Section II.

C. Using MPCA for spatio-temporal saliency detection

The previous approach, Section III-B, uses both spatial and temporal information at the same time. However, the vector representation of the data removes the strong spatio-temporal correlation that exists between consecutive frames in a sequence. Therefore, as a third option, we propose to use a multilinear PCA method (MPCA) that extends classical PCA to multidimensional data [19]. MPCA is a multilinear subspace learning method that represents multidimensional data as tensors rather than vectors [20]. MPCA, thus, preserves the structure of the data, in our case the 3D space-time structure of the video sequence, and extracts features directly from this natural tensor representation.

For each pixel location $\mathbf{x} = (x, y)$, we extract a 3D spatio-temporal neighbourhood and represent it as a third-order tensor. Following the notations in [19], we represent each local 3D sub-volume as a tensor $\mathcal{X} \in \mathbb{R}^{W \times H \times N}$, where $W \times H$ is the spatial size of the patch and N is the temporal dimension as shown in Fig. 4.

The main idea of MPCA is to project the tensor $\mathcal{X}$ onto a lower dimensional tensor $\mathcal{Y} \in \mathbb{R}^{W' \times H' \times N'}$ as:

$$\mathcal{Y} = \mathcal{X} \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \times_3 \mathbf{U}^{(3)^T}, \tag{2}$$

where $\mathbf{U}^{(1)} \in \mathbb{R}^{W \times W'}$ is a projection matrix along the first mode of the tensor, and similarly for $\mathbf{U}^{(2)}$ and $\mathbf{U}^{(3)}$. $\times_n$ is the n-mode projection.

Therefore, MPCA uses three projections, one for each of the three modes. Also, since $W' < W$, $H' < H$ and $N' < N$, the tensor dimensions are reduced from $W \times H \times N$ to $W' \times H' \times N'$. The projection matrices are obtained iteratively using an alternating projection method [19], where each iteration involves 3 modewise eigen-decompositions.

Finally, the saliency values of each pixel $\mathbf{x}$ is computed, in a similar manner as in [18], as the L_1 norm of the coordinates of $\mathcal{Y}$ along all three modes:

$$P(\mathbf{x}) = \sum_x \sum_y \sum_t |\mathcal{Y}(x, y, t)|. \tag{3}$$

D. Implementation details and parameters

There are two main parameters to set in the basic PCA-based method [18]. One is the dimension K of the linear subspace, which is given by the number of principal components. In all experiments, we set K such as at least 95% of the data variance is preserved. Specifically, we set K such that $\sum_{i=1}^{K} \lambda_i / \sum_{i=1}^{d} \lambda_i \geq 0.95$, the λ_i's being the eigenvalues of the covariance matrix sorted in descending order. The second parameter is the size $W \times H$ of the local patches. In Section IV, we try different values and select the best one.

When we extend the PCA-based method to consider temporal information, an additional parameter is the size of the temporal window, i.e. the number N of consecutive frames to consider. This parameter is empirically set by trying several values in Section IV.

For the MPCA approach, the main parameter is a threshold value Q for determining the tensor subspace dimensions $\{W' \times H' \times N'\}$. Specifically, the first P_n eigenvectors are kept in the n-mode so that the same amount of variances is kept in each mode: $Q^{(1)} = Q^{(2)} = Q^{(3)} = Q$, where $Q^{(n)}$ is the ratio of variance kept in the n-mode. As in the PCA-based method, we set $Q = 0.95$.

IV. Experiments and Results

This section describes the experiments conducted to evaluate the performances of the proposed spatio-temporal saliency detection method. In particular, we evaluate the performance of the method in locating salient foreground objects in complex dynamic scenes using a publicly available dataset. We also compare the proposed multilinear PCA based method with other methods proposed in literature.

