# Learning Shared, Discriminative, and Compact Representations for Visual Recognition

Hans Lobel, René Vidal, and Alvaro Soto

**Abstract**—Dictionary-based and part-based methods are among the most popular approaches to visual recognition. In both methods, a mid-level representation is built on top of low-level image descriptors and high-level classifiers are trained on top of the mid-level representation. While earlier methods built the mid-level representation without supervision, there is currently great interest in learning both representations jointly to make the mid-level representation more discriminative. In this work we propose a new approach to visual recognition that jointly learns a shared, discriminative, and compact mid-level representation and a compact high-level representation. By using a structured output learning framework, our approach directly handles the multiclass case at both levels of abstraction. Moreover, by using a group-sparse prior in the structured output learning framework, our approach encourages sharing of visual words and thus reduces the number of words used to represent each class. We test our proposed method on several popular benchmarks. Our results show that, by jointly learning mid- and high-level representations, and fostering the sharing of discriminative visual words among target classes, we are able to achieve state-of-the-art recognition performance using far less visual words than previous approaches.

**Index Terms**—Image Categorization, Dictionary Learning, Max-margin Learning, Structural SVMs, Group Sparsity.

## 1 Introduction

The success of recognition methods based on visual descriptors and off-the-shelf machine learning techniques [1], [2] is one of the main reasons for the new enthusiasm in computer vision technologies. These methods have shown robustness against visual complexities, such as changes in illumination, scale, affine distortions, and mild intraclass and pose variations. Unfortunately, more sophisticated visual complexities such as object deformations, partial occlusions, and severe intra-class and pose variations, require more elaborate solutions.

Many of these difficulties have been successfully overcome by methods that use a mid-level representation, such as parts [3] or dictionaries of visual words [4]. These representations model local areas of objects/scenes and exploit their spatial relations, which allows them to deal with deformations and occlusions. While it is still not clear what level of abstraction these representations should have and how such representations should be built, in this paper we advocate for three relevant properties that a mid-level representation should have.

First, the mid-level representation should be shared among class-level classifiers [5]. In fact, mid-level representations should serve as a shared and flexible set of building blocks that allow us to model a large variety of visual classes. This compositional view of visual perception is prevalent in several domains, such as language or our own visual system, and has been already noticed by early attempts in computer vision [6]. Furthermore, it is a key aspect to the scalability of a visual system.

Second, the mid-level representation should facilitate discriminative inference. In particular, dictionary words should favor the representation of discriminative visual patterns, like a wheel or a window, that are common only to some categories [7] so that high-level classifiers can more easily discriminate among the target classes.

Third, the mid-level representation should be compact, *i.e.*, it should avoid including redundant, misleading, or uninformative words. A mid-level representation with an overestimated budget size increases representational power by modeling fine-scale details. However, it also increases computational load, the risk of overfitting, and the rate of spurious responses, which can potentially impair classification. In contrast, a mid-level representation that is too small might be insufficient to represent large class variations. While preserving representational flexibility and discrimination capabilities, a suitable compact representation must provide computational efficiency by adapting its size to the complexity of the target problem.

However, it is difficult to learn compact dictionaries that are, at the same time, discriminative of each class and shared across classes, as these are conflicting goals. Indeed, without a mechanism that ensures a meaningful balance, we could easily fall in one of the following cases:

- A word/part that is shared by most of the categories: In this case, it is very likely that the word/part is not discriminative at all.
- A word/part that is too specific and appears in only one category: This case might seem desirable at first, but this specificity becomes a problem as the number of categories increases.

**Paper Contributions.** In this paper we propose a new approach to visual recognition that aims to learn shared, discriminative, and compact representations to boost not

- *H. Lobel and A. Soto are with the Department of Computer Science, Pontificia Universidad Católica de Chile. E-mail: halobel@uc.cl, asoto@ing.puc.cl*
- *R. Vidal is with the Center for Imaging Science, Department of Biomedical Engineering, The Johns Hopkins University. E-mail: rvidal@cis.jhu.edu*

*Manuscript submitted July 24, 2014; accepted February 05, 2015.*

only recognition performance but also efficiency. Our method is based on a mid-level dictionary of shared discriminative words that is learned jointly with high-level classifiers for each visual category. By learning a common mid-level representation among visual classifiers, we foster shared representations. By using a max-margin approach to jointly learn dictionary words and high-level classifiers, we obtain discriminative representations. Finally, by using a regularization that fosters group sparsity, we obtain compact representations at the mid and top-level layers. At the mid-level layer compactness is provided by the use of short size dictionaries, while at the top-level layer compactness is provided by word specialization where each category classifier uses only a subset of the available visual words. Consequently, the proposed method is able to adaptively balance classification complexity among the target classes, assigning a higher number of visual words to represent patterns in complex categories, leading naturally to compact representations for mid and top-level layers.

In terms of dictionary learning, we depart from the usual vector quantization [8] or sparse coding schemes [9] commonly used in Bag-of-Visual-Word (BoVW) models. Instead, we use linear SVMs to characterize each word, similar to [3], [10], [11], [12]. Moreover, we complement dictionary learning with a max-pooling strategy, as suggested in [9], [13]. In terms of classifier learning, we depart from the max-margin framework in [14], [15], which uses an $\ell_2$-regularizer on the classifier weights. Instead, we use a group-sparse regularizer, which encourages choosing very few words to represent each class, and very few classes to utilize each visual word.

Our experiments show that the proposed method learns visual words that are *shared* among classes according to the classification complexity of each class, where larger proportions of the visual words are used by categories displaying challenging visual variabilities. Moreover, the obtained visual words are highly *discriminative* in the sense that they are specialized to specific appearance patterns. Furthermore, we show that the proposed model leads to state-of-the-art performance in categorization tasks on many standard benchmarks, using an order of magnitude less words than previous approaches, thus leading to *compact* representations, which are critical for the scalability of recognition algorithms.

While prior work has highlighted the relevance of learning shared representations [5] and there is a vast literature about methods to obtain discriminative representations [16], the problem of learning compact representations has attracted less attention [15]. As far as we know, this is the first attempt to develop an explicit approach to jointly target these goals. Consequently, this work makes the following main contributions:

1) A group-sparse structured output learning method for jointly learning shared, discriminative, compact, mid-level dictionaries and high-level classifiers.
2) State-of-the-art categorization performance with a significant reduction in dictionary size and word usage with respect to previous approaches.

**Paper Outline.** The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 describes the proposed model; Section 4 describes our method for learning the model parameters; Section 5 presents qualitative and quantitative experiments on standard benchmarks; and Section 6 presents concluding remarks and possible future research directions.

## 2 Related Work

The proposed method is most closely related to two popular approaches to visual recognition, namely dictionary-based models and part-based models.

Dictionary-based models build a mid-level representation that corresponds to the output of a pooling scheme acting on a visual dictionary that encodes appearance information from small local image patches. Early approaches, such as BoVW, were based on vector quantization, generally using K-Means to cluster low-level local descriptors [8], [17]. Afterwards, several variations have been proposed using alternative quantization methods, discriminative dictionaries, or different pooling strategies [18], [19], [20], [21]. Additionally, spatial information has also been incorporated by concatenating BoVW representations from different local image areas and different scales [4]. Sparse coding schemes have also emerged as a powerful alternative to vector quantization, providing dictionaries that achieve lower reconstruction errors and attractive computational properties. In particular, [9] shows that a combination of sparse coding, spatial pyramidal decomposition, max-polling, and high dimensional linear SVM classifiers provide a powerful scheme to perform object and scene recognition.

One drawback of the aforementioned approaches is that the dictionary of visual words is learnt in an unsupervised manner and independently from the classifier parameters for each visual category. To address this problem, discriminative representations that jointly learn mid-level representations and class-level classifiers have also been proposed. In [16], the authors propose a discriminative joint learning procedure, based on sparse coding and linear or bilinear classifiers. While this work effectively addressed the problem of joint dictionary learning and classifier training, they use a completely different model and evaluation setting compared to ours. In [13], a dictionary is learnt also by means of discriminative sparse coding, using a joint max-margin formulation that explicitly includes a linear classifier. Unlike our work, this classifier is not used then for the final categorization of images, limiting the discriminative power of the approach. More similar to ours, [22] also presents a joint max-margin learning problem, where the classifier and the dictionary are effectively trained and then used in the final categorization. Despite the clear resemblance to our work, they model images using the standard BoVW formulation with a spatial pyramid, leading to a completely different problem formulation, with a different cost and a different optimization problem. In [23], the authors take advantage of inter-object visual correlation, using a joint discriminative optimization framework based on the Fisher criterion, to learn a common dictionary for each group of visually similar categories and a set of category-specific dictionaries. A critical difference with our work is that they use a clustering step previous to dictionary learning, to generate the class groupings. On the other hand, by using a group-sparse regularizer, we are also able to capture visual correlations between classes, using only one common dictionary. In particular, none of the above mentioned work has targeted the problem of learning compact representations or using group sparse regularizers to adaptively handle model complexity in terms of the dictionary.

Joint learning has also been explored in other visual recognition scenarios. In terms of attribute discovery, [24] proposes a method where visual attributes and low-level features are integrated to learn a common feature space. We share some similarities with this work, namely the use of a joint loss function and the use of a group-sparse regularizer. In [25], the authors used joint learning in the context of grouping categories that share similarity metrics. This context is very different from ours and is not directly comparable in terms of joint dictionary learning.

