Top-Down Visual Saliency via Joint CRF and Dictionary Learning

Jimei Yang and Ming-Hsuan Yang
University of California at Merced
{jyang44,mhyang}@ucmerced.edu

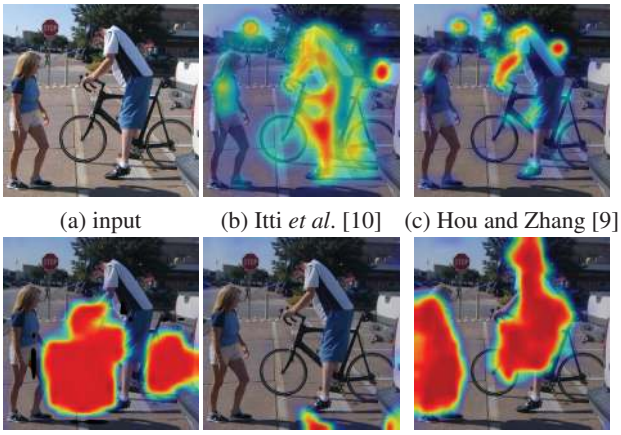
Abstract

Top-down visual saliency facilities object localization by providing a discriminative representation of target objects and a probability map for reducing the search space. In this paper, we propose a novel top-down saliency model that jointly learns a Conditional Random Field (CRF) and a discriminative dictionary. The proposed model is formulated based on a CRF with latent variables. By using sparse codes as latent variables, we train the dictionary modulated by CRF, and meanwhile a CRF with sparse coding. We propose a max-margin approach to train our model via fast inference algorithms. We evaluate our model on the Graz-02 and PASCAL VOC 2007 datasets. Experimental results show that our model performs favorably against the state-of-the-art top-down saliency methods. We also observe that the dictionary update significantly improves the model performance.

1. Introduction

Bottom-up visual saliency models the unconscious visual processing in early vision and is mainly driven by low-level cues (e.g., oriented filter responses and color). In the last two decades, some basic principles, such as center-surround contrast [10], self-information [3], topological connectivity [8] and spectral residual [9], have been established for computing bottom-up saliency maps, which are shown to be effective for predicting human eye movements [3, 8] and for highlighting the informative regions of images [10, 9].

However, the data-driven nature of bottom-up saliency limits its applications in target-oriented computer vision tasks, such as object localization, detection and segmentation. In some cases when backgrounds are highly cluttered, due to lack of top-down prior knowledge, bottom-up saliency algorithms usually respond to numerous unrelated low-level visual stimuli (i.e., false positives) and thus miss the objects of interest (i.e., false negatives). In Figure 1, for instance, two typical bottom-up saliency maps ((b) and (c)) highlight a stop sign as interesting regions and do not distinguish the bicycle from two persons. In contrast, top-down saliency models learn from training examples to generate probability maps for localizing objects of interest, which are bicycle (d), car (e) and person (f), respectively.



(a) input (b) Itti *et al*. [10] (c) Hou and Zhang [9]

(d) bicycle (e) car (f) person

Figure 1. Bottom-up and top-down saliency. Given an input image (a), we present two bottom-up saliency maps that are produced by [10] in (b) and [9] in (c). In the bottom panel, we present three top-down saliency maps for bicycle (d), car (e) and person (f) generated by our algorithm (best viewed in color).

Classic visual recognition problems entail detection (position and scale) and identification of objects (on the instance or category level). The difficulties of visual recognition mainly result from exploring large search space (over position and scale) and modeling high variability of object appearance (due to the changes of pose and illumination as well as occlusions).

Recent progress on Bag-of-Words (BoW) models [27, 5, 2] reveals the effectiveness of patch-based representation. On the patch level, we represent the object appearance by a dictionary of visual words and sample the image patches to reduce the complexity of searching over parametric space. The performance of BoW models highly depends on the dictionary [20] and sampling strategy [21]. We propose a novel top-down saliency model that facilitates visual recognition from those two perspectives. The central idea of our top-down saliency model is to build a conditional random

field (CRF) upon sparse coding of image patches with a joint learning approach. For any image patch, we use a binary variable to label the presence or absence of target objects. The use of conditional random field enables us to exploit the connectivity of adjacent image patches so that we can compute the saliency maps by incorporating local context information. On the other hand, the use of sparse coding facilities us to model feature selectivity for saliency map, which typically results in a more compact and discriminative representation.

We note that the proposed model is more than a straightforward combination of CRF and sparse coding. Instead, we formulate a novel CRF with sparse latent variables. By using sparse codes as latent variables, we learn a discriminative dictionary modulated by CRF, and meanwhile a CRF driven by sparse coding. We propose a max-margin approach to train the model by exploiting fast inference algorithms, such as graph cut [13]. We empirically evaluate our model on the Graz-02 [22] and PASCAL VOC 2007 [4] datasets and measure the quality of saliency maps by patch-level precision-recall rates. The experimental results show that our model performs favorably against several state-of-the-art top-down saliency algorithms [6, 12]. We also show that the dictionary update component of our algorithm significantly improves the performance of our model.

2. Related Work

We first discuss the related algorithms on top-down saliency maps and then briefly describe CRF and dictionary learning methods that are related to the proposed joint learning algorithm.

2.1. Top-Down Saliency Maps

Top-down visual saliency involves the processes of feature learning and saliency computation [6]. Gao *et al*. [6] propose a top-down saliency algorithm by selecting discriminant features from a pre-defined filter bank. Their discriminant features are characterized by statistical difference of target presence or absence in the training images. With the selected features, the saliency values of interest points can be computed based on mutual information. Instead of using pre-defined filter bank, Kanan *et al*. [12] propose to learn features with independent component analysis (ICA) from natural images, and construct a top-down saliency model by training a support vector machine (SVM). In our model, the target object features are learned from training images by CRF-modulated dictionary learning. In [12], the top-down saliency map is evaluated by both the appearance component (probabilistic output of SVM) and contextual prior of target location. This location prior performs well when there is a strong correlation between the target locations and holistic scenes, such as cars in urban scenes, but becomes less effective when target objects appear randomly anywhere in general cases (e.g., images from the Graz-02 and PASCAL VOC2007 datasets). In contrast, we compute the saliency map by inference on CRF, which is more flexible to leverage local context information.

