A Reconfigurable Tangram Model for Scene Representation and Categorization

Jun Zhu, Tianfu Wu†, Song-Chun Zhu, *Fellow, IEEE,* Xiaokang Yang *Senior Member, IEEE,* and Wenjun Zhang, *Fellow, IEEE*

***Abstract*—This paper presents a hierarchical and compositional scene layout (i.e., spatial configuration) representation and a method of learning reconfigurable model for scene categorization. Three types of shape primitives (i.e., triangle, parallelogram and trapezoid), called "tans", are used to tile scene image lattice in a hierarchical and compositional way, and a directed acyclic And-Or graph (AOG) is proposed to organize the overcomplete dictionary of tan instances placed in image lattice, exploring a very large number of scene layouts. With certain "off-the-shelf" appearance features used for grounding terminal-nodes (i.e., tan instances) in the AOG, a scene layout is represented by the globally optimal parse tree learned via a dynamic programming algorithm from the AOG, which we call tangram model. Then, a scene category is represented by a mixture of tangram models discovered with an exemplar-based clustering method. On basis of the tangram model, we address scene categorization in two aspects: (i) Building a "tangram bank" representation for linear classifiers, which utilizes a collection of tangram models learned from all categories, and (ii) Building a tangram matching kernel for kernel-based classification, which accounts for all hidden spatial configurations in the AOG. In experiments, our methods are evaluated on three scene datasets for both the configuration-level and semantic-level scene categorization, and outperform the spatial pyramid model consistently.**

***Index Terms*—Tangram Model, Scene Layout, And-Or Graph, Dynamic Programming, Scene Categorization.**

I. Introduction

A. Motivation and objective

Recent psychological experiments have shown that human visual system can recognize categories of scene images (such as streets, bedrooms) in a single glance (often less than 80ms) by exploiting the spatial layout [1], [2], [3], [4], and human can memorize thousands of scene configurations in an effective and compact way [5]. Generally, a scene consists of visual constituents (e.g., surfaces and objects) arranged in a meaningful and reconfigurable spatial layout. From the perspective of scene modeling, one may ask what representation facilitates scene categorization based on spatial layout? In the literature of scene categorization by computer vision, most work [6], [7], [8], [9] adopt a predefined and fixed spatial pyramid which is a quad-tree like representation for scene layouts (see Fig. 1 (a)), and then rely on rich appearance features for improving performance.

J. Zhu, X. Yang and W. Zhang are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China (e-mail:zhujun.sjtu@gmail.com,xkyang@sjtu.edu.cn,zhangwenjun@sjtu.edu.cn).

T. Wu and S.-C. Zhu are with the Department of Statistics, University of California, Los Angeles (email:{tfwu,sczhu}@stat.ucla.edu).

†T. Wu is the corresponding author.

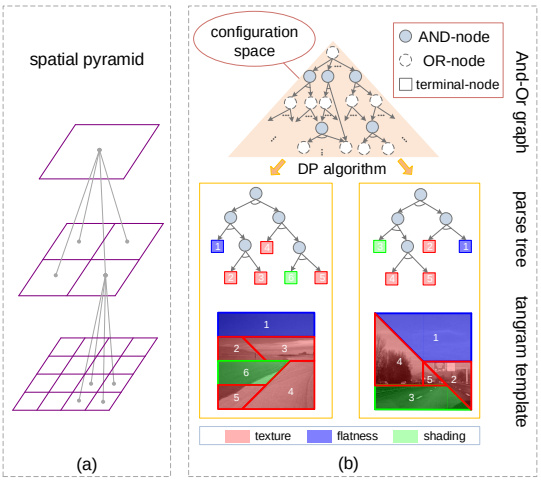


Fig. 1. (a) Illustration of a 3-layer spatial pyramid, which is a quad-tree like scene layout representation. (b) Illustration of our tangram model on scene layout representation. In the tangram model, we represent scene layout by an explicit template composed of a small number of tan instances (i.e., tangram template), for capturing meaningful spatial layout and appearance (we use different color to illustrate the appearance models for diverse visual patterns of tan instances, such as texture, flatness and shading surfaces). The tangram template is collapsed from a reconfigurable parse tree, which is adaptive to the configuration of different scene layouts. In this paper, we propose a DP algorithm to seek the globally optimal tangram model, from the configuration space defined based on an And-Or graph of tan instances. See Sec. I-A for details. (Best viewed in color)

In this paper, we address the issue above by leveraging a hierarchical and compositional model for representing scene layouts. Our method is motivated by recent progress made in object modeling, for which compositional hierarchical models [10], [11], [12] have shown increasing significance such as the deformable part-based model [13] and the stochastic And-Or templates [14]. The success lies in that they are capable of learning reconfigurable representation to account for both structural and appearance variations.

The proposed model on scene layout representation has a very intuitive explanation analogous to "tangram", which is an ancient invention from China. Literally, the tangram is called "seven boards of skill" which can form a large number of object shapes by arranging seven boards (so-called "tans") in different spatial layouts. We use three types of shape primitives (i.e., triangle, parallelogram including rectangle, and trapezoid) to tile scene image lattice which play roles analogous to the roles of tans in tangram, so we call our scene model *tangram model*. It often consists of a small number of tan instances of different shape types partitioning the scene image lattice.

Our tangram model has two characteristics as follows:

(i) *Compactness*. It entails a sparse representation on image lattice to capture meaningful scene layouts. As illustrated in the bottom-right corner of Fig. 1 (b), our tangram model for a highway scene consists of five tan instances, which can capture the scene configuration in a compact yet meaningful way. Note that we introduce triangle in our tan types to gain sparser representation on the object or surface boundaries in scene images. Meanwhile, we currently do not use different types of curve shapes because (1) we need to keep our tans simple and generic, and (2) we focus on the scene categorization task rather than pixel-level scene labeling or region-level parsing.

(ii) *Reconfigurability*. To account for various scene categories and large intra-class variations of spatial layouts (i.e., sub-categories), it entails adaptivity in representation and selectivity in learning. Our tangram model is learned from the quantized configuration space of scene layout by a dynamic programming algorithm. Hence, it is adaptive to different scene layouts (see another example of our tangram model for a coast scene layout in the bottom-left corner of Fig. 1 (b)).

B. Method overview

In this paper, the learning of our tangram model consists of five components as follows, which are also our main contributions to the field of scene representation and categorization.

(i) Hierarchical and compositional quantization on the configuration space of scene layouts. For a given scene image lattice, we first generate a variety of tans (i.e., shape primitives) with different scales through recursive shape composition, and then enumerate all valid instances of the tans by placing them at different locations. Thus, we can construct an overcomplete dictionary of tan instances (as the "parts" of decomposing scene layouts, see Fig. 3 for illustration) for "quantizing" the configuration space of scene layouts. We organize all the tan instances into an *And-Or graph* (AOG) structure by exploiting their compositional relationships, as illustrated in Fig. 4. There are three types of nodes in our AOG: (1) An AND-node represents the decomposition of a sub-lattice into two child ones, (2) An OR-node represents alternative ways of decomposing the same sub-lattice (which can terminate directly to the tan instance or use one of different decompositions represented by AND-nodes), and (3) A terminal-node represents a tan instance which links to image data in practice. Through traversing the AOG from the root OR-node, we obtain a reconfigurable parse tree from the AOG. As shown in Fig. 1 (b), the *parse tree* is a binary tree composed of non-terminal pass-by nodes and terminal leaf nodes. See Sec. II for details.

(ii) Learning a tangram template from roughly aligned scene images by a dynamic programming algorithm. In corporation with certain "off-the-shelf" feature to describe the appearance of image data for a tan instance, we present a *tangram template* to model the scene layout explicitly (see Sec. III-A and III-B for details). Suppose a set of roughly aligned scene images are given (i.e., images which share similar scene layout). we present a generative formulation of learning tangram template under *information projection principle* [15] and propose a *dynamic programming* (DP) algorithm for seeking the globally optimal parse tree in the AOG. Through collapsing the parse tree onto image lattice, we obtain the tangram template. The DP algorithm consists of two successive steps: (1) A bottom-up step computes the information gains (i.e. the log-likelihood ratio defined in Sec. IV-A) for the nodes of the AOG and determines the optimal state for each OR-node based on maximization of information gain. (2) A top-down step retrieves the globally optimal parse tree in the AOG, according to the optimal states of encountered OR-nodes. See Sec. IV-A and IV-B for details.

(iii) Learning multiple tangram templates from non-aligned scene images by combining an exemplar-based clustering method and the DP algorithm stated above. The assumption above of having roughly aligned scene images usually fails to hold in practice due to the well-known large structural variations of a semantic-level scene category, which often consists of an unknown number of configuration-level sub-categories. E.g., a street scene category can have different configurations caused by distinct photographing angles. We address this issue with two steps: (1) Assigning the hidden sub-category labels for each training scene image based on an unsupervised exemplar-based clustering method, i.e. the *affinity propagation* algorithm [16]. (2) After that, we learn a tangram template for each cluster according to the DP algorithm mentioned in (ii). The details are given in Sec. IV-C.

(iv) Building a tangram bank representation for scene categorization by using the learned tangram templates as configuration "filters". Given a training dataset with a variety of scene categories (i.e., the semantic-level scene category labels are given), we first learn multiple tangram templates for each scene category using methods stated in (iii), and collect all the learned tangram templates to form a "tangram bank" of representative scene configurations, each of which works as a configuration "filter". Then, we present a new *tangram bank representation* for a scene image, which is composed of the tangram template scores (i.e., the "filter responses") on this image. Based on the proposed tangram bank image representation, we employ linear classifiers (i.e., SVM and Logistic regression) for scene categorization. The details are given in Sec. III-D.

(v) Building a tangram matching kernel for scene categorization. Besides the generative learning of tangram templates mentioned in (ii) an (iii), we propose a matching kernel [17], [6] based on tangram model, called *tangram matching kernel*, for discriminative kernel-based classification. It takes into account all the hidden spatial configurations in our tangram AOG, and thus leverages more flexible and richer configuration cues than the spatial pyramid to facilitate discrimination. See details in Sec. V.

In experiments, we build a new scene dataset (called $\mathrm{SceneConfig\_33}$), which consists of 33 different configuration classes distributed in 10 semantic categories, for facilitating evaluation of our learning method on scene configurations. We also test our method on two public scene datasets (i.e., $\mathrm{Scene\_15}$ [6] and $\mathrm{MIT\_Indoor}$ [7]). The experimental results on these three datasets show advantage of the proposed tangram model for scene representation and categorization:

(1) With much less dimensionality, our tangram bank representation shows significant performance gain w.r.t. traditional spatial pyramid "bag of visual words" (BOW) scene representation [6], for both of the configuration-level and semantic-level scene categorizations. Moreover, it even outperforms the spatial pyramid model with high-level appearance features such as the Object Bank (OB) representation [9]. (2) In corporation with a kernel SVM classifier, our tangram matching kernels can achieve superior scene classification performance than spatial pyramid matching [6] consistently.

C. Related work

The scene representation and analysis is one of the most fundamental topics in computer vision, making for many important applications such as scene recognition and parsing [3], [18], [6], [9], [19], object detection [13], [20], [21], image classification [6], [8], [22], image matching and registration [23], [24]. In literature, there are mainly two complementary views about the mechanisms (routes) utilized in recognizing the scene category: (1) *object-centered methods*, which first recognize the objects involved in the image and then infer the scene category based on the knowledge of the object contents; (2) *scene-centered methods*, which identify the scene category by directly using "scene-centered" visual cues such as global perceptual property and spatial layout, instead of recognizing its object contents first. The scene-centered methods either directly utilize the holistic low-level features such as global color and texture histogram [25], the spectrum information or induce the scene-level intermediate representation of perceptual dimensions such as naturalness, roughness, etc. [3] to facilitate scene recognition. More recently, the object-centered methods appear to become dominant. They take advantage of certain object-level intermediate representation for scene recognition (e.g., the occurrence frequency of object semantic concepts from local image patches [26] or the orderless image representation (e.g., "bag of visual words" model) with generative topic models such as probabilistic Latent Semantic Analysis (pLSA) [27] [28], Latent Dirichlet Analysis (LDA) [18]) for scene categories. Besides, to leverage the spatial distribution information of the localized appearance features for boosting recognition performance further, other high-level semantic information is also investigated in scene representation [9]. In addition, recent scene recognition systems [6], [29], [8] usually divide the image lattice into sub-windows or spatial pyramid to leverage the spatial distribution information of the localized appearance features for boosting recognition performance.

