Category co-occurrence modeling for large scale scene recognition

Xinhang Song[a], Shuqiang Jiang*[a], Luis Herranz[a], Yan Kong[b], Kai Zheng[c]

[a] *The Institute of Computing Technology of the Chinese Academy of Sciences, Beijing, China*
[b] *The Institute of Automation of the Chinese Academy of Sciences, Beijing, China*
[c] *School of Computer Science, Soochow University, China*

Abstract

Scene recognition involves complex reasoning from low-level local features to high-level scene categories. The large semantic gap motivates that most methods model scenes resorting to mid-level representations (e.g. objects, topics). However, this implies an additional mid-level vocabulary and has implications in training and inference. In contrast, the semantic multinomial (SMN) represents patches directly in the scene-level semantic space, which leads to ambiguity when aggregated to a global image representation. Fortunately, this ambiguity appears in the form of scene category co-occurrences which can be modeled a posteriori with a classifier. In this paper we observe that these patterns are essentially local rather than global, sparse, and consistent across SMNs obtained from multiple visual features. We propose a co-occurrence modeling framework where we exploit all these patterns jointly in a common semantic space, combining both supervised and unsupervised learning. Based on this framework we can integrate multiple features and design embeddings for large scale recognition directly in the scene-level space. Finally, we use the co-occurrence modeling framework to develop new scene representations, which experiments show that outperform previous SMN-based representations.

Keywords:
scene recognition; co-occurrence modeling; semantic space; feature embedding; multiple feature combination; large scale image recognition

1. Introduction

Visual understanding is essentially a complex process of abstraction, from purely local visual information to abstract semantic entities such as objects and scenes. The conventional visual recognition strategy has consisted of extracting local visual features[1], and encoding them into a global representation of the image using some variation of the bag-of-words (BOW) model[2, 3, 4, 5]. While very effective for object recognition, scenes often require more abstract representations composed of other lower-level semantic entities, such as objects or themes, which appear in the scene in a loose layout, contrasting with the much more rigid structure of parts in objects. Thus, it may be difficult to model scene categories directly from low-level visual descriptors, due to a larger semantic gap.

An intermediate abstraction level can represent the presence of local concepts (e.g. *sky*, *rock*, *street, car*)[6] (see Figure 1a and c), and then scene categories (e.g. *coast*, *inside city*, *kitchen*) are recognized based on this intermediate representation. Thus, the semantic gap is reduced by performing the abstraction gradually in two steps. Note that now several problems arise related with the mid-level representation. First, it requires selecting a set of mid-level vocabulary of local concepts. In addition, training intermediate classifiers[6, 7, 8] requires images with regions annotated with these mid-level concepts, which is much more costly than annotating just one scene label. Some works avoid this problem by using latent topics for the intermediate representation[9, 10, 11], where topics are discovered during learning. However, these methods often have limited performance due to poor supervision[12], and are often based on complex generative models difficult to scale to large datasets.

Alternatively, the *semantic multinomial (SMN)*[13] represents the probability that a given patch belongs to each scene category, being a local but not mid-level representation. Image-SMNs are obtained by aggregating patch-

*Corresponding author.
Email addresses: xinhang.song@vipl.ict.ac.cn (Xinhang Song), sqjiang@ict.ac.cn (Shuqiang Jiang*), luis.herranz@vipl.ict.ac.cn (Luis Herranz), kongyanwork@gmail.com (Yan Kong), zhengkai@suda.edu.cn (Kai Zheng)

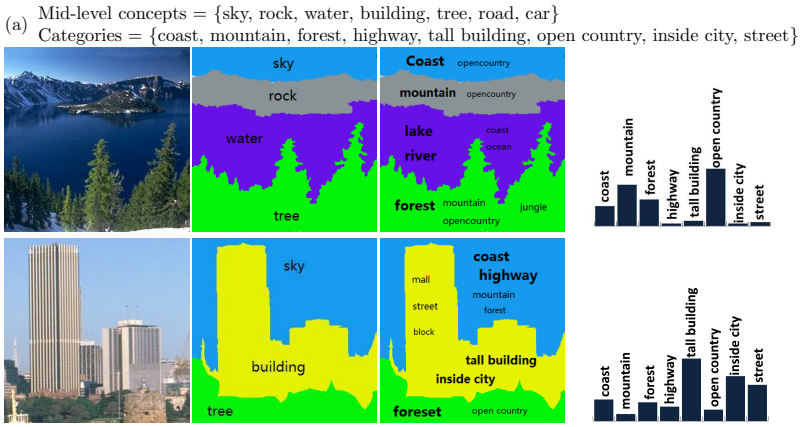


Figure 1: Scene category co-occurrences: (a) categories and mid-level local concepts, (b) images from the *open country* (top row) and *tall building* (bottom row) categories, (c) regions with their corresponding mid-level concepts, (d) regions modeled in terms of scene categories, and (e) resulting image-SMNs. Note that categories co-occurrences appear both at global (e) and local (d) levels.

SMNs (see Figure 1a and d). Thus, both regions and images use the same *semantic space*, both are in the scene level, without need for mid-level vocabulary nor local annotation. Patch models are learned using weak-supervision via the scene category label (i.e. all the patches share the same label). Patches independently represent different lower-level concepts that may be also common in other scene categories (e.g. patches depicting walls are common in *bedroom*, *livingroom* and *office*). For this reason, weakly-supervised learning leads to the particular phenomenon of *(scene) category co-occurrences*[1]: related scene categories, sharing regions with the same mid-level concepts, will always show certain probability in their image-SMNs[13] (see Figure 1e). This problem makes SMNs too ambiguous and prevents them from being reliable to predict the scene category. Fortunately, Rasiwasia and Vasconcelos[13, 14] showed that these category co-occurrences are consistent across the images in the same category, so they can be modeled with a second classifier, that separates them from purely accidental co-occurrences, that can be regarded as noise. Thus, the *co-occurrence noise* is simply the noise in the probabilities that compose the SMN, and appears due to the intrinsic ambiguity of the underlying patch representation learned with weak supervision via scene labels. The main motivation in our paper is to find more robust ways to exploit consistent co-occurrences to better model scene categories, while filtering out co-occurrence noise.

In this sense, previous frameworks[13, 14] have several limitations, notably considering category co-occurrences *only at global level*, using *only supervised* modeling over a *single feature*. In this paper we propose a broader and integrated view of scene category co-occurrences, which provides us with an alternative framework to address many problems exploiting the common semantic space (i.e. the simplex of scene categories) for both patches and images, and across different visual features. Since co-occurrences are essentially local (see Figure 4 c), we also focus on modeling the co-occurrences locally. However, since patch-SMNs are much noisier than image-SMNs (global), we exploit the common space also to integrate multiple features, obtaining more consistent SMNs. In addition and in contrast to previous works, where modeling is only supervised, we include an unsupervised filtering approach, and an extended kernelized version, and propose two new representations suitable for large scale classification. From more general to more specific, there are three levels of contributions:

- A more general *framework* for modeling scene category co-occurrences.
- Within this framework we develop *new tools* described in terms of modeling category co-occurrences, such as the integration of multiple features in the semantic space and unsupervised filtering of co-occurrence noise.
- Three different *semantic representations* (filtered SMNs, co-codes and KCNF embedding), two of them suitable

[1]In [13], the authors use the term *contextual co-occurrences* to refer to consistent and thus desirable co-occurrence patterns. Here, we refer to them as *(scene) category co-occurrences* to emphasize that they are high-level categories rather than low or mid-level co-occurrences. Similarly, we use the term *co-occurrence noise* instead of *contextual noise*.

for large scale classification with linear classifiers.

The overview of the framework is shown in Figure 2 with the proposed components and features highlighted. We organized the rest of the paper as follows. Section 2 reviews previous works in related areas. Sections 3, 4 and 5 introduce the proposed framework, category co-occurrence modeling techniques and semantic representations. Experimental evaluations and conclusions are presented in Sections 6 and Section 7.

2. Related work

2.1. Intermediate representations for scene recognition

Table 1: Semantic representations in different scene recognition approaches.

Approach	Intermediate representations						Uses external data
	Name	Abstraction level	Local/ global	Vocabulary	Explicit/ implicit	Labeling	
BOW/FV[3, 15, 5]	-	-	-	-	-	-	No
Mid-level[6, 11]	Themes/ objects	Medium	Local	Themes/ objects	Explicit	Yes	Sometimes
CNNs[16, 17]	CNN layers	Low to high	Local to global	-	-	No	Yes(very large)
CMN[13], SPMSM[14]	SMN	High	Local/global	Scenes	Implicit	No	No
Proposed	SMN	High	Local	Scenes	Implicit	No	No

Vogel and Schiele[6] proposed to model natural scenes in two steps, using a local mid-level representation of local concepts such as *water*, *rocks* or *foliage*. Similarly, Object bank[11] uses classifiers learned from ImageNet and includes multiple scales, as in a spatial pyramid[18], to obtain a more descriptive representation. Locality in the intermediate representation is not always necessary. The *classemes* representation[8] is based on a set of fixed basis classes. Attributes[19, 20] follow a similar idea, where classifiers are trained to detect whether certain attributes are present or not. Attributes can be modeled at both local and global levels, and defined for both objects[19] and scenes[21].

Latent topics are often modeled using Latent Dirichlet Allocation (LDA)[9, 12]. However, most LDA variants have been shown to capture irrelevant general regularities rather than the semantic regularities of interest, due to poor supervision[12]. Spatial context can be included to model the global layout and enforce local coherence in the topics[10]. A series of related mid-level latent representations, more focused on indoor scenes, are distinctive parts [22, 23] and exemplars[24, 25]. Quattoni and Torralba[22] observe that many indoor scenes can be clearly represented by the objects they contain, and propose finding distinctive parts as mid-level representations. This problem can be complex, as these mid-level parts are unknown and must be discovered. Lin et al[23] jointly learn appearance and spatial pooling regions. Zuo et al[24] propose an approach to detect the exemplars, which is used to hierarchically learn filter banks to transform raw pixel patches to features. CNNs extract intermediate representations progressively using multilayered neural networks[16]. While achieving impressive results[17], the success of a CNN for scene recognition is largely dependent on the pre-training dataset[17]. Table 1 compares several scene recognition approaches in terms of intermediate representations.

