Int J Comput Vis
DOI 10.1007/s11263-015-0853-6

Visual Saliency Detection Using Group Lasso Regularization in Videos of Natural Scenes

Nasim Souly[1] · **Mubarak Shah**[1]

Received: 6 August 2014 / Accepted: 31 July 2015

Abstract Visual saliency is the ability of a vision system to promptly select the most relevant data in the scene and reduce the amount of visual data that needs to be processed. Thus, its applications for complex tasks such as object detection, object recognition and video compression have attained interest in computer vision studies. In this paper, we introduce a novel unsupervised method for detecting visual saliency in videos of natural scenes. For this, we divide a video into non-overlapping cuboids and create a matrix whose columns correspond to intensity values of these cuboids. Simultaneously, we segment the video using a hierarchical segmentation method and obtain super-voxels. A dictionary learned from the feature data matrix of the video is subsequently used to represent the video as coefficients of atoms. Then, these coefficients are decomposed into salient and non-salient parts. We propose to use group lasso regularization to find the sparse representation of a video, which benefits from grouping information provided by super-voxels and extracted features from the cuboids. We find saliency regions by decomposing the feature matrix of a video into low-rank and sparse matrices by using robust principal component analysis matrix recovery method. The applicability of our method is tested on four video data sets of natural scenes. Our experiments provide promising results in terms of predicting eye movement using standard evaluation methods. In addition, we show our video saliency can be used to improve the performance of human action recognition on a standard dataset.

Keywords Visual saliency · Sparse coding · Super-voxels · Group lasso

Communicated by Jakob Verbeek.

✉ Nasim Souly
nsouly@eecs.ucf.edu

Mubarak Shah
shah@crcv.ucf.edu

[1] The Center for Research in Computer Vision, 4000 Central Florida Blvd., Orlando, Florida 32816, USA

1 Introduction

Images of natural scenes contain large amounts of data which need to be processed. However, significant portions of scenes are redundant and visual systems have limitations in fully processing a complex scene. Therefore, a method to select informative data is required.

The human vision system has a built in cognitive mechanism that differentiates the relevant from the irrelevant parts in visual stimulus received from complex scenes. The gaze is naturally equipped to be directed to important aspects of a scene. The part of an image or a video that captures human attention is said to be salient; that is where people focus when looking at any scene. For instance, in Fig. 1, we are likely to be attracted to the pigeons, and not the ground, the person riding the bike and not the road or the boats and the ship and not the sea.

Despite the fact that extensive psychological and neurophysiological research has studied the human visual system (e.g. Ungerleider and Leslie 2000; Rensink et al. 1997), it is not completely understood how one's gaze so easily zeros in on only the relevant stimulus. Saliency detection is a challenging problem that has yet to be fully solved.

Visual saliency has been the focus of many studies in computer vision in recent years, because of its broad potential applications. Saliency detection can be used for object detection (Navalpakkam and Itti 2006), automatic image cropping, predicting human gaze (Marat et al. 2009), image

Published online: 15 August 2015

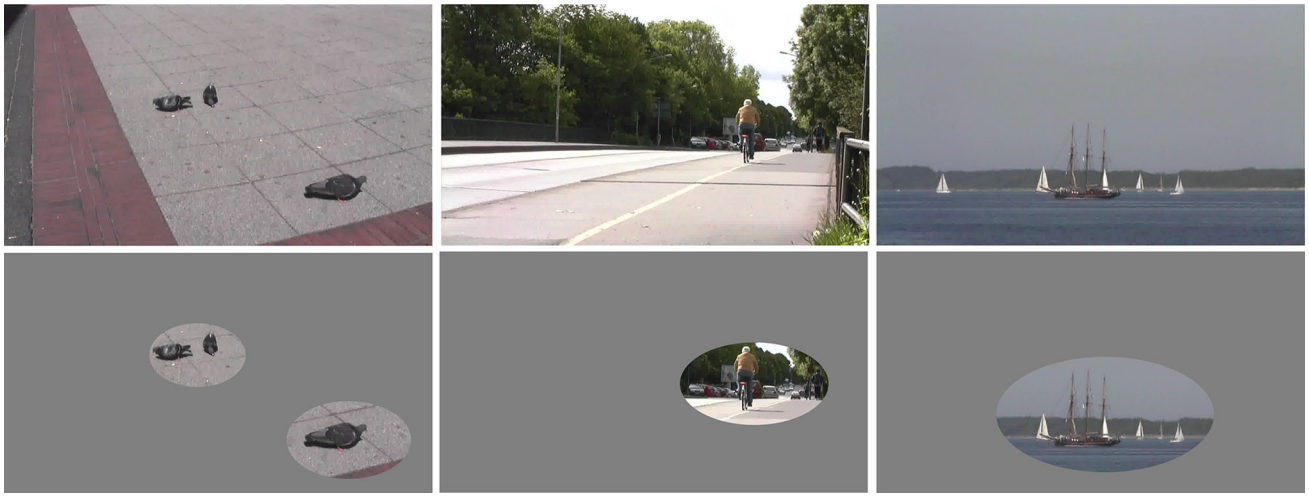


Fig. 1 High-saliency areas of a natural scene are the small portions that hold the most important information and can be identified easily by the human vision system. In this figure, the top row shows frames from videos of natural scenes and the bottom row shows specifics of the scenes which are the most relevant to understand the scenes in reference to saliency

and video compression, video summarization (Ma et al. 2005, 2002), and more. For example, in object detection, determining saliency can help to reduce the size of the area to be processed. And during image resizing, saliency can aid in preserving structure. By identifying the important parts of a video, the human gaze can be predicted as well. Another beneficial application of saliency detection is in video summarization; for instance, using saliency maps different changes in frames of a video can be highlighted and key frames can be selected. If maps look similar, and their difference is under a threshold, the redundant frames could be eliminated (Marat et al. 2007). Also, in aerial video summarization, several key frames from each scene can be selected based on the visual saliency index of each frame computed from their visual saliency map (Wang et al. 2011b).

In determining saliency, the computational vision model seeks to find the part of an image or video which stands out from the rest of the scene. Changes in the scene, such as color variation, spatial contrast, or sudden movement are important factors since they redirect the observer's gaze. A variety of methods exist to predict exactly what captures the eye (Marat et al. 2009).

In general, saliency detection methods are divided into two types: top-down (Borji et al. 2011; Triesch et al. 2003), which is a task-driven method involving a high-level cognitive process that models attention by task; and bottom-up (Seo and Milanfar 2009a; Kienzle et al. 2007b), which is a stimuli-driven and extracts eye-catching regions from an image or video without any prior knowledge. Top-down models indicate a biased selection process, considering the expectation, "will" of a target. They are the subject of interactive studies such as driving and game playing (Borji and Itti 2013). On the other hand, bottom-up approaches try to find novel parts of a scene using low-level features without prior knowledge about the scene. The latter have mostly been investigated using eye movement prediction in free-viewing of videos. Bottom-up methods are usually faster because they use low-level features which are characterized by stimulus driven factors (Seo and Milanfar 2009a).

While there is much attention being given to saliency detection in images (e.g. Bruce and Tsotsos 2009; Gao and Vasconcelos 2009; Borji and Itti 2013), relatively few methods have been proposed for videos. We live in a dynamic world where videos capture more realistic and detailed models of the environments to which one's vision system is exposed. In this paper, we use spatio-temporal visual features to develop a method for detecting saliency in videos.

The goal of this paper is to find salient objects and actions with no presumption about the target in free-viewing videos. Therefore, we propose a bottom-up approach to find visual saliency to predict gaze based on visual features. In addition, this approach is independent of a training process in which similar videos would be required first.

The proposed model focuses on the concept of a "saliency map", which indicates the saliency of a specific location over the entire scene. The task of saliency detection consists of three major steps: initially, an extraction of the features that could be used to find salient areas effectively, followed by determining salient regions based on those features, and lastly assigning a saliency value or score to each part.

Since salient regions in videos are only a small part of a video, we use a sparse-signal analysis technique to represent the information as redundant plus salient parts. In this way, non-salient areas, such as background, are expressed by a

low-dimensional subspace and salient parts are specified by sparse parts (Rudoy et al. 2013). In perception, saliency is related to homogeneity, in a manner such that when homogeneity increases saliency decreases (Poirier et al. 2008).

An overview of our proposed model is shown in Fig. 2. We use robust principal component analysis low-rank matrix recovery (Wright et al. 2009) method in order to decompose the obtained feature matrix. The essential task here is to come up with a feature (descriptor) matrix that determines a space in which the non-salient regions reside in a low-dimensional subspace. For this reason, the main part of our work involves providing an appropriate feature matrix as input for the decomposition step. Since sparse coding representation, inspired by neuroscience studies, has been successful in modeling natural scenes, and because psychological studies, e.g. (Poirier et al. 2008), show heterogeneous surfaces are more salient than homogeneous ones, we propose to use sparse representation in our model.

Sparse coding suppresses slight changes in a scene so that the strong variations stand out. In this new representation redundant data lies in a low rank space. In our approach, a video is represented as a collection of spatiotemporal cuboids expressed in terms of an over-complete dictionary. In doing so, a dictionary is created whose atoms are learned based on the feature data matrix of a video. By using the L_1-minimization approach, a coefficients matrix is obtained, and is then divided into salient and non-salient parts. However, the coefficients could be noisy possibly due to salient regions not being sparse, inasmuch as a large area divided into small patches. In order to address this problem, we propose to use super-voxels and sparsity among super-voxels rather than cuboids. Consequently, in addition to decomposing a video into cuboids, we group these cuboids into super-voxels.

Next, a group lasso regularization method—which uses $L_{1,2}$-norm minimization to encourage the columns within a group to be zero– is used to transform the video feature matrix into a sparse coding space that would be used by Robust PCA. The final step of our method is to create a saliency map via the sparse matrix that was found by the decomposition step. Each vector in the acquired sparse matrix corresponds to a cuboid in the original video. By computing L_1-norm of these vectors, the saliency values of super-voxels are achieved. An overview of our proposed model is shown in Fig. 2.

The rest of the paper is organized as follows: The next section deals with related work, Sect. 3 describes our approach to find appropriate feature mapping, dictionary learning and decomposing a video into salient and non-salient parts. Section 4 consists of implementation details of the method and results for different data sets, and comparison of our method with baseline models. Finally, Sect. 5 summarizes the approach and provides conclusions.

