Visual Dictionary Learning for Joint Object Categorization and Segmentation

Aastha Jain, Luca Zappella, Patrick McClure, and René Vidal

Center for Imaging Science, Johns Hopkins University

Abstract. Representing objects using elements from a visual dictionary is widely used in object detection and categorization. Prior work on dictionary learning has shown improvements in the accuracy of object detection and categorization by learning discriminative dictionaries. However none of these dictionaries are learnt for joint object categorization and segmentation. Moreover, dictionary learning is often done separately from classifier training, which reduces the discriminative power of the model. In this paper, we formulate the semantic segmentation problem as a joint categorization, segmentation and dictionary learning problem. To that end, we propose a latent conditional random field (CRF) model in which the observed variables are pixel category labels and the latent variables are visual word assignments. The CRF energy consists of a bottom-up segmentation cost, a top-down bag of (latent) words categorization cost, and a dictionary learning cost. Together, these costs capture relationships between image features and visual words, relationships between visual words and object categories, and spatial relationships among visual words. The segmentation, categorization, and dictionary learning parameters are learnt jointly using latent structural SVMs, and the segmentation and visual words assignments are inferred jointly using energy minimization techniques. Experiments on the Graz02 and CamVid datasets demonstrate the performance of our approach.

1 Introduction

Joint categorization and segmentation (JCaS) refers to the problem of assigning an object category label to each pixel in a given image. Most existing solutions to this problem use conditional random field (CRF) formulations. The sites of the CRF are image pixels [1], patches [2, 3], superpixels [4–7], or a hierarchy of regions [8, 9]. Local interactions among these sites are captured by unary and pairwise potentials, which model, respectively, the cost of assigning a category label to each site and the spatial smoothness of the segmentation. Long-range interactions among many sites of the CRF are captured by higher-order potentials, which model the statistics of an object and/or encode contextual information. However, such long-range interactions are typically restricted to fairly local neighborhoods to avoid crossing the boundaries of an object. Notable exceptions are [10], which uses interactions among several sites to model co-occurrence statistics between object categories, and [11, 12], which use a bag-of-features (BoF) model to define a potential over the whole region occupied by the object.

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part V, LNCS 7576, pp. 718–731, 2012.

The BoF approach is one of the most widely used models for object categorization. In this approach, an object is represented by the distribution of a set of *visual words*, which are usually obtained by K-means clustering of a set of feature descriptors obtained from the training images. While BoF methods have shown good performance in object categorization [13–15], the visual dictionary may not be descriptive of the object categories, because the words are learnt in an unsupervised manner. Discriminative dictionary learning methods for object categorization such as [16] incorporate class-specific information during dictionary learning, which improves the discriminative capability of the dictionary. However, one drawback of this technique is that the dictionary learning and classifier training steps are done separately. This leads to reduced categorization accuracy because the words, while individually discriminative, may not be optimal for categorization. The work of [17] overcomes this drawback by learning the dictionary and the classifiers simultaneously, and shows improved performance.

In our view, the method used for learning the dictionary depends heavily on the task at hand. To the best of our knowledge, none of the existing dictionary learning techniques has been used to learn dictionaries that are specifically designed for the JCaS problem. For instance, the work of [18] combines CRFs with dictionary learning for object detection purposes, but it does not address segmentation or categorization. Also, the work of [12] uses dictionaries to construct the higher-order CRF potentials, but the method for learning the dictionary is unsupervised K-means. Moreover, this dictionary is kept fixed while learning the categorization and segmentation parameters of the CRF. We believe that unsupervised, discriminative or object-specific dictionaries are suboptimal for solving the JCaS problem, because they are learnt independently from the categorization and segmentation parameters of the CRF, and hence their discriminative power is compromised.

In this work we propose a JCaS framework in which the visual dictionary is learnt jointly with the CRF categorization and segmentation parameters. In our framework, the assignment of key-points to visual words is assumed to be unknown and is modeled as a latent variable of the CRF, which needs to be inferred during inference and training. In addition to the standard potentials defined to ensure smoothness of the segmentation, we introduce a set of potentials that model the interaction between feature descriptors and visual word assignments, and another set of potentials that model the probability that each visual word belongs to a particular object category. We also extend this framework beyond the BoF model and introduce additional potentials that take into account interactions between the visual word assignments of neighboring key-points.

We show that the parameters of this model can be learnt jointly using latent structural SVMs. The visual dictionary learnt in this manner uses the information from the categorization and segmentation parameters, which increases the discriminative power of our model. Given the model parameters, we show that the segmentation and visual word assignments can be found using graph cuts or loopy-belief propagation. Experiments on the Graz02 and CamVid databases show that our approach improves segmentation accuracy of structured objects.

2 A CRF-BoF Model for JCaS

In this section, we review the CRF model for JCaS proposed in [12], which is based on a BoF categorization cost. While [12] uses kernel SVM classifiers with the intersection kernel, we will describe the model using linear classifiers.

CRF Structure. The CRF is defined over a set of superpixels $\mathcal{V}$ extracted from the image I. Each site $i \in \mathcal{V}$ is associated with an object category label $x_i \in \mathcal{L} = \{1, \ldots, L\}$. The labelling of the image is denoted by the vector $x \in \mathcal{L}^{|\mathcal{V}|}$. The interaction between various sites of the CRF is captured by the set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, where each edge $e_{ij} \in \mathcal{E}$ corresponds to a pair of superpixels $i, j \in \mathcal{V}$ that share a boundary. Besides sites and edges, we can also define higher-order cliques. A clique is a subset of sites $c \subset \mathcal{V}$ whose labels, x_c, are conditionally dependent on each other. For example, a site $i \in \mathcal{V}$ is a clique of size 1 and an edge $e_{ij} \in \mathcal{E}$ corresponds to a clique of size 2.