Contrary to the object-centered methods which treat objects as the atoms in scene recognition, there are psychological and behavioral research work [30] shown that recognizing the semantic category of most real world scenes at a glance does not need to identify the objects in a scene at first but can be directly perceived from the *scene configuration*, which involves the spatial layout of contours [3], [31], the arrangement of basic geometrical forms such as simple *Geons* clusters [32], and the spatial organization of atomic regions or color blobs with particular size and aspect ratio [33] [34], etc. This motivates us to exploit an explicit model for representing the scene configurations. Our tangram model is related to the hybrid image template (HiT) [15], which learns explicit templates for object recognition, but differs from it in two aspects: (1) The primitives. Instead of using the sketch features for representing object shape [35], we propose an overcomplete dictionary of shape primitives to build the tangram like scene layout representation. (2) The learning algorithm. In [15], the HiT is learned by a greedy shared matching pursuit algorithm [36], while our tangram model adopts DP algorithm to achieve the globally optimal configuration. Besides, a very recent work [37] presented a reconfigurable "bag of words" (RBoW) model for scene recognition, which leverages semantically meaningful configuration cues via a latent part-based model.

Very recently, there are some work [38], [39], [40] showing enormous success on the scene categorization task (especially on the MIT_Indoor dataset), by using a collection of automatically discovered local HOG templates or part-based models to better leverage appearance cues. However, in our paper we focus on modeling reconfigurable structure of scene category, which learn a series of global templates at the configuration level. Although we only use standard SIFT BOW as appearance feature, the observations from these work and our paper are complementary that the classification performance can be improved by cooperating better appearance model with a predefined spatial pyramid or learning better configuration with simple appearance feature. To further boost the performance, the two aspects could be integrated in future work.

Our preliminary work has been published in [41], and extended in this paper as follows: (i) We propose a method of learning multiple tangram models from non-aligned scene images, by combining the affinity propagation clustering algorithm [16] and the DP algorithm. By collecting all the learned templates from different categories, we build a tangram bank representation of scene images to improve the classification performance significantly. (ii) We present a new formulation (i.e. *SOFT_MAX_OR*) on the tangram matching kernel, which includes the *MAX_OR* and *MEAN_OR* ones in [41] as its two extreme cases. The classification performance is also enhanced accordingly. (iii) We provide more detailed experimental evaluations and analysis on the proposed methods.

D. Paper organization

The rest of this paper is organized as follows: In Sec. II, we elaborate a compositional tan dictionary as well as associated AOG, and the reconfigurable parse tree for quantizing the configuration space of scene layout. In Sec. III, we present a generative formulation of learning tangram template, and build a tangram bank representation for scene images. In Sec. IV, we introduce a DP algorithm to learn the globally optimal parse tree from roughly aligned scene images, and then propose a clustering-based method for discovering multiple tangram templates from non-aligned scene images. After that, a tangram matching kernel is presented for discriminative learning and classification in Sec. V. Finally, we evaluate our tangram model by a series of experiments in Sec. VI, and then conclude this paper in Sec. VII.

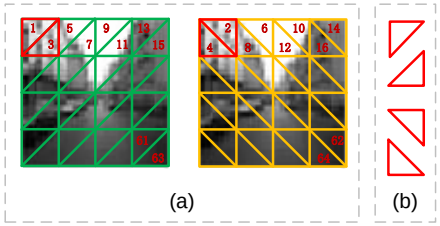


Fig. 2. Illustration on tiling the image lattice by shape primitives. (a) triangular tiling of image lattice for a 4×4 grid; (b) four types of primitives (i.e., triangular tiles) used in this paper. (Best viewed in color)

II. The Reconfigurable Tangram Model

A. The tan dictionary

1) Tiling the image lattice by shape primitives: Let Λ denote the image lattice, and we partition Λ into a grid of $n_c = n_w \times n_h$ cells. For each cell of the grid, it is further decomposed into two triangular tiles in two alternative ways (in diagonal or back-diagonal direction). Fig. 2 illustrates the tiling of image lattice for a 4×4 grid as well as these four types of triangular primitives. To achieve the compactness and reconfigurablity discussed in Sec. I-A, it asks for an over-complete dictionary of shape primitives with a variety of shape types, scales and locations on Λ. In this paper, a **tan** is defined as a connected polygon composed of several non-overlapping triangular tiles, and its size is defined by the number of its triangular constituents (i.e., how many triangular tiles it is composed of, and the maximum value the size can take is $2n_c$). Compared to the rectangular primitives, the elementary primitives of triangular tiles are capable of composing the tans with more shape types (e.g., trapezoid, parallelogram) and thus lead to more flexible quantization on scene configuration.

2) A layered tan dictionary: The tan dictionary is a layered collection of tans with various sizes. The layer index, denoted by l, of a tan is defined by its size. In this paper, the term of "layer" is used only to imply the relative size of a tan w.r.t. that of the smallest triangular primitives, not the actual layer (or depth) of a tan in the AOG built later on. Given the image lattice Λ with n_c cells, a **tan dictionary**, denoted by Δ, is defined as the union of L (e.g., $L = 2n_c$ in the case of using triangular primitives) subsets: $\Delta = \bigcup_{l=1}^{L} \Delta^{(l)}$, where $\Delta^{(l)}$ denotes the subset of tans at the l^{th} layer. For $\Delta^{(l)}$, it consists of N_l tans. That is $\Delta^{(l)} = \{B_{(l,i)} \mid i = 1, 2, \cdots, N_l\}$. Besides, one tan can produce a series of different instantiations (called **tan instances**) through placing it onto different valid positions in the cell grid of Λ. For each tan $B_{(l,i)}$, we denote its instances by $\{B_{(l,i,j)} \mid j = 1, 2, \cdots, J_{(l,i)}\}$, where each tan instance $B_{(l,i,j)}$ is associated with domain $\Lambda_{(l,i,j)} \subseteq \Lambda$.

For example, Fig. 3 illustrate a 32-layer tan dictionary. We can see that there are four types of triangular primitives as the tans in the 1st layer, and the most top (i.e., 32nd) layer has only one tan (also the instance) such that $\Lambda_{(32,1,1)} = \Lambda$. In addition, it is shown on the top-right corner of Fig. 3 that the tan $B_{(8,18)}$ has 6 instances with different translated positions on the cell grid of image lattice. The tans define conceptual shape of polygonal ones, and the instances, linking to the image data, are their instantiations when placed on Λ.

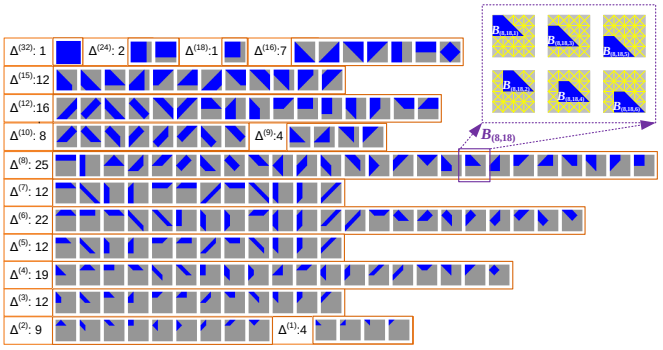


Fig. 3. Illustration on a 32-layer tan dictionary for the 4×4 tiling grid. It consists of 166 tans in total, with 889 instances placed on different locations on the grid of image lattice Λ. We show only one instance for each tan for clarity. In the upper-right corner, it illustrates the tan $B_{(8,18)}$ has 6 instances with different translated positions on the cell grid of image lattice. (Best viewed in color and magnification)

B. Organizing the tan dictionary into AOG

Although the taxonomy of tan dictionary has been elaborated so far, there are still two problems to be addressed: (1) The tans with large size tend to become exponentially innumerable if any number of k-way composition is allowed for decomposing a sub-lattice, which may prohibit a dictionary with potentially great number of layers for covering shape variations on larger image domain. (2) The tans in this layered dictionary are defined independently with each other, without consideration of the compositionality among them. Motivated by the image grammar model [11], we propose a method of recursive shape composition to construct the tan dictionary, which is organized into an associated AOG.

Similar to the relationship between a tan and its tan instances as discussed in Sec. II-A, there are two isomorphic AOGs built (denoted by Υ_Δ and $\Upsilon_\Delta^{'}$) in this paper, which correspond to the tans and their instances in Δ respectively. The AOG Υ_Δ retains all the compositional relationship of canonical shapes as shown in Fig. 3, while the other AOG $\Upsilon_\Delta^{'}$ makes copies of these shapes at all valid translations like the upper-right inset of Fig. 3.

1) The And-Or graph of tans: The **AOG** Υ_Δ is defined as a hierarchical directed acyclic graph to describe the compositional relationship among the tans in Δ. Meanwhile, the **AND-node** represents the composition from a set of tans to a larger one (e.g. composing two triangular tiles to a square tan shown in Fig. 4), and the **OR-node** indicates the alternative ways on shape composition (e.g. the two different ways of composing two triangular tiles to a square tan in Fig. 4).

As illustrated in Fig. 4 (a), one tan can be alternatively generated by different way of composing two child ones at the lower layers. Consequently, it leads to an **And-Or unit** for each tan $B_{(l,i)}$: $\{v^T_{(l,i)}, v^{OR}_{(l,i)}, \{v^{AND}_{(l,i),o}\}_{o=1}^{O_{(l,i)}}\}$, where $v^T_{(l,i)}$, $v^{OR}_{(l,i)}$ and $v^{AND}_{(l,i),o}$ denote terminal-node, OR-node and AND-node, respectively. The terminal-node $v^T_{(l,i)}$ is namely $B_{(l,i)}$. The AND-node $v^{AND}_{(l,i),o}$ represents that $B_{(l,i)}$ can be composed by two child tans at layers below. The OR-node $v^{OR}_{(l,i)}$ represents that $B_{(l,i)}$ can either directly terminate into $v^T_{(l,i)}$ or further be decomposed into two child tans, in one of $O_{(l,i)}$ different ways. Thus, the AOG Υ_Δ is constituted by And-Or units, to

organize the tans generated in Δ, as illustrated in Fig. 4 (a).

2) Constructing the tan dictionary by recursive shape composition: In this paper, we employ two successive steps to construct the tan dictionary Δ as well as associated AOG:

(i) generating the tans through recursive composition in a bottom-up manner, from which an AOG Υ_Δ is simultaneously built to retain their compositional relationships;
(ii) generating tan instances with another AOG $\Upsilon_\Delta^{'}$ by tracing Υ_Δ in a top-down manner.

For constructing Δ, the quantity of tans at each layer should be controlled as an intermediate number, to achieve trade off between the representative ability on shape variation and the computational tractability. Additionally, because the top-layer tan in Δ amounts to Λ exactly, the size of Λ is considered as an upper-bound constraint of the tans generated (also the number of layers for Δ) so that their instances could be within Λ. Thus, starting from the 1st layer (i.e. $\Delta^{(1)}$ shown in Fig. 3), a valid tan is generated by composing the ones at layers below with all of the following three rules satisfied:

(i) we relax the valid tans to be one of three shape types: triangle, trapezoid and parallelogram. It accounts for non-rectangular shape of regions appeared in complex scene configurations, while avoiding combinatorial explosion at higher layers.
(ii) The size of each tan in the AOG should not be larger than that of Λ (i.e. $2n_c$).
(iii) By allowing deep hierarchical structure in building Υ_Δ, we only apply the binary production rule to keep the graph structure tractable.

Actually, the top-layer tan $B_{(L,1)}$ defines the root node for Υ_Δ. This suggests a post-processing operation to prune the tans which are not involved in the path of composing it to the end. Moreover, there could be no valid tans available at some layers in Δ, due to that it cannot find any two tans at layers below to compose a valid one according to the compositional rules. E.g., for a 32-layer tan dictionary, there is no tan available obtained at the layers of $l \in \{\{11, 13, 14, 17\} \cup [19, 23] \cup [25, 31]\}$, which are ignored and thus not shown in Fig. 3.