2 Related Work

Most saliency detection models in the bottom-up category are biologically inspired and follow the Feature Integration Theory of Treisman and Gelade (1980). This suggests that when perceiving a stimulus, features are registered early, automatically, and in parallel, while objects are identified separately and at a later stage during processing. These bottom-up methods decompose visual input into separate low-level feature

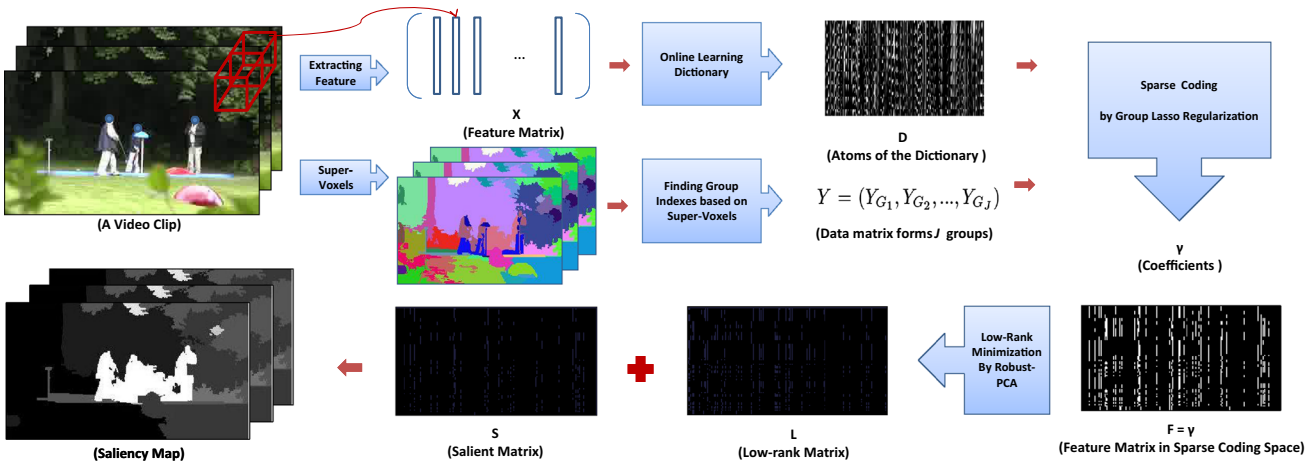


Fig. 2 An overview of the proposed approach: We begin with extracting the feature matrix, X, of a video, and segmenting the video into super-voxels. A dictionary, D, is learned online. The video is then represented by F in terms of coefficients γ obtained from group lasso regularization over the dictionary. Salient parts, represented by Sparse matrix (S), and non-salient parts (L) are recovered via low-rank minimization technique (Robust PCA). Finally, a saliency map is generated based on the L_1 norm of columns of the matrix S belonging to super-voxels

maps, e.g. orientation, contrast, and color. For every individual feature, a different map is computed and normalized. Then, a saliency map is formed by the weighted combination of them. Peaks in the map reflect the attention (saliency) (Itti et al. 1998). In addition, some new mathematical and statistical tools (e.g Itti et al. 1998; Bruce and Tsotsos 2009) have been used more recently in order to obtain precise results which have been mostly evaluated on eye movement data provided by gaze location of viewers.

On the other hand, top-down approaches are mostly investigated by cueing experiments, in which a "cue" brings one's notice to the target. The cue could be what the target is or where it will be (Frintrop et al. 2010). Also worth noting is that while bottom-up approaches are mainly driven by the visual characteristics of a scene, top-down models mostly define attention models according to the task of interest. Gao (2004, 2009), and Gao et al. (2009) proposed top-down approaches which used decision-theoretic models. They introduced the concept of discriminant saliency, which is based on the definition of the target and null hypotheses. They defined top-down saliency as a classification task with which locations where a target could be distinguished from a non-target, with minimum error , is categorized as salient.

Saliency detection approaches can also be categorized on the basis of the techniques that they use to obtain saliency maps. For instance, different computational principles such as Information theoretic models and Bayesian models have been employed in bottom-up saliency methods to define the concept of saliency. Some approaches use information theory to determine "distinctiveness". Bruce and Tsotsos (2005) proposed a model, Saliency based on Information Maximization, which tries to find the most informative locations by maximizing Shannon's self-information from local visual feature vectors. To find these features, independent component analysis (ICA) is applied on small RGB patches from the image. The probability of detecting RGB values for a particular patch is determined using ICA bases likelihood. The same authors in (Bruce and Tsotsos 2009) further elucidate saliency as self-information of the visual features, by extending the method to find a joint-likelihood. In doing so, each ICA coefficient turns into a probability based on its likelihood from the probability distribution of surrounding patches coefficients. The joint likelihood for a particular region is found by the product of all comprised likelihoods. To find saliency map, the joint likelihood is converted to Shannon's measure of Self-Information. The attention model and eye movement prediction on complex scenes have been formulated using Bayesian methods as well. Using these methods, prior knowledge about the scene, such as visual attribute statistics or descriptions, can be combined with layout. Itti and Baldi (2005), developed a metric for surprise by calculating the mismatches between viewer expectations and perceived reality. This method finds the saliency map by applying center-surround linear filters on different feature channels, such as color and intensity. This approach is only advantageous to pin pointing the focus of the scene if one of the features is distinct, and not so if all the features perform evenly.

Likewise, the Bayesian framework has been employed by model SUN (Zhang et al. 2008) and (Seo and Milanfar 2009a) to study fixations. The SUN model attempts to detect saliency by estimating the probability of presenting a target given visual features at every location in the scene. In a free viewing condition, where there is no notion of target, this model also finds bottom-up saliency using a maximum information approach. Unlike (Bruce and Tsotsos 2005) it obtains self-information by finding differences between a particular image's statistics and natural images' statistics. The challenge here is cluttered background. Consider, if the salient parts have simpler context in comparison with non-salient parts, the entropy of the former would be lower, due to the fact that they have been obtained locally. Seo and Milanfar (2009a, b) also proposed to compute some local descriptors, called local regression kernels, from images or videos to measure the likeness of a pixel or voxel to its surroundings. Visual saliency is estimated using "self resemblance" measures. Therefore, a saliency map is attained, wherein salient regions are determined by dissimilarity (using matrix cosine similarity) compared to their surroundings. This method only compares local neighborhoods and so it suffers from the aforementioned problem of local estimation.

Learning techniques which infer the model structure from the data too have also been employed in visual saliency modeling. Kienzle developed operators to detect saliency from human eye movement data using machine learning techniques using the pixel intensities of static scenes (Kienzle et al. 2007b) and Hollywood movies (Kienzle et al. 2007a). They showed that learned discriminative features have a center-surround pattern. Judd et al. (2009) also proposed a top-down method, in which a model of saliency based on low, middle and high-level image features (computed by some saliency methods) is learned from eye tracking data on static scenes. Liu et al. (2011) proposed a supervised method that uses learning to detect salient objects. Databases of manually labeled images and video segments were used for the learning task. Learning based methods are unfeasible as they not only rely on eye tracking data and manual labeling, but are also heavily dependent on training data.

Several recent works also deal with the extensions and applications of image saliency detection methods for videos. Guo et al. (2008) proposed spatiotemporal saliency detection in frequency domain by extending a two-dimensional Fourier Transform to a quaternion Fourier Transform. Zhang et al. in (2009) extended their model to videos by applying spatio-temporal filters on video frames and computing the features. The bottom-up saliency map is then computed

using these features. In (Mahadevan and Vasconcelos 2010) spatio-temporal cuboids are modeled by using dynamic textures based on the center-surround contrast hypothesis. In (Zhai and Shah 2006) a spatiotemporal video attention detection technique was proposed to detect attention regions and interesting actions in video sequences. Interest-point correspondences and geometric transformations between images are used to compute the motion contrast in the scene. For the spatial attention model, a pixel-level saliency map is computed using color histograms.

Some more current methods attempt to learn a model from gaze data, with the aim of detecting saliency in videos (Rudoy et al. 2013) or using obtained saliency maps to accomplish action recognition tasks (Mathe and Sminchisescu 2012b). These methods are mostly dependent on gaze points, and it is well known that cumbersome amount of effort goes into capturing data from different subjects. Also, the authors in Zhong et al. (2013) proposed a dynamic consistent optical flow model based on human visual dynamic continuity assumption. They exploit a face detector and spatial saliency models (e.g. Itti et al. 1998) to find a spatio-temporal attention model. Many methods (e.g. Zhang et al. 2008) place emphasis on object boundaries and assign high saliency to borders rather than salient regions. In contrast, saliency maps obtained by gaze locations show that the object regions are most frequently the target of interest. To address this issue, we incorporate super-voxels and early video segmentation to saliency detection. Previous methods (e.g.Vig et al. 2012) mostly depended on training videos and learning features for saliency from these videos. However, the visual attributes for a region need to be distinctive, irregular and infrequent for a region to be salient. Toward this end, we detect saliency by finding irregularities in videos via sparse representation. Furthermore, our method, which does not require any training videos, is able to deal with cluttered background and videos with noise due to the fact that it does not merely consider local contrast or saliency in small areas.

3 Our Approach

We decomposed a video into salient and redundant parts, where the salient parts are sparse and the redundant parts correspond to homogeneous and highly regular portions of videos. Let F represent a features matrix, whose columns correspond to features from frames of a video. Our aim is to decompose F into low rank matrix L, and sparse matrix S, as follows

$$F = L + S. \tag{1}$$

Thus, the problem can be formulated as low-rank and sparse recovery, for which Robust PCA (RPCA) (Wright et al. 2009) can be used to solve. RPCA attempts to decompose the given matrix F, into the low-rank matrix and the sparse matrix by solving the following optimization problem

$$\begin{aligned} &\min_{L,S} \ rank(L) + \lambda \|S\|_0 \ , \\ &s.t. \quad L + S = F \ and \ \|S\|_0 \leqslant k. \end{aligned} \tag{2}$$

If this problem can be solved for appropriate λ, L and S may be recovered exactly to generate the data F. However, (2) is a highly nonconvex optimization problem, and there is no known efficient solution for it. The low rank matrix computation problem and the L_0-minimization problem are both NP-hard and difficult to approximate. Since the formal hardness result for (2) is not known, the reasonable guess is that it is NP-hard (Wright et al. 2009). By using the relaxed convex alternative, in which L_0-norm is replaced with L_1-norm and the rank with the nuclear norm, a tractable optimization problem is obtained,

$$\begin{aligned} &\min_{L,S} \ \|L\|_* + \lambda \|S\|_1 \ , \\ &\quad s.t. \ L + S = F, \end{aligned} \tag{3}$$

where $\|L\|_*$ is the nuclear norm of L and $\|S\|_1$ is L_1-norm. The rank of a matrix is the number of nonzero singular values, so an alternative for the rank function in (2) could be a nuclear norm, which denotes the trace norm of the matrix, then (3) minimizes the sum of the singular values over the constraint set.

The main objective here is to find a feature space in which the assumption of non-salient parts being low-rank and salient parts being sparse remains valid. The connection between sparsity and saliency is due to the fact that the human vision system is attracted to informative rare scene regions and processes merely a small amount of the entire observed information (Koch et al. 2006; Borji and Itti 2013). Hence, we use sparse representation as mid-level features. In addition, correlation between redundant parts is retained via group lasso regularization (Sect. 3.1.2). In the following sections, we describe different steps of the proposed approach to obtain the appropriate features, and finally find the saliency maps.

3.1 Feature Space Selection

In this section we explain our method to obtain the feature matrix in order to decompose it into low-rank and salient matrices.

3.1.1 Low-Level Features

We divide a given video into non-overlapping cuboids of size $p \times q \times t$ and construct matrix $X = [\mathrm{x}_1, \ldots, \mathrm{x}_n] \in R^{m \times n}$

where x_i is the visual feature vector (e.g. intensity) from cuboid i.

