González-Díaz, Ivá; Buso, Vincent; Benois-Pineau, Jenny (2016). Perceptual modeling in the problem of active object recognition in visual scenes. *Pattern Recognition,* v. 56, pp.: 129-141.

DOI: https://doi.org/10.1016/j.patcog.2016.03.007

Perceptual modeling in the problem of active object recognition in visual scenes

Iván González-Díaz[a], Vincent Buso[b], Jenny Benois-Pineau[b]

[a] *Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Leganés, 28911, Madrid. e-mail: igonzalez@tsc.uc3m.es. This research was carried out when he was working at the LaBRI.*
[b] *LaBRI, Laboratoire Bordelais de Recherche en Informatique, Université de Bordeaux, 33405 Talence, France. e-mail: {vbuso, benois-p}@labri.fr*

Abstract

Incorporating models of human perception into the process of scene interpretation and object recognition in visual content is a strong trend in computer vision. In this paper we tackle the modeling of visual perception via automatic visual saliency maps for object recognition. Visual saliency represents an efficient way to drive the scene analysis towards particular areas considered 'of interest' for a viewer and an efficient alternative to computationally intensive sliding window methods for object recognition. Using saliency maps, we consider biologically-inspired independent paths of central and peripheral vision and apply them to fundamental steps of the so-called Bag-of-Words (BoW) paradigm, such as features sampling, pooling and encoding. Our proposal has been evaluated addressing the challenging task of active object recognition, and the results show that our method not only improves the baselines, but also achieves state-of-the-art performance in various datasets at very competitive computational times.

Keywords: Perceptual modeling, Visual Saliency, Active Object recognition, foveal and peripheral pathways

1. Introduction

Object recognition is a very active research field for the computer vision community. For such a task, the Bag-of-Words (BoW) model [1, 2] is still one

of the most prevalent approaches due to its simplicity. However, its performance is greatly limited in case of occlusions or small objects in cluttered backgrounds. In contrast, sliding window methods have turned out to be more robust against these problems. They perform a window-based scanning process that searches for objects in several locations and scales in the image, thus addressing both the detection and accurate localization of objects even when they are small. Examples of these methods can be found in the literature for detecting faces [3], pedestrians [4], more generic objects [5], and even mixed with the BoW [6]. Nevertheless, these methods still suffer from several drawbacks: a) although efficient implementations exist, the computational complexity due to the computation of features within each candidate window, and the evaluation of the objective function cannot be neglected; b) they require a strong human effort to manually annotate bounding boxes in the training data; c) an exhaustive scanning might cause more false detections; or d) unless explicitly incorporated, context information around the object that might become a valuable cue of its presence is usually discarded.

Alternatively, modeling the selective process of human perception of visual scenes represents an efficient way to drive the scene analysis towards particular areas considered 'of interest' or 'salient'. This is why it has become a very active trend in computer vision [7]. Due to the use of saliency maps, the search for objects in images is more focused, thus improving the recognition performance and additionally reducing the computational burden. Even more, saliency methods can be naturally applied to both BoW [8] and sliding window approaches [9, 10].

Models of visual attention, such as the one proposed by Itti et al. [11] or Harel's graph implementation [12] are frequently used in literature for computing saliency maps. Various authors have shown how driving the processing to those particular areas with high values in the saliency maps improves the system performance in various computer vision tasks, such as image retrieval [13], object recognition [14, 15], object tracking [16, 17], or action recognition [18, 19]. However, although much fundamental work has been done to generate good representations of visual saliency from still images or video content, their ap-

plication to object recognition has not been yet explored in-depth. Indeed, it is still commonly restricted to a pre-processing stage that filters out non-relevant areas from the process [8].

In this paper, therefore, we provide a systematic study of the application of saliency to the challenging task of active object recognition. In a given scene, active objects are those objects which are interacted (manipulated, observed) by the users and, therefore, play a key role to understand the semantics of the scene. Furthermore, we claim that, in many scenarios in which humans perform activities by manipulating objects, an action can be effectively defined as a sequence of 'active' objects [20]. Hence, we do not aim to detect every object in the scene, but only those ones considered as active. This problem fits well with the nature of saliency since it aims to drive the recognition process to the areas of interest in the image, therefore preventing from the detection of non-active objects that belong to the background of the scene.

Our saliency-based approach aims to model the retina in the Human Visual System (HVS), and consider biologically-inspired independent foveal and peripheral visual paths. By plugging our contributions in the BoW paradigm, we investigate how visual attention modeling can be applied to various steps in the processing pipeline. To the best of our knowledge, this is the first in-depth study about the application of visual saliency to object recognition with BoW approach at several stages, as: i) we extend the state-of-the-art on *Saliency-sensitive non-uniform feature sampling* in a new *Saliency-sensitive variable-resolution feature space*, ii) we introduce a completely new *Saliency-Sensitive Coding of features* and use the iii) *Saliency-based feature pooling* which has been shown to be efficient in referenced research [20, 13].

The benefits of our approach are multiple: i) the computation of saliency maps is category-independent and a common step for any object detector, ii) compared to sliding window methods, by looking at the salient area we can avoid much of the computational overhead caused by an exhaustive scanning process, iii) our automatic saliency maps not only focus on the object of interest of a scene but usually contain some context around the object, iv) an object

recognition method working with saliency maps does not need ground-truth bounding boxes for training, which dramatically reduces the human resources devoted to the database annotation. In contrast, a known limitation of the use of saliency is that, as it focuses on the objects/area of interest of the scene, it may prevent systems from detecting those objects located outside these areas and that belong to the background of the scene.

In order to assess these benefits, we have selected an experimental benchmark composed of both video and image datasets containing scenes in which just a few objects are considered and have been manually labeled as active. The video datasets are 1st-person camera view (egocentric videos), which have recently gained a lot of attention due to the emerging end-user applications involving the use of wearable cameras in scenarios such as robotics, telemedicine or life-logging [21]. Furthermore, as this kind of content fits well with the problem being addressed, we can find previous works in the literature that have previously applied visual saliency to egocentric video analysis [22, 8, 23, 24]. On the contrary, the image datasets are 3rd-person camera view and demonstrates that our method is not restricted to egocentric contents.

The remainder of the paper is organized as follows: in section 2 we discuss the work related to the application of perceptual modeling to computer vision and, particularly, to object recognition. Next, in section 3, and just for the sake of completeness, we provide a brief description of the method used to compute saliency in video. Section 4 describes in detail our saliency-based approach for active object recognition. In section 5 an in-depth evaluation is provided that assesses our model under the various scenarios, and compares it to other state-of-the-art approaches. Finally, section 6 summarizes our conclusions and gives perspectives.

2. Related Work in Saliency-based Object Recognition

Modeling visual perception in the problem of object recognition consists in the automatic prediction of the areas in the scene which, by their spatial,

luminance, color and motion properties, would attract human gaze [12]. This is the so-called "bottom-up" visual attention prediction. The rationale of using such low-level prediction is in the hypothesis that objects are characterized by peculiarities in these description channels. There exist different predictors for visual attention: e.g. those that predict the dynamics, that is saccadic motion [25], or those which focus on fixations [26]. Furthermore, the predicted visual attention is often expressed in the form of "saliency maps" [11, 8].

In any case, this paper does not focus on the particular method to compute saliency but, alternatively, studies how this valuable information can be plugged into an object recognition pipeline. In general, previous approaches tackling this problem can be broadly divided into three categories: methods using *binary segmentation masks*, *saliency-based pooling*, and *saliency-based sampling*.

Traditionally, most works have relied on binary saliency maps, also known as foreground masks, as a way to delimit the particular area of the image to be processed. This is the case of [27], where object matching is improved by filtering out the local descriptors located in non-salient areas, or the more recent proposal [8], where the authors incorporated foreground masks to the BoW paradigm by restricting the detection of local features to particular salient areas of the image. A similar approach is followed in [22], where a method for object recognition in egocentric video firstly identifies foreground areas in each frame, and consequently detects and labels regions associated with the hands and the object being manipulated.

Following the second strategy, the works in [13, 15] substitute these binary masks by a soft-pooling scheme over real-valued saliency maps. In particular, both works build over the BoW paradigm, and consider the continuous values of a saliency map to weigh the contribution of each visual word. In addition, in [13] two complementary image signatures are considered: one associated with the foreground, and another modeling the background. These signatures enable foreground and background-based object recognition, or even combined recognition in which both the object of interest and the context are considered. In [14], a discriminative approach for pooling visual features is proposed that integrates

within a unified framework the computation of saliency maps and the learning of SVM-based classifiers. In this case, saliency maps are category-dependent functions that learn the spatial distribution of visual words associated with particular object categories. This approach has been successfully applied to various computer vision tasks, such as action recognition or scene classification.

