J. Vis. Commun. Image R. 35 (2016) 169–183

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Compressed domain video saliency detection using global and local spatiotemporal features ☆

Se-Ho Lee [a], Je-Won Kang [b], Chang-Su Kim [a,*]

[a] School of Electrical Engineering, Korea University, Seoul, Republic of Korea
[b] Department of Electronics Engineering, Ewha Womans University, Seoul 120-750, Republic of Korea

ARTICLE INFO

Article history:
Received 16 September 2015
Accepted 15 December 2015
Available online 24 December 2015

Keywords:
Video saliency detection
Spatiotemporal feature
Compressed domain
Visual attention
Partial decoding
Image understanding
Image analysis
Motion analysis

ABSTRACT

A compressed domain video saliency detection algorithm, which employs global and local spatiotemporal (GLST) features, is proposed in this work. We first conduct partial decoding of a compressed video bitstream to obtain motion vectors and DCT coefficients, from which GLST features are extracted. More specifically, we extract the spatial features of rarity, compactness, and center prior from DC coefficients by investigating the global color distribution in a frame. We also extract the spatial feature of texture contrast from AC coefficients to identify regions, whose local textures are distinct from those of neighboring regions. Moreover, we use the temporal features of motion intensity and motion contrast to detect visually important motions. Then, we generate spatial and temporal saliency maps, respectively, by linearly combining the spatial features and the temporal features. Finally, we fuse the two saliency maps into a spatiotemporal saliency map adaptively by comparing the robustness of the spatial features with that of the temporal features. Experimental results demonstrate that the proposed algorithm provides excellent saliency detection performance, while requiring low complexity and thus performing the detection in real-time.

1. Introduction

Saliency detection is a process to identify regions of interest (ROIs) in images and video sequences automatically. It can be used for focusing on ROIs, thus saving limited system resources in image processing and computer vision applications. For saliency detection, perceptual features are extracted to measure the significance of each region and then are combined to create a saliency map, in which high pixel intensities indicate visually important regions. Efficient saliency detection techniques can facilitate various tasks in real-time image and video applications, including image resizing [1], image segmentation [2], and object recognition [3]. The accuracy and the computational complexity are two important factors in designing a practical saliency detection technique.

Saliency detection techniques for images have been extensively studied. Itti et al. [4] and Seo and Milanfar [5], respectively, estimate visual saliency using feature differences between a center patch and its surrounding patches. Li et al. [6,7] detect saliency by attempting to represent a block as a combination of its neighboring blocks. Harel et al. [8] and Kim et al. [9], respectively, compute visual saliency maps based on random walk models. Also, saliency detection algorithms, considering spectral properties of an image, have been proposed in [10–12]. Recently, region-based saliency detection algorithms have been developed in [13–16], which first divide an input image into multiple meaningful regions and then assign saliency levels to those regions.

These algorithms [4–16] are mainly for still images, yet some algorithms can be applied to video saliency detection. Seo and Milanfar [5] and Li et al. [6,7] consider spatiotemporal surroundings to extract features in video sequences. Harel et al. [8] adopt frame differences as temporal features. However, the performance of video saliency detection can be improved by employing motion information, since human visual system readily perceives object motions in video frames. Hence, Guo and Zhang [17] use motion features as well as color and intensity features to describe each region by using the quaternion representation. Chen et al. [18] derive a motion attention map to indicate irregular motions. Lee et al. [19] exploit motion intensity and motion contrast, in addition to spatial features, in their learning-based video saliency detection algorithm. Kim et al. [20] detect spatiotemporal saliency maps, by incorporating a spatial transition matrix and a temporal restarting distribution, based on the theory of random walk with restart (RWR).

☆ This paper has been recommended for acceptance by M.T. Sun.

* Corresponding author.
E-mail addresses: seholee@mcl.korea.ac.kr (S.-H. Lee), jewonk@ewha.ac.kr (J.-W. Kang), changsukim@korea.ac.kr (C.-S. Kim).

http://dx.doi.org/10.1016/j.jvcir.2015.12.011
1047-3203/

Most conventional algorithms for saliency detection extract features in the uncompressed domain. However, image and video contents are, in almost every case, transmitted in compressed formats for reducing the data sizes. The conventional algorithms cannot be directly applied to the compressed data. Their feature extraction processes are possible only after the full decoding, which incurs not only extra computational complexity but also latency to the saliency detection. In this context, several feature extraction schemes in the compressed domain have been developed for a variety of applications, such as video transcoding [21], video retargeting [22], and object recognition [23]. Furthermore, recently, image and video saliency detection algorithms in the compressed domain [24–27] have been developed.

