From Region Based Image Representation to Object Discovery and Recognition

Narendra Ahuja[1] and Sinisa Todorovic[2]

[1] Department of Electrical and Computer Engineering,
Coordinated Science Lab, and Beckman Institute,
University of Illinois Urbana-Champaign
[2] School of Electrical Engineering and Computer Science,
Oregon State University

Abstract. This paper presents an overview of the work we have done over the last several years on object recognition in images from region-based image representation. The overview focuses on the following related problems: (1) discovery of a single 2D object category frequently occurring in a given image set; (2) learning a model of the discovered category in terms of its photometric, geometric, and structural properties; and (3) detection and segmentation of objects from the category in new images. Images in the given set are segmented, and then each image is represented by a region graph that captures hierarchy and neighbor relations among image regions. The region graphs are matched to extract the maximally matching subgraphs, which are interpreted as instances of the discovered category. A graph-union of the matching subgraphs is taken as a model of the category. Matching the category model to the region graph of a new image yields joint object detection and segmentation. The paper argues that using a hierarchy of image regions and their neighbor relations offers a number of advantages in solving (1)-(3), over the more commonly used point and edge features. Experimental results, also reviewed in this paper, support the above claims. Details of our methods as well of comparisons with other methods are omitted here, and can be found in the indicated references.

1 Introduction

This paper presents an overview of the region based approach to object recognition and related problems that we have developed over the last several years, and briefly explains its advantages over the more commonly used methods based on point and edge features (e.g., [1, 20, 21, 32, 39, 52, 59, 65]). We briefly describe the major components of our work; details can be found in [6, 54–56].

As a way of addressing recognition-related issues, we consider the following problem. Suppose we are given a set of arbitrary, unlabeled images that contains frequent occurrences of 2D objects from an unknown category. Whether, and where, any objects from the category occur in a specific image from the set is unknown. We are interested in extracting instances of the category from the image set, and obtaining a compact category model in terms of photometric, and geometric and other structural properties. A model derived from such training can then be used to determine whether a new test

E.R. Hancock et al. (Eds.): SSPR & SPR 2010, LNCS 6218, pp. 1–19, 2010.

image contains objects from the learned category, and when it does, to segment all occurrences of the object.

This problem brings together most recognition related issues of interest here, and serves well to highlight the strengths and shortcomings of different approaches. Our region based formulation of this problem, originally presented in [6, 54–56], offers a general framework, subsumes most existing region-based methods, and achieves best performance on challenging benchmark datasets, including Caltech-101 and Caltech-256 [20], and Weizmann Horses [9]. We have shown that our approach:

1. Facilitates access to important object properties that are frequently used as recognition cues, including
   (a) Photometric (e.g., color, brightness),
   (b) Geometric (e.g., size, shape), and
   (c) Structural properties (e.g., layout and recursive embedding of object parts), and
2. Allows simultaneous detection and segmentation, of the target objects and their parts;
3. Simplifies object representation, e.g., for use as statistical models for object classification;
4. Allows efficient and robust learning and inference of object models; and
5. Enables object modeling under various degrees of supervision, including no supervision.

In this paper, we review the part of our work related to objects belonging to a single category [6, 54–56]. Our approach therein consists of four major steps. Given an arbitrary image set, in step 1, each image is segmented using a multiscale segmentation algorithm, and then represented by a region graph capturing the hierarchical and neighbor relations among image regions. Nodes of this graph correspond to regions, ascendant-descendant edges capture their recursive embedding, and lateral edges represent neighbor relations with sibling regions, i.e., those other regions that are embedded within the same parent region. The root of the graph represents the entire image. Step 2 discovers frequent occurrences of an object category in the images by searching for their similar subimages. This is done by matching the corresponding region graphs, and finding their common subgraphs. The set of maximally matching subgraphs is interpreted as occurrences of the discovered object category. In step 3, the matching subgraphs are fused into a single graph-union, which is taken to constitute the canonical model of the discovered object. The graph-union is defined as the smallest graph which contains every subgraph extracted in step 2. In step 4, a newly encountered image is also represented by the region graph that captures the hierarchical and neighbor relations among the image regions. This region graph is then matched with the graph-union model learned in step 3 to simultaneously detect and segment all occurrences of the category in the new image. This matching also identifies object parts along with their containment and neighbor relationships present, which can be used as an explanation of why each object is recognized.

We have also investigated the following other closely related recognition problems, the work on which we will not review in this paper. In [5], we presented a region-based method for extracting a taxonomy of categories from an arbitrary image set. The taxonomy captures hierarchical relations between the categories, such that layouts of

frequently co-occurring categories (e.g., head, body, legs, and tail) define more complex, parent categories (e.g., horse). The taxonomy also encodes sharing of categories among different ascendant categories. In the rest of this paper, by "hierarchy" we will refer to both region embedding and their neighbor relations, or layout. As demonstrated in [5], the above hierarchical region-based image representation improves the efficiency of search for shared categories; the available inter-category taxonomy yields sublinear complexity of recognizing all categories that may be present in the image set. Also, in [55], we showed that a hierarchy of regions helps capturing contextual properties of an object (e.g., co-occurrence statistics, and layout of other objects in the vicinity). This is used for estimating the significance of detecting a category in pointing to the presence of other, co-occurring categories in the image. Finally, in [4, 57], we addressed two related problems, that of texture segmentation, and detecting and segmenting the texture elements, called texels. An image texture can be characterized by statistical variations of the photometric, geometric, and structural properties of texels, and relative orientations and displacements of the texels. Since regions facilitate direct capturing these texel properties, our region-based approach outperforms existing methods on benchmark datasets.

The remainder of this paper is organized as follows. Sec. 2 briefly reviews different image features frequently used for recognition. Extraction of a hierarchy of regions from an image is presented in Sec. 3. Sec. 4.1 explains how to discover frequent occurrences of an object category by matching the region hierarchies of a given set of images. Fusing the matching subgraphs into a graph-union, which constitutes the object model, is presented in Sec. 4.2. Finally, Sec. 5 presents some of our empirical results that demonstrate the advantages of using hierarchical region-based image representations for single-category discovery, modeling, and recognition.

2 Regions as Image Features

Recent work typically uses point-based features (e.g., corners, textured patches) and edges (e.g., Canny, Berkeley's edge map) to represent images [16, 35–37, 48]. Interest points and edges have been shown to exhibit invariance to relatively small affine transforms of target objects across the images [35, 37, 48]. However, there are a number of unsatisfying aspects associated with point features and edges. They are usually defined only in terms of local, gray-level discontinuities (e.g., gradients of brightness), whereas target object occurrences in the image occupy regions. Therefore, the inherent locality of points and edges is dimensionally mismatched with the full 2D spatial extent of objects in the image. As a direct consequence, point-based object detection requires the use of scanning windows of pre-specified size and shape, and often result in multiple, overlapping, candidate detections that need to be resolved in a postprocessing step (e.g., non-maxima suppression). This postprocessing is usually based on heuristic assumptions about the numbers, sizes, and shapes of objects present. Since the final result of this is identification of the points associated with detected objects, it leads to only approximate object localization, not exact object segmentation . To obtain object segmentation, usually the probabilistic map is thresholded which provides likely object locations. This suffers from errors because both locations of local features and the threshold values depend on the particular scene and imaging conditions.