Concerning the third category of methods, other works have used saliency to perform non-uniform sampling of local features in images, so that more information is gathered on those areas considered as salient. In [28] the authors propose a classification method based on the use of decision trees over randomly sampled square patches of different sizes. To improve this random sampling process, category-specific saliency maps store the most likely locations and scales of positive patches of each class. The works in [18] and [19] also explore the same idea in the BoW paradigm, so that local descriptors are computed over regions randomly sampled using saliency maps. Finally, [9] and [10] are yet other examples of this kind of approach, where saliency maps drive the search process of sliding-window object detectors, thus drastically reducing the number of windows being evaluated. Finally, there exist other approaches that follow the so-called "fixation point strategy", whereby they sequentially analyze a set of image representations or 'glimpses' from each visual fixation a human would perform on a scene. This is the case of [29], where a Boltzmann Machine integrates the information of several glimpses and locations of several fixations of an object.

3. Visual Saliency for Object Recognition in Video

The variety of bottom-up visual saliency models available nowadays is very rich [7]. Nevertheless, one cannot speak about a universal model: whereas for video content the model has to take into account the response of HVS on motion singularities, in stills, the models on the basis of contrast and orientation have proved to be efficient. Here, for the sake of completeness, we briefly introduce the model of visual attention that we have used for the video experiments.

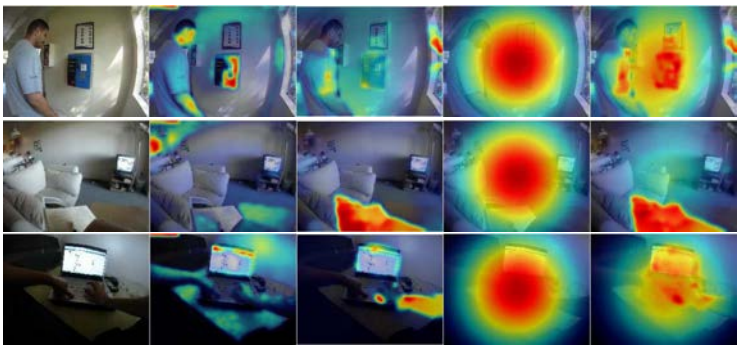


Figure 1: Three examples of visual saliency maps: from left to right, original frame, spatial, temporal, geometric and fused spatio-temporal-geometric saliency.

However, it is worth noting that our proposal does not depend on the particular saliency method and could be easily applied together with other alternatives, such as classical bottom-up methods [11, 12], or top-down attention models which are trained from human fixations, as the one proposed in [19].

We aim to model the focus of attention by means of a pixel-based saliency map on the whole video frame. Then, this map is used to guide the features selection and pooling processes in the BoW approach to generate perceptually significant features. Hence our saliency model responds to these requirements as confirmed by previous psychovisual experiments in [30]. Our method considers three sources of information, namely: a) *Spatial saliency*, with the method described in [31, 32]; b) *Temporal saliency*, associated to the residual motion once the camera motion is estimated, parametrized and compensated [33]; and c) *Geometric saliency*, which follows a previously confirmed hypothesis on general purpose video: the so-called center bias hypothesis, that is the attraction of human gaze to the geometrical center of an image [31] and is computed as 2D Gaussian located at the screen center as in [30]. This saliency is specially tailored for egocentric video content, as it approximately models the gaze of the subject wearing the camera and interacting with the scene.

Once all the individual saliency maps are computed, a weighted linear fusion is performed to generate a unified spatio-temporal-geometric saliency map $S(i)$,

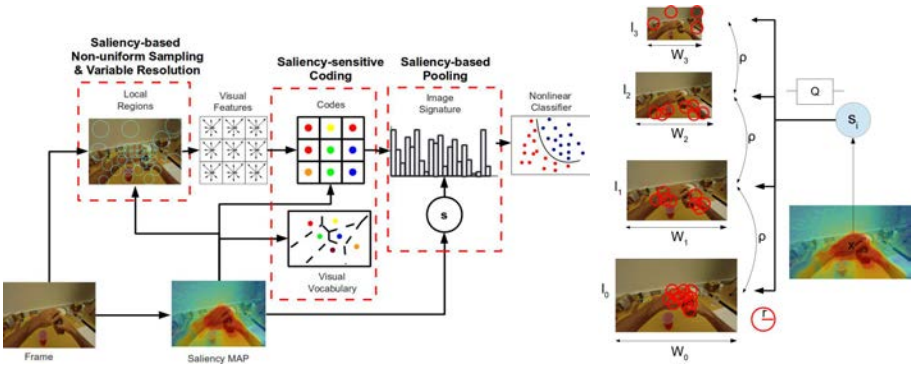
that is then normalized to the interval $[0, 1]$. Fig. 1 contains three visual examples in which a particular information channel is of special importance, and shows how our unified saliency maps successfully cover the object of interest in the frame. The interested reader is referred to the supplementary material for a detailed description of the approach.

4. A saliency-based approach for active object recognition

In this section we will describe our approach for active object recognition using saliency. As shown in Figure 2(a), we take the BoW paradigm as our baseline, and propose to improve its spatial precision using saliency maps.

Our baseline implementation of the BoW is briefly described as follows: for each frame/image, we extract a set of N local descriptors using a dense grid of overlapped circular patches. Based on several experiments, we have set the radius of the circular patches to 25px, and the step size between each local patch to 6px. Next, each local patch $n = 1..N$ is described using a 64-dimensional SURF descriptor x_n [34], which has shown similar performances to the SIFT descriptor [35] in our experiments. Each descriptor x_n is then assigned to the most similar word $b_k, k = 1..K$ in a visual vocabulary by following a vector-quantization process. The visual vocabulary B, computed using a k-means algorithm over a large set of descriptors in the training dataset (we use about 1M descriptors), has a size of K visual words. The vector-quantization process allows the generation of image signatures as L1-normalized histograms H of word occurrences. Finally, to detect the presence of a category in the image, we use an SVM with a nonlinear χ^2 kernel, which has shown good performances working with normalized histograms [36].

In parallel, our system generates a saliency map S of the frame which is used to model two differentiated pathways found in retinal vision: *foveal or central* vision, and *peripheral* vision. It is known that, due to the varying morphology of neurons in the retina, the human eye simultaneously allows for a high-resolution and detailed perception in the visual field associated with the



(a) (b)

Figure 2: Processing pipeline for saliency-based object recognition in first-person camera videos. a) A general view of the pipeline, where the three modules that incorporate saliency are surrounded with red dotted lines. b) Detail of the module ''Saliency-based Non-uniform Sampling and Variable Spatial Resolution", where fixed-size circular patches are sampled over an image pyramid based on saliency values.

fovea, and a low-resolution one in the peripheral visual field [37]. Furthermore, the human perception of a scene is based on information acquired during periods of relative gaze stability known as fixations [38]. For each fixation, a well-defined location of the image corresponds to the fovea location, whereas the rest of the image is associated with the peripheral visual field. Consequently, given the saliency map of an image, our system models both pathways at various stages of the processing pipeline (denoted with red dotted lines in Fig. 2(a)).

In the following sections, we will describe each processing module using saliency. It is worth noting that our objective is to improve system performance, while keeping the computational burden of the final solution as bounded as possible. Hence, an important requirement is that the enhancement in the performance is not achieved at the expense of a dramatic increase in the computational time.

4.1. Feature Computation by Saliency-based Non-Uniform Sampling in a Variable-Resolution space

In this section we describe our approach towards the emulation of visual fields through the Non-Uniform Sampling of features at Variable Spatial Resolutions (NUS+VSR). As already mentioned, due to the varying morphology of neurons in the human retina, it simultaneously enables a high-resolution detailed perception in the visual field associated with the fovea, and a low-resolution one in the peripheral visual field [37]. This leads to a non-uniform spatial resolution image analysis, commonly known as *foveation* [39].

Our approach combines space-variant sampling [18] and multi-resolution image foveation [40] to model these two differentiated pathways. The NUS+VSR approach is depicted in Fig. 2(b). We implement the non-uniform sampling as a pruning process that considers and initial set of features, and then filters out many of them depending on their saliency value. Hence, the first step of our approach is to define a grid of circular local patches of radius r which is more dense than the one of the baseline BoW (we use here a step size of 3px, whereas in BoW it was 6px). Let us note that this step just involves the definition of the grid (which is not costly at all) and that the computation of the descriptors (which requires an important computational burden) is just made after the pruning process is finished.

In order to simplify the model description, we split it into its constituent elements: variable spatial resolution and non-uniform sampling.