Most relevant to this paper is the semantic multinomial (SMN)[26] and the related approaches for scene recognition[13, 14], where both patches and images are represented in terms of scene categories. In contrast to latent topics, the models for the two stages can be learned independently, as SMNs are defined directly over scene categories. This allows patch models to be learned directly without resorting to LDA to discover a latent vocabulary. Patch models are learned via weak-supervision using image labels, and then a second classifier (referred to as contextual model in [13]) will correct the ambiguity resulting from the weak labeling. This unconventional approach is mainly used in the contextual multinomial[13] and the semantic manifold[14], which differ in the contextual model (Dirichlet mixture models and SVMs, respectively), and the latter using a spatial pyramid (see Section 3.1 for more details). Table 2 compares these approaches with the proposed one in detail.

Table 2: Approaches based on the semantic multinomial representation.

Approach	Multiple features	Co-occurrences modeling	Feature embedding		Classifier		Spatial layout
			small dataset	large dataset	small dataset	large dataset	
CMN[13]	No	Global	None		DMM	-	None
SPMSM[14]	No	Global	None	Square root	Kernel SVM	Linear SVM	SPM
Proposed	Yes	Local+global	Co-codes, KCNF		Linear SVM		SPM

2.2. Large scale scene recognition

Computational efficiency is critical to scale to large datasets. Most image recognition methods use SVMs with an appropriate non-linear kernel. However, non-linear SVMs do not scale well for large datasets, with a time complexity at least quadratic with the number of the images[27]. In contrast, linear SVMs can be efficiently learned in linear time. non-linear SVMs with a given kernel are equivalent to linear SVMs in a certain feature space, so finding an appropriate embedding to project into that feature space would allow using linear SVMs and thus enable large scale classification. Unfortunately, exact embeddings are rarely available in close form or are too complex, so embeddings are often approximations[27].

In the case of SMNs, Kwitt et al[14] suggest that the negative geodesic distance kernel is suitable for the semantic space of SMNs, and propose an approximate embedding. With this approximation the accuracy decreases around 0.5-2%, but it enables large scale classification in the semantic space using linear classifiers. Here we propose an exact embedding that also integrates unsupervised filtering, outperforming in practice the original NGD kernel even using non-linear SVMs.

Methods using explicit mid-level representations scale well to large scale, as inference is split into two independent steps where (linear) classifiers are trained independently. However, discovering latent topics often require complex generative models (most being variations of LDA[9, 10, 12]), which require costly iterative methods which are only practical for relatively small datasets with few categories and topics. Training CNNs is also very demanding, but feasible in practice using efficient GPU implementations[16].

2.3. Multiple feature fusion

As independent features cannot capture the complexity of visual scenes, combining multiple low-level features can improve the performance by modeling complementary visual aspects resulting in a better descriptive ability. Note that each visual feature lies in a different feature space, so combining multiple features is not trivial. At region level, region descriptors from multiple features can be concatenated[28]. The color attention method[29] combines shape and color, weighting the histogram of shape visual words using concept-dependent color information. Or multiple features are combined by learning and selecting a small number of features[30]. At a global level, [31] uses multiple kernel learning (MKL) to combine features using a kernel resulting from a weighted average of each feature kernel. Lin et al [32] combine the multiple features by learning multiple BOW features, which are used to guide the graph partitions.

However, combining multiple features in the semantic space has not been explored. Our method exploits the semantic simplex, which is common for all the features and thus provides a convenient way to combine them.

3. Scene category co-occurrence modeling framework

The proposed framework is built upon the semantic manifold framework (in particular the extension SPMSM)[14], based itself on the SMN representation[26]. We first describe them, and then analyze the different types of scene category co-occurrences motivating the new extensions we include in the framework: multi-feature combination and unsupervised modeling of category co-occurrences (Section 4) and new semantic feature representations (Section 5). We combine these extensions in the common semantic space, and each of them aims to model different types of scene category co-occurrences. Figure 2 shows the whole extended framework with the different variations discussed in this paper.

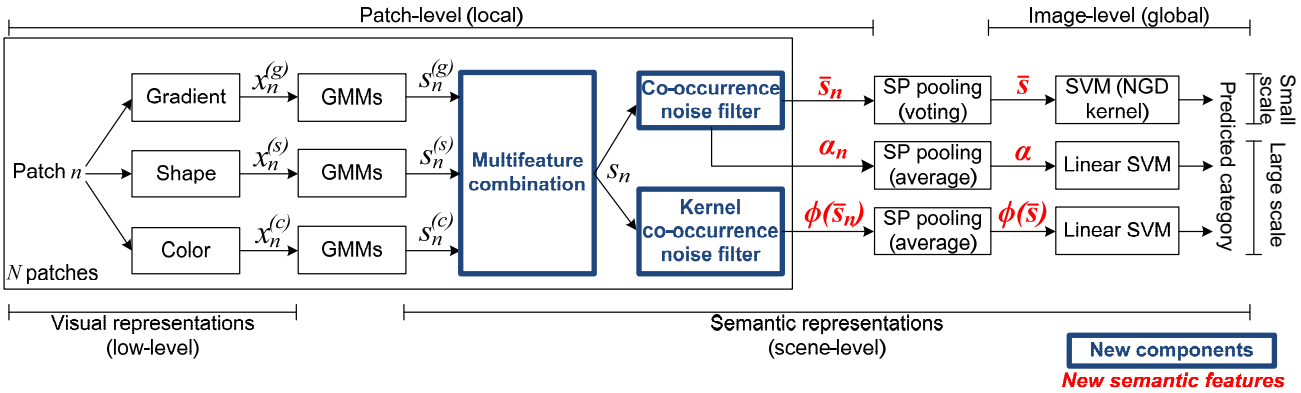


Figure 2: Overview of the recognition framework with the proposed new components highlighted, and the three types of semantic features.

3.1. Semantic multinomial and semantic manifold

The probability distribution of the semantic concepts (i.e. scene categories) is estimated from a set of local visual features defined in some visual feature space X. Each image from the dataset is represented as a bag of local visual descriptors $I = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, $\mathbf{x}_n \in X$, densely sampled in a grid with N local patches. Given a vocabulary of scene categories $\{w_1, \ldots, w_M\}$, each image is labeled with one of those M categories. As patch labels are not available, conditional distributions $P_{\mathbf{X}|W}(\mathbf{x}_n|w)$ are learned using weak supervision via image labels. Images are modeled using the generative process shown in Figure 3a, in which a scene category w is first sampled, and then N (patch) feature vectors are generated from $P_{\mathbf{X}|W}(\mathbf{x}_n|w)$[13]. Given a new image, the category can be predicted using the Bayes rule

$$P_{W|\mathbf{X}}(w|I) = \frac{\prod_{n=1}^{N} P_{\mathbf{X}|W}(\mathbf{x}_n|w)\, P_W(w)}{\prod_{n=1}^{N} P_{\mathbf{X}}(\mathbf{x}_n)} \tag{1}$$

which assumes a uniform prior for $P_W(w)$. Each patch model $P_{\mathbf{X}|W}(\mathbf{x}_n|w)$ is modeled as a Gaussian mixtures model (GMM), learned over a training set with D images.

Once we have learned the GMMs, we can estimate the distribution of concepts for a new image I using the posterior probability $P_{W|\mathbf{X}}(w|I)$. For a vocabulary with M scene categories, the vector of posterior probabilities $\mathbf{s} = (s_1, ..., s_M)^T$ with $s_w = P_{W|\mathbf{X}}(w|I)$, is referred to as the *semantic multinomial* (SMN) of the image I[26]. The SMN is a probability vector of concepts that lies in the simplex Δ^{M-1} (referred to also as semantic space or semantic simplex). The whole process can be seen as a mapping from the set of local visual descriptors in the image to the semantic space. Finally, given a new image, a label can be predicted from a SMN by simply selecting the concept with maximum probability (we will refer to this decision method as Bayes classification).

Similarly, we can also use this mapping over patches to extract local SMNs. This alternative view allows us to infer the image-SMN from patch-SMNs, using some a certain patch-to-image operation. In particular, for (1) the corresponding operation is just a product of the semantic multinomials

$$s_w = \Omega_w^{\text{prod}}(I) = \prod_{n=1}^{N} s_{nw} \tag{2}$$

where

$$s_{nw} = P_{W|\mathbf{X}}(w|\mathbf{x}_n) = \frac{P_{\mathbf{X}|W}(\mathbf{x}_n|w)\, P_W(w)}{P_{\mathbf{X}}(\mathbf{x}_n)} \tag{3}$$

is the w-th element of the patch-SMN $\mathbf{s}_n$. This is equivalent to using the same generative model of Figure 3a to infer the image-SMN from patches.

However, directly using the image-SMNs learned with (2) for classification (i.e. predicting the category with maximum probability in the SMN) has some limitations. As Rasiwasia and Vasconcelos[13] observed, using (2) and

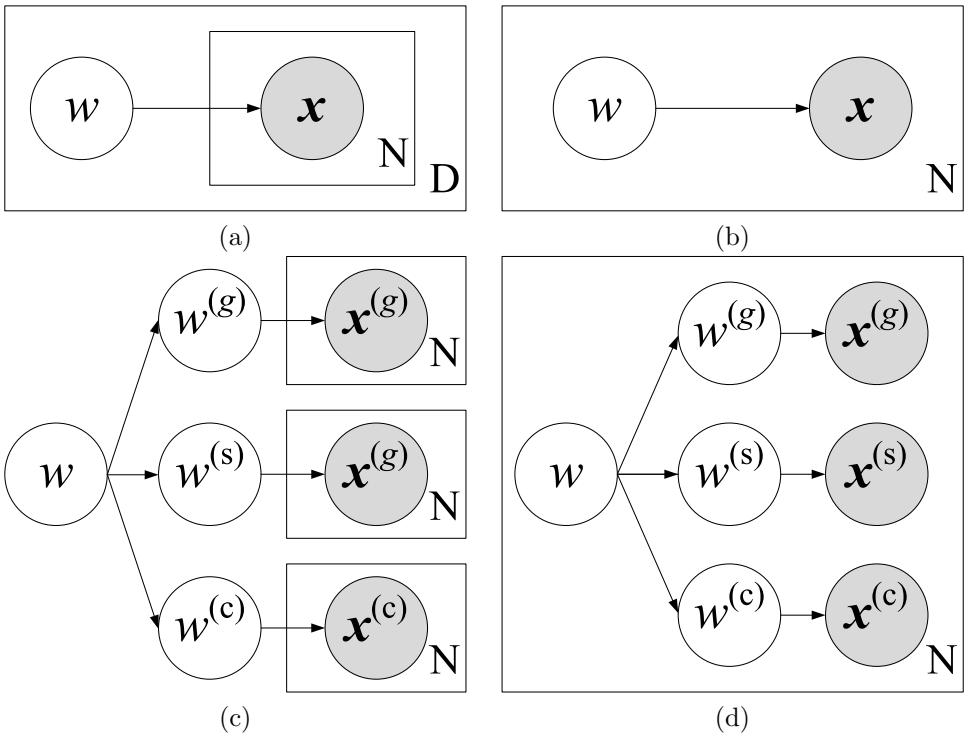


Figure 3: Generative models used to obtain semantic representations of patches: (a) single feature learning (one concept per image), (b) single feature inference (one concept per patch), (c) multiple feature inference (one concept per image), and (d) multiple feature inference (one concept per patch).

