Context Based Object Categorization: A Critical Survey

Carolina Galleguillos[1] Serge Belongie[1,2]

[1] Computer Science and Engineering, University of California, San Diego
[2] Electrical Engineering, California Institute of Technology
{cgallegu,sjb}@cs.ucsd.edu

Abstract. The goal of object categorization is to locate and identify instances of an object category within an image. Recognizing an object in an image is difficult when images present occlusion, poor quality, noise or background clutter, and this task becomes even more challenging when many objects are present in the same scene. Several models for object categorization use appearance and context information from objects to improve recognition accuracy. Appearance information, based on visual cues, can successfully identify object classes up to a certain extent. Context information, based on the interaction among objects in the scene or on global scene statistics, can help successfully disambiguate appearance inputs in recognition tasks. In this work we review different approaches of using contextual information in the field of object categorization and discuss scalability, optimizations and possible future approaches.

1 Introduction

Traditional approaches to object categorization use appearance features as the main source of information for recognizing object classes in real world images. Appearance features, such as color, edge responses, texture and shape cues, can capture variability in objects classes up to certain extent. In face of clutter, noise and variation in pose and illumination, object appearance can be disambiguated by the coherent composition of objects that real world scenes often exhibit. An example of this situation is presented in Figure 1.

Information about typical configurations of objects in a scene has been studied in psychology and computer vision for years, in order to understand its effects in visual search, localization and recognition performance [2–4, 19, 23]. Biederman *et al.* [4] proposed five different classes of relations between an object and its surroundings, *interposition*, *support*, *probability*, *position* and *familiar size*. These classes characterize the organization of objects in real-world scenes. Classes corresponding to *interposition* and *support* can be coded by reference to physical space. *Probability*, *position* and *size* are defined as *semantic relations* because they require access to the referential meaning of the object. Semantic relations include information about detailed interactions among objects in the scene and they are often used as *contextual features*.

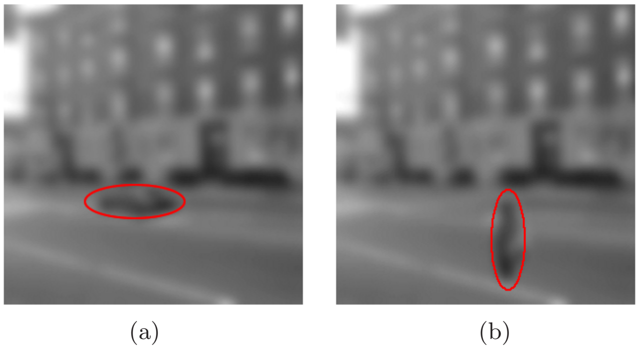


Fig. 1. (a) A car in the street. (b) A pedestrian in the street. The pedestrian is the same patch as the car except for a 90 degrees rotation. The different orientations of both patches within the context defined by the street scene makes the car be perceived as a pedestrian. Example taken from [34].

Several different models [6, 7, 11, 13, 25, 34] in the computer vision community have exploited these semantic relations in order to improve recognition. Semantic relations, also known as context features, can reduce processing time and disambiguate low quality inputs in object recognition tasks. As an example of this idea, consider the flow chart in Figure 2. An input image containing an aeroplane, trees, sky and grass (top left) is first processed through a segmentation-based object recognition engine. The recognizer outputs an ordered shortlist of possible object labels; only the best match is shown for each segment. Without appealing to context, several mistakes are evident. Semantic context (*probability*) in the form of object co-occurrence allows one to correct the label of the aeroplane, but leaves the labels of the sky, grass and plant incorrect. Spatial context (*position*) asserts that sky is more likely to appear above grass than *vice versa*, correcting the labels of the segments. Finally, scale context (*size*) corrects the segment labeled as "plant" assigning the label of tree, since plants are relatively smaller than trees and the rest of the objects in the scene.

In this report, we review a variety of different approaches of context based object categorization models. In Section 2 we assess different types of contextual features used in object categorization: semantic, spatial and scale context. In Section 3 we review the use of context information from a global and local image level. Section 4 presents four different types of local and global contextual interactions: pixel, region, object and object-scene interactions. In Section 5 we consider common machine learning models that integrate context information into object recognition frameworks. Machine learning models such as classifiers and graphical models are discussed in detail. Finally we conclude with the discussion of scalability, optimizations and possible future approaches.

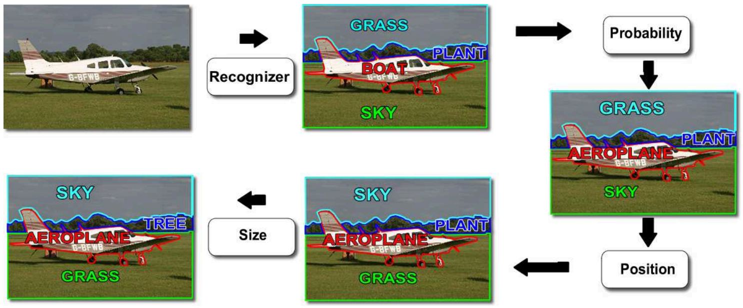


Fig. 2. Illustration of an idealized object categorization system incorporating Biederman's classes: *probability*, *position* and (familiar) *size*. First, the input image is segmented, and each segment is labeled by the recognizer. Next, the different contextual classes are enforced to refine the labeling of the objects leading to the correct recognition of each object in the scene.

2 Types of Context

In the area of computer vision many approaches for object categorization have exploited Biederman's semantic relations [4] to achieve robust object categorization in real world scenes. These contextual features can be grouped into three categories: semantic context (*probability*), spatial context (*position*) and scale context (*size*). Contextual knowledge can be any information that is not directly produced by the appearance of an object. It can be obtained from the nearby image data, image tags or annotations and the presence and location of other objects. Next, we describe in detail each type of context and their most representative object categorization methods.