In the case of part-based models, the mid-level representation corresponds to basic semantic visual structures that can usually be mapped to relevant object components. Common strategies to obtain these parts are manual selection [26], greedy latent models [3], or the output of a large set of part-based classifiers trained using a costly labeling process [27]. Spatial information is also incorporated into the models by learning common spatial configurations among parts [3]. After the seminal work in [3], latent models have been used to jointly learn parts and object classifiers under a common optimization scheme that maximizes object classification performance. Lately, [10] proposed an extension to [3] that directly considers the multiclass classification case, but in the context of an action recognition application.

Recently, there has been an increasing interest to develop semi-supervised methods for part/word discovery that avoid the part-labeling restrictions in [27]. [12] proposes an automatic method to discover relevant parts/words under the assumption that only the image category label is known. Each part detector is modeled using a linear SVM and an iterative process refines these detectors using a heuristic approach. Performance evaluation shows state-of-the-art results for a scene classification task, however, the training process has to be conducted carefully to avoid convergence to a poor solution. To alleviate this problem, [7] generates the initial set of parts using exemplar SVMs [28]. Afterwards, an efficient retraining procedure allows them to outperform the results in [12]. In [29], a Latent SVM model and a group sparse regularizer are used to discover and train the parts in a more principled way. Finally [30] improves upon [12] by posing the discovery of parts as a discriminative mode seeking problem using the mean-shift algorithm. In contrast to our approach, these methods do not learn parts and class level classifiers jointly. As we show in this work, joint learning leads to more efficient mid-level representations.

In terms of hierarchical compositional models, our work is related to recent recognition approaches based on deep belief networks (DBNs) [31], [32], [33]. In general, DBNs consist of a hierarchical compositional model that incorporates spatial pooling schemes and intermediate representations based on linear filters, that are similar to our work. DBNs are usually applied over a raw image representation using several layers of generic structures. As a consequence, DBNs have many parameters and they are usually difficult to train. In contrast, we embed semantic knowledge in our model by explicitly exploiting compositional relations among low-level visual features, visual words, and high-level classifiers. This leads to simpler architectures that are less complex to train and do not operate as a black box. We use this last feature to easily and explicitly add suitable regularization terms to our main objective function that foster shared, discriminative and compact representations. Furthermore, our objective function is based on a max-margin approach using a hinge loss and an $\ell_1/\ell_2$-norm group regularizer, and not a quadratic or a logistic function commonly used to train DBNs, leading to a very different optimization setup.

Our optimization framework shares some similarities with [34], [35] and [36]. In [34], the authors use a multiple kernel learning framework, while we use a max-margin framework which captures a structured representation, while in [35], a joint learning of regions and classifiers is performed, but not of the dictionary. In [36], a structured group-sparse regularizer is used to control the number of components of a DPM model, while we use it to control the number of words used per category.

Finally, the three main components of our method, i) a structured max-margin energy representation [11], ii) a group-sparse regularizer [36] and iii) max-pooling operator and latent representations [37], have been used before in the context of visual recognition. However, to the best of our knowledge, these components have never been used together in a joint formulation applied to image categorization, as we do. It is worth to mention that the combination of these components is not trivial, posing challenging modeling and optimization issues. Addressing these challenges is the core contribution of our work.

## 3 Proposed Model

This section describes our proposed model for visual recognition. Given a set of feature descriptors extracted from multiple image regions, we propose a mid-level representation based on a dictionary of linear classifiers, each one representing a visual word. These classifiers are applied to each feature descriptor and a max-pooling strategy is used to obtain the response for each region. The high-level representation is then obtained by applying linear classifiers to these responses.

### 3.1 Low-Level Image Representation

We use a simple low-level representation that consists of a set of feature descriptors extracted from a diverse set of local regions. This representation is motivated by previous work, which showed the relevance of including spatial information and a max-pooling scheme in the image coding stage. For instance, in the case of image classification, [4] improves the performance of the regular BoVW model (average pooling) by adding a coding scheme based on a spatial pyramid decomposition. This decomposition is generated by dividing the image into fixed rectangular regions at multiple spatial resolutions. Similarly, [9] shows that using a max-pooling scheme instead of average-pooling yields a substantial increase in performance on several image recognition tasks.

Recently, [35] shows that a random region selection generally boosts performance compared to pre-defined schemes, such as a spatial pyramid, when the dimensionality is the same. In our work, we follow this scheme and we depart from the restrictions of the regular spatial pyramid regarding its structure and amount of overlapping between regions. Following [35], we randomly define $L$ rectangular regions of any aspect ratio, allowing them to overlap with each other. Once defined, these regions are shared among all images. Then, given an image $i$ and a set of $L$ pooling regions, we extract multiple local visual features from each region, either centered at interest points or by using a dense sampling scheme. In particular, we use visual features given by a combination HOG and LBP descriptors [38].

### 3.2 Mid-Level Dictionary of Visual Word Classifiers

Our mid-level representation is obtained by encoding each feature descriptor with respect to a visual dictionary $\Theta$. Inspired by [11], we define this dictionary as

$$\Theta = [\theta_1 \ \theta_2 \ \theta_3 \ \dots \ \theta_K] \in \mathbb{R}^{(T+1)\times K}, \tag{1}$$

where $T$ is the dimension of the feature descriptor and each word $\theta_k$ represents a linear classifier with bias:

$$\theta_k = [\theta_{k,1}, \theta_{k,2}, \dots, \theta_{k,T}, b_k]^\top \in \mathbb{R}^{T+1}. \tag{2}$$

We encode a set of $N_{i_l}$ feature vectors $v^{i_l} = [v_1^{i_l}, \dots, v_{N_{i_l}}^{i_l}]$ extracted from a local region $l$ in image $i$ using a linear classification score (dot product) with each one of the visual words $\theta_k$ in $\Theta$, *i.e.*,

$$c_{\theta_k}(v^{i_l}) = [\langle v_1^{i_l}, \theta_k\rangle, \ \dots, \langle v_{N_{I_l}}^{i_l}, \theta_k\rangle] \in \mathbb{R}^{N_{i_l}}. \tag{3}$$

We then code image region $l$ by applying a max-pooling operator that selects for each visual word $\theta_k$, its highest response among all codes $c_{\theta_k}(v^{i_l})$ in region $l$, *i.e.*:

$$x_\Theta(i_l) = [\max_{j\in\bar{N}_{i_l}} c_{\theta_1}(v^{i_l}), \ \dots, \ \max_{j\in\bar{N}_{i_l}} c_{\theta_K}(v^{i_l})]^\top \in \mathbb{R}^K, \tag{4}$$

where we have used the notation $\bar{N} = \{1, \dots, N\}$. Finally, we build a low-level image descriptor for image $i$ by concatenating the descriptors of its $L$ regions, *i.e.*:

$$x_\Theta(i) = [x_\Theta(i_1), x_\Theta(i_2), \dots, x_\Theta(i_L)]^\top \in \mathbb{R}^{KL}. \tag{5}$$

A natural question that arises is what this encoding scheme provides. Intuitively, if the resulting visual words are discriminative and different from each other, descriptor $v^{i_l}$ should be similar to very few words in the dictionary. Furthermore, as we only keep the maximum response for each word inside each region, descriptor $x_\Theta(i)$ is effectively providing some level of robustness to local image translations of salient visual features.

It is important to note that in Equation (3), we assume that dictionary words $\theta_k$ act as linear SVM classifiers, *i.e.*, a response is given by the dot product between the classifier and a feature vector. High positive responses indicate high similarity, while low negative scores indicate low similarity. As we encode each region by applying a max-pooling operator, a negative value within the region code means that none of the local features are similar to the word associated to that dimension. Similarly to [9], we avoid negative scores for words that do not participate in the feature reconstruction. We achieve this by adding to each region a null feature vector $\vec{0}$, whose dot product with any of the dictionary words is zero.

### 3.3 Top-Level Object Classifiers

Given a descriptor for image $i$, $x_\Theta(i)$, we define the classification score, or energy function, for image $i$ as:

$$E(i, y, W, \Theta) = \quad w_y^\top x_\Theta(i). \tag{6}$$

Here, $w_y \in \mathbb{R}^{KL}$ represents the parameters of a classifier learnt for object class $y \in \{1, 2, \dots, Y\}$ and

$$W = [w_1 \ w_2 \ \cdots \ w_Y] \in \mathbb{R}^{KL\times Y} \tag{7}$$

represents all the object classifier parameters.