4.1.1. Variable Spatial Resolution (VSR)

In order to provide a multi-resolution analysis of an input image, we first discretize the resolution space by generating a multi-scale Gaussian image pyramid of L levels. Lower levels are meant to represent foveal vision whereas upper ones model peripheral vision. As shown in Fig. 2(b), we define the *scale resolution factor* ρ that stands for the ratio between the widths of two contiguous images in the pyramid $W_l = \frac{W_{l-1}}{\rho}$, where l denotes the level in the pyramid.

Then, using values in the saliency map, we can compute the saliency value of each local circular patch n as:

$$s_n = \underset{m \in \Omega_n}{max} (S(m)) \tag{1}$$

where $S(m)$ has been already defined Sec. 3 and Ω_n stands for the pixels within the local patch n. We have found that max pooling here is more efficient for the target recognition task than mean pooling. Hence, depending on this value $s_n \in [0, 1]$, we assign each local patch to a particular level of the pyramid. In our case, this is done by a simple linear quantization ($Q(s)$ in Fig. 2(b)) that uniformly splits the resolution space into equally sized segments. Intuitively, based on the saliency value, we are modeling the foveal visual field as a high-resolution pathway that pays attention to small image details, whereas the peripheral vision path acquires information at a lower resolution, thus focusing on coarse visual patterns.

Let us note that we do not change the scale of the local regions (defined by the radius r in Fig. 2(b)); alternatively, we decrease the size of the image in each level so that the relative size of the local regions increases with the level l. Indeed, the spatial location of each local patch in the initial grid is also adapted to the dimensions of the selected image of the pyramid. Our approach therefore discretizes the resolution space, which differs from previous works towards foveated video displays [39, 40], where continuous-resolution image representations (foveated images) were generated by interpolating previously computed discrete-resolution image representations. Although a continuous resolution space might seem appealing for our problem, its implementation leads to two problems which discourage its application: first, using interpolation between images at various resolutions and generating a foveated image based on the saliency map will lead to local regions containing pixels at various resolutions. This would produce the undesirable scenario in which local descriptors are computed over areas with non-uniform resolution. Indeed, it could be seen as a retinal visual cell working at variable resolution in its visual field, which does not coincide with our objective of modeling various visual cells, each of

them working at a specific spatial resolution. In addition, computing image interpolations on-the-fly, ensuring that all pixels in a local patch correspond to the same resolution, would solve this issue at the expense of an important increase in the computational burden. As we will show in the experimental section, increasing the number of levels L (which, if we keep the resolution of the last level as a constant, would tend to a continuous resolution space if L is large enough) does not notably improve classification results.

4.1.2. Non-uniform Sampling (NUS)

In this section we introduce the pruning process that filters out non-relevant visual information in order to provide more compact image representations focusing on areas of high saliency.

We follow a similar approach to [18], in which a Weibull cumulative distribution was proposed to perform random sampling based on saliency values. In particular, defining a random variable S associated with visual saliency, the Weibull cumulative density function obeys:

$$F_S(s) = P(S \leq s) = 1 - e^{-(s/\lambda)^{\kappa}} \tag{2}$$

where κ is called the *shape* parameter and λ the *scale* parameter. Hence, for each n-th local region with a particular value s_n we randomly decide if it is pruned or not based on the value of $F_S(s_n)$.

Intuitively, the *shape* parameter κ controls the influence of the saliency value on the pruning process. Whereas low values of κ give less influence to the saliency value (both salient and non-salient areas have similar opportunities to survive the pruning process), high values will prune almost all the non-salient local regions in the final image representation. Furthermore, for a given κ value, the *scale* parameter λ controls the total amount of local regions being pruned.

We aim at improving classification results while avoiding any additional processing burden. Hence, in order to produce almost the same number of visual descriptors as in the uniform case, we have designed the following random sampling procedure. Let us consider a *desired number* of patches N, that corre-

sponds with the number of processed patches in the baseline BoW (no saliency). Since we are following a pruning process, our initial dense grid will produce a large enough set of N_0 points so that $N_0 >> N$. Then, we will set a value for the shape parameter κ in the Weibull distribution and, in order to keep the computational complexity constant, we will automatically calculate the corresponding λ value producing a final number of points $\hat{N} \sim N$.

For that end, let us consider $\hat{N}$ as a random variable and therefore compute its expected value as:

$$E[\hat{N}] = \sum_{n=1}^{N_0} 1 \cdot F_S(s_n) = N_0 - \sum_{n=1}^{N_0} e^{-(s_n/\lambda)^\kappa} \tag{3}$$

where we have considered that the probability of a patch n being included in the final image representation is the value of the Weibull cumulative density function $F_S(s_n)$ on the saliency of the patch s_n.

Unfortunately, from eq. (3) it is not possible to obtain an analytic optimal value of λ that makes $E[\hat{N}] = N$. However, since $x_n = -(s_n/\lambda)^\kappa$ is a real value, we know that e^{x_n} is a convex function over x_n, which allows us to apply Jensen's inequality to obtain an upper bound of eq. (3) as:

$$E[\hat{N}] \leq N_0 \left(1 - e^{-\frac{1}{N_0}\sum_{n=1}^{N_0}(s_n/\lambda)^\kappa}\right) \tag{4}$$

That is, we can obtain an upper bound of the number of points being processed, which allows us to successfully keep the computational complexity bounded for each value of κ. Working out λ in eq. (4) gives a final expression for λ:

$$\lambda = \left[\frac{E[s^\kappa]}{\ln\left(\frac{N_0}{N_0 - N}\right)}\right]^{1/\kappa} \tag{5}$$

where $E[s^\kappa] = \frac{1}{N_0}\sum_n s_n^\kappa$. The upper bound in (4) is tight when the values $(s_n/\lambda)^\kappa$ are very similar for every n. This means that we get better approximations $\hat{N} \sim N$ when κ is small than when it is very large (where we might get $\hat{N} < N$). As we will show in the experimental section, where we will cross-validate the value of κ, the influence of the approximation on the results is negligible and, in fact, better results are achieved for high values of κ.

4.2. Saliency-based Pooling (SP)

This section corresponds with the stage in the pipeline that generates the image signatures using saliency. In the traditional BoW approach [2], the image signature H is the statistical distribution of the image descriptors according to the visual codebook. This is made by first assigning each local descriptor to a visual word in the vocabulary, and then computing a histogram of word occurrences by counting the times that a visual word appears in an image.

In our *Saliency-based Pooling*, we use saliency to weigh the selected features, giving place to a sort of soft-assignment based on saliency maps. In particular, the contribution of each image descriptor is defined by the weight s_n in eq. (1). In other words, descriptors over salient areas will get more weight in the image signature than descriptors over non-salient areas. Therefore, the image signature can be computed as follows:

$$H_k = \sum_{n=1}^{N} s_n \alpha_{nk} \tag{6}$$

where H_k represents the k-th bin of a histogram, and α_{nk} is an index variable so that $\alpha_{nk} = 1$ for the visual word in the vocabulary associated with the n-th descriptor in the image and $\alpha_{nk} = 0$ for the rest.

Finally, the histogram H is L1-normalized. This method of saliency weighting is similar to the spatial weighting proposed in [41] but, in our case, the weights are not learned from data as, in contrast, are directly derived from saliency, therefore being category-independent.

Furthermore, an extension of the basic saliency-pooling has been explored in [13], where the authors considered two independent signatures, foreground and background ones, which were defined using a soft fuzzy approach based on saliency. This method can be directly plugged into our perceptual approach modeling our two pathways in retinal vision. Hence, the image signature would be a concatenation of two histograms $[H_f, H_p]$:

$$H_f = \sum_{n=1}^{N} s_n \alpha_{nk}; \quad H_p = \sum_{n=1}^{N} (1 - s_n) \alpha_{nk} \tag{7}$$

where H_f stands for the foveal channel, while H_p models the peripheral one. If we keep the vocabulary length K fixed, it will produce image signatures of length $2K$, with a consequent increase in the computational complexity. Alternatively, if we divide the vocabulary length by two and keep the computational complexity constant (same signature length), we might be losing precision in the foveal representation with the new reduced vocabulary. To avoid this limitation, in the next section we reformulate our problem as follows: given a total signature length, and using saliency, we would like to optimally allocate the respective proportions for the foveal and peripheral channels.

4.3. Saliency-sensitive Coding of features (SC)

This section is devoted to the description of the *Saliency-sensitive Coding of features*. As we already mentioned, this work is inspired by previous proposals in which information belonging to foreground and background is encoded independently [13], as well as by the principles of sparse coding [42] and locality coding [43]. However, we provide a self-organized approach that automatically learns the optimal vocabularies for each spatial resolution and then assigns each visual descriptor taking into account both its visual appearance and its associated saliency value.