When Υ_Δ is built, a top-down step is triggered to generate the tan instances. At first, we place the top-layer tan $B_{(L,1)}$ on Λ, from which only one instance $B_{(L,1,1)}$[1] is created in the top layer of Δ. Then, an isomorphic AOG $\Upsilon_\Delta^{'}$, whose root node is imitated from that of Υ_Δ, is built to organize all the tan instances in Δ. By iterations, the tan instances at lower layers are top-down generated through the following procedures:

(i) Given a tan instance $B_{(l,i,j)}$, we retrieve the child tans (denoted by $B_{(l_1,i_1)}$ and $B_{(l_2,i_2)}$) of $B_{(l,i)}$ for each AND-node $v^{AND}_{(l,i),o}$ ($o \in \{1, 2, \cdots, O_{(l,i)}\}$) in Υ_Δ;
(ii) Then, we generate the tan instances $B_{(l_1,i_1,j_1)}$ and $B_{(l_2,i_2,j_2)}$, by placing $B_{(l_1,i_1)}$ and $B_{(l_2,i_2)}$ onto $\Lambda_{(l,i,j)}$ such that $\Lambda_{(l,i,j)} = \Lambda_{(l_1,i_1,j_1)} \cup \Lambda_{(l_2,i_2,j_2)}$.
(iii) A new And-Or unit of $B_{(l,i,j)}$ is built for the AOG $\Upsilon_\Delta^{'}$, by replicating the counterpart of $B_{(l,i)}$ in Υ_Δ.

[1] $B_{(L,1,1)}$ has the same size as $B_{(L,1)}$ such that $\Lambda_{(L,1,1)} = \Lambda$.

This process recursively runs over the tan instances, starting from the L^{th} layer to the 1st one in $\Upsilon_\Delta^{'}$. As illustrated in Fig. 3, one tan in Υ_Δ can produce multiple instances in the same layer of $\Upsilon_\Delta^{'}$, at the locations of different grid coordinates on Λ. Besides, due to the correspondence between a tan and their instances, there is also an And-Or unit associated with each tan instance in $\Upsilon_\Delta^{'}$, which inherits all the And-Or compositionalities from corresponding tan from Υ_Δ. Fig. 4 (b) illustrates a small portion of $\Upsilon_\Delta^{'}$. We can see that the OR-nodes A and B are generated by copying the common one, which is shown in the top of Fig. 4 (a), from Υ_Δ but with different positions in the image lattice. Given a particular image lattice (e.g., 2×2 or 4×4 grid), the tan dictionary and associated AOG are automatically built, without any manual manipulation, based on the rules mentioned above.

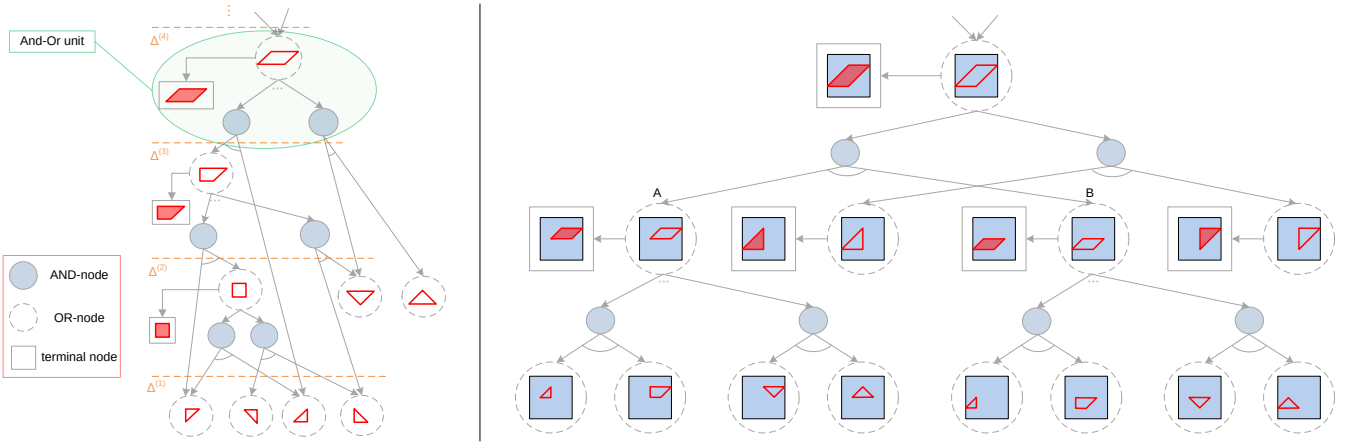
C. The reconfigurable parse tree for quantizing spatial configuration

In this paper, the tangram model is defined via a reconfigurable **parse tree** in $\Upsilon_\Delta^{'}$, to quantize spatial configuration of scene layout. The parse tree, denoted by *Pt*, is a binary tree composed of a set of non-terminal **pass-by nodes** V_N^{Pt} and a set of terminal **leaf nodes** V_T^{Pt}. It can be regarded as a derivative of the AOG $\Upsilon_\Delta^{'}$, through selecting a unique child node for each OR-node. In fact, we can generate a parse tree via a recursive parsing process from the root node of AOG.

For convenience, we first introduce a state variable (denoted by $\omega_{(l,i,j)} \in \{0, 1, 2, \cdots, O_{(l,i,j)}\}$) to indicate the selection of child node for the OR-node $v^{Or}_{(l,i,j)}$ of $\Upsilon_\Delta^{'}$. To be consistent with the notations in Sec. II-B1, $O_{(l,i,j)}$ denotes the number of different ways of composing $B_{(l,i,j)}$ from child tan instances. $\omega_{(l,i,j)}$ taking the value of $1 \le o \le O_{(l,i,j)}$ represents that $B_{(l,i,j)}$ is decomposed into two child tan instances according to the AND-node $v^{AND}_{(l,i,j),o}$, while $\omega_{(l,i,j)} = 0$ implies the selection of its terminal node $v^{T}_{(l,i,j)}$ and the decomposition process will stop. Then we define a recursive operation,

TABLE I
MAIN NOTATIONS USED IN THE TANGRAM MODEL

Notation	Meaning
Λ	image lattice
Δ	tan dictionary
$B_{(l,i)}$	the i^{th} tan in the l^{th} layer of Δ
$B_{(l,i,j)}$	the j^{th} instance for $B_{(l,i)}$
$\Lambda_{(l,i,j)}$	the image domain associated with $B_{(l,i,j)}$
Υ_Δ	the AOG of tans
$v^{T}_{(l,i)}$	the terminal-node for $B_{(l,i)}$
$v^{OR}_{(l,i)}$	the OR-node for $B_{(l,i)}$
$v^{AND}_{(l,i),o}$	the o^{th} AND-node for $B_{(l,i)}$
$\Upsilon_\Delta^{'}$	the AOG of tan instances
$v^{T}_{(l,i,j)}$	the terminal-node for $B_{(l,i,j)}$
$v^{OR}_{(l,i,j)}$	the OR-node for $B_{(l,i,j)}$
$v^{AND}_{(l,i,j),o}$	the o^{th} AND-node for $B_{(l,i,j)}$
$\omega_{(l,i,j)}$	the state variable of $v^{OR}_{(l,i,j)}$
Pt	parse tree
V_N^{Pt}	the set of non-terminal pass-by nodes in *Pt*
V_T^{Pt}	the set of terminal leaf nodes in *Pt*



(a) (b)

Fig. 4. Illustration on organizing tan dictionary into And-Or graph. (a) the AOG of tans (i.e., Υ_{Δ}); (b) the AOG of tan instances (i.e., $\Upsilon_{\Delta}^{'}$). Υ_{Δ} is built in a bottom-up manner to retain all the compositional relationship among the tans in Δ. After that, $\Upsilon_{\Delta}^{'}$ can be generated for tan instances by tracing Υ_{Δ} in a top-down manner. We only show a small portion of structure on the AOGs for clarity. (Best viewed in color and magnification)

denoted by $\mathbf{PARSE}(B_{(l,i,j)}; \Upsilon_{\Delta}^{'})$, to parse $B_{(l,i,j)}$ given the value of $\omega_{(l,i,j)}$:

(i) Starting from the OR-node $v_{(l,i,j)}^{Or}$, select one of the child nodes (i.e., $v_{(l,i,j),o}^{AND}$ or $v_{(l,i,j)}^{T}$) according to $\omega_{(l,i,j)}$;

(ii) If an AND-node $v_{(l,i,j),o}^{AND}$ (i.e., $o = \omega_{(l,i,j)} \neq 0$) is selected, join $B_{(l,i,j)}$ into V_N^{Pt} and call $\mathbf{PARSE}()$ to each of its child tans;

(iii) If reaching the terminal node $v_{(l,i,j)}^{T}$ (i.e., $\omega_{(l,i,j)} = 0$), join $B_{(l,i,j)}$ into V_T^{Pt} and stop traveling further in $\Upsilon_{\Delta}^{'}$.

By applying $\mathbf{PARSE}()$ from the top-layer tan instance in Δ, a parse tree *Pt* can be generated from $\Upsilon_{\Delta}^{'}$ according to the state variable values at its encountered OR-nodes. In *Pt*, the pass-by nodes specify intermediate splitting process in the hierarchy, while the leaf nodes partition the image lattice to form a spatial configuration. Fig. 1 (b) illustrate two examples of parse trees for different scene configurations. In table I, we summarize main notations used in our tangram model.

Rather than the fixed layout used in the spatial pyramid, the parse tree of tangram model is "reconfigurable", in the sense that it can provide a compact representation adaptive to diverse spatial configurations of scene layout. Based on its associated AOG, the tan dictionary actually defines a "quantization space" on continuously variable spatial configuration for representing scene layouts. Through inducing the OR-nodes and reusing the tans in shape composition, the tangram AOG can represent an exponentially increasing number of spatial configurations w.r.t. the cardinality of tan dictionary.

III. The Tangram Templates for Scene Representation

A. The tangram template

On basis of the tan dictionary and reconfigurable parse tree of AOG introduced in Sec. II, we present the tangram template for explicitly modeling a scene layout. Given a parse tree *Pt*, we define a **tangram template**, denoted by *Tgm*, as a set of non-overlapping tan instances specified by the leaf nodes of *Pt*:

$$Tgm = \{(B_k, \Lambda_k, \rho_k) \mid k = 1, 2, \cdots, K\}, \quad (1)$$
$$\Lambda_{Tgm} = \cup_{k=1}^{K} \Lambda_k \subseteq \Lambda \text{ and } \Lambda_i \cap \Lambda_j = \emptyset \ (\forall i \neq j),$$

where each selected tan instance B_k, associated with domain Λ_k and an appearance model ρ_k, corresponds to a leaf node of *Pt*. Here the subscript k is a linear index of tan instance to replace the triple-tuple index (l, i, j) used in Sec. II for notation simplicity. K denotes the total number of tan instances in *Tgm*. As shown in Fig. 1 (b), the tangram template explicitly represents scene configuration as well as the appearance for each tan through the collapse of a parse tree.

B. Appearance model for a tan instance

For a tan instance B_k, we represent its appearance pattern by a parametric prototype model h_k. Let I_{Λ_k} and $H(I_{\Lambda_k})$ denote the image patch on Λ_k and a feature mapping function on I_{Λ_k}, respectively. Generally, it can be any type of "off-the-shelf" visual feature as the appearance descriptor for B_k, e.g. HOG [42], Gist [3] or SIFT BOW features [43], [18], [6]. Furthermore, we define the appearance model's feature response r_k for B_k. It maps original appearance descriptor feature $H(I_{\Lambda_k})$ to a bounded scalar value, which would obtain a large value when $H(I_{\Lambda_k})$ is "close" to h_k. For vector-wise appearance features such as SIFT BOW used in this paper, we can compute responses by employing any valid similarity measurement between H and corresponding prototype model h_k. In this case, h_k is a vector with the same dimension as H. In this paper, we adopt the histogram intersection kernel (HIK) [44], [45], which is an effective but simple measurement for histogram features. That is

$$r_k = \sum_{b=1}^{\mathcal{B}} \min[H^{(b)}(I_{\Lambda_k}), h_k^{(b)}], \quad (2)$$

where $H^{(b)}$ and $\mathcal{B}$ refer to the value of the b^{th} bin and the dimension of H respectively.

