Context-based vision system for place and object recognition

Antonio Torralba
MIT AI lab
Cambridge, MA 02139

Kevin P. Murphy
MIT AI lab
Cambridge, MA 02139

William T. Freeman
MIT AI lab
Cambridge, MA 02139

Mark A. Rubin
Lincoln Labs
Lexington, MA 02420

Abstract

While navigating in an environment, a vision system has to be able to recognize where it is and what the main objects in the scene are. In this paper we present a context-based vision system for place and object recognition. The goal is to identify familiar locations (e.g., office 610, conference room 941, Main Street), to categorize new environments (office, corridor, street) and to use that information to provide contextual priors for object recognition (e.g., tables are more likely in an office than a street). We present a low-dimensional global image representation that provides relevant information for place recognition and categorization, and show how such contextual information introduces strong priors that simplify object recognition. We have trained the system to recognize over 60 locations (indoors and outdoors) and to suggest the presence and locations of more than 20 different object types. The algorithm has been integrated into a mobile system that provides real-time feedback to the user.

1. Introduction

We want to build a vision system that can tell where it is and what it is looking at as it moves through the world. This problem is very difficult and is largely unsolved. Our approach is to exploit *visual context*, by which we mean a low-dimensional representation of the whole image (the "gist" of the scene) [6]. Such a representation can be easily computed without having to identify specific regions or objects. Having identified the overall type of scene, one can then proceed to identify specific objects within the scene.

The power of, and need for, context is illustrated in Figure 1. In Figure 1(a), we see a close-up view of an object; this is the kind of view commonly studied in the object recognition community. The recognition of the object as a coffee machine relies on knowing detailed local properties (its typical shape, the materials it is made of, etc.). In Figure 1(b), we see a more generic view, where the object occupies a small portion of the image. The recognition now relies on contextual information, such as the fact that we are in a kitchen. Contextual information helps to disambiguate the identity of the object despite the poverty of the local object features (Figure 1(c)).

Object recognition in context is based on our knowledge of scenes and how objects are organized. The recognition of the scene as a kitchen reduces the number of objects that need to be considered, which allows us to use simple features for recognition. Furthermore, the recognition of this scene as a *particular* kitchen (here, the kitchen of our lab) further increases the confidence about the identity of the object. Hence place recognition can be useful for object recognition.

Traditionally, place and object recognition are considered separate problems. Place recognition is mostly studied in the mobile robotics community, where the problem is called topological localization. (Topological localization refers to identifying the discrete location of the robot (e.g., which room it is in), as opposed to metric localization, which refers to establishing the precise coordinates of the robot.) The main novelty of our approach for place recognition compared to previous ones, such as [17, 3], is that we also try to identify the category of the location, which works even for places which have not been seen before.

Previous approaches to object recognition have focused, for the most part, on using local features to classify each image patch independently (see e.g., [4, 7, 9, 10, 16]. Our approach uses global image features to predict the scene, and then uses the scene as a prior for the local detectors.

Another difference between our approach to object recognition and previous approaches is that we try to recognize a large set of object types (24) in a natural, unconstrained setting (images are collected by a wearable camera), as opposed to recognizing a small number of classes (such as faces and cars [7, 10, 16]), or using a constrained setting (such as uniform backgrounds [9, 4]). Finally, when performing object detection and localization, we only use global image features. At the end of this paper, we discuss how our approach to detection can be combined with more traditional object localization methods, which make use of local features.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

(a) Isolated object (b) Object in context (c) Low-res Object

Figure 1. (a) A close-up of an object; (b) An object in context; (c) A low-res object out of context. Observers in our lab, addicts to coffee, have difficulties in recognizing the coffee machine in figure (c), however, they recognize it in figures (a) and (b).

2. Global image features

The regularities of real world scenes suggest that we can define features correlated with scene properties without having to specify individual objects within a scene, just as we can build face templates without needing to specify facial features. Some scene features, like collections of views [2, 17] or color histograms [12], perform well for recognizing specific places, but they are less able to generalize to new places (we show some evidence for this claim in Section 3.4). We would like to use features that are related to functional constraints, as opposed to accidental (and therefore highly variable) properties of the environment. This suggests examining the textural properties of the image and their spatial layout.