To do so, we start by considering the Locality-constrained Linear Coding (LLC) approach presented in [43] and [44]. In these works, some exponentially-increasing locality functions were used to provide sparse codes which represented a particular descriptor using a small subset of visual words from a vocabulary. In our case, while keeping the sparse requirement, we aim to provide a Saliency-sensitive Coding (SC) of features. The objective of SC is two-fold: first, we aim to generate particularized vocabularies for each spatial resolution so that each descriptor is coded as a linear combination of visual words acquired at close spatial resolutions; second, we aim to automatically set the optimal number of words assigned to each spatial resolution, so that more words are used to represent visually salient image locations and vice versa.

Our problem formulation is as follows: for a given set of N descriptors

defined by the pair $\{\mathbf{x}_n, s_n\}$, where $\mathbf{x}_n \in \mathbb{R}^{D\times 1}$ stands for the visual descriptor and s_n is the saliency value associated with the region (see eq. (1)), we define an over-complete visual vocabulary $\{B, \mathbf{p}\}$. The matrix $B \in \mathbb{R}^{D\times K}$ contains the visual words of the vocabulary, whereas the vector $\mathbf{p} \in \mathbb{R}^{K\times 1}$ defines the retinal path associated with each visual word. This path is a continuous variable in the range [0,1] that models a fuzzy membership to the foveal/peripheral pathways, in which 0 stands for a path completely associated with the peripheral vision (low spatial resolution) and 1 corresponds to the foveal path (high spatial resolution).

Hence, given a visual descriptor $\mathbf{x}_n$ computed at a particular spatial resolution (that depends on its associated saliency), we aim to represent it as a linear combination of a small set of visual words of the vocabulary, strengthening those of similar type (similar spatial resolution).

To that end, we formulate the following minimization problem:

$$\min_{\boldsymbol{\alpha},B,\mathbf{p}} \sum_{n=1}^{N} \left\{ \|\mathbf{x}_n - B\boldsymbol{\alpha}_n\|_2^2 + \lambda_l \|\mathbf{l}_n \odot \boldsymbol{\alpha}_n\|_2^2 + \lambda_t \|\mathbf{t}_n \odot \boldsymbol{\alpha}_n\|_2^2 \right\} \; s.t. \; \mathbf{1}^T\boldsymbol{\alpha}_n = 1 \quad (8)$$

where $\alpha_n \in \mathbb{R}^{K\times 1}$ represents the vector of weights in the linear combination and is called the *code*, $\odot$ stands for the Hadamard product (element-wise) between two vectors, and $\mathbf{1}$ represents a vector of ones. The first element in eq. (8) corresponds to the coding error between the original and the reconstructed descriptor. The second element ensures locality by incorporating a *locality adaptor* $\mathbf{l}_n \in \mathbb{R}^{K\times 1}$ to the problem. This locality adaptor, previously introduced in [43], stands for the visual distance l_{nk} between the descriptor and each word in the vocabulary. By using an exponentially-increasing adaptor of the form:

$$l_{nk} = \sqrt{\exp\left(\frac{\|\mathbf{x}_n - \mathbf{b}_k\|_2^2}{\sigma_l^2}\right)} \quad (9)$$

we are able to generate sparse codes α_n in which just a few α_{nk} associated with words that are close in the feature space get non-zero values. It is easy to notice that the lower the parameter σ_l^2, the sparser is the resulting code.

Finally, with the third term in eq. (8) we aim to code each descriptor using words in the vocabulary with similar spatial resolution. Therefore, we introduce

a new *type* adaptor $\mathbf{t}_n \in \mathbb{R}^{K\times 1}$ that compares the retinal paths of the descriptor and visual word as:

$$t_{nk} = \sqrt{\exp\left(\frac{\|s_n - p_k\|_2^2}{\sigma_t^2}\right)} \tag{10}$$

where, again, we have made use of an exponentially-increasing adaptor with its own parameter σ_t^2.

4.3.1. Approximate Inference

Since eq. (8) is independently convex in $\{\boldsymbol{\alpha}, B, \mathbf{p}\}$, we have followed a *coordinate descent - gradient descent* approach to find the optimal values. That is, by iteratively optimizing the functional with respect to each parameter, it is ensured that the algorithm will converge to a local minimum.

In particular, in order to provide a solution for the coding stage ($\boldsymbol{\alpha}$), we can rewrite eq. (8) as:

$$\min_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \boldsymbol{\alpha}_n^T C_n \boldsymbol{\alpha}_n + \boldsymbol{\alpha}_n^T diag(\lambda_l \mathbf{l}_n^2 + \lambda_t \mathbf{t}_n^2)\boldsymbol{\alpha}_n + \eta(\mathbf{1}^T\boldsymbol{\alpha}_n - 1) \tag{11}$$

where we have defined a new matrix $C \in \mathbb{R}^{KxK}$, computed as $C = (\mathbf{x}_n\mathbf{1}^T - B)^T(\mathbf{x}_n\mathbf{1}^T - B)$. We have additionally converted the equality constraint over $\boldsymbol{\alpha}$ into a new term with a Lagrange multiplier η.

Then, by computing the derivative of (11) with respect to $\boldsymbol{\alpha}_n$ and setting it to zero, we can obtain the update equations for the codes $\boldsymbol{\alpha}_n$ as:

$$\boldsymbol{\alpha}_n = \frac{\tilde{\boldsymbol{\alpha}_n}}{\mathbf{1}^T\tilde{\boldsymbol{\alpha}_n}}; \;\; \tilde{\boldsymbol{\alpha}_n} = U^{-1}\mathbf{1} \tag{12}$$

with $U = 2C + 2diag(\lambda_l \mathbf{l}_n^2 + \lambda_t \mathbf{t}_n^2)$.

Unfortunately, exact inference becomes impractical when the size K of the vocabulary increases, as computing the inverse of the matrix $U \in \mathbb{R}^{KxK}$ is very computationally intensive. Hence, we have developed an approximate inference process as follows: (1) for each descriptor, we consider a reduced vocabulary of size $\hat{K} << K$, containing only those visual words k that minimize the partial functional $\lambda_l\|l_{nk}\|_2^2 + \lambda_t\|t_{nk}\|_2^2$; (2) then, we solve the simplified problem stated

in eq. (11) for this reduced vocabulary. In our experiments, a value of $\hat{K} = 100$ has shown a good compromise between performance and complexity.

For the p_k parameter, we need to solve the following unconstrained convex optimization problem:

$$\min_{\mathbf{p}} = \lambda_t \|\mathbf{t}_n \odot \boldsymbol{\alpha}_n\|_2^2 \tag{13}$$

It is easy to note that setting the derivative of (13) with respect to $\mathbf{p}$ equal to zero leads to a nonlinear equation on $\mathbf{p}$. Hence, we can obtain an optimal value of $\mathbf{p}$ using a Newton-Raphson method that, in the iteration i updates $p_k^{(i+1)}$ as:

$$p_k^{(i+1)} = p_k^{(i)} + \frac{\sum_{n=1}^{N} \alpha_{nk}^2 \left(\mathbf{t}_{nk}^{(i)}\right)^2 \left(s_n - p_k^{(i)}\right)}{\sum_{n=1}^{N} \alpha_{nk}^2 \left(\mathbf{t}_{nk}^{(i)}\right)^2 \left(1 + \frac{2}{\sigma_t^2}\left(s_n - p_k^{(i)}\right)^2\right)} \tag{14}$$

Finally, since the term associated with the *type adaptor* does not depend on B, the dictionary can be updated by following the Newton method proposed in [44]. The interested reader is referred to that work for the complete derivation of the B update formulas.

It is worth noting that variables B and $\mathbf{p}$ are just learned in the dictionary building phase and remain fixed during the computation of the image signatures (when only the $\boldsymbol{\alpha}$ is computed). In addition, let us note that this method shows various open parameters, namely $\{\sigma_l, \lambda_l, \sigma_t, \lambda_t\}$. In the experimental section, we will show the influence of these terms and the optimal values for our problem.

5. Experiments and results

As discussed in the introduction, we aim to solve the problem of 'active object recognition' in visual scenes. For that end, we have selected several video and image datasets in which, although each frame/image may contain several of objects, only a few (in most cases one) are considered as 'active ones'. In particular, we have used four datasets which will be described in-depth in the next section: three video datasets (GTEA, ADL, Dem@care) with 1st-person camera view in which the user recording the scene is interacting with it, and

a still image dataset (PPMI) with 3rd-person camera view contents, in which users manipulate musical instruments.

5.1. Scenarios, Datasets and Evaluation Metrics

We consider two different scenarios: *constrained* and *unconstrained*.

We call *constrained* a scenario in which all the subjects perform actions in the same room and, therefore, interact with the same objects in the same context, e.g. a hospital scenario in which patients perform several activities. Here, the limited intra-class variation is only due to natural conditions: occlusions, lighting, etc. We have used two video datasets to model this scenario: GTEA and Dem@care.