C. A generative log-linear model of tangram template

Based on the information projection theory [15], we present a generative model on the tangram template in this subsection. Let $f(I)$ and $q(I)$ denote the underlying probability distribution of a target scene layout category and the background model of natural image statistics, respectively. For tangram template *Tgm*, we define a generative probabilistic model of scene image, denoted by $p(I; Tgm)$. Then, a model space $\Omega_p(Tgm)$ can be given by

$$\Omega_p(Tgm) = \{\, p(I; Tgm) \,|\, E_p[r_k] = E_f[r_k], \forall k \,\}, \quad (3)$$

where $E_p[r_k] = E_f[r_k]$ accounts for that the expectation of feature response on each selected tan instance subjects to match the empirical statistics. According to the maximum entropy principle [46], $p(I; Tgm)$ is suggested to be the one closest to $q(I)$ by means of KL-divergence (denoted by $KL(\cdot\,||\,\cdot)$) [15]:

$$\begin{aligned} \hat{p} &= \arg\min_{p\in\Omega_p(Tgm)} KL(p\,||\,q) \quad (4)\\ &= \arg\min_{p\in\Omega_p(Tgm)} E_p[\log \tfrac{p(I;Tgm)}{q(I)}]. \end{aligned}$$

Besides, considering the non-overlapping tan instances in *Tgm* [15], a factorized log-linear model is obtained as follow

$$\hat{p}(I; Tgm) = q(I) \prod_{k=1}^{K} [\frac{1}{z_k} \exp(\lambda_k r_k)], \quad (5)$$

where λ_k and z_k refer to the parameters of weight and normalizing factor for B_k respectively. Meanwhile, thanks to the factorization assumption, z_k can be computed by using a one-dimensional marginal distribution $q(r_k)$ as shown below:

$$z_k = E_q[\exp(\lambda_k r_k)] = \int_{r_k} \exp(\lambda_k r_k) q(r_k). \quad (6)$$

D. Building a tangram bank representation on scene image

In this subsection, we present a new representation of scene image, called **tangram bank** (TBank) representation, based on a collection of tangram templates each of which works as a "filter" of scene configuration. Let $\mathbb{D}$ denote a set of tangram templates $\{Tgm^{(t)} \mid t = 1, 2, \cdots, \mathcal{T}\}$. For each tangram template (i.e., the configuration "filter") in $\mathbb{D}$, we compute its score of image I as a configuration "filter response":

$$\begin{aligned} \phi_t(I;\, Tgm^{(t)}) &= \log \tfrac{p(I;Tgm^{(t)})}{q(I)} \quad (7)\\ &= \sum_{B_k \in V_T^{Pt}} (\lambda_k^{(t)} r_k^{(t)} - \log z_k^{(t)}),\ \forall\, t = 1, 2, \cdots, \mathcal{T}, \end{aligned}$$

where $\lambda_k^{(t)}$, $z_k^{(t)}$ and $r_k^{(t)}$ are respectively the model parameters and appearance feature response for the k^{th} tan instance ($K^{(t)} = |V_T^{Pt}|$ in total) selected in $Tgm^{(t)}$. Thus, based on the scores of tangram templates in $\mathbb{D}$, we build a $\mathcal{T}$-dimensional TBank representation on I: $\Phi(I; \mathbb{D}) = [\phi_1(I;\, Tgm^{(t)}), \phi_2(I;\, Tgm^{(t)}), \cdots, \phi_{\mathcal{T}}(I;\, Tgm^{(t)})]^{\mathrm{T}}$.

As illustrated in Fig. 5, the "tangram bank" $\mathbb{D}$ actually defines a new feature space by using a series of tangram templates as representative scene configurations. In this feature space, each dimension of resultant TBank representation corresponds to the similarity between image I and a tangram template in $\mathbb{D}$. Thus, any scene image can be projected into such feature representation according to Equ. (7), which is more semantically meaningful and compact than original low-level BOW representation. On basis of this TBank representation, we simply adopt a linear classifier (e.g., SVM or logistic regression) for scene categorization in our experiments.

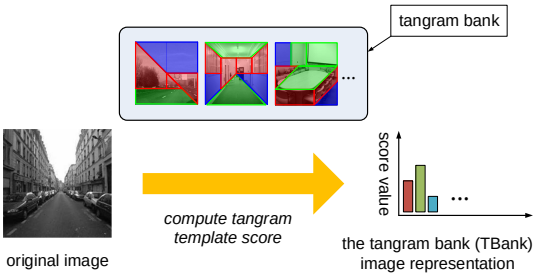


Fig. 5. Illustration on building the tangram bank representation of a scene image (Best viewed in color).

Besides, we find that the tangram template defined in Equ. (1) is no more than a flat structure, which only includes the tan instances of terminal leaf nodes in a parse tree. According to the observation that it is preferable to use a multi-layered spatial representation [6], we can alternatively build a multi-layered tangram template through including the tan instances of non-terminal pass-by nodes besides the leaf ones. Accordingly, the scoring function $\phi_t(I;\, Tgm^{(t)})$ in Equ. (7) is redefined as follow[2]:

$$\phi_t(I;\, Tgm^{(t)}) = \sum_{B_k \in V_T^{Pt} \bigcup V_N^{Pt}} (\lambda_k^{(t)} r_k^{(t)} - \log z_k^{(t)}). \quad (8)$$

IV. Learning the Tangram Templates

A. Learning by maximizing information gain

In this subsection, similar to [15], we use roughly aligned training images to learn a tangram template, as explicit modeling of scene layout. Let $D^+ = \{I_1^+, I_2^+, \cdots, I_N^+\}$ denote a set of N positive images, which are assumed sampled from the target distribution $f(I)$, for the scene layout category to be learned. Besides, we characterize the background model $q(I)$ by an image set $D^- = \{I_1^-, I_2^-, \cdots, I_M^-\}$, which consists of all the training images collected from various scene categories in practice. Our objective is learning a model $p(I; Tgm)$ of tangram template *Tgm* from D^+, to approach $f(I)$ starting from $q(I)$. To simplify notation as in Sec. III-C, let $H_k = H(I_{\Lambda_k}; \psi_k)$ for B_k. We denote its appearance descriptors on D^+ and D^- by $\{H_{k,n}^+\}_{n=1}^N$ and $\{H_{k,m}^-\}_{m=1}^M$, respectively. Likewise, corresponding feature responses are respectively abbreviated by $\{r_{k,n}^+\}_{n=1}^N$ and $\{r_{k,m}^-\}_{m=1}^M$.

Similar to [36], [15], we define a regularized **information gain** as the learning objective of the tangram template *Tgm*:

$$\begin{aligned} \mathcal{IG}(Tgm) &= KL(f\,||\,q) - KL(f\,||\,\hat{p}) - \mathcal{M}(Tgm) \quad (9)\\ &= \textstyle\sum_{k=1}^{K} \{\lambda_k E_f[r_k] - \log z_k - \frac{1}{2}\beta {\lambda_k}^2\} - \alpha K, \end{aligned}$$

where $[KL(f\,||\,q) - KL(f\,||\,\hat{p})]$ is an information-theoretical measurement on the improvement of the learned model $\hat{p}(I; Tgm)$ approaching $f(I)$ relative to $q(I)$. $\mathcal{M}(Tgm) = \sum_{k=1}^{K} \frac{1}{2}\beta {\lambda_k}^2 + \alpha K$ refers to the regularization term on model

[2] In Equ. (1) and (5), $K = |V_T^{Pt}| + |V_N^{Pt}|$ for multi-layered tangram template.

complexity, in which β and α denote the trade-off parameters on shrinking the weight λ_k and punishing large number of tan instances selected in *Pt*, respectively. Thus, learning the optimal tangram template, denoted by *Tgm**, (as well as corresponding model parameters λ_k^* and z_k^*) from D^+ is achieved by maximizing its information gain $\mathcal{IG}(Tgm)$. Intuitively, it implies how many bits can be saved for coding the positive images of D^+ by using the learned model of tangram template instead of the natural image statistics.

As in [36], [15], the positive images, which share similar target scene configuration to be learned, in D^+ are assumed roughly aligned up to a certain of variations. Thus, we estimate the prototype parameter h_k of the appearance model for each candidate tan instance B_k by simply averaging the feature descriptors H_k over all the positive images from D^+: $h_k^* = \frac{1}{N}\sum_{n=1}^{N} H_{k,n}^+$. Then, we estimate λ_k and z_k for B_k. Through solving $\frac{\partial \mathcal{IG}}{\partial \lambda_k} = 0$, the optimum values are given by

$$(\lambda_k^*, z_k^*):\ E_f[r_k] - E_{\hat{p}}[r_k] = \beta\lambda_k. \tag{10}$$

In practice, we calculate the empirical expectation $E_f[r_k]$ by the mean response value on positive images. That is $E_f[r_k] \approx \frac{1}{N}\sum_{n=1}^{N} r_{k,n}^+$. The term of $E_{\hat{p}}[r_k]$ is approximately calculated by using the feature responses on D^-:

$$\begin{aligned} E_{\hat{p}}[r_k] &= E_q[\tfrac{1}{z_k}\exp(\lambda_k r_k) r_k] \\ &\approx \tfrac{1}{M}\textstyle\sum_{m=1}^{M}[\tfrac{1}{z_k}\exp(\lambda_k r_{k,m}^-) r_{k,m}^-]. \end{aligned} \tag{11}$$

Likewise, the normalization factor z_k can be estimated by approximating Equ. (6) with all M background examples:

$$z_k \approx \frac{1}{M}\sum_{m=1}^{M}\exp(\lambda_k r_{k,m}^-). \tag{12}$$

Noting that we only need to approximate the one-dimensional marginal distribution in Equ. (6), it is feasible to use a number of samples in D^- for parameter estimation, which actually correspond to all the image examples collected from different scene categories in our experiments. By replacing Equ. (12) into (11), we can derive a monotonic function of λ_k for estimating $E_{\hat{p}}[r_k]$ but the Equ. (10) cannot be solved analytically. On the implementation of Equ. (10), it can be solved by Newton method or the line search [36], [15].

Thus, we obtain the information gain for B_k as follow

$$g_k = \max(\lambda_k^* E_f[r_k] - \log z_k^* - \frac{1}{2}\beta{\lambda_k^*}^2 - \alpha, 0), \tag{13}$$

where $\max(\cdot, 0)$ implies that the tan instances giving negative information gain values would be not involved in *Tgm**. After that, a DP algorithm, which will be introduced in Sec. IV-B, is called to find *Tgm** over the solution space of parse trees.

B. The DP algorithm on learning a tangram template

The recursive And-Or structure with deep hierarchy is able to represent a huge space of spatial configurations on scene layout, each of which is specified by a parse tree instantiated from the AOG. Although an exponential number of parse trees (as well as tangram templates) need to be considered in the solution space, the direct acyclic characteristic of AOG makes the globally optimal solution can be efficiently searched through a DP algorithm.

Algorithm 1: The DP Algorithm for Searching Globally Optimal Parse Tree of Tangram Template

Input: AOG $\Upsilon_\Delta^{'}$, information gain on terminal-nodes: $\{g_{v_{(l,i,j)}^T} \mid \forall l, i, j\}$
Output: the optimal parse tree *Pt**

Step I: bottom-up propagating information gain:
foreach $l = 1$ *to* L **do**
  foreach $i = 1$ *to* N_l *and* $j = 1$ *to* $J_{(l,i)}$ **do**
    foreach *AND-node* $o = 1$ *to* $O_{(l,i)}$ **do**
      Let $g_{v_{(l,i,j),o}^{AND}} = \sum_{u \in Ch(v_{(l,i,j),o}^{AND})} g_u$;
    end
    Let $g_{v_{(l,i,j)}^{OR}} = \max_{u \in Ch(v_{(l,i,j)}^{OR})} g_u$,
    and $\omega_{(l,i,j)}^* \rightarrow \Upsilon_\Delta^{'}$;
  end
end
Step II: top-down parsing from the root node of $\Upsilon_\Delta^{'}$*:*
$\mathbf{PARSE}(B_{(L,1,1)}; \Upsilon_\Delta^{'})$.

For a node v in $\Upsilon_\Delta^{'}$, let g_v and $Ch(v)$ denote its information gain and the set of child nodes, respectively. Before starting the DP algorithm, we assume the gain of each terminal-node is computed by Equ. (13). Then, in this DP algorithm, it propagates their gains to AND-nodes (by the sum operation: $g_{v^{AND}} = \sum_{u \in Ch(v^{AND})} g_u$) and OR-nodes (by the max operation: $g_{v^{OR}} = \max_{u \in Ch(v^{OR})} g_u$, with recording the optimal state ω^* of v^{OR} at the same time) through a bottom-up step. After that, the globally optimal parse tree *Pt**, which is defined as the one with maximum information gain value at the root node, can be top-down retrieved according to the optimal states of encountered OR-nodes by calling the parsing operation $\mathbf{PARSE}()$ on the top-layer tan instance. We summarize the proposed DP algorithm in Alg. 1.

C. learning multiple tangram templates for scene configuration discovery

So far, we have focuses on the problem of learning a single tangram template of scene configuration from a set of roughly aligned images. However, the assumption above of having roughly aligned scene images usually fails to hold in practice due to the well-known large structural variations of a semantic-level scene category, i.e., which often consists of an unknown number of configuration-level sub-categories. For instance, the images belonging to the street scene can be photographed from various views. It motivates us to learn multiple tangram templates for a scene category, each of which corresponds to a representative scene configuration explaining out a potential cluster of training images.