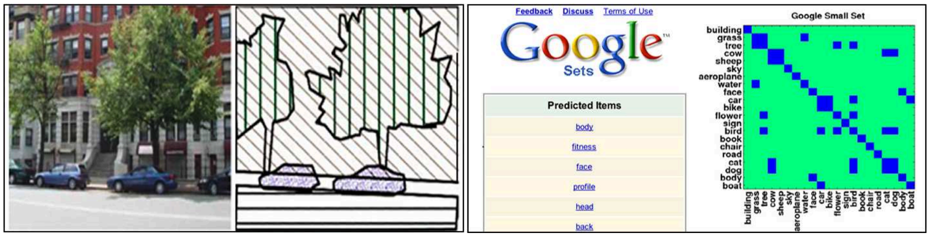
2.1 Semantic Context

Our experience with the visual world dictates our predictions about what other objects to expect in a scene. In real world images a scene is constituted by objects in a determined configuration. Semantic context corresponds to the likelihood of an object to be found in some scenes but not others. Hence, we can define semantic context of an object in terms of its co-occurrence with other objects and in terms of its occurrence in scenes. Early studies in psychology and cognition show that semantic context aids visual recognition in human perception. Palmer [23] examined the influence of prior presentation of visual scenes on the identification of briefly presented drawings of real-world objects. He found that observers accuracy at an object-categorization task was facilitated if the target (e.g. a loaf of bread) was presented after an appropriate scene (e.g. a kitchen counter) and impaired if the scene-object pairing was inappropriate (e.g. a kitchen counter and bass drum).

Early computer vision systems adopted these findings and defined semantic context as pre-defined rules [8, 12, 33] in order to facilitate recognition of objects in real world images. Hanson and Riseman [12] proposed the popular VISIONS schema system where semantic context is defined by hand coded rules. The system's initial expectation of the world is represented by different hypotheses (rule-based strategies) that predict the existence of other objects in the scene. Hypotheses are generated by a collection of experts specialized for recognizing different types of objects.

Recently, some computer vision approaches [11, 25, 34, 37, 38] have used statistical methods that can generalize and exploit semantic context in real world scenes for object categorization. The work by Wolf and Bileschi [38] used semantic context obtained from "semantic layers" available in training images, as shown in Figure 3 (a). Semantic layers indicate the presence of a particular object in the image. Each image present several semantic layers, one per each category present in the scene. In a semantic layer, each pixel is labeled with a value $v = 1$ if the pixel belongs to the object in the layer and $v = 0$ otherwise. Then, semantic context is presented in the form of a list of labels per pixel indicating the occurrence of a pixel in a particular object.

Context is commonly obtained from strongly labeled training data, but it can also be obtained from an external knowledge base as in [25]. Rabinovich *et al.* [25] derived semantic context querying the Google Sets[3] web application. Google Sets generates a list of possibly related items from a few examples. This information is represented by a binary co-occurrence matrix $\phi(i, j)$ that relates objects i and j in a scene. Each entry is set $\phi(i, j) = 1$ if objects i and j appear as related, or 0 otherwise. Figure 3 (b) shows the co-occurrence matrix used in [25].



(a) (b)

Fig. 3. (a) Example of training images and semantic layers used in [38]. Semantic layers encode ground truth information of the objects in the scene and also present useful information to elaborate semantic context features. (b) Google Sets web application and context matrix obtained from the predicted items list used in [25].

[3] labs.google.com/sets

Sources of semantic context in early works were obtained from common expert knowledge [8, 12, 33] which constrained the recognition system to a narrow domain and allowed just a limited number of methods to deal with uncertainty of real world scenes. On the other hand, annotated image databases [38] and external knowledge bases [25] can deal with more general cases of real world images. A similar evolution happened when learning semantic relations from those sources: pre-defined rules [8] were replaced by methods that learned the implicit semantic relations as pixel features [38] and co-occurrence matrices [25].

2.2 Spatial Context

Biederman's *position* class, also known as spatial context, can be defined by the likelihood of finding an object in some position and not others with respect to other objects in the scene. Bar *et al.* [2] examined the consequences of pairwise spatial relations on human performance in recognition tasks, between objects that typically co-occur in the same scene. Their results suggested that (i) the presence of objects that have an unique interpretation improve the recognition of ambiguous objects in the scene, and (ii) proper spatial relations among objects decreases error rates in the recognition of individual objects. These observations refer to the use of (i) semantic context and (ii) spatial context to identify ambiguous objects in a scene. Spatial context encodes implicitly the co-occurrence of other objects in the scene and offers more specific information about the configuration in which those objects are usually found. Therefore, most of the systems that use spatial information also use semantic context in some way.

The early work of Fischler [8] in scene understanding proposed a bottom-up scheme to recognize various objects and the scene. Recognition was done by segmenting the image into regions, labeling each segment as an object and refining object labels using spatial context as relative locations. Refining objects can be described by breaking down the object into a number of more "primitive parts" and by specifying an allowable range of spatial relations which these "primitive parts" must satisfy for the object to be present. Spatial context was stored in the form of rules and graph-like structures making the resulting system constrained to a specific domain.

In the last decade many approaches have considered using spatial context to improve recognition accuracy. Spatial context is incorporated from inter-pixel statistics [7, 13, 15, 20, 24, 27, 30, 34, 37, 38] and from pairwise relations between regions in images [6, 11, 16, 19, 31].

Recently, the work by Shotton *et al.* [30] acquired spatial context from inter-pixel statistics for object categorization. The framework learns a discriminative model of object classes that incorporates texture, layout and spatial information for object categorization of real world images. An unary classifier λ_i captures spatial interactions between class labels of neighboring pixel, and it is incorporated into a conditional random field [17]. Spatial context is represented by a look-up table with an entry for each class c_i and pixel index i:

$$\lambda_i(c_i, i; \theta_\lambda) = \log \theta_\lambda(c_i, \hat{i}) \tag{1}$$

The index $\hat{i}$ is the normalized version of the pixel index i, where the normalization allows for images of different sizes: the image is mapped onto a canonical square and $\hat{i}$ indicates the pixel position within this square. The model parameters are represented by θ_λ. An example of the learned classifiers are shown in Figure 4 (a).

Spatial context from pairwise relations has been addressed by the work of Kumar and Hebert [16] as well. Their method presents a two-layer hierarchical formulation to exploit different levels of spatial context in images for robust classification. Layer 1 models region to region interactions and layer 2 objects to objects interactions, as shown in Figure 4 (b). Objects to regions interactions are modeled between layer 1 and layer 2. Pairwise spatial features between regions are binary indicator attributes for three pre-defined interactions: *above*, *beside* or *enclosed*. The pairwise features between the object to object and object to region are simply the difference in the coordinates of the centroids of a region and a patch. Then, spatial context is defined in two different levels: as a binary feature for each interaction in layer 1 and as the difference in the coordinates of the centroids in layer 2.