GTEA is a publicly available dataset for Object Recognition [22] in egocentric video, that contains cap-mounted videos showing 7 types of daily activities, each performed by 4 different subjects, and comprising 16 categories of manipulated objects.

Dem@acare is a dataset generated under the Dem@care[1] research project. It contains 27 egocentric videos, captured by a shoulder-mounted GoPro camera with real Alzheimer patients performing various instrumental daily activities in a controlled hospital environment. This dataset contains 18 categories of active objects.

In contrast, in an *unconstrained* scenario the recordings are made at different locations and users are interacting with different instances of the same object categories. The intra-class variation here is strong and the amount of training data is small (just a few instances of each object category). For this scenario, we have used two datasets: ADL and PPMI.

ADL is a publicly available egocentric video dataset [45], which contains videos captured by a chest-mounted GoPro camera on users performing various daily activities at their homes, showing objects from 44 categories. We have just considered objects labeled as 'active' in both training and testing. This

[1]Dem@acare Project: http://www.demcare.eu/

dataset was used for two purposes: a) a reduced version was utilized to validate the free parameters of our proposal. Here a strong temporal sub-sampling was applied and the data was split into training and test sets with 1464/1251 frames respectively, ensuring that frames of the same video were not contained in both sets. And b) for the final evaluation, the 20 videos were divided into 5 sets of 4 videos each, so that a leave-one-out assessment was performed at this subset level.

The *People Playing Musical Instruments* (PPMI) dataset [46] contains images of humans interacting with 12 different musical instruments in two different ways: simply holding or playing the instrument. Therefore, each image contains an object considered as active: the instrument. In this dataset we aim to address the problem of fine-grained activity recognition with the set of 24 categories (12 instruments and 2 manipulations) and, as we will describe later in the manuscript, we have followed the experimental setup of the authors [46], which slightly differs from the rest of the experiments.

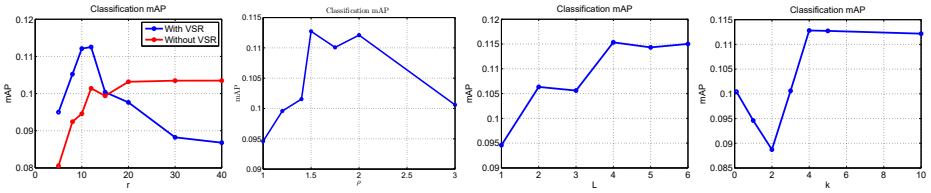
As evaluation metrics, again following the setup of the original authors, mean Average Precision (mAP) was used for the Dem@acare, ADL and PPMI, and multiclass accuracy was applied in the GTEA.

We finally isolate the case of PASCAL VOC 2010 dataset [47], as a widely used for performance comparison in object recognition problem. Generally speaking it fits to the *unconstrained* scenario. Nevertheless, we do not follow the *active object* scheme and consider all objects present in the scene instead for the sake of comparison.

5.2. Validation of model parameters

5.2.1. Variable Spatial Resolution and Non-Uniform Sampling

As we have four open parameters in this module, and although they are not independent, performing a joint fine cross-validation becomes impractical. Hence, after an initial very coarse parameter selection leading to an initial set of values, we have sequentially performed various 'one-at-a-time' optimizations, namely:



(a) (b) (c) (d)

Figure 3: Validation of the VSR+NUS parameters in the ADL unconstrained dataset: (a) radius r of the circular regions, (b) resolution factor ρ, (c) number of levels in the pyramid L, (d) shape parameter k in the Weibull distribution of the NUS.

1) *Radius r of the circular local regions*: in Fig. 3(a) we evaluate the influence of this parameter in two scenarios: with our VSR approach (blue), and (red) with a basic Dense Sampling approach corresponding to the baseline BoW (red). Let us list the values of other parameters as: $\rho = 2$, $L = 4$ and NUS activated with $k = 5$ (Section 4.1.2). Whereas the optimal value for the basic dense grid was $r = 30$, for the VSR it was $r = 10$, as going up in the resolution pyramid increases the relative size of circular regions with respect to image dimensions. Furthermore, the better results achieved by the VSR scheme demonstrate its capacity for removing very fine details and thus focusing on coarser shapes at upper levels of the pyramid.

2) *The maximum scale in VSR*: It is easy to notice that, for a given number of L levels, the sizes of images in the bottom and top levels in the pyramid can be related as $\frac{W_0}{W_{L-1}} = \rho^{L-1}$. Hence, fixing the rest of the parameters ($r = 10$, $L = 4$ and NUS activated with $k = 5$), we study the influence of the maximum scale in VSR by validating the value of ρ. As shown in Fig. 3(b), good results are obtained in the range $\rho \in [1.5, 2]$. Consequently, we have selected $\rho = 1.5$, which leads to maximum scale in VSR expressed by an approximate effective size $W_{L-1} = 0.3W_0$.

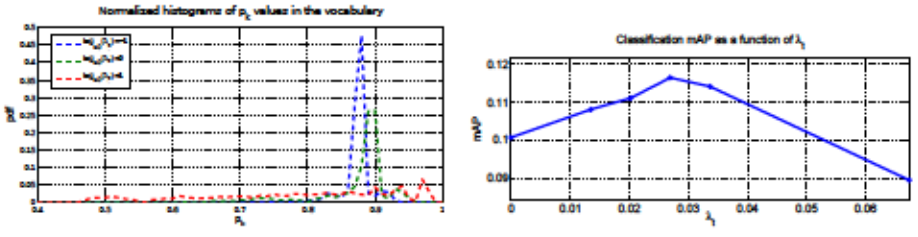
3) *The number of levels in the pyramid L*: this parameter controls the degree of discretization of the resolution space. As discussed in Sec. 4.1, in order to keep this complexity as bounded as possible, we work with a discretized resolution. To isolate the influence of L and the maximum scale in VSR, we have fixed

$W_{L-1} = 0.3W_0$ as proposed in the previous paragraph; then, for each evaluated L, we have accordingly set the resolution factor ρ that produces this W_{L-1} in the top level of the pyramid. The results provided in Fig. 3(c) demonstrate that the performance grows until $L = 4$, when it stabilizes and more continuous representations of the resolution space do not improve the performance.

4) *The shape parameter k of NUS*: for each value of k in the NUS module, we have accordingly computed the scale value λ that produces the desired final number of points $\hat{N} \sim N$ (see Sec. 4.1). The results of this study are presented in Fig. 3(d). The dependence of the AP with respect to this parameter is the only one that is neither monotonically increasing nor concave (getting a clear global maximum). Instead, here we can find how two opposite trends are preferred: very small and large values of k. This means that it is better to have classifiers that either look at the whole image, using a large context surrounding the object ($k = 0$), or focus on the area of interest (large k). Unlike other parameters, we have observed that the influence of k strongly differs from one category to another, which makes the average result more unstable. For some categories that are highly correlated with their spatial context, small values of k are preferred, as they draw descriptors in both active and non-active areas. In contrast, those categories of objects that may appear in many different scenarios or locations, are better represented with very high values of k. In average, it seems that the later are dominant in the ADL dataset. This makes high values of k more interesting (we get an optimal value of $k = 5$). For these values, the sampling process is highly unbalanced so that many more points are selected in high-saliency areas than in low ones.

5.2.2. Saliency Sensitive Coding

Since our Saliency Sensitive Coding is built over LLC [43, 44], we have firstly set the values for the locality adaptor. As it is not the scope of this paper, we do not include figures about their influence but simply note that the optimal values were $\lambda_l = 0.10$, $\sigma_l^2 = 0.25$. On the contrary, we are interested in the study of the influence of the saliency-based *type adaptor* in the coding process. Two are



(a) (b)

Figure 4: (a) Normalized histograms (pdf) of the p_k in the visual vocabulary for various values of λ_t. (b) Validation of λ_t parameter of the SC in ADL dataset.

the parameters of this adaptor (Sec. 4.3): λ_t, which controls the weight of the saliency over the coding process, and σ_t, that handles the degree of nonlinearity of the coding process with the saliency.

The influence of the saliency type adaptor is illustrated in Fig. 4(a). For several values of the parameter λ_t, we show the normalized histograms of the p_k values in the visual dictionary. As can be seen, when λ_t is small, the visual path of the visual words has less influence in the coding process so that the values of p_k tend to the sample average of the descriptors' saliency s_i (histograms with a sharp peak around this sample average saliency value of descriptors in the dataset). In contrast, if λ_t is high, each word in the vocabulary is associated with a particular visual retinal path and, consequently, with a smaller range of saliency values, thus giving place to a more uniform histogram. Note that we present here a process in which the number of visual words devoted to foveal and peripheral paths is automatically learned from data. Furthermore, the assignment of a descriptor to a particular path is performed softly, and depends on the distance between the saliency of the descriptor s_n and the type value p_k of each word in the vocabulary.