A number of approaches, including our previous work, use image regions as features [2, 6, 8, 11, 18, 27, 28, 31, 43, 55, 56, 64, 67–69]. These methods argue that regions are in general richer descriptors, more discriminative, and more noise-tolerant than interest points and edges. Regions are dimensionally matched with object occurrences in the image. Therefore, regions make various constraints, frequently used in object recognition—such as those dealing with continuation, smoothness, containment, and adjacency—implicit and easier to incorporate than points and edges. Region boundaries coincide with the boundaries of objects and their subparts. This allows simultaneous object detection and segmentation. Since there are fewer regions than local features, using regions often leads to great computational savings, and better performance because, e.g., the number of outliers is significantly reduced.

As always, it is worth noting that the impact of any shortcomings of an image segmentation algorithm should not be confused with the weaknesses of region based representation. For example, oversimplifying assumptions made by some segmentation algorithms about shape, curvature, size, gray-level contrast, and topological context of regions to be expected in an image [24, 38] may lead to segmentation errors of specific types. The same holds for algorithms that implement scale as input parameter which controls the degree of image blurring and subsampling for segmentation [10, 34], or pre-select the number of regions as input parameter [49]. In addition, most segmentation algorithms also use an oversimplified model of photometric profiles of image regions, as being homogeneous and surrounded by step discontinuities, instead of the more realistic ramp (non-step) discontinuities. Therefore, many regions in real images with small intensity gradients do not get segmented, thus adversely affecting object recognition. These limitations of specific segmentation algorithms aside, the use of regions as primitives well serves the objectives of object recognition.

To obtain good segmentation results, we use a multiscale segmentation algorithm presented in [3, 7, 53]. It partitions an image into homogeneous regions of a priori unknown shape, size, gray-level contrast, and topological context. A region is considered to be homogeneous if variations in intensity within the region are smaller than intensity change across its boundary, regardless of its absolute degree of variability. Image segmentation is performed at a range of homogeneity values, i.e., intensity contrasts. As the intensity contrast increases, regions with smaller contrasts strictly merge. A sweep of the contrast values thus results in the extraction of all the segments present in the image.

3 Segmentation Tree and Region Descriptors

After segmenting an image, the resulting regions and their spatial and structural relationships can be used for recognition. A number of approaches do not exploit region relationships, but account for region intrinsic properties, and treats the regions as a bag of visual words [28, 45]. Other methods additionally account for pairwise region relations [27], and the contextual information provided by larger ancestor regions within which smaller regions are embedded [33]. Our work [6, 55, 56], along with several other methods [25], generalizes previous approaches by additionally accounting for the spatial layout and recursive embedding of regions in a segmentation tree.

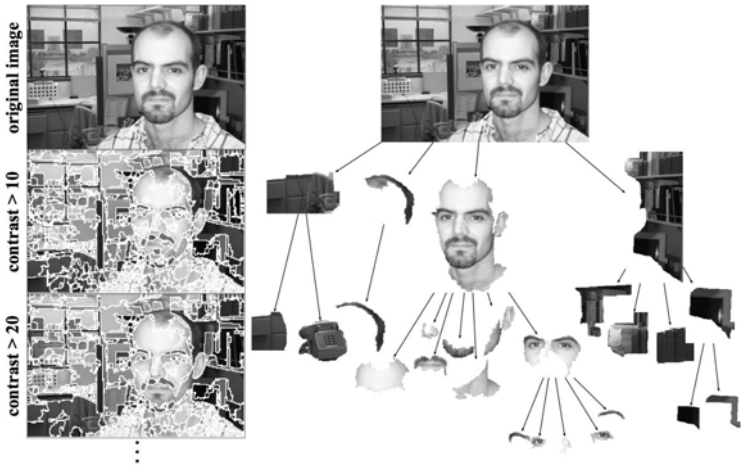


Fig. 1. Segmentation trees of sample Caltech-101 images [20]: (left) segmentations obtained for two sample intensity contrast values from the exhaustive range [1,255]; (right) sample nodes of the corresponding segmentation tree, where the root represents the whole image, nodes closer to the root represent large regions, while their children nodes capture smaller embedded details. The number of nodes (typically 50–100), branching factor (typically 0–10), and the number of levels (typically 7–10) in different parts of the segmentation tree are image dependent, and automatically determined.

In the segmentation tree, the root represents the whole image, nodes closer to the root represent large regions, while their children nodes capture smaller embedded details, as depicted in Fig. 1. The tree in general may not have regular structure (e.g., quad-tree). For example, the multiscale segmentation of [3, 7, 53] gives the number of nodes (typically 50–100), branching factor (typically 0–10), and the number of levels (typically 7–10) that are image dependent in different parts of the tree. Thus, the segmentation tree is a rich image representation that is capable of capturing object properties (a)–(d), mentioned in Sec. 1.

The segmentation tree (ST), however, cannot distinguish among many different ways in which the same set of subregions may be spatially distributed within the parent region. This may give rise to significantly different visual appearances, while the region-embedding properties remain the same. Consequently, STs for many visually distinct objects are identical. The ST can be extended by including the information about 2D spatial adjacency among the regions – while retaining the information about their recursive embedding. This new model augments ST with region adjacency graphs, one for the children of each ST node. A neighbor edge is added between two sibling nodes in ST if the corresponding two regions are neighbors in the image. This transforms ST into a graph, consisting of two distinct sets of edges – one representing the original, parent-child hierarchy, and the other, consisting of lateral links, representing the newly added neighbor relationships (Fig. 2). The neighbor relationships between any nonsibling nodes in CST can be easily retrieved by examining the neighbor relations of their ancestor nodes. To highlight the presence of the complementary, neighbor information modifying the segmentation tree, the new representation is referred to as connected

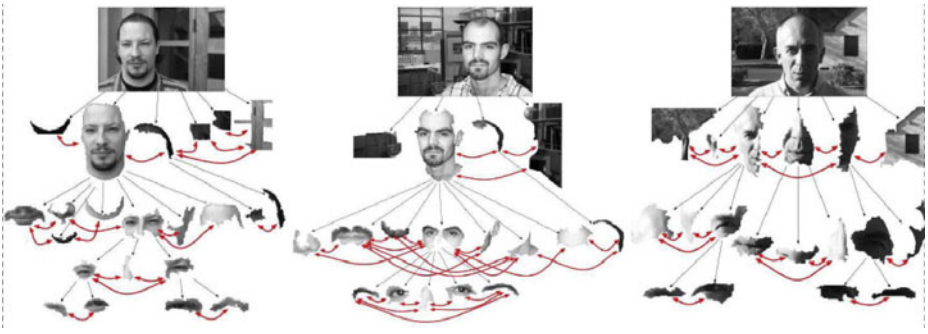


Fig. 2. Example Connected Segmentation Trees (CSTs): Lateral edges (red) that link neighboring image regions are added to the corresponding segmentation trees (black) of the images. CSTs reduce ambiguity about the region layout.