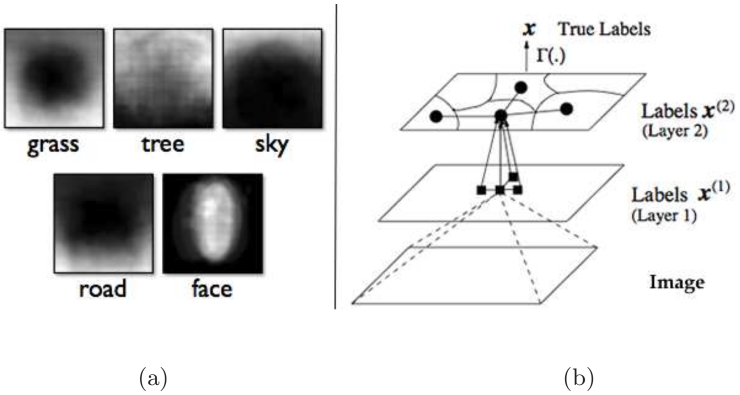


Fig. 4. (a) Spatial context classifiers learned in [30] for five different classes: grass, tree, sky, road and face. (b) Two-layer hierarchical formulation for spatial context used in [16].

Same as semantic context in early works, spatial context sources were obtained from common expert knowledge [8, 12, 33] which constrained the recognition system to a specific domain failing to deal with uncertainty of real world scenes. On the contrary, recent works in computer vision such as [16, 30] use strongly annotated training data as main source of spatial context with the hope of generalize cases. Even though Fischler [8] used pre-defined rules to define spatial interactions, Kumar and Hebert [16] also pre-define interactions that correspond to general object configurations in real world scenes. On the other hand, Shotton *et al.* [30] learned these interactions implicitly from training data using statistical methods that can capture many more object configurations than the pre-defined rules.

2.3 Scale Context

Common approaches to object recognition require exhaustive exploration of a large search space corresponding to different object models, locations and scales. Prior information about the sizes in which objects are found in the scene can facilitate object detection. It reduces the need for multiscale search and focuses computational resources into the more likely scales.

Biederman's *familiar size* class is a contextual relation based on the scales of an object with respect to others. This contextual cue establishes that objects have a limited set of size relations with other objects in the scene. Scale context requires not only the identification of at least one other object in the setting, but also the processing of the specific spatial and depth relations between the target and this other object.

The CONDOR system by Strat and Fischler [33] was one of the first computer vision systems that added scale context as a feature to recognize objects. Scale information of an object was obtained from the camera's meta-data such as camera position and orientation, geometric horizon, digital terrain elevation data and map. This information was integrated into the system to generate hypothesis about the scene in which object's configurations are consistent with a global context.

Lately, a handful of methods for object recognition have used this type of context [19, 20, 24, 34–36]. Torralba *et al.* [34] introduced a simple framework for modeling the relationship between context and object properties. Scale context is used to provide a strong cue for scale selection in the detection of high level structures as objects. Contextual features are learned from a set of training images where object properties are based on the correlation between the statistics of low-level features across the entire scene. Figure 5 (a) shows an example of a training image and its corresponding annotation.

In [34] , an object in the image is defined as $O = \{o, x, \sigma\}$ where o is the category label, x is the location and σ is the scale of the object in the scene. Scale context (σ) depends on both the relative image size of the object at one fixed distance and the actual distance D between the observer and the object. Using these properties and the contextual feature of the entire scene v_C for a category C, automatic scale selection is performed by the PDF $P(\sigma|o,\mathrm{v}_C)$. Examples of automatic scale selection are shown in Figure 5 (b).

Scale context shows to be the hardest relation to access, since it requires a more detailed information about the objects in the scene. While analyzing generic 2D images, camera's meta-data used in [33] is generally not available. Instead, one needs to derive context directly from the input image itself as done in [34].

The majority of the models reviewed here use one or two explicit types of context. Spatial and scale context are the most exploited types of context by

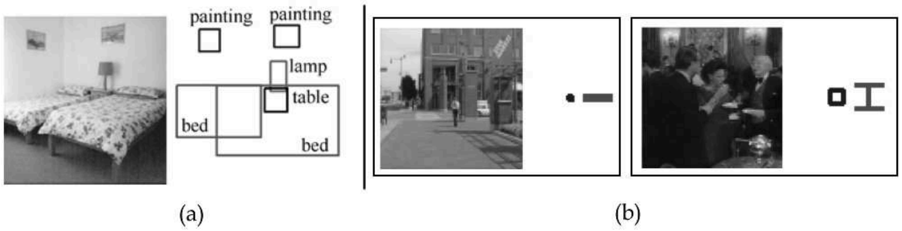


Fig. 5. (a) An example of training image and ground truth annotation used in [34]. (b) Scale context permits automatic scale detection for face recognition. The square's size corresponds to the expected height of heads given the scale prior. The line at the right hand indicates the real height of the heads in the image.

recognition frameworks. Generally, semantic context is implicitly present in spatial context, as information of object co-ocurrences come from identifying objects for the spatial relations in the scene. The same happens to scale context, as scale is measured with respect to others objects. Therefore, using spatial and scale context involve using all forms of contextual information in the scene.

Although semantic context can be inferred from other types of context, it is the only context type that brings out the most valuable information for improving recognition. Considering the variability of the object configurations in the scene, scale and spatial relations vary in grater extent than the co-ocurrence of objects. Co-occurrences are much easier to access than spatial or scale relationships and much faster to process and compute. On the other hand, using all types of context can give a better representation of the configuration of objects in the scene, producing better performance in recognition tasks.

With respect to the sources of contextual information, very little has been done for using external sources in cases where training data is weakly labeled. In most of the cases, contextual relations are computed from training data, which can sometimes fail to express general cases. To model sources of variability in real world images, approaches to object categorization require large labeled data sets of fully annotated training images. Typical annotations in these fully labeled data sets provide masks or bounding boxes that specify the locations and scales of objects in each training image. Though extremely valuable, this information is prone to error and expensive to obtain. Using publicly available knowledge-bases can contribute to add contextual information for recognition tasks.

3 Contextual Levels

Object recognition models have considered the use of context information from a "global" or "local" image level. *Global context* considers image statistics from the image as a whole (e.g. a kitchen will predict the presence of a stove). *Local context*, on the other hand, considers context information from neighboring areas of the object (e.g. a nightstand will predict the presence of an alarm clock). These two different trends have found their motivation from psychology studies in object recognition. Next we review these two types of contextual levels together with examples of models that have adopted these directions.

3.1 Global Context

Studies in psychology [21, 26] suggest that perceptual processes are hierarchically organized so they proceed from global structuring towards more and more detailed analysis. Thus the perceptual system treats every scene as if it were in a process of being focused or zoomed in on. These studies imply that information about scene identity may be available before performing a more detailed analysis of the individual objects.