If $w_y$ is divided into $L$ sub-vectors of size $K$, each one assigned to a different region, we can rewrite the energy function in the following form:

$$E(i, y, W, \Theta) = \quad \sum_{l=1}^{L}\sum_{k=1}^{K} w_{y,l,k} \cdot \max_{j\in\bar{N}_{i_l}} \langle v_j^{i_l}, \theta_k\rangle, \tag{8}$$

where $w_{y,l,k}$ refers to the $k$-th element of the $l$-th sub-vector of $w_y$. This formulation makes explicit the fact that the total energy of an image is a linear combination of max functions. It can also be seen that the energy function shows a nonlinear dependence between the weights $w_y$ and the dictionary words $\theta_k$. Given the parameters of the classifiers for the different object categories, $W$, and the parameters of the classifiers for the different visual words, $\Theta$, we classify an image $i$ by:

$$y^* = \operatorname*{argmax}_y E(i, y, W, \Theta). \tag{9}$$

# 4 Proposed Learning Algorithm

## 4.1 Shared, Compact, and Discriminative Representations via Structured Norm Regularization

The model described in the previous section depends on two sets of parameters: visual words classifiers $\Theta$ and object classifiers $W$. In this section we propose an algorithm for learning both sets of parameters jointly. Specifically, given a set of training examples $\{i, y^i\}_{i=1}^N$, where $i$ refers to the $i$-th image and $y^i$ to its corresponding object class, we propose to find $\Theta$ and $W$ by solving the following regularized max-margin learning problem:

$$\begin{aligned} \min_{W,\Theta,\{\xi^i\}} \quad & C_W \Omega(W) + C_\Theta \Gamma(\Theta) + \frac{1}{N}\sum_{i=1}^{N} \xi^i \qquad (10)\\ \text{s.t.} \quad & E(i, y^i, W, \Theta) - E(i, y, W, \Theta) \geq \Delta(y^i, y) - \xi^i,\\ & \forall i \in \bar{N} \wedge \forall y \in \bar{Y}, \end{aligned}$$

where $\Omega$ and $\Gamma$ are convex regularizers and $C_W, C_\Theta > 0$ are regularization constants. The objective function in (10) fosters the construction of visual words that behave like linear SVMs, *i.e.*, classifiers that jointly maximize the margin and minimize the loss. Moreover, the set of constraints fosters image classification according to ground truth labels. Specifically, the value of the energy function $E(i, y^i, W, \Theta)$ for the ground true label $y_i$ should be higher than that for alternative classification labels $y \neq y_i$ by a margin given by the loss function $\Delta(y^i, y)$:

$$\Delta(y_1, y_2) = \begin{cases} 0 & \text{if } y_1 = y_2 \\ 1 & \text{otherwise} \end{cases}. \qquad (11)$$

Additionally, slack variables $\xi^i \geq 0$ provide a mechanism for a soft violation of these constraints.

To fully specify the formulation in (10), we need to select suitable regularizers for the parameters $W$ and $\Theta$ that lead to shared, discriminative, compact mid-level representations, and compact high-level representations.

In terms of the regularizer for parameter $\Theta$, as in [15], we use the squared $\ell_2$-norm: $\Gamma(\Theta) = \frac{1}{2}\|\Theta\|_F^2$. Following [15], this regularizer does not discard words a priori, increases discriminative power and reduces overfitting.

In terms of the regularizer for parameter $W$, we notice that the squared $\ell_2$-norm regularizer used in [15],

$$\Omega_1(W) = \frac{1}{2}\|W\|_F^2, \qquad (12)$$

penalizes all dictionary words equally. As will be shown in Fig 2, this encourages all the words to be shared across all classes. However, the resulting words may not be fully discriminative because they may not be specialized to recognize patterns appearing in a subset of the classes. Moreover, the representation may be compact with respect to the total number of words used, but not with respect to the number of words used by each class.

To obtain a representation that is both discriminative and compact, we propose to use a regularizer $\Omega(W)$ that penalizes the number of words used by each class independently. In this way, each class can select adaptively the words needed to perform categorization: simple classes can select few words and complex classes can select more words. To estimate the number of words used by class $y$, observe that if word $k$ contributes to the classification of instances from class $y$, then we must have $w_{y,l,k} \neq 0$ for some $l$, hence $\sum_l w_{y,l,k}^2 \neq 0$. Thus, the total number of words used by all classes is given by $\sum_y^Y \sum_k^K 1(\sum_l w_{y,l,k}^2 \neq 0)$. Since this is a non-convex function of $W$, we define the *contribution* of word $k$ to the classification of instances from class $y$ as:

$$C(y,k) = \sqrt{\sum_l w_{y,l,k}^2} \qquad (13)$$

and use it to approximate the total number of words used by the $\ell_1/\ell_2/\ell_1$-norm regularizer on $W$ defined as:

$$\Omega_2(W) = \sum_y^Y \sum_k^K C(y,k) = \sum_y^Y \sum_k^K \sqrt{\sum_l w_{y,l,k}^2}. \qquad (14)$$

Since our goal is to obtain a mid-level representation that is shared, discriminative and compact, we combine the $\ell_2$ and $\ell_1/\ell_2/\ell_1$ norms into the following regularizer:

$$\Omega(W) = (1-\alpha)\frac{1}{2}\|W\|_F^2 + \alpha \sum_y^Y \sum_k^K \sqrt{\sum_l w_{y,l,k}^2}\ , \qquad (15)$$

where the constant $\alpha \geq 0$ indicates the relative importance of the regularizers $\Omega_1$ and $\Omega_2$. This regularization strategy, which resembles the elastic net regularizer [39], brings three important benefits. First, it produces a compact representation, as it adaptively reduces the dimensionality of each category, something that is helpful to reduce the risk of overfitting. Second, words can *focus* only on some of the categories, generating a meaningful sharing and increasing their discriminability. Third, most of the words are not discarded by the model, keeping high the representational power of the dictionary.

## 4.2 Alternating Minimization with Latent Variables

Although the learning problem in (10) is similar to that for Structural SVMs (SSVMs) [40], existing techniques to train SSVMs are not applicable because the constraints in (10) are non-linear in the parameters $(W, \Theta)$. Moreover, block-coordinate optimization strategies that alternate between solving for $W$ and $\Theta$ can not be directly applied because, when fixing $W$ and solving for $\Theta$, the resulting constraints are still not linear in the parameters.

To tackle this issue, we reformulate the learning problem in (10) using latent variables to avoid the nonlinearity introduced by the max-pooling operator. If we recall, the descriptor of a region $l$ in image $i$ is given by (4). We modify this expression by removing the *max* operator and adding a set of latent variables $z = \{z_{(l,k)}\}$, for $l \in \bar{L} \wedge k \in \bar{K}$. Here, $z_{(l,k)}^i$ is the index of the descriptor extracted from region $l$ in image $i$ with maximum response to the application of word $\theta_k$, *i.e.*:

$$z_{(l,k)}^i = \underset{j \in \bar{N}_{i_l}}{\operatorname{argmax}} \langle \theta_k, v_j^{i_l} \rangle. \qquad (16)$$

As it can be seen, the latent variables serve as proxies for the *max-pooling* operation. To simplify the notation, we will denote $\hat{v}^i_{l,k} = v^{il}_{z^i_{(l,k)}}$ and $v^i_{l,k} = v^{il}_{z_{(l,k)}}$. Using these, Eq. (8) becomes:

$$E(i, y, W, \Theta) = \sum_{l}^{L} \sum_{k}^{K} w_{y,l,k} \langle \theta_k, \hat{v}^i_{l,k} \rangle. \tag{17}$$

Based on this energy formulation, we can now state the problem in a form similar to Eq. (10):

$$\begin{aligned} \min_{W,\Theta,\{\xi^i\}} \; & C_W \Omega(W) + C_\Theta \Gamma(\Theta) + \frac{1}{N} \sum_{i=1}^{N} \xi^i \\ \text{s.t.} \quad & \sum_{l}^{L} \sum_{k}^{K} w_{y^i,l,k} \langle \theta_k, \hat{v}^i_{l,k} \rangle - \\ & \sum_{l}^{L} \sum_{k}^{K} w_{y,l,k} \langle \theta_k, v^i_{l,k} \rangle \geq \Delta(y^i, y) - \xi^i, \\ & \forall i \in \bar{N} \wedge \forall y \in \bar{Y} \wedge \forall z_{l,k} \in \bar{N}_l. \end{aligned} \tag{18}$$

This new problem is now similar to a Latent Structural SVM (LS-SVM) [41], but the constraints are still non-linear in $(W, \Theta)$. Nonetheless, if we fix $W$, the constraints become linear in $\Theta$ and vice versa. Thus, we can solve it efficiently using alternating minimization.

More specifically, notice that Eq. (18) can be rewritten as two different unconstrained problems. Fixing $\Theta$, we obtain the following optimization problem over $W$:

$$\begin{aligned} \min_{W} \; & C_W \Omega(W) + \\ & \frac{1}{N} \sum_{i=1}^{N} \max_{y,z} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{y,l,k} \langle \theta_k, v^i_{l,k} \rangle + \Delta(y^i, y) \\ & - \frac{1}{N} \sum_{i=1}^{N} \max_{z^i} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{y^i,l,k} \langle \theta_k, \hat{v}^i_{l,k} \rangle. \end{aligned} \tag{19}$$

Likewise, fixing $W$, we obtain the following optimization problem over $\Theta$:

$$\begin{aligned} \min_{\Theta} \; & C_\Theta \Gamma(\Theta) + \\ & \frac{1}{N} \sum_{i=1}^{N} \max_{y,z} \sum_{k=1}^{K} \langle \theta_k, \sum_{l=1}^{L} w_{y,l,k} v^i_{l,k} \rangle + \Delta(y_i, y) \\ & - \frac{1}{N} \sum_{i=1}^{N} \max_{z^i} \sum_{k=1}^{K} \langle \theta_k, \sum_{l=1}^{L} w_{y^i,l,k} \hat{v}^i_{l,k} \rangle. \end{aligned} \tag{20}$$

The structure of the above optimization problems is very similar to that of the LS-SVM problem, which can be solved using the CCCP algorithm [42]. This algorithm is designed for problems whose objective can be decomposed as the sum of a convex and a concave term. It proceeds by iterating between the optimization of the concave and the convex parts leading to a local minimum or saddle point. In our case, if we follow the steps of the CCCP algorithm, the estimation of the latent variables reduces to the following problem:

$$z^i = \operatorname*{argmax}_{z} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{y^i,l,k} \langle \theta_k, v^i_{l,k} \rangle. \tag{21}$$

Eq. (21) shows that the local feature vector selected for region $l$, image $i$, and dictionary word $\theta_k$, depends on the sign of $w_{y^i,l,k}$ and the value of the inner product $\langle \theta_k, \hat{v}^i_{l,k} \rangle$. Now, since one of the purposes of the latent variable $z$ is to induce the max-pooling behavior on dictionary word responses inside each region, the value of each $z^i$ should not depend on the sign of $w_{y^i,l,k}$. To avoid this problem, we enforce the non-negativity constraints:

$$w_{y,l,k} \geq 0, \forall y, l, k. \tag{22}$$

In this way, the value of the inner product $\langle \theta_k, v^i_{l,k} \rangle$ will only be scaled by $w_{y,l,k}$, thus preserving the semantics of max-pooling. With this assumption, $z^i$ depends only on $\Theta$, thus making it unnecessary to update the latent variables after recomputing $W$.