segmentation tree (CST), even though it is strictly a graph. Both nodes and edges of CST have attributes, i.e., they are weighted, where the node (edge) weight is defined in terms of properties of the corresponding region (spatial relationship between regions). Thus, CST generalizes ST to represent images as a hierarchy of region adjacency graphs. As multiscale regions may be viewed as a basic vocabulary of object categories, the CST may be seen as a basis for defining general purpose image syntax, which can serve as an intermediate stage to isolate and simplify inference of image semantics. In the following, we will interchangeably use CST and region hierarchy to denote the same image representation—namely, the hierarchical graph representation that captures recursive embedding of regions, as well as region layout at all levels.

Each node v in the region hierarchy can be characterized by a vector of properties of the corresponding region, denoted as $\boldsymbol{\psi}_v$. In our previous work, we use intrinsic photometric and geometric properties of the region, as well as relative inter-region properties describing the spatial layout of the region and its neighbors. In this way, $\boldsymbol{\psi}_v$ encodes the spatial layout of regions, while the CST structure itself captures their recursive containment. The properties are defined to allow scale and rotation-in-plane recognition invariance. In particular, elements of $\boldsymbol{\psi}_v$ are defined relative to the corresponding properties of v's parent-node u, and thus ultimately relative to the entire image.

Let w, v, and u denote regions forming a child-parent-grandparent triple. Then, the properties of each region v we use are as follows: (1) normalized gray-level contrast g_v, defined as a function of the mean region intensity G, $g_v \triangleq \frac{|G_u - G_v|}{|G_v - G_w|}$; (2) normalized area $a_v \triangleq A_v/A_u$, where A_v and A_u are the areas of v and u; (3) area dispersion AD_v of v over its children $w \in C(v)$, $\text{AD}_v \triangleq \frac{1}{|C(v)|}\sum_{w\in C(v)}(a_w - \overline{a_{C(v)}})^2$, where $\overline{a_{C(v)}}$ is the mean of the normalized areas of v's children; (4) the first central moment μ_v^{11}; (5) squared perimeter over area $\text{PA}_v \triangleq \frac{\text{perimeter}(v)^2}{A_v}$; (6) angle γ_v between the principal axes of v and u; the principal axis of a region is estimated as the eigenvector of matrix $\frac{1}{\mu^{00}}\begin{bmatrix}\mu^{20} & \mu^{11}\\ \mu^{11} & \mu^{02}\end{bmatrix}$ associated with the larger eigenvalue, where the μ's are the standard central moments; (7) normalized displacement $\overrightarrow{\Delta}_v \triangleq \frac{1}{\sqrt{A_u}}\overrightarrow{d}_v$, where $|\overrightarrow{d}_v|$ is the

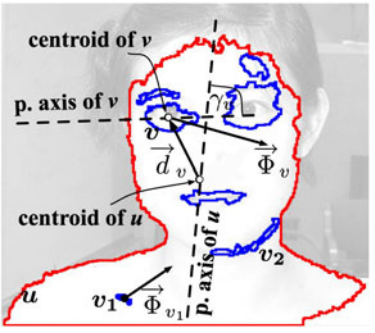


Fig. 3. Fig. 3. Properties of a region associated with the corresponding node in the segmentation tree: Region u (marked red) contains a number of embedded regions $v, v_1, v_2, \ldots$ (marked blue). The principal axes of u and v subtend angle γ_v, the displacement vector $\boldsymbol{d}_v$ connects the centroids of u and v, while the context vector $\boldsymbol{F}_v$ records the general direction in which the siblings $v_1, v_2, \ldots$ of v are spatially distributed.

distance between the centroids of u and v, and $\measuredangle \vec{d}_v$ is measured relative to the principle axis of parent node u, as illustrated in Fig. 3; $\sqrt{A_u}$ represents an estimate of the diameter of parent region u; and (8) context vector $\vec{F}_v \triangleq \sum_{s \in S(v)} \frac{A_s}{|\vec{d}_{vs}|^3} \vec{d}_{vs}$, where $S(v)$ is the set of v's sibling regions s, and $|\vec{d}_{vw}|$ is the distance between the centroids of v and s, and $\measuredangle \vec{d}_{vs}$ is measured relative to the principle axis of their parent node u; as illustrated in Fig. 3, the context vector records the general direction v sees its sibling regions and disallows matching of scrambled layouts of regions at a specific tree level. In summary, the vector of region properties associated with node v is $\boldsymbol{\psi}_v = [g_v, a_v, \mathrm{AD}_v, \mu_v^{11}, \mathrm{PA}_v, \gamma_v, \vec{\Delta}_v, \vec{F}_v]^{\mathrm{T}}$. Each element of $\boldsymbol{\psi}_v$ is normalized over all multiscale regions of all training images to take a value in the interval $[0, 1]$. This list of useful region properties, can be easily modified to reflect the needs of different applications.

The aforementioned hierarchical region-based image representation will allow recognition performance with the following desirable invariance characteristics with respect to: (i) Translation, in-plane rotation and object-articulation (changes in relative orientations of object parts): because the segmentation tree itself is invariant to these changes; (ii) Scale: because subtree matching is based on relative properties of nodes, not absolute values; (iii) Occlusion in the training set: because subtrees are registered and stitched together within the tree-union encoding the entire (unoccluded) category structure; (iv) Occlusion in the test set: because subtrees corresponding to visible object parts can still be matched with the model; (v) Small appearance changes (e.g. due to noise): because changed regions may still be the best matches; (vi) Region shape deformations (e.g., due to minor depth rotations of objects): because changes in geometric/topological properties of regions (e.g., splits/mergers) are accounted for during matching; and (vii) Clutter: because clutter regions, being non-category subimages, are not repetitive and therefore frequent.

Any object occurrences in images will correspond to subgraphs within the corresponding CSTs. The goal of learning is to identify these subgraphs and capture their canonical node and node-connectivity properties. The goal of inference is to use this graph model to identify, within the CST of a new image, subgraphs that represent instances of the learned class. In the following two sections, we explain object learning and recognition using the region hierarchy.

4 Learning Object Properties

This section argues that hierarchical region-based representations of images possess two major features—namely, that they: (a) facilitate learning under various degrees of supervision, and (b) relax the requirements for complex object models and classifiers.