Under this premise, global context exploits scene configuration (image as a whole) as an extra source of global information across categories. The structure of a scene image can be estimated by the mean of global image features, providing a statistical summary of the spatial layout properties. Many object categorization frameworks have incorporated this prior information for their localization tasks [20, 27, 34, 36, 37].

Murphy *et al.* [20] exploited context features using a scene "gist" [34], which influences priors of object existence and global location within a scene. The "gist" of an image is a holistic, low-dimensional representation of the whole image. Figure 6 (b) shows an example of the "gist" of a corridor. The work of Torralba *et al.* [34] shows that this is sufficient to provide a useful prior for what types of objects may appear in the image, and at which location/scale. The background (or scene)provides an a likelihood of finding an object in the image (for example, one is unlikely to find a boat in a room). It can also indicate the most likely positions and scales at which an object might appear (e.g. pedestrians on walkways in an urban area).

In [20] the "gist" is defined as the feature vector v^G that summarizes the whole image. In order to obtain v^G, a set of spatially averaged filter-banks are applied to the whole image. Then, Principle Component Analysis (PCA) is used to reduce the high dimensionality of the resulting output vector. By combining v^G with the outputs of boosted object detectors, final detectors are ran in locations/scales that the objects are expected to be found, therefore improving speed and accuracy. Using context by processing the scene as a whole and without first detecting other objects can help to reduce false detections.

3.2 Local Context

Local context information is derived from the area that surrounds the object to detect (other objects, pixels or patches). The role of local context has been studied in psychology for the task of object [23] and face detection [32]. Sinha and Torralba [32] found that inclusion of local contextual regions such as the facial bounding contour substantially improves face detection performance, indicating that the internal features for facial representations encode this contextual information.

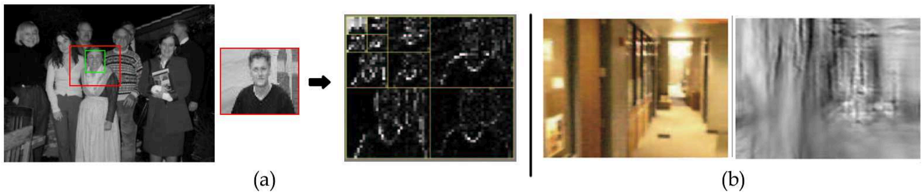


Fig. 6. (a) Red window indicates local context and the green window indicates the region of interest for the appearance features. Local context feature by Kruppa and Schiele [15] (b) A training image (on the left) and the "gist" (on the right) for the "corridor" category by Murphy *et al.* [20].

Local context features can capture different local relations such as pixel, region and object interactions. Many object categorization models have used local context from pixels [6, 7, 13, 15, 30], patches [16, 19, 31] and objects [11, 12, 25, 33, 35, 38] that surrounds the target object, greatly improving the task of object categorization. These interactions are reviewed in detail in Section 4.

Kruppa and Schiele [15] investigated the role of local context for face detection algorithms. In their work, an appearance-based object detector is trained with instances that contain a persons entire head, neck and part of the upper body (as shown in Figure 6 (a)). The features of this detector capture local arrangements of quantized wavelet coefficients. The wavelet decomposition showed that local context captured most parts of the upper bodys contours, as well as the collar of the shirt and the boundary between forehead and hair. At the core of the detector there is a Naive Bayes classifier:

$$\prod_{k=1}^{n}\prod_{x,y\in region}\frac{p_k(pattern_k(x,y),i(x),j(y)|object)}{p_k(pattern_k(x,y),i(x),j(y)|nonobject)} > \theta \tag{2}$$

where θ is an acceptance threshold and p_k are two likelihood functions that depend on coarse quantizations $i(x)$ and $j(y)$ of the feature position within the detection window (2). This spatial dependency allows to capture the global geometric layout: within the detection windows certain features might be likely to occur at one position but unlikely to occur at another.

Using local context yields correct detections that are beyond the scope of the classical object-centered approach, and holds not only for low resolution cases but also for difficult poses, occlusion and difficult lighting conditions.

One of the principal advantages of global over local context is that global context is computationally efficient as there is no need to parse the image or group components in order to represent the spatial configuration of the scene. However when the number of objects to recognize and scenes increases, global context cannot discriminate well between scenes since many objects may share the same scene, and scenes may look similar to each other. In this case, computation becomes expensive as we have to run all object detectors on the image.

Local context improves recognition over the capabilities of object-centered recognition frameworks since it captures different range of interactions between objects. Its advantage over global context is based on the fact that for global context scene must be taken as one complete unit and spatially localized processing can not take place.

The fact that local context representation is still object-centered, as it requires object recognition as a first step, is one of the key differences with global context. The image patches that do not satisfy the similarity criteria to objects are discarded and modeled as noise. Global context propose to use the background statistics as an indicator of object presence and properties. However, one drawback of the current "gist" implementation is that it cannot carry out partial background matching for scenes in which large parts are occluded by foreground objects.

4 Contextual Interactions

We have seen that object categorization models exploit context information from different types of contextual interactions and consider different image levels. When we consider local context, contextual interactions can be grouped in three different types: pixel, region and object interactions. However, when we consider global context, we have contextual interactions between objects and scenes. Next, we review in detail these different interactions and discuss the different object categorization models that have made key contributions using these contextual interactions.

4.1 Local Interactions

The work by Rutishauser *et al.* [28] proposed that recognition performance for objects in highly cluttered scenes can be improved dramatically with use of bottom-up attentional frameworks. Local context involves bottom-up processing of contextual features across images, improving performance and recognition accuracy. Next we review local interactions from different local context levels that are incorporated into bottom-up fashion for categorization models.

Pixel Interactions Pixel level interactions are based on the notion that neighboring pixels tend to have similar labels, except at the discontinuities. Several object categorization frameworks model interactions at pixel level in order to implicitly capture scene contextual information [6, 13, 15, 27, 30, 34, 38]. Pixel level interactions can also derive information about object boundaries, leading to an automatic segmentation of the image into objects [6, 13, 30] and a further improvement in object localization accuracy.