Having defined the structure of the random field, in what follows we define the CRF energy, which consists of a segmentation cost and a categorization cost.

Bottom-Up Segmentation Cost. We define the *segmentation cost* as:

$$E_{seg}(x, I) = \lambda_U \sum_{i \in \mathcal{V}} \psi_i^U(x_i, I) + \lambda_P \sum_{e_{ij} \in \mathcal{E}} \psi_{ij}^P(x_i, x_j, I) = w_{seg}^\top \Psi_{seg}(x, I), \qquad (1)$$

where $\lambda_U \geq 0$ and $\lambda_P \geq 0$ are the relative weights of the unary and pairwise potentials, $w_{seg}^\top = \begin{bmatrix} \lambda_U & \lambda_P \end{bmatrix}$ and $\Psi_{seg}(x, I) = \begin{bmatrix} \sum_{i \in \mathcal{V}} \psi_i^U(x_i, I) \\ \sum_{e_{ij} \in \mathcal{E}} \psi_{ij}^P(x_i, x_j, I) \end{bmatrix}$.

The unary potential, $\psi_i^U(x_i, I)$, models the cost of assigning a class label $x_i \in \mathcal{L}$ to superpixel i in image I. It is defined as the score of a kernel SVM classifier for class x_i applied to a normalized histogram of quantized SIFT features extracted from a neighborhood of superpixel i of size τ (see [19]). The classifier for class $l \in \mathcal{L}$ is trained using the normalized histograms extracted from the superpixels in the training set whose label is l. We use the RBF-χ^2 kernel $k(f, g) = e^{-\gamma \chi^2(f,g)}$.

The pairwise potential, $\psi_{ij}^P(x_i, x_j, I)$, models the cost of assigning labels x_i and x_j to sites i and j, respectively. When using a CRF formulation for segmentation, the pairwise potentials are typically used to ensure the smoothness of the label assignments. We use a contrast sensitive cost $\frac{L_{ij} \delta(x_i \neq x_j)}{1 + \|\bar{I}_i - \bar{I}_j\|}$, where L_{ij} is the length of the shared boundary between superpixels i and j, and $\bar{I}_i$ and $\bar{I}_j$ are the mean color (in the LUV space) of superpixels i and j, respectively.

BoF Categorization Cost. The segmentation cost models only the local features and characteristics of the image. To infer the class of an object, we also need to account for long-range interactions between various sites in the CRF.

One way to capture such long-range interactions is to represent each object class with a BoF model, and define a categorization potential over all the sites corresponding to an object class. To that end, let $\{\theta_k\}_{k \in \mathcal{K}}$ be a dictionary of $|\mathcal{K}|$ visual words. This dictionary is obtained by applying K-means to the SIFT features extracted from all the superpixels in the training images. We represent each image I with $|\mathcal{L}|$ histograms of these visual words. More specifically, let S_I denote a set of key-points extracted from image I and let d_p^I denote the feature descriptor of key-point $p \in S_I$. Each descriptor d_p^I is assigned to its closest word, $\theta(d_p^I)$, and a histogram of these words, $h_l(x, I) \in \mathbb{R}_+^K$, is used to represent the portion of the image occupied by object class l. We can write the histogram bin count for class l corresponding to visual word k as

$$h_{l,k}(x, I) = \sum_{p \in S_I} \delta(\theta(d_p^I) = \theta_k)\delta(x_{i_p} = l) = \sum_{p \in S_I^k} \delta(x_{i_p} = l), \tag{2}$$

where $i_p \in \mathcal{V}$ is the superpixel associated with key-point $p \in S_I$ and $S_I^k \subset S_I$ is the set of key-points associated to word k.

Let us now define a classifier Φ_l for each class label l, where $\Phi_l(h) : \mathbb{R}_+^{|\mathcal{K}|} \mapsto \mathbb{R}$ represents the score assigned to the histogram h. Using a linear classifier we have

$$\Phi_l(h) = \alpha_l^\top h + \beta_l = \sum_{k \in \mathcal{K}} \alpha_{l,k} h_k + \beta_l, \tag{3}$$

where $\alpha_l \in \mathbb{R}^{|\mathcal{K}|}$ and $\beta_l \in \mathbb{R}$ are the parameters of the linear classifier for class l. With the above notation, we define a *categorization cost*, E_{cat}, as the sum of the categorization costs for each class, i.e.,

$$E_{cat}(x, I) = \sum_{l \in \mathcal{L}} \Phi_l(h_l(x, I))\delta(||h_l(x, I)|| > 0). \tag{4}$$

Notice that we pay no cost when no key-point is assigned to category l, i.e., $E_{cat}(x, I) = 0$ when $h_l(x, I) = 0$. Also, since we wish to minimize this cost, we train classifiers Φ_l to assign low scores to histograms in class l and high scores to histograms corresponding to other classes (contrary to the usual convention).

Although the energy E_{cat} may seem a complicated function of x, notice that we can write it linearly in terms of the classifier parameters as

$$E_{cat}(x, I) = w_{cat}^\top \Psi_{cat}(x, I) = \begin{bmatrix} \cdots & \alpha_{l,k} & \beta_l & \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ \psi_{l,k}^H(x, I) \\ \psi_l^\delta(x, I) \\ \vdots \end{bmatrix}, \tag{5}$$

where $\psi_{l,k}^H(x, I) = \sum_{p \in S_I^k} \delta(x_{i_p} = l)$ and $\psi_l^\delta(x, I) = \min\{1, \sum_{p \in S_I} \delta(x_{i_p} = l)\}$.