In Fig. 4(b), we show additional results of our validation of the λ_t parameter for a heuristically computed optimal σ_t value of $\sigma_t^2 = 0.1$. We can see that the best results are achieved for a $\lambda_t \sim 0.025$, which leads to an approximately uni-modal p_k distribution similar to that one in blue in Fig. 4(a). We would like to note that, as SC is implemented jointly with NUS (with $k = 5$), most of

Table 1: A comparison of various configurations of Saliency-based Object Recognition for the whole (44) and reduced (10) sets of categories in the ADL dataset: Mean AP and p-value of a paired t-test taking NUS +V SR + SC as reference.

Algorithm/ mAP (p-value)	ADL (44 cat)	ADL (10 cat)
BoW	13.6 (0.08)	32.8 (0.02)
BoW + GT Masks	16.8 (0.77)	50.4 (0.47)
NUS [18]	13.9 (0.03)	35.1 (0.00)
NUS+VSR	15.1 (0.14)	38.3 (0.20)
SP-F	14.5 (0.06)	39.4 (0.04)
SP-FP (2000) [13]	13.8 (0.11)	35.6 (0.11)
SP-FP (4000) [13]	14.7 (0.25)	38.2 (0.19)
NUS+VSR+SP-F	14.2 (0.01)	37.5 (0.01)
NUS+VSR+LLC [43, 44]	15.3 (0.11)	42.4 (0.35)
NUS+SC	14.4 (0.11)	38.3 (0.12)
NUS+VSR+SC	**16.2**	**44.0**

the points are computed in salient areas, that leads to a vocabulary in which many more words are devoted to the foveal vision than to the peripheral one. In particular, by manually selecting the point corresponding to the left side of the large bump in the p_k distribution, we have categorized the visual words into two classes: high saliency (p_k is larger than this point), and low saliency (p_k is lower). The resulting proportions are 10% for words corresponding with peripheral vision and 90% for words corresponding with central vision. Hence, we can conclude that SC automatically sets the importance of the two retinal pathways.

5.3. Comparing Saliency Approaches

To assess the influence of each saliency-based stage in the active object recognition problem, we have compared several versions of our approach, namely:

a) *Reference methods*:

1. *BoW*: Baseline BoW with a vocabulary size of 4000 visual words.
2. *BoW + GT Masks*: This approach utilizes (human-annotated) Ground Truth bounding boxes of the active objects, and filters out the descriptors associated with local regions located outside the objects of interest.

b) *Visual Fields with VSR and NUS*:

1. *NUS*: BoW with our NUS module described in sec. 4.1.2, that extends the work in [18].
2. *NUS+VSR*: We add the VSR module (sec. 4.1.1) to the previous approach.

c) *Saliency Pooling Methods*:

1. *SP-F*: BoW + Saliency Pooling considering only the foveal weights as stated in eq. (6).
2. *SP-FP*: BoW + Saliency Pooling considering both the contributions to the foveal and peripheral vision, as stated in eq. (7) and [13]. We have tested two vocabulary sizes: 2000 words (keeping the length of the image signatures constant), or 4000 (doubling the length of the image signatures).

d) *Combined methods*:

1. *NUS+VSR+SP-F*: We add Foveal Saliency Pooling to the NUS+VSR version of our approach to study the combination of both.
2. *NUS+VSR+LLC*: As a reference, and in order to evaluate the effect of the Saliency-based Coding over the system performance, this is a combined approach using the Locally-constrained Linear Coding (LLC) proposed in [43, 44] to compute a vocabulary size of 4000 words.
3. *NUS+VSR+SC*: We substitute the Saliency pooling by our Saliency-sensitive Coding with a vocabulary size of 4000 words.
4. *NUS+SC*: As the previous one, but we switched off the VSR module in order to evaluate its contribution to the system performance.

Results for every method are shown in Table 1. Regarding the methods implementing the VSR+NUS visual fields, we appreciate the positive influence of both elements in the results: although the NUS already enhances the basic BoW, the VSR approach yet provides notable improvements to system performance. This is a consequence of the spatial resolution adaptation to the foveal and peripheral vision, and raises the need for independent and different scale processing paths for the objects of interest (active objects) and their surrounding context.

In parallel, SP also helps to enhance system performance, even if we merely model the foveal vision (SP-F). The approach in [13], which concurrently models foveal and peripheral vision obtains varying results depending on the scenario: if we aim to keep the computational complexity constant and then reduce the vocabulary to half of the size (2000 words), we get a dramatic loss in performance. This result might be expected as we are adding a new path with peripheral vision but at the expense of decreasing the precision of foveal vision. Although we consider that peripheral vision provides useful information, giving the same weight to both pathways has turned to be inappropriate. Furthermore, the fact that keeping the vocabulary size constant (image signatures of double length) improves the performance demonstrates that modeling context is also useful due to the general correlation between objects and locations.

Finally, the combination of various saliency-aware approaches reveals disparate results. Whereas we have observed that the combination of variable resolution visual fields and saliency pooling has not improved performances, saliency coding successfully combines with other saliency-based modules in the processing pipeline. From our point of view, the rationale behind this is that the automatic approach of saliency-sensitive coding correctly handles the relative importance of foveal and peripheral visual pathways, even in the presence of previous blocks in the processing pipeline (like the visual fields described in Sec 4.1). This is something that does not occur with Saliency Pooling which, although by itself provides good performance, when combined with other modules weighs in excess the foveal with respect to the peripheral path and therefore cancels the influence of the context in the recognition process. Even more, by comparing the NUS+VSR+LLC and NUS+VSR+SC alternatives, it is worth noting how the last term in eq. (8) notably contributes to the system performance. Other interesting comparisons, such as the one between NUS+SC and NUS+VSR+SC, offer coherent results with the previous observations about VSR.

From these results, in the following we will assess the best performing approach *NUS+VSR+SC* in other scenarios, and in comparison with various meth-

Table 2: A comparison between our method and some state-of-the-art approaches for various datasets. The p-value of a paired t-test taking 'Ours' as reference is included when available.

	Constrained		Unconstrained	
Dataset	GTEA	Dem@	ADL(44)	ADL(10)
Algorithm/Metrics	Acc	mAP	mAP	mAP
BoW	35.0	45.3 (0.17)	13.6 (0.08)	32.8 (0.02)
BoW + GT Masks	-	54.8 (0.53)	16.8 (0.77)	50.4 (0.47)
DPM [5]	-	34.9 (0.01)	15.3 (0.61)	42.4 (0.66)
DPM [5] + obj. [9]	-	-	13.1 (0.06)	35.9 (0.05)
BoW + SS [10]	-	-	13.4 (0.08)	36.9 (0.19)
Fathi et al. [22]	35.0	-	-	-
Ours	**45.4**	**50.9**	**16.2**	**44.0**

ods that have reported state-of-the-art results in the considered datasets.

5.4. Comparison with the State-of-the-Art

In order to provide a meaningful comparison of our approach with other methods in the state-of-the art, we have divided this section into two blocks: experiments with video, and experiments with still images. The rationale behind is that the sets of reported methods for each kind of data are quite different.

5.4.1. Active object recognition in video

Starting by comparing with the state-of-the-art in egocentric video, in Table 2 we include a comparison between our approach (denoted as 'Ours'), the reference methods, and some techniques that reported State-of-the-art results in each dataset: a) the discriminatively-trained Deformable Part Model (DPM) [5], a sliding window technique that has reported the state-of-the-art results for the ADL dataset [45]; and b) the object recognition approach designed and reported by the authors of the GTEA dataset [22]. In addition, for the ADL dataset, we have also evaluated two methods that combine well-known object recognition approaches and saliency, namely: c) DPM over bounding boxes proposed in [9] (DPM + obj.), which randomly samples bounding boxes and selects the most appropriate based on a measure of their objectness (likelihood to contain an object); and d) BoW applied over candidate bounding boxes proposed by the method described in [10] (BoW + SS), which applies a Selective Search

(SS) to generate potential candidate locations for objects. In both cases, we have followed the same setup described in the original papers [9, 10] to develop the object detector. However, in order to establish a fair comparison with our method, in BoW + SS we have used the same features (dense SURF features) of our proposal.