In summary, we propose to solve the learning problem in (18) by alternating between the following two steps until the energy defined by (18) stops decreasing:

**Estimation of $W$.** Given fixed values of $\Theta$ and $\{z^i\}$, solve the following convex minimization problem

$$\begin{aligned} \min_{W} \; & C_W \Omega(W) + \\ & \frac{1}{N} \Big( \sum_{i=1}^{N} \max_{y} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{y,l,k} \langle \theta_k, \hat{v}^i_{l,k} \rangle + \Delta(y^i, y) \Big) \\ & - \frac{1}{N} \sum_{i=1}^{N} \sum_{l=1}^{L} \sum_{k=1}^{K} w_{y^i,l,k} \langle \theta_k, \hat{v}^i_{l,k} \rangle \\ \text{s.t.} \quad & w_{y,l,k} \geq 0, \forall y, l, k. \end{aligned} \tag{23}$$

This is a max-margin multiclass learning problem with a structured norm and non-negativity constraints on $W$. To solve this problem, we need a procedure that can handle large visual dictionaries. Thus the optimization algorithm must be fast and with low memory requirements. Given these constraints, the L-BFGS quasi-Newton method [43] is an attractive option. This algorithm has been successfully used in visual recognition problems [44], and it is able to handle non-convex and non-smooth optimization problems [45]. To obtain a suitable search direction, the L-BFGS algorithm estimates the inverse of the Hessian matrix by means of an implicit reconstruction. This is based on the values of the solution and its gradient during the last iterations, allowing it to work with a very limited amount of memory.

**Estimation of $\Theta$.** Repeat the following two steps until objective function defined in (20) stops decreasing:

- Given fixed values of $W$ and $\{z^i\}$, solve the following convex minimization problem:

$$
\begin{aligned}
&\min_{\Theta} \; C_{\Theta}\Gamma(\Theta) + \qquad (24)\\
&\quad \frac{1}{N}\Big(\sum_{i=1}^{N}\max_{y,z}\sum_{k=1}^{K}\langle\theta_k,\sum_{l=1}^{L}w_{y,l,k}v^{i}_{l,k}\rangle + \Delta(y_i,y)\Big)\\
&\quad -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\langle\theta_k,\sum_{l=1}^{L}w_{y^i,l,k}\hat{v}^{i}_{l,k}\rangle.
\end{aligned}
$$

- Given a fixed value of $\Theta$, compute for each example the optimum value for its latent variables $z^i$ as:

$$z^i = \operatorname*{argmax}_{z}\sum_{l=1}^{L}\sum_{k=1}^{K}\langle\theta_k, v^{i}_{l,k}\rangle. \qquad (25)$$

This is a LS-SVM, which we solve using the CCCP algorithm [41], as described in [15].

Although the convergence to a local minimum or saddle point of the block coordinate descent method cannot be guaranteed theoretically [46], experimentally we found that for a suitable selection of the regularization constants $C_W$, $C_\Theta$ and $\alpha$, our procedure does converge. In practice, we repeat the three steps of the proposed algorithm until the ratio of energy reduction between iterations is lower than a fixed threshold.

# 5 Experimental evaluation

This section presents a qualitative and quantitative evaluation of our method with respect to three aspects. First, we evaluate the effect of the group-sparse regularizer $\Omega$ by analyzing the word contributions, the effective number of words used after training and the amount of word sharing. Second, we visually analyze the effect of jointly training the dictionary and classifiers by inspecting the activations of the dictionary words. We evaluate them with respect to discriminability, specialization and sharing. Finally, we analyze the categorization performance of our method on 4 different datasets. We evaluate the relation between performance and dictionary size and end with a comparison between our method and other alternative state-of-the art techniques.

## 5.1 Implementation Details

**Pooling Regions and Feature Extraction.** We downsize each image to no more than 300 pixels in each direction, keeping the aspect ratio untouched. Then, we normalize image coordinates to be in the range [0-1] and randomly define $L$, possibly overlapping, rectangular regions with width and height in the range [0.25-1.0]. For each evaluation performed, we use 30 regions, based on the results of [15]. Finally, we extract local HOG+LBP features [38] from each image over a dense grid of cells of $16 \times 16$ pixels, with a stride of 8 pixels in each direction.

**Initial Dictionary.** We sample between 70 and 100 local feature vectors per training image, and cluster them using the K-Means algorithm. A linear SVM is trained for each cluster, using as positive examples the ones belonging to that cluster, and as negative examples a random sample of features from other clusters.

**Dataset Details.**

- *15 scene categories* [4]: This dataset contains images of 15 natural scene categories. We use 5 random splits of the data, using 100 images per class for training and the rest for testing.
- *Caltech101* [47]: This dataset contains 102 object categories (101 objects and background). We use 5 random splits of the data, using 30 images per class for training and the rest for testing.
- *UIUC-Sports* [48]: This dataset contains scenes of 8 sport events. We use 5 random splits of the data, using 70 images per class for training and 60 for testing.
- *MIT67 Indoor* [49]: This dataset contains 67 indoor scene categories. We use the standard evaluation procedure, using 80 images per class for training and 20 for testing.

## 5.2 Effect of the Structured Norm Regularizer

**Effect of the Structured Norm Regularizer.** We start this analysis, by measuring the word contribution per category, $C(y,k)$. Figure 1 shows the word contributions for the *MITcoast* category of the *15 Scene Categories* dataset using two different regularization configurations, $\alpha = 0$ and $\alpha = 0.1$. In both cases we use the same initial dictionary composed of 1000 words and sort the contributions in decreasing order. As it can be seen in Figure 1b, the distribution of word contributions shows a more abrupt slope compared with Figure 1a. Moreover, it clearly shows that fewer words are used.

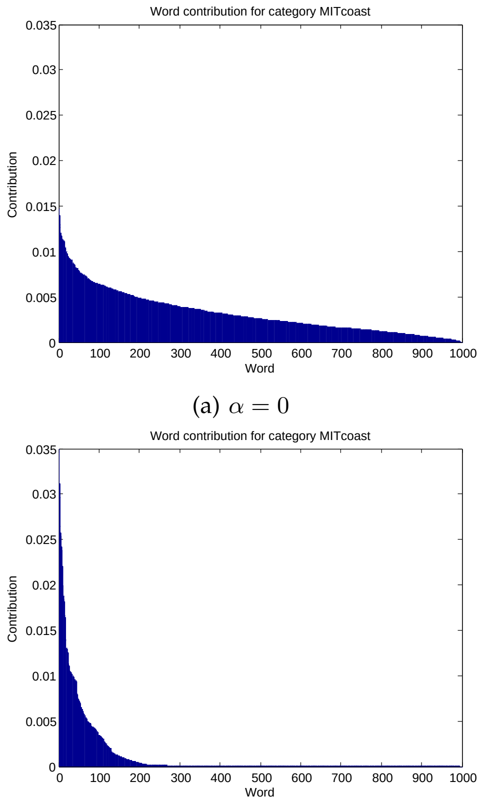
**Effective Dictionary Size.** As discussed in Section 4.1, the structured norm regularizer should encourage the use of fewer visual words to describe each category. To assess this numerically, we need a criterion to count the number of words used by the categories. We assume that, for a given category $y$, words are sorted in a descending order based on its contribution, *i.e.*, $C(y,k_1) \geq C(y,k_2), \forall\, k_1 \leq k_2 \in \bar{K}$. Given this, we define the effective number of words used by a category $y$ as:

$$\tilde{K}^{\tau}_{y} = \min\Big\{k : \frac{\sum_{i=1}^{k}C(y,i)}{\sum_{i=1}^{K}C(y,i)} \geq \tau\Big\}, \qquad (26)$$

where $\tau \in [0,1]$. Given a fixed value of $\tau$, we define the effective number of words used by a model as:

$$\tilde{K} = \sum_{y=1}^{Y}\frac{\tilde{K}^{\tau}_{y}}{Y}. \qquad (27)$$

To select a suitable value for $\tau$, we look for the smallest value $\tau^*$ such that the performance in the training set does not decrease. We measure performance as the average hit rate, *i.e.*, the mean of the diagonal of the confusion matrix. In practice, starting from $\tau = 1$, we reduce