Inference and Learning. The inference task is to minimize the *segmentation and categorization cost*:

$$E_{seg+cat}(x, I) = w_{seg+cat}^\top \Psi_{seg+cat}(x, I) = \begin{bmatrix} w_{seg}^\top & w_{cat}^\top \end{bmatrix} \begin{bmatrix} \Psi_{seg}(x, I) \\ \Psi_{cat}(x, I) \end{bmatrix}. \tag{6}$$

The segmentation cost is a standard unary+pairwise cost that can be minimized by graph cuts [20]. On the other hand, the categorization cost is a higher-order cost defined over a clique of size $|S_I|$ formed by the all the superpixels that contain key-points. It is shown in [12] that this higher-order potential belongs to the class of robust Potts model [21], which can be minimized by graph cuts.

The learning task is to estimate the parameters of the energy $w_{seg+cat}$ from a training set of segmented images $\{I^j\}_{j=1}^N$ and their corresponding labellings $\{x^j\}_{j=1}^N$. Since the energy is linear in the parameters λ_U, λ_P, $\{\alpha_l, \beta_l\}_{l\in\mathcal{L}}$, we can use the cutting plane training algorithm for structural SVMs [22] to learn these parameters, as shown in [12]. Since negative examples (wrong segmentations) are not given, at each step of the cutting plane algorithm, the wrong segmentation with the worst margin is selected. This can also be done using graph cuts.

For more details on inference and learning, we refer the reader to [12].

3 A Latent CRF-BoF Model for JCaS

In the BoF model for JCaS described in the previous section, K-means clustering is used to generate the visual dictionary. Therefore, this visual dictionary is learnt independently from the categorization and segmentation parameters. Since dictionary learning helps categorization and segmentation, and knowledge about the categorization and segmentation parameters helps dictionary learning, a more meaningful dictionary could be obtained by learning the visual words together with the categorization and segmentation parameters.

In this section, we will re-formulate the energy E to incorporate the visual words as additional parameters of the energy and the visual word assignments as latent variables. We introduce a *categorization cost* that depends on both the segmentation and the visual word assignments. This cost captures the relationships among the visual words and the object categories. We also introduce a new *dictionary learning* cost, which relates the image features to the visual words.

Latent BoF Categorization Cost. Let $\{\theta_k\}_{k\in\mathcal{K}}$ be an (unknown) dictionary of $|\mathcal{K}|$ visual words. Instead of fixing the visual word assignment prior to categorization, we associate a random variable $z_p \in \mathcal{K}$ with every key-point $p \in S_I$. The vector $z \in \mathcal{K}^{|S_I|}$ is then a latent variable of a CRF defined over the key-points. Let us recall the categorization cost introduced in (4)-(5). This cost depends on the histogram counts $h_{l,k}(x, I) = \sum_{p\in S_I^k} \delta(x_{i_p} = l)$, where the set of key-points S_I is divided into $|\mathcal{K}|$ disjoint subsets S_I^k containing the key-points assigned to visual word k. Since the word assignments are unknown, so are the sets S_I^k. Nonetheless, we can easily express $h_{l,k}$ in terms of the word assignment variables as

$$h_{l,k}(x, z, I) = \sum_{p\in S_I} \delta(x_{i_p} = l, z_p = k). \tag{7}$$

Therefore, the categorization cost in (5) can be re-written as:

$$E_{cat}(x, z, I) = w_{cat}^\top \Psi_{cat}(x, z, I) = \begin{bmatrix} \cdots & \alpha_{l,k} & \beta_l & \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ \psi_{l,k}^H(x, z, I) \\ \psi_l^\delta(x, I) \\ \vdots \end{bmatrix}, \tag{8}$$

where $\psi_{l,k}^H(x, z, I) = \sum_{p\in S_I} \delta(x_{i_p} = l, z_p = k)$.

Dictionary Learning Cost. We now define a *dictionary learning cost* that relates the word assignments $\{z\}_{p\in S_I}$ to the image features $\{d_p^I\}_{p\in S_I}$. In standard K-means, the assignment of a feature d_p^I to a visual word θ_k is given by

$$z_p = \operatorname*{arg\,min}_{k\in\mathcal{K}} ||\theta_k - d_p^I||^2. \tag{9}$$

In our formulation, however, θ_k is unknown and our goal is to learn the dictionary together with the segmentation and categorization parameters. To that end, we re-interpret θ_k as the parameters of a classifier, rather than as a cluster center. Specifically, we let θ_k be the parameters of a linear classifier for visual word k. If the visual words were learnt independently from the other parameters, we could determine z by assigning d_p^I to the classifier with the lowest score,[1] i.e.,

$$z_p = \operatorname*{arg\,min}_{k\in\mathcal{K}} \theta_k^\top d_p^I. \tag{10}$$

However, since our goal is to learn all the parameters simultaneously, we define an additional *dictionary learning cost*, which captures the cost of assigning feature descriptor d_p^I to visual word θ_k with a word assignment z. This cost is defined as:

$$E_{dict}(z,I) = \sum_{p\in S_I}\sum_{k\in\mathcal{K}} \delta(z_p = k)\theta_k^\top d_p^I = \sum_{k\in\mathcal{K}} \theta_k^\top \psi_k^D(z,I) = w_{dict}^\top \Psi_{dict}(z,I), \tag{11}$$

where $\psi_k^D(z,I) = \sum_{p\in S_I} \delta(z_p = k) d_p^I$, $w_{dict}^\top = \begin{bmatrix}\theta_1^\top & \theta_2^\top & \cdots & \theta_{|\mathcal{K}|}^\top\end{bmatrix}$, and

$$\Psi_{dict}(z,I) = \begin{bmatrix} \psi_1^D(z,I) \\ \vdots \\ \psi_{|\mathcal{K}|}^D(z,I)\end{bmatrix}. \tag{12}$$