(3) for inference results in highly peaked SMNs, with one category with a probability close to one, and the rest having very low probability (i.e. very low entropy). However, the recognition accuracy is limited, and these overconfident SMNs prevent from any further processing to correct a wrong prediction. The underlying generative process of Figure 3a, used to learn the patch models, implies that *one* concept is sampled per image and then N patch visual appearances x_n are generated according to that concept. The main reason to use this model is that we lack patch labels. However, a different model can be used for inference during the test stage. The contextual multinomial[13] and the semantic manifold[14] use an alternative model where patches are independent (see Figure 3b). In this case, one concept is sampled per patch, and then the corresponding visual appearance x_n is generated. This process is repeated N times per image. Using this model, the contextual multinomial approach[13] uses the geometric mean to infer image-SMNs (rather than the product resulting from the model of Figure 3a). Similarly, the semantic manifold framework[14] uses a voting method, which is the method we will use in our experiments. First, a scene label is assigned to each patch as $w_n^* = \max_w s_{nw}$. Then a histogram is obtained by counting the occurrences of each scene label in the image as $o_w = |\{w_n : w_n^* = w\}|$. The image-SMN $\mathbf{s}$ is obtained as

$$s_w = \Omega_w^{\text{vot}}(I) = \frac{o_w + \beta - 1}{\sum_{w=1}^{M}(o_w + \beta - 1)} \tag{4}$$

where β is a regularization parameter.

Figure 4 shows image-SMNs obtained from for images from two different categories. This alternative inference (independent patches) method results in SMNs with higher entropy, and with more categories having noticeable probabilities. The key observation is that, for a given category, those categories with significant probability are often the same, and follow a similar pattern in the image-SMNs (e.g. note that forest, mountain and opencountry, the top row of Figure 4). We refer to them as *scene category co-occurrences* (or co-occurrence patterns). Note that we can observe that these patterns are repeated across all the images-SMNs in the same category, and thus can be modeled. This is the main idea underlying in the contextual multinomial[13] and the semantic manifold[14]

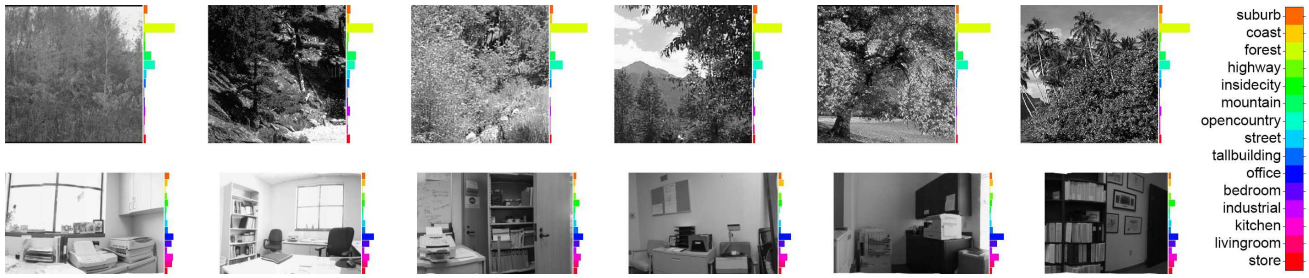


Figure 4: Images from *forest* (top row) and *office* (bottom row) categories of the *15 scenes* dataset. The corresponding semantic representations are shown next to each image, with higher bars for categories with higher probability. Consistent co-occurrence patterns between related categories can be observed across the different images (e.g. *mountain* and *opencountry* for *forest*; *bedroom*, *kitchen* and *livingroom* for *office*), while each image also exhibits some level of noise in the representation (co-occurrence noise).

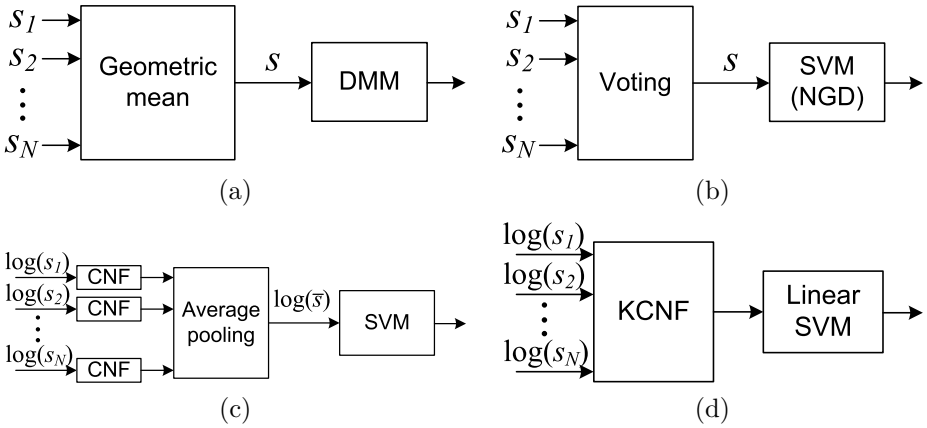


Figure 5: Patch to image combination and and co-occurrence modeling: (a) contextual models with DMM[13], (b) semantic manifold[14] (using SVM), (c) proposed framework with CNF (in logarithmic space and patch level modeling), and (d) kernel CNF.

approaches. We emphasize the importance of using the model in Figure 3b, since inferring image-SMNs using product does not exhibit co-occurrence patterns. In contrast, using geometric mean and voting do, thus enabling co-occurrence modeling.

The next stage includes supervised co-occurrence modeling where the image-SMNs are the inputs. The contextual multinomial is obtained using DMMs (see Figure 5a). We follow the semantic manifold, which uses SVMs (see Figure 5b). This additional layer is likely to correct many mistakes after learning those co-occurrence patterns that are consistent across each category. In order to exploit the geometry of the semantic simplex, a suitable distance is the geodesic distance $d_{GD}(\mathbf{s},\mathbf{s}') = 2\arccos\left(\left\langle\sqrt{\mathbf{s}},\sqrt{\mathbf{s}'}\right\rangle\right)$ where $\sqrt{\mathbf{s}}$ denotes element-wise square root. A negative geodesic distance (NGD) kernel can be defined from this distance as $k_{NGD}(\mathbf{s},\mathbf{s}') = -d_{GD}(\mathbf{s},\mathbf{s}')$[33], which is suitable to used in SVMs when the inputs are SMNs.

Note that using kernels limits the application of SVM classifiers to large datasets, due to the high computational cost, so Kwitt et al[14] also propose an approximate embedding of the NGD kernel. Thus, the same framework can be used for large scale scene recognition combined with linear SVMs. For better performance, a spatial pyramid representation[18] is included to rough encode spatial context (i.e. Spatial Pyramid Matching Semantic Manifold or SPMSM).

3.2. Types of scene category co-occurences

So far we have seen that the SMN representation, and particularly after aggregating from patches to image, may exhibit scene category co-occurrence patterns, which can be modeled by a classifier to improve recognition. Now we consider more in detail the phenomenon of scene category co-occurrences and their causes.

Semantic representations learned from bags of local features (i.e. without considering contextual information such as spatial relations or related themes) suffer from the same ambiguity problems of purely visual BOW models[13]. This is aggravated in the case of the SMN representation due to the use of scene categories as vocabulary for patch-level representations. In particular, patches with similar appearances can appear in different unrelated categories (i.e. visual polysemy). Thus when a similar appearance is found in a new image, the related category will get certain non-zero probability, resulting in ambiguity in the representation. On the other hand, patches from multiple local concepts co-occur for different related scene categories (i.e. concept synonymy). For instance, indoor categories such as *kitchen*, *livingroom* or *bedroom* contain patches representing walls, door edges, tables, etc. Similarly, natural scenes often contain patches depicting the same local concept, such as sky, rocks, water, trees, etc. When aggregated into image-SMNs, this results in that some related categories have significant probabilities, as shown in the examples of Figure 4.

3.2.1. Consistent co-occurrence versus co-occurrence noise

In practice, category co-occurrence patterns are noisy (see Figure 4), and thus we distinguish between *consistent (scene category) co-occurrences* and *co-occurrence noise* (contextual and ambiguity co-occurrences in [13][1]). The former are desirable and consistent across different images of the same concept, while the latter are accidental, not desirable and can be regarded as noise. Furthermore, consistent co-occurrences are usually sparse.

Previous works[13, 14] only analyze consistent co-occurrences in image-SMNs, obtained from a single type of local visual feature. Here we describe co-occurrences from complementary angles, which will be useful to better understand the tools developed in the next sections.

3.2.2. Local versus global co-occurrences

An important observation is that co-occurrence patterns are essentially *local*, even for scene-level vocabularies, and certain co-occurrence patterns are localized in some regions. When the patch-SMNs are aggregated into image-SMNs, location information is lost, and only global co-occurrences can be modeled. While scene category co-occurrences (and the corresponding weights) are different depending on the region (see Figure 1c), image-SMNs only contain global co-occurrences (see Figure 1d), and thus some local co-occurrence patterns that may be very discriminative are lost.

Local category co-occurrences and local topics have some similarities. In both cases we try to discover co-occurrences of concepts, but there two subtle differences. Topics result from co-occurrences of low-level concepts (i.e. visual words) and lie in an intermediate level of abstraction (see Figure 1b) and this mid-level vocabulary is often unknown (i.e. latent topics), while category co-occurrences (see Figure 1c) result from co-occurrences of high level concepts defined over an explicit vocabulary (i.e. scene categories).

3.2.3. Feature-specific versus inter-feature co-occurrences

Different visual features (e.g. color, shape, gradient) may capture different yet complementary properties of patches. Consequently, patch-SMNs learned from those features will also show co-occurrence patterns that depend on those visual features. These patterns may be specific to one feature (i.e. feature-specific co-occurrences) or consistent across two or more features (i.e. inter-feature co-occurrences).

Figure 6a-c illustrate image-SMNs obtained from different features, using a three-categories toy example. Feature-specific SMNs can separate different concepts, but they are still too close to the center of the simplex and there is some overlap between categories. By exploiting inter-feature co-occurrences (using the method proposed in Section 4.1), we obtain multi-feature SMNs in which the categories can be better separated (see Figure 6d). Note that this toy example is not too realistic, as categories are already clearly separated by feature-specific image-SMNs, but it is useful to illustrate the idea.