Motivated by neuroscience studies which show that sparse coding successfully simulates the V1 population responses to natural stimuli (e.g. Olshausen and Field 1997, 2004), we propose to model videos of natural scenes as sparse representation. The idea is to represent observed data, i.e., vectorized cuboids, in terms of a linear combination of bases of a known dictionary. Assume $D = [d_1, \ldots, d_k] \in R^{m \times k}$ is a dictionary matrix. We can represent x_i as follows

$$x_i = \sum_{j=1}^{k} d_j \beta_{ji} + \varepsilon, \tag{4}$$

where d_j is an atom of the dictionary, β_{ji} is the corresponding coefficient, a scalar value, that needs to be found, and ε is a Gaussian noise. We can rewrite (4) as

$$x_i = D\beta_i + \varepsilon. \tag{5}$$

Therefore, x_i is represented by $\beta_i = [\beta_{1i}, \ldots, \beta_{ki}] \in R^{k \times 1}$ in the sparse coding space. In other words, each data point is represented as a sparse linear combination of the atom vectors in the dictionary.

Although the popular loss function used for regression problems is the Least Squares Error (minimization of residual sum of squared errors) with a penalty on the $L2$-norm regularization as follows,

$$\min_{\beta} \|X - D\beta\|_2^2 + \lambda \|\beta\|_2, \tag{6}$$

it does not impose sparsity, and the resulting coefficients have non-zero values.

To address this issue, we use lasso, proposed by Tibshirani (1996), replacing $L2$-norm regularization with L_1-norm and formulate it as follows

$$\min_{\beta} \|X - D\beta\|_2^2 + \lambda \|\beta\|_1, \tag{7}$$

where X is a matrix of observed data, D is a given dictionary of bases, and $\beta = [\beta_1, \ldots, \beta_n]$ is a $k \times n$ coefficient matrix, where each column is a sparse representation for a data point. In (7) $\|.\|_1$ denotes the entry-wise matrix L_1-norm ($\|\beta\|_1 = \sum_{i=1}^{n} \|\beta_i\|_1$), and λ is a regularization parameter that controls the sparsity level.

However, if the salient object or region is large, the number of cuboids belonging to the region will be enormous; the cuboids, which we expect to be outliers and indicate saliency, cannot be considered as sparse. It has been shown that the lasso tends to select only one data point (feature vector) from a group of highly correlated data points, and is not concerned with which one is selected (Zou and Hastie 2005). In order to overcome this, we use instead group lasso regularization to find the coefficients, in which group structure of coefficients is determined by super-voxels in a video.

Note that the goal is to select an important subset of variables imposing sparsity among the groups. Intuitively, this should drive all the weights in one group to zero together. With this approach, not only would the noise be suppressed, but also the variation in the features for finding saliency would not be as large as the sparse representation based on individual cuboids.

3.1.2 Group Lasso Regularization

We can formulate our problem as a general regression,

$$Y = D\gamma + \varepsilon, \tag{8}$$

in which, Y is a low-level feature matrix the columns of which are vectorized cuboids from the video. Y is constructed from the X matrix in a way that each division of Y consists several columns of X. D is the dictionary and γ is a coefficient matrix. Assume that Y, the feature matrix, is structured in J disjointed groups $\{G_1, G_2, \ldots, G_J\}$, $G_i \cap Gj = \varnothing$, and is represented as $Y = (Y_{G_1}, Y_{G_2}, \ldots, Y_{G_J})$ where $Y_{G_j} = (X_1, X_2, \ldots, X_{n_j})$, in which group indices are determined by the super-voxels in the video.

The group lasso is an extension of the lasso which assumes covariates are clustered in groups. It aims to obtain a regularization of the empirical error that finds a sparse solution to preserve the groups of variables together. It solves the optimization problem via $L_{1,2}$-regularization, which imposes sparsity on groups by using the sum of Euclidean norms of coefficients in each group instead of L_1-norm of each single coefficient. This could drive all the coefficients in one group to zero together, and can result in group selection (Yuan and Lin 2006). The group lasso regularization problem would be as follows

$$\min_{\gamma} \|Y - D\gamma\|_2^2 + \lambda \|\gamma\|_{1,2}, \tag{9}$$

where $\gamma = [\gamma_{G_1}, \gamma_{G_2}, \ldots, \gamma_{G_J}]$ is the matrix of coefficients that must be obtained, γ_{G_j} is a division of γ that corresponds to the j_{th} group of coefficients and consists of several columns of the coefficients matrix, and $\|\gamma\|_{1,2} = \sum_{j=1}^{J} \|\gamma_{G_j}\|_2$. The parameter λ determines the level of group sparsity to be imposed in the solution. This model assumes group structure is given. In our case, the structure is provided by super-voxels in a way that cuboids indicate feature vectors and each super-voxels consists of a group of cuboids.

3.2 Dictionary Learning

In sparse coding, we want to approximate a signal $\mathbf{x}$ over a *dictionary* $\mathbf{D}$ (which has k columns referred to as *atoms*) in

such a way that the obtained signal by linear combination of a *few* atoms is as close as possible to $\mathbf{x}$.

Various types of dictionaries have been used for this task, for example, a predefined dictionary which is based on different wavelets for natural images (Rubinstein et al. 2010a; Mallat 2009). An alternative approach determines the dictionary from the training samples using techniques such as Principal Component Analysis (PCA) and Generalized PCA. These algorithms, nevertheless, generate unstructured dictionaries which are computationally expensive to apply and limit the size of the learning dictionary because of its complexity (Rubinstein et al. 2010b). Therefore, sparse dictionaries, which are structured based on a sparsity model, have been proposed to be used in sparse signal approximation. These dictionaries perform with significantly more efficiency and function better for larger dictionaries and higher-dimensional data (Rubinstein et al. 2010b). It has also been shown that learning a structured dictionary improves signal reconstruction and results in a better representation (Elad and Aharon 2006).

Most algorithms for dictionary learning are batch-based, which access the whole data at each iteration and cannot handle large data efficiently. We resolve this by using an *online* approach that processes mini batches and uses sparse coding in the optimization procedure to find atoms. This method reduces memory consumption and lowers computational cost, hence it could be advantageous for image and video processing.

For learning dictionary on a given set of signals, in our case the cuboids of a given video, $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathrm{R}^{m\times n}$, the classic approach is to optimize a cost function

$$f_n(D) \triangleq \frac{1}{n}\sum_{i=1}^{n} l(\mathbf{x}_i, D), \tag{10}$$

where matrix $\mathbf{D}$ in $\mathrm{R}^{m\times k}$ is the dictionary whose columns are bases (atoms), and $l(\mathbf{x}, \mathbf{D})$ is the loss function that shows how "good" D is in representing $\mathbf{x}$ via a sparse representation. In the online learning method (Mairal et al. 2010), $l(\mathbf{x}, \mathbf{D})$ is defined as the result of L_1-sparse representation problem

$$l(\mathbf{x}, \mathbf{D}) \triangleq \min_{\alpha} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\alpha\|_2^2 + \lambda\|\alpha\|_1. \tag{11}$$

There is a common constraint, call it C, on the dictionary's atoms $\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_k$ having an L2-norm less or equal to one, which prevents atoms from having large values and consequently, coefficients having arbitrarily small values. A convex set of matrices validates this constraint:

$$\mathfrak{C} \triangleq \left\{\mathbf{D} \in \mathrm{R}^{m\times k}\ s.t.\ \forall j = 1, \ldots, k,\ \mathbf{d}_j^T\mathbf{d}_j \leq 1\right\}. \tag{12}$$

Since the cost function $f_n(D)$ is not convex with respect to D, it is rewritten as a joint optimization problem with respect to the dictionary D and the coefficients α of the sparse decomposition. While the function in Eq. (11) is not jointly convex, when one of the two variables D or α are fixed it becomes convex with respect to the other:

$$\min_{D\in\mathfrak{C},\alpha\in\mathbb{R}^{k\times n}} \sum_{i=1}^{n}\left(\frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\alpha_i\|_2^2 + \lambda\|\alpha_i\|_1\right). \tag{13}$$

To solve this problem, the common approach is alternatively minimizing one variable while keeping the other one fixed. We use SPAMS open source toolbox (Mairal 2012), which implements the aforementioned online dictionary learning method.

3.3 Representing Feature Space by Sparse Coding

Once we determine the dictionary, we need to find coefficients by solving the objective function (9). Group lasso regularization has been the subject of many studies recently, and several methods have been proposed for solving (9) (see Bach 2008; Meier et al. 2008; Roth and Fischer 2008). In this paper, we use a block-coordinate descent (BCD) approach that is an extension of the classic method to the group lasso (Yuan and Lin 2006), where minimization is performed over each group of variables. The BCD method utilizes an objective function that can be efficiently optimized over one group of variables. Each group subproblem can be solved in closed form. Another category of methods is gradient-based methods, in which gradient information is used to optimize the objective function (Liu et al. 2009). This also generates subproblems that have closed form solutions. However, Qin et al. (2010) has shown that the BCD approach often outperforms the existing gradient-based approaches. In our implementation, we use simultaneous signal decomposition methods based on block coordinate descent, which efficiently solves (9) by computing the covariance matrix DD^T first and then $D^T Y_{G_i}$. We then compute a matrix of coefficients using a Cholesky-based decomposition method (Mairal 2012).

3.4 Finding the Saliency Map

Once we transform the data matrix to feature space with sparse coding, low-rank and salient parts are recovered by using Robust PCA. The feature matrix is considered as a combination of non-salient parts in a low dimensional space, and salient objects or motion as sparse portions. Thus, given the feature matrix the augmented Lagrange multiplier method is used for recovering low-rank matrices via optimization Eq. (3) where λ balances rank and sparsity. For

Table 1 Our results in comparison to state of the art methods in Bias-Free configuration: this table summarizes the performance of our method on INB data set, in terms of AUC, comparing with the Bayesian surprise (Itti and Baldi 2009), SUNDAy (Zhang et al. 2009) and Intrinsic dimensionality methods (Vig et al. 2012)

Video	Surp	SUN	Intr.K	GL	GL-S
Beach	0.61	0.65	0.71	0.63	**0.78**
Breite strasse	0.70	0.70	**0.76**	0.60	0.75
Bridge1	0.52	0.50	0.59	0.70	**0.75**
Bridge2	0.64	0.60	0.53	0.72	**0.75**
Bumblebee	0.54	0.56	**0.63**	0.54	0.57
Doves	0.71	0.72	**0.83**	0.77	0.80
Ducks boat	0.65	0.63	**0.70**	0.62	0.54
Ducks children	0.56	0.70	**0.78**	0.65	0.66
Golf	0.67	0.77	0.77	0.81	**0.82**
Holsten gate	0.51	0.61	0.66	**0.79**	0.75
Koenigstrasse	0.60	**0.62**	0.60	0.61	**0.62**
Puppies	0.71	0.65	**0.75**	0.68	0.70
Roundabout	0.62	0.63	0.70	0.62	**0.72**
Sea	0.83	0.84	**0.86**	0.74	0.74
St Petri Gate	0.56	0.51	0.60	0.58	**0.66**
St Petri Market	0.52	0.58	0.63	0.74	**0.82**
St Petri McDonald	0.51	0.57	0.50	**0.60**	0.57
Street	0.58	0.68	0.77	0.78	**0.81**
Average	0.61	0.64	0.69	0.67	**0.71**

Bold values indicate the best result in terms of AUC achieved for each video among the methods
GL is our method and GL-S[mooth] shows the results after smoothing

an appropriate λ, the F matrix, which is the coefficient matrix computed from group lasso regularization, is estimated properly by obtained L and S matrices. There exist various methods to extract low-rank and sparse matrices by this optimization problem. We use a technique of augmented Lagrange multiplier, named ALM. This method can handle large matrices and has Q-linear convergence speed which makes it suitable for image and video processing applications. This simple implementation iteratively computes a partial SVD of a matrix and converges to the solution in a small number of iterations. The algorithm also has a faster version, i.e the inexact ALM algorithm, which requires a smaller number of partial SVDs (Lin et al. 2010).