2.2. CRFs

CRFs have been demonstrated as a flexible framework of incorporating different kinds of features for visual recognition [25, 24, 5, 7, 1]. In particular, CRFs are used 1) to learn an optimal combination of low-level cues (color and edge) and pre-learned high-level modules (e.g., part-based detector, Bag-of-Words classifiers), and 2) to accommodate inference functions (e.g., graph cut and belief propagation) for graphical models of specific visual recognition problems. In this sense, CRFs are used to integrate different cues [24] or refine labeling results [5]. In our model, the CRF parameters include the node classifier built on sparse coding so that the number of CRF parameters is not several combination coefficients but hundreds or thousands of classifier coefficients up to the number of bases in sparse coding. A similar idea has been explored in the Discriminative Random Field model [14] which learns node and edge logistic classifiers simultaneously. We note that it is rather challenging to learn a large set of parameters from limited training samples. Instead of using the pseudo-likelihood method [14], we take a discriminative training approach by converting the likelihood maximization into an inequality constrained optimization problem [11, 25]. Aside from the node classifier, our model also involves learning a dictionary which is essential for representing object appearance on the patch level. Therefore, our saliency formulation can be considered as a latent variable model by training a CRF classifier jointly with dictionary learning. Although our model bears some resemblance to the hidden CRFs [23, 26] developed for object and action recognition, they are intrinsically different. The hidden CRFs use a vector of latent variables to represent unobserved part labels of local patches in an observed image whereas our model uses latent variables to model the sparse representations of local observations with the dictionary. In addition, the hidden CRFs predict one category label of the input image while our model produces a saliency map of predicting the presence of target objects.

2.3. Dictionary Learning

Recent advances in machine learning enable us to train task-specific dictionaries in a supervised manner [18, 28, 17]. Mairal *et al*. [18] combine sparse coding and classification loss in a single optimization objective. Although this method shows promising results on digit recognition and texture classification, it is not clear how it performs on complex object images as no mechanism for integrating local evidences. Yang *et al*. [28] propose to learn translation-invariant dictionary for image classification via

back-projection techniques. Their method perform well for face and digit recognition because of the translation invariance property obtained from max pooling. Our model is able to learn dictionaries from complex object images (e.g., bicycles, cars, persons) in cluttered backgrounds. Unlike [28] that uses max pooling to resolve geometric ambiguity, we use CRF to regulate the patches within their local contexts for learning salient visual words in complex scenes.

3. Problem Formulation

Given an image, we are interested in knowing whether and where the target objects appear. For a local image patch $\mathbf{x} \in \mathbb{R}^p$, we assign a binary label $\mathbf{y}$ to indicate the presence ($\mathbf{y} = 1$) or absence ($\mathbf{y} = -1$) of a target object. We sample a set of patches $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ from different locations of the image as the observations. The corresponding labels $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ...\mathbf{y}_m\}$ carry the information of target presence. In a particular scale, a sampled patch $\mathbf{x}_i$ usually carries partial information about the target object. It is thus challenging to directly infer the presence of the target from $\mathbf{x}_i$ without considering the others due to the semantic and geometric ambiguities of patch-level representations.

Suppose that there exists a dictionary $\mathbf{D} \in \mathbb{R}^{p \times k}$ that stores the most representative object parts (visual words) $\{\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_k\}$ learned from the training data. We introduce a vector of latent variables $\mathbf{s}_i \in \mathbb{R}^k$ to model the sparse representation of $\mathbf{x}_i = \mathbf{D}\mathbf{s}_i$, which is usually obtained by optimizing the following problem,

$$\mathbf{s}(\mathbf{x}, \mathbf{D}) = \arg\min_s \frac{1}{2}\|\mathbf{x} - \mathbf{D}s\|^2 + \lambda\|s\|_1, \qquad (1)$$

where λ is a parameter controlling the sparse penalty. We denote the latent variables for all the patches by $\mathbf{S}(\mathbf{X}, \mathbf{D}) = [\mathbf{s}(\mathbf{x}_1, \mathbf{D}), \mathbf{s}(\mathbf{x}_2, \mathbf{D}), ..., \mathbf{s}(\mathbf{x}_m, \mathbf{D})]$. Note that we use the notations $\mathbf{s}(\mathbf{x}, \mathbf{D})$ and $\mathbf{S}(\mathbf{x}, \mathbf{D})$ to emphasize that the sparse latent variables are a function of the dictionary. In the following sections, we simplify the notations by $\mathbf{s}_i \triangleq \mathbf{s}(\mathbf{x}, \mathbf{D})$ and $\mathbf{S} \triangleq \mathbf{S}(\mathbf{x}, \mathbf{D})$ for presentation clarity when necessary. The sparse coding formulation in Eqn. 1 can be solved efficiently [15]. Through our sparse coding formulation, the visual information contained in the dictionary is transferred into the latent variables by $\mathbf{S}(\mathbf{X}, \mathbf{D})$ which is thus more informative than image patches $\mathbf{X}$.