Among all the training images, we assume there exist a small portion of representative ones, called the exemplars, corresponding to underlying typical scene configurations. Moreover, the exemplars potentially define the "centers" of non-overlapping clusters, each of which involves a subset of

training images. We consider the similarity between a pair of images, and define an $N \times N$ affinity matrix $\mathbf{S}$, where the element $\mathbf{S}(i,j)$ denotes the affinity of the i^{th} training image w.r.t. the cluster with the j^{th} one as its exemplar. Based on the generative formulation in Sec. III-C, a tangram template $Tgm_{(i,j)}$ can be learned for the i^{th} training image, by using the j^{th} one as its reference image for appearance prototypes. Specifically, we first use the i^{th} training image as the unique positive sample, and set the prototype parameter of each candidate tan instance by corresponding appearance descriptor of the j^{th} image. Then, for each pair of training images (i,j), we learn the optimal tangram template $Tgm^{*}_{(i,j)}$ according to the DP algorithm presented in Sec. IV-B, and the information gain $\mathcal{IG}(Tgm^{*}_{(i,j)})$ is used as the value of $\mathbf{S}(i,j)$. Intuitively, $\mathcal{IG}(Tgm^{*}_{(i,j)})$ measures the similarity of the i^{th} training image w.r.t. the j^{th} one as an exemplar of tangram template. Thus, we can construct $\mathbf{S}$ by learning the tangram template as mentioned above for every pair of training images.

Given the affinity matrix $\mathbf{S}$ of all training images, an exemplar-based affinity propagation clustering algorithm [16] is applied to discover the exemplars as well as the clusters. After that, we can learn one tangram template for each of the clusters through the following two steps:

(i) Let the training images belonging to this cluster compose a set of positive samples D^{+} as defined in Sec. IV-A;
(ii) Learn the optimal tangram template according to the method described in Sec. IV-A and IV-B.

V. The tangram matching kernel

Besides the generative formulation of learning tangram model in Sec. IV, we present a **tangram matching kernel** (TMK) for discriminative learning in this section, by taking into account all the hidden spatial configurations in our tangram AOG. Given a pair of images, we first compute the matching score s_{v^T} for each terminal-node in $\Upsilon^{'}_{\Delta}$ as the intersection value between the histogram features (i.e. the matched features on this tan instance) according to the histogram intersection function as in Equ. (2). Then, the matched features are bottom-up accumulated from the 1st layer to the top one: the matching score of an AND-node v^{AND} is computed by accumulating the ones of its child tan instances, plus the weighted increment of the intersection value which corresponds to the matched features newly found for relaxing the spatial constraint imposed by the AND-node. That is

$$s_{v^{AND}} = \sum_{u^{OR}} s_{u^{OR}} + \frac{1}{l}\left(s_{v^T} - \sum_{u^T} s_{u^T}\right), \tag{14}$$

where $u^{OR} \in Ch(v^{AND})$ and $u^{T} \in Ch^{'}(v^{AND})$ respectively denote the OR-node and the terminal one of a child tan instance for v^{AND}. Similar to the spatial pyramid matching (SPM) [6], we simply set the weight of matched features newly found by $\frac{1}{l}$, which is inverse to the layer index of v^{AND}, implying that the features matched in larger tan instances are more penalized due to the relaxation of spatial constraints. For an OR-node v^{OR}, the matching score $s_{v^{OR}}$ is obtained as follows: if corresponding tan instance lies in the first layer, we directly set it by that of terminal-node (i.e., $s_{v^{OR}} = s_{v^T}$), otherwise we use a *SOFT_MAX_OR* operation to calculate $s_{v^{OR}}$ by:

$$s_{v^{OR}} = \sum_{u^{AND} \in Ch(v^{OR})} (\pi_{u^{AND}} \cdot s_{u^{AND}}), \tag{15}$$

where $u^{AND} \in Ch(v^{OR})$ denotes one of the child AND-nodes for v^{OR} and $\pi_{u^{AND}} = \frac{\exp(\gamma s_{u^{AND}})}{\sum_{u' \in Ch(v^{OR})} \exp(\gamma s_{u'})}$ is the weight of soft-max function to fuse the matching values obtained by different child AND-nodes for the OR-node v^{OR}. Meanwhile, γ is a predefined tuning parameter to adjust the degree of "soft" maximization over candidate child AND-nodes. When γ becomes larger, it tends to give more weights to the AND-nodes with higher matching values, which subjects to the prior that the partial matching of two images on a tan is preferable to choose the way of partition with the most matched features newly found. Finally, the value of TMK for these two images is returned by the matching score of root OR-node in $\Upsilon^{'}_{\Delta}$. We summarize the computing process of our TMK in Alg. 2. Based on the proposed TMK, any kernel-based classifier can be applied to perform scene categorization.

If we set γ by its extreme values (i.e., ∞ and 0), the *MAX_OR* and *MEAN_OR* TMKs proposed in our previous work [41] can be deduced as follows: If $\gamma = \infty$, we obtain the *MAX_OR* TMK via a max operation over candidate child AND-nodes (i.e., $s_{v^{OR}} = \max_{u^{AND} \in Ch(v^{OR})} s_{u^{AND}}$). If $\gamma = 0$, the *MEAN_OR* TMK will be obtained by averaging the matching values of all child AND-nodes (i.e., $s_{v^{OR}} = \frac{1}{|Ch(v^{OR})|}\sum_{u^{AND}} s_{u^{AND}}$). Intuitively, the *MAX_OR* TMK adaptively searches a tangram parse tree with the most accumulated matched features between two images, while the *MEAN_OR* TMK is the most smooth TMK by means of averaging the matching values found w.r.t. different spatial constraints for the OR-nodes. Note that the proposed TMK cannot guarantee the positive-semi-definiteness and hence does not satisfy the Mercer's condition. However, as shown in Sec. VI-C, it can be

Algorithm 2: The Algorithm on Computing TMK

Input: images X and Y, AOG $\Upsilon^{'}_{\Delta}$
Output: the TMK value $TMK(X,Y)$

foreach $l = 1$ *to* L **do**
  foreach $i = 1$ *to* N_l *and* $j = 1$ *to* $J_{(l,i)}$ **do**
    Compute $s_{v^T_{(l,i,j)}}$ by the HIK on the histogram features of X and Y for $B_{(l,i,j)}$;
    foreach *AND-node* $o = 1$ *to* $O_{(l,i)}$ **do**
      Compute $s_{v^{AND}_{(l,i,j),o}}$ by Equ. (14);
    end
    if $l = 1$ **then**
      $s_{v^{OR}_{(l,i,j)}} = s_{v^T_{(l,i,j)}}$,
    end
    else
      Compute $s_{v^{OR}_{(l,i,j)}}$ by Equ. (15);
    end
  end
end
$TMK(X,Y) = s_{v^{OR}_{(L,1,1)}}$.

Fig. 6. Illustration of all the 33 scene configuration classes for the SceneConfig_33 dataset (see Sec. VI-A). We show one example image for each configuration class. The caption below each example image corresponds to its configuration class.

flawlessly used as a kernel for SVMs in practice and improves the performance consistently on scene categorization task.

VI. Experiments

In experiments, we first create a new image dataset on a variety of configurations of scene layout, and then construct a series of experimental evaluations on our tangram model for scene categorization.

A. The scene configuration dataset

In the literature of scene recognition, previous image datasets [3], [18], [6], [7] are mainly contributed to sematic-level categorization tasks, and thus do not have configuration-level ground-truth annotation information. To facilitate our investigation of learning configuration-level scene representation via the proposed tangram model, we build a new scene dataset[3] (called SceneConfig_33 in this paper) by selecting images from the MIT 8-class scene dataset [3], MIT_Indoor dataset and the LHI scene dataset [47]. It contains 10 semantic categories in total, consisting of 6 outdoor scene categories (*coast*, *highway*, *mountain*, *open country*, *street*, *city view*) and 4 indoor ones (*bedroom*, *store*, *meeting room*, *corridor*). For each semantic category, there are 120 to 250 images manually divided into 3 to 4 different configuration sub-categories (33 in total). Fig. 6 illustrates example images of all the 33 configuration classes for our SceneConfig_33 dataset.

B. Scene categorization based on the configuration bank representation

On basis of the learned tangram templates in Sec. IV, we apply the proposed TBank representation in Sec. III-D to scene categorization task, and compare it with the widely-used spatial pyramid representation in literature. In this subsection, we first test our method on the SceneConfig_33 dataset for configuration-level classification, and then evaluate the semantic-level classification performance on SceneConfig_33 as well as two public scene datasets (i.e., Scene_15 [6] and MIT_Indoor [7]) in scene categorization literature.

[3]http://www.stat.ucla.edu/ junzhu/dataset/SceneConfig_33.zip.

1) Experimental setup: To be consistent with [6] for comparison, we adopt the same densely sampled SIFT BOW feature in our experiments. Concretely, the SIFT features [43] are extracted from densely sampled 16×16 patches, in a grid with the step size of 8 pixels. Then, we randomly sample $100,000$ patches from training images, and construct a codebook with 200 visual words by using standard K-means clustering algorithm on their SIFT feature descriptors [6]. After that, a L1-normalized histogram of visual word frequency is computed for each tan instance, which is the BOW feature as the appearance descriptor of tan instance in Sec. III-B. According to Sec. III-D, we test the both cases of flat tangram template and multi-layered tangram template (abbreviated by *fTgm* and *mTgm* in following discussion) for our TBank representation.

Based on the proposed TBank representation of scene images, we use "one-vs-rest" classifiers for multi-class discrimination. Specifically, we train a binary linear support vector machine (SVM) or logistic regression (LR) classifier for each class individually, and then the class label of testing image is predicted as the one with the highest confident value output by corresponding classifier. We implement these linear classifiers by LIBLINEAR code package [48]. Following the scene categorization literature [3], [18], [6], [7], [9], the classification performance is measured by the average of per-class classification rates, which can be calculated as the mean value over the diagonal elements of resultant confusion matrix.

2) Evaluation on configuration-level scene categorization: In this experiment, we run 10 rounds of experiments with different random splits of training and testing images on SceneConfig_33 dataset. For each round, we randomly select 15 images from each configuration class for training and use the rest ones for testing. At first, we test the classification performance for the case of directly using the tangram template scores for classification. For each configuration class, we learn one tangram template from training images based on the DP algorithm in Sec. IV-B. Thus, the class label of testing image is simply identified as the one with maximum tangram template score according to Equ. (7). As shown in Fig. 8, we can see that the learned tangram templates consistently outperform the fixed-structure SP_BOW representation, for both of two different granularity levels of image lattice (i.e. 2×2 and 4×4 grids). Fig. 7 illustrates some top-ranked true positive

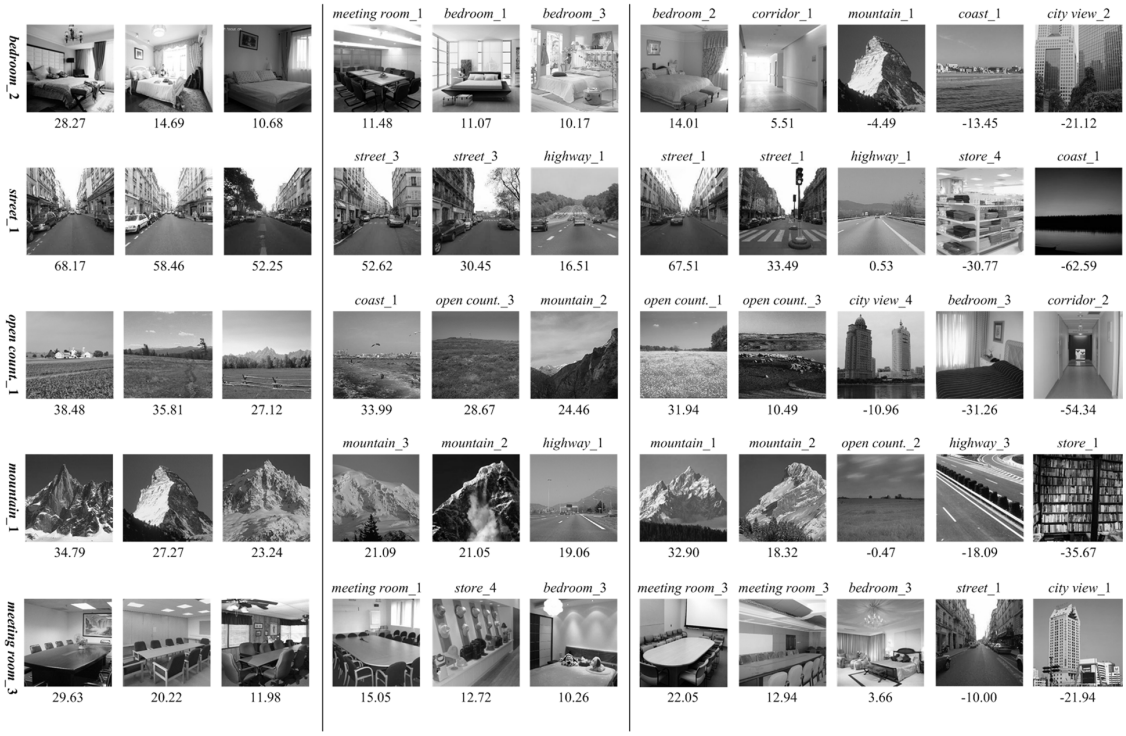


Fig. 7. Illustration of binary classification results based on tangram template scores (SceneConfig_33). Each row corresponds to a target scene configuration class to be learned. The caption on top of each image in panels (b) and (c) refers to its ground-truth configuration class, and the number below the image is the score of learned tangram template obtained for binary classification. (a) top-ranked true positive testing images; (b) top-ranked false positive testing images; (c) image examples from the testing set, which are sampled with roughly equal distance in descending order of tangram template scores. The goal is to visualize which images/classes are near and far away from the target scene configuration learned. (See Sec. VI-B2)

and false positive image examples by binary classification based on tangram template scores. We find that the tangram templates learned by our method can effectively capture visually meaningful spatial layout for different configuration-level scene categories.