Final step of our approach is in computing the saliency map using the sparse matrix values found in the previous step. The L_1-norm of columns of S matrix, corresponding to cuboids, indicates the saliency value. Then, saliency value of the super-voxels covering these cuboids is obtained by counting salient cuboids and normalizing them based on the super-voxel size. The higher the norm, the more salient the corresponding region (Fig. 2).

4 Experiments and Evaluation

For evaluating the proposed method, we first generate a saliency map for all regions in each video using the proposed approach. Each saliency map acts like a maximum likelihood binary classifier for each video, and determines the salient and non-salient regions. After thresholding, regions in the saliency maps that have value greater than the threshold are considered as to belong to the salient class.

4.1 Data Sets

We have evaluated our method on four different data sets. The first is INB by Dorr et al. (2010), which consists of 18 high-resolution movie clips of natural outdoor scenes. Each video is 1280 by 720 pixels in size, has 30 frames per second and is about 20 s in length. The gaze data of 54 human subjects freely viewing these videos is available. About 40,000 saccades have been extracted from the gaze data using a dual-threshold velocity based procedure. Salient locations are labeled positive by using these saccade points. Because of the latency of the oculomotor system, the gaze response to a salient event is not necessarily matched with the time of the event. Hence, some methods consider a temporal offset. However, Vig et al. (2011) have shown the average lag in natural scenes to be near zero, and so there is no need to consider any offset, therefore we do not consider temporal offset.

The second data set is the UCF Sports Action data set (Rodriguez et al. 2008), which consists of 150 videos from nine different types of actions such as *Diving*, *Horseback riding* and *Swinging*. The gaze data for this data set, including eye fixation information from 16 subjects viewing the videos, is available via Mathe and Sminchisescu (2012a, b).

Third data set is our own, UCF Saliency data set, created for this paper, which is a more challenging data set. In this data set, the quality of videos is poor, the resolution is low, and camera motion could be problematic, unlike INB and UCF Sports which have high resolution videos. This data set consists of 6 different videos from different events, such as Person Running, Moving Car, Jumping and Sailing. In Fig. 9, a set of still frames and their corresponding results are shown. In order to find ground-truth saliency maps, similar to Borji et al. (2013), we asked four subjects to mark freely some points (the average is 6), on regions in each frame of the video which they believe are important in understanding the scene or are interesting and capture the attention. This was done by using an annotation tool which was developed in our group, and subjects did not have any prior knowledge about the video. Finding the saliency maps for this data set can be considered as "human explicit saliency judgment" prediction problem, which is different from saliency predic-

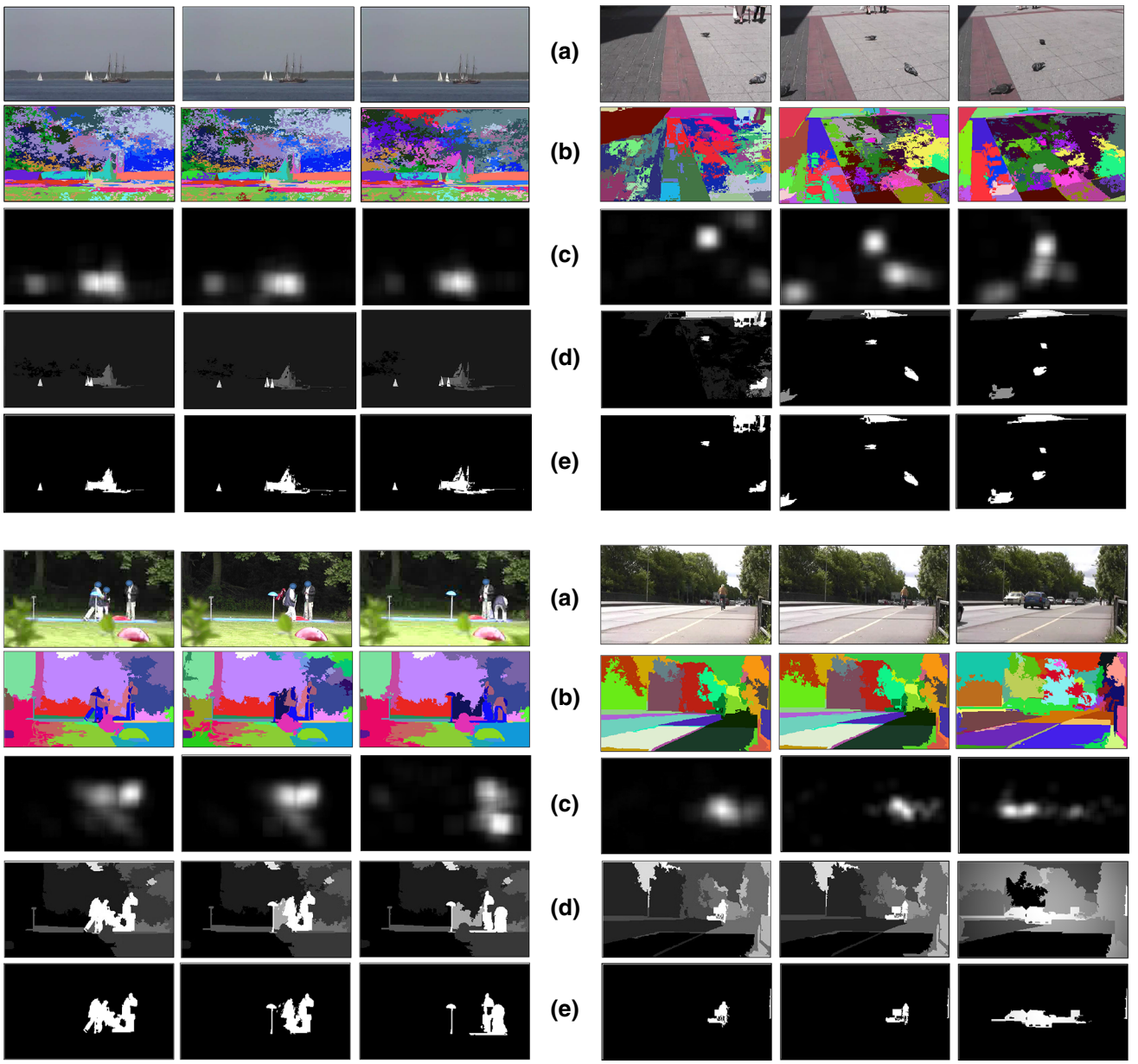


Fig. 3 Examples of frames from **a** data set videos, **b** super-voxels, **c** empirical saliency maps obtained by gaze data, **d** our saliency map results and **e** binary maps showing the most salient regions. Comparing empirical saliency maps and our results illustrated that the maxima in saliency maps is matched

tion from eye movement data. However, the results of our method for both type of data are promising, since the objects and motions in videos comprise the most informative part of data.

The last data set that we have tested our method on is Hollywood2 Actions dataset (Marszałek et al. 2009). This is a large scale dataset with camera motion and clutter, which consists of 2517 videos of which 884 are selected as a test subset. Human fixations from 16 subjects are also available for this data set (Mathe and Sminchisescu 2012a, b).

4.2 Evaluation Methods

In order to compare our method with Vig et al. (2012), we perform the same experiments, which they regard as *Bias-Free*. In doing so, we consider the set of saccade landing points in a video as a positive class, and randomly selected gaze locations from different videos are considered as a negative class. Since this labeling method leads to overlap between positive and negative samples, another labeling model has been proposed called *Default-Labeling*.

In *Default-Labeling*, for each video an empirical saliency map is generated using gaze locations. These maps specify the density of the gaze points via all subjects. At each gaze point a spatiotemporal Gaussian is placed, and for all subjects these Gaussian filters are superimposed. We use the same Gaussian filter with a spatial support of 2.4° of the visual angle, of 0.17 s temporal support, and standard deviations of 0.6° (spatial) and 600 ms (temporal). In this case, positive samples are selected from the highest density of the eye movement data in the empirical saliency map and negative class samples are picked from the lowest density. After thresholding, these saliency maps are treated as ground truth data, and for quantitative analysis we report ROC Scores (AUC area under curve). Our results are compared with ground truth data and AUC is reported for each video.

Since studies show the probability of directing attention in the center of a scene is higher, as a post-processing step we apply a Gaussian filter to smooth the map and emphasize the center in terms of saliency values. Generally, this step leads to better results in terms of predicting eye movement locations among all videos, even though for some videos AUC scores get slightly reduced.

4.3 Implementation Details and Computational Complexity

One of the primary steps in our method is grouping similar voxels in videos into meaningful segments called super-voxels. For finding super-voxels in a video, we use the Efficient Hierarchical Graph-Based Video Segmentation method. Basically, it is a spatiotemporal segmentation approach that uses hierarchical graph-based algorithm (Grundmann et al. 2010). This method is chosen based on Xu and Corso (2012), which using existing benchmarks, evaluates several video segmentation methods and concludes that the a hierarchical graph-based method is one of the best in terms of accuracy and efficiency.

In parallel, by finding super-voxels, we extract intensity feature vectors from the video cuboids. The size of the cuboids in our experiment is $4 \times 4 \times 4$. Afterward, a dictionary is created on the video feature vectors via Online Dictionary Learning for Sparse Coding method. We use SPAMS (SPArse Modeling Software) optimization toolbox for this purpose. The parameter that needs to be tuned in this phase is the number of dictionary atoms. Since the dictionary is over-complete, the number of bases must be greater than the vector size, which is 64 in our case. Therefore, we tried different numbers such as 100, 300, 640 and 1000 and found empirically that 640 is the most proper choice as a trade-off between efficiency and effectiveness. The mentioned toolbox also is used for obtaining sparse coding through group lasso regularization.