(a) $\alpha = 0$

(b) $\alpha = 0.1$

Fig. 1: Contribution of word $k$ to the categorization of class $y =$ *MITcoast* as measured by $C(y, k)$ in (13). When $\alpha = 0$, the $\ell_2$ regularizer encourages most of the words to be used. On the contrary, when $\alpha = 0.1$, the group sparse term encourages the use only a few words. This trend repeats for every category in every analyzed dataset.

| **UIUC-Sports** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $K$ | 50 | 200 | 400 | 600 | 800 | 1000 | 1200 |
| $\tilde{K}$ | 40±1 | 121±11 | 169±12 | 205±8 | 223±11 | 236±8 | 235±10 |
| min $\tilde{K}_y$ | 32±2 | 92±9 | 127±9 | 151±9 | 169±11 | 171±5 | 173±26 |
| max $\tilde{K}_y$ | 43±1 | 141±15 | 196±17 | 254±27 | 285±16 | 316±25 | 307±40 |
| Discarded | 0±0 | 11±7 | 54±4 | 117±11 | 188±21 | 286±11 | 437±19 |

| **Caltech 101** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $K$ | 50 | 200 | 400 | 600 | 800 | 1000 | 1200 |
| $\tilde{K}$ | 39±0 | 106±1 | 179±2 | 218±10 | 248±8 | 272±5 | 277±8 |
| min $\tilde{K}_y$ | 25±2 | 56±5 | 98±5 | 91±4 | 114±8 | 122±19 | 114±14 |
| max $\tilde{K}_y$ | 45±1 | 136±3 | 233±11 | 299±13 | 370±36 | 431±20 | 442±32 |
| Discarded | 0±0 | 0±0 | 4±1 | 15±3 | 32±4 | 51±4 | 91±5 |

| **15 Scene Categories** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $K$ | 50 | 200 | 400 | 600 | 800 | 1000 | 1200 |
| $\tilde{K}$ | 35±1 | 93±8 | 140±10 | 179±7 | 192±9 | 206±9 | 207±10 |
| min $\tilde{K}_y$ | 22±2 | 45±2 | 67±9 | 98±5 | 101±5 | 110±7 | 100±15 |
| max $\tilde{K}_y$ | 43±1 | 121±10 | 199±13 | 257±17 | 288±20 | 328±50 | 323±36 |
| Discarded | 0±0 | 4±2 | 19±4 | 54±7 | 98±13 | 160±21 | 259±26 |

| **MIT 67** | | | | | | | |
|---|---|---|---|---|---|---|---|
| $K$ | 50 | 200 | 400 | 600 | 800 | 1000 | 1200 |
| $\tilde{K}$ | 43 | 116 | 208 | 241 | 286 | 326 | 328 |
| min $\tilde{K}_y$ | 30 | 68 | 134 | 148 | 156 | 168 | 171 |
| max $\tilde{K}_y$ | 49 | 151 | 281 | 343 | 390 | 515 | 521 |
| Discarded | 0 | 0 | 0 | 9 | 20 | 30 | 56 |

TABLE 1: Effective number of visual words $\tilde{K}$, minimum and maximum number of words used per class $y$, and number of discarded words as a function of the dictionary size $K$ for four different datasets.

the value in steps of $0.005$, until we see a performance drop larger than or equal to $0.1$ in the training set.

Table 1 presents, the effective number of visual words $\tilde{K}$ used by the method, the minimum and maximum number of words used per class $y$, and the number of discarded words, *i.e.*, not used by any class, as a function of the dictionary size $K$. Results are presented for each of the four datasets and are the average of the different training/testing splits runs, except for *MIT 67*, which has a fixed training/testing split. As it can be observed, the effective number of dictionary words, as well as the minimum and the maximum, stabilize in each of the four datasets, when using dictionaries larger than 1000-1100 words. This indicates that there is a limit, after which there is no more information to be extracted from a dictionary. Also, as expected, the number of discarded words keeps increasing with the size of the dictionary.

Although a stable point is achieved for each dataset, the effective number of words used is different in each case. As expected, datasets with fewer categories, such as *UIUC-Sports* and *15 scene categories*, need less words than datasets with more categories, such as *MIT 67* and *Caltech 101*. This makes sense as, intuitively, the dimensionality needed to discriminate has a direct relationship with the complexity of data.

**Word Sharing.** While the numbers presented in Table 1 clearly indicate that each category uses a small fraction of the visual words, they do not give information about how the words are being used, *i.e.*, how many categories use a specific word for classification. To analyze this, we use again the word contribution, $C(y, k)$. Here, we assume that, for a given word $k$, categories are sorted in descending order, based on their contribution, *i.e.*, $C(y_1, k) \geq C(y_2, k), \forall\ y_1 \leq y_2 \in \bar{Y}$. Given this, we define the number of categories that use word $k$ as:

$$\tilde{Y}_k^{\tau} = \min\Big\{ y : \frac{\sum_{i=1}^{y} C(i,k)}{\sum_{i=1}^{Y} C(i,k)} \geq \tau \Big\}. \tag{28}$$

We select the value of $\tau$ as $\tau = \tau^*$ as described before.

Figure 2 shows an example of the number of categories that use a given visual word on the *15 scene categories* dataset, with a dictionary of 1000 words, using $\alpha = 0$ and $\alpha = 0.1$. Words are sorted in descending order, based on the number of categories in which they participate.

As it can be seen in Figure 2, when the group-sparse term is not present, approximately 80% of the words are used by ten or more classes, representing a very low specialization level (on average, each word is used

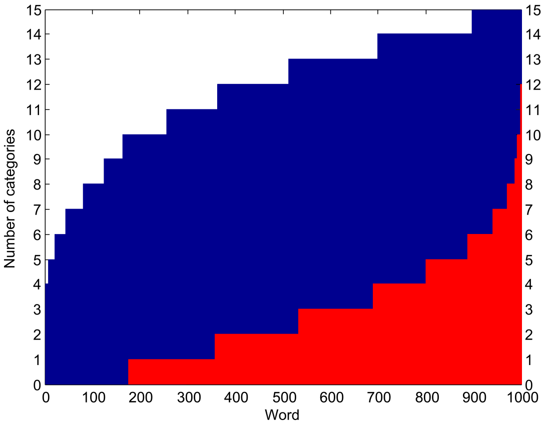


Fig. 2: Number of classes of dataset *15 scene categories* in which word $k$ participates. When $\alpha = 0$ (blue), most words are used by many classes. When $\alpha = 0.1$ (red), each word is used by very few classes.

on 11.8 classes). On the contrary, if the group-sparse term is present, we observe that around 55% of the words are used on 1-3 categories, while around 15% are discarded. If we restrict only to the non discarded ones, then each word is used, on average, in 3.2 classes. This supports the fact that our method tends to select a small group of more specialized words, thus increasing the discriminative power of the words.

### 5.3 Visualization of Dictionary Word Activations

**Discriminative Words.** One of the expected effects of the estimation of $\Theta$, is the increase of discriminative power of the dictionary. In practice, this means that, after the training process, the top activations of words should be more precise, consistent and get higher scores in areas similar to the visual pattern represented by the word. Figure 3 shows the evolution of the top activations of a visual word, in an image of the class *MIThighway* of the *15 scene categories* dataset. In this case, we observe a clear increase in the activation score (circles with larger radii). Moreover, top activations appear in places that seem to be more discriminative and consistent (clouds). As it will be shown later, this increase in discriminative power is one of the factors that allows our method to improve categorization performance and reduce the effective dictionary size, when compared to the case where classifiers and dictionary are not jointly trained.

**Specialized and Shared Words.** We will now show activations of both specialized and shared words and analyze them in terms of the visual patterns they capture. To measure the level of specialization of a word, we compute the specialization ratio, $S(k)$, defined as:

$$S(k) = \frac{\tilde{Y}_k^{\tau}}{\tilde{Y}_k^{\tau^\star}}, \tag{29}$$

where $\tau^\star$ is the optimal value for $\tilde{Y}_k^{\tau}$, as described before. Intuitively, if $\tau$ is sufficiently small, the numerator of $S(k)$ will count the number of categories for which word $k$ is highly important. Thus, by computing the ratio between this value and the number of times word $k$ is used by a class, we obtain a measure on how specialized the word is. This allows us to focus only on words that are highly discriminative and representative of the classes in which they participate. In practice, we found that $\tau = 0.4$ gives satisfactory results on the experiments.

We first focus on words that are highly specialized in only one category. Figure 4 shows the activations of a word that is highly specialized in the *bedroom* category of the *15 scene categories* dataset. For the images in Figures

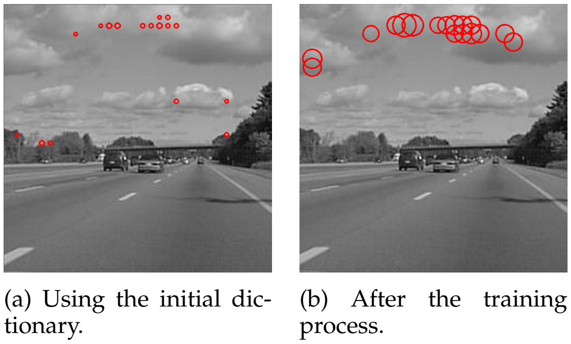

(a) Using the initial dictionary.
(b) After the training process.