After that, we investigate the classification performance of our TBank representation based on learned tangram templates. Given the tangram templates learned for all the configuration categories (33 in total), we build the TBank representation of scene images according to Sec. III-D for configuration-level classification. Fig. 8 shows that it can obtain superior accuracy than the case discussed above, in which we perform scene classification via maximization on the tangram template scores. This implies that the proposed TBank representation can provide useful information to make for scene recognition by considering the tangram template scores of other configurations. Moreover, from table II we can see that it achieves much higher classification performance than the high-dimensional SP_BOW representation (i.e., the performance gain is $6.6 - 8.4\%$) even with a fraction of dimension, i.e. only 33 dimension for TBank w.r.t. 1,000 (2×2 grid) or $4,200$ (4×4 grid) dimension for SP_BOW. It accounts for that our TBank representation based on learned tangram templates can provide higher-level information than original SIFT BOW features through effective knowledge abstraction of jointly capturing meaningful configuration and appearance cues.

3) Evaluation on semantic-level scene categorization: Besides the configuration-level scene categorization, we further apply our method to semantic-level categorization, which is one of the most concerned task in scene recognition literature. Rather than training only one tangram template for each class as in Sec. VI-B2, we first learn multiple tangram templates for each semantic-level scene category according to the scene configuration discovery method in Sec. IV-C. Then, the tangram bank is constructed by collecting all the learned tangram templates from different categories, and corresponding TBank representation of scene images can be obtained according to Sec. III-D for semantic-level scene categorization.

As mentioned in Sec. VI-A, the 33 configuration classes in SceneConfig_33 are collected from 10 different semantic categories, each of which consists of 3 or 4 manually divided configuration classes. Similar to Sec. VI-B2, we run 10-round experiments with different random splits of training and testing images. For each round, we randomly select 50 image examples from each semantic-level scene category for training and use the rest ones for testing. To construct the tangram bank, we learn 8 tangram templates for each semantic category, and thus it will produce an 80-dimensional TBank representation for each image (i.e., $\mathcal{T} = 8 \times 10 = 80$). As shown in table III, our TBank representation can obtain consistent performance gain ($5.2 - 6.0\%$) of semantic-level scene categorization w.r.t. the SP_BOW representation in each combination of image lattice granularity (i.e., 2×2 or 4×4 grid) and classifier type (i.e., SVM or LR).

For deeper analysis of our method, we further investigate the clustering-based scene configuration discovery method based on the information gain of learned tangram templates as similarity measurement, which is an intermediate step of

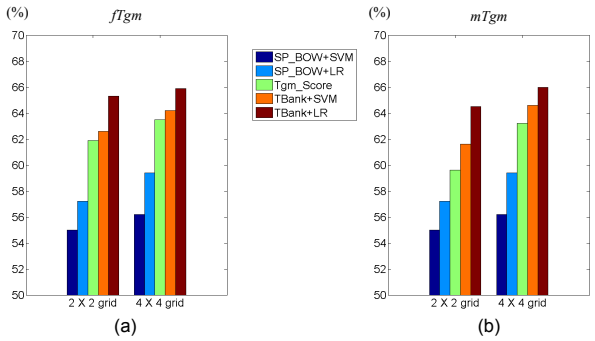


Fig. 8. Performance comparison for configuration-level scene categorization on the SceneConfig_33 dataset (see Sec. VI-B2). (a) *fTgm*, (b) *mTgm*. (Best viewed in color)

TABLE II
CLASSIFICATION RATES (%) FOR CONFIGURATION-LEVEL SCENE CATEGORIZATION (SceneConfig_33)

	2×2 grid		4×4 grid	
	SVM	LR	SVM	LR
SP_BOW [6]	55.0 ± 1.0	57.2 ± 1.2	56.2 ± 0.9	59.4 ± 0.8
TBank_*fTgm*	**62.6** ± 1.0	**65.3** ± 0.7	64.2 ± 1.2	65.9 ± 1.6
TBank_*mTgm*	61.6 ± 0.9	64.5 ± 0.9	**64.6** ± 1.0	**66.0** ± 1.1

TABLE III
CLASSIFICATION RATES (%) FOR SEMANTIC-LEVEL SCENE CATEGORIZATION (SceneConfig_33)

	2×2 grid		4×4 grid	
	SVM	LR	SVM	LR
SP_BOW [6]	78.4 ± 0.9	79.1 ± 1.0	78.7 ± 1.4	79.5 ± 1.1
TBank_*fTgm*	83.8 ± 0.9	84.4 ± 0.9	83.4 ± 0.9	83.4 ± 0.9
TBank_*mTgm*	**84.4** ± 1.0	**84.6** ± 0.9	**84.4** ± 0.7	**84.7** ± 0.9

building the TBank representation for semantic-level scene categorization. Concretely, we use the exemplar-based clustering method described in Sec. IV-C on the training images to learn the same number of scene configurations as the manually divided ones for each semantic-level scene category, and then compute the empirical purity and conditional entropy of clustering results, which are the common evaluation criteria used in unsupervised object category discovery literature [49], [50], [51]. Let $\mathcal{X}$ and $\mathcal{Y}$ denote the sets of ground-truth class labels and the resultant cluster labels, respectively. As described in [50], the *purity* is defined defined as the mean of the maximum class probabilities of $\mathcal{X}$ given $\mathcal{Y}$. That is

$$\text{Purity}(\mathcal{X}|\mathcal{Y}) = \sum_{y \in \mathcal{Y}} p(y) \max_{x \in \mathcal{X}} p(x|y), \tag{16}$$

where $p(y)$ and $p(x|y)$ represent the prior distribution of cluster label y and the conditional probability of ground-truth label x given y respectively. In practice, we can only compute the frequency estimation of $p(y)$ and $p(x|y)$ from the observed samples used in clustering, and thus obtain the empirical purity on a given set of images as the clustering quality metric. In this experiment, the manual annotation of scene configurations is used to determine the ground-truth class label for each image of SceneConfig_33. Besides purity, we can use the *conditional entropy* of $\mathcal{X}$ given $\mathcal{Y}$ to assess the clustering result. As defined in 17, it measures the average uncertainty of $\mathcal{X}$ if the value of $\mathcal{Y}$ is known [50].

$$\text{Entropy}(\mathcal{X}|\mathcal{Y}) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p(x|y) \log \frac{1}{p(x|y)}. \tag{17}$$

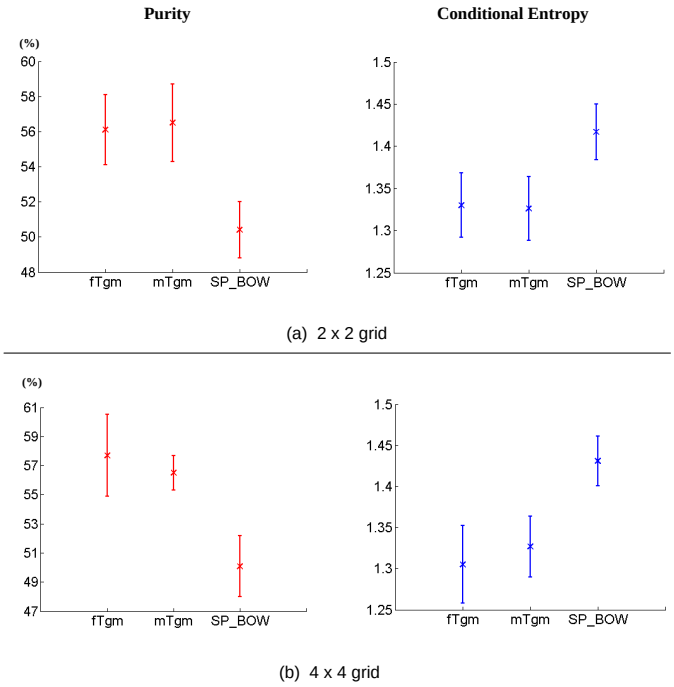


Fig. 9. Analysis on the exemplar-based clustering method for scene configuration discovery. We compare the quality of unsupervised clustering results obtained by three different similarity measurements (i.e., *fTgm*, *mTgm*, SP_BOW). The fTgm and mTgm on horizontal axis correspond to the similarity measurements based on pair-wise information gain of flat tangram template and the multi-layered one respectively. SP_BOW indicates the similarity measurement based on spatial pyramid BOW representation. All of them use the same affinity propagation clustering algorithm [16] to obtain the results. For performance evaluation, the empirical purity and conditional entropy are adopted, and we show their mean value and standard deviation of 10-round experiments with randomly selected training images by using an error bar plot. Please see Sec. VI-B3 for details. (a) and (b) show the results of 2×2 grid and 4×4 grid respectively.

Please refer to [50] for detailed description about purity and conditional entropy. Intuitively, the quality of unsupervised category discovery will be better as the purity is higher or the conditional entropy is smaller.

Fig. 9 shows the average empirical purity and conditional entropy of clustering results on SceneConfig_33 dataset. We can see that our methods (i.e., *fTgm* and *mTgm*) consistently outperform the SP_BOW. The purity of our methods is higher than that of SP_BOW by $5.7 - 7.6\%$. The conditional entropy shows similar tendency of performance superiority as the purity measurement. This experiment shows our tangram model can produce more applausive clusters w.r.t. the manual annotation than the SP_BOW representation, and validates the effectiveness of the exemplar-based clustering algorithm for learning multiple scene configurations from a single semantic-level category. Besides, Fig. 10 compares the histograms of intra-class and inter-class information gain values for some semantic categories, which are obtained from two images of same configuration class and different classes respectively. We can see that the intra-class information gain has a heavy tail distribution than the inter-class one, implying its effectiveness as similarity measurement used for the exemplar-based clustering algorithm.