Joint Inference of the Segmentation and Visual Words Assignments. We propose to solve the JCaS problem by minimizing the following energy over both the class labels and the visual word labels $(x,z) \in \mathcal{L}^{|\mathcal{V}|} \times \mathcal{K}^{|S_I|}$

$$E_{seg+cat+dict}(x,z,I) = w^\top \Psi(x,z,I) = \begin{bmatrix} w_{seg}^\top & w_{cat}^\top & w_{dict}^\top\end{bmatrix} \begin{bmatrix} \Psi_{seg}(x,I) \\ \Psi_{cat}(x,z,I) \\ \Psi_{dict}(z,I)\end{bmatrix}. \tag{13}$$

Notice that the minimization over z is equivalent to

$$\min_{z\in\mathcal{K}^{|S_I|}} \sum_{k\in\mathcal{K}}\sum_{p\in S_I} \Big(\sum_{l\in\mathcal{L}} \alpha_{l,k}\delta(x_{i_p} = l) + \theta_k^\top d_p^I\Big)\delta(z_p = k). \tag{14}$$

Therefore, given x, the optimal z can be computed in closed form as:

[1] Notice that, modulo the sign change due to the change on the standard convention for defining the classifiers, the proposed assignments are equivalent to those of K-means when the features and the words are normalized such that $\|d_p^I\| = 1$ and $\|\theta_k\| = 1$, because $z_p = \operatorname{arg\,min}_{k\in\mathcal{K}} ||\theta_k - d_p^I||^2 = \operatorname{arg\,min}_{k\in\mathcal{K}} ||\theta_k||^2 + ||d_p^I||^2 - 2\theta_k^\top d_p^I$.

$$z_p = \arg\min_{k \in \mathcal{K}} \Big(\sum_{l \in \mathcal{L}} \alpha_{l,k} \delta(x_{i_p} = l) + \theta_k^\top d_p^I \Big). \tag{15}$$

Conversely, given z, the word assignments are fixed, and the optimization problem over x reduces to that considered in §2, which can be solved using graph cuts, as shown in [12]. By alternating between these two steps till convergence, we obtain a local minimizer of $E_{seg+cat+dict}(x, z, I)$.

4 A Latent Structural CRF-BoF Model for JCaS

The latent CRF model for JCaS discussed in Section 3 represents objects as a collection of latent words. A drawback of this model is that it fails to take into account the relative positions of these words with respect to each other. The work of [7] shows that the structure of an object defined by the relative arrangement of its features can play an important role in identifying it. In this section, we extend the model for JCaS discussed in Section 3 to capture the dependencies between the visual word assignments of the neighboring key-points.

Dictionary Structure Cost. Let us define a second CRF $(\mathcal{V}', \mathcal{E}')$, whose nodes are the key-points, i.e., $\mathcal{V}' = S_I$, and whose edges $e'_{pq} \in \mathcal{E}' \subset \mathcal{V}' \times \mathcal{V}'$ connect a key-point $p \in S_I$ to its n nearest neighbors $q \in S_I$. We define the CRF energy as

$$\begin{aligned} E_{struct}(z, I) &= \sum_{e'_{pq} \in \mathcal{E}'} \sum_{(k_1,k_2) \in \mathcal{K} \times \mathcal{K}} \rho_{k_1,k_2} \delta(z_p = k_1, z_q = k_2) && (16) \\ &= \sum_{(k_1,k_2) \in \mathcal{K} \times \mathcal{K}} \rho_{k_1,k_2} \psi^S_{k_1,k_2}(z, I) = w_{struct}^\top \Psi_{struct}(z, I) && (17) \end{aligned}$$

where $\rho_{k_1,k_2} \in \mathbb{R}$ models the negative log probability of the co-occurrence of visual words k_1 and k_2 as neighbors, $\psi^S_{k_1,k_2}(z, I) = \sum_{e'_{pq} \in \mathcal{E}'} \delta(z_p = k_1, z_q = k_2)$,

$$w_{struct}^\top = \begin{bmatrix} \cdots & \rho_{k_1,k_2} & \cdots \end{bmatrix} \text{ and } \Psi_{struct}(z, I) = \begin{bmatrix} \vdots \\ \phi^S_{k,l}(z, I) \\ \vdots \end{bmatrix}.$$

Joint Inference of the Segmentation and Visual Words Assignments. We propose to solve the JCaS problem by minimizing the following energy over both the class labels and the visual word labels $(x, z) \in \mathcal{L}^{|\mathcal{V}|} \times \mathcal{K}^{|S_I|}$

$$E(x, z, I) = E_{seg}(x, I) + E_{cat}(x, z, I) + E_{dict}(z, I) + E_{struct}(z, I). \tag{18}$$

As before, given z, the word assignments are fixed, and the optimization problem over x reduces to that considered in §2, which can be solved using graph cuts. However, given x, the minimization over z cannot be done by graph cuts, because the pairwise potentials in E_{struct} may not satisfy the sub-modularity constraint $\rho_{k_1,k_2} < \rho_{k_1,k_3} + \rho_{k_3,k_1}$. While this simple linear constraint could easily be enforced during learning, doing so would mean that we disregard the actual visual word co-occurrence probabilities and impose an artificial constraint that might not necessarily hold. This can cause the co-occurrence probabilities of visual words to result in a model which is not coherent with the object structure. Therefore, we choose not to impose these constraints on the parameters and resort to loopy-belief propagation techniques [23, 24] for computing the optimal z given x. We alternate these two steps till convergence to a local minimum.