To compute texture features, we use a wavelet image decomposition. Each image location is represented by the output of filters tuned to different orientations and scales. We use a steerable pyramid [11] with 6 orientations and 4 scales applied to the intensity (monochrome) image. (Similar results are obtained using Gabor filters.) The local (L) representation of an image at an instant t is then given by the jet $v_t^L(x) = \{v_{t,k}(x)\}_{k=1,N}$, where $N = 24$ is the number of subbands.

We would like to capture global image properties, while keeping some spatial information. Therefore, we take the mean value of the magnitude of the local features averaged over large spatial regions:

$$m_t(x) = \sum_{x'} |v_t^L(x')| w(x' - x)$$

where $w(x)$ is the averaging window. The resulting representation is downsampled to have a spatial resolution of $M \times M$ pixels (here we use $M = 4$). Thus, m_t has size $M \times M \times N = 384$. We further reduce the dimensionality by projecting m_t onto the first D principal components (PCs) computed using a database of thousands of images collected with our wearable system. The resulting D-dimensional feature vector will be denoted by v_t^G. This representation proves to be rich enough to describe important scene context, yet is of low enough dimensionality to allow for tractable learning and inference.

Figure 2. Two images from our data set, and noise patterns which have the same global features. This shows that the features pick up on coarse-grained texture, dominant orientations and spatial organization.

Figure 2 illustrates the information that is retained using this representation with $D = 80$ PCs. Each example shows one image and an equivalent textured image that shares the same 80 global features. The textured images are generated by coercing noise to have the same features as the original image, while matching the statistics of natural images [8].

3. Place recognition

In this section we describe the context-based place recognition system. We start by describing the set-up used to capture the image sequences used in this paper. Then we study the problem of recognition of familiar places. Finally we discuss how to do scene categorization when the system is navigating in a new environment.

3.1 Wearable test bed

As a test-bed for the approach proposed here, we use a helmet-mounted mobile system. The system is composed of a web-cam that is set to capture 4 images/second at a resolution of 120x160 pixels (color). The web-cam is mounted on a helmet in order to follow the head movements while the user explores their environment. The user receives feedback about system performance through a head-mounted display.

This system allows us to acquire images under realistic conditions while the user navigates the environment. The resulting sequences contain many low quality images, due to motion blur, saturation or low-contrast (when lighting conditions suddenly change), non-informative views (e.g.,

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

a close-up view of a door or wall), unusual camera angles, etc. However, our results show that our system is reasonably robust to all of these difficulties.

Two different users captured the images used for the experiments described in the paper while visiting 63 different locations at different times of day. The locations were visited in a fairly random order.

3.2 Model for place recognition

The goal of the place recognition system is to estimate the most likely (discrete) location of the user given all the visual features up to time t. Let the place be denoted by $Q_t \in \{1, \ldots, N_p\}$, where $N_p = 63$ is the number of places, and let the global features up to time t be denoted by $v^G_{1:t}$. We can use a hidden Markov model (HMM) to solve the localization problem by recursively computing $P(Q_t|v^G_{1:t})$ as follows:

$$\begin{aligned} P(Q_t = q|v^G_{1:t}) &\propto p(v^G_t|Q_t = q)P(Q_t = q|v^G_{1:t-1}) \\ &= p(v^G_t|Q_t = q)\sum_{q'} A(q', q)P(Q_{t-1} = q'|v^G_{1:t-1}) \end{aligned}$$

where $A(q', q) = P(Q_t = q|Q_{t-1} = q')$ is the transition matrix (topological map) and $p(v^G_t|Q_t)$ is the observation likelihood, described below. [1]

We learn the transition matrix from labelled sequence data by counting the number of transitions from location i to location j. We add a Dirichlet smoothing prior to the count matrix so that we do not assign zero likelihood to transitions which did not appear in the training data.

We model the appearance of each location as a set of K views (a mixture of K spherical Gaussians). We set the variance, σ_p, and the number of mixture components, K, using cross-validation; we found $\sigma_p = 0.05$ and $K = 100$ to be the best. We could compute the means and mixture weights using EM. However, in this paper, we adopt the simpler strategy of picking the means to be prototypes (views), chosen uniformly from all views assigned to the location; weights are then set uniformly; the result is essentially a sparse Parzen window density estimator. In the future, we plan to investigate ways of picking the most informative (discriminative) prototypes, possibly based on entropy minimizing techniques such as those discussed in [14].