It is important to emphasize that all the category co-occurrences we describe here are represented in the same feature space, i.e. the scene-level semantic space of SMNs. This provides a natural way to combine them (e.g. local to global, feature-specific to multi-feature) without the hassle of dealing with feature-specific or scale-specific spaces.

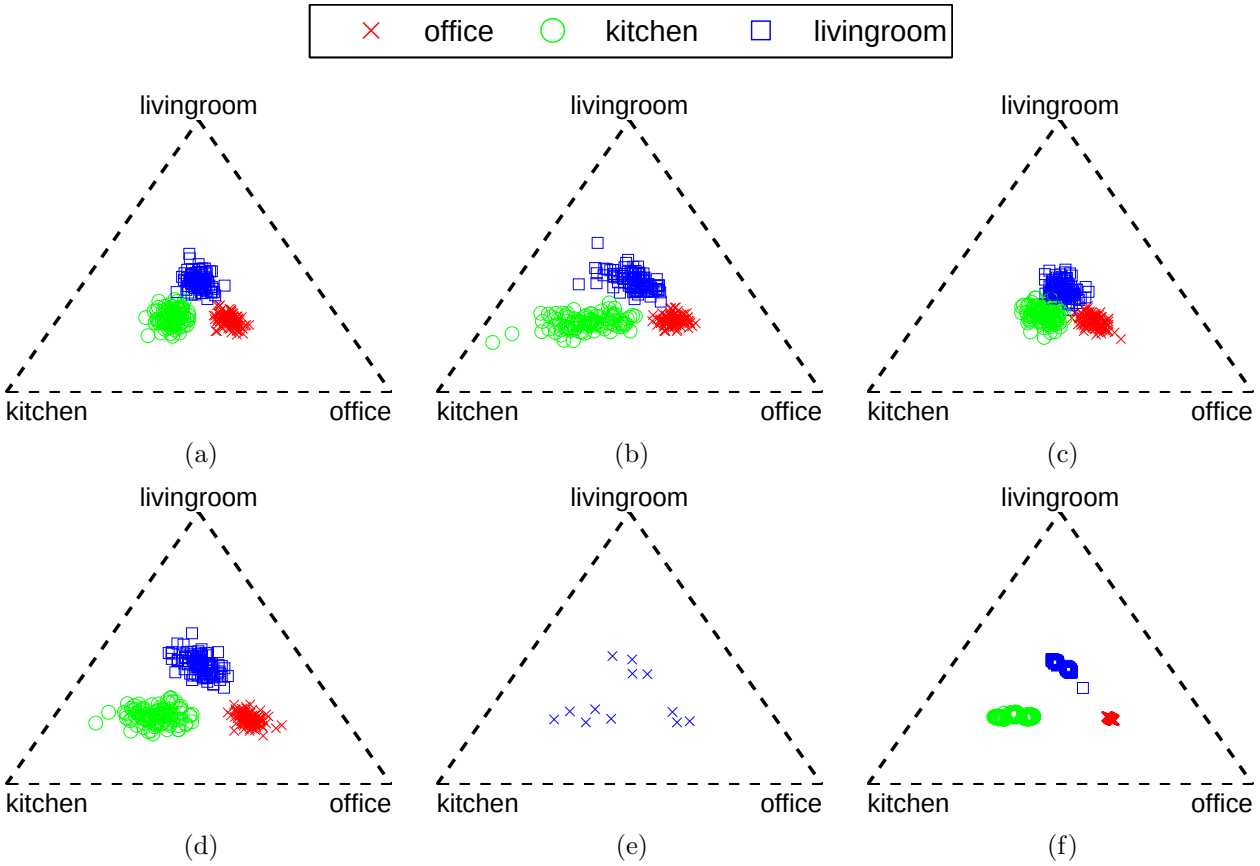


Figure 6: Toy example to illustrate multi-feature combination and co-occurrence noise filtering. Each point is represented in a 3-categories semantic simplex: (a) gradient, (b) shape, and (c) color feature-specific SMNs, (d) multi-feature SMNs, (e) co-occurrence dictionary, (f) filtered SMNs.

4. Modeling category co-occurrences

4.1. Multi-feature combination

Instead of a single type of visual feature, we now consider a set of complementary ones V (in our experiments $V = \{\text{gradient, shape, color}\}$). Each feature $v \in V$ generates a set of local visual descriptors $I^v = \{\mathbf{x}_1^v, \ldots, \mathbf{x}_N^v\}$, $\mathbf{x}_n^v \in \mathbf{X}^v$, and $I = \{I^1, \ldots, I^{|v|}\}$ represents all the features in the image. Now we assume that we learn feature-specific theme models $P_{\mathbf{X}^v|W}\left(\mathbf{x}_n^v|w^v\right)$, learned independently in the same way as in the single feature case. Similarly to (3), we can define the feature-specific patch-SMN as

$$s_{nw}^v = P_{W^v|\mathbf{X}^v}\left(w^v|\mathbf{x}_n^v\right) = \frac{P_{\mathbf{X}^v|W^v}\left(\mathbf{x}_n^v|w^v\right) P_{W^v}\left(w^v\right)}{P_{\mathbf{X}^v}\left(\mathbf{x}_n^v\right)} \tag{5}$$

Introducing explicit dependence between feature-specific categories w^v and the combined category w we obtain the model shown in Figure 3c. Thus, we define the (image) multi-feature SMN as

$$s_w = \prod_{v \in V} P_{W|W^v}\left(w|w^v\right) \prod_{n=1}^{N} P_{W^v|\mathbf{X}^v}\left(w^v|\mathbf{x}_n^v\right) \tag{6}$$

Using the Bayes rule we can extend (1) as

$$s_w = \frac{\prod_{v\in V} P_{W^v|W}\left(w^v|w\right) P_W\left(w\right) \prod_{n=1}^{N} P_{\mathbf{X}^v|W^v}\left(\mathbf{x}_n^v|w^v\right)}{\prod_{v\in V}\prod_{n=1}^{N} P_{\mathbf{X}^v}\left(\mathbf{x}_n^v\right)} \tag{7}$$

where $P_{W^v|W}\left(w^v|w\right) = \frac{P_{W|W^v}(w|w^v)P_{W^v}(w^v)}{P_W(w)}$. Using (5) we can further rearrange (7) into

$$s_w \quad = \quad \prod_{n=1}^{N}\prod_{v\in V} P_{W|W^v}\left(w|w^v\right) s_{nw}^v \tag{8}$$

which gives us a way to infer the image-SMN from multiple feature-specific SMNs. Furthermore, we can obtain multi-feature patch-SMNs as $s_{nw} = \prod_{v\in V} P_{W|W^v}\left(w|w^v\right) s_{nw}^v$. Note that with this definition we still can combine multi-feature patch-SMNs into image-SMN using (2).

As in the case of single feature, inference with the model in Figure 3c leads to few co-occurrences. So instead of that model we prefer the model in Figure 3d which infers concepts per patch independently and encourages richer co-occurrence patterns by using voting with (4). In sum, we use the product of feature-specific patch-SMNs to obtain multi-feature patch-SMNs, and voting to obtain the final image-SMN.

In (8) we can assume that each feature contributes equally to model the concept, thus $P_{W^v|W}\left(w^v|w\right)$ is constant. However, this assumption is not realistic, as certain features may capture discriminative aspects better than others, and also some concepts may be better modeled with specific features. Thus, given a concept, more discriminative features should have higher probability in order to achieve more accurate classification. We can learn $p_v = P_{W|W^v}\left(w|w^v\right)$ using a grid search and evaluating the accuracy over a validation set. In our experiments we have three types of features, so we need to find the optimal two values in a grid (p_1, p_2), subject to $0 \leq p_1 \leq 1$, $0 \leq p_2 \leq 1$ (since $p_3 = 1 - p_1 - p_2$). While we can use this approach for Bayes classification, if we use a classifier over the SMNs this approach becomes impractical since we need to retrain the classifier for each point in the grid. Since test is much faster, we use an iterative two step approach to estimate some reasonable values as follows:

1. First initialize $q = 0$, $p_v^{(0)} = 1/|V|$and get accuracy $Acc^{(0)}$ with $p_v^{(0)}$.
2. Obtain the multi-feature SMNs with (8) and retrain the classifier for $p_v^{(q)}$.
3. Find $p_v^{(q+1)}$ with maximum accuracy over the validation set by grid search over G.
4. Increase $q = q+1$, and return to step 2 (repeat for a few iterations or until validation accuracy not increases).

4.2. Unsupervised filtering of co-occurrence noise

As we discussed before, the co-occurrence patterns we observe in image-SMNs are combinations of consistent co-occurrences and co-occurrence noise. The former are desirable and allow us to model the categories in terms of co-occurrences, while the latter is undesirable and distorts the data. Thus, we can consider consistent co-occurrences as a signal we want to preserve. In previous works classifiers are fed with the original noisy SMNs. In contrast, we learn co-occurrence models from *filtered* SMNs. We propose a new model termed *co-occurrence noise filter* (CNF) that aims at filtering out co-occurrence noise while keeping consistent co-occurrences. This filter is based on an encoding-reconstruction approach over a dictionary of co-occurrences. The CNF can be considered as an unsupervised pre-processing stage, prior to the supervised co-occurrence modeling by the SVM classifier.

An SMN can be modeled as a linear combination of contextual co-occurrences$\mathbf{s} = \sum_{i=1}^{K} \alpha_i \mathbf{q}_i$, where α are the combination coefficients (i.e. codes), $\mathbf{q}_i \in \Delta^{M-1}$ are co-occurrence words, modeled as points in the semantic simplex. We can learn $\mathbf{q}_i$ using a dictionary learning method, in our case simply the K-means algorithm, obtaining the co-occurrence dictionary $Q = [\mathbf{q}_1, \ldots, \mathbf{q}_K]$. Thus, Q forms a basis of co-occurrence words that quantizes the semantic space.

We can obtain the code $\bar{\alpha}$ by minimizing the square error between the new SMN $\mathbf{s}$ and the reconstructed SMN

$$\bar{\alpha} = \arg\min_{\alpha} \|\mathbf{s} - Q\alpha\|^2 \tag{9}$$

Although (9) has the analytic solution

$$\bar{\alpha} = \left(Q^T Q\right)^{-1} Q^T \mathbf{s} \tag{10}$$

in practice the inverse of Q is often too large to be solved directly, so it can be approximated. We can further assume that the combination of co-occurrences is sparse, and include a regularization term to enforce sparsity in the problem, such as

$$\bar{\alpha} = \arg\min_{\alpha} \|\mathbf{s} - Q\alpha\|^2 + \lambda \|\alpha\|_1 \tag{11}$$

Furthermore, we can also enforce locality, which will suggest that the co-occurrence words used to reconstruct a particular SMN should be those in its neighborhood. In particular, we use locality-constrained linear coding (LLC)[3] that enforces locality in the coding by solving the following problem

$$\bar{\alpha} = \arg\min_{\alpha} \|\mathbf{s} - Q_{\mathbf{s}}\alpha\|^2 \; st. \; a^T\mathbf{1} = 1 \tag{12}$$

where $Q_{\mathbf{s}}$ is a local basis composed by the k-nearest co-occurrence words to $\mathbf{s}$. This locality also leads to sparse codes.