If a local patch shows evidence about target objects, it is likely that nearby patches also exhibit similar support. We build a four-connected graph $\mathcal{G} = < \mathcal{V}, \mathcal{E} >$ on the sampled patches based on their spatial adjacency, where $\mathcal{V}$ denote the nodes and $\mathcal{E}$ the edges. Assuming that the labels $\mathbf{Y}$ enjoy the Markov property on the graph $\mathcal{G}$ conditioned on the sparse latent variables $\mathbf{S}(\mathbf{X}, \mathbf{D})$, we formulate a novel CRF model by

$$P(\mathbf{Y}|\mathbf{S}(\mathbf{X}, \mathbf{D}), \mathbf{w}) = \frac{1}{Z} e^{-E(\mathbf{S}(\mathbf{X}, \mathbf{D}), \mathbf{Y}, \mathbf{w})}, \qquad (2)$$

where Z is the partition function, $E(\mathbf{S}(\mathbf{X}, \mathbf{D}), \mathbf{Y}, \mathbf{w})$ is the energy function and $\mathbf{w}$ is the CRF weight vector. This formulation enables us to jointly learn the CRF weight $\mathbf{w}$ and the dictionary $\mathbf{D}$. Given the CRF weight $\mathbf{w}$, the model in Eqn. 2 can be viewed as CRF supervised dictionary learning, whereas given the dictionary $\mathbf{D}$, it can be viewed as CRF learning with sparse coding. In this model, we can easily retrieve the target information at a particular node $i \in \mathcal{V}$ from its marginal probability

$$p(\mathbf{y}_i|\mathbf{s}_i, \mathbf{w}) = \sum_{\mathbf{y}_{\mathcal{N}(i)}} p(\mathbf{y}_i, \mathbf{y}_{\mathcal{N}(i)}|\mathbf{s}_i, \mathbf{w}), \qquad (3)$$

where $\mathcal{N}(i)$ denotes the neighbors of node i on the graph $\mathcal{G}$. We define the saliency value of the patch i as

$$\mathbf{u}(\mathbf{s}_i, \mathbf{w}) = p(\mathbf{y}_i = 1|\mathbf{s}_i, \mathbf{w}), \qquad (4)$$

and thus the saliency map $\mathbf{U}(\mathbf{S}, \mathbf{w}) = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m\}$ can be inferred by message passing algorithms. This probabilistic definition of top-down saliency map leverages not only the appearance information [6, 12], but also the local contextual information through the marginalization in Eqn. 3.

We decompose the energy function $E(\mathbf{S}(\mathbf{X}, \mathbf{D}), \mathbf{Y}, \mathbf{w})$ into node and pairwise energy terms. For each node $i \in \mathcal{V}$, the energy is measured by the total contribution of sparse codes $\psi(\mathbf{s}_i, \mathbf{y}_i, \mathbf{w}_1) = -\mathbf{y}_i \mathbf{w}_1^\top \mathbf{s}_i$, where $\mathbf{w}_1 \in \mathbb{R}^k$ is the weight vector. For each edge $(i, j) \in \mathcal{E}$, we only consider data-independent smoothness $\psi(\mathbf{y}_i, \mathbf{y}_j, \mathbf{w}_2) = \mathbf{w}_2 \mathbb{I}(\mathbf{y}_i, \mathbf{y}_j)$, where the scaler $\mathbf{w}_2$ measures the weight of labeling smoothness and $\mathbb{I}$ is an indicator function equaling one for different labels. Therefore, the random field energy can be detailed as

$$E(\mathbf{S}, \mathbf{Y}, \mathbf{w}, \mathbf{D}) = \sum_{i \in \mathcal{V}} \psi(\mathbf{s}_i, \mathbf{y}_i, \mathbf{w}_1) + \sum_{(i,j) \in \mathcal{E}} \psi(\mathbf{y}_i, \mathbf{y}_j, \mathbf{w}_2). \qquad (5)$$

Note that our energy function is linear with the parameter $\mathbf{w} = [\mathbf{w}_1; \mathbf{w}_2]$ which is similar to most CRF models [24, 25, 1], but is nonlinear with the dictionary $\mathbf{D}$ that is implicitly defined by $\mathbf{s}(\mathbf{x}, \mathbf{D})$ in Eqn. 1. This nonlinear parametrization makes it challenging to learn the model. We discuss our learning approach in the next section.

Let us now assume that we have learned the optimal CRF parameters $\hat{\mathbf{w}}$ and the dictionary $\hat{\mathbf{D}}$. Our top-down saliency formulation in Eqn. 2 does not involve complex evaluations of latent variables [23, 18], and makes it feasible to infer the saliency map in a straight-forward manner without alternating between evaluation of latent variables and label inference. For a test image $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$, we compute its saliency map $\mathbf{U}$ as follows:

1. evaluate the sparse latent variables $\mathbf{S}(\mathbf{X}, \hat{\mathbf{D}})$ by Eqn. 1;

2. infer the saliency map $\mathbf{U}(\mathbf{S}, \hat{\mathbf{w}})$ by Eqn. 3 and Eqn. 4.

4. Joint CRF and Dictionary Learning

Let $\mathcal{X} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, ...\mathbf{X}^{(N)}\}$ be a collection of training images and $\mathcal{Y} = \{\mathbf{Y}^{(1)}, \mathbf{Y}^{(2)}, ...\mathbf{Y}^{(N)}\}$ be the corresponding labels. We aim to learn the CRF parameters $\hat{\mathbf{w}}$ and the dictionary $\hat{\mathbf{D}}$ to maximize the joint likelihood of training samples,

$$\max_{\mathbf{w}\in\mathbb{R}^{(k+1)}, \mathbf{D}\in\mathcal{D}, \mathbf{S}^{(n)}} \prod_{n=1}^{N} P(\mathbf{Y}^{(n)}|\mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D}), \mathbf{w}), \quad (6)$$

where $\mathbf{S}^{(n)}$ is a shorthand of $\mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D})$ and $\mathcal{D}$ is the convex set of dictionaries that satisfies the following constraint:

$$\mathcal{D} = \{\mathbf{D} \in \mathbb{R}^{p\times k}, \|\mathbf{d}_j\|_2 \leq 1, \forall j = 1, 2, ..., k\}. \quad (7)$$