Fig. 3: Activations of the word are marked with red circles, with the radii being proportional to the activation score. Using the initial dictionary, activations tend to appear on the clouds, but also on some trees. After the training process, the same word shows higher activation scores and appears only on the clouds.

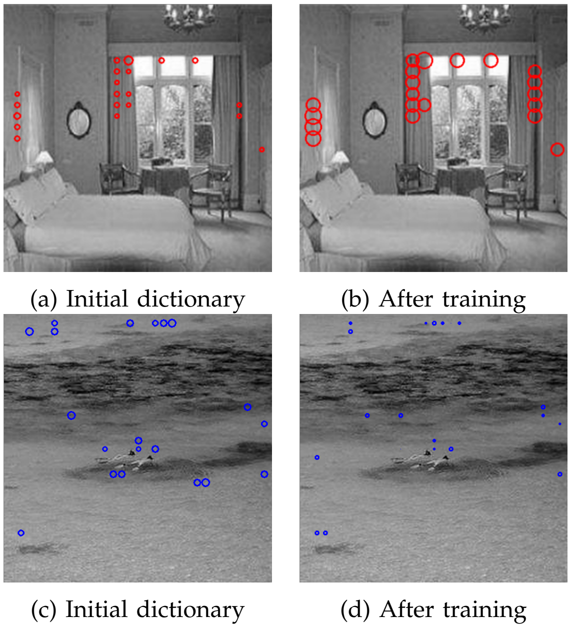
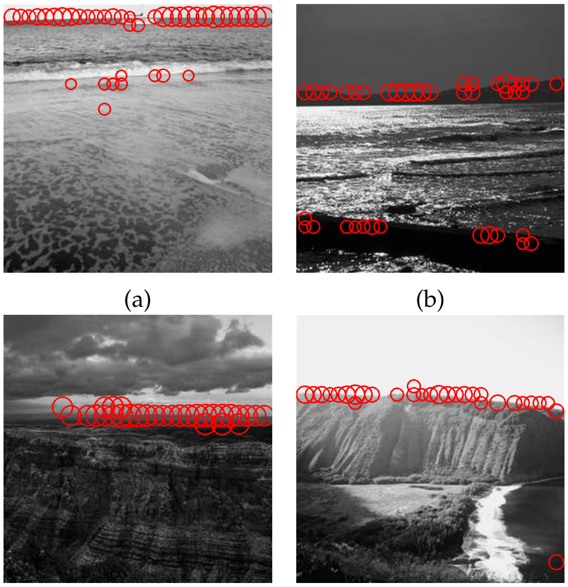

(a) Initial dictionary
(b) After training
(c) Initial dictionary
(d) After training

Fig. 4: Activations of a word specialized in the bedroom category. Figures 4a and 4b show activations for category *bedroom* before and after training, respectively. Figures 4c and 4d show activations for category *MITcoast* before and after training, respectively. Red circles indicate a positive response, while blue circles indicate a negative response.



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

Fig. 5: Activations of a word specialized in multiple categories. This word seems associated to horizontal edge patterns such as the horizon, a pattern shared by categories *MITcoast* (5a,5b) and *MITopencountry* (5c,5d).

4a and 4b, which correspond to the *bedroom* category, the word initially shows consistent responses to vertical patterns. As this seems to be highly characteristic of this category, after updating the dictionary our algorithm increases the contribution of the word to that category. On the other hand, for the images in Figures 4c and 4d, which correspond to the *MITcoast* category, the same word shows initially very low responses. In this case, our method reduces the contribution of the word to this category. In summary, our algorithm is able to learn visual words that are specialized to a single class.

The second case we analyze is very interesting from a visual point of view, as it presents a word that is highly specialized in more than one category. This allows us to observe highly discriminative and characteristic visual patterns, that are shared only by some of the categories. Figure 5 gives an example of a word that shows strong activations on horizontal edge patterns, like the ones visible on scenes containing the horizon. By analyzing the specialization ratio of this word, we find that it is highly specialized in 2 categories, *MITcoast* and *MITopencountry*. These two categories contain multiple images where the horizon is a main element of the scene. Besides showing strong activations on the horizon, there are very few activations outside that region. This shows that the specialization of the word, is able to discriminate very specific visual patterns without many false activations.

**Discarded Words.** There are cases were the contribution of a visual word to all image categories is very low, due to bad quality cluster generated by k-Means. When this happens, the word get progressively shrank to zero, which means that it is completely discarded from the model. In Figure 6, we present a case where a dictionary word shows poor discriminative power from the beginning, activating everywhere on images, without a clear pattern. In that case, the contribution for each class will be near to zero, eventually leading the algorithm to discard the word completely, *i.e.*, zeroing the word.

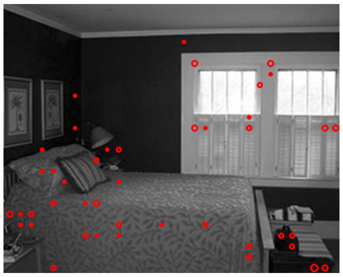


Fig. 6: A word with low activation for all classes. The strongest activations of the word show no clear pattern.

## 5.4 Performance Evaluation

**Regularization Constants.** The regularization constants, $C_W$ and $C_\Theta$, are obtained by performing 5-fold cross-validation on all four datasets. For each value of $C_W$, we restrict the search for the value of $C_\Theta$ based on the norm of the first estimation obtained for $W$. We found that a suitable rule to set the initial value of $C_\Theta$ is given by:

$$C_W \cdot \Omega(W) \approx C_\Theta \cdot \Gamma(\Theta) \qquad (30)$$

Eq. (30) seeks to balance between the two regularizers, as both estimations, $W$ and $\Theta$, are linked by the same loss. If one of the regularizers is more aggressive than the other by a large margin, the reduction of the norm of that parameter will be more important than learning, thus harming the performance of the whole system.

Regarding the constant $\alpha$, we test values in the range $[0, 1]$, with a step size of $0.1$. Empirically, we found that for $\alpha \approx 1$, the regularizer $\Omega$ is dominated by the group sparse term, resulting in overfitting problems, probably due to the non-smoothness of the objective function and the lack of generalization power of the group sparse norm alone. A similar problem is appreciated in standard $\ell_1$-norm SVMs, where the common solution is to use the elastic net regularizer [39]. Again, we empirically observe that a balance is reached between the two terms of $\Omega(W)$, when the value of $\alpha$ is around $0.1$.

**Classifier post-processing.** As shown before, the group-sparse regularizer provides a high degree of word specialization. However, this restriction over the energy function implies that the top-level classifier is forced to operate over a limited number of words. To improve this situation, after the training procedure is finished, we perform a retraining of the top-level classifier, using only the $\ell_2$-norm regularizer, and keeping the mid-level dictionary fixed. Our intuition is that, during the joint training step, the group-sparsity constraint plays its role to foster class specialization among the mid-level words.

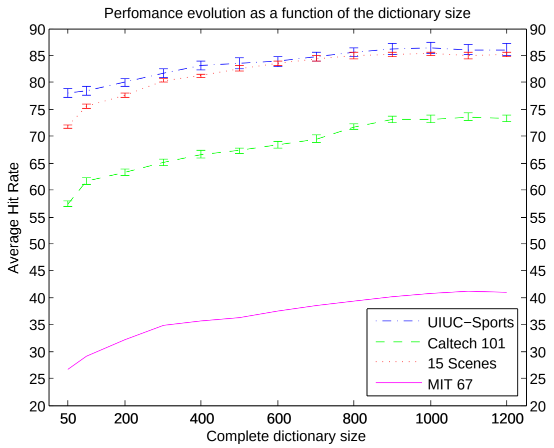


Fig. 7: Performance of the proposed approach as a function of dictionary size for different datasets. Near 1000-1100 words, our method achieves a stability point, where performance remains almost constant and sees no benefit from adding more words.

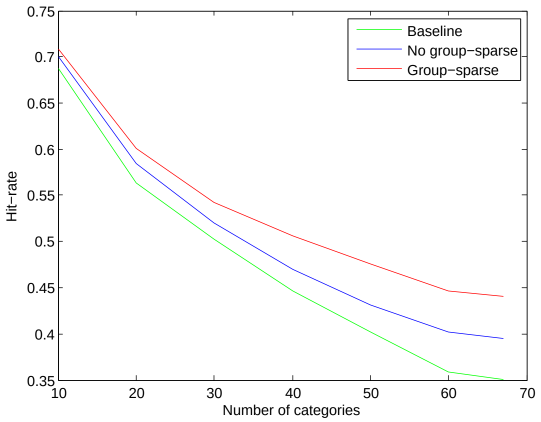


Fig. 8: Performance of the proposed approach ($\alpha = 0.1$), the method of [15] ($\alpha = 0$) and a baseline method with no joint optimization ($\alpha = 0.1$, $C_\Theta = 0$), as a function of the number of categories of the MIT67 dataset.

Afterwards, by removing this regularizer, the top-level classifier can capture new relations that were not possible before. In the following sections, we make explicit when the results use the post-processing step.

**Dictionary Size and Performance.** Following the experiments presented in Table 1, we explore categorization performance as we increase dictionary size from $50$ to $1200$ words. Figure 7 shows the average hit rate in the four datasets. As expected, there is a gradual improvement in performance as the dictionary size increases and performance eventually stabilizes. This marks a clear difference with the method of [15], where performance drops after the maximum is achieved due to the overfitting effect introduced by using more dictionary words. Although the highest performance of that method is reached with a smaller dictionary, the effective number of words used here is comparable to that number in all four cases. This indicates that the dictionary words obtained here are more discriminative.