Moreover, We give a quantitative analysis on the semantic-level classification performance w.r.t. the number of tangram

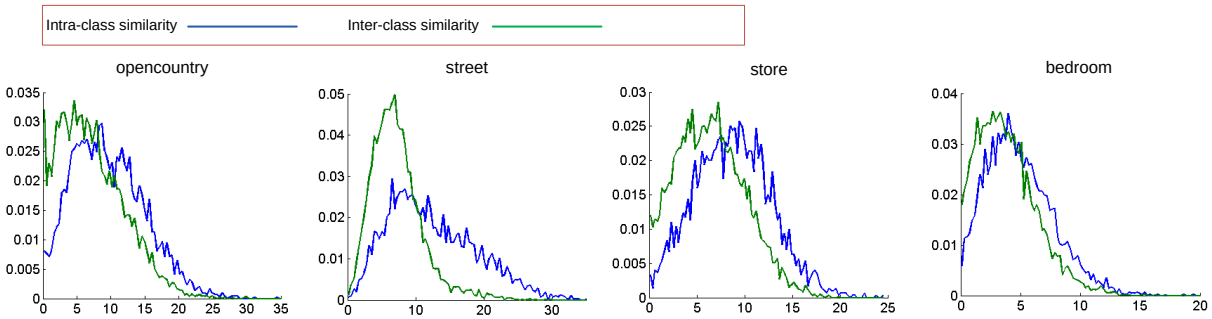


Fig. 10. Analysis on the information gain of learned tangram templates as similarity measurement used in exemplar-based clustering. The horizontal axis represents the information gain, and the vertical axis represents normalized histogram value. The intra-class similarity and inter-class one correspond to the pair-wise information gain obtained from two images of same configuration class and different classes, respectively. Please see Sec. IV-C for the details of our exemplar-based learning algorithm. (best viewed in color)

TABLE IV
CLASSIFICATION RATES (%) FOR SEMANTIC-LEVEL SCENE CATEGORIZATION (Scene_15)

	2×2 grid		4×4 grid	
	SVM	LR	SVM	LR
SP_BOW [6]	73.5 ± 0.8	75.0 ± 0.6	74.5 ± 0.6	75.8 ± 0.7
TBank_*fTgm*	79.8 ± 0.6	80.2 ± 0.7	79.7 ± 0.5	80.3 ± 0.6
TBank_*mTgm*	**80.0** ± 0.7	**80.3** ± 0.6	**80.8** ± 0.7	**81.1** ± 0.7

TABLE V
CLASSIFICATION RATES (%) FOR SEMANTIC-LEVEL SCENE CATEGORIZATION (MIT_Indoor)

	2×2 grid		4×4 grid	
	SVM	LR	SVM	LR
SP_BOW [6]	28.5	31.4	30.8	32.4
TBank_*fTgm*	34.9	37.3	35.8	37.9
TBank_*mTgm*	**36.3**	**38.5**	**36.9**	**39.7**

templates learned per semantic category, and compare our clustering-based TBank representation to that based on manual scene configuration annotation. As shown in Fig. 11 (a), the classification accuracy generally increases as more tangram templates used for constructing the TBank representation (i.e., $\mathcal{T}$ becomes larger). However, the performance improvement tends to be saturated when $\mathcal{T}$ achieves a certain intermediate number, and continued increase on the dimension of TBank does not result in notable performance improvement further. Particularly, the performance gain on the use of 8 templates w.r.t. only one per category is $4.4 - 6.2\%$, validating the effectiveness of discovering multiple tangram templates for semantic-level scene categorization. Compared to the TBank representation built from manually annotated configurations (see the green-circle and purple-triangle markers), our method also obtains superior accuracy consistently when the number of clusters per category is more than 3. Above observations demonstrate that the proposed method in Sec. IV-C can effectively learn a variety of informative tangram templates for each category, leading to a discriminative and compact TBank representation for semantic-level scene categorization.

Besides SceneConfig_33, we further test the semantic-level classification performance of our method on two benchmark scene datasets (i.e., Scene_15 and MIT_Indoor). The experimental settings are listed as follows:

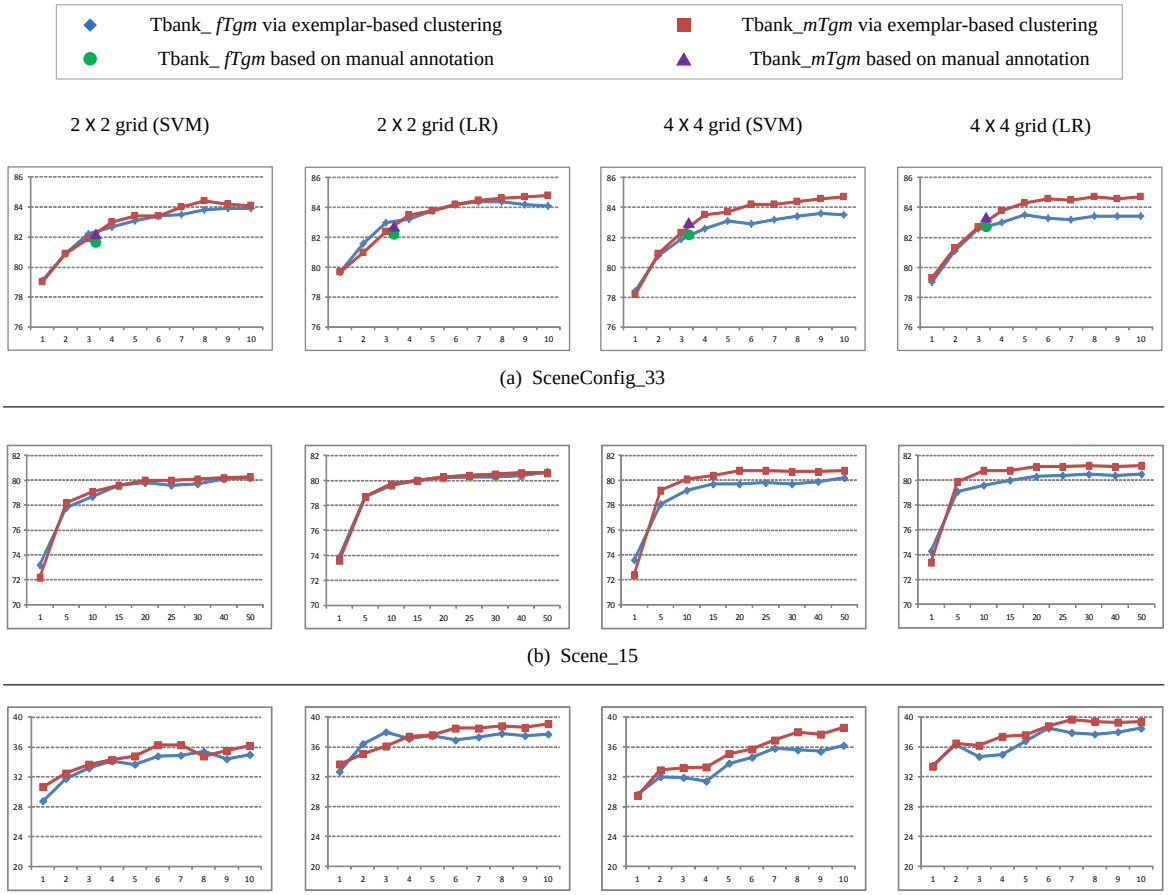
- Scene_15: This dataset consists of 15 different semantic scene categories involving outdoor natural scenes (e.g., coast, mountain and street) and indoor ones (e.g., bedroom, office room). It contains 4485 images in total, with a varying number of images from 200 to 400 per category. Following [6], we repeat 10 rounds of experiments with different randomly selected training and testing images. For each round, there are 100 images per class used for training and the remaining ones for testing. For this dataset, we learn 20 tangram templates for each class, leading to a 300-dimensional TBank representation for each image (i.e., $\mathcal{T} = 20 \times 15 = 300$).
- MIT_Indoor: It contains 15,620 images in total, which are distributed into 67 indoor scene categories. We use the same training images (80 samples per class) and testing ones (20 samples per class) in [7].[4] The number of tangram templates learned per class is set by 7, and thus we obtain a 469-dimensional TBank representation for each image(i.e., $\mathcal{T} = 7 \times 67 = 469$).

Tables IV and V list the classification rates for Scene_15 and MIT_Indoor, respectively. As shown in table IV, our TBank representation outperforms SP_BOW by $5.3-6.5\%$ for Scene_15 dataset. For more challenging MIT_Indoor dataset, the performance gain increases to $6.1 - 7.8\%$ as shown in table V. As shown in table VI, the dimension of our TBank representation is much less than that of SP_BOW. Besides, the Fig. 11 (b) and (c) illustrates the curves of classification performance w.r.t. the number of tangram templates learned per category for Scene_15 and MIT_Indoor respectively. They show similar observations of variation trend as SceneConfig_33 in Fig. 11 (a), indicating good generalizability of our TBank representation based on the exemplar-based clustering method. Besides, in table VII we also compare it with other scene representations (i.e., Gist [3] and OB [9]) in literature. All these feature representations are tested with the linear LR classifier. We can see that our method even outperforms the spatial pyramid model with high-level features, e.g. the object detectors' responses in OB representation [9], which validates the significance and advantage of leveraging configuration cues for scene recognition.

C. Scene categorization by tangram matching kernel

Besides the methods of generatively learned tangram templates for the TBank representation in Sec. III and IV, we

[4]This dataset as well as the list of training and testing samples can be downloaded from *http://web.mit.edu/torralba/www/indoor.html*.



(a) SceneConfig_33

(b) Scene_15

(c) MIT_Indoor

Fig. 11. The effect of the number of tangram templates learned per class for our TBank representation (see Sec. VI-B3). The horizontal axis indicates the number of tangram templates learned per category, and the vertical axis represents the classification rate (%). (a), (b) and (c) correspond to the results of SceneConfig_33, Scene_15 and MIT_Indoor, respectively. We show four cases of different combination of scene image lattice and classifier type from left to right: 2×2 grid with SVM, 2×2 grid with LR, 4×4 grid with SVM, and 4×4 grid with LR. The curves with blue diamond markers and red square ones correspond to *fTgm* and *mTgm*, respectively. In (a), the results based on manually annotated 33 scene configurations are also shown and compared with those via exemplar-based clustering. The green circles and purple triangles correspond to *fTgm* and *mTgm*, respectively. (best viewed in color and magnification)

TABLE VI
COMPARISON ON THE DIMENSION OF REPRESENTATION (THE DIMENSIONS OF OUR TANGRAM BANK REPRESENTATION ARE EQUAL FOR THE TWO CASES OF *fTgm* AND *mTgm*.)

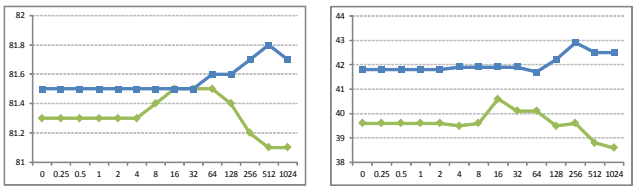
	2×2 grid		4×4 grid	
	SP_BOW	TBank	SP_BOW	TBank
Scene_15	1000	300	4200	300
MIT_Indoor	1000	469	4200	469

TABLE VII
CLASSIFICATION RATE (%) COMPARISON OF OUR TBANK WITH OTHER SCENE REPRESENTATIONS IN LITERATURE (WITH LR CLASSIFIER)

	SP_BOW [6]	Gist [3]	OB [9]	TBank	
				fTgm	*mTgm*
Scene_15	75.8	71.8	80.9	80.3	**81.1**
MIT_Indoor	32.4	23.5	37.6	37.9	**39.7**

evaluate the TMK proposed in Sec. V on the Scene_15 and MIT_Indoor datasets, and compare it with other methods in scene categorization literature. We adopt the same appearance feature and experimental settings as in Sec. VI-B3. In this experiment, the "one-vs-rest" criterion is used for multi-class classification, and each binary classifier is trained via a kernel SVM with the implementation of LIBSVM code package [52].

At first, we analyze the effect of parameter γ used in the *SOFT_MAX_OR* TMK, and draw a comparison with its



(a) Scene_15 (b) MIT_Indoor

Fig. 12. Illustration on classification performance vs. the value of parameter γ (see Sec. VI-C). The horizontal axis indicates the value of γ, and vertical axis represents the classification rate (%). For each panel, we show the results of 2×2 and 4×4 grids by green and blue curves respectively. (a) Scene_15, (b) MIT_Indoor. (best viewed in color and magnification)

extreme cases (i.e., *MEAN_OR* and *MAX_OR* TMKs). As shown in Fig. 12, we can find that there exists an intermediate number as the optimal value of γ for achieving the highest classification performance, which is superior to either the *MEAN_OR* or *MAX_OR* TMK. It implies that the optimum matching kernel based on the tangram model should be in an intermediate degree of smoothness to "marginalize" the parse trees corresponding to various spatial configurations.

Besides, the *MEAN_OR* and *MAX_OR* TMKs define t-wo different kinds of image similarity measurement: The *MEAN_OR* TMK is the most smooth one among the family of

TABLE VIII
CLASSIFICATION RATES (%) FOR TMKS

	Scene_15		MIT_Indoor	
	2×2	4×4	2×2	4×4
SPM [6]	79.2 ± 0.5	81.2 ± 0.4	37.5	38.8
SOFT_MAX_OR TMK	81.5 ± 0.4	**81.8** ± 0.5	40.6	42.9
MEAN_OR TMK	81.3 ± 0.5	81.5 ± 0.5	39.6	41.8
MAX_OR TMK	81.1 ± 0.4	81.7 ± 0.5	38.5	42.3
The composite TMK	**81.7** ± 0.5	**81.8** ± 0.4	**43.2**	**43.9**
RBoW [37]	78.6 ± 0.7		37.93	
DPM+GIST-color+SP [54]	N/A		43.1	
CENTRIST [29]	83.88 ± 0.76		36.88	
OB [9]	80.9		37.6	
MM-Scene [55]	N/A		28.1	
ScSPM [8]	80.28 ± 0.93		N/A	
[7]	N/A		26	

all possible TMKs, which indicates to "average" the matching measurements over all the parse trees of AOG. On the contrary, the *MAX_OR* TMK only consider the parse tree of spatial configuration giving the highest matching similarity between two images, out of all possible parse trees generated by the AOG. Thus, these two kinds of TMKs correspond to diverse underlying feature spaces and have distinct properties for classification. According to the kernel combination theory [53], we propose to use a product composite kernel based on the *MEAN_OR* and *MAX_OR* TMKs to boost classification performance further.