4.1 Object Discovery as Graph Matching

To communicate the natural variations of objects to a recognition algorithm, typically, a set of training images has to be manually annotated. Supervision in training may involve the following: manually segmented object instances in training images, bounding boxes placed around the objects, or only object labels associated with the entire images. In case the bounding boxes are available in training, they immediately provide access to similar subgraphs of region hierarchies corresponding to instances of the target object class. If the bounding boxes are not available, the object occurrences can be discovered by matching the region hierarchies of images from the same class, and thus identifying their similar subgraphs. Below, we explain how to match CSTs, and thus obtain a set of their similar subgraphs, which will be used then to learn the object model or classifier.

Two images may have a number of similar regions, which may confuse the matching algorithm. However, if similar regions also have similar nesting and layout properties, then it is very likely that they represent meaningful image parts, e.g., instances of the same object class, which indeed should be matched. Our algorithm achieves robustness by pairing regions whose photometric, geometric, and structural properties match, and the same holds for their neighbors, and these two conditions recursively hold for their embedded subregions. Such region matching can be formalized using the graph matching techniques. In the following, we first briefly review graph-based image matching methods, and then present our approach.

Image matching using graph image representations may be performed by: (a) exploiting spectral properties of the graphs' adjacency matrices [44, 50, 51]; (b) minimizing the graph edit-distance [12, 47, 62]; (c) finding a maximum clique of the association graph [41]; (d) using energy minimization or expectation-maximization of a statistical model [23, 63]. All these formulations can be cast as a quadratic assignment problem, where a linear term in the objective function encodes node compatibility functions, and a quadratic term encodes edge compatibility functions. Therefore, approaches to graph matching mainly focus on: (i) finding suitable definitions of the compatibility functions; and (ii) developing efficient algorithms for approximately solving the quadratic assignment problem (since it is NP-hard), including a suitable reformulation of the quadratic into linear assignment problem. However, most popular approximation algorithms (e.g.,

relaxation labeling, and loopy belief propagation) critically depend on a good initialization and may be easily trapped in a local minimum, while some (e.g., deterministic annealing schemes) can be used only for graphs with a small number of nodes. Graduated nonconvexity schemes [26], and successive convexification methods [30] have been used to convexify the objective function of graph matching, and thus alleviate these problems. In our work, we use the replicator dynamics algorithm to solve the underlying convex problem, as explained in the sequel.

Let $H = (V, E, \psi, \phi)$ denote the region hierarchy, where $V = \{v\}$ and $E = \{(v, u)\} \subseteq V \times V$ are the sets of nodes and edges, and ψ and ϕ are functions that assign attributes to nodes, $\psi : V \rightarrow [0,1]^d$, and to edges, $\phi : E \rightarrow [0,1]$. Given two shapes, H and H', the goal of the matching algorithm is to find a subgraph isomorphism, $f: U \rightarrow U'$, where $U \subseteq V$ and $U' \subseteq V'$, which minimizes the cost, C, defined as

$$C = \min_{f} \left[\beta \sum_{(v,v') \in f} a_{vv'} + (1-\beta) \sum_{(v,v',u,u') \in f \times f} b_{vv'uu'} \right], \tag{1}$$

where the a's are non-negative costs of matching nodes v and $v' = f(v)$, and the b's are non-negative costs of matching edges $(v, u) \in E$ and $(v', u') \in E'$, and $\beta \in [0, 1]$ weights their relative significance to matching.

To minimize C, we introduce a confidence vector, X, indexed by all node pairs $(v, v') \in V \times V'$, whose each element $x_{vv'} \in [0, 1]$ encodes the confidence that node pair (v, v') should be matched. Matching can then be reformulated as estimating X so that C is minimized. That is, we relax the discrete problem of (1) to obtain the following quadratic program (QP):

$$\begin{array}{ll} \min\limits_{X} & \left[\beta A^{\mathrm{T}} X + (1-\beta) X^{\mathrm{T}} B X\right], \\ \text{s.t.} & \forall (v, v') \in V \times V', \quad x_{vv'} \geq 0, \\ & \forall v' \in V', \ \sum_{v \in V} x_{vv'} = 1, \\ & \forall v \in V, \ \sum_{v' \in V'} x_{vv'} = 1, \end{array} \tag{2}$$

where A is a vector of costs $a_{vv'}$, and B is a matrix of costs $b_{vv'uu'}$. We define $a_{vv'} = \|\psi(v) - \psi(v')\|_2$. Also, we define $b_{vv'uu'}$ so that matching edges of different types–namely, hierarchical and neighbor edges—is prohibited, and matches between edges of the same type with similar weights are favored in (2): $b_{vv'uu'} = \infty$ if edges (v, u) and (v', u') are not of the same type; and $b_{vv'uu'} = |\phi(v, v') - \phi(u, u')|$ if edges (v, u) and (v', u') are of the same type. Both the a's and b's are normalized to [0,1].

To satisfy the isomorphism constraints of matching, the algorithm matches regions with regions, and separately region relationships with corresponding relationships, while preserving the original node connectivity of H and H'. The constraints in (2) are typically too restrictive, because H and H' may have relatively large structural differences in terms of the number of nodes and their connectivity, even if H and H' represent two objects from the same class. These structural differences may, e.g., arise from different outputs of the segmentation algorithm on images of the same object class but captured under varying illumination. In this case, splitting or merging regions along their shared, low-contrast boundary may occur which affects the structure of H and H'. Therefore, a more general many-to-many matching formulation would be more appropriate for

our purposes. The literature reports a number of heuristic approaches to many-to-many matching [19, 42, 58], which however are developed only for weighted graphs, and thus cannot be used for our region hierarchies that have attributes on both nodes and edges. To relax the constraints in (2), we first match H to H', which yields solution X_1. Then, we match H' to H, which yields solution X_2. The final solution, $\tilde{X}$, is estimated as an intersection of non-zero elements of X_1 and X_2. Formally, the constraints in (2) are relaxed as follows: (i) $\forall(v, v') \in V \times V', x_{vv'} \geq 0$; and (ii) $\forall v \in V, \sum_{v' \in V'} x_{vv'} = 1$ when matching H to H'; and $\forall v' \in V', \sum_{v \in V} x_{vv'} = 1$ when matching H' to H. Thus, by using an auxiliary matrix $W = \beta \text{diag}(A) + (1 - \beta)B$, we reformulate (2) and arrive at the following one-to-many matching problem

$$\begin{array}{ll} \min\limits_{X} & X^{\mathrm{T}} W X, \\ \text{s.t.} & \begin{array}{ll} \forall (v, v') \in V \times V', & x_{vv'} \geq 0, \\ \forall v' \in V', & \sum_{v \in V} x_{vv'} = 1, \end{array} \end{array} \tag{3}$$

which can be efficiently solved by using the replicator dynamics update rule [40]:

$$X \leftarrow \frac{WX}{X^{\mathrm{T}} W X}. \tag{4}$$

The proof that the optimization of (3) results in the subgraph isomorphism follows from the well-known Motzkin-Strauss theorem, as shown in [40, 41].