3.3 Performance of place recognition

In this section, we discuss the performance of the place recognition system when tested on a sequence that starts indoors (in building 400) and then (at frame $t = 1500$) moves outdoors. The test sequence was captured in the same way as the training sequences, namely by walking around the environment, in no particular order, but with an attempt to capture a variety of views and objects in each place. A qualitative impression of performance can be seen by looking at Figure 3 (top). This plots the belief state, $P(Q_t|v^G_{1:t})$, over time. We see that the system believes the right thing nearly all of the time. Some of the errors are due the inherent ambiguity of discretizing space into regions. For example, during the interval $t = 2100 : 2200$, the system is not sure whether to classify the location as "Draper street" or "Draper plaza". Other errors are due to poorly estimating the transition matrix. For example, just before $t = 1500$, there is a transition from "elevator 200/6" to "elevator 400/1", which never occurred in the training set. The Dirichlet prior prevents us from ruling out this possibility, but it is considered unlikely; hence the incorrect belief that the system is in "corridor 6a" just before $t = 1500$.

A more quantitative assessment of performance can be obtained by computing precision-recall curves. The recall rate is the fraction of frames which the system is required to

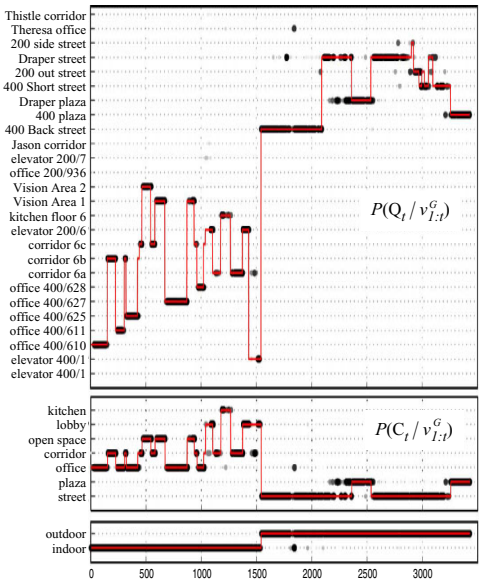


Figure 3. Performance of place recognition for a sequence that starts indoors and then (at frame $t = 1500$) goes outdoors. **Top**. The solid line represents the true location, and the dots represent the posterior probability associated with each location, $P(Q_t|v^G_{1:t})$, where shading intensity is proportional to probability. There are 63 possible locations, but we only show those with non-negligible probability mass. **Middle**. Estimated category of each location, $P(C_t|v^G_{1:t})$. **Bottom**. Estimated probability of being indoors or outdoors.

[1] Note that this use of HMMs is quite different from previous approaches in wearable computing such as [12]. In our system, states represent 63 different locations, whereas Starner et al. used a collection of separate left-to-right HMMs to classify approach sequences to one of 14 rooms.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

label (with the most likely location); this can be varied by adjusting a threshold, θ, and only labeling frames for which $\max_q P(Q_t = q|v^G_{1:t}) > \theta$. The precision is the fraction of frames that are labeled correctly.

The precision-recall framework can be used to assess performance of a variety of parameters. In Figure 4(a) we compare the performance of three different features, computed by subsampling and then extracting the first 80 principal components from (1) the intensity image, (2) the color image, and (3) the output of the spatially averaged steerable pyramid filter bank. We see that the filter bank works the best, then color and finally PCA applied to the raw intensity image.

In Figure 4(b), we show the effect of "turning the HMM off", by using a uniform transition matrix (i.e., setting $A(i,j) = \frac{1}{N_p}$). It is clear that the HMM provides a significant increase in performance (at negligible computational cost), because it performs temporal integration. We also compared to a more ad hoc approach of using the average of $\{p(Q_t|v^G_{t-W}), \ldots, p(Q_t|v^G_t)\}$, as was done in [14]. We found (by cross validation) that $W = 10$ works best, and this is what is shown in Figure 4(b); nevertheless, the HMM is much better. (Results for $W = 1$ (i.e., without any temporal averaging) are significantly worse (not shown).)