Once we get the codes, we can project back to the semantic space to obtain the reconstructed SMN $\bar{\mathbf{s}}$=$Q\bar{\alpha}$. In this process we expect that $\bar{\mathbf{s}}$ contains mostly consistent co-occurrences, and hopefully little co-occurrence noise. Note that the CNF does not improve the classification accuracy by itself, so it still must be followed by a classifier. Figure 6e-f illustrates the co-occurrence dictionary and the filtered SMNs obtained for the samples of three categories from *15 scenes*, using 12 co-occurrences and LLC for coding. Note that SMNs are condensed in smaller regions. In this particular toy example, it is very unlikely that the classification accuracy improves, as classes are already clearly separated. However, in larger datasets, with a larger number of dimensions (i.e. categories), the filtering process can help to pull similar images to regions where categories can be easier to separate.

4.3. Patch-level processing and logarithmic scaling

So far, the strategy has been combining patch-SMNs into image-SMNs and filter co-occurrence noise by modeling consistent co-occurrences. However, category co-occurrences are already present in patch-SMNs, since co-occurrences are often local. Thus, it is also possible to process and filter patch-SMNs before they are combined into image-SMNs.

We observe first that patch-SMNs tend to have low entropy, that is, only few concepts are co-occurring. In order to increase the dynamic range we use log-probability rather than probability. For instance, using sparse encoding, the equivalent of (11) would be

$$\bar{\alpha} = \arg\min_{\alpha} \|\log \mathbf{s} - Q\alpha\|^2 + \lambda \|\alpha\|_1 \tag{13}$$

where $\log \mathbf{s} = (\log s_1, \ldots, \log s_M)^T$ indicates element-wise logarithm (abusing the notation). The filtered SMN is recovered as $\bar{\mathbf{s}}$=$\exp(Q\bar{\alpha})$ (exp is also element-wise). Then, voting can be used to infer image-SMNs.

Another alternative is working directly with log-SMNs rather than SMNs (see Figure 5c). In that case we reconstruct the filtered log-SMN as $\log \bar{\mathbf{s}} = Q\bar{\alpha}$. A suitable way to combine SMNs in the logarithmic scale is average pooling

$$\log \bar{\mathbf{s}} = \frac{1}{N} \sum_{n=1}^{N} \log \bar{\mathbf{s}}_n$$

equivalent to the product of SMNs in the linear scale.

The impact of modeling co-occurrences in a logarithmic scale and using patch-level processing are compared to previous image-level processing in Table 3. We use suitable kernels combined with SVMs, depending on the case (NGD kernel for SMNs and RBF kernel for log-SMNs). Both logarithmic scaling and patch modeling improve the performance. The best performance is achieved when both variations are combined.

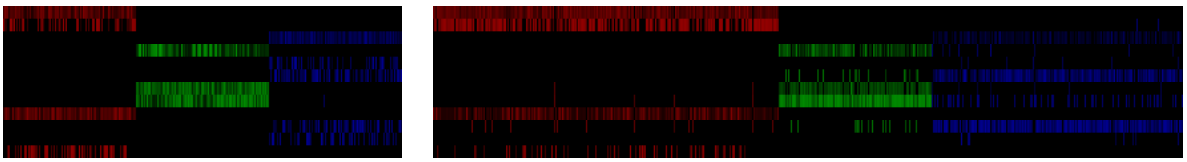
5. Semantic feature representations for large scale recognition

The proposed filtered SMNs (or log-SMNs) still have the limitation of requiring a kernel to obtain good discrimination with SVMs. In this section we introduce two new representations that can be used combined with linear classifiers.

Table 3: Impact of patch-level processing and logarithmic scale (*15 scenes*).

Method	Accuracy (%)						
	SM[b]	DMM[a]	CNF[b]	CNF (log)[b]	CNF (patch)[b]	CNF (patch, log)[c]	
Bayes (unfiltered)	65.5	66.6	65.5	65.5	-	-	
Bayes (filtered)	-	71.4	66.4	66.4	67.0	73.2	
SVM (kernel)	76.0 (NGD)	-	76.3 (NGD)	76.8 (RBF)	77.6 (NGD)	79.2 (RBF over $\log \bar{\mathbf{s}}$)	79.5 (NGD over $\bar{\mathbf{s}}$)

[a] Uses geometric mean to infer image-SMNs from patch-SMNs.
[b] Uses voting to infer image-SMNs from patch-SMNs.
[c] Uses average pooling to combine patch log-SMNs into image log-SMNs.



(a) (b)

Figure 7: Co-occurrence codes obtained for the images in the toy example of Figure 6: (a) training set, and (b) test set. Each column corresponds to an image, each row corresponds to a co-occurrence word.

5.1. Co-occurrence codes

An alternative to SMNs is using directly the coefficients $\bar{\alpha}$. As we discussed earlier, we model each SMNs as a (sparse) combination of co-occurrences. Then the space of co-occurrences is also suitable for semantic modeling of images. For example, using (11) on the toy example we obtain the codes shown in Figure 7. These codes show how different categories are represented as combinations of few co-occurrence words, with very clear patterns for each category. Note that co-occurrence codes are a higher level representation compared to traditional codes used in the BOW framework, which are obtained from lower-level visual features (i.e. visual words) rather than higher level semantic features (i.e. co-occurrence words). Thus, co-occurrence words may be more suitable for tasks involving complex images with higher-level concepts, such as scenes, while visual word codes may be more appropriate for lower level tasks where modeling visual appearance is enough for a good discrimination between classes.

In contrast to this toy example, we compute co-occurrence codes in a patch basis rather than in image basis. The codes are combined into an image representation, using average pooling

$$\bar{\alpha} = \frac{1}{N} \sum_{n=1}^{N} \bar{\alpha}_n \tag{14}$$

Another advantage of co-occurrence codes is that linear classifiers are often able to achieve good discrimination between categories, which makes these codes suitable for large scale scene recognition.

5.2. Kernel co-occurrence noise filter (KCNF)

5.2.1. Extension to the kernel space

We also can assume a mapping $\phi(\mathbf{s})$ to a different space (i.e. kernel space) where linear classification is more effective. Even if $\phi(\mathbf{s})$ is unknown or difficult to compute (i.e. infinite dimensional), a suitable kernel $k(\mathbf{s}, \mathbf{s}') = \phi(\mathbf{s})^T \phi(\mathbf{s}')$ can still be used. Focusing on the sparse formulation of (11), we can reformulate it in the kernel space as

$$\bar{\alpha} = \arg\min_{\alpha} \|\phi(s) - U\alpha\|^2 + \lambda \|\alpha\|_1 \tag{15}$$

where $U = [\phi(\mathbf{q}_1), \ldots, \phi(\mathbf{q}_K)]$ is the kernel dictionary, formed by the co-occurrence words in Q mapped to the kernel space. The filtered SMN in the kernel space is obtained as $\phi(\bar{\mathbf{s}}) = U\bar{\alpha}$.

In particular s, $\bar{s}$ and each q_i are defined over a probabilistic simplex, so a more suitable distance is the geodesic distance and a suitable kernel is the NGD kernel[33]. The problem is that there is no explicit formulation for the mapping $\phi_{NGD}(\mathbf{s})$, and U cannot be computed explicitly.

Similarly, filtering in the logarithmic scale we extend (13)

$$\bar{\alpha} = \arg\min_{\alpha} \|\phi(\log \mathbf{s}) - U\alpha\|^2 + \lambda \|\alpha\|_1 \tag{16}$$

In this case, log-SMNs are no longer in any simplex, so the NGD kernel is no necessarily suitable for this space. We tried other kernels, such as linear and radial basis functions (RBFs). Surprisingly, we found empirically that better performance is obtained when the input log-SMNs are renormalized and combined with the NGD kernel. These renormalized log-SMNs lie again in a (different) simplex so the NGD kernel seems suitable again. In the experiments, we include this renormalization before KCNF.

5.2.2. Computing the dictionary and embedding

Now we focus on the non-sparse case (i.e. $\lambda = 0$). Instead of learning the dictionary in the kernel space, we simply use K-means in the semantic space to learn Q (in fact, as we will see, we do not need to explicitly compute U). In order to use the geodesic distance between two filtered SMNs $\mathbf{s}$ and $\mathbf{s}'$, we formulate the whole filtering and embedding as a new kernel, obtained from the projected vectors on the NGD kernel space as

$$k_{KCNF}(\mathbf{s}, \mathbf{s}') = k_{NGD}\left(\bar{\mathbf{s}}, \bar{\mathbf{s}'}\right) = \phi_{NGD}(\bar{\mathbf{s}})^T \phi_{NGD}\left(\bar{\mathbf{s}'}\right) = (U\bar{\alpha})^T \left(U\bar{\alpha'}\right) \tag{17}$$

where $\bar{\alpha}$ and $\bar{\alpha'}$ are the codes for $\bar{\mathbf{s}}$ and $\bar{\mathbf{s}}'$. The corresponding analytic solution is $\bar{\alpha} = \left(U^T U\right)^{-1} U^T \phi_{NGD}(\bar{\mathbf{s}})$ (note that we do not have this solution for $\lambda \neq 0$). By applying some appropriate matrix transformations we can rearrange (17) as

$$k_{KCNF}(\mathbf{s}, \mathbf{s}') = \left(U^T \phi_{NGD}(\mathbf{s})\right)^T \left(U^T U\right)^{-1} \left(U^T \phi_{NGD}(\mathbf{s}')\right) \tag{18}$$