4.1. Max-Margin Approach

The difficulties in CRF learning mainly lie in evaluating the partition function Z of Eqn. 2. Inspired by the max-margin CRF learning approaches [25, 1], we pursue the optimal $\mathbf{w}$ and $\mathbf{D}$ so that for all $\mathbf{Y} \neq \mathbf{Y}^{(n)}, n = 1, ..., N$

$$P(\mathbf{Y}^{(n)}|\mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D}), \mathbf{w}) \geq P(\mathbf{Y}|\mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D}), \mathbf{w}). \quad (8)$$

This constrained optimization allows us to cancel the partition function Z from both sides of the constraints and express them in terms of energies

$$E(\mathbf{Y}^{(n)}, \mathbf{S}^{(n)}, \mathbf{w}) \leq E(\mathbf{Y}, \mathbf{S}^{(n)}, \mathbf{w}). \quad (9)$$

Furthermore, we expect the ground truth energy $E(\mathbf{Y}^{(n)}, \mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D}), \mathbf{w})$ is less than any other energies $E(\mathbf{Y}, \mathbf{S}(\mathbf{X}^{(n)}, \mathbf{D}), \mathbf{w})$ by a large margin $\Delta(\mathbf{Y}, \mathbf{Y}^{(n)})$. We thus have a new constraint set

$$E(\mathbf{Y}^{(n)}, \mathbf{S}^{(n)}, \mathbf{w}) \leq E(\mathbf{Y}, \mathbf{S}^{(n)}, \mathbf{w}) - \Delta(\mathbf{Y}, \mathbf{Y}^{(n)}). \quad (10)$$

In this paper, we define the margin function $\Delta(\mathbf{Y}, \mathbf{Y}^{(n)}) = \sum_{i=1}^{m} \mathbb{I}(\mathbf{y}_i, \mathbf{y}_i^{(n)})$. There are exponentially large number of constraints with respect to labeling $\mathbf{Y}^{(n)}$ for each training sample. Similar with the cutting plane algorithm [11], we seek for the most violated constraints by solving

$$\hat{\mathbf{Y}}^{(n)} = \arg\min_{\mathbf{Y}} E(\mathbf{Y}, \mathbf{S}^{(n)}, \mathbf{w}) - \Delta(\mathbf{Y}, \mathbf{Y}^{(n)}). \quad (11)$$

Therefore, we are able to learn the weight $\mathbf{w}$ and the dictionary $\mathbf{D}$ by minimizing the following objective function,

$$\min_{\mathbf{w}, \mathbf{D}\in\mathcal{D}} \frac{\gamma}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} \ell^n(\mathbf{w}, \mathbf{D}), \quad (12)$$

where $\ell^n(\mathbf{w}, \mathbf{D}) \triangleq E(\hat{\mathbf{Y}}^{(n)}, \mathbf{S}^{(n)}, \mathbf{w}) - E(\mathbf{Y}^{(n)}, \mathbf{S}^{(n)}, \mathbf{w})$ and γ controls the regularization of $\mathbf{w}$.

We note that our approach shares a similar objective function with the latent structural SVM [29]. The difference is that the latent structural SVM is linearly parameterized while ours is nonlinear with the dictionary $\mathbf{D}$.

4.2. Learning Algorithm

We propose a stochastic gradient descent algorithm for optimizing the objective function in Eqn. 12. The basic idea is simple and easy to implement. At the t^{th} iteration , we randomly select a training instance $(\mathbf{X}^{(n)}, \mathbf{Y}^{(n)})$, and then

1. evaluate the sparse latent variables with the dictionary $\mathbf{D}^{(t-1)}$ by Eqn. 1,
2. obtain the most violated labeling with the weight $\mathbf{w}^{(t-1)}$ by Eqn. 11,
3. update the weight $\mathbf{w}^{(t)}$ and the dictionary $\mathbf{D}^{(t)}$ by the gradients of the loss function ℓ^n.

We next describe the methods of computing the gradients with respect to the weight and the dictionary.

When the latent variables $\mathbf{S}$ are known, the energy function $E(\mathbf{Y}, \mathbf{S}, \mathbf{w})$ is linear with $\mathbf{w}$ (Eqn. 5),

$$E(\mathbf{Y}, \mathbf{S}, \mathbf{w}) =< \mathbf{w}, f(\mathbf{S}, \mathbf{Y}) >, \quad (13)$$

where $f(\mathbf{S}, \mathbf{Y}) = [-\sum_{i\in\mathcal{V}} \mathbf{s}_i\mathbf{y}_i; \sum_{(i,j)\in\mathcal{E}} \mathbb{I}(\mathbf{y}_i, \mathbf{y}_j)]$. We can thus compute the gradient with respect to $\mathbf{w}$,

$$\frac{\partial \ell^n}{\partial \mathbf{w}} = f(\mathbf{S}^{(n)}, \hat{\mathbf{Y}}^{(n)}) - f(\mathbf{S}^{(n)}, \mathbf{Y}^{(n)}) + \gamma\mathbf{w}. \quad (14)$$

The dictionary is not explicitly defined in the energy function 12 but implicitly by sparse coding (Eqn. 1). We use the chain rule of differentiation to compute the gradient of ℓ^n with respect to the dictionary,

$$\frac{\partial \ell^n}{\partial \mathbf{D}} = \sum_{i\in\mathcal{V}} (\frac{\partial \ell^n}{\partial \mathbf{s}_i})^\top \frac{\partial \mathbf{s}_i}{\partial \mathbf{D}}, \quad (15)$$