**Impact of increasing the Problem Complexity.** The higher discriminativity and specificity of the mid-level words provided by our method should positively affect its capability to handle the complexity of a target problem. To quantify this issue, we compare in terms of recognition accuracy, the effect of increasing the number of categories in the *MIT67* dataset. We consider in our evaluation our method and two baselines: the method in [15] and a variant of [15] that does not perform joint optimization. Results are presented in Figure 8, using post-processing after training. As expected, as the number of categories increases, the recognition accuracy of the proposed method is far less affected than the two baselines. This result shows a relevant advantage of the proposed model, in particular the effect of the group sparsity constraint to manage the complexity of the target problem.

**Class complexity and dictionary size.** As each category adaptively selects the amount of words it uses, it is possible to study, for each of them, the relation between performance and number of words used. Intuitively, one would expect a negative correlation, *i.e.*, the easier it is to classify a category, the fewer words are needed, and vice versa. To analyze this, we define for each class $y$, the normalized number of words used as follows:

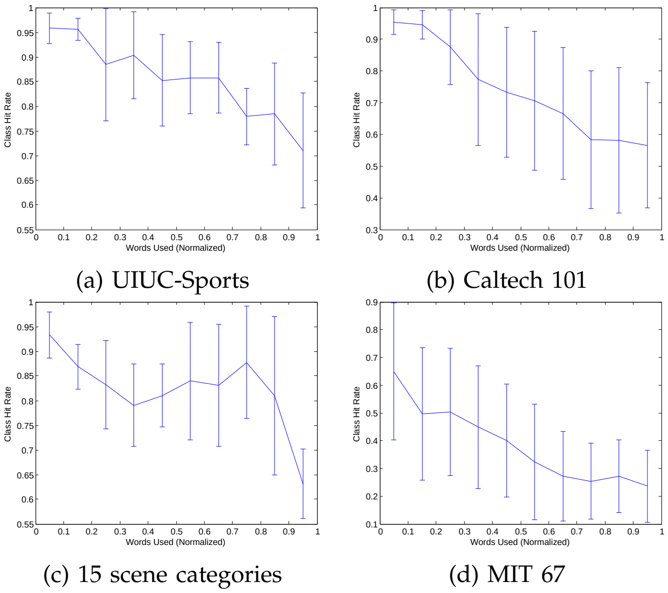
$$\tilde{K}_y^\star = \frac{\tilde{K}_y - \min_y \tilde{K}_y}{\max_y \tilde{K}_y - \min_y \tilde{K}_y} \in [0, 1]. \quad (31)$$

Notice that the class that uses the fewest words receives a 0, and the one that uses the most words receives a 1.

Given this, for each possible dictionary size $K$, we have $Y$ pairs $(p_y, \tilde{K}_y^\star)$, where $p_y$ is the hit-rate for class $y$. In practice, we generate pairs by increasing total dictionary size from $50$ to $1200$ words. To visualize this information in a meaningful way, for each of the four datasets, we divide the interval of the normalized amount of words, $[0, 1]$, in $10$ regular segments. For each segment, we plot in Figure 9 the average hit rate and the standard deviation of points lying in the segment. Although the standard deviation is large in some cases, we can clearly see that categories with higher performance tend to use less words than categories with lower performance, which is the intuitive behavior we expected. Although this behavior is less clear on *15 scene categories* and *UIUC-Sports*, this is something to be expected, as those datasets have very few categories.

**Categorization Performance Comparison.** The next experiment compares our results against alternative methods based on intermediate representations. We divide these methods in two different types:

- Small patches: These methods use as visual words features extracted from patches of small size, e.g., $16 \times 16$ pixels. Words are generally described by only one cell of local feature descriptors, like HOG [2] or



(a) UIUC-Sports

(b) Caltech 101

(c) 15 scene categories

(d) MIT 67

Fig. 9: Class performance (hit rate) as a function of the effective number of words used. Classes with lower complexity (higher hit rate), tend to use less visual words than classes with higher complexity (lower hit rate)

SIFT [50], thus they are low-dimensional.

- Large patches (Parts): These methods use features from patches of different sizes as visual words, generally ranging from $40 \times 40$ to $80 \times 80$. These larger patches allow for more complex visual words that capture higher levels concepts, like object parts or even full objects, but at the expense of an increase in dimensionality, as each is generally described by multiple cells of local feature descriptors.

Table 2 shows the results of methods that use small patches. We also include a baseline method in the comparison, which corresponds to the approach described in this paper, but without performing the dictionary update step (estimation of $\Theta$). By analyzing the results, we observe that our method achieves state-of-the-art performance on almost all datasets. As previously stated, even though the total dictionary size is comparable to other methods and is larger than the one in [15], this direct comparison is not meaningful, as the effective dictionary size is adaptively chosen by each category. Thus, the correct way is to compare against the effective number of words used, $\tilde{K}$. This number is significantly smaller than the dictionary size of the rest of the methods, and is similar to [15]. In both cases, the difference is that our method allows the specialization of words for certain categories, giving a clear performance advantage. We also report the results of our method using the post-processing step, under the name **Proposed2**.

Notice also that our best results outperform the baseline by $5\%$ to $9\%$ depending on the dataset used. This difference allows us to confirm our hypothesis joint learning of visual words and object classifiers increases the discriminative power of the dictionary. It is also interesting to see that the post-processing step allows for a measurable and consistent increase in performance in all datasets, confirming our intuition regarding the use of the words.

| | | **Dataset** | | | |
|---|---|---|---|---|---|
| **Method** | **# Words** | UIUC-Sports | Caltech101 | 15 Scenes | MIT67 |
| Baseline | 500-600 | 82.3 ± 1.2 | 66.8 ± 0.6 | 81.2 ± 0.4 | 35.1 |
| [4] | 400 | - | 64.6 ± 0.8 | 81.4 ± 0.5 | - |
| [9] | 1024 | - | 73.2 ± 0.5 | 80.3 | - |
| [49] | 200 | - | - | - | 26.5 |
| [51] | 2048 | - | 73.4 | 80.5 ± 0.6 | - |
| [22] | 5250 | - | - | 82.7 ± 0.5 | - |
| [52] | 300 | 71.7 | - | - | 28 |
| [53] | 1024 | 87.2 ± 1.1 | 73.2 ± 0.8 | 82.7 ± 0.5 | - |
| [54] | 200 | - | - | 78.6 ± 0.7 | 37.9 |
| [15] | 200-300 | 85.6 ± 1.1 | 72.9 ± 0.6 | 84.6 ± 0.4 | 39.5 |
| Proposed | 200-330 | 86.4 ± 1.2 | 73.6 ± 0.7 | 85.3 ± 0.4 | 41.2 |
| Proposed2 | 1000-1100 | **87.5 ± 1.3** | **75.4 ± 0.6** | **86.3 ± 0.5** | **44.1** |

TABLE 2: Categorization performance of methods that use *small patches* on 4 different datasets. Our method is able to achieve state-of-the-art performance using a smaller dictionary than competing methods.

| | | **Dataset** | | | |
|---|---|---|---|---|---|
| **Method** | **# Words** | UIUC-Sports | Caltech101 | 15 Scenes | MIT67 |
| Baseline | 500-600 | 82.3 ± 1.2 | 66.8 ± 0.6 | 81.2 ± 0.4 | 35.1 |
| [55] | 200 p.c. | 76.3 | - | 80.9 | 37.6 |
| [56] | 9 p.c. | - | - | - | 30.4 |
| [56] | 9 p.c. | - | - | - | 43.1 |
| [12] | 210 p.c. | - | - | - | 38.1 |
| [7] | 50 p.c. | - | - | - | 46.1 |
| [29] | 90 p.c. | 86.4 ± 0.9 | **78.8 ± 0.5** | 86.0 ± 0.8 | 51.4 |
| [30] | 200 p.c. | - | - | - | **64.0** |
| Proposed | 200-330 | 86.4 ± 1.2 | 73.6 ± 0.7 | 85.3 ± 0.4 | 41.2 |
| Proposed2 | 1000-1100 | **87.5 ± 1.3** | 75.4 ± 0.6 | **86.3 ± 0.5** | 44.1 |

TABLE 3: Categorization performance of methods that use *large patches* on 4 different datasets (p.c. = per class). Although performance on indoor scenes is below the state of the art, our methods remains highly competitive on the other datasets.

Table 3 shows the results of methods that use large patches on the same four datasets. On three of the datasets (*UIUC-Sports*, *Caltech101*, *15 scene categories*), our method shows competitive performance, generally using a smaller dictionary than competing methods. Only when comparing performance on the *MIT67* dataset, results of methods that use large patches outperforms our technique by a significant margin. Given the characteristics of our method and the particularities of indoor scenes [57], this is an expected situation. On these type of datasets, where intra-class image appearance varies drastically, categorization is generally improved by using semantic information, commonly captured by larger structures, like object parts. Although, in theory, groups of small patches are able to capture the same information that larger templates do [58], they need to be associated by some type of relation, in order to keep the structure of a larger visual pattern, an aspect that is not included in our method. In the case of rigid objects, as the ones generally found on indoor scenes, large patches are capable of capturing their discriminative parts or even the objects themselves. Unless small-patch methods include a mechanism to model the relations between words, this puts them in a clear disadvantage for indoor scenes, as generally they only capture information of small regular textured patterns. Unfortunately, in our case, increasing the dimensionality and size of visual words is not an option, as the dimensionality of the learning problem grows to a point where our current implementation is not able solve it in a reasonable time.that is been used by humans to solve the problem, like...