At this point we recall that $U = [\phi_{NGD}(\mathbf{q}_1), \ldots, \phi_{NGD}(\mathbf{q}_K)]$, so the first factor (transposed) and the third factor can be expressed as a kernel matrix

$$K_{qs}(\mathbf{s}) = U^T \phi_{NGD}(\mathbf{s}) = \begin{bmatrix} k_{NGD}(\mathbf{q}_1, \mathbf{s}) \\ \vdots \\ k_{NGD}(\mathbf{q}_K, \mathbf{s}) \end{bmatrix}$$

which depends only on $\mathbf{s}$ (or $\mathbf{s}'$), the co-occurrence words $\mathbf{q}_1, \ldots, \mathbf{q}_K$ and the NGD kernel. Similarly, the second factor (inverted) in (18) can be expressed as

$$K_{qq} = U^T U = \begin{bmatrix} k_{NGD}(\mathbf{q}_1, \mathbf{q}_1) & \cdots & k_{NGD}(\mathbf{q}_N, \mathbf{q}_1) \\ \vdots & & \vdots \\ k_{NGD}(\mathbf{q}_1, \mathbf{q}_N) & \cdots & k_{NGD}(\mathbf{q}_N, \mathbf{q}_N) \end{bmatrix}$$

leading to (18) represented as $k_{KCNF}(s, s') = K_{qs}(s)^T K_{qq}^{-1} K_{qs}(s')$. Note that no explicit mapping ϕ_{NGD} is necessary to compute any of those matrices. As K_{qq} is positive definite, we can find a decomposition $G^T G = K_{qq}^{-1}$ (e.g. using the Cholesky decomposition), which leads to

$$k_{KCNF}(\mathbf{s}, \mathbf{s}') = (GK_{qs}(\mathbf{s}))^T (GK_{qs}(\mathbf{s}')) = \phi_{KCNF}(\mathbf{s})^T \phi_{KCNF}(\mathbf{s}')$$

where we have an explicit mapping

$$\phi_{KCNF}(\mathbf{s}) = GK_{qs}(\mathbf{s}) \tag{19}$$

that only depends on a set of co-occurrences $\mathbf{q}_i$ and the original (unfiltered) SMN $\mathbf{s}$ and $\mathbf{s}'$. With (19) we can obtain embeddings of semantic features that can be used for large scale classification.

Note that (19) is an *exact* embedding that integrates both co-occurrence noise filtering and mapping to a

suitable space. We do not approximate the implicit mapping in the NGD kernel $k_{NGD}(\mathbf{s}, \mathbf{s}')$, as in [14], but develop a kernelized version of the CNF integrated with the geodesic distance. Another advantage is that we do not need to explicitly compute the codes $\overline{\alpha}$ in (9).

As discussed earlier, we can also filter patch-SMNs rather than image-SMNs. However, the embedded features do not have a probabilistic interpretation anymore, so we use simple average pooling to combine $\phi_{KCNF}(\mathbf{s}_n)$, which we found has satisfactory performance in practice. Thus, the resulting image-level embedding is

$$\varphi_{KCNF}(\mathbf{s}) = \frac{1}{N} G \left(\sum_{n=1}^{N} \sum_{i=1}^{K} k_{KCNF}(\mathbf{s}_n, \mathbf{q}_i) \right) \tag{20}$$

We refer to this method as *kernel co-occurrence noise filter* (KCNF). Similarly, the same formulation (with $\lambda = 0$) can be used in the logarithmic scale, where $\log \mathbf{s}_n$ are the inputs to the KNCF. Note that the NGD kernel is no longer suitable, but we can still obtain an exact embedding for other kernels, such as RBF, that in other cases would need to be approximated. Figure 5d shows the architecture for large scale classification using patch-level KCNF in the logarithmic scale.

6. Experiments

6.1. Experimental setup

Datasets. We evaluated our method on three small datasets. *15 scenes*[9, 18] consists of 4485 natural scene images labeled into 15 categories. *LabelMe*[34] contains 8 outdoor scene categories, with a total of 2600 color images. *UIUC-Sports*[35] consists of eight sport event categories, and each category has 137 to 250 images. We resize large images to no more than 300×200. Following settings in previous works, we use 100, 100 and 70 images for training, respectively.

For large scale evaluation we used the *MIT67*[22] and *SUN397*[36] datasets, using the suggested training/test configurations. *MIT67* includes 67 indoor scene categories and 15620 images in total. The similarity of the objects present in different indoor scenes makes it an especially difficult dataset compared to outdoor scene datasets. *SUN397* consists of 397 categories and 108762 images in total.

Visual and semantic features. We use kernel descriptors[15] for all local visual features extracted on a regular 16×16 pixel dense grid (step 8 pixels). For each patch we extract gradient, shape and color kernel descriptors. For semantic features (i.e. SMN) we train GMM with 512 mixtures for each concept and visual feature. In methods using SVM classification (i.e. all but Bayes) we also extend the descriptor using a spatial pyramid[18] with four levels (1×1, 2×2, 3×3, 4×4).

Baselines. We compare our approach with two BOW methods in the visual space, and also with related methods in the semantic space using SMNs. The baselines in the visual space are:

- *LLC*[3]: visual features are encoded with a dictionary using LLC and then classified with a linear SVM.
- *EMK*[37]: visual features are encoded with efficient match kernels[37] and classified with a linear SVM.

The corresponding **baselines in the semantic space** are (the input features are SMNs):

- *Bayes*[13]: selects the category with higher probability in the image-SMN.
- *SPMSM*[14]: combines SPM and non-linear SVM with the NGD kernel.
- *SPMSM (embedding)*: uses a linear SVM and the embedding in [14] to approximate the NGD kernel.

Variations of the proposed methods. We evaluate:

- *CNF*: proposed framework including the CNF. We use non-linear SVM with the NGD kernel over filtered SMNs.
- *CNF* (embedding): CNF using the same approximate embedding of *SPMSM (embedding)*.
- *KCNF*: proposed framework including the KCNF and linear SVM.
- *Co-codes*: uses a linear SVM over co-occurrence codes.

LLC, *CNF* and *co-codes* use LLC as coding method. Its main parameter is the number of neighbors, which we set to 5.

6.2. Multiple feature combination

We evaluate *Bayes* classification and SVM classification with SPM and *KCNF*. We also compare with methods in the visual space (*LLC* and *EMK*). From earlier to later combination:

- *Concatenation (visual)*: patch descriptors are concatenated in the visual space (just before the GMM model) as $x_n = \left[x_n^{\text{gradient}}, x_n^{\text{shape}}, x_n^{\text{color}}\right]$.
- *Semantic space*: proposed method using (8). For $P_{W|W^v}(w|w^v)$ we evaluate both *uniform* prior ($P_{W|W^v}(w|w^v) = {}^1/_{|V|}$) and empirically estimated (*adapted*).
- *Concatenation (KCNF)*: concatenation of multiple image descriptors in the kernel space, which is denoted as $\varphi_{KCNF}(s) = \left[\varphi_{KCNF}^{\text{gradient}}(s), \varphi_{KCNF}^{\text{shape}}(s), \varphi_{KCNF}^{\text{color}}(s)\right]$. For the baseline in the visual space, we use the same scheme.
- *Multi-kernel learning (MKL)*: features are combined in the SVM via a weighted kernel sum. We use Simple MKL[38] with the default settings (ten Gaussian and three polynomial kernels).

Table 4 shows the results for the different datasets, combination schemes and settings. At the SMN level, the proposed Bayesian combination method consistently improves the classification accuracy with total gains around 4.5% over the best single feature when $P_{W|W^v}(w|w^v)$ is estimated. The gain due to this estimation varies more, from 1% to 4%. Note that the dimension of the descriptor does not change.

We also evaluated multi-feature combination in the proposed KCNF framework. We set the size of the co-occurrence dictionary to $K = 1000$. In the iterative estimation of the weights we observed that at the second iteration the improvement in accuracy is almost imperceptible, so in practice we just do two iterations. The proposed combination in the semantic space with adapted weights is again the method with best results with an overall gain around 2-3.5%. In this case, the improvement over equal weights is lower than 1%. Note that concatenation in the KCNF has similar accuracy to *Bayesian (uniform)*. However, in this case the dimensionality increases by the number of features. The performance of MKL is around 2% worse than the proposed method, while it requires considerably more training time and is much more complex. The proposed method also outperforms concatenation in visual spaces.

6.3. Co-occurrence modeling and filtering

In this section we evaluate the impact of the proposed tools for co-occurrence modeling and co-occurrence noise filtering on the classification accuracy and the relation with the size of the co-occurrence dictionary. The results for the different datasets are shown in Figure 8. We included *SPMSM (linear)* as a simplified version of *SPMSM* using linear SVM, to show the gain due to the NGD kernel. The results also show that using CNF significantly and consistently outperforms the counterpart without CNF (i.e. *SPMSM*).

The size of the co-occurrence dictionary is also critical for high accuracy, but even small sizes like $K = 300$ can help to gain around 1-2% in accuracy. We also observed an increasing performance with K and then a moderate decrease. We also observe an optimum dictionary size, around $K = 3000$ for *15 scenes* and $K = 2400$ for *LabelMe* and *Sports*. Note that the number of categories in *15 scenes* is 15 and in the other two dataset is 8. This result suggests that the more concepts the more number of co-occurrence words required to model co-occurrence patterns. In fact, when K is too low, the model complexity is also too low and genuine co-occurrences are also filtered, while when K is too high, the CNF also models undesired co-occurrence noise resulting in overfitting and the corresponding decrease in performance.

The performance of CNF and co-codes also depends on the number of nearest neighbors B, since both of them use LLC coding. Since [3] has shown that B is not affected by the size of dictionary, we use the same $K = 2400$ for this evaluation. We evaluate the impact of the nearest neighbor B in a range from 2 to 40 nearest neighbors (see Figure 9). The best performance is achieved for $B = 3$ and $B = 10$, for CNF and co-codes, respectively.

Table 4: Classification accuracy (%) for the different feature combination methods.

Feature combination method	15 scenes	LabelMe	Sports
	Visual features (EMK+SVM, $K = 1000$)		
Gradient KDES	80.8	85.9	82.8
Shape KDES	78.9	83.5	83.1
Color KDES	72.1	72.1	72.5
Concatenation	82.2	87.3	85.6
	Visual features (LLC+SVM, $K = 1000$)		
Gradient KDES	81.4	87.4	82.3
Shape KDES	81.4	85.5	83.9
Color KDES	68.2	69.5	69.2
Concatenation	82.8	88.4	86.1
	Semantic features (Bayes classifier)		
Gradient SMN	73.7	81.8	78.3
Shape SMN	70.3	79.5	79.2
Color SMN	69.1	67.3	69.8
Semantic space (uniform)	77.5	82.4	82.5
Semantic space (adapted)	**79.7**	**86.5**	**83.9**
	Semantic features (KCNF+SVM, $K = 1000$)		
Gradient SMN	78.9	86.5	83.7
Shape SMN	80.0	85.0	84.3
Color SMN	75.4	72.4	72.8
Concatenation (visual)	81.8	84.0	84.6
Semantic space (uniform)	82.7	88.6	85.7
Semantic space (adapted)	**84.4**	**89.3**	**86.9**
Concatenation (KCNF)[a]	82.1	87.3	84.9
Concatenation (KCNF)[b]	83.3	88.7	85.9
MKL	81.6	87.2	84.7

[a] We use $K = 334$ for an aggregated output dimension of $3 \times K \approx 1000$.
[b] We use $K = 1000$ (aggregated output dimension $3 \times K = 3000$).