4.3.1 Computational Complexity

As shown in Fig. 2, our method consists of several steps identified by different blocks. Therefore we analyze the com-

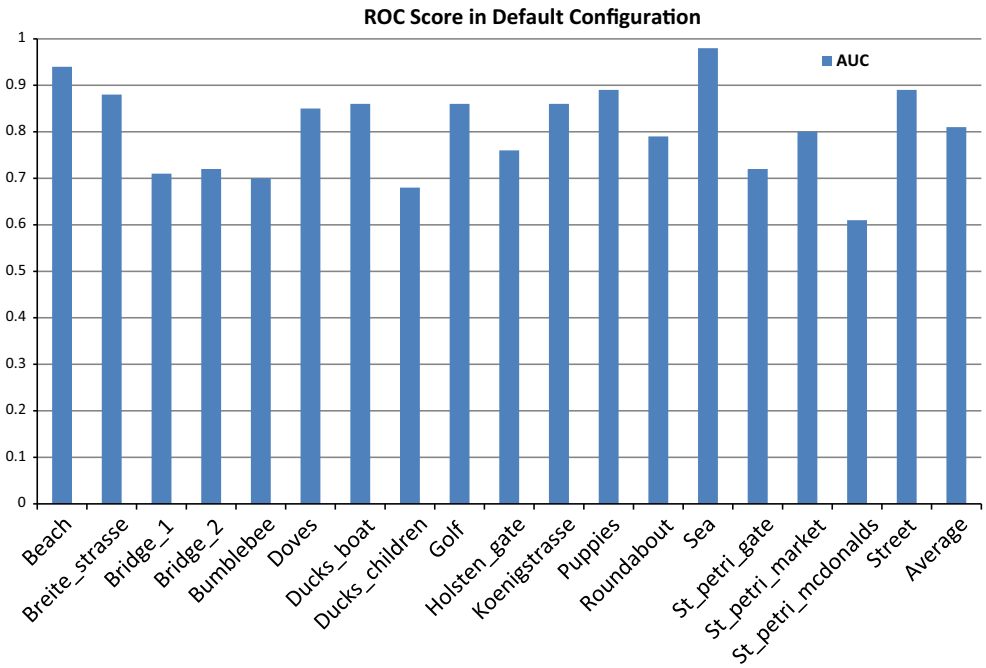


Fig. 4 The results of Default-Labeling for each video using our method and smooth version. The performance improvement over Bias-Free labeling is remarkable

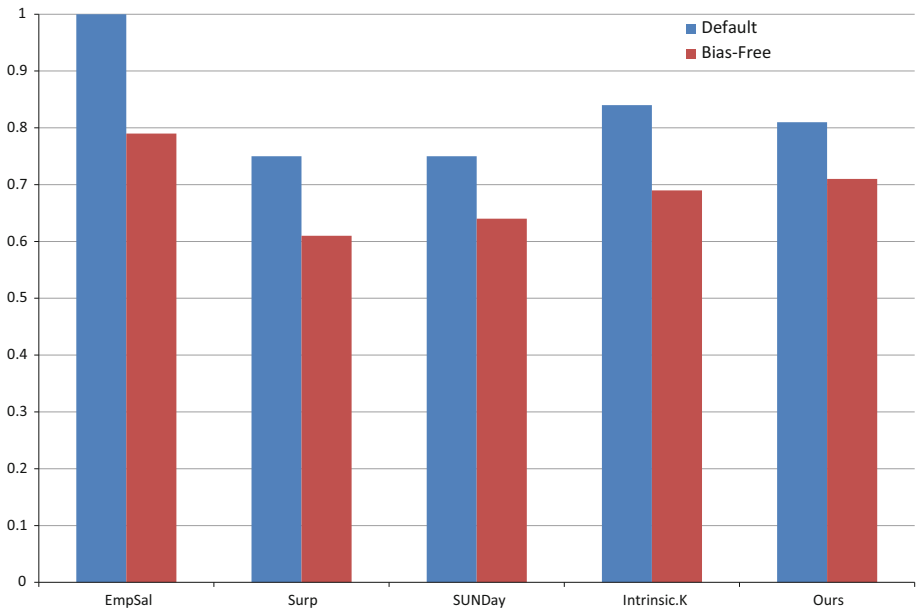
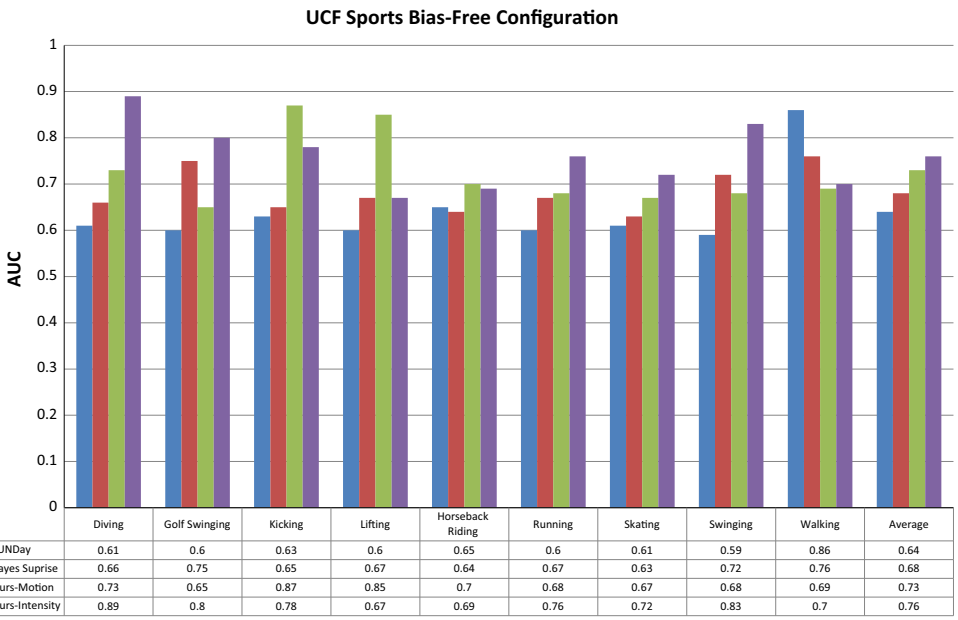


Fig. 5 Average AUC of the empirical saliency for the baseline methods: Bayesian "surprise" (Itti and Baldi 2009), SUNDAy (Zhang et al. 2009), Intrinsic dimensionality methods (Vig et al. 2012) and our model

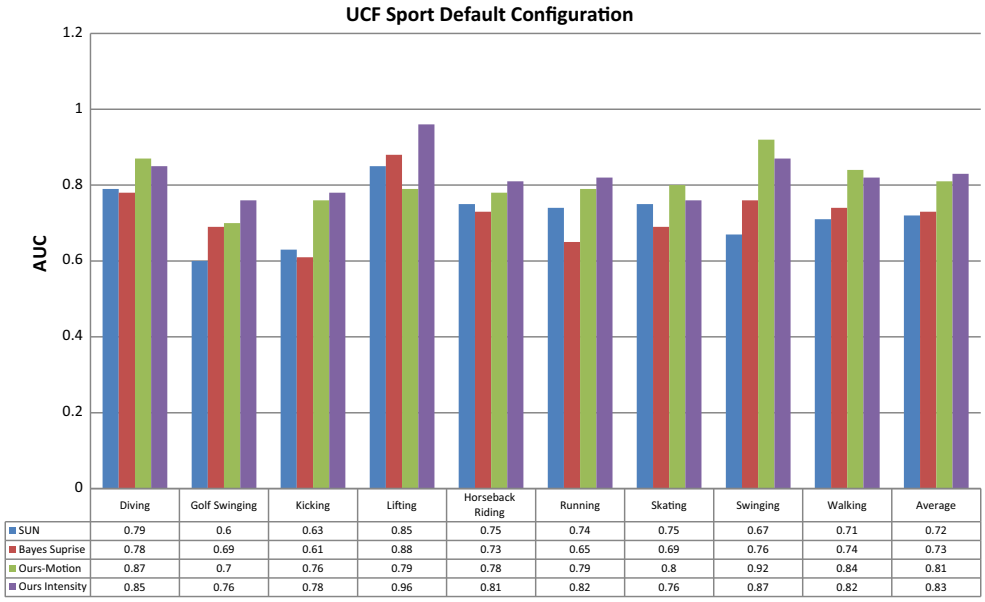


	Diving	Golf Swinging	Kicking	Lifting	Horseback Riding	Running	Skating	Swinging	Walking	Average
SUNDay	0.61	0.6	0.63	0.6	0.65	0.6	0.61	0.59	0.86	0.64
Bayes Suprise	0.66	0.75	0.65	0.67	0.64	0.67	0.63	0.72	0.76	0.68
Ours-Motion	0.73	0.65	0.87	0.85	0.7	0.68	0.67	0.68	0.69	0.73
Ours-Intensity	0.89	0.8	0.78	0.67	0.69	0.76	0.72	0.83	0.7	0.76

Fig. 6 AUC scores for videos in UCF Sports data set using Bias-Free labeling configuration

putational complexity of each step separately. The initial step is segmentation, which is linear in terms of n number of pixels in the whole clip. For learning a dictionary, we used the online learning method which solves the problem by optimizing dictionary atoms and coefficients iteratively. When the dictionary is fixed, k lasso problems are required to be optimized, where $k = n/64$ is the number of cuboids in our case, which is a fraction of number of pixels. And for a fixed coefficient matrix, optimizing the dictionary is a least squares problem of pm variables and m constraints, where p and m are respectively dimension of data matrix and number of atoms in dictionary which are constants. For instance, in our experiments with cuboids of size $4 \times 4 \times 4$, p is 64 and m, the size of the dictionary is 640, therefore this part has linear time complexity as well. In the last part of method, which is matrix decomposition, IALM method is used. This is a fast implementation of Robust PCA which has the complexity of $O[min(nm^2, mn^2)]$ where, in our method $m \ll n$, so the algorithm has linear complexity.

In our implementation, the dictionary and super-voxels are created in parallel then the saliency map is obtained. We have written our program in MATLAB code and have used a system with Intel(R) Xeon(R) CPU which has 6 cores and 12 threads with Windows 7 operating system. The memory



	Diving	Golf Swinging	Kicking	Lifting	Horseback Riding	Running	Skating	Swinging	Walking	Average
SUN	0.79	0.6	0.63	0.85	0.75	0.74	0.75	0.67	0.71	0.72
Bayes Suprise	0.78	0.69	0.61	0.88	0.73	0.65	0.69	0.76	0.74	0.73
Ours-Motion	0.87	0.7	0.76	0.79	0.78	0.79	0.8	0.92	0.84	0.81
Ours Intensity	0.85	0.76	0.78	0.96	0.81	0.82	0.76	0.87	0.82	0.83

Fig. 7 AUC scores for videos in UCF Sports data set based on Default-Labeling configuration