The problem of obtaining contextual features by using pixel level interactions is addressed by the work of He *et al.* [13]. The model combines local classifiers with probabilistic models of label relationships. *Regional label features* and *global label features* describe pixel level interactions, as shown in Figure 7 (a). Regional

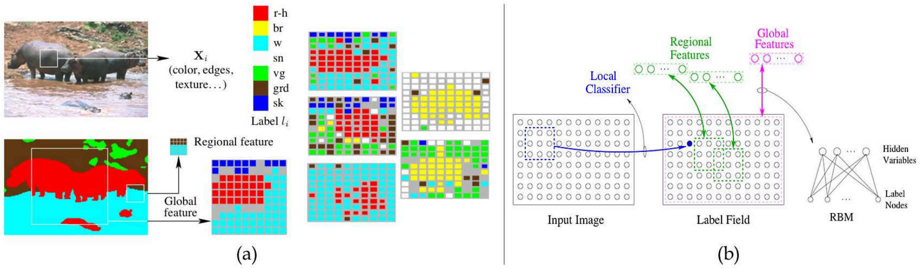


Fig. 7. (a) Contextual features and example of regional label features in [13]. (b) Framework for contextual features by He *et al.* [13].

features represent local geometric relationships between objects, such as edges, corners or T-junctions. Actual objects involved in the interaction are specified by the features, thus avoiding impossible combinations such as ground-above-sky border. Regional features are defined on 8 × 8 regions with an overlapping of 4 pixels in each direction and extracted from training data.

Global features correspond to domains that go from large regions in the scene to the whole image. Pixel level interactions are encoded in the form of a label pattern, which is configured as a Restricted Boltzmaann Machine (RBM) as shown in Figure 7 (b). These features are defined over the entire image, but in principle smaller fields anchored at specific locations can be used. Regional and global label features are described by probabilistic models and their distributions are combined into a conditional random field [17].

Region Interactions Region level interactions have been extensively investigated in the area of context-based object categorization tasks [6, 7, 15, 16, 19, 20, 27, 31, 37] since regions follow plausible geometrical configurations. These interactions can be divided into two different types: interaction between image patches/segments and interaction between object parts.

Interactions between object parts can derive contextual features for recognizing the entire object. Fink and Perona [7] proposed a method termed *Mutual Boosting* to incorporate contextual information for object detection from object's parts. Multiple objects and part detectors are trained simultaneously using AdaBoost [9, 10]. Context information is incorporated from neighboring parts or objects using training windows that capture wide regions around the detected object. These windows, called *contextual neighborhoods*, capture relative position information from objects or parts around and within the detected object. The framework simultaneously trains M object detectors that generate M intensity maps $H^{m=1,..,M}$ indicating the likelihood of object m appearing at different positions in a target image. At each boosting iteration t the M detectors emerging at the previous stage $t-1$ are used to filter positive and negative training images, thus producing intermediate m detection maps H^m_{t-1}. Next, the

Mutual Boosting stage takes place and all the existing H^m_{t-1} maps are used as additional channels out of which new contrast features are selected. Mutual dependencies must be computed in a iterative fashion, first updating one object then the other. Figure 8 (a) shows contextual neighborhoods $C^{m[i]}$ for positive and negative training images.

Models that exploit interactions between patches commonly involve some type of image partitioning. The image is usually divided into patches [16, 15, 19, 20, 27, 31, 37] or into segments by a semantic scene segmentation algorithm [6].

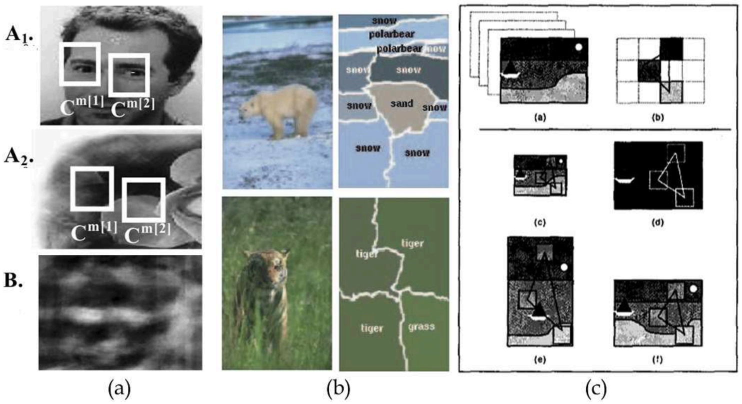


Fig. 8. (a) A1 & A2: position of positive and negative examples of eyes in natural images and B: Eye intensity (eyeness) detection map of an image in [7]. (b) Results of [6] using segment interactions. (c) Lipson's [19] patch level interactions that derive spatial templates.

The work by Lipson *et al.* [19] exploits of patch level interactions for capturing scene's global configuration. The approach employes qualitative spatial and photometric relationships within and across regions in low resolution images, as shown in Figure 8 (c). The algorithm first computes all pairwise qualitative relationships between each low resolution image region. For each region, the algorithm also computes a rough estimate of its color from a coarsely quantized color space as a measure of perceptual color. Images are grouped into directional equivalence classes, such as "above" and "below", with respect to explicit interactions between region. Figure 8 (c) shows examples of images and common region configurations.

The framework introduced by Carbonetto *et al.* [6] uses Normalized Cuts [29] algorithm to partition images into regions for learning both word-to-region associations and segment relations. The contextual model learns the co-occurrence of blobs (set of features that describe a segment) and formulates a spatially consistent probabilistic mapping between continuous image feature vectors and word tokens. Segment level interactions describe the "next to" relation between blob annotations. Interactions are embedded as clique potentials of a Markov Random Field (MRF) [18]. Figure 8 (b) shows examples of test images regions and labels.

Object Interactions The most intuitive type of contextual interactions correspond to the object level, since object interactions are more natural to the human perception. They have been extensively studied in the areas of psychology and cognitive sciences [1–4, 23]. Several frameworks have addressed these interactions including early works in computer vision [8, 11, 12, 15, 16, 25, 35].

The recent work of Torralba *et al.* [35] exploits contextual correlations between object classes by using Boosted Random Fields (BRFs). BRFs build on both boosting [9, 10] and conditional random fields (CRFs) [17], providing a natural extension of the cascade of classifiers by integrating evidence from other objects. The algorithm is computationally efficient given that quickly rejects possible negative image regions.

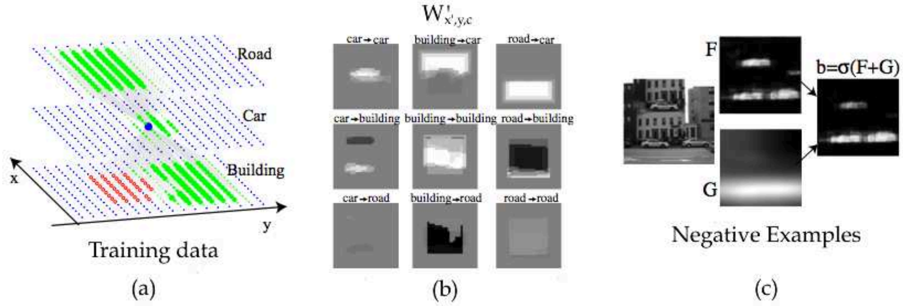


Fig. 9. (a) Training data with semantic layer labels (same as in [38]). (b) Example of binary kernel. (c) Negative examples for the boosting phase of the model.