A. Dataset and evaluation metric

To evaluate the different spatio-temporal saliency models, we use a publicly available complex video scenes datasets: the SVCL dataset [14]. The dataset contains a variety of natural videos which are composed of dynamic entities such as waving trees, crowd, moving water, waves, and snow. The detection of salient foreground objects against these dynamic backgrounds is very challenging. The SVCL dataset also includes manually segmented objects for each frame which served as ground truth

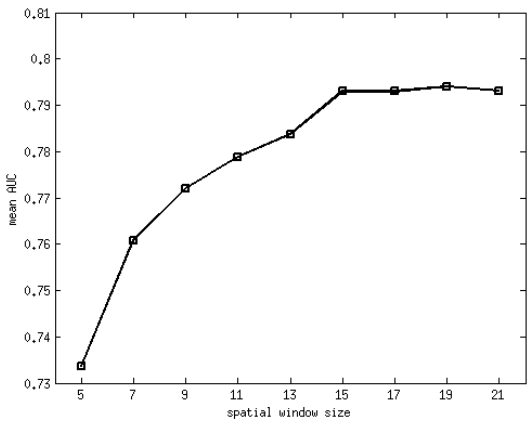


Fig. 5. Variation of the mean AUC value with varying spatial window size.

data, thus allowing us to perform a quantitative analysis of the method's performance.

The performances are evaluated using Receiver Operating Charecteristics (ROC) curves and evaluating the Area Under Curve (AUC). A saliency map obtained from a method is first normalized to the range $[0, 1]$ and is binarized using varying thresholds $t \in [0, 1]$. Then, we compute the True Positive Rate (TPR) and False Positive Rate (FPR) using these binarized maps, and generate a ROC curve. The AUC value measures the similarity of the detected saliency map with the ground truth.

B. Parameters setting

As mentioned in Section III-D, there are two parameters that need to be set in the proposed saliency detection method. The first one is the spatial size of the local patches extracted in each frame of a sequence; $W \times H$. In the experiments, we use square patches of size $W \times W$ and try the following values: $\{5, 7, 9, 11, 13, 15, 17, 19, 21\}$.

We use all sequences in the dataset, and for each sequence we apply the PCA-based method of Section II to each frame individually. Then we compute the mean AUC value over the entire dataset. Figure 5 shows the variation of the mean AUC value, computed over all sequences in the dataset, for varying spatial window sizes. As we can see, the AUC value increases with the window size until it stabilizes around $W = 15$. We use this value in the rest of our experiments for the spatial window size.

The second important parameter is the size of the temporal window, i.e. the number of consecutive frames, N, to use for including temporal information. We set the spatial window size to $W = 15$ according the previous results, and try different values for N: $\{5, 7, 9, 11, 13\}$, using the PCA-based saliency method with 3D local patches as described in Section III-B. The results shown in Table I suggest that $N = 7$ is a good tradeoff between accuracy and computation time. Larger values of N do not lead to a significant increase in the mean AUC values. Therefore, we adopt the value $N = 7$ for the rest of our experiments.

Temporal window size	5	7	9	11	13
mean AUC	0.81	0.84	0.84	0.83	0.83

TABLE I. VARIATION ON THE MEAN AUC VALUE WITH VARYING TEMPORAL WINDOW SIZE.

Method	PCA [18]	Fusion	Vectorized	MPCA
mean AUC	0.7930	0.7440	0.8436	0.9041

TABLE II. COMPARISON OF THE DIFFERENT STRATEGIES FOR INCLUDING TEMPORAL INFORMATION.

C. Methods comparison

We now compare the different proposed strategies for extending the PCA-based approach of Margolin *et al.* [18] in the temporal domain. The three strategies are described in Section III-A, III-B and III-C respectively, and we refer to them as follows:

- **Fusion**: computing a static and a dynamic saliency maps for each frame, and fusing both maps to get the final spatio-temporal saliency map.
- **Vectorized**: computing a spatio-temporal saliency map by extracting local 3D patches and representing them as vectors (Fig. 3).
- **MPCA**: computing a spatio-temporal saliency map using the full 3D tensor representation of the data and applying a multilinear PCA method.