5 Max-Margin Learning Using Latent SVMs

So far, we have proposed a new framework for JCaS based on minimizing the energy $E(x,z,I) = w^\top \Psi(x,z,I)$. As shown in §3 and §4, this problem can be solved using graph cuts and/or loopy-belief propagation.

In this section, we consider the learning problem. That is, given a collection of training images $\{I^j\}_{j=1}^N$ and their corresponding ground-truth segmentations $\{\hat{x}^j\}_{j=1}^N$, our goal is to learn the parameters w. The main challenge is that we do not know the latent variables $\{z^j\}_{j=1}^N$. We address this problem using the latent structural SVM framework proposed in [25]. In this framework, the parameters w are learnt using an iterative algorithm that alternates between solving for the latent variables z^j given the energy parameters w, and solving for the parameters w given the latent variables. More specifically, the algorithm proceeds as follows:

- Step 1: Given a current estimate of the energy parameters $\hat{w}$, compute an estimate $\hat{z}^j$ of the latent variables as
$$\hat{z}^j = \arg\min_{z \in \mathcal{K}^{|S_{I^j}|}} \hat{w}^\top \Psi(\hat{x}^j, z, I^j) \quad \forall j \in \{1, \cdots, N\}. \tag{19}$$
- Step 2: Given an estimate $\hat{z}^j$ of the latent variables, learn w by solving the following structural SVM training problem [22]
$$\begin{aligned} &\{w^*, \{\xi_j^*\}_{j=1}^N\} = \arg\min\nolimits_{w,\{\xi_j\}_{j=1}^N} \frac{1}{2}\|w\|^2 + \frac{\mu}{N}\sum_{j=1}^N \xi_j, \text{ subject to} \\ &\text{(a) } \forall i = 1, \dots, N\text{: } \forall (x,z) \in \mathcal{L}^{|\mathcal{V}|} \times \mathcal{K}^{|S_I|} : \\ &\qquad w^\top \left(\Psi(x,z;I^j) - \Psi(\hat{x}^j, \hat{z}^j; I^j)\right) \geq \Delta(x, \hat{x}^j) - \xi_j, \\ &\text{(b) } \forall i = 1, \dots, N\text{: } \xi_i \geq 0 \quad \text{and} \quad \text{(c) } w \geq \mathbf{0}. \end{aligned} \tag{20}$$

In step 2, we want to learn the parameter vector w such that the value of E for the ground truth segmentation $\hat{x}$ and imputed word assignments $\hat{z}$ is smaller than its value for other possible labellings and words assignments, i.e., $\forall (x,z) \in \mathcal{L}^{|\mathcal{V}|} \times \mathcal{K}^{|S_I|} \setminus (\hat{x}, \hat{z})$, $E(\hat{x}, \hat{z}, I) < E(x,z,I)$. However, all wrong segmentations and assignments can not be penalized equally. For example, a segmentation that labels one superpixel wrong is better than one that labels 80% of the superpixels wrong. The loss function $\Delta(x, \hat{x})$ measures the deviation of a given segmentation from the ground truth as
$$\Delta(x, \hat{x}) = \frac{d(x, \hat{x})}{|x|}, \tag{21}$$
where $d(x, \hat{x})$ measures the number of sites which have different labels in x and $\hat{x}$. Finally, ξ_j represents the slack variable for training example j. The introduction of slack variables is necessary because, otherwise, there may not be a set of parameters w that satisfies all the constraints described in (20).

We refer the reader to [22] for the details of the cutting plane method used to solve this optimization problem efficiently.

6 Experiments

Datasets. We performed our experiments on the Graz-02 dataset [26] and the CamVid dataset [27]. The Graz dataset contains 900 images of bikes, humans and cars. 450 images were used for training and the rest of the images were used for testing. The CamVid dataset has 700 labelled images out of which we used 350 for training and 350 for testing. The CamVid dataset has been annotated into 32 classes out of which we chose the 11 classes that were considered in [28]. Also, to make our results comparable with prior work, we down-sampled the images by 1/3 and did not consider the void class. For both databases, we created the superpixels using the quickshift method [19].

Metrics. We compared the different methods using two performance metrics. The *pixel accuracy* is the percentage of correctly labeled pixels per image averaged over all the images. The *intersection/union metric* considers not only the true positives (TP), but also the false positives (FP) and false negatives (FN). For each image, the intersection/union metric is computed as $\frac{100\times\#\text{TP}}{\#\text{TP}+\#\text{FP}+\#\text{FN}}$.

Methods and Baselines. We compared the following methods:

1. CRF-U: this is a simplified version of the method described in Section 2, which uses only the unary segmentation cost. This method is a particular case of that in [6]. We report results from our implementation.
2. CRF-UP: this is a simplified version of the method described in Section 2, which uses only the unary and pairwise segmentation costs. This is the method proposed in [6]. We report results from our implementation.
3. CRF-BoF-L: this method is described in Section 2. It uses a CRF model with a BoF categorization cost constructed using linear classifiers. This method is a particular case of that in [12]. We report results from our implementation.
4. CRF-BoF-IK: this is the method proposed in [12], which uses a CRF model with a BoF categorization cost constructed using kernel-SVM classifiers with the intersection kernel. We report results from [12].
5. LCRF-BoF-L: this method is described in Section 3. It uses a latent CRF model with a BoF cost with linear classifiers plus a dictionary learning cost.
6. SLCRF-BoF-L: this method is described in Section 4. It uses a latent CRF model with a BoF cost with linear classifiers plus a structured dictionary learning cost.