The difficulty of computing this gradient lies in that there is no explicit differentiation of sparse code $\mathbf{s}$ with respect to the dictionary $\mathbf{D}$. We overcome this difficulty by using implicit differentiation on the fixed point equation, in a way similar with [28] and [17]. We first establish the fixed point equation of Eqn. 1,

$$\mathbf{D}^\top(\mathbf{D}\mathbf{s} - \mathbf{x}) = -\lambda \text{sign}(\mathbf{s}), \quad (16)$$

where sign($\mathbf{s}$) denotes the sign of $\mathbf{s}$ in a point-wise manner and sign(0) = 0. We calculate the derivative of $\mathbf{D}$ on both sides of Eqn. 16, and have

$$\frac{\partial \mathbf{s}_\Lambda}{\partial \mathbf{D}} = (\mathbf{D}_\Lambda^\top \mathbf{D}_\Lambda)^{-1}(\frac{\partial \mathbf{D}_\Lambda^\top \mathbf{x}}{\partial \mathbf{D}} - \frac{\partial \mathbf{D}_\Lambda^\top \mathbf{D}_\Lambda}{\partial \mathbf{D}}), \quad (17)$$

where we denote Λ as the index set of non-zero codes of $\mathbf{s}$ and $\bar{\Lambda}$ as the index set of zero codes. To simplify the gradient computation in Eqn. 15, we introduce a vector of auxiliary variables $\mathbf{z}$ for each $\mathbf{s}$,

$$\mathbf{z}_{\bar{\Lambda}} = 0, \mathbf{z}_\Lambda = (\mathbf{D}_\Lambda^\top \mathbf{D}_\Lambda)^{-1} \frac{\partial \ell^n}{\partial \mathbf{s}_\Lambda}, \quad (18)$$

where $\partial \ell^n / \partial \mathbf{s}_\Lambda = (\mathbf{y}_i - \hat{\mathbf{y}}_i)\mathbf{w}_\Lambda$. In addition, we denote $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_m]$. Therefore, the gradient of ℓ^n with respect to $\mathbf{D}$ is computed by

$$\frac{\partial \ell^n}{\partial \mathbf{D}} = -\mathbf{D}\mathbf{Z}\mathbf{S}^\top + (\mathbf{X} - \mathbf{D}\mathbf{S})\mathbf{Z}^\top. \quad (19)$$

The proposed joint learning algorithm is summarized in Algorithm 1.

Algorithm 1 Joint CRF and dictionary learning.

Input: $\mathcal{X}$ (training images) and $\mathcal{Y}$ (ground truth labels); $\mathbf{D}^{(0)}$ (initial dictionary); $\mathbf{w}^{(0)}$ (initial CRF weight); λ (in Eqn. 1); T (number of cycles); γ (in Eqn. 12) ρ_0 (initial learning rate).

Output: $\hat{\mathbf{D}}$ and $\hat{\mathbf{w}}$.

for $t = 1, \ldots, T$ **do**

  Permute training samples $(\mathcal{X}, \mathcal{Y})$

  for $n = 1, \ldots, N$ **do**

    Evaluate the latent variables $\mathbf{s}_i$ by Eqn. 1, $\forall i \in V$;

    Solve the most violated labeling $\hat{\mathbf{Y}}^{(n)}$ by Eqn. 11;

    Update the weight $\mathbf{w}$ by Eqn. 14:

$$\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} - \rho_t \frac{\partial \ell^n}{\partial \mathbf{w}^{(t-1)}};$$

    Find the active set Λ_i for $\mathbf{s}_i$, $\forall i \in V$;

    Compute the auxiliary variables $\mathbf{z}_i$ by Eqn. 18;

    Update the dictionary $\mathbf{D}$ by Eqn. 19:

$$\mathbf{D}^{(t)} = \mathbf{D}^{(t-1)} + \rho_t \frac{\partial \ell^n}{\partial \mathbf{D}^{(t-1)}};$$

    Project the dictionary $\mathbf{D}^{(t)}$ onto $\mathcal{D}$ by Eqn. 7;

    Update the learning rate ρ: $\rho_t = \rho_0 / n$

  end for

end for

5. Experiments

We evaluate the proposed top-down saliency model on the Graz-02 and PASCAL VOC 2007 datasets. We choose these two datasets because they both contain real-world images with large amount of intra-class variations, occlusions and background clutters. The MATLAB code and experimental results are available from our website (`http://faculty.ucmerced.edu/mhyang/pubs.html`).

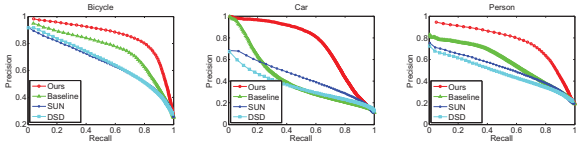
5.1. Graz-02

The Graz-02 dataset contains three categories (bicycles, cars and persons) and a background class. Each category has 300 images of size 640×480 pixels and the corresponding pixel-level foreground/background annotations. The task is to evaluate the performance of top-down saliency maps to localize target objects against the background. We sample image patches of 64×64 pixels by shifting 16 pixels so that we collect 999 patches on a 27×37 grid for each image. We use the same patch sampling method for all the following experiments. The SIFT descriptors [16] are extracted from each image patch to represent the object appearance. We label a patch as positive if at least one quarter of its total pixels are foreground; otherwise we label it as negative. We thus obtain patch-level ground truth from the original pixel-level annotations. For each category, we use the 150 odd-numbered images of its category and additional 150 odd-numbered images from background class as the training set, and the remaining 150 even-numbered images of its category and 150 even-numbered background images as the test set.

To train our saliency model by Algorithm 1, we need to initialize the dictionary and the CRF. We collect all the SIFT descriptors from training set and use the K-means algorithm to initialize the dictionary $\mathbf{D}^{(0)}$. After evaluating the latent variables by sparse coding, we initialize the CRF node energy weight $\mathbf{w}_1^{(0)}$ by training a linear SVM on the sparse codes and the corresponding patch labels. For the pairwise energy weight $\mathbf{w}_2^{(0)}$, we simply set it to 1. All the models are trained with 20 cycles.