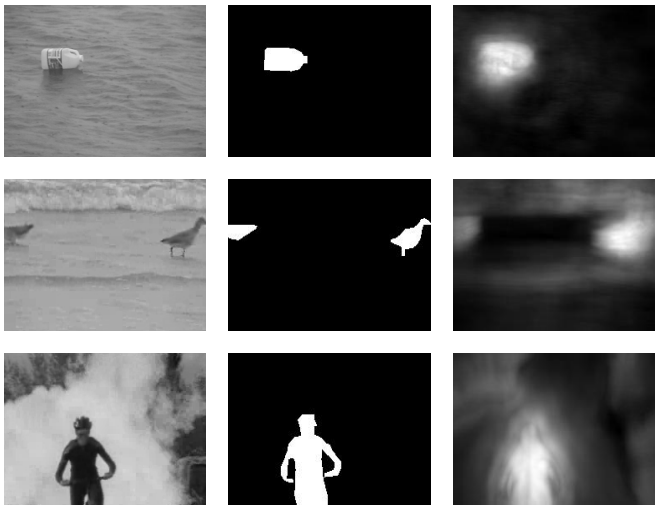
Table II summarizes the results obtained with the different strategies, along with the results obtained with the original PCA method [18] applied to each frame individually. We can clearly observe that extracting 3D local patches leads to a significantly improvement of the performance. In particular the *Vectorized* method achieves an average AUC value of 0.8440 while the *MPCA* method achieves an average AUC of 0.9041. On the other hand, the *Fusion* approach achieves low performance with an average AUC of 0.7440, which is lower than applying the PCA method to each frame individually. This can be explained by the fact that extracting pixels intensities along the temporal axis, as shown in Fig 2, does not provide enough context information for saliency detection. On the contrary, extracting 3D sub-volumes provides both spatial and temporal context, which leads to better performance. Moreover, using a tensor representation rather than vector representation significantly improves the results. This is because, the tensor representation preserves the structure of the data, the 3D space-time structure of the video sequence, whereas the vectorized approach does not exploit this spatio-temporal correlation.

Finally, we compare our MPCA based method with four state-of-the art methods: a method that fuses optical flow and color features (OF) [17], the self-resemblance method (SF) [21], the phase discrepancy based saliency detection method (PD) [12], and a method that uses LBP features extended to temporal domain (LBP) [16].

The results obtained with all sequences by the different saliency detection methods are shown in Table III. We can observe that the MPCA based method achieves competitive results with an average AUC value of 0.9041 for all twelve sequences. It achieves similar results than the optical flow based

Sequence	MPCA	LBP [16]	OF [17]	SR [21]	PD [12]
Birds	0.9757	0.9586	0.9664	0.9379	0.8221
Boats	0.9059	0.9794	0.9827	0.9227	0.9765
Bottle	0.9936	0.9953	0.8787	0.9961	0.8285
Cyclists	0.9790	0.9317	0.9602	0.8682	0.9551
Chopper	0.9843	0.9717	0.9850	0.7447	0.6470
Freeway	0.8042	0.7775	0.5456	0.7760	0.7318
Peds	0.9405	0.9552	0.9512	0.8603	0.8548
Ocean	0.9037	0.9271	0.7810	0.8016	0.8235
Surfers	0.8448	0.9674	0.9545	0.9455	0.9352
Skiing	0.7857	0.8389	0.9796	0.8872	0.9367
Jump	0.9368	0.8957	0.9481	0.8321	0.6616
Traffic	0.7946	0.7693	0.9615	0.5491	0.8720
Mean AUC	**0.9041**	**0.9140**	**0.9079**	**0.8434**	**0.8371**

TABLE III. EVALUATION OF DIFFERENT SPATIO-TEMPORAL SALIENCY DETECTION METHODS.



(a) Input frame (b) groung truth map (c) saliency map

Fig. 6. Examples of saliency maps obtained with the MPCA based approach.

method (OF), and shows better performance than the self-resemblance method (SF) and the phased discrepancy method (PD). The LBP based method achieves slightly better results with an AUC of 0.9140. However, this method combines a static map computed using color features with a dynamic map computed with LBP features, while our MPCA based method uses only the pixels intensity values.

Some examples of the saliency maps obtained by the multilinear PCA (MPCA) based approach are shown in Fig. 6.