Implementation Details. The parameters of the unary segmentation cost are chosen as follows. The size of the superpixel neighborhood used to define the features for the unary classifiers is set to $\tau = 8$. The number of visual words is set to $|\mathcal{K}| = 400$. The parameter of the RBF kernel is set to $\gamma = 1/\xi^2_{0.25}$, where $\xi^2_{0.25}$ is the first quartile of the ξ^2 distances in the training set. The parameter of the categorization cost is set to $|\mathcal{K}| = 20$ visual words. The parameter of the structural SVM learning method is set to $\mu = 10^6$.

Table 1. Performance of different methods on the Graz02 database using the intersection/union metric. CRF-U gives the results of using only the unary segmentation cost in [6] (CRF-U), while CRF-UP gives the results of using both unary and pairwise segmentation costs in [6]. CRF-BoF-L and CRF-BoF-IK give the results of using a bag of features models for the objects using the linear kernel and the intersection kernel, respectively, as described in [12]. LCRF-BoF and SLCRF-BoF give the results of our latent CRF models without and with structure in the dictionary cost, respectively.

	background	bikes	cars	humans	mean
CRF-U	69.39	36.85	29.27	28.30	40.95
CRF-UP	79.12	42.02	**45.06**	37.09	50.82
CRF-BoF-L	79.48	42.82	44.34	37.38	51.01
CRF-BoF-IK [12]	**82.32**	46.18	36.49	**38.99**	50.99
LCRF-BoF-L	76.40	53.57	39.22	37.05	51.19
SLCRF-BoF-L	77.97	**55.60**	41.51	37.26	**53.08**

Table 2. Performance of the different methods on the Graz02 database using the pixel accuracy metric. The acronyms for the different methods are as in Table 1.

Method	background	bikes	cars	humans	mean
CRF-U	70.63	86.42	**77.36**	**81.81**	79.05
CRF-UP	81.42	86.40	76.46	77.17	**80.36**
CRF-BoF-L	81.81	**86.47**	74.48	77.86	80.16
CRF-BoF-IK [12]	**86.44**	73.01	68.71	71.32	74.87
LCRF-BoF-L	78.34	84.33	74.71	78.51	78.97
SLCRF-BoF-L	75.90	84.91	76.74	79.78	79.33

Results on the Graz02 Dataset. Table 1 shows the values of the intersection-union metric for the different methods we compared. We observed a significant improvement in the segmentation accuracy for bikes and cars when latent variable models were used. Incorporating structure by adding the pairwise term between the latent variables led to a further improvement in the segmentation accuracy (again, for bikes and cars). We did not see this improvement in the humans category because of the significant variations in human poses. It is harder to capture the structural variations in humans than in bikes and cars. The accuracy for background did not increase significantly for the same reason. Because of the clutter in the background, there is no structure to capture, so the improvement attained by adding the pairwise term between the latent variables is very small. Table 2 shows the pixel accuracies for the Graz02 dataset. Notice that the method based on unary costs only (CRF-U) performs very well according to this metric. This is because the objective function used to learn the unary terms is directly related to the pixel accuracy, which does not penalize false positives.

Figure 1 shows some qualitative results on six images from the Graz02 dataset. Overall, SLCRF-BoF-L performs better than LCRF-BoF-L, which in turn performs better than CRF-BoF-L. For example, in the first row the error on the back wheel is corrected. Likewise, in the third row the size of person hallucinated in lieu of the trash can is reduced. However, important errors still persist. In the first row, for example, a piece of the background is labeled as bike. We believe this is partly due to the fact that in the Graz02 dataset the interior of

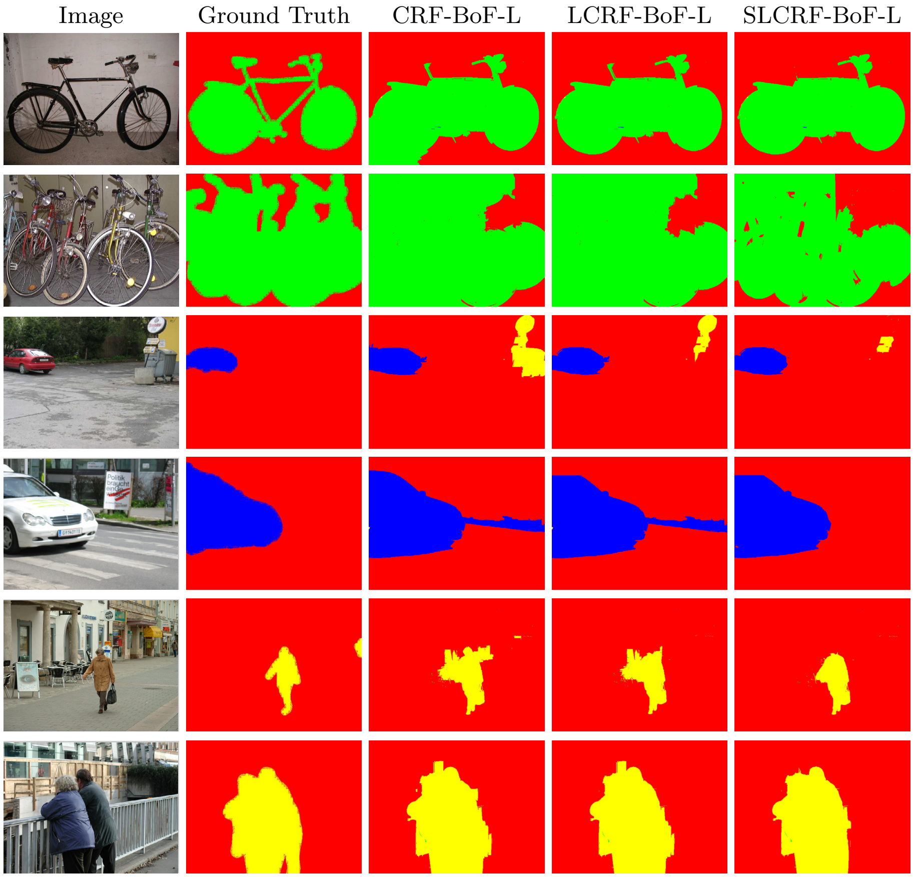

Fig. 1. JCaS results for the Graz-02 dataset: Background, bikes, cars and humans are color coded as red, green, blue and yellow respectively

the wheels is labeled as bike, while the background is also visible. This may also be the cause for the erroneous segmentation of the bikes in the second row.