When using the HMM, we noticed that the observation likelihood terms, $b_t(q) = p(v^G_t|Q_t = q)$, often dominated the effects of the transition prior. This is a well-known problem with HMMs when using mixtures of high-dimensional Gaussians (see e.g., [1, p142]). We adopt the standard solution of rescaling the likelihood terms; i.e., we use

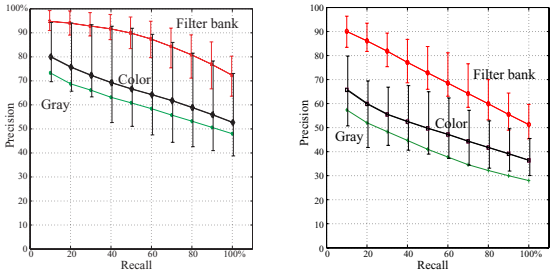
$$\tilde{b}_t(q) = \frac{p(v^G_t|Q_t = q)^{\gamma_p}}{\sum_{q'} p(v^G_t|Q_t = q')^{\gamma_p}}$$

where the exponent γ_p is set by cross-validation. The net effect is to "balance" the transition prior with the observation likelihoods. (It is possible that a similar effect could be achieved using a density more appropriate to images, such as a mixture of Laplace distributions.)

3.4 Categorizing novel places

In addition to recognizing the specific location, we would like the system to be able to categorize places into various high-level classes such as office, corridor, street, etc. There are several ways to do this. The simplest is to use the HMM described above, and then to sum up the probability mass assigned to all places which belong to the same category, e.g., the probability I'm in an office is P(in Antonio's office) + P(in Kevin's office) + P(in Bill's office). However, this will not work in new, unfamiliar environments.

Instead, we train an independent HMM on category labels, i.e., we estimate the category, $P(C_t|v^G_{1:t})$, independently of the place, $P(Q_t|v^G_{1:t})$. (Below we will see that



(a) HMM (b) No HMM

Figure 4. Precision-recall curves for different features for place recognition. The solid lines represent median performance computed using leave-one-out cross-validation on all 17 sequences. The error bars represent the 80% probability region around the median. The curves represent different features. From top to bottom: filter bank, color, monochrome (see text for details). **(a)** With HMM ($\gamma_p = 0.2$, $W = 1$, A = learned). **(b)** Without HMM ($\gamma_p = 1, W = 10$, A = uniform).

the category appearance model, $P(v_t|C_t)$, uses different features of v_t than the specific place appearance model, $P(v_t|Q_t)$.) We used 17 categories, including "office", "corridor", "street", etc. We also used the same methodology (a third, independent HMM) to classify the scene as indoor or outdoor.

We train the categorization system on outdoor and indoor sequences from building 200, and then test it on a sequence which starts in building 400 (not in the training), and then, at $t \simeq 1500$, moves outside (included in the training). The results are shown in Fig. 5. Before the transition, the place recognition system has a uniform belief state, representing complete uncertainty, but the categorization system performs well. As soon as we move to familiar territory, the place recognition system becomes confident again (the system is able to re-localize itself within the map).

We also computed precision recall curves to assess the performance of different features at the categorization task. The results are shown in Figure 6. Categorization performance is worse than recognition performance, despite the fact that there are fewer categories than places (17 instead of 63). There are several reasons for this. First, the variability of a class is much larger than the variability of a place, so the problem is intrinsically harder. Second, some categories (such as "open space" and "office") are visually very similar, and tend to get confused, even by people. Third, we have a smaller training set for estimating $P(C_t|C_{t-1})$, since we observe fewer transitions between categories than between instances.

Interestingly, we see that color performs very poorly at the categorization task. This is due to the fact that the color

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

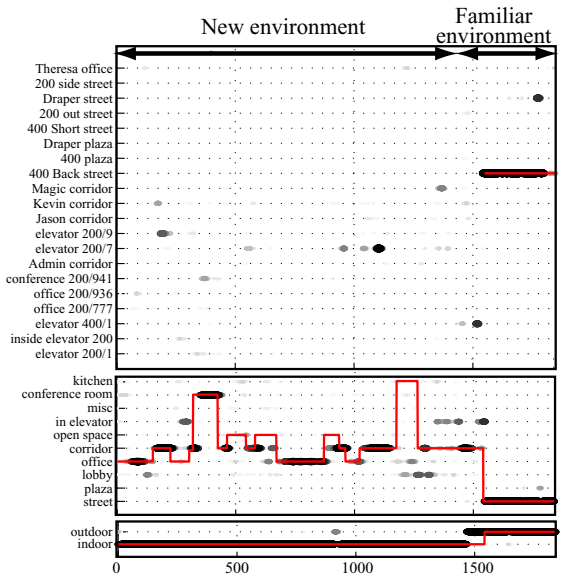


Figure 5. Place categorization when navigating in a new environment not included in the training set (frames 1 to 1500). During the novel sequence, the place recognition system has low confidence everywhere, but the place categorization system is still able to classify offices, corridors and conference rooms. After returning to a known environment (after $t = 1500$), performance returns to the levels shown in Figure 3.