In this paper, we propose a compressed domain video saliency detection algorithm using global and local spatiotemporal (GLST) features. We extend our preliminary work [19], which generates saliency maps by extracting a set of features in the uncompressed domain. First, we decode a compressed bitstream partially to reconstruct motion vectors and DCT coefficients. Then, we extract GLST features from the partially decoded data to quantify visual saliency of each block. More specifically, the GLST features are composed of rarity, compactness, center prior, texture contrast, motion intensity, and motion contrast. The former four are spatial features derived from DCT coefficients, while the latter two are temporal features obtained from motion vectors. Also, rarity, compactness, center prior are global features to describe the color distribution in an entire frame, whereas texture contrast, motion intensity, and motion contrast are local features to describe local properties of each block in itself or in comparison with surrounding blocks. Finally, we combine these diverse features into an overall saliency map adaptively, based on the reliability of each feature. Experimental results show that the proposed algorithm detects salient regions effectively and provides comparable or better quantitative performance than the conventional algorithms [5–16,27], even though the proposed algorithm can be executed efficiently in real-time.

The rest of this paper is organized as follows. Section 2 surveys saliency detection techniques and compressed domain image and video processing schemes. Section 3 describes the proposed saliency detection algorithm. Section 4 presents experimental results. Finally, Section 5 concludes this paper.

2. Related work

2.1. Saliency detection

This section reviews conventional algorithms on image and video saliency detection, most of which will be compared with the proposed algorithm experimentally in Section 4.

Itti et al. [4] propose an early algorithm for image saliency detection based on the center-surround hypothesis that a region, whose features are distinct from those of surrounding regions, is salient. They use color, intensity, and orientation features and generate a saliency map by combining the center-surround differences of those features. Seo and Milanfar [5] also assume the center-surround hypothesis, and adopt the self-resemblance measurement to compute the differences between patches. Li et al. [6] attempt to express a center patch as a sparse linear combination of surrounding patches. They regard the center patch as salient, if the sparse representation is unsuccessful and thus the linear weights require a long coding length. Li et al. [7] also detect saliency by measuring the minimum conditional entropy, which represents the uncertainty of a patch given its surrounding patches.

Harel et al. [8] propose a saliency detection algorithm using a random walk model in graph theory. They regard an image block as a node in the graph, and assume that human eyes tend to move from one node to another frequently when the two nodes are dissimilar from each other. Hence, they set the saliency of each block to be proportional to the visiting frequency of the random walker to the corresponding node. However, their performance is affected significantly by the adopted block size. To alleviate this scale sensitivity, Kim et al. [9] apply the RWR theory in order to detect saliency in multiple scales. They use a coarse-to-fine approach, based on the observation that HVS first catches object appearance roughly and then recognizes fine details.

Spectral domain algorithms have been proposed in [10–12]. Achanta et al. [10] propose the frequency-tuned approach, which computes the difference between the Gaussian blurred version of an image and the average pixel value. Hou and Zhang [11] propose the spectral residual approach, which exploits spectral singularities to detect saliency. Schauerte and Stiefelhagen [12] also exploit the residual spectrum, but they analyze the spectrum using the quaternion Fourier transform.

Most of the aforementioned algorithms divide an image into regular patches or blocks, regardless of regionally different characteristics in the feature extraction. To overcome this drawback and improve the saliency detection performance, region-based algorithms have been proposed in [13–16]. These algorithms group neighboring pixels, which are similar to one another, into a region or a superpixel. Cheng et al. [14] compute the feature distances of a region to the other regions to determine the contrast. Perazzi et al. [16] extend the Cheng et al.'s algorithm by considering the spatial variance of a color distribution. Yang et al. [13] assume that boundary regions in an image tend to belong to the background. They then rank each region based on its similarity to background cues and discriminate salient foreground regions from the background. However, because of this boundary prior, their algorithm may fail to detect salient objects near the boundary. To alleviate this problem, Zhu et al. [15] adopt the boundary connectivity to detect background regions more robustly.

These algorithms [4–16] are primarily for image saliency detection, although they can be used for video saliency detection, *e.g.* by computing the saliency map of each frame in a video sequence independently. To exploit temporal relationship more efficiently in video saliency detection, Guo and Zhang [17] exploit motion features, in addition to color and intensity features, to describe an input video in the quaternion representation. Chen et al. [18] detect salient points using the spatiotemporal Harris detector, and employ them as seeds to extend salient regions in a motion attention map. Kim et al. [20] adopt RWR to detect spatially and temporally salient regions.

2.2. Compressed domain image and video processing

In image and video processing, DCT coefficients and motion vectors are widely used as spatial and temporal features, respectively. Since they can be extracted from a compressed video bitstream without the full decoding, various compressed domain algorithms for image and vision applications have been proposed [21–23,26,27].

Lin and Lee [21] propose a transcoding system to reduce the bitrates of a video bitstream. They truncate DCT coefficients in the compressed domain to bypass the complete decoding and re-encoding processes. Zhang et al. [22] conduct video retargeting in the compressed domain. They estimate visual importance of each block using motion and texture features, and determine the content-aware mesh deformation based on the importance map. Sukmarg and Rao [23] perform object segmentation and detection in the compressed domain. They partition an image into small regions using DCT coefficients, and merge those regions based on