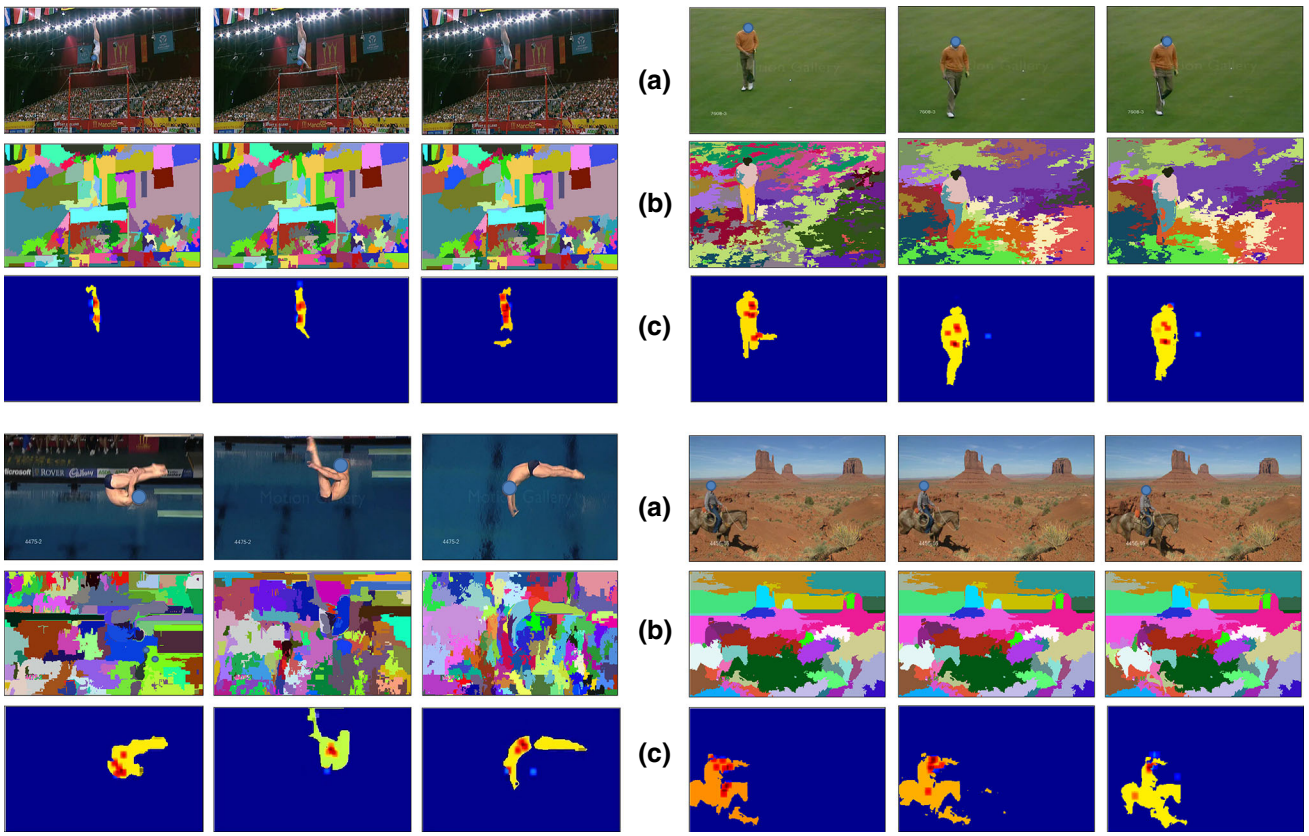
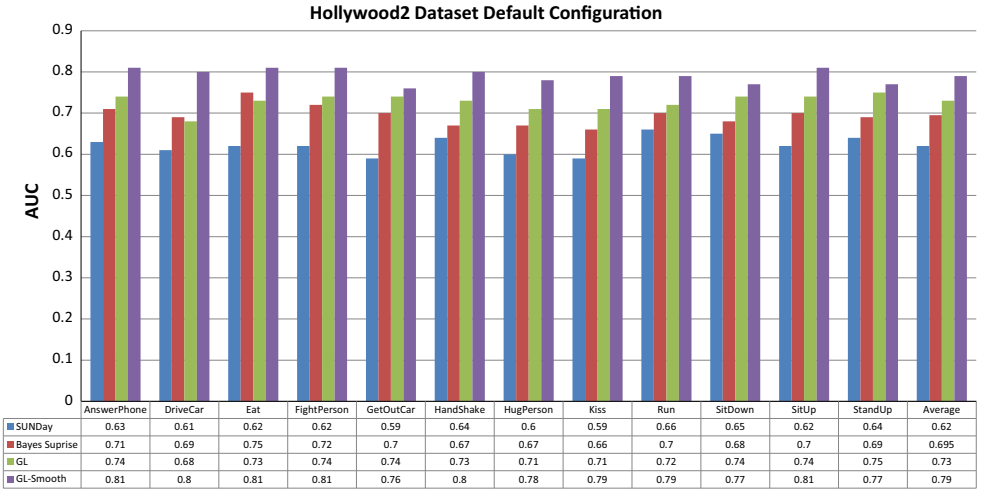


Fig. 8 Examples of frames from **a** UCF Sports data set videos, **b** super-voxels, **c** our results showing most salient regions plus gaze points shown in red considering calibration errors

of our system is 24 GB, and regardless, the program uses at most 12 GB for 12 threads. Using this configuration, finding the dictionary and generating the saliency map takes 0.83 s per frame, in other words the rate of generating the saliency map is 1.2 frames of size 320 × 240 per s. Rewriting the code in C++ or a faster platform than MATLAB could aid



	AnswerPhone	DriveCar	Eat	FightPerson	GetOutCar	HandShake	HugPerson	Kiss	Run	SitDown	SitUp	StandUp	Average
SUNDay	0.63	0.61	0.62	0.62	0.59	0.64	0.6	0.59	0.66	0.65	0.62	0.64	0.62
Bayes Suprise	0.71	0.69	0.75	0.72	0.7	0.67	0.67	0.66	0.7	0.68	0.7	0.69	0.695
GL	0.74	0.68	0.73	0.74	0.74	0.73	0.71	0.71	0.72	0.74	0.74	0.75	0.73
GL-Smooth	0.81	0.8	0.81	0.81	0.76	0.8	0.78	0.79	0.79	0.77	0.81	0.77	0.79

Fig. 9 AUC scores for videos in Hollywood2 data set based on Default-Labeling configuration

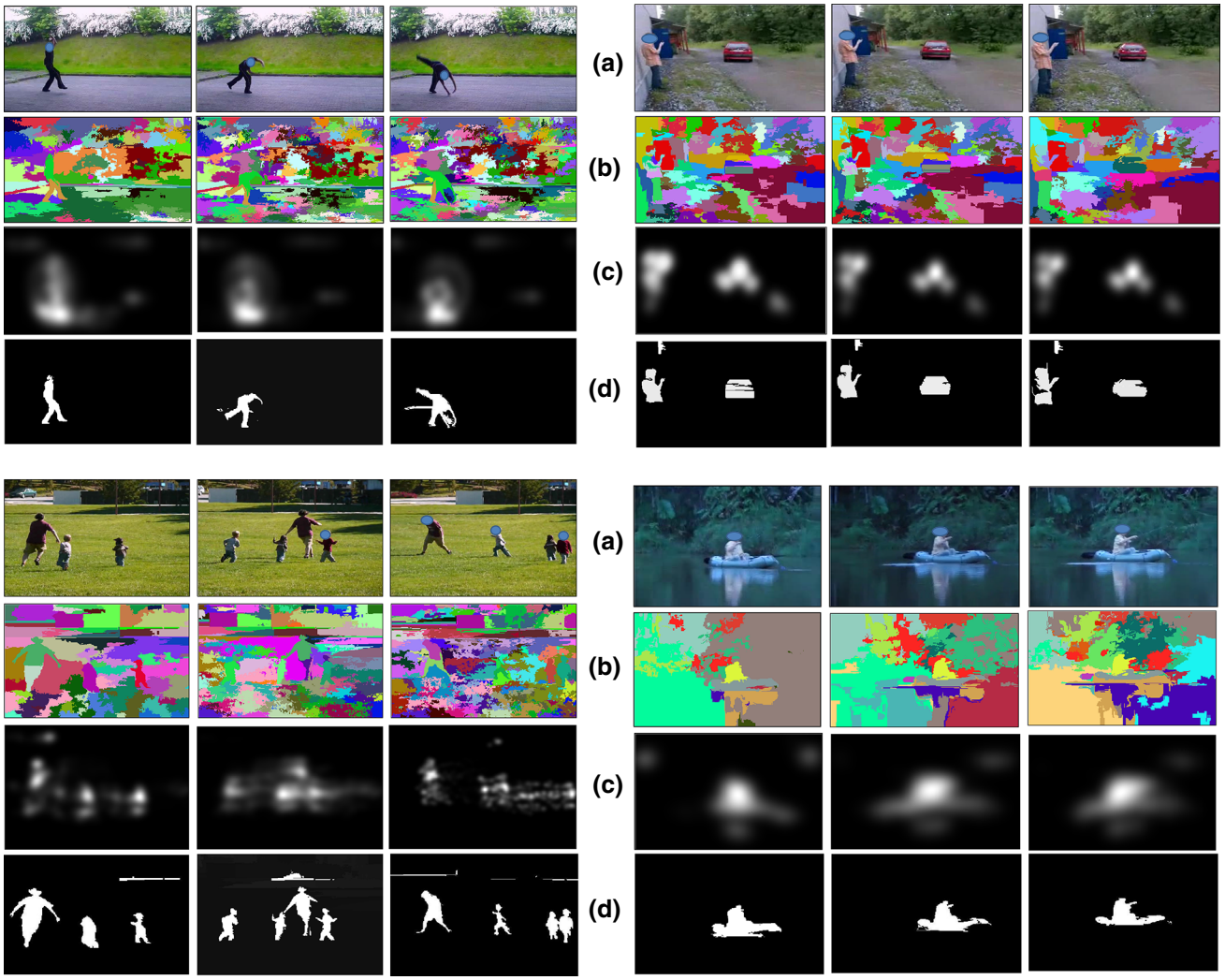
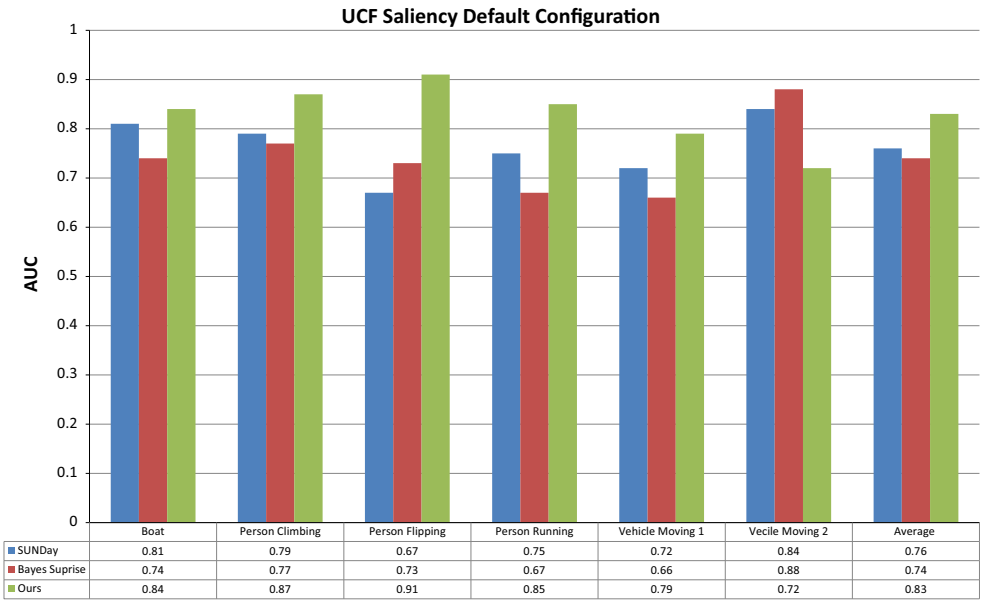


Fig. 10 Examples of frames from **a** UCF saliency data set videos, **b** super-voxels, **c** empirical saliency maps and **d** our results showing most salient regions. These salient regions correspond to meaningful objects such as person filliping, a person walking with kids, boat and car



	Boat	Person Climbing	Person Flipping	Person Running	Vehicle Moving 1	Vecile Moving 2	Average
SUNDay	0.81	0.79	0.67	0.75	0.72	0.84	0.76
Bayes Suprise	0.74	0.77	0.73	0.67	0.66	0.88	0.74
Ours	0.84	0.87	0.91	0.85	0.79	0.72	0.83

Fig. 11 AUC score for videos in UCF Saliency data set based on Default-Labeling configuration

in the performance of the program. Moreover, our method is scalable using distributed systems, different parts of the method can perform in parallel (using multi-core or multi-nodes systems).