Information from object level interactions is embedded into binary kernels $W'_{x',y',c'}$ that define, for each node x, y of object class c, all the nodes from which it has contextual interactions. These kernels are chosen by sampling patches of various sizes from the training set image labeling. This allows generating complicated patterns of connectivity that reflect the statistics of object co-occurrences in the training set. The algorithm learns to first detect easy (and large) objects, reducing the error of all classes. Then, the easy-to-detect objects pass information to the harder ones.

Combining more than one interaction level into a context-based object categorization model has been addressed by few models [6, 15, 16] in order to achieve better recognition performance. One advantage of these models over the ones that use one interaction level is that these models utilize a more complete information about the context in the scene. Each level captures a different range of relationships: objects interactions capture in a better way interactions from objects that can be fairly apart in the scene from each other. Pixel interactions, in the other hand, can capture more detailed interactions between objects that are closely to each other (e.g. boundaries between objects). A clear disadvantage of combining different interaction levels is that the expensive and complex computations needed to obtain and merge the different information.

Pixel level interactions are more computationally intensive to obtain, since we need to consider several combination of small windows from the image. On the other extreme, using object level interactions presents efficient extraction as the number of regions to consider is equal to the number of objects present in the scene (usually small). When considering region level interactions, models that pre-process the image using segmentation algorithms are more efficient capturing contextual interactions that the ones that use grid-like segmentations, as the number of regions considered tend to be smaller.

4.2 Global Interactions

Recent behavioral and modeling research suggests that early scene interpretation may be influenced by global image properties that are computed by processes that do not require selective visual attention [22]. Global context, represented as a scene prior, has been considered by categorization models as a single entity that can be recognized by means of a scene-centered representation bypassing the identification of the constituent objects. Top-down processing is necessary when exploiting and incorporating global context into with recognition tasks. Next we review different frameworks in recognition that use global context by capturing object-scene interactions.

Object-Scene Interactions A number of recognition frameworks [27, 34–37] have exploited object-scene interactions to efficiently use context in their models. The work by Russell *et al.* [27] exploits scene context by formulating the object detection problem as one of aligning elements of the entire scene to a large database of labeled images. The background, instead of being treated as a set of outliers, is used to guide the detection process. The system transfers labels from the training images that best match the query image. Commonalities amongst the labeled objects are assumed in the retrieved images and images are clustered to form candidate scenes.

Object-scene interactions are modeled using training image clusters, which give hints as to what objects are depicted in the query image and their likely location. The relationship between object categories o, their spatial location x within an image, and their appearance g is modeled by computing the following joint distribution:

$$p(o,x,g|\theta,\phi,\eta) = \prod_{i=1}^{N}\prod_{j=1}^{M_i}\sum_{h_{i,j}=0}^{1} p(o_{i,j}|h_{i,j},\theta)p(x_{i,j}|h_{i,j},\phi)p(g_{i,j}|o_{i,j},h_{i,j},\eta) \quad (3)$$

where N is the number of images, each having M_i object proposals over L object categories. The likelihood of which object categories appear in the image is modeled by $p(o_{i,j}|h_{i,j} = m, \theta_m)$, which corresponds to the object-scene interactions.

Many advantages and disadvantages can be considered when analyzing local interaction and global interactions. Global interactions are efficiently when recognizing novel objects since applying an object detector densely across the entire image for all object categories is not needed. Global context constrains which object categories to look for and where. The down side of this great advantage is that training is computationally expensive due to the inference that has to be done for finding parameters in the graphical model. On the other hand, local interactions are easily accessible from training data, without expensive computations. The problem arises when combining local context features with local appearance features.

5 Integrating Context

The problem of integrating contextual information into an object categorization framework is a challenging task since it needs to combine objects appearance information with contextual constraints imposed on those objects given the scene. In order to address this problem, machine learning techniques are borrowed as they provide efficient and powerful probabilistic algorithms. The choice of these models is based on the flexibility and efficiency of combining context features at a given stage in the recognition task. Here, we grouped different approaches for integrating context in two distinctive groups: classifiers and graphical models.

5.1 Classifiers

Several methods [7, 15, 20, 38] have chosen classifiers over other statistical models to integrate their context with their appearance features. The main motivation for using classifiers is to combine the outputs of local appearance detectors (use as appearance features) with contextual features obtained from either local or global statistics. Some discriminative classifiers have been used for this purpose, such boosting [7, 38] and Logistic Regression [20] in the attempt to maximize the quality of the output on the training set. Generative classifiers have been also used to combine these features, such as Naive Bayes classifier [15]. Discriminative learning often yields higher accuracy than modeling the conditional density functions. However, handling missing data is often easier with conditional density models.

Wolf and Bileschi [38] utilize boosted classifiers [9] in a rejection cascade for incorporating local appearance and contextual features. The construction of the context feature is done in two stages. In the first stage, the image is processed to calculate the low level and semantic information. In the second stage, the context feature is calculated at each point by collecting samples of the previously computed features at pre-defined relative positions. Then, a semantic context detector is learned via boosting, trained to discriminate between positive and negative examples of the classes. Same approached is used to learned appearance features. In order to detect objects in a test image, the context based detector is applied first. All pixels classified as object-context with confidence greater than the confidence threshold TH_C are then passed to the appearance detector for a secondary classification. In the same way as with context, pixels are classified by a appearance detector as objects with confidence greater than the confidence threshold TH_A. A pixel is judged to be an object detection only if the pixel passes both detectors.

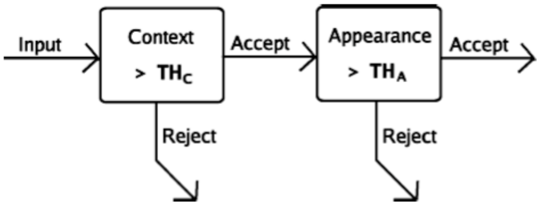


Fig. 10. Classification scheme using boosted classifiers in [38].

Advantages of boosting include rapid classification, simplicity and easily programming. Prior knowledge about the base learner is not required, so boosting can be flexibly combined with any method for finding base classifiers. Instead of trying to design a learning algorithm that is accurate over the entire space, boosting focus on finding base learning algorithms that only need to be better than random. One of the drawbacks of the model is that performance on a particular problem is dependent on the data and the base learner. Consistent with theory, boosting can fail to perform well given insufficient data, overly complex base classifiers or base classifiers that are too weak.