V. CONCLUSION

This paper describes approaches for spatio-temporal saliency detection in dynamic scenes using principal component analysis (PCA). Our work is based on a previous PCA-based approach for saliency detection in static images which is extended to deal with video sequences. In particular, we have explored different strategies to include temporal information into the PCA-based approach, and experimental results with a public dataset show that a multilinear PCA (MPCA) approach which computes 3D local patches saliency both in space and time provides the best performance. The MPCA approach uses a tensor representation that preserves the structure of the data rather than vector representation that removes spatio-temporal correlation in the data. Comparison with other state-of-the-art methods shows that the proposed MPCA approach achieves competitive results.

A possible extension of this work could be the integration of depth cues into the spatio-temporal saliency model. The current availability of RGB-D sensors makes this possible and we will investigate in this direction in the future.

REFERENCES

[1] L. Chang, P. C. Yuen, and G. Qiu, "Object motion detection using information theoretic spatio-temporal saliency," *Pattern Recogn. 2009*, vol. 42, no. 11, pp. 2897–2906.

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

[3] C. Siagian and L. Itti, "Biologically inspired mobile robot vision localization," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 861–873, July 2009.

[4] D. Sidibé, D. Fofi, and F. Mériaudeau, "Using visual saliency for object tracking with particle filters," in *EUSIPCO*, 2010.

[5] T. Lu, Z. Yuan, Y. Huang, D. Wu, and H. Yu, "Video retargeting with nonlinear spatial-temporal saliency fusion," in *ICIP,2010*.

[6] C. L. Guo and L. M. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Transactions on Image Processing,*, vol. 19, no. 1, pp. 185 –198, 2010.

[7] Y. Pinto, A. R. van der Leij, I. G. Sligte, V. A. F. Lamme, and H. S. Scholte, "Bottom-up and top-down attention are independent," *Journal of Vision*, vol. 23, no. 3, p. 16, 2013.

[8] S. Frintrop, *Computational Visual Attention*. Springer, 2011.

[9] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Transactions on PAMI,*, vol. 35, no. 1, pp. 185–207, 2013.

[10] S. Marat, T. H. Phuoc, L. Granjon, N. Guyader, D. Pellerin, and A. Guérin-Dugué, "Modelling spatio-temporal saliency to predict gaze direction for short videos," *IJCV,2009*, vol. 82, no. 3, pp. 231–243.

[11] W. Kim, C. Jung, and C. Kim, "Spatiotemporal saliency detection and its applications in static and dynamic scenes," *IEEE Trans. Circuits Syst. Video Techn,2011*, vol. 21, no. 4, pp. 446–456.

[12] B. Zhou, X. Hou, and L. Zhang, "A phase discrepancy analysis of object motion," in *Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III*, ser. ACCV'10. Springer-Verlag, 2011, pp. 225–238.

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

[14] V. Mahadevan and N. Vasconcelos, "Spatiotemporal saliency in dynamic scenes," *IEEE Transactions on PAMI,*, vol. 32, no. 1, pp. 171 –177, 2010.

[15] M. Mancas, N. Riche, J. Leroy, and B. Gosselin, "Abnormal motion selection in crowds using bottom-up saliency," in *ICIP*, 2011, pp. 229–232.

[16] S. Muddamsetty, D. Sidibé, A. Trémeau, and F. Mériaudeau, "Spatio-temporal saliency detection in dynamic scenes using local binary patterns," in *ICPR*, 2014, pp. 2353–2358.

[17] ——, "A performance evaluation of fusion techniques for spatio-temporal saliency detection in dynamic scenes," in *ICIP*, 2013.

[18] R. Margolin, T. Ayellet, and L. Zelnik-Manor, "What makes a patch salient," in *IEEE CVPR*, 2013, pp. 1139–1146.

[19] H. Lu, K. Plataniotis, and A. Venetsanopoulos, "MPCA: Multilinear principal component analysis of tensor objects," *IEEE Trans. on Neural Networks*, vol. 19, no. 1, pp. 18–39, 2008.

[20] ——, "A survey of multilinear subspace learning for tensor data," *Pattern Recognition*, vol. 44, no. 7, pp. 1540–1551, 2011.

[21] H. J. Seo and P. Milanfar., "Nonparametric bottom-up saliency detection by self-resemblance," in *Computer Vision and Pattern Recognition Workshops, 2009*, 2009, pp. 45 –52.