Results on the CamVid Dataset. Since the CamVid dataset has 11 classes, we could not run the experiments using the CRF-BoF-IK method proposed in [12], because the optimization with graph cuts became prohibitive due to the number of auxiliary variables needed to implement the intersection kernel. Table 3 shows the values of the intersection-union metric for the remaining methods. We observed that for objects with a clearly defined structure, such as cars, signs and buildings, the proposed latent models (LCRF-BoF-L and SLCRF-BoF-L) performed well. However for textured objects such as sky, road and fence, the purely bottom-up methods (CRF-U and CRF-UP) gave better results, because the top-down cost captures an object model that is not very relevant for such. Overall, SLCRF-BoF-L is not as effective as LCRF-BoF-L because of the absence of structure in object categories like sky. This leads to very bad results for the

Table 3. Performance of different methods on the CamVid database using the intersection/union metric. The acronyms for the different methods are as in Table 1.

	Bldg	Roads	Tree	Sky	Car	Ped	Fence	Col.	SW	Bike	Sign	Mean
CRF-U	46.84	68.03	43.26	54.84	27.75	6.33	24.53	2.84	35.73	16.50	5.50	30.20
CRF-UP	51.61	**74.96**	46.61	**65.54**	31.09	7.43	**27.94**	3.33	**39.47**	17.83	7.21	33.91
CRF-BoF-L	42.95	40.53	38.96	48.55	12.79	6.19	9.47	1.58	17.18	4.67	1.17	20.37
LCRF-BoF-L	48.73	72.22	**52.01**	45.58	37.15	9.79	23.64	**8.62**	33.88	**29.35**	12.18	**33.92**
SLCRF-BoF-L	**51.91**	53.75	48.84	34.09	**37.18**	**9.90**	23.21	8.09	23.00	28.87	**14.80**	30.33

Table 4. Performance of the different methods on the CamVid database using the pixel accuracy metric. The acronyms for the different methods are as in Table 1.

Method	Bldg	Roads	Tree	Sky	Car	Ped	Fence	Col.	SW	Bike	Sign	Mean
CRF-U	52.35	71.89	60.14	64.91	54.05	42.44	59.06	15.25	71.83	65.22	40.76	54.36
CRF-UP	56.18	80.16	61.43	77.08	53.86	41.10	**60.60**	15.97	73.39	60.16	41.59	56.50
CRF-BoF-L	47.98	42.05	59.50	56.74	15.14	13.62	35.33	17.54	51.11	60.86	38.72	39.87
LCRF-BoF-L	51.65	80.30	65.73	74.47	53.47	49.75	56.45	**28.77**	71.75	**73.70**	**46.37**	**59.31**
SLCRF-BoF-L	55.60	60.01	65.37	54.82	55.02	43.62	52.21	28.50	70.11	71.68	44.03	54.63
STL [28]	68.90	82.10	**79.30**	**96.70**	54.10	18.20	43.10	1.00	75.90	40.80	19.10	54.65
2D-3D [29]	**71.10**	**88.40**	56.10	89.50	**76.50**	**59.10**	4.80	11.40	**84.70**	28.10	12.50	52.99

SLCRF-BoF-L method because it tries to model the structure of these categories, which leads to over-fitting.

Table 4 shows the values of the pixel accuracies for the CamVid database. In this case, we also compare our results to those obtained using supervised label transfer (STL) [28] and joint 2D-3D segmentation of street scenes (2D-3D) [29]. For textured classes such as trees, road and sky, STL performs better than the latent variable models. However, on structured objects such as bikes and cars, 2D-3D performs better because it captures the 3D structure of the object. The overall performance of these methods is worse than that of LCRF-BoF-L because they perform very poorly for certain classes, while LCRF-BoF-L and SLCRF-BoF-L are more consistent in their performance.

Finally, Figure 2 shows some qualitative results on two images from the CamVid dataset. Overall, SLCRF-BoF-L performs slightly better than LCRF-BoF-L, which in turn performs better than CRF-BoF-L.

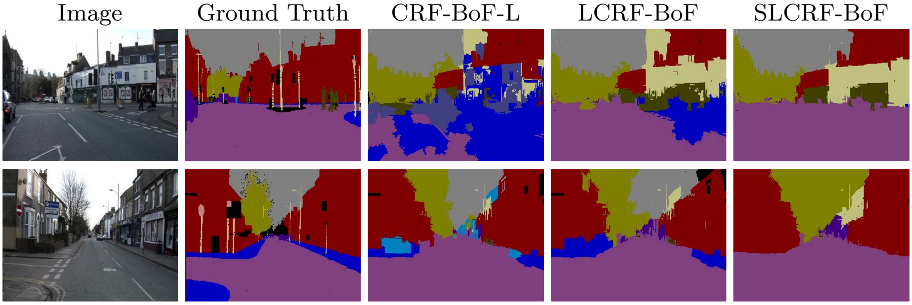

Fig. 2. JCaS results for the CamVid dataset: buildings, roads, tree, sky, car, pedestrian, fence, column, sidewalk, bike, sign are color coded as dark red, light purple, green, gray, dark purple, dark green, dark blue, light green, blue, sky blue, and pink, respectively.