5.2 Graphical Models

Graphical models provide a simple way to visualize the structure of a probabilistic model. The graphical model (graph) captures the way in which the joint distribution over all random variables can be decomposed into a product of factors each depending on a subset of the variables. Hence they provide a powerful yet flexible framework for representing and manipulating global probability distributions defined by relatively local constraints. Many object categorization frameworks have used graphical models to model context since they can encode the structure of local dependencies in an image from which we would like to make globally consistent predictions.

A handful of frameworks have exploited *directed graphical models* [27, 31, 34] to incorporate contextual features with their appearance-based detectors.

Directed graphical models are global probability distributions defined on directed graphs using local transition probabilities. They are useful for expressing causal relationships between random variables since they assume that the observed image has been produced by a causal latent process. Directed graphical models compute the joint distribution over the node variables as follows:

$$P(x) = \prod_i P(x_i|pa_i) \tag{4}$$

where pa_i is the parent of node x_i. These graphical models assume that objects are conditionally independent given the scene.

On the other hand, a majority of object categorization models uses *undirected graphical models* [6,13,16,25,30,35,37] since they are better suited to express soft constraints between random variables. Undirected graphical models are global probability distributions defined on undirected graphs using local clique potentials. They are better suited to handle interactions over image partitions since usually there exists no natural causal relationships among image components. The joint probability distribution is expressed as:

$$P(x) = \frac{1}{Z}\prod_C \psi_C(x_C), \quad \textit{where} \quad Z = \sum_x \prod_C \psi_C(x_C) \tag{5}$$

and $\psi_C(x_C)$ are the potential function over the maximal cliques C of the graph. Special cases of undirected graphical models used for modeling context include Markov Random Fields (MRFs) [6] and Conditional Random Fields (CRFs) [13,16,25,30,35,37]. MRFs are typically formulated in a probabilistic generative framework modeling the joint probability of the image and its corresponding labels. Due to the complexity of inference and parameter estimation in MRFs, only local relationships between neighboring nodes are incorporated into the model. Also, MRFs do not allow the use of global observations to model interactions between labels. CRFs provide a principled approach to incorporate these data-dependent interactions. Instead of modeling the full joint distribution over the labels with an MRF, CRFs model directly the conditional distribution which requires fewer labeled images and the resources are directly relevant to the task of inferring labels.

Therefore, CRFs models have become popular owing to their ability to directly predict the segmentation/labeling given the observed image and the ease with which arbitrary functions of the observed features can be incorporated into the training process. Scene regions and object regions are related through geometric constraints. CRF models can be applied either at the pixel-level [13,16,30] or at the coarser level [25,37]. Next we review in detail how CRFs are commonly used for integrating context.

Conditional Random Fields Conditional Random Fields are used to learn the conditional distribution over the class labeling given an image. The structure permits incorporating different types of cues in a single unified model, maximizing object label agreement according to contextual relevance. Since object presence is assumed conditionally independent given the scene, the conditional probability of the class labels $\boldsymbol{x}$ given an image y has the form:

$$P(\boldsymbol{x}|y,\boldsymbol{\theta}) = \frac{1}{Z(y,\boldsymbol{\theta})}\exp(\sum_j F_j(x_j,y;\theta_j)) \tag{6}$$

$$F_j(x_j,y;\theta_j) = \sum_{i=1}^{n}\theta_j f_j(x_{i-1},x_i,y,i) \tag{7}$$

where F_j are the potential functions and $Z(y,\boldsymbol{\theta})$ is the normalization factor, also known as the partition function. Potentials can be unary or pairwise functions and they represent transition or state functions on the graph. The main challenge in probability calculation is to compute the partition function $Z(y,\boldsymbol{\theta})$. This function can be calculated efficiently when matrix operations are used. For this, the conditional probability can be written as:

$$P(\boldsymbol{x}|y,\boldsymbol{\theta}) = \frac{1}{Z(y,\boldsymbol{\theta})}\prod_{i=1}^{n+1} M_i(x_{i-1},x_i|y) \tag{8}$$

$$M_i(x',x|y) = \exp(\sum_j \theta_j f_j(x',x,y,i)) \tag{9}$$

The normalization factor $Z(y,\theta)$ for labels $\boldsymbol{x}$, may be computed from the set of M_i matrices using closed semi-rings. For contex-based categorization, each matrix M_i embeds contextual interactions to be imposed in the recognition task.

Context based object categorization models [11,13,16,25,30,35,37] use CRFs to integrate contextual and appearance information from pixel level [30], object level [25] and multiple image levels [13,16,35,37]. The conditional probability over the true labels can be computed given the entire image [13,16,30] or given a set of image patches or segments [25,35,37].

Maximum likelihood chooses parameter values such that the logarithm of the likelihood, known as the log-likelihood, is maximized. Parameter estimation is commonly computed by recognition frameworks [11,13,16,25,30,35,37] using methods such as gradient descend, alpha expansion graph cut [5] and contrastive divergence (CD) [14]. Different techniques are used to find the maximum marginal estimates of the labels on the image, such as loopy belief propagation (BP), maximum posterior marginal (MPM) and Gibbs sampling since exact maximum a posteriori (MAP) is infeasible.

In the case where the entire image y is considered for the conditional probability, CRF potentials are learned from low level features, including output labels from pixel classifiers, texture features, edge information and low level context interactions. Potential functions encode a particular constraint between the image and the labels within a pixel/region of the image. Given that each pixel or a small image region is considered to be a node in the graph, parameter estimation and inference become computationally expensive. However, this technique achieves both recognition and segmentation on the images.

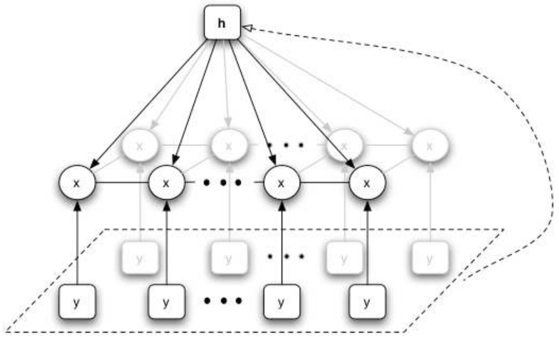


Fig. 11. Conditional Random Field used in [37]. Squares indicate feature functions and circles indicate variable nodes x_i. Arrows represent single node potentials due to feature functions, and undirected edges represent pairwise potentials. Global context is represented by **h.**

When the conditional probability considers a set of image patches or segments $y_i \in \boldsymbol{y}$, classifier outputs (labels) are combined with pairwise contextual interactions between the regions. Potential functions represent transition functions between object labels, capturing important long distance dependencies between whole regions and across classes. Since each patch/segment is a node in the graph, parameter computation is cheaper and the framework can scale more favorably when the number of categories to recognize increases.