As we can see from the results, our method consistently outperforms any other automatic approach in every dataset, as well as achieves close performance to the hypothetic case in which ground truth masks/bounding boxes are available (GTEA lacks Ground Truth in all videos so we cannot provide Bow + GT results for this dataset). In particular, in the GTEA dataset we achieve absolute improvements of 10% compared to the best reported approach for this dataset [22]. The performance of DPM is significantly worse than ours under the constrained scenario (Dem@care dataset), whereas it gets closer results under the unconstrained one (ADL dataset). The rationale behind this is that the design of this method is intended to provide good generalizations of the objects' appearance. This property, although desirable under unconstrained scenarios, leads to a loss in performance for the detection of particular object instances (constrained scenario). The very high p-value between DPM and our approach in the unconstrained scenario means that the improvement of our method is not consistent through all the categories, and that DPM is the best choice for some of them (see Fig. 5).

Furthermore, the two state-of-the-art approaches using saliency show lower performance as they strongly restrict the set of locations and scales to be evaluated by the detectors. In particular, the restricted set of candidate windows in DPM+obj causes non-detections with respect to the full DPM approach, whereas for the BoW+SS we have found that, although learning object appearance from accurate ground truth bounding boxes may provide additional information such as accurate object localization, it is very sensitive to the quality of the automatically proposed boxes in test images.

In Fig. 5 we also include per-category results in the ADL dataset. Together with the visual examples in Fig. 6, they yield additional conclusions. In general,

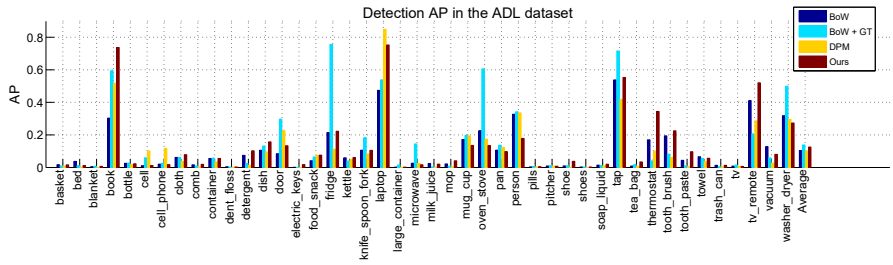


Figure 5: Detailed per-category results of various approaches in ADL dataset

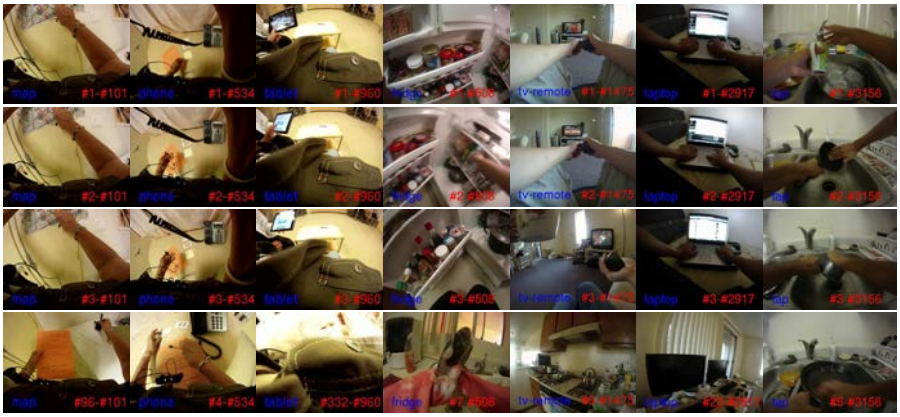


Figure 6: Some visual examples of the ranking provided by our system. Each column represents an object category (columns 1-3 Dem@care dataset, columns 4-7 ADL dataset). For each sample we show the top three ranked results and the first non-relevant ranked image, including the #Ranking Position - #Number of relevant images.

the results vary significantly from one class to another. The poor results in some categories can be explained as follows: although the number of image samples of a class may be high enough (hundreds of thousands), they correspond to a small set of different object instances (no more than 10-15 different instances per category). Hence, if a category shows high intra-class variation (e.g. bed, blanket, container or cloth), it is not possible to obtain good generalizations with such limited training sets. Although using external databases might seem appealing, the work in [45] showed how the application of detectors trained in ImageNet [48] yielded poor results for this particular dataset.

For categories in which the BoW + GT achieves notably better results than

the automatic approaches, we have observed that some of the errors are found in images in which the object is present but not considered as active. This particularly holds for static objects which, although can be manipulated by humans, are rarely moved (e.g. phone and tap belong to the context in the last row in Fig. 6). In those cases only the GT bounding boxes guide the recognition process exactly to the active object, which is hardly achieved by any automatic saliency method. We have studied the effect of this kind of error by removing from the evaluation those frames where an object is present but not active and concluded that it equally affects all compared algorithms. Conversely, if the object of interest is very small (e.g. tv_remote in Fig. 6, thermostat, tooth_brush, etc.), the GT boxes do not contain enough discriminative information whereas our automatic saliency-based approach considers both the salient area and its context, therefore enhancing the detection of the object. Furthermore, we have also observed that if two objects are jointly used in a task and tend to appear in the active areas of scenes (e.g. dishes and tap, detergent and washer, etc.), our method may find more than one active object in a scene. In fact, the proposed saliency methods may locate multiple unconnected salient areas, each of them showing an active object (e.g. each hand manipulates a different object).

5.4.2. Active object recognition and fine-grained activity recognition in images

With PPMI we aim to address the problem of fine-grained activity recognition with the set of 24 categories (12 instruments, considered the active objects, and 2 kind of manipulations). As already mentioned, for this dataset we have followed a slightly different experimental setup in order to provide a fair comparison with the other approaches reported in the literature. Following the setup described in the original paper [46], we have used a multi-scale grid with SIFT features [35] and incorporated a Spatial Pyramid Matching (SPM) with 4 levels and a vocabulary size of 1024 visual words. Due to the multi-scale grid, the VSR has been removed from our solution (although we still kept the non-uniform sampling). Furthermore, since we are working with still images, our spatio-temporal-geometric saliency has been substituted by the spatial method

Table 3: Results (mAP) on PPMI dataset.

BoW	SPM [49]	Grouplet [46]	R.F. [50]	D.S. [14]	Ours
22.7	45.3	36.7	47.0	49.4	**49.7**

Table 4: Comparison of S.T. and M.T. execution times.

Case	DPM [5]	Proposal w/o SC	Proposal w SC
S.T.	60.4s	6.7s	15.1s
M.T.	10.9s	2.6s	6.0s

in [11]. As reported in Table 3, our approach achieves state-of-the-art results in this dataset. Although not very significantly, our method even outperforms [14], that learns discriminative spatial saliency maps associated with each category in the dataset. From our point of view, our results demonstrate that the application of our approach is not restricted to egocentric content and can therefore be applied to the detection of active objects in any kind of scene.

Finally, we would also like to show that the limitations of our method arise when all the objects in a scene have to be recognized (and not only the active ones). This is the case of the Pascal VOC 2010 dataset, for which our method gives an $AP = 56.9$, approximately ranking in the average of the official submissions.

5.5. A study of the computation time

In Table 4, we show a comparison between the average execution times of our proposal and the DPM to run one category object-detector in a test frame. We include results using a single threading (S.T.) and multi-threading (M.T.) in a 2.10GHz computer with 4 cores and hyper-threading. For our proposal, the execution time comprises the whole processing pipeline shown in Fig. 2(a). It is worth noting that some of the computations for the spatial saliency map are implemented in GPU so they cannot be translated to S.T. case (spatial saliency takes about 0.05 sec per frame in the GPU). The rest of the calculations are made with the CPU under the aforementioned circumstances. For the DPM, we run the implementation in [5], made in Matlab with optimized c routines

for all the steps in the process that require most of the execution time. Our approach shows much lower computational times in comparison with DPM. The rationale behind this is the fact that using the saliency maps, we avoid the heavy scanning process of a sliding window approach such as the DPM. Furthermore, Saliency Coding becomes an important source of overhead in the execution time but, as we have shown in the experimental section, it also achieves a notable enhancement of the performance.

In our experiments, as we kept constant the number of features ($\hat{N} \sim N$ in Sec. 4.1), the computational complexity of BoW is similar to our method without SC (in except for the saliency map calculation). However, if the goal is to decrease the complexity, we could aim to obtain similar performances as BoW, and consequently strongly reduce the number of feature points and the computational complexity.

6. Discussion

The application of saliency to computer vision has been traditionally restricted to a pre-processing stage that filters out non-relevant areas of an image. In this paper, instead, we have proposed perceptual model that incorporates visual attention to the challenging task of active object recognition in video and images. To do so, we have modeled independent foveal and peripheral pathways found in human retina, with particular properties in terms of spatial location, resolution, or sampling. In particular, we have introduced saliency into three particular processing modules of the well-known BoW paradigm: a) Visual Fields with Variable-Resolution and Non Uniform Sampling, b) Saliency-based Pooling, and c) Saliency-sensitive Coding of features.