7 Conclusion

We have shown that learning a task specific dictionary jointly with the classification and segmentation parameters leads to a significant improvement in the accuracy of the results. Our experiments suggest that for object classes where spatial context is important, using context-based dictionary learning (latent CRFs with connected hidden variables) increases the accuracy of results. However, for classes such as sky or fence, which do not have a fixed structure, context-based dictionary learning leads to inaccurate segmentations.

In future work, it would be interesting to identify the set of object models which can be used along with the latent CRF formulation for JCaS. A comparison of the results of task specific context dependent dictionary learning with different kinds of neighborhood structures (not just nearest neighbors) can provide us with more accurate object models and better segmentations. Identifying the neighborhood structures and potentials which would allow us to use faster inference techniques (unlike loopy belief propagation) and studying the effects of using a different feature descriptor or a combination of feature descriptors for the interest points are other promising directions.

Acknowledgments. This work was partially supported by grants DARPA FA8650-11-1-7153, ONR N00014-09-10839, NSF 1004782.

References

1. Winn, J.M., Shotton, J.: The layout consistent random field for recognizing and segmenting partially occluded objects. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 37–44 (2006)
2. Larlus, D., Jurie, F.: Combining appearance models and Markov random fields for category level object segmentation. In: IEEE Conf. on Computer Vision and Pattern Recognition (2008)
3. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: IEEE Conf. on Computer Vision and Pattern Recognition (2008)
4. Galleguillos, C., Rabinovich, A., Belongie, S.: Object categorization using co-occurrence, location and appearance. In: IEEE Conf. on Computer Vision and Pattern Recognition (2008)
5. Gould, S., Gao, T., Koller, D.: Region-based segmentation and object detection. In: Neural Information Processing Systems (2009)
6. Fulkerson, B., Vedaldi, A., Soatto, S.: Class segmentation and object localization with superpixel neighborhoods. In: IEEE Int. Conf. on Computer Vision (2009)
7. Micusik, B., Kosecka, J.: Semantic segmentation of street scenes by superpixel co-occurrence and 3d geometry. In: IEEE Workshop on Video-Oriented Object and Event Classification (2009)
8. Ladicky, L., Russell, C., Kohli, P., Torr, P.: Associative hierarchical CRFs for object class image segmentation. In: IEEE Int. Conf. on Computer Vision (2009)
9. Lempitsky, V.S., Vedaldi, A., Zisserman, A.: A pylon model for semantic segmentation. In: Neural Information Processing Systems (2011)
10. Russell, C., Ladicky, L., Kohli, P., Torr, P.: Graph Cut Based Inference with Co-occurrence Statistics. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 239–253. Springer, Heidelberg (2010)

11. Verbeek, J., Triggs, B.: Scene segmentation with CRFs learned from partially labeled images. In: Neural Information Processing Systems (2008)
12. Singaraju, D., Vidal, R.: Using global bag of features models in random fields for joint categorization and segmentation of objects. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
13. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: IEEE International Conference on Computer Vision, pp. 1470–1477 (2003)
14. Dance, C., Willamowski, J., Fan, L., Bray, C., Csurka, G.: Visual categorization with bags of keypoints. In: European Conference on Computer Vision (2004)
15. Winn, J.M., Criminisi, A., Minka, T.P.: Object categorization by learned universal visual dictionary. In: IEEE International Conference on Computer Vision (2005)
16. Moosmann, F., Triggs, B., Jurie, F.: Fast discriminative visual codebooks using randomized clustering forests. In: Neural Information Processing Systems, vol. 19, pp. 985–991 (2007)
17. Yang, L., Jin, R., Sukthankar, R., Jurie, F., Yang, L., Jin, R., Sukthankar, R., Jurie, F.: Unifying discriminative visual codebook generation with classifier training for object category recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
18. Yang, J., Yang, M.: Top-down visual saliency via joint crf and dictionary learning. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
19. Vedaldi, A., Soatto, S.: Quick Shift and Kernel Methods for Mode Seeking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 705–718. Springer, Heidelberg (2008)
20. Kolmogorov, V., Zabih, R.: What Energy Functions Can Be Minimized Via Graph Cuts? IEEE Trans. on Pattern Analysis and Machine Intelligence 26, 147–159 (2004)
21. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. International Journal of Computer Vision 82, 302–324 (2009)
22. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. Journal of Machine Learning Research 6, 1453–1484 (2005)
23. Murphy, K.P., Weiss, Y., Jordan, M.I.: Loopy belief propagation for approximate inference: An empirical study. In: Proceedings of Uncertainty in AI, pp. 467–475 (1999)
24. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publishers Inc., San Francisco (1988)
25. Yu, C.N.J., Joachims, T.: Learning structural svms with latent variables. In: International Conference on Machine Learning, ICML (2009)
26. Opelt, A., Pinz, A.: The TU Graz-02 database (2002), `http://www.emt.tugraz.at/~pinz/data/GRAZ_02/`
27. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and Recognition Using Structure from Motion Point Clouds. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 44–57. Springer, Heidelberg (2008)
28. Zhang, H., Xiao, J., Quan, L.: Supervised Label Transfer for Semantic Segmentation of Street Scenes. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 561–574. Springer, Heidelberg (2010)
29. Floros, G., Leibe, B.: Joint 2D-3D temporally consistent semantic segmentation of street scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)