4.4 Results

In Table 1, the results of our method and comparison with other methods using Bias-Free labeling, are reported separately for each video. As we can see, after smoothing, our final results outperform the state of the art. Also, even with no post processing (smoothing) our results are reasonable and encouraging and better than other two unsupervised methods. Also in some videos, on which other methods work poorly, such as St petri market and golf videos, we obtained better performance. It should be noted that average AUC value for empirical saliency maps using Bias-Free labeling for determining saliency locations is 0.79. Also, qualitative results for some sample frames from INB data set are shown in Fig. 3.

Furthermore, the AUC score obtained for each video via the Default labeling model is reported in Fig. 4. Most of the videos perform noticeably better in terms of ROC scores. Also, Fig. 5 shows baseline methods, and results obtained by our method using Default-Labeling. In this case, the empirical average of saliency is the upper-bound with the value of AUC being 1.

We have used the same experiments setup as aforementioned in the configuration for INB data set. In doing so, for Bias-Free labling experiments, we have used gaze points from current videos as the positive samples, and randomly selected fixations from different classes of videos as the negative ones. Since the measurements have some errors and calibration errors are provided to ensure that the data is accurate, and to get the likely positions of point-of-regard, we have used gaze samples where the calibration error is less than 0.5. For evaluation based on the Default-Labeling method, a probabilistic distribution of the gaze point is required. Therefore, we create a Gaussian model with sigma equal to the calibration error for each point, then the top 10 mixture of Gaussian are considered as salient parts. The quantitative results for these experiments for our method as well as Bayesian-surprise (Itti and Baldi 2009) and SUNDay (Zhang et al. 2009) are presented in Figs. 6 and 7. In Fig. 8 some

Table 2 Accuracy results using HOG+MBH descriptor for action recognition in UCF sports data set

Action	Baseline (reproduced) (%)	Saliency sampling (%)
Diving	100	100
Golf	100	100
Kicking	100	100
Lifting	50	100
Horse riding	100	100
Running	25	75
Skating	50	50
Swing bench	83.33	50
Swing side	100	100
Walking	71.43	85.71
Average (per video)	80.85	**85.10**

Bold value indicates the best accuracy result obtained for action recognition in UCF Sports data set

Table 3 Accuracy results using DTF descriptor for action recognition in UCF sports data set

Action	Baseline (reproduced) (%)	Saliency sampling (%)
Diving	100	100
Golf	100	66.67
Kicking	50	83.33
Lifting	100	50
Horse riding	75	100
Running	75	75
Skating	0	50
Swing bench	100	100
Swing side	75	75
Walking	71.43	71.43
Average (per video)	76.59	**78.72**

Bold value indicates the best accuracy result obtained for action recognition in UCF Sports data set

frames from sample videos, including Swinging, Walking, Diving and Horse-Riding, the corresponding super-voxels and saliency maps, in which gaze locations are indicated.

Similarly, we have tested our method on the Hollywood2 dataset, since our method is unsupervised, we have used only test subset of the data. For the sake of comparison, we have applied the SUNDay and Bayes Surprise method on this data set. As Fig. 9 indicates, the proposed method has higher performance in terms of AUC scores, and the SUNDay method has the lowest due to more emphasis on the edges and borders of regions, which is misleading in a cluttered background.

In Fig. 10, a set of still frames and their corresponding results from the UCF Saliency data set is shown. The results show the saliency maps are in accordance with the saliency distribution obtained by the salient points. Additionally, quantitative results in terms of AUC scores are reported in Fig. 11

4.5 Visual Action Recognition

Next we present an application for saliency in an action recognition problem in the UCF Sports data set. Local spatio-temporal descriptors are being widely used for action recognition in videos. In these experiments we use saliency maps to prune these features, and we show that even after discarding roughly 30 percent of descriptors, the method still outperforms the baseline. We use the bag of visual words framework for action recognition, which consists of obtaining features by descriptor extraction, K-means clustering and codebook generation, feature quantization and classification using SVM classifier. For the first experiment, we extract dense space-time interest point descriptors (Laptev et al. 2008), with a 50 overlap, using a single spatial and temporal scale. We use HOG and MBH descriptors for the first experiment. Afterward, we remove the descriptors which do not belong to the salient areas, then we generate a codebook of size 1000, and for each video we compute a histogram using the codebook. For classification, we use a non-linear SVM with a chi-squared kernel. In order to divide the data into a train set and a test set, we use a training-testing split provided in (Lan et al. 2011). In this, 103 of 150 videos in the UCF Sports data set are used for training and the 47 remaining videos for testing. We reproduced the baseline using the same framework, except the pruning part, where we use all the descriptors. In Table 2 the results are shown, as one can see using saliency, the results have been improved. This can

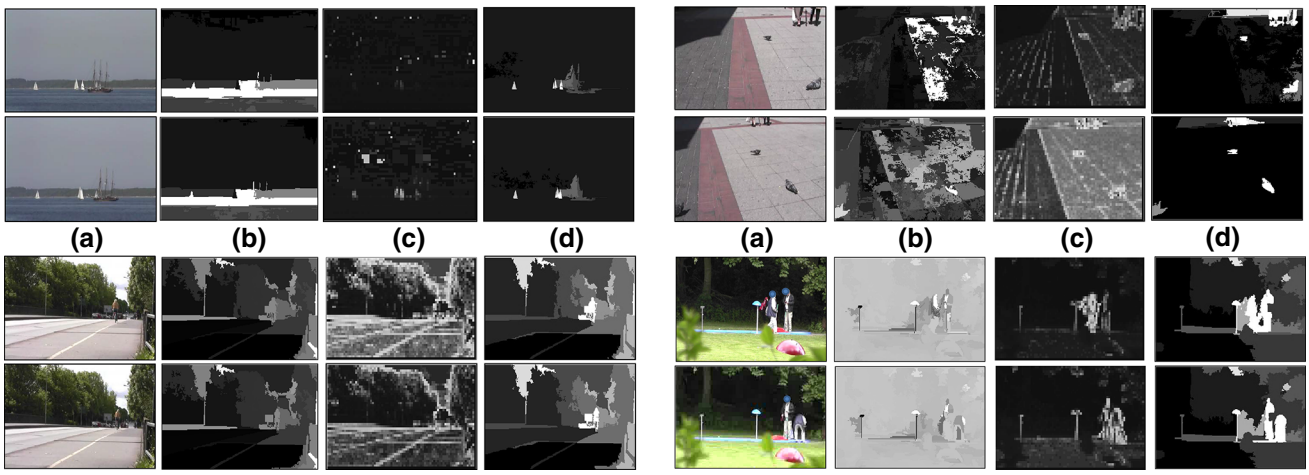


Fig. 12 Examples of results for street, sea, doves and golf video scenes from INB dataset. **a** video sample frames set, **b** saliency map using low rank decomposition on intensity data **c** saliency maps via L_1-minimization with no grouping and **d** results of our method. The AUC scores obtained by low rank decomposition are respectively 0.68, 0.56, 0.51 and 0.59. For L_1-minimization they are 0.63, 0.52, 0.54 and 0.71, which are noticeably lower in accuracy than our results

Table 4 This table shows some examples of rank reduction and imposing sparsity before and after using group lasso

Video	Rank-sparsity Non-Zero w/o GL (%)	Non-Zero w/ GL (%)
Diving 001	81	12
Walking 022	85	9
Skating 004	28	6
Kicking side 001	82	4.5
Swing side 009	72	19
	Rank w/o GL	Rank w/ GL
Diving 001	35	9
Walking 022	37	4
Skating 004	16	2
Kicking side 001	39	12
Swing side 009	33	6

Non-zero represents the percentage of non-zero elements in the sparse matrix after decomposition by RPCA, and rank shows the rank of the low-rank matrix after decomposition

be justified that by using the saliency map redundant features from the background, which are not discriminative and are common between classes, for example features from the sky, are removed. However, the approach is different from using merely foreground, the context of action is also captured by the saliency mask.

We also repeat the experiment using dense trajectory features (Wang et al. 2011a), with the same configuration as mentioned. Visibly more features are pruned in this experiment by the saliency mask. The results in Table 3 indicate that even though in some classes the accuracy drops, on average the method outperforms the baseline.

4.6 Comparison

In order to show the effectiveness of the proposed method, which uses super-voxels as the basic elements to find saliency in videos, and group-lasso regularization to provide appropriate feature space for decomposing via low-rank minimization, we have also implemented a saliency detection method using cuboids only and L_1-minimization (lasso, not group lasso). We applied and tested the latter method on some samples, which we used in our experiments, and compared the results with ours.

As Fig. 12 shows, decomposition based only on the results of the referenced baseline (Yan et al. 2010) methods is noisy and vague. The reason is, if the salient object or region is large, the number of cuboids would be enormous; and they could not be considered sparse. The assumption of the salient parts being sparse would not be valid anymore. Furthermore, this approach does not consider the correlation between variables, therefore it does not enforce that the non-salient part should have a low rank.

On the other hand, by using grouping of cuboids and utilizing group lasso regularization, as we have done, highly correlated variables are selected together and sparsity is applied among groups. Our approach does not need object detection methods or training. It is able to fairly accurately detect most dominant objects, which by using only lasso regularization and cuboids is not feasible.

One of the key aspects of our approach is that it does not use any gaze locations or labeled data to train the system, and we do not need to adjust our method to specific type of videos or objects.

For more demonstration on effectiveness of group lasso to impose sparsity and make non-salient parts low-rank, in

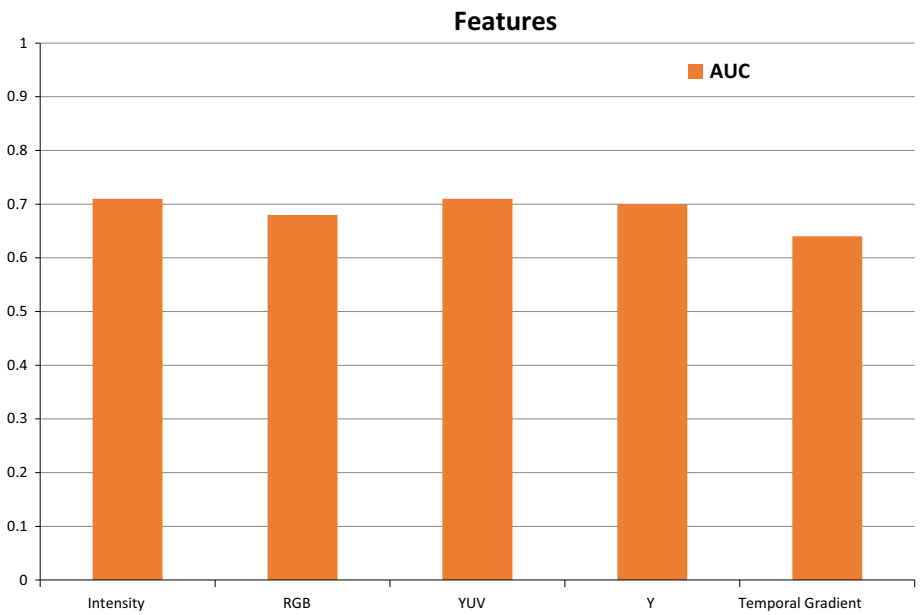


Fig. 13 AUC scores using different features: intensity, RGB, luminance channel (Y), YUV and temporal gradients features for Bias-Free labeling configuration from INB data set

Table 4 rank and sparsity percentage of data after decomposition using RPCA are shown. The first column shows using only intensity, and the second one indicates the data after applying group lasso, which shows dramatic reduction in rank as well as number of non-zero values.