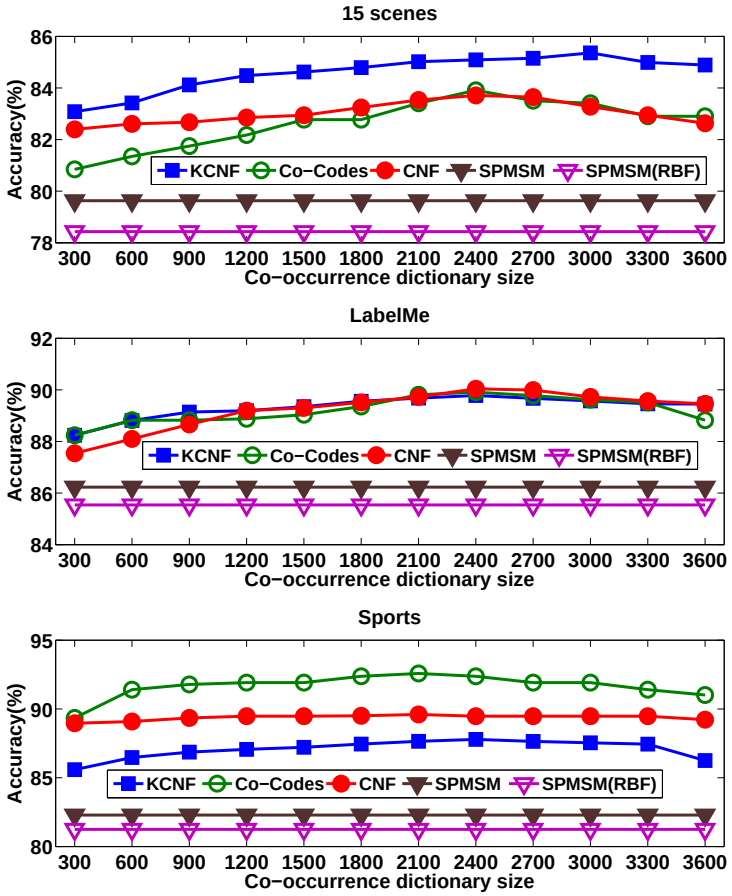


Figure 8: Performance of the CNF-SM on different dictionary sizes.

Table 5: Comparison with previous works on small scale datasets in accuracy (%).

	Method	15 scenes	LabelMe	Sports
Proposed	CNF-SMN (K=2400)	82.9	**90.0**	89.6
	Co-codes (K=2400)	83.1	89.9	**92.3**
	KCNF (K=3000)	85.2	89.8	87.8
State-of-the-art	Latent Dirichlet Allocation (LDA)[12]	76.6	-	-
	Contextual multinomial (CMN)[13]	77.2	-	-
	Object bank[11]	80.9	-	76.3
	Spatial pyramid[18]	81.2	-	-
	LLC[3]*	81.6	86.1	85.0
	FV[39]*	81.9	87.6	85.6
	Kernel descriptors[15]*	82.2	87.3	85.2
	SPMSM[14]	82.3	87.5	83.0
	ImageNet-CNN[17]	84.2	-	**94.4**
	ISPR[23]	85.1	-	89.5
	Object-to-Class kernels[40]	88.8	-	86.0
	IFV[23]	89.2	-	90.8
	Places-CNN[17]	**90.2**		94.1

* Based on the code provided by the authors. Other accuracies are those reported in the corresponding papers.

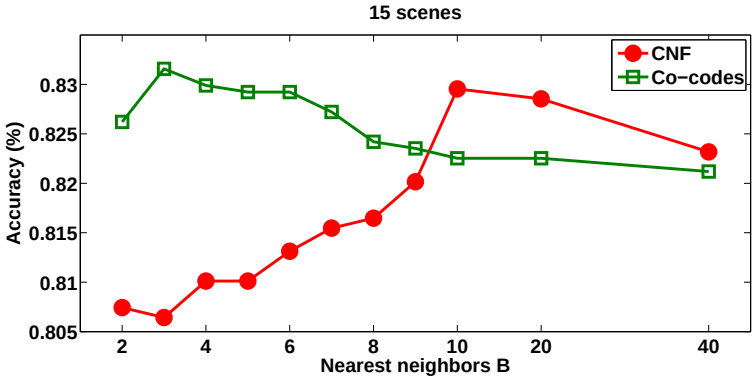


Figure 9: Performance of the CNF on different nearest neighbors.

6.4. Comparison with state-of-the-art on small datasets

The proposed approach is compared with several state-of-the-art BOW methods [18, 3, 39, 15], methods based on the SMN framework[13, 14, 12] and methods using other intermediate semantic representations[11, 40]. Whenever possible we used the code available from the authors with the same features used in our experiments; when not possible, we include the results reported in their papers. Although a completely fair comparison is not possible, due to different implementations, dictionary sizes and other parameters, our framework at least seems to be very competitive in the three evaluated datasets. Comparing with methods based on SMNs is of particular interest. Note that all proposed methods outperform the contextual multinomial (CMN) and SPMSM. The latter integrates both discriminative co-occurrence modeling via SVM and rough spatial coding via SPM, achieving better performance, but does not integrate unsupervised co-occurrence modeling. Combining these three characteristics and multiple features our method achieves better performance.

We also report results for methods based on the improved Fisher vector (IFV), mining discriminative parts (ISPR)[23] and CNNs, some of them outperforming the proposed ones. For *15 scenes* we obtain an accuracy comparable to ImageNet-CNN, even when this method exploits a large external dataset. Places-CNN obtains much better performance using the scene-centric dataset places[17], demonstrating the importance of the pre-training dataset in CNNs . For *Sports*, co-codes obtains a very remarkable accuracy of 92.3% only outperformed by CNNs.

6.5. Evaluation on larger datasets

We also evaluated the proposed methods on the larger datasets *MIT67[22]* and *SUN397*[36]. Training SVM classifiers with non-linear kernels is impractical for large datasets, so we only consider linear SVMs and embeddings (exact or approximate) when available. KCNF and co-codes can be used directly with linear classifiers. In the case of CNF we use an approximate embedding for the NGD kernel instead of the kernel itself[14]. For the same complexity reasons, we fixed $K = 2000$.

The results for the mid-scale dataset *MIT67* are shown in Table 6. Comparing with Bayes classification, the proposed multi-feature combination method improves around 8% the accuracy. KCNF obtains a very remarkable accuracy of 48.1%, outperforming BOW methods and other object-based approaches, while recent methods based on CNNs, FV[5, 41, 42] and mining discriminative parts[23] and mode seeking[43]) are better. DMM-FV[41] is also based on FV, using Dirichlet-derived GMMs for encoding. LASC[42] improves LLC with an affine subspace dictionary, and concatenates first-order (LLC-based) and second-order (FV-based) features. On the one hand, since the number of categories and the complexity in this dataset is much higher, CNNs benefit more from pre-training data, in particular from the Places dataset. On the other hand, FV and part-based approaches are particularly suited for modeling objects, achieving very good recognition performance in object datasets. It is not surprising their superior performance for MIT67, which contains indoors scenes, rich in objects and where recognizing certain key objects or parts can be very helpful to predict the scene category.

Table 6: Comparison on MIT67 dataset.

MIT67	Method	Accuracy (%)
SMN+Bayes	Gradient	29.7
	Shape	28.9
	Color	20.8
Proposed	Bayesian (multi-feature uniform)	34.0
	Bayesian	39.7
	CNF (embedding, K=2000)	42.1
	Co-codes (K=2000)	42.6
	KCNF (K=2000)	**48.1**
State-of-the-art	Object bank[7]	37.6
	Object-to-Class kernels[40]	39.6
	Deformable part-based models[44]	43.1
	SPMSM[14]	44.0
	Linear Distance Coding[4]	46.7
	ISPR[23]	50.1
	ImageNet-CNN[17]	56.8
	IFV[23]	60.8
	DMM-FV[41]	63.4
	LASC[42]	63.4
	Mode seeking[43]	64.0
	Places-CNN[17]	68.2

* Multi-feature using adapted weights, unless specified otherwise.

The results for SUN397 are shown in Table 7. The gain with the proposed methods is also higher than in the previous datasets, suggesting that the co-occurrence problem is more pronounced when the number of categories is high (i.e. high dimensional semantic simplices), and recognition in the semantic space benefits more from appropriate modeling of consistent co-occurrences and filtering undesired noise. In particular, filtering in the kernel space has a very significant gain (compare *KCNF* and *SPMSM*), suggesting than a suitable distance and sparsity are both critical in this framework. Also note that *KCNF* achieves comparable performance to CNNs trained over ImageNet[17], although still far from the performance of FV and Places-CNN. Note that this high performance of CNNs is due to the use of much larger external datasets (ImageNet or Places), which we do not leverage.

Table 7: Comparison on SUN397 dataset.

SUN397	Method	Accuracy (%)
SMN+Bayes	Gradient	18.8
	Shape	16.9
	Color	12.2
Proposed	Bayesian (multi-feature uniform)	21.6
	Bayesian	26.1
	SPMSM (embedding)	30.0
	KCNF single-feature (gradient, K=2000)	33.0
	CNF (embedding, K=2000)	33.2
	Co-codes (K=2000)	35.4
	KCNF (K=2000)	**40.8**
State-of-the-art	Xiao et al (best single feature: HOG)[36]	27.2
	SPMSM[14]	28.9
	Meta-classes[8]	36.8
	Xiao et al (multi kernel, 14 features)[36]	38.0
	ImageNet-CNN[17]	42.6
	LASC[42]	45.3
	DMM-FV[41]	46.1
	FV[5]	47.2
	Places-CNN[17]	**54.32**

6.6. Embedding

Finally, we include a more detailed comparison (see Table 8) of the descriptors embeddings in the semantic space for large scale classification: the approximate NGD embedding[14] in SPMSM (with and without CNF), co-codes and KCNF. All the proposed methods are evaluated for K=2000. As expected, *KNCF* also outperforms the SPMSM version with embedding, with a gain around 4-6% in small datasets and 9-14% in larger datasets. Thus, KCNF can be used with linear classifiers while keeping a high accuracy. Note that KCNF increases the dimensionality from the number of classes (between 8 and 397 in these datasets) to the number of co-occurrences in the dictionary (300-3000), while the approximation in [14] does not. However, the dimensionality is still reasonable for SVM. Even with low K the proposed embedding still outperforms the other methods.