There are two important parameters in our model. One is the number of visual words (atoms) k in the dictionary, which controls the capacity of modeling the appearance variations. Usually, a dictionary of larger size will produce better results but is more difficult to learn as it requires more training examples with a higher computational cost. In our experiments, we train the models with 256 or 512 visual words. The other parameter λ controls the sparse penalty defined in Eqn. 1. The greater the λ is, the more sparse the latent variables are and less visual words are selected to represent an image patch. We use two values, 0.15 and 0.30, for λ in the experiments. In Algorithm 1, we set the initial learning rate $\rho_0 = 1e{-}3$ and the weight penalty $\gamma = 1e{-}5$. To demonstrate the effectiveness of joint CRF and dictionary learning, we build a baseline model by directly combining sparse coding and CRF, which means learning CRF weight by using sparse codes computed from the initial dictionary as features. We also compare our model with two state-of-the-art top-down saliency algorithms [6, 12] by using our own implementations. For the discriminant saliency detection algorithm [6] (DSD), we first construct a DCT (Discrete Cosine Transform) dictionary with 256 filters of size 64×64, and then select 100 salient features with largest mutual information. More details can be found in [6]. For the saliency using natural statistics algorithm [12] (SUN), we first reduce the dimension of the image patches by Principle Component Analysis (PCA) and then learn 724 ICA filters from the training data. By using the ICA filter responses as features, a linear SVM is trained to compute the saliency values of patches.

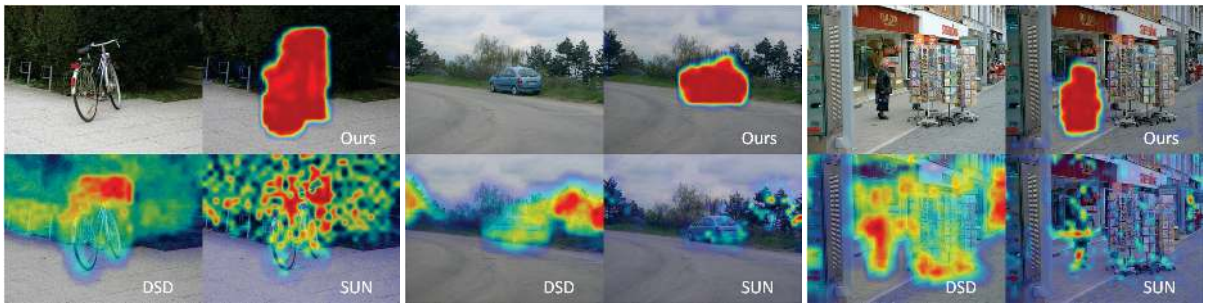
All those models (ours, baseline, DSD, SUN) are evaluated by patch-level precision-recall rates on the test set of each category. Figure 2 shows the precision-recall curves for three object categories, respectively.



(a) bicycle (b) car (c) person

Figure 2. Patch-level precision-recall curves on Graz-02 dataset.

In Table 1, we compare our results for different parameters (k, λ) with other models by precision rates at equal error rates (EER where precision is equal to recall). The best



(a) Bicycle (b) Car (c) Person

Figure 3. Comparing top-down saliency maps produced by the proposed, DSD and SUN models.

	Bicycle	Car	Person
DSD [6]	62.5	37.6	48.2
SUN [12]	61.9	45.7	52.2
Baseline,$k = 512, \lambda = 0.15$	71.9	39.3	56.8
Ours, $k = 256, \lambda = 0.15$	73.3	57.5	64.2
Ours, $k = 512, \lambda = 0.15$	**80.1**	**68.6**	**72.4**
Ours, $k = 512, \lambda = 0.30$	73.5	66.6	69.6

Table 1. Precision rates (%) at EER on the Graz-02 dataset.

	Bicycle	Car	Person
Ours	**62.4**	**60.0**	**62.0**
[19] (full framework)	61.8	53.8	44.1

Table 2. Precision rates (%) at EER against shape mask [19].

results are obtained by our model with the parameters $k = 512, \lambda = 0.15$. We can see the clear improvements of our models over the baseline and other algorithms. The DSD algorithm selects salient features based on image-level statistics that usually has limited ability of suppressing background image patches. In general, the DSD method generates a high recall rate but a low precision rate. The SUN algorithm performs better than the DSD method due to its use of strong classifier. Without considering the local context, the SUN algorithm tends to produce noisy saliency maps. Our models are able to produce clear saliency maps when target objects appear in different viewpoints and scales with partial occlusions. We compare saliency maps of the DSD, SUN and proposed models in Figure 3.

In Figure 4, we present more saliency maps produced by our models. Note that our saliency model is able to locate objects heavily occluded (e.g., bicycle and cars) whereas state-of-the-art object detection methods are not expected to perform well in such cases.