We have also experimented with different initial features including intensity, RGB, luminance channel (Y), YUV and temporal gradients. As Fig. 13 shows, intensity and luminance channel have the best results, and the other features lead to slightly lower performance. In this case, the temporal gradient has the lowest AUC score. It can be explained as some videos like the bumblebee has no dominant and meaningful action in them. We also repeated this experiment for the UCF Sports data set since actions and motion are the main focus in this data set, the temporal gradient performs better than the intensity in some videos. However, the difference is not that remarkable and on average the intensity performs slightly better, therefore for the sake of consistency we have reported results for other data sets using intensity.

5 Conclusion

In summary, we present an entirely unsupervised bottom-up method that detects the regions of videos to which people's eyes are drawn. Using spatio-temporal information to represent a video as a matrix, and by using super-voxels, we group the columns into the matrix to cluster related data together. We propose using group lasso regularization to transform data into a sparse representation. In this, redundant parts remain low rank while salient parts are sparse. The correlation between the data is retained and non-salient parts tend to be of low rank. We show that without using data labeling, and learning techniques requiring eye movement data, we are able to determine salient regions of videos accurately.

Acknowledgments This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

Bach, F. R. (2008). Consistency of the group lasso and multiple kernel learning. *The Journal of Machine Learning Research*, *9*, 1179–1225.

Borji, A., & Itti, L. (2013). State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*, 185.

Borji, A., Sihite, D. N., & Itti, L. (2011). Computational modeling of top-down visual attention in interactive environments. In *British Machine Vision Conference* (pp. 1–12).

Borji, A., Sihite, D. N., & Itti, L. (2013). What stands out in a scene? A study of human explicit saliency judgment. *Vision Research*, *91*, 62–77.

Bruce, N., & Tsotsos, J. (2005). Saliency based on information maximization. In *Advances in Neural Information Processing Systems* (pp. 155–162).

Bruce, N. D., & Tsotsos, J. K. (2009). Saliency, attention, and visual search: An information theoretic approach. *Journal of Vision*, *9*(3), 5.

Dorr, M., Martinetz, T., Gegenfurtner, K. R., & Barth, E. (2010). Variability of eye movements when viewing dynamic natural scenes. *Journal of Vision*, *10*(10), 28.

Elad, M., & Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, *15*, 3736.

Frintrop, S., Rome, E., & Christensen, H. I. (2010). Computational visual attention systems and their cognitive foundations: A survey. *ACM Transactions on Applied Perception (TAP)*, *7*(1), 6.

Gao, D., Han, S., & Vasconcelos, N. (2009). Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(6), 989–1005.

Gao, D., & Vasconcelos, N. (2004). Discriminant saliency for visual recognition from cluttered scenes. In *Advances in Neural Information Processing Systems* (pp. 481–488).

Gao, D., & Vasconcelos, N. (2009). Decision-theoretic saliency: Computational principles, biological plausibility, and implications for neurophysiology and psychophysics. *Neural Computation*, *21*(1), 239–271.

Grundmann, M., Kwatra, V., Han, M., & Essa, I. (2010). Efficient hierarchical graph-based video segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2141–2148).

Guo, C., Ma, Q., & Zhang, L. (2008). Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–8).

Itti, L., & Baldi, P. (2005). A principled approach to detecting surprising events in video. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Itti, L., & Baldi, P. (2009). Bayesian surprise attracts human attention. *Vision Research*, *49*, 1295.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(11), 1254–1259.

Judd, T., Ehinger, K., Durand, F., & Torralba, A. (2009). Learning to predict where humans look. In *IEEE International Conference on Computer Vision* (pp. 2106–2113).

Kienzle, W., Schölkopf, B., Wichmann, F. A., & Franz, M. O. (2007a). How to find interesting locations in video: a spatiotemporal interest point detector learned from human eye movements. In *Pattern Recognition* (pp. 405–414). Springer.

Kienzle, W., Wichmann, F., Schölkopf, B., & Franz, M. (2007b). A nonparametric approach to bottom-up visual saliency. In *Advances in Neural Information Processing Systems*.

Koch, K., McLean, J., Segev, R., Freed, M. A., Berry, M. J, I. I., & Balasubramanian, V. (2006). How much the eye tells the brain. *Current Biology*, *16*(14), 1428–1434.

Lan, T., Wang, Y., & Mori, G. (2011). Discriminative figure-centric models for joint action localization and recognition. In *International Conference on Computer Vision (ICCV)*.

Laptev, I., Marszalek, M., Schmid, C., & Rozenfeld, B. (2008). Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008 (CVPR 2008)*.

Lin, Z., Chen, M., & Ma, Y. (2010). The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. arXiv preprint arXiv:1009.5055.

Liu, J., Ji, S., & Ye, J. (2009). *SLEP: Sparse Learning with Efficient Projections*. Tempe: Arizona State University.

Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., et al. (2011). Learning to detect a salient object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(2), 353–367.

Ma, Y. F., Hua, X. S., Lu, L., & Zhang, H. J. (2005). A generic framework of user attention model and its application in video summarization. *IEEE Transactions on Multimedia*, *7*(5), 907–919.

Ma, Y. F., Lu, L., Zhang, H. J., & Li, M. (2002). A user attention model for video summarization. In *ACM international conference on Multimedia*, MULTIMEDIA '02.

Mahadevan, V., & Vasconcelos, N. (2010). Spatiotemporal saliency in dynamic scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(1), 171–177.

Mairal, J. (2012). Spams: A sparse modeling software [online], available: http://spams-devel.gforge.inria.fr.

Mairal, J., Bach, F., Ponce, J., & Sapiro, G. (2010). Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, *11*, 19–60.

Mallat, S. (2009). *A wavelet tour of signal processing*. New York: Academic Press.

Marat, S., Guironnet, M., Pellerin, D., et al. (2007). Video summarization using a visual attention model. In *European Signal Processing Conference*.

Marat, S., Phuoc, T. H., Granjon, L., Guyader, N., Pellerin, D., & Guérin-Dugué, A. (2009). Modelling spatio-temporal saliency to predict gaze direction for short videos. *International Journal of Computer Vision*, *82*(3), 231–243.

Marszałek, M., Laptev, I., & Schmid, C. (2009). Actions in context. In *IEEE Conference on Computer Vision & Pattern Recognition*.

Mathe, S., & Sminchisescu, C. (2012a). Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition. Technical report, Institute of Mathematics of the Romanian Academy and University of Bonn.

Mathe, S., & Sminchisescu, C. (2012b). Dynamic eye movement datasets and learnt saliency models for visual action recognition. In *IEEE European Conference on Computer Vision*.

Meier, L., Van De Geer, S., & Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *70*(1), 53–71.

Navalpakkam, V., & Itti, L. (2006). An integrated model of top-down and bottom-up attention for optimizing detection speed. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, *37*(23), 3311–3325.

Olshausen, B. A., & Field, D. J. (2004). Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, *14*(4), 481–487.

Poirier, F. J., Gosselin, F., & Arguin, M. (2008). Perceptive fields of saliency. *Journal of Vision*, *8*(15), 14.

Qin, Z., Scheinberg, K., & Goldfarb, D. (2010). Efficient block-coordinate descent algorithms for the group lasso. *Mathematical Programming Computation*, *5*, 143.

Rensink, R. A., O'Regan, J. K., & Clark, J. J. (1997). To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science*, *8*(5), 368–373.

Rodriguez, M. D., Ahmed, J., & Shah, M. (2008). Action mach: a spatio-temporal maximum average correlation height filter for action recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition*.

Roth, V., & Fischer, B. (2008). The group-lasso for generalized linear models: uniqueness of solutions and efficient algorithms. In *International Conference on Machine Learning* (Vol. 104).

Rubinstein, R., Bruckstein, A. M., & Elad, M. (2010a). Dictionaries for sparse representation modeling. *Proceedings of the IEEE*.

Rubinstein, R., Zibulevsky, M., & Elad, M. (2010b). Double sparsity: Learning sparse dictionaries for sparse signal approximation. *IEEE Transactions on Signal Processing*, *58*(3), 1553–1564.

Rudoy, D., Goldman, D. B., Shechtman, E., & Zelnik-Manor, L. (2013). Learning video saliency from human gaze using candidate selection. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1147–1154).

Seo, H. J., & Milanfar, P. (2009a). Nonparametric bottom-up saliency detection by self-resemblance. In *Computer Vision and Pattern Recognition Workshops* (pp. 45–52).

Seo, H. J., & Milanfar, P. (2009b). Static and space-time visual saliency detection by self-resemblance. *Journal of Vision*, *9*(12), 15.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, *58*, 267.

Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, *12*(1), 97–136.

Triesch, J., Ballard, D. H., Hayhoe, M. M., & Sullivan, B. T. (2003). What you see is what you need. *Journal of Vision*, *3*, 9.

Ungerleider, S. K., & Leslie, G. (2000). Mechanisms of visual attention in the human cortex. *Annual Review of Neuroscience*, *23*(1), 315–341.

Vig, E., Dorr, M., Martinetz, T., & Barth, E. (2011). Eye movements show optimal average anticipation with natural dynamic scenes. *Cognitive Computation*, *3*(1), 79–88.

Vig, E., Dorr, M., Martinetz, T., & Barth, E. (2012). Intrinsic dimensionality predicts the saliency of natural dynamic scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(6), 1080–1091.

Wang, H., Klaser, A., Schmid, C., & Liu, C.-L. (2011a). Action recognition by dense trajectories. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wang, J., Wang, Y., & Zhang, Z. (2011b). Visual saliency based aerial video summarization by online scene classification. In *International Conference on Image and Graphics* (pp. 777–782).

Wright, J., Ganesh, A., Rao, S., Peng, Y., & Ma, Y. (2009). Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *Advances in Neural Information Processing Systems* (pp. 2080–2088).

Xu, C., & Corso, J. J. (2012). Evaluation of super-voxel methods for early video processing. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1202–1209).

Yan, J., Zhu, M., Liu, H., & Liu, Y. (2010). Visual saliency detection via sparsity pursuit. *IEEE Signal Processing Letters*, *17*(8), 739–742.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*, 49.

Zhai, Y., & Shah, M. (2006). Visual attention detection in video sequences using spatiotemporal cues. In *ACM international conference on Multimedia* (pp. 815–824).

Zhang, L., Tong, M. H., & Cottrell, G. W. (2009). Sunday: Saliency using natural statistics for dynamic analysis of scenes. In *Annual Cognitive Science Conference* (pp. 2944–2949).

Zhang, L., Tong, M. H., Marks, T. K., Shan, H., & Cottrell, G. W. (2008). Sun: A bayesian framework for saliency using natural statistics. *Journal of Vision*, *8*(7), 32.

Zhong, S.-h., Liu, Y., Ren, F., Zhang, J., & Ren, T. (2013). Video saliency detection via dynamic consistent spatio-temporal attention modelling. In *AAAI*.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320.