Complexity of our matching is $O((|V|+|E|)^2)$. Our implementation in C takes about 1min on a 2.8GHz, 2GB RAM PC for two CSTs with approximately 50 nodes.

The matching subgraphs may represent complete object occurrences or their parts (e.g., due to partial occlusion, or changes in illumination, viewpoint, or scale variations across the images). Therefore, the extracted similar subgraphs provide for many observations of entire objects or their parts in the class. This allows robust estimation of the region-based object model. Note that as a result of matching region hierarchies, we immediately have access to correspondences between nodes and edges of all extracted subgraphs. These correspondences can be used to learn a canonical graph of the object class that subsumes all extracted instances, and thus represents the object model.

4.2 Region-Based Object Model

The region-based object model is aimed at capturing how image regions are recursively laid out to comprise an object, and what their geometric and photometric properties are. From a set of given or extracted similar CSTs, as explained in the previous section, our goal is to obtain a compact, canonical model of the target class. In our work we formulate this canonical graph as graph-union.

Graph-unions are well studied graph structures, the detailed treatment of which can be found, for example, in [13–15, 29, 60, 61]. The graph-union $\mathcal{T}$ is the smallest graph, which contains every graph from a given set $\mathbb{D}$. Ideally, $\mathcal{T}$ should be constructed by first finding the maximum common subgraph of $\mathbb{D}$, and then by adding to the common subgraph, and appropriately connecting, the remaining nodes from $\mathbb{D}$. However, finding this maximum common subgraph would entail prohibitive complexity if D is large. Therefore, we resort to a suboptimal sequential approach. In each iteration $\mathcal{T}$ is

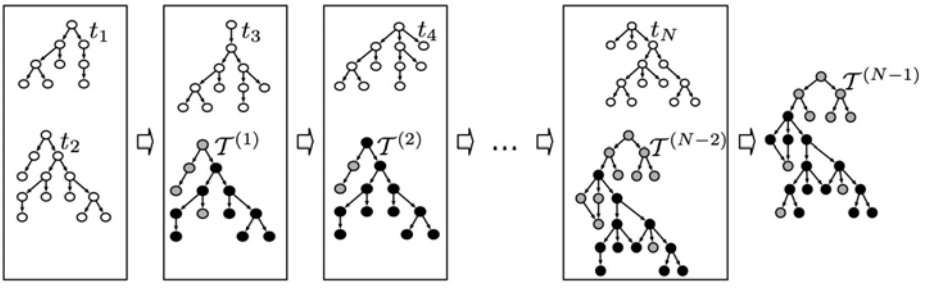


Fig. 4. Construction of graph-union $\mathcal{T}$ from the extracted set of similar CSTs $\mathbb{D}=\{t_1, t_2, \ldots, t_N\}$: In each iteration, a selected CST t from $\mathbb{D}$ is first matched against the current estimate $\mathcal{T}^{(n)}$, which yields their maximum common subgraph τ (marked black). Then the unmatched nodes from t are added and appropriately connected (marked gray), to form $\mathcal{T}^{(n+1)}$. The result is the graph-union.

extended by adding a new CST t from $\mathbb{D}$ until every CST from $\mathbb{D}$ has been added to the graph-union, as illustrated in Fig. 4. As can be seen, the selected t is first matched against the current estimate $\mathcal{T}^{(n)}$, which results in their common subtree τ, and then the unmatched nodes from t are added and appropriately connected to τ in order to form $\mathcal{T}^{(n+1)}$. For matching t and $\mathcal{T}^{(n)}$, we use the same algorithm presented in Sec. 4.1. After adding the unmatched nodes, the result is the graph-union, which preserves the node connectivity from $\mathbb{D}$.

5 Results

Region hierarchies, as our image representations, allow joint object detection, recognition and segmentation. This can be achieved by matching the learned graph-union model, presented in the previous section, with the region hierarchy of a new image. In our approach, the matching subgraphs whose similarity measure is larger than a specified threshold are taken as detected objects. This detection simultaneously delineates object boundaries, due to using regions as basic image features. This section reviews the empirical validation of our approach, presented in [6]. The experiments demonstrate advantages of using region-based image representations and object modeling for recognition versus alternative approaches.

We consider 14 categories from four datasets: 435 faces, 800 motorbikes, 800 airplanes, 526 cars (rear) from Caltech-101 [20]; 328 Weizmann horses [9]; 1554 images queried from LabelMe [46] to contain cars, trees, and buildings together; and 200 images with 715 occurrences of cows, horses, sheep, goats, camels, and deer from UIUC Hoofed Animals dataset [6]. Caltech-101 images contain only a single, prominently featured object from the category, except for images of cars (rear) containing multiple, partially occluded cars appearing at different scales, with low contrast against textured background. The Weizmann dataset contains sideviews of walking/galloping horses of different breeds, colors and textures, with different object articulations in their natural (cluttered) habitat. LabelMe is a more difficult collection of real-world images which

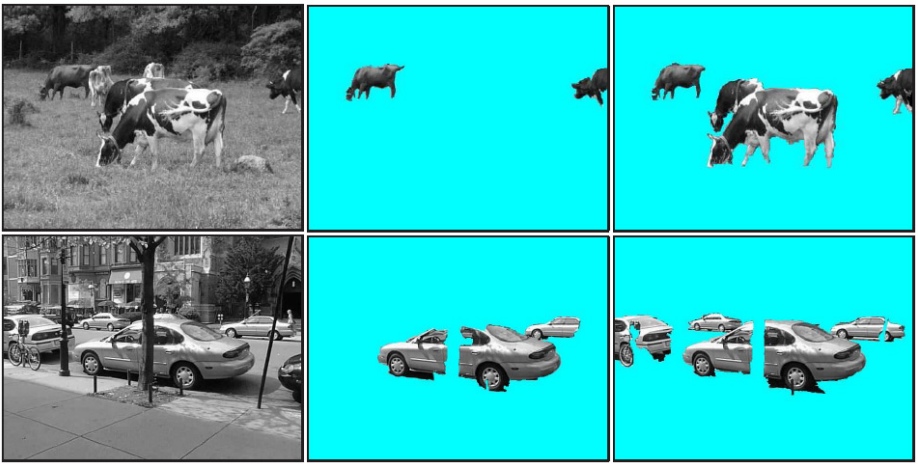
contain many other object categories along with the queried ones, captured under different lighting conditions, and at varying scales. The Hoofed Animals dataset presents the mentioned challenges, and has higher complexity as it contains multiple instances of multiple very similar animal categories per image, requiring high inter-category resolvability.