A saliency map of an image has the size of its patch grid, i.e., 27×37. To visualize the localization performance, we upsample the original saliency map to the size of image so that we get pixel-level results. We notice that our pixel-level saliency maps are similar to the output of shape mask model [19] (approximate object regions). In Table 2, we compare our results with shape masks by measuring pixel-level precision recall rates on the same test set (150 even-numbered images from each object category). Our results are consistently better than those by the shape mask

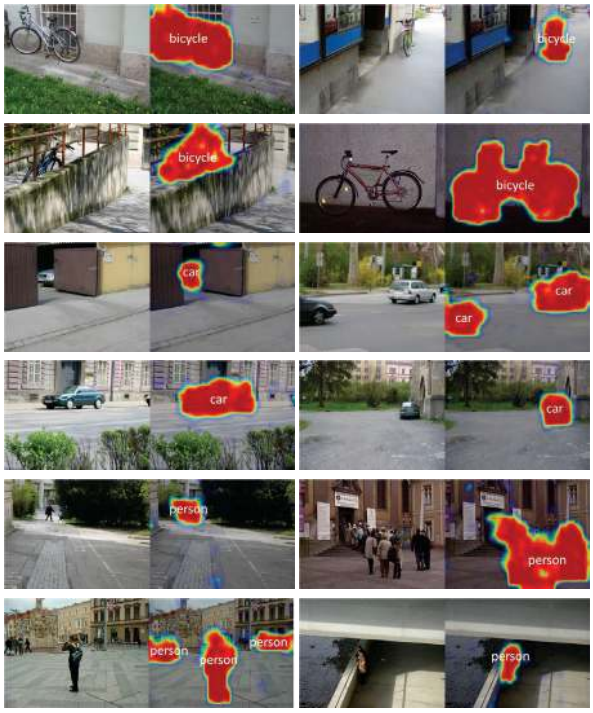


Figure 4. Our saliency maps of bicycle, car and person categories from the Graz-02 dataset. Our model is robust to viewpoint changes, scale variations and partial occlusions.

model [19]. Compared with patch-level results, the performance drop we observe in Table 2 (compared with Table 1) is mainly because: 1) there are many background pixels included with object regions, especially for bicycle images; 2) object boundaries are not preserved in our saliency maps.

Our saliency model jointly learns CRF weight and dictionary from the training examples by gradient updates (Algorithm 1). We are interested in how the dictionary update help improve the model performance. We record the CRF weight and the dictionary at each training cycle and evaluate them on the test set. Figure 6 shows the precision rates at EER of each cycle. It can be seen that the performance improves dramatically in the first several cycles and get converged after 10 cycles. The stochastic nature of our learning algorithm results in some performance perturbation at some

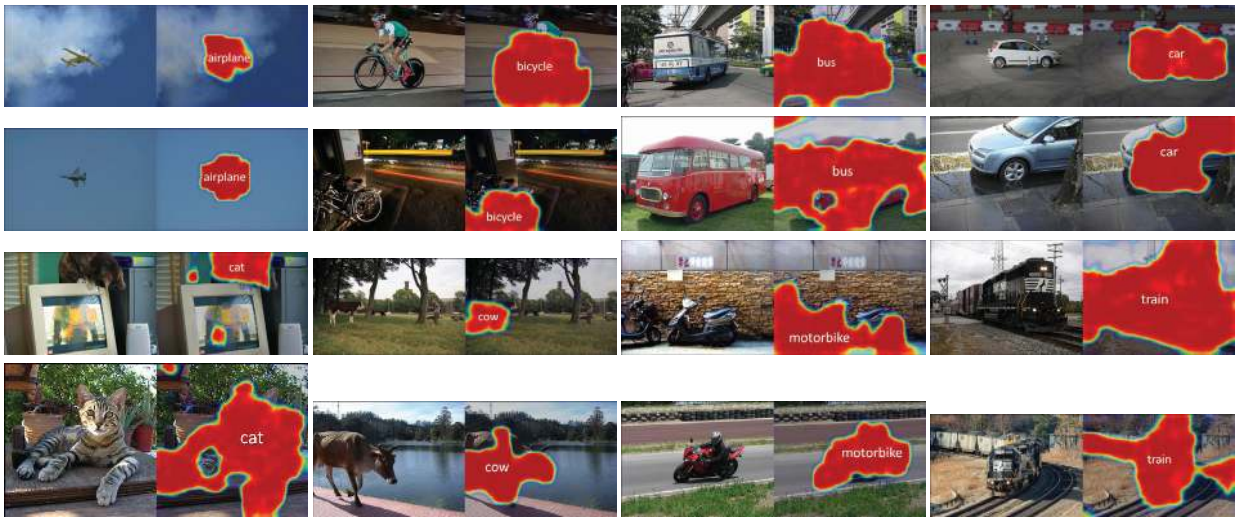
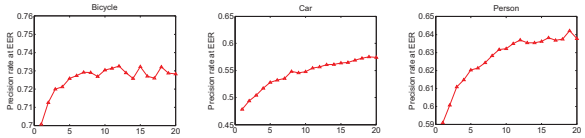


Figure 5. Saliency maps generated by our model.

aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
15.2	39.0	9.4	5.7	3.4	22.0	30.5	15.8	5.7	8
dining table	dog	horse	motorbike	person	potted plant	sheep	sofa	train	tv monitor
11.1	12.8	10.9	23.7	42.0	2.0	20.2	10.4	24.7	10.5

Table 3. Precision rates (%) at EER on the PASCAL VOC 2007 dataset.

cycles. The results show that dictionary update significantly improves the model performance.



(a) bicycle (b) car (c) person

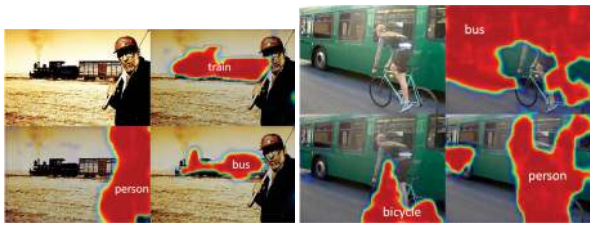
Figure 6. Performance gain with training cycles. The dictionary size $k = 256$ and the sparse penalty $\lambda = 0.15$.