## 6 Conclusions and Future Work

We have proposed a new method for visual recognition based on a group-sparse structured output learning framework. As a main feature, the proposed method is able to jointly learn a suitable mid-level dictionary of visual words along with a set of top-level category-based classifiers. As a main contribution, our experiments provide evidence that the proposed method learns shared, discriminative, and compact representations, three relevant properties for the effectiveness and scalability of a hierarchical visual system. Among our main findings, we demonstrate the relevance of a joint training of mid and top-level layers, as well as, the effectiveness of a max-margin approach to achieve this goal. In particular, our results indicate a performance gain between 5% to 9% by using a joint learning scheme. Furthermore, this joint learning allows us to introduce group sparsity constraints that foster the specialization of the visual patterns captured by the mid-level representations. As shown by our experiments, this specialization is highly effective to manage model complexity by adapting visual words according to the classification complexity of each target class. This leads to compact representations that achieve state-of-the-art performance using an order of magnitude less visual words than previous approaches. In future work, we plan to investigate the effect of introducing further intermediate levels to our model in conjunction with suitable group sparsity constraints to manage model complexity and spatial relations among visual words. Furthermore, we plan to implement distributed version of our optimization scheme to scale the approach to larger datasets.

## Acknowledgments

This work was partially funded by grants FONDECYT 1120720, NSF 11-1218709 and ONR N000141310116.

## References

[1] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.

[3] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32(9), pp. 1627–1645, 2010.

[4] S. Lazebnik, C. Schmid, and 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.

[5] A. Torralba, K. Murphy, and W. Freeman, "Sharing features: efficient boosting procedures for multiclass object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, June 2004, pp. 762–769.

[6] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychological Review*, vol. 94, pp. 115–147, 1987.

[7] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 923–930.

[8] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *IEEE International Conference on Computer Vision*, 2003.

[9] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[10] Y. Wang and G. Mori, "Hidden part models for human action recognition: Probabilistic versus max margin," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1310–1323, 2011.

[11] A. Jain, L. Zappella, P. McClure, and R. Vidal, "Visual dictionary learning for joint object categorization and segmentation," in *European Conference on Computer Vision*, 2012.

[12] S. Singh, A. Gupta, and A. Efros, "Unsupervised discovery of mid-level discriminative patches," in *European Conference on Computer Vision*, 2012.

[13] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[14] H. Lobel, R. Vidal, D. Mery, and A. Soto, "Joint dictionary and classifier learning for categorization of images using a max-margin framework," in *Pacific-Rim Symposium on Image and Video Technology*, 2013.

[15] H. Lobel, R. Vidal, and A. Soto, "Hierarchical joint max-margin learning of mid- and top-level representations for visual recognition," in *IEEE International Conference on Computer Vision*, 2013.

[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Neural Information Processing Systems*, 2008, pp. 1033–1040.

[17] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

[18] F. Jurie and B. Triggs, "Creating efficient codebooks for visual recognition," in *IEEE International Conference on Computer Vision*, 2005.

[19] S. Lazebnik and M. Raginsky, "Supervised learning of quantizer codebooks by information loss minimization," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1294–1309, 2009.

[20] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *IEEE International Conference on Computer Vision*, 2005, pp. 1800–1807.

[21] F. Moosmann, B. Triggs, and F. Jurie, "Fast discriminative visual codebooks using randomized clustering forests," in *Neural Information Processing Systems*, 2007.

[22] X.-C. Lian, Z. Li, B.-L. Lu, and L. Zhang, "Max-margin dictionary learning for multiclass image categorization," in *European Conference on Computer Vision*, 2010, pp. 157–170.

[23] N. Zhou and J. Fan, "Jointly learning visually correlated dictionaries for large-scale visual recognition applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 715–730, 2014.

[24] S. Hwang, F. Sha, and K. Grauman, "Sharing features between objects and their attributes," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2011, pp. 1761–1768.

[25] B. Babenko, S. Branson, and S. Belongie, "Similarity metrics for categorization: from monolithic to category specific," in *IEEE International Conference on Computer Vision*, 2009.
[26] P. Felzenszwalb and D. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.
[27] L. Bourdev, S. Maji, T. Brox, and J. Malik, "Detecting people using mutually consistent poselet activations," in *European Conference on Computer Vision*, 2010.
[28] T. Malisiewicz, A. Gupta, and A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *IEEE International Conference on Computer Vision*, 2011.
[29] J. Sun and J. Ponce, "Learning discriminative part detectors for image classification and cosegmentation," in *IEEE International Conference on Computer Vision*, 2013.
[30] C. Doersch, A. Gupta, and A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *Neural Information Processing Systems*, 2013.
[31] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
[32] G. E. Hinton and S. Osindero, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, p. 2006, 2006.
[33] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural Information Processing Systems*, 2012.
[34] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *IEEE International Conference on Computer Vision*, 2009, pp. 221–228.
[35] Y. Jia, C. Huang, and T. Darrell, "Beyond spatial pyramids: Receptive field learning for pooled image features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3370–3377.
[36] D. Chen, D. Batra, and W. Freeman, "Group norm for learning structured svms with unstructured latent variables," in *IEEE International Conference on Computer Vision*, 2013, pp. 409–416.
[37] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *International Conference on Artificial Neural Networks*, 2010, pp. 92–101.
[38] X. Wang, T. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *IEEE International Conference on Computer Vision*, 2009, pp. 32–39.
[39] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.
[40] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *International Conference on Machine Learning*, 2004.
[41] C. J. Yu and T. Joachims, "Learning structural SVMs with latent variables," in *International Conference on Machine Learning*, 2009.
[42] A. L. Yuille and A. Rangarajan, "The concave-convex procedure," *Neural Computation*, vol. 15, no. 4, pp. 915–936, Apr. 2003.
[43] R. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, Sep. 1995. [Online]. Available: http://dx.doi.org/10.1137/0916069
[44] P. Ott and M. Everingham, "Shared parts for deformable part-based models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1513–1520.
[45] J. Yu, S. Vishwanathan, S. Günter, and N. Schraudolph, "A quasi-newton approach to nonsmooth convex optimization," in *International Conference on Machine Learning*, 2008.
[46] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of Optimization Theory and Applications*, vol. 109, no. 3, pp. 475–494, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1023/A:1017501703105
[47] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.
[48] L.-J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.
[49] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
[50] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60(2), pp. 91–110, 2004.
[51] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.
[52] J. Zhu, L. Li, L. Fei-Fei, and E. Xing, "Large margin training of upstream scene understanding models." in *Neural Information Processing Systems*, 2010.
[53] A. Shabou and H. Le-Borgne, "Locality-constrained and spatially regularized coding for scene categorization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
[54] S. Parizi, J. Oberlin, and P. Felzenszwalb, "Reconfigurable models for scene recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2775–2782.
[55] E. P. X. Li-Jia Li, Hao Su and L. Fei-Fei, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Neural Information Processing Systems*, 2010.
[56] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *IEEE International Conference on Computer Vision*, 2011.
[57] P. Espinace, T. Kollar, N. Roy, and A. Soto, "Indoor scene recognition by a mobile robot through adaptive object detection," *Robotics and Autonomous Systems*, vol. 61, no. 2, 2013.
[58] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures of parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.

**Hans Lobel** received his B.S. and M.S. degrees in Computer Science from the Pontificia Universidad Católica de Chile in 2007 and 2009, respectively. He is currently a Ph.D. student in the Computer Science Department at Pontificia Universidad Católica de Chile, doing research on visual recognition. He received the best student paper award at PSIVT 2013. His research interests are in the application of machine learning to computer vision tasks.

**René Vidal** received his B.S. degree in Electrical Engineering (valedictorian) from the Pontificia Universidad Católica de Chile in 1997 and his M.S. and Ph.D. degrees in Electrical Engineering and Computer Sciences from the University of California at Berkeley in 2000 and 2003, respectively. He has been a faculty member in the Department of Biomedical Engineering of The Johns Hopkins University since 2004. He was co-editor of the book "Dynamical Vision" and has co-authored more than 180 articles in biomedical image analysis, computer vision, machine learning, hybrid systems, and robotics. He has received many awards for his work including the 2012 J.K. Aggarwal Prize, the 2009 ONR Young Investigator Award, the 2009 Sloan Research Fellowship, the 2005 NFS CAREER Award, and best paper awards at ICCV-3DRR 2013, PSIVT 2013, CDC 2012, MICCAI 2012, CDC 2011 and ECCV 2004. Engineering of the Pontificia Universidad Católica de Chile to the best graduating student of the school. He is a fellow of the IEEE and a member of the ACM and SIAM.

**Alvaro Soto** received his Ph.D. in Computer Science from Carnegie Mellon University in 2002; and a M.Sc. degrees in Electrical and Computer Engineering from Louisiana State University in 1997. He joined the Computer Science Department at Pontificia Universidad Católica de Chile, where he became Associate Professor in 2007. His research interests are in visual recognition, machine learning, and cognitive robotics.