The Caltech-101 and Weizmann categories are learned one category at a time on the training set that consists of M_p randomly selected examples showing the category, and $M_n \geq 0$ images from the background category in Caltech-101 ($M=M_p+M_n$). The LabelMe and Hoofed Animals categories are all learned together by randomly selecting M images from the corresponding dataset. To recognize and segment any category occurrences in a test image, the learned category model is matched with CST of the image. The matched subtrees (i.e., detections) whose similarity measure is larger than a threshold are adjudged as detected objects. Results shown in tables and figures are obtained for the threshold that yields equal error rate. We use the following definitions of detection (DE), and segmentation (SE) errors. Let D denote the area that a detection covers in the test image, and G denote the ground-truth object area. Then, $\text{DE} \triangleq \frac{D \cap G}{D \cup G}$, and $\text{SE} \triangleq \frac{\text{XOR}(D,G)}{D \cup G}$. A detection is a false positive if $\text{DE} < 0.5$, otherwise it is a true positive (TP). Recognition is evaluated only on TP's by visual inspection.

5.1 Qualitative Evaluation – Segmentation

Figs. 5–6 demonstrate high accuracy of simultaneous object detection and segmentation in images from LabelMe and Hoofed Animals datasets, using $M=50$ training images. Each TP in the figures is correctly recognized. CSTs outperform STs in both object detection and segmentation, especially in cases of partial occlusion (e.g., cars and cows in Fig. 6), and for objects defined rather as a region spatial layout than containment (e.g., spotted cows in Fig. 6). In these cases, modeling of the region adjacency by CSTs proves advantageous. Segmentation is good even in cases when object boundaries are jagged and blurred (e.g., trees in Fig. 5), and when objects from the same category occlude each other, forming a complex region topology with low-intensity contrasts

Fig. 5. Samples from Hoofed Animals (left) and LabelMe (right). Segmentation results of CST are overlaid on the original. Different colors denote recognized categories. CST successfully resolves small differences between the categories sheep and goats.



(a) original image (b) STs (c) CSTs

Fig. 6. CSTs outperform STs in both detection and segmentation on samples from Hoofed Animals (top) and LabelMe (bottom). Undetected image parts are masked out.

(e.g., cars in Fig. 5). Objects that are not detected, for the most part, have low intensity contrasts with the surround, and thus do not form category-characteristic subgraphs within CSTs that can be matched with the category model.

5.2 Qualitative Evaluation – Model

Fig. 7 illustrates the model $\mathcal{G}$ obtained for the category horses, learned on six, randomly selected images $\mathbb{D}$ from the Weizmann dataset. Nodes v in $\mathcal{G}$, depicted as rectangles, contain regions from $\mathbb{D}$ that got matched with v during learning. As can be seen, the structure of $\mathcal{G}$ correctly captures the recursive containment and neighbor relations of regions occupied by the horses in $\mathbb{D}$. For example, nodes *head*, *neck*, and *mane* are found to be children of node *head&neck*, and they are all identified as neighbors. Also, it is correct that *head&neck* and *tail* are not neighbors. Similar background regions that co-occur with horses in $\mathbb{D}$ may also be included in the model (e.g., nodes corresponding to *fence*). Typically, the percentage of background nodes out of the total number of model nodes is small (3-5%).

5.3 Quantitative Evaluation

Fig. 8 (left) presents the recall-precision curves (RPC) of detection for the Caltech-101 categories using CSTs and STs. Detection performance in the presence of occlusion is tested by masking out a randomly selected rectangular area in the image, and replacing this area with a patch from the background category of Caltech-101. CST increases the area under the RPC of ST by $6.5 \pm 0.3\%$, and by $3.1 \pm 0.2\%$ in the presence of the occluding patch covering 20% of the image. Invariance to in-plane rotation is tested by randomly rotating test images. Performance on these rotated images is the

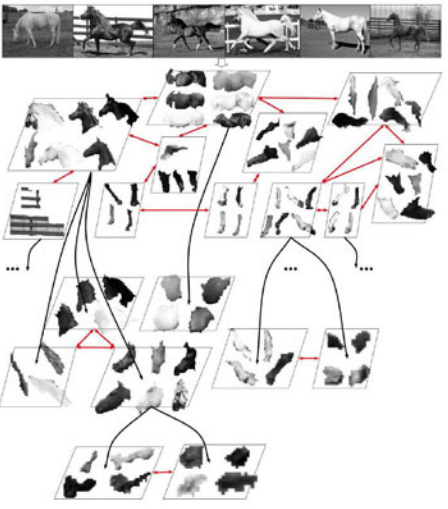


Fig. 7. CST-based model of Weizmann horses learned on the input images shown in the top row

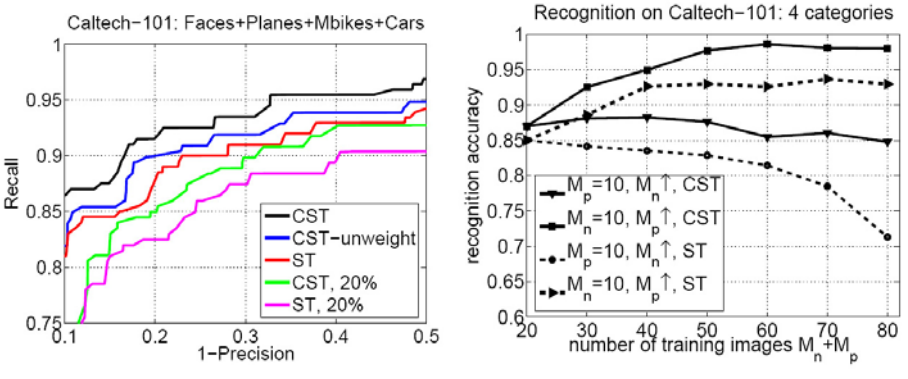


Fig. 8. (left) Detection recall-precision curves: "CST-unweight" means that edges in CST are not weighted. 20% is the size of a rectangular occlusion w.r.t. the image size. M_p=10, M_n=10. ST is the method of [54]. **(right)** Recognition accuracy of CST and ST for the varying ratio of M_p and M_n in the training set.

same as the one presented in Fig. 8. Measuring the strength of neighborliness using the generalized Voronoi diagram improves performance over the case when the weights of links in CST are set to take only values 1 or 0, referred to as CST-unweight. CST increases the area under the RPC of CST-unweight by $2.3 \pm 0.3\%$. Fig. 8 (right) shows recognition accuracy of CST and ST. A small increase in M_n does not downgrade the accuracy. As M_n becomes larger, objects belonging to other categories start appearing more frequently, and thus get learned, making the training set inappropriate. Increasing M_p yields smaller recognition error. CST outperforms ST in recognition, and longer maintains high accuracy with the increase of M_n. In general, the number of nodes in the