5.2. PASCAL VOC 2007

The PASCAL VOC 2007 dataset consists of 9963 images from 20 categories and background class where object segmentation annotations are available for 632 images. We evaluate our top-down saliency models for the task of localizing target objects against the background and the objects from other categories. Objects from different categories often share similar part appearance. For example, bicycles, motorbikes and buses share similar wheel structures. This phenomenon makes it challenging to discriminate target appearance from the others on the patch level. We use the same training and test split as the PASCAL VOC 2007 object segmentation challenge, i.e., 422 images for training and 210 images for tests. Similar to the experiments on the Graz-02 dataset, we create 20 object saliency masks from segmentation annotations for each image. We notice that only few examples contain target objects for each category in the training set, compared with negative examples. To learn a model from a unbalanced dataset, we also use the bounding box annotations of the positive examples for training. We create saliency masks for those images by measuring whether the sampled patches fall into target bounding boxes. For each category, we train a saliency model with 20 cycles. The number of visual words in the dictionary is 512 and the sparse penalty λ is 0.15.

We present some representative saliency maps in Figure 5. We observe that our saliency model performs well for those objects that have rich inner structures, such as bicycle, motorbike, person and train; while it does not perform well for objects that are identified by their global shapes and colors, such as dinning tables, potted plants, bottles and sofas. We quantitatively evaluate our results with the precision-recall rates. The precision rates at EER are shown in Table 3. Figure 7 shows saliency maps on two images that contain instances from more than one categories. There are some categories that share similar part appearance on the patch level, which causes confusions between relevant category models (7(a)), such as (1) bicycle and motorbike; (2) train and bus; (3) dog and cat.

Our model partially depends on whether the target objects contain rich structured information on the patch level. Taking airplane as example, our model does not work well for the cases where large airplanes are the dominant in the images (e.g., only parts of airplanes are viewable) because local patches of those images contain limited relevant information (i.e., plain patches only) while our model successfully localizes small airplanes at a scale close to the patch size used in the experiments. Considering we sample image



(a) person, train (b) bicycle, bus and person

Figure 7. Multi-class results. The image in the left panel (a) contains a person and a train. We test it with person, train and bus saliency models. The bus model confuse with the train region. The image in the right panel (b) shows a person riding bicycle with a bus as background. Both the bus and bicycle models correctly localize the targets while the person model generates false positives.

patches of same size on a regular grid, our model has limited ability of handling the information loss due to this scale variation. We notice that the object instances are easier to identify by global shapes in many categories (e.g., dinning tables, potted plants, bottles and sofas). Thus, better results can be expected by extending our model in multi-scale or use scale adaptive patch sampling strategies. For top-down visual saliency, we do not incorporate boundary information in our model although such cues are critical for CRF based object segmentation. Nevertheless, our model can be easily extended to exploit superpixels or boundary-preserving regions for object segmentation.

6. Conclusion and Future Work

We have presented a novel top-down visual saliency model via joint CRF and dictionary learning. For each target class, the saliency map is generated on a sampling grid of image patches using the proposed model. Compared to computing saliency values individually on each patch by [6, 12], our saliency map is generated by considering spatial consistency via the proposed CRF model with latent variables. Our model thus produces clear saliency maps by incorporating local context information. The dictionary defines the capacity of representing target appearance from different viewpoints and scales. We observe significant improvements can be achieved by updating dictionary modulated by the proposed CRF model. Our future work includes extending our model with multi-scale patches to better account for large scale variation. In addition, we will also extend our model with boundary-preserving regions or superpixels for object segmentation.

Acknowledgments

This work is supported in part by the NSF CAREER Grant #1149783 and NSF IIS Grant #1152576.

References

[1] L. Bertelli, T. Yu, D. Vu, and B. Gokturk. Kernelized structural svm learning for supervised object segmentation. In *CVPR*, 2011.

[2] Y.-L. Boureau, J. Ponce, and Y. LeCun. Learning mid-level features for recognition. In *CVPR*, 2010.

[3] N. D. B. Bruce and J. K. Tsotsos. Saliency based on information maximization. In *NIPS*, 2006.

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. `http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html`.

[5] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *ICCV*, 2009.

[6] D. Gao, S. Han, and N. Vasconcelos. Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. *PAMI*, 31(6):989–1005, 2009.

[7] J. M. Gonfaus, X. Boix, J. V. D. Weijer, A. D. B. J. Serrat, and J. Gonzlez. Harmony potentials for joint classification and segmentation. In *CVPR*, 2010.

[8] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, 2006.

[9] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007.

[10] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene anaysis. *PAMI*, 20(11):1254–1259, 1998.

[11] T. Joachims, T. Finley, and C.-N. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[12] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell. Sun: Top-down saliency using natural statistics. *Visual Cognition*, 17(8):979–1003, 2009.

[13] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts. *PAMI*, 26:65–81, 2004.

[14] S. Kumar and M. Hebert. Discriminative random fields. *IJCV*, 68(2):179–201, 2006.

[15] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, 2006.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[17] J. Mairal, F. Bach, and J. Ponce. Task-driven dictionary learning. *PAMI*, 32(4):791–804, 2012.

[18] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *NIPS*, 2008.

[19] M. Marszalek and C. Schmid. Accurate object recognition with shape masks. *IJCV*, 97(2):191–209, 2012.

[20] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In *NIPS*, 2006.

[21] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006.

[22] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer. Generic object recognition with boosting. *PAMI*, 28(3):416–431, 2006.

[23] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell. Hidden conditional random fields. *PAMI*, 29(10):1848–1853, 2007.

[24] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-Class object recognition and segmentation by jointly modeling texture, layout, and context. *IJCV*, 81(1):2–23, 2009.

[25] M. Szummer, P. Kohli, and D. Hoiem. Learning crfs using graph cuts. In *ECCV*, 2008.

[26] Y. Wang and G. Mori. Max-margin hidden conditional random fields for human action recognition. In *CVPR*, 2009.

[27] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 2009.

[28] J. Yang, K. Yu, and T. Huang. Supervised translation-invariant sparse coding. In *CVPR*, 2010.

[29] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009.