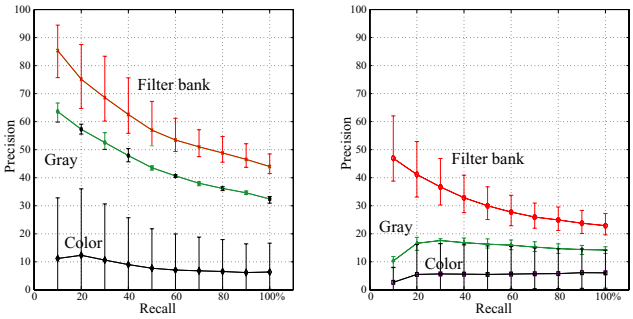


Figure 6. Precision-recall curves for categorization of non-familiar indoor environments. The curves represent different features sets. From top to bottom: filter bank, monochrome and color. Note that now color performs worse than monochrome, the opposite to Figure 4.

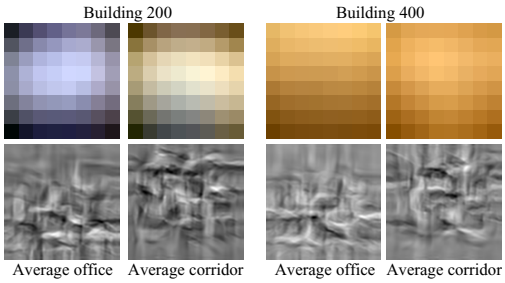


Figure 7. Average of color (top) and texture (bottom) signatures of offices and corridors for two different buildings. While the algorithm uses a richer representation than simply the mean images shows here, these averages show that the overall color of offices/corridors varies significantly between the two buildings, whereas the texture features are more stable.

of many categories of places (such as offices, kitchens, etc.) may change dramatically from one environment to the next (see Figure 7). The structural composition of the scene, on the other hand, is more invariant. Hence, although color is a good cue for recognition, it is not so good for categorization (with the exception of certain natural "objects", such as sky, sun, trees, etc., whose color is more constrained).

4. From scenes to objects

Most approaches to object detection and recognition involve examining the local visual features at a variety of positions and scales, and comparing the result with the set of all known object types. However, the context can provide a strong prior for which objects are likely to appear, as well as their expected size and position within the image, thus reducing the need for brute force search [15, 13]. In addition, the context can help disambiguate cases where local features are insufficient. In this paper, the context consists of both the global scene representation, v_t^G, the current location, Q_t, and/or the current place category, C_t. We show how we can use the context to predict properties of objects without even looking at the local visual evidence.

Let $O_{t,i}$ represent the attributes of all objects of type i in image v_t; these attributes could include the number of such objects (zero or more), their size, shape, appearance, etc. Let $\vec{O}_t = (O_{t,1}, \ldots, O_{t,N_o})$, where $N_o = 24$ is the number of object types considered here (bicycles, cars, people, buildings, chairs, computers, etc.). We can compute $P(\vec{O}_t|v_t, Q_t)$ as follows:

$$P(\vec{O}_t|v_{1:t}) \approx \sum_q P(\vec{O}_t|v_t, Q_t = q)P(Q_t = q|v_{1:t})$$

where we assume that knowing the current location, Q_t, renders previous frames, $v_{1:t-1}$, irrelevant. The second term is the output of the HMM, as discussed in Section 3. The first term can be computed using Bayes' rule:

$$P(\vec{O}_t|v_t, Q_t) \quad \propto \quad p(v_t|\vec{O}_t, Q_t)P(\vec{O}_t|Q_t)$$