Table 1. Detection recall, segmentation and recognition errors (in %) on LabelMe and Weizmann Horses datasets, using the same number of training and test images as in [17, 45, 66]

	LabelMe Trees	LabelMe Buildings	LabelMe Cars	Weizmann Horses
Recall	47.6±6.9	92.6±6.9	67.6±6.9	91.9±5.2
Seg. error	41.6±7.9	34.6±13.4	32.5±8.2	7.2±2.5
Rec. error	19.7±3.8	11.6±2.9	12.9±4.8	7.9±4.1

Table 2. Detection recall, segmentation and recognition errors (in %) on UIUC Hoofed Animals dataset, using the same number of training and test images as in [17, 45, 66]

	Horses	Cows	Deer	Sheep	Goats	Camels
Recall	81.2±10.3	78.4±4.2	88.1±6.9	81.2±5.3	78.2±8.6	89.9±7.2
Seg. error	15.9±5.3	17.1±4.6	11.1±8.4	24.8±7.2	20.1±8.1	11.5±5.1
Rec. error	7.8±4.2	6.5±6.2	7.7±3.4	7.8±4.1	12.2±5.4	3.2±3.9

model quickly reaches saturation as new positive examples are added to the training set, and continues to very slowly increase, in part, due to chance repetitions of background regions.

Table 1 and Table 2 summarize detection recall, and segmentation and recognition errors obtained for the equal error rates on LabelMe, Weizmann, and Hoofed Animals datasets. For Hoofed Animals, CST outperforms ST in detection recall by 7.5%, segmentation by 10.7%, and recognition by 8.6%. For comparison, we obtained SE=6.5% on a relatively simple UIUC (multiscale) car dataset, using the same set-up as in [22], while their result is SE=7.9%. The other hierarchical approaches cited here use non-benchmark datasets, or report a single retrieval result for the entire Caltech-101, beyond the focus of this paper. Non-hierarchical approaches that model objects using image segments obtained at only one pre-selected scale, report the following state-of-the-art results: [45] – SE=47% for buildings, and SE=79% for cars of LabelMe; [66] – SE=7% for Weizmann horses; and [17] – SE=18.2% for Weizmann horses. In comparison with these approaches, Table 1 indicates that the CSTs yield better, or, in only a few cases, very similar performance. Regarding recognition accuracy, Fig. 8 shows that we outperform by $1.8 \pm 0.3\%$ the recognition rate of 94.6% of [17] on the four Caltech-101 categories. Other approaches cited here use a different, less demanding recognition evaluation based on classifying either the entire images or bounding boxes around objects.

The results demonstrate that our approach is invariant with respect to: (i) translation, in-plane rotation and object articulation, since CST itself is invariant to these changes; (ii) certain degree of scale changes, since matching is based on relative properties of regions; (iii) occlusion in the training and test sets, since graph-union registers the entire (unoccluded) category structure from partial views of occurrences in the training set, while subgraphs of visible object parts in the CST of a test image can still be matched with the model; (iv) minor depth rotations of objects causing their shape deformations, because structural instability of CSTs (e.g., due to region splits/mergers) is accounted for during matching; and (v) clutter, since clutter regions are not frequent and thus not learned.

6 Conclusions

We have argued in this paper that using multiscale regions as basic image features: (a) Facilitates capturing photometric, geometric, and structural properties of objects; (b) Allows simultaneous object discovery, recognition and segmentation; and (c) Enables efficient and robust learning and inference of region-based object representations. We have reviewed our region-based object recognition framework developed over the last several years. While the framework is capable of extracting a taxonomy of object categories from an arbitrary image set, and segmenting textures into texels, we have focused here on a compact subset of these problems. We have considered the related problems of single category discovery, detection, and segmentation. We have discussed how this set of problems poses many recognition related challenges, which are inadequately addressed by existing methods that use point and edge features. The summary of our experimental results that we have presented here shows that use of regions offers a number of advantages for object recognition over point and edge features.

References

1. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. IEEE TPAMI 26(11), 1475–1490 (2004)
2. Ahmadyfard, A.R., Kittler, J.V.: Using relaxation technique for region-based object recognition. Image and Vision Computing 20(11), 769–781 (2002)
3. Ahuja, N.: A transform for multiscale image segmentation by integrated edge and region detection. IEEE TPAMI 18(12), 1211–1235 (1996)
4. Ahuja, N., Todorovic, S.: Extracting texels in 2.1D natural textures. In: ICCV (2007)
5. Ahuja, N., Todorovic, S.: Learning the taxonomy and models of categories present in arbitrary images. In: ICCV (2007)
6. Ahuja, N., Todorovic, S.: Connected segmentation tree – a joint representation of region layout and hierarchy. In: CVPR (2008)
7. Arora, H., Ahuja, N.: Analysis of ramp discontinuity model for multiscale image segmentation. In: ICPR, vol. 4, pp. 99–103 (2006)
8. Basri, R., Jacobs, D.: Recognition using region correspondences. IJCV 25(2), 145–166 (1997)
9. Borenstein, E., Ullman, S.: Class-specific, top-down segmentation. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 109–124. Springer, Heidelberg (2002)
10. Bouman, C.A., Shapiro, M.: A multiscale random field model for Bayesian image segmentation. IEEE Trans. Image Processing 3(2), 162–177 (1994)
11. Brice, C.R., Fennema, C.L.: Scene analysis using regions. Artificial Intelligence 1, 205–226 (1970)
12. Bunke, H., Allermann, G.: Inexact graph matching for structural pattern recognition. Pattern Rec. Letters 1(4), 245–253 (1983)
13. Bunke, H., Foggia, P., Guidobaldi, C., Vento, M.: Graph clustering using the weighted minimum common supergraph. In: Hancock, E.R., Vento, M. (eds.) GbRPR 2003. LNCS, vol. 2726, pp. 235–246. Springer, Heidelberg (2003)
14. Bunke, H., Jiang, X., Kandel, A.: On the minimum common supergraph of two graphs. Computing 65(1), 13–25 (2000)