In order to assess the performance of our approach, we aim to address to task of active object recognition in video and images. After discussing the influence of each module and its parameters, we have shown how our biologically-inspired saliency-based model helps to enhance current system performance. It not only achieves notable improvements with respect to the baseline BoW, but

also provides state-of-the-art results in all the considered egocentric datasets at very competitive computational times. Furthermore, it avoids human efforts devoted to bounding-box level database annotation as in both training and test sets the saliency maps are automatically computed. In addition, experiments over both 1st-person and 3rd-person camera view have demonstrated that our method can be applied to various types of content, as long as they contain active objects. The limitation of our method is revealed in scenarios where all present objects (active and non-active) have to be identified. In this case our saliency maps remove important visual information and restrict the performance of our approach.

In the future, we aim to continue exploring novel ways to introduce perceptual modeling into classical pattern recognition problems in computer vision.

7. Acknowledgments

This research is supported by the EU FP7 PI Dem@Care project #288199.

References

[1] J. Sivic, A. Zisserman, Video Google: A text retrieval approach to object matching in videos, in: Proceedings of the International Conference on Computer Vision, volume 2, pp. 1470–1477.

[2] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: In Workshop on Statistical Learning in Computer Vision, ECCV, pp. 1–22.

[3] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pp. 511–518.

[4] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: International Conference on Computer Vision & Pattern Recognition, volume 2, pp. 886–893.

[5] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (2010) 1627–1645.

[6] C. H. Lampert, M. B. Blaschko, T. Hofmann, Beyond sliding windows: Object localization by efficient subwindow search, in: IEEE Conference on Computer Vision and Pattern Recognition.

[7] A. Borji, L. Itti, State-of-the-art in visual attention modeling, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (2013) 185–207.

[8] X. Ren, C. Gu, Figure-Ground Segmentation Improves Handled Object Recognition in Egocentric Video, in: IEEE Conference on Computer Vision and Pattern Recognition.

[9] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Transactions on Pattern Analysis and Machine Intelligence, 34 (2012) 2189–2202.

[10] J. Uijlings, K. van de Sande, T. Gevers, A. Smeulders, Selective search for object recognition, International Journal of Computer Vision 104 (2013) 154–171.

[11] L. Itti, C. Koch, Computational modelling of visual attention, Nature Reviews Neuroscience 2 (2001) 194–203.

[12] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: Advances in Neural Information Processing Systems 19, MIT Press, Cambridge, MA, 2007, pp. 545–552.

[13] R. de Carvalho Soares, I. da Silva, D. Guliato, Spatial locality weighting of features using saliency map with a bovw approach, in: International Conference on Tools with Artificial Intelligence, 2012, pp. 1070–1075.

[14] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3506–3513.

[15] M. San Biagio, L. Bazzani, M. Cristani, V. Murino, Weighted bag of visual words for object recognition, in: IEEE International Conference on Image Processing (ICIP), 2014, pp. 2734–2738.

[16] V. Mahadevan, N. Vasconcelos, Biologically inspired object tracking using center-surround saliency mechanisms, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (2013) 541–554.

[17] Y. Su, Q. Zhao, L. Zhao, D. Gu, Abrupt motion tracking using a visual saliency embedded particle filter, Pattern Recognition 47 (2014) 1826 – 1834.

[18] E. Vig, M. Dorr, D. Cox, Space-Variant Descriptor Sampling for Action Recognition Based on Saliency and Eye Movements, Springer, Firenze, Italy, pp. 84–97.

[19] S. Mathe, C. Sminchisescu, Dynamic eye movement datasets and learnt saliency models for visual action recognition, in: European Conference on Computer Vision (ECCV), 2012, pp. 842–856.

[20] I. González-Díaz, V. Buso, J. Benois-Pineau, G. Bourmaud, R. Megret, Modeling instrumental activities of daily living in egocentric vision as sequences of active objects and context for alzheimer disease research, in: ACM MM MIIRH Workshop.

[21] S. Karaman, J. Benois-Pineau, V. Dovgalecs, R. Mégret, J. Pinquier, R. André-Obrecht, Y. Gaëstel, J.-F. Dartigues, Hierarchical hidden markov model in detecting activities of daily living in wearable videos for studies of dementia, Multimedia Tools and Applications (2011) 1–29.

[22] A. Fathi, X. Ren, J. M. Rehg, Learning to recognize objects in egocentric activities, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 3281–3288.

[23] A. Fathi, Y. Li, J. M. Rehg, Learning to recognize daily actions using gaze, in: European Conference on Computer Vision, ECCV'12, pp. 314–327.

[24] K. Ogaki, K. M. Kitani, Y. Sugano, Y. Sato, Coupling eye-motion and ego-motion features for first-person activity recognition., in: IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2012, pp. 1–7.

[25] H. L. Fernandes, I. H. Stevenson, A. N. Phillips, M. A. Segraves, K. P. Kording, Saliency and saccade encoding in the frontal eye field during natural scene search, Cerebral Cortex (2013).

[26] D. Wooding, Eye movements of large populations: Ii. deriving regions of interest, coverage, and similarity using fixation maps, Behavior Research Methods, Instruments, & Computers 34 (2002) 518–528.

[27] D. Walther, U. Rutishauser, C. Koch, P. Perona, On the usefulness of attention for object recognition, in: Workshop on Attention and Performance in Computational Vision at ECCV, pp. 96–103.

[28] F. Moosmann, D. Larlus, F. Jurie, Learning saliency maps for object categorization, in: ECCV'06 Workshop on the Representation and Use of Prior Knowledge in Vision.

[29] H. Larochelle, G. E. Hinton, Learning to combine foveal glimpses with a third-order boltzmann machine., in: Advances in Neural Information Processing Systems 23, pp. 1243–1251.

[30] H. Boujut, J. Benois-Pineau, R. Megret, Fusion of multiple visual cues for visual saliency extraction from wearable camera settings with strong motion, in: European Conference on Computer Vision. Workshops, 2012.

[31] O. Brouard, V. Ricordel, D. Barba, Cartes de Saillance Spatio-Temporelle basées Contrastes de Couleur et Mouvement Relatif, in: Compression et representation des signaux audiovisuels.

[32] C. Chamaret, J.-C. Chevet, O. Le Meur, Spatio-temporal combination of saliency maps and eye-tracking assessment of different strategies, in: IEEE International Conference on Image Processing (ICIP), 2010, pp. 1077–1080.

[33] D. Ramirez-Moreno, O. Schwartz, J. Ramirez-Villegas, A saliency-based bottom-up visual attention model for dynamic scenes analysis, Biological Cybernetics 107 (2013) 141–160.

[34] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (surf), Computer Vision and Image Understanding 110 (2008) 346–359.

[35] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2004) 91–110.

[36] V. Sreekanth, A. Vedaldi, C. Jawahar, A. Zisserman, Generalized RBF feature maps for efficient detection, in: British Machine Vision Conference 2010.

[37] B. A. Wandell, Foundations of Vision, Sinauer Associates, Inc., 1995.

[38] S. Liversedge, I. Gilchrist, S. Everling, The Oxford Handbook of Eye Movements, Chapter 33, Oxford Library of Psychology, OUP Oxford, 2011.

[39] E.-C. Chang, S. Mallat, C. Yap, Wavelet foveation, Applied and Computational Harmonic Analysis 9 (2000) 312 – 335.

[40] J. S. Perry, W. S. Geisler, Gaze-contingent real-time simulation of arbitrary visual fields, in: In Human Vision and Electronic Imaging, SPIE Proceedings, pp. 57–69.

[41] M. Marszałek, C. Schmid, Spatial weighting for bag-of-features, in: IEEE Conference on Computer Vision & Pattern Recognition, volume 2, pp. 2118–2125.

[42] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009., pp. 1794–1801.

[43] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained Linear Coding for Image Classification, in: IEEE Conference on Computer Vision and Pattern Recognition.

[44] C.-P. Wei, Y.-W. Chao, Y.-R. Yeh, Y.-C. F. Wang, Locality-sensitive dictionary learning for sparse representation based classification, Pattern Recognition 46 (2013) 1277–1287.

[45] H. Pirsiavash, D. Ramanan, Detecting activities of daily living in first-person camera views, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[46] B. Yao, L. Fei-Fei, Grouplet: A structured image representation for recognizing human and object interactions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, USA.

[47] M. Everingham, L. Gool, C. K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, International Journal of Computer Vision 88 (2010) 303–338.

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[49] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: IEEE Conference on Computer Vision and Pattern Recognition - 2006, pp. 2169–2178.

[50] B. Yao, A. Khosla, L. Fei-Fei, Combining randomization and discrimination for fine-grained image categorization, in: IEEE Conference on Computer Vision and Pattern Recognition. 2011.