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

$$\approx \prod_i p(v_t|O_{t,i},Q_t) \prod_i P(O_{t,i}|Q_t)$$

where we have assumed that the likelihood and prior are both fully factorized. This allows us to treat each object (type) independently. In particular, the posterior marginals can be computed as follows:

$$P(O_{t,i}|v_{1:t}) \propto \sum_q P(v_t|O_{t,i},q)P(O_{t,i}|q)P(q|v_{1:t}).$$

In order to compute $p(v_t|O_{t,i},Q_t)$, we have to make some approximations. A common approximation is to assume that the object's properties (presence, location, size, appearance, etc.) only influence a set of local features, v_t^o (a subset of v_t). Thus

$$p(v_t|O_{t,i}=o,Q_t=q) = p(v_t^o|o,q)$$

However, the global context is a very powerful cue that we want to exploit. Hence we include some global scene features, v_t^G (a deterministic function of v_t):

$$\begin{aligned} p(v_t|o,q) &= p(v_t,v_t^G|o,q) \\ &= p(v_t|v_t^G,o,q)p(v_t^G|o,q) \\ &\approx p(v_t^o|o,q,v_t^G)p(v_t^G|o,q) \end{aligned}$$

The first term, $p(v_t^o|o,q,v_t^G)$, can be approximated by $p(v_t^o|o,q)$ assuming that the object attributes o specify the object appearance (although this ignores the effect of some global scene factors, such as lighting). For this paper, we ignore the first term (i.e., the local appearance model $p(v_t^o|o,q,v_t^G)$), and focus on the second term, $p(v_t^G|o,q)$, which is related to the global context.

4.1 Contextual priming for object detection

In this section, we assume $O_{t,i}$ is just a binary random variable, representing whether any object of type i is present in the image or not. $P(O_{t,i}=1|v_{1:t}^G)$ can be used to do object priming, i.e., to decide which object detectors to run. It can also be used to rank-order images for image retrieval tasks in which we try to find images likely to contain instances of O_i. We can compute the probability an object type is present using Bayes rule:

$$P(O_{t,i}|v_t^G,Q_t=q) = \frac{p(v_t^G|O_{t,i},j)P(O_{t,i}|q)}{p(v_t^G|O_{t,i},q)P(O_{t,i}|q)+p(v_t^G|\bar{O}_{t,i},q)P(\bar{O}_{t,i}|q)}$$

We labeled a set of about 1000 images to specify whether or not each type of object was present. We estimated $P(O_i|q)$ by counting (analogous to estimating the HMM transition matrix). We model the conditional likelihood using another mixture of spherical Gaussians. We chose the means for $p(v_t^G|O_{t,i},Q_t=q)$ from the set of all images labelled as location q in which object i was present. Similarly, we chose the means for $p(v_t^G|\bar{O}_{t,i},Q_t=q)$ from the set of all images labelled as location q in which object i was absent. We chose the number of mixture components and the variance by cross validation.

Figure 8 shows the results of applying this procedure to the same test sequence as used in Figure 3. The system is able to correctly predict the presence of 24 different kinds of objects quite accurately, without even looking at the local image features. Many of the errors are "inherited" from the place recognition system. For example, just before $t = 1500$, the system believes it is in corridor 6a, and predicts objects such as desks and printers (which are visible in 6a); however, the system is actually in the floor 1 elevator lobby, where the only identifiable object is a red couch.

A more quantitative assessment of performance is provided in Figure 9, where we plot ROC (receiver operating characteristic) curves for 20 of the most frequent object classes. (This can be computed by varying a threshold θ and declaring an object to be present if $P(O_{t,i}=1|v_{1:t}^G) > \theta$; we then count compare the number of estimated positive frames with the true number. We did this for the same indoor-outdoor sequence as used in Figures 3 and 5.) The easiest objects to detect are things like buildings, which are almost always present in every outdoor scene (in this data set at least). The hardest objects are moving ones such as people and cars, since they are only present in a given context for a small fraction of the time.

4.2 Contextual priors for object localization

In this section, we try to predict the location of an object within the image. We can use this information e.g., to run an object detector only in the most probable image locations. One approach [15] is to model the expected location, $P(X_{t,i}|v_t^G,O_{t,i}=1)$, using cluster weighted regression (a variant on the mixtures-of-experts model). In this paper, we adopt a different approach: We divide the image into 80 non-overlapping patches, and represent the location using an 8×10 bit mask: $M_{t,i,l}=1$ iff any object of type i overlaps image region l, where $l \in \{1,\ldots,80\}$. This provides a crude way of representing location and size/shape.