15. Bunke, H., Kandel, A.: Mean and maximum common subgraph of two graphs. Pattern Rec. Letters 21(2), 163–168 (2000)
16. Canny, J.: A computational approach to edge detection. IEEE TPAMI 8(6), 679–698 (1986)
17. Cao, L., Fei-Fei, L.: Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In: ICCV (2007)
18. Darwish, A.M., Jain, A.K.: A rule based approach for visual pattern inspection. IEEE Trans. Pattern Analysis Machine Intelligence 10(1), 56–68 (1988)
19. Demirci, M.F., Shokoufandeh, A., Keselman, Y., Bretzner, L., Dickinson, S.J.: Object recognition as many-to-many feature matching. Int. J. Computer Vision 69(2), 203–222 (2006)
20. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE TPAMI 28(4), 594–611 (2006)
21. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR, vol. 2, pp. 264–271 (2003)
22. Fidler, S., Leonardis, A.: Towards scalable representations of object categories: Learning a hierarchy of parts. In: CVPR (2007)
23. Finch, A.M., Wilson, R.C., Hancock, E.R.: An energy function and continuous edit process for graph matching. Neural Computation 10(7) (1998)
24. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE TPAMI 6(6) (1984)
25. Glantz, R., Pelillo, M., Kropatsch, W.G.: Matching segmentation hierarchies. Int. J. Pattern Rec. Artificial Intelligence 18(3), 397–424 (2004)
26. Gold, S., Rangarajan, A.: A graduated assignment algorithm for graph matching. IEEE TPAMI 18(4), 377–388 (1996)
27. Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: ICCV (2009)
28. Gu, C., Lim, J.J., Arbelaez, P., Malik, J.: Recognition using regions. In: CVPR (2009)
29. Gupta, A., Nishimura, N.: Finding largest subtrees and smallest supertrees. Algorithmica 21(2), 183–210 (1998)
30. Jiang, H., Drew, M.S., Li, Z.N.: Matching by linear programming and successive convexification. IEEE TPAMI 29(6), 959–975 (2007)
31. Kittler, J., Hancock, E.R.: Contextual decision rule for region analysis. Image Vision Comput. 5(2), 145–153 (1987)
32. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: Workshop on Statistical Learning in Computer Vision, ECCV, pp. 17–32 (2004)
33. Lim, J.J., Arbelaez, P., Gu, C., Malik, J.: Context by region ancestry. In: ICCV (2009)
34. Lindeberg, T.: Scale-Space Theory in Computer Vision. Kluwer Academic Publishers, Norwell (1994)
35. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
36. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. PAMI 26, 530–549 (2004)
37. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. IJCV 65(1/2), 43–72 (2005)
38. Mumford, D., Shah, J.: Boundary detection by minimizing functionals. In: CVPR, pp. 22–26 (1985)
39. Opelt, A., Pinz, A., Zisserman, A.: Incremental learning of object detectors using a visual shape alphabet. In: CVPR, vol. 1, pp. 3–10 (2006)
40. Pelillo, M.: Matching free trees, maximal cliques, and monotone game dynamics. IEEE TPAMI 24(11), 1535–1541 (2002)

41. Pelillo, M., Siddiqi, K., Zucker, S.W.: Matching hierarchical structures using association graphs. IEEE TPAMI 21(11), 1105–1120 (1999)
42. Pelillo, M., Siddiqi, K., Zucker, S.W.: Many-to-many matching of attributed trees using association graphs and game dynamics. In: Arcelli, C., Cordella, L.P., Sanniti di Baja, G. (eds.) IWVF 2001. LNCS, vol. 2059, pp. 583–593. Springer, Heidelberg (2001)
43. Peng, J., Bhanu, B.: Closed-loop object recognition using reinforcement learning. IEEE Trans. Pattern Analysis Machine Intelligence 20(2), 139–154 (1998)
44. Qiu, H., Hancock, E.R.: Graph matching and clustering using spectral partitions. Pattern Recognition 39(1), 22–34 (2006)
45. Russell, B.C., Efros, A.A., Sivic, J., Freeman, W.T., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: CVPR, vol. 2, pp. 1605–1614 (2006)
46. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. IJCV 77(1-3), 157–173 (2008)
47. Sebastian, T.B., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing their shock graphs. IEEE Trans. Pattern Anal. Machine Intell. 26(5), 550–571 (2004)
48. Serre, T., Wolf, L., Poggio, T.: Object recognition with features inspired by visual cortex. In: CVPR, vol. 2, pp. 994–1000 (2005)
49. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE TPAMI 22(8), 888–905 (2000)
50. Shokoufandeh, A., Macrini, D., Dickinson, S., Siddiqi, K., Zucker, S.W.: Indexing hierarchical structures using graph spectra. IEEE TPAMI 27(7), 1125–1140 (2005)
51. Siddiqi, K., Shokoufandeh, A., Dickinson, S.J., Zucker, S.W.: Shock graphs and shape matching. IJCV 35(1), 13–32 (1999)
52. Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Learning hierarchical models of scenes, objects, and parts. In: ICCV, vol. 2, pp. 1331–1338 (2005)
53. Tabb, M., Ahuja, N.: Multiscale image segmentation by integrated edge and region detection. IEEE Trans. Image Processing 6(5), 642–655 (1997)
54. Todorovic, S., Ahuja, N.: Extracting subimages of an unknown category from a set of images. In: CVPR, vol. 1, pp. 927–934 (2006)
55. Todorovic, S., Ahuja, N.: Learning subcategory relevances to category recognition. In: CVPR (2008)
56. Todorovic, S., Ahuja, N.: Unsupervised category modeling, recognition, and segmentation in images. IEEE TPAMI 30(12), 1–17 (2008)
57. Todorovic, S., Ahuja, N.: Texel-based texture segmentation. In: ICCV (2009)
58. Todorovic, S., Ahuja, N.: Region-based hierarchical image matching. IJCV (to appear)
59. Torralba, A., Murphy, K., Freeman, W.: Sharing features: efficient boosting procedures for multiclass object detection. In: CVPR, vol. 2, pp. 762–769 (2004)
60. Torsello, A., Hancock, E.R.: Matching and embedding through edit-union of trees. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2352, pp. 822–836. Springer, Heidelberg (2002)
61. Torsello, A., Hancock, E.R.: Learning shape-classes using a mixture of tree-unions. IEEE Trans. PAMI 28(6), 954–967 (2006)
62. Torsello, A., Robles-Kelly, A., Hancock, E.R.: Discovering shape classes using tree edit-distance and pairwise clustering. IJCV 72(3), 259–285 (2007)
63. Tu, Z., Yuille, A.: Shape matching and recognition - using generative models and informative features. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3023, pp. 195–209. Springer, Heidelberg (2004)
64. Weiss, I., Ray, M.: Recognizing articulated objects using a region-based invariant transform. IEEE Trans. Pattern Analysis Machine Intelligence 27(10), 1660–1665 (2005)

65. Winn, J., Criminisi, A., Minka, T.: Object categorization by learned universal visual dictionary. In: ICCV, vol. 2, pp. 1800–1807 (2005)
66. Winn, J., Jojic, N.: Locus: learning object classes with unsupervised segmentation. In: ICCV, pp. 756–763 (2005)
67. Worthington, P.L., Hancock, E.R.: Object recognition using shape-from-shading. IEEE TPAMI 23(5), 535–542 (2001)
68. Worthington, P.L., Hancock, E.R.: Region-based object recognition using shape-from-shading. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1842, pp. 455–471. Springer, Heidelberg (2000)
69. Zhang, R., Zhang, Z.: Hidden semantic concept discovery in region based image retrieval. In: CVPR, vol. 2, pp. 996–1001 (2004)