In table VIII, we show the classification rates of different TMKs on the Scene_15 and MIT_Indoor datasets, and compare it with the spatial pyramid counterpart (i.e. SPM kernel [6]) as well as previous methods in scene categorization literature. As shown in table VIII, our TMKs can obtain superior classification performance than SPM in both 2×2 and 4×4 grids of image lattice, which supports our motivation that using richer configuration cues as well as inducing the OR-nodes would make for scene recognition. Particularly, we observe that our method outperforms SPM in a large margin (i.e., performance improvement of 4.1% for *SOFT_MAX_OR* TMK and 5.1% for the composite TMK) on MIT_Indoor dataset. It may be caused by the fact that the indoor scene categories involve more complicated configuration variations than natural outdoor scenes, asking for a more sophisticated way to explore scene layouts as our tangram model does.

VII. CONCLUSION

Exploring scene layouts is a challenging task and also very important for scene categorization. In this paper, we present a reconfigurable tangram model for scene layout representation, and propose a method of learning a mixture of tangram models for representing scene category by combing an exemplar-based clustering method and a DP algorithm. The proposed tangram model goes beyond the traditional quad-tree like decomposition methods which explore scene layouts in a predefined and fixed manner. On basis of the tangram model, two methods are proposed to address scene categorization: building a configuration bank representation of scene images for linear classification, and building a tangram matching kernel for kernel-based classification. The experimental results show that our methods consistently outperform the widely used spatial pyramid representation on three scene datasets (i.e., a 33-category scene configuration dataset, an 15-category scene dataset [6] and a 67-category indoor scene dataset [7]).

ACKNOWLEDGMENT

This work is mainly done when Jun Zhu is a visiting student at LHI. We thank the support of the DARPA SIMPLEX project N66001-15-C-4035 and NSFC programs (61025005, 61129001, 61221001). The authors would like to thank Dr. Yingnian Wu and Dr. Alan Yuille for helpful discussions.

REFERENCES

[1] S. Thorpe, D. Fize, and C. Marlot, "Speed of processing in the human visual system," *Nature*, vol. 381, pp. 520–522, 1996.
[2] M. V. Peelen, L. Fei-Fei, and S. Kastner, "Neural mechanisms of rapid natural scene categorization in human visual cortex," *Nature*, vol. advanced online publication, 2009.
[3] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.
[4] P. Lipson, W. Grimson, and P. Sinha, "Configuration based scene classification and image indexing." in *CVPR*, 1997.
[5] T. Konkle, T. Brady, G. Alvarez, and A. Oliva, "Scene memory is more detailed than you think: the role of categories in visual long-term memory." *Psychological Science*, vol. 21, no. 11, pp. 1551–1556, 2010.
[6] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.
[7] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *CVPR*, 2009.
[8] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *CVPR*, 2009.
[9] L.-J. Li, H. Su, E. P. Xing, and L. Fei-Fei, "Object bank: A high-level image representation for scene classification semantic feature sparsification," in *NIPS*, 2010.
[10] S. Geman, D. Potter, and Z. Y. Chi, "Composition systems," *Quarterly of Applied Mathematics*, vol. 60, no. 4, pp. 707–736, 2002.
[11] S.-C. Zhu and D. Mumford, "A stochastic grammar of images," *Foundations and Trends in Computer Graphics and Vision*, vol. 2, no. 4, pp. 259–362, 2006.
[12] B. Li, T. Wu, and S.-C. Zhu, "Integrating context and occlusion for car detection by hierarchical and-or model," in *ECCV*, 2014.
[13] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, 2010.
[14] Z. Si and S.-C. Zhu, "Learning and-or templates for object recognition and detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2013.
[15] Z. Z. Si and S.-C. Zhu, "Learning hybrid image template (hit) by information projection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1354–1367, 2012.
[16] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007.
[17] K. Grauman and T. Darrell, "The pyramid match kernel: Efficient learning with sets of features," 2005.
[18] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *CVPR*, 2005.
[19] Q. Zhou, J. Zhu, and W. Liu, "Learning dynamic hybrid markov random field for image labeling," *IEEE Trans. on Image Process.*
[20] Y. Zhu, J. Zhu, and R. Zhang, "Discovering spatial context prototypes for object detection," in *ICME*, 2013.
[21] Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler, "segdeepm: Exploiting segmentation and context in deep neural networks for object detection," in *CVPR*, 2015.
[22] J. Zhu, W. Zou, X. Yang, R. Zhang, Q. Zhou, and W. Zhang, "Image classification by hierarchical spatial pooling with partial least squares analysis," in *BMVC*, 2012.
[23] J. Ma, H. Zhou, J. Zhao, Y. Gao, J. Jiang, and J. Tian, "Robust feature matching for remote sensing image registration via locally linear transforming," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6469–6481, 2015.
[24] J. Ma, W. Qiu, J. Zhao, Y. Ma, A. L. Yuille, and Z. Tu, "Robust L_2E estimation of transformation for non-rigid registration," *IEEE Trans. Signal Process.*, vol. 63, no. 5, pp. 1115–1129, 2015.

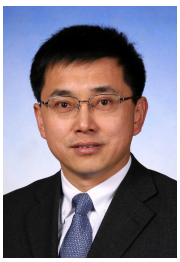
[25] M. Szummer and R. W. Picard, "Indoor-outdoor image classification," in *IEEE Intl. Workshop on Content-Based Access of Image and Video Databases*, 1998.
[26] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 133–157, 2007.
[27] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. J. V. Gool, "Modeling scenes with local descriptors and latent aspects," in *ICCV*, 2005.
[28] A. Bosch, A. Zisserman, and X. Muñoz, "Scene classification via plsa," in *ECCV*, 2006.
[29] J. Wu and J. M. Rehg, "Centrist: A visual descriptor for scene categorization." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489–1501, 2011.
[30] A. Oliva and A. Torralba, "Building the gist of a scene: The role of global image features in recognition," *Visual Perception, Progress in Brain Research*, 2006.
[31] A. Torralba and A. Oliva, "Statistics of natural images categories." *Network: Computation in Neural Systems*, pp. 391–412, 2003.
[32] I. Biederman, *Visual object recognition*. MIT Press, 1995, vol. 2.
[33] P. Schyns and A. Oliva, "From blobs to boundary edges: Evidence for time- and spatial-scale-dependent scene recognition," *Psychological Science*, vol. 5(4), pp. 195–200, 1994.
[34] A. Oliva and P. Schyns, "Diagnostic colors mediate scene recognition," *Cognitive Psychology*, vol. 41, pp. 176–210, 2000.
[35] X. Wang, B. Feng, X. Bai, W. Liu, and L. J. Latecki, "Bag of contour fragments for robust shape classification," *Pattern Recognition*, vol. 47, no. 6, pp. 2116–2125, 2014.
[36] Y. Wu, Z. Si, H. Gong, and S.-C. Zhu, "Learning active basis model for object detection and recognition," *International Journal of Computer Vision*, vol. 90, no. 2, pp. 198–235, 2010.
[37] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb, "Reconfigurable models for scene recognition." in *CVPR*, 2012.
[38] J. Sun and J. Ponce, "Learning discriminative part detectors for image classification and cosegmentation," in *ICCV*, 2013.
[39] C. Doersch, A. Gupta, and A. A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *NIPS*, 2013.
[40] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification," in *ICCV*, 2013.
[41] J. Zhu, T. Wu, S.-C. Zhu, X. Yang, and W. Zhang, "Learning reconfigurable scene representation by tangram model," in *WACV*, 2012.
[42] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.
[43] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[44] A. Barla, F. Odone, and A. Verri, "Histogram intersection kernel for image classification," in *ICIP*, 2003.
[45] J. Wu and J. Rehg, "Beyond the euclidean distance: creating effective visual codebooks using the histogram intersection kernel," in *ICCV*, 2009.
[46] S. D. Pietra, V. J. D. Pietra, and J. D. Lafferty, "Inducing features of random fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 380–393, 1997.
[47] B. Yao, X. Yang, and S. C. Zhu., "Introduction to a large scale general purpose ground truth dataset: methodology, annotation tool, and benchmarks," in *EMMCVPR*, 2007.
[48] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.
[49] J. Sivic, B. C. Russell, A. Efros, A. Zisserman, and W. T. Freeman, "Discovering object categories in image collections," in *CVPR*, 2005.
[50] T. Tuytelaars, C. H. Lampert, M. B. Blaschko, and W. Buntine, "Unsupervised object discovery: A comparison," *International Journal on Computer Vision*, vol. 88, no. 2, pp. 284–302, 2009.
[51] D. X. Dai, T. F. Wu, and S.-C. Zhu, "Discovering scene categories by information projection and cluster sampling," in *CVPR*, 2010.
[52] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.
[53] T. Damoulas and M. A. Girolami, "Combining feature spaces for classification," *Pattern Recognition*, vol. 42, no. 11, pp. 2671–2683, 2009.
[54] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *ICCV*, 2011.
[55] J. Zhu, L.-J. Li, L. Fei-Fei, and E. P. Xing, "Large margin learning of upstream scene understanding models," in *NIPS*, 2010.

Jun Zhu received a Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2013. He is currently a postdoctoral research fellow in UCLA Center for Cognition, Vision, and Learning. His research interests include computer vision and machine learning, mainly focusing on (i) Hierarchical and compositional models for visual scene and object recognition; (ii) Weakly-supervised learning for semantic segmentation and object detection; (iii) Human action recognition in videos.

Tianfu Wu received a Ph.D. degree in Statistics from University of California, Los Angeles (UCLA) in 2011. He is currently a research assistant professor in the Center for Vision, Cognition, Learning and Autonomy at UCLA. His research interests are in computer vision and machine learning, with a focus on (i) Statistical learning of large scale hierarchical and compositional models (e.g., And-Or graphs). (ii) Statistical inference by near-optimal cost-sensitive decision policies. (iii) Statistical theory of performance guaranteed learning and inference algorithms.

Song-Chun Zhu received a Ph.D. degree from Harvard University in 1996. He is currently professor of Statistics and Computer Science at UCLA, and director of the Center for Vision, Cognition, Learning and Autonomy. His research interests include computer vision, statistical modeling and learning, cognition, robot autonomy, and visual arts. He received a number of honors, including the J.K. Aggarwal prize from the Int'l Association of Pattern Recognition in 2008 for "contributions to a unified foundation for visual pattern conceptualization, modeling, learning, and inference", the David Marr Prize in 2003 with Z. Tu et al. for image parsing, twice Marr Prize honorary nominations in 1999 for texture modeling and in 2007 for object modeling with Z. Si and Y.N. Wu. He received the Sloan Fellowship in 2001, a US NSF Career Award in 2001, an US ONR Young Investigator Award in 2001, and the Helmholtz Test-of-time award in ICCV 2013. He is a Fellow of IEEE since 2011.

Xiaokang YANG received the B. S. degree from Xiamen University, Xiamen, China, in 1994, the M. S. degree from Chinese Academy of Sciences, Shanghai, China, in 1997, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2000. He is currently a Distinguished Professor of School of Electronic Information and Electrical Engineering, and the deputy director of the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China. He has published over 200 refereed papers, and has filed 40 patents. He is Associate Editor of IEEE Transactions on Multimedia and Senior Associate Editor of IEEE Signal Processing Letters. His current research interests include visual signal processing and communication, media analysis and retrieval, and pattern recognition.



Wenjun Zhang received his B.S., M.S. and Ph.D. degrees in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 1984, 1987 and 1989, respectively. He is a full professor of Electronic Engineering in Shanghai Jiao Tong University. As the project leader, he successfully developed the first Chinese HDTV prototype system in 1998. He was one of the main contributors of the Chinese DTTB Standard (DTMB) issued in 2006. He holds more than 40 patents and published more than 110 papers in international journals and conferences. He is the Chief Scientist of the Chinese Digital TV Engineering Research Centre, an industry/government consortium in DTV technology research and standardization, and the director of Cooperative MediaNet Innovation Center (CMIC)an excellence research cluster affirmed by the Chinese Government. His main research interests include digital video coding and transmission, multimedia semantic processing and intelligent video surveillance.