Let $M_{t,i}$ be the whole image mask (an 80-dimensional bit vector). Since $P(M_{t,i,l}=1) = E[M_{t,i,l}]$, we can summarize the distribution in terms of its marginals using the expected mask. This can be computed as follows:

$$\begin{aligned} E[M_{t,i}|v_{1:t}^G] &= \sum_{o\in\{0,1\}} \sum_q P(O_{t,i}=o,Q_t=q|v_{1:t}^G) \\ &\quad \times E[M_{t,i}|v_t^G,Q_t=q,O_{t,i}=o] \end{aligned}$$

where $P(O_{t,i},Q_t|v_{1:t}^G)$ was computed by the object priming system discussed in Section 4.1. When the object is absent

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

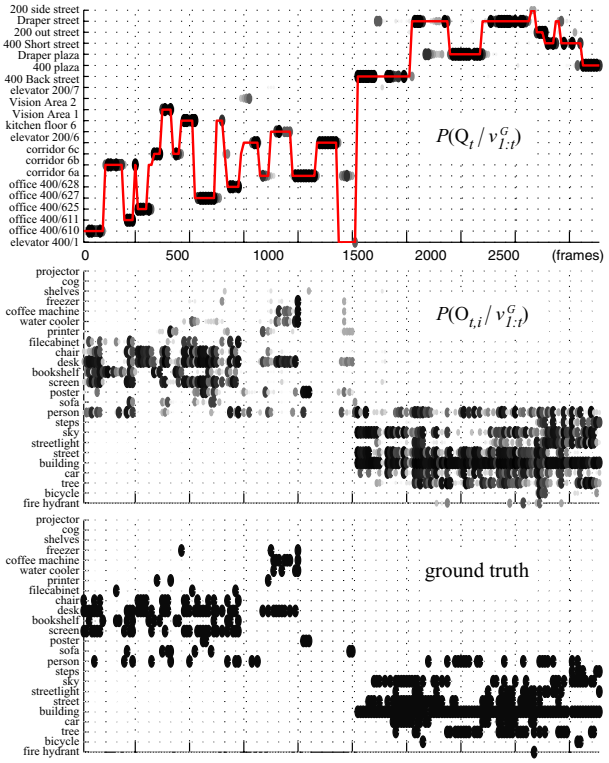


Figure 8. Contextual priors for object detection. We have trained the system to predict the presence of 24 objects. **Top**. The predicted place, $P(Q_t|v^G_{1:t})$ (the same as Figure 3). **Middle**. The probability of each object being present, $P(O_{t,i} = 1|v^G_{1:t})$. **Bottom**. Ground truth: a black dot means the object was present in the image. We only show results for the frames that have ground truth.

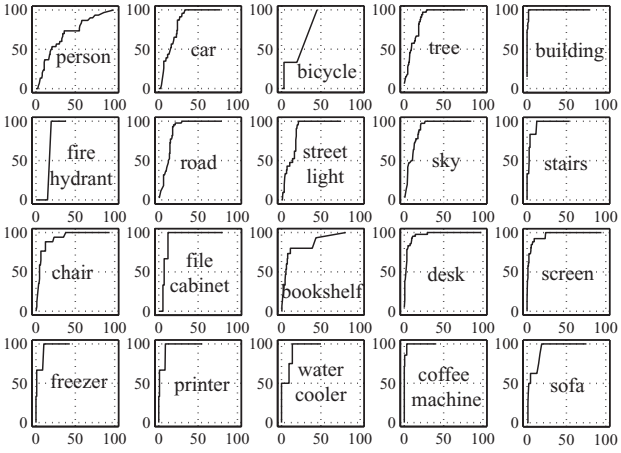


Figure 9. ROC curves for the prediction of object presence in the image. We plot hit rate vs false alarm rate as we vary the threshold on $P(O_{t,i} = 1|v^G_{1:t})$.

$(O_{t,i} = 0)$, we have $E[M_{t,i}|v^G_t, Q_t, O_{t,i} = 0] = \vec{0}$. If the object is present $(O_{t,i} = 1)$, the expected mask is given by

$$E[M_{t,i} = \vec{m}|v^G_t, Q_t, O_{t,i} = 1] = \sum_{\vec{m}} \vec{m} \ \frac{p(\vec{m}, v^G_t|Q_t, O_{t,i} = 1)}{p(v^G_t|Q_t, O_{t,i} = 1)}$$

We adopt a product kernel density estimator to model the joint on V^G_t and $M_{t,i}$:

$$p(\vec{m}, v^G_t|Q_t = q, O_{t,i} = 1) = \sum_{k=1}^{N_{i,q}} \frac{1}{N_{i,q}} K(\vec{m} - \mu^m_{k,i,q}) K(v^G_t - \mu^o_{k,i,q})$$

where $K(\vec{m} - \mu^m_{k,i,q}) = \delta(\vec{m}, \mu^m_{k,i,q})$ and $K(v^G_t - \mu^o_{k,i,q})$ is the same Gaussian kernel as used in the object priming system. Since the mask kernel is a delta function[2] we have

$$\sum_{\vec{m}} \vec{m} \ \ p(\vec{m}, v^G_t|Q_t = q, O_{t,i} = 1) = \sum_k \frac{1}{N_{i,q}} \mu^m_{k,i,q} K(v^G_t - \mu^o_{k,i,q})$$

Putting it all together, we get the intuitive result that the expected mask is a set of weighted prototypes, $\mu^m_{k,i,q}$,

$$E[M_{t,i}|v^G_t] = \sum_q \sum_k w_{k,i,q} \times \mu^m_{k,i,q}$$

where the weights are given by how similar the image v^G_t is to previous ones associated with place q and object i (as encoded by the $\mu^o_{k,i,q}$ prototypes), times the probability of being in place q and seeing object i:

$$w_{k,i,q} = \frac{K(v^G_t - \mu^o_{k,i,q}) \times P(Q_t = q, O_{t,i} = 1|v^G_{1:t})}{\sum_{k'} K(v^G_t - \mu^o_{k',i,q})}$$

We trained this model as follows. We manually created a set of about 1000 image masks by drawing polygons around the objects in each image. (The images were selected from the same training set as used in the previous sections.) We then randomly picked up to 20 prototypes $\mu^m_{k,i,q}$ for each location q and object i. A small testing set was created in the same way.

Some preliminary results are shown in Figure 10. The figure shows the probability of each object type appearing in each grid cell (the expected mask $E[M_{t,i}|v^G_{1:t}]$), along with

[2] We can use kernels with better generalization properties than a delta function. This can be done, for instance, by using other representations for $\vec{m}$ instead of a bit mask. We can model the distribution of masks as $p(\vec{m}) = p(f(\vec{m}))$ where f is a one-to-one mapping. For instance, f can be the function that converts a binary number to an integer. Then, we can use a gaussian kernel $G(f(\vec{m}) - f(\mu^m_{k,i,q}))$.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE

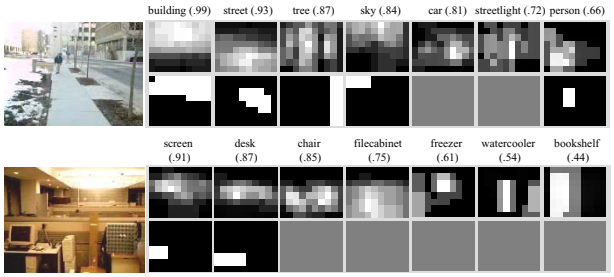


Figure 10. Some results of object localization. The gray-level images represent the probability of the objects being present at that location; the black-and-white images represent the ground truth segmentation (gray indicates absent object). Images are ordered according to $P(O_{t,i}|v^G_{1:t})$.

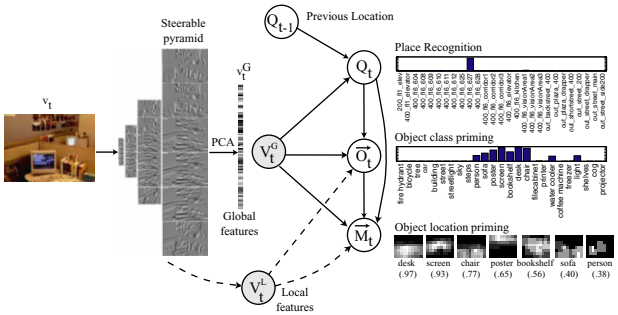


Figure 11. Summary of the context-based system. The dotted lines show how local features could be included in the model (not implemented in the system presented here).

the ground truth segmentation. In some cases, the corresponding ground truth image is blank (gray), indicating that this object does not appear, even though the system predicts that it might appear. Such false positives could be easily eliminated by checking the local features