One of the advantages of using CRFs in general is that the conditional probability model can depend on arbitrary non-independent characteristics of the observation, unlike a generative image model which is forced to account for dependencies in the image, and therefore requires strict independence assumptions to make inference tractable. The down side of using CRFs is that inferring labels from the exact posterior distribution for complex graphs is intractable.

6 Conclusions and Open Issues

The importance of context in object recognition and categorization has been discussed for many years. Scientists from different disciplines such as cognitive sciences and psychology have considered context information as a path to efficient understanding of the natural visual world. In computer vision, several object categorization models have addressed this point, confirming that contextual information can help to successfully disambiguate appearance inputs in recognition tasks.

In this critical study, we have addressed the problem of incorporating different types of contextual information for robust object categorization in computer vision. We reviewed a variety of different approaches of context based object categorization models, the most common levels of extraction of context and the different levels of contextual interactions. We have also examined common machine learning models that integrate context information into object recognition frameworks.

We believe that contextual information can benefit categorization tasks in two ways: (i) as a prior to recognize certain objects in images and (ii) as an advocate for label agreement to disambiguate objects appearance. However if the target object is the only labeled object in the database there are no sources of contextual information we can exploit. This fact points out the need for external sources of context (as in [25]) that can provide this information when training data is weakly or not labeled.

Considering the image level interactions, pixel level models have comparable performance to state-of-the-art patch and object level models, however complexity of these models can grow quickly as the number of classes increase. Scalability can be a problem for pixel level models, so considering a coarser level can optimize expensive computations.

Models that combine different interaction levels of context can potentially benefit from the extra information, nevertheless parameter estimation for the different levels and cue combination can result in complex and expensive computations. Same happens when both levels of contextual extraction are combined into a single model.

The majority of the context-based models include at most two different types of context, semantic and spatial, since the complexity to determine scale context is still high for 2D images. Future work will include incorporating semantic, spatial and scale context into a recognition framework to assess the contribution of these features. Also, other machine learning models will be considered for a better integration of context features.

References

1. M. Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5(8):617–629, 2004.
2. M. Bar and S. Ullman. Spatial context in recognition. *Perception. 25:343-352.*, 1993.
3. I. Biederman. Perceiving real-world scenes. *Science*, 177(7):77–80, 1972.
4. I. Biederman, R. J. Mezzanotte, and J. C. Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14(2):143–177, April 1982.
5. Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *ICCV*, 1:105–112, 2001.
6. P. Carbonetto, N. de Freitas, and K. Barnard. A statistical model for general contextual object recognition. *ECCV*, pages 350–362, 2004.
7. M. Fink and P. Perona. Mutual boosting for contextual inference. *NIPS*, 2003.

8. M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 100(22):67–92, 1973.
9. Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.
10. J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Technical Report , Standford University*, 1998.
11. C. Galleguillos, A. Rabinovich, and S. Belongie. Object categorization using co-occurrence, location and appearance. *CVPR*, 2008.
12. A. Hanson and E. Riseman. Visions: A computer vision system for interpreting scenes. *Computer Vision Systems*, pages 303–334, 1978.
13. X. He, R. S. Zemel, and M. Á. Carreira-Perpiñán. Multiscale conditional random fields for image labeling. *CVPR*, pages 695–702, 2004.
14. G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.
15. H. Kruppa and B. Schiele. Using local context to improve face detection. *BMVC*, 2003.
16. S. Kumar and M. Hebert. A hierarchical field framework for unified context-based classification. *ICCV*, pages 1284–1291, 2005.
17. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*, 2001.
18. S. Li. Markov random field modeling in computer vision. *Springer Computer Science Workbench Series*, page 264, 1995.
19. P. Lipson, E. Grimson, and P. Sinha. Configuration based scene classification and image indexing. *CVPR*, page 1007, 1997.
20. K. Murphy, A. Torralba, and W. Freeman. Using the forest to see the tree: a graphical model relating features, objects and the scenes. *NIPS*, 2003.
21. D. Navon. Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology*, 9(3):353–383, 1977.
22. A. Oliva, A. Torralba, M. Castelhano, and J. Henderson. Top-down control of visual attention in object detection. *Image Processing*, 1, 2003.
23. S. E. Palmer. The effects of contextual scenes on the identification of objects. *Memory and Cognition*, 1975.
24. D. Parikh, C. Zitnick, and T. Chen. From appearance to context-based recognition: Dense labeling in small images. *CVPR*, 2008.
25. A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.
26. R. Rensink, J. O'Regan, and J. Clark. The need for attention to perceive changes in scenes. *Psychological Science*, 8(5):368–373, 1997.
27. B. C. Russell, A. Torralba, C. Liu, R. Fergus, and W. T. Freeman. Object recognition by scene alignment. *NIPS*, 2007.
28. U. Rutishauser, D. Walther, C. Koch, and P. Perona. Is bottom-up attention useful for object recognition? *CVPR*, 2004.
29. J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22, 2000.
30. J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling appearance, shape and context. *IJCV*, 2007.
31. A. Singhal, J. Luo, and W. Zhu. Probabilistic spatial context models for scene content understanding. *CVPR*, 01:235, 2003.
32. P. Sinha and A. Torralba. Detecting faces in impoverished images. *Journal of Vision*, 2(7):601, 2002.

33. T. Strat and M. Fischler. Context-based vision: Recognizing objects using information from both 2-d and 3-d imagery. *Pattern Analysis and Machine Vision*, 13(10):1050–1065, October 1991.
34. A. Torralba. Contextual priming for object detection. *International Journal of Computer Vision (IJCV), 53(2):153–167, 2003.*, 2003.
35. A. Torralba, K. Murphy, and W. Freeman. Contextual models for object detection using boosted random fields. *NIPS*, 2004.
36. A. Torralba, K. Murphy, W. Freeman, and M. Rubin. Context-based vision system for place and object recognition. *ICCV*, 2003.
37. J. Verbeek and B. Triggs. Scene segmentation with crfs learned from partially labeled images. *NIPS*, 11 2008.
38. L. Wolf and S. Bileschi. A critical view of context. *International Journal of Computer Vision (IJCV)*, 2006.