Table 8: Comparison of different embeddings in the semantic space.

Method	K=2000				
	15 scenes	LabelMe	Sports	MIT67	SUN397
SPMSM (emb)	78.9	85.9	81.3	37.6	30.0
Co-codes	83.1	**89.7**	**92.3**	42.1	35.4
CNF (emb)	82.9	89.6	89.4	42.6	33.2
KCNF	**84.9**	89.6	87.5	**48.1**	**40.8**

7. Conclusion

Representing images in a scene-level provides a higher level of abstraction that may be helpful for recognizing complex scenes. While visual representations and related recognitions have been largely investigated, many aspects related with semantic representations still remain largely unexplored. In this paper we propose a different point of view to scene modeling. In our framework, we avoid the problems related with mid-level representations (e.g. mid-level annotation, discovering of latent models) by directly using image labels to learn patch models. This weakly supervised learning results in scene categories co-occurring in the representation. As in traditional mid-level

representations, we use two levels, but they are not progressive (in terms of level of abstraction), and the second is focusing on correcting a posteriori the ambiguity related with category co-occurrences.

We also showed how the scene category co-occurrence point of view brings new ways to address old problems. Exploiting the common semantic space, we can combine information from heterogeneous visual features and combine local patches into global image representations. We can also pose problems in terms of modeling category co-occurrences, where we represent images in a semantic space as a sparse combination of co-occurrence words (i.e. co-codes), and find consistent co-occurrence patterns by separating genuine co-occurrence patterns from noise (i.e. co-occurrence noise filter). Furthermore, we can develop features and embeddings specifically designed for the semantic space that can be efficient for large scale recognition. In particular, our method shows very competitive results in this scenario, particularly in large datasets, where the number of categories is high and the category co-occurrences are sparser.

Acknowledgment. This work was supported in part by National Basic Research Program of China (973 Program): 2012CB316400, in part by the National Natural Science Foundation of China: 61322212 and 61550110505, in part by the National Hi-Tech Development Program (863 Program) of China: 2014AA015202, in part by the Key Technologies R&D Program of China under Grant no. 2012BAH18B02. This work is also funded by Lenovo Outstanding Young Scientists Program (LOYS).

References

[1] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2004) 91–110. 1

[2] J. Yang, K. Yu, Y. Gong, T. S. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: CVPR, 2009. 1

[3] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: CVPR, 2010. 1, 1, 4.2, 6.1, 6.3, 5, 6.4

[4] Z. Wang, J. Feng, S. Yan, H. Xi, Linear distance coding for image classification, IEEE Trans. on Image Process. 22 (2) (2013) 537–548. doi:10.1109/TIP.2012.2218826. 1, 6

[5] J. Sanchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the fisher vector: Theory and practice, Int. J. Comput. Vision 105 (3) (2013) 222–245. doi:10.1007/s11263-013-0636-x. 1, 1, 6.5, 7

[6] J. Vogel, B. Schiele, Semantic modeling of natural scenes for content-based image retrieval, Int. J. Comput. Vision 72 (2) (2007) 133–157. doi:10.1007/s11263-006-8614-1. 1, 1, 2.1

[7] L. Li, H.Su, E. Xing, L. Fei-Fei, Object bank: A high-level image representation for scene classification and semantic feature sparsification, in: NIPS, 2010. 1, 6

[8] A. Bergamo, L. Torresani, Classemes and other classifier-based features for efficient object categorization, in: IEEE Trans. on Pattern Anal. and Mach. Intell., 2014. 1, 2.1, 7

[9] L. Fei-Fei, P. Perona, A bayesian hierarchical model for learning natural scene categories, in: CVPR, 2005. 1, 2.1, 2.2, 6.1

[10] Z. Niu, G. Hua, X. Gao, Q. Tian, Context aware topic model for scene recognition, in: CVPR, 2012. 1, 2.1, 2.2

[11] L.-J. Li, H. Su, Y. Lim, L. Fei-Fei, Object bank: An object-level image representation for high-level visual recognition, Int. J. Comput. Vision 107 (1) (2014) 20–39. doi:10.1007/s11263-013-0660-x. 1, 1, 2.1, 5, 6.4

[12] N. Rasiwasia, N. Vasconcelos, Latent dirichlet allocation models for image classification, IEEE Trans. on Pattern Anal. and Mach. Intell. 35 (11) (2013) 2665–2679. doi:10.1109/TPAMI.2013.69. 1, 2.1, 2.2, 5, 6.4

[13] N. Rasiwasia, N. Vasconcelos, Holistic context models for visual recognition, IEEE Trans. on Pattern Anal. and Mach. Intell. 34 (5) (2012) 902–917. 1, 1, 1, 2.1, 2, 3.1, 3.1, 3.1, 5, 3.2, 3.2.1, 6.1, 5, 6.4

[14] R. Kwitt, N. Vasconcelos, N. Rasiwasia, Scene recognition on the semantic manifold, in: ECCV, 2012. 1, 1, 2.1, 2, 2.2, 3, 3.1, 3.1, 5, 3.1, 3.2.1, 5.2.2, 6.1, 5, 6.4, 6.5, 6, 7, 6.6

[15] L. Bo, X. Ren, D. Fox, Kernel descriptors for visual recognition, in: NIPS, 2010. 1, 6.1, 5, 6.4

[16] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: NIPS, 2012, pp. 1106–1114. 1, 2.1, 2.2

[17] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva, Learning deep features for scene recognition using places database, in: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Weinberger (Eds.), NIPS, 2014, pp. 487–495. 1, 2.1, 5, 6.4, 6, 6.5, 7

[18] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: CVPR, 2006. 2.1, 3.1, 6.1, 5, 6.4

[19] A. Farhadi, I. Endres, D. Hoiem, D. A. Forsyth, Describing objects by their attributes, in: CVPR, 2009. 2.1

[20] C. H. Lampert, H. Nickisch, S. Harmeling, Attribute-based classification for zero-shot visual object categorization, IEEE Trans. on Image Process. 36 (3) (2014) 453–465. 2.1

[21] G. Patterson, C. Xu, H. Su, J. Hays, The sun attribute database: Beyond categories for deeper scene understanding, Int. J. Comput. Vision 108 (1-2) (2014) 59–81. 2.1

[22] A. Quattoni, A. Torralba, Recognizing indoor scenes, in: CVPR, 2009. 2.1, 6.1, 6.5

[23] D. Lin, C. Lu, R. Liao, J. Jia, Learning important spatial pooling regions for scene classification, in: CVPR, 2014, pp. 3726–3733. doi:10.1109/CVPR.2014.476. 2.1, 5, 6.4, 6.5, 6

[24] Z. Zuo, G. Wang, B. Shuai, L. Zhao, Q. Yang, Exemplar based deep discriminative and shareable feature learning for scene image classification, Pattern Recognition 48 (10) (2015) 3004–3015. 2.1

[25] X. Liu, W. Yang, L. Lin, Q. Wang, Z. Cai, J. Lai, Data-driven scene understanding with adaptively retrieved exemplars, IEEE Trans. on Multimedia 22 (3) (2015) 82–92. 2.1

[26] N. Rasiwasia, N. Vasconcelos, Bridging the gap: Query by semantic example, IEEE Trans. on Multimedia 9 (5) (2007) 923–938. 2.1, 3, 3.1

[27] A. Vedaldi, A. Zisserman, Efficient additive kernels via explicit feature maps, IEEE Trans. on Pattern Anal. and Mach. Intell. 34 (3) (2012) 480–492. doi:10.1109/TPAMI.2011.153. 2.2

[28] J. van de Weijer, C. Schmid, Coloring local feature extraction, in: ECCV, 2006. 2.3

[29] F. Shahbaz Khan, J. van de Weijer, M. Vanrell, Top-down color attention for object recognition, in: ICCV, 2009, pp. 979–986. doi:10.1109/ICCV.2009.5459362. 2.3

[30] L. Lin, P. Luo, X. Chen, K. Zeng, Representing and recognizing objects with massive local image patches, Pattern Recognition 45 (1) (2012) 231–240. 2.3

[31] M.-E. Nilsback, A. Zisserman, Automated flower classification over a large number of classes, in: ICVGIP, 2008, pp. 722–729. 2.3

[32] L. Lin, R. Zhang, X. Duan, Adaptive scene category discovery with generative learning and compositional sampling, IEEE Trans. on Circuits and Systems for Video Technology 25 (2) (2015) 251–260. 2.3

[33] D. Zhang, X. Chen, W. S. Lee, Text classification with kernels on the multinomial manifold, in: RDIR, 2005, pp. 266–273. doi:10.1145/1076034.1076081. 3.1, 5.2.1

[34] A. Oliva, A. Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, Int. J. Comput. Vision 42 (3) (2001) 145–175. doi:10.1023/A:1011139631724. 6.1

[35] L. F.-F. L.J. Li, What, where and who? classifying events by scene and object recognition, in: ICCV, 2007. 6.1

[36] J. Xiao, J. Hayes, K. Ehringer, A. Olivia, A. Torralba, SUN database: Largescale scene recognition from abbey to zoo, in: CVPR, 2010. 6.1, 6.5, 7

[37] L. Bo, C. Sminchisescu, Efficient match kernel between sets of features for visual recognition, in: NIPS, 2009. 6.1

[38] A. Rakotomamonjy, F. Bach, Y. Grandvalet, S. Canu, SimpleMKL, J. Mach. Learn. Res. 9 (2008) 2491–2521. 6.2

[39] F. Perronnin, J. Sanchez, T. Mensink, Improving the fisher kernel for large-scale image classification, in: ECCV, 2010. 5, 6.4

[40] L. Zhang, X. Zhen, L. Shao, Learning object-to-class kernels for scene classification, IEEE Trans. on Image Process. 23 (8) (2014) 3241–3253. 5, 6.4, 6

[41] T. Kobayashi, Dirichlet-based histogram feature transform for image classification, in: CVPR, 2014. 6.5, 6, 7

[42] P. Li, X. Lu, Q. Wang, From dictionary of visual words to subspaces: Locality-constrained affine subspace coding, in: CVPR, 2015. 6.5, 6, 7

[43] C. Doersch, A. Gupta, A. A. Efros, Mid-level visual element discovery as discriminative mode seeking, in: NIPS, 2013, pp. 494–502. 6.5, 6

[44] M. Pandey, S. Lazebnik, Scene recognition and weakly supervised object localization with deformable part-based models, in: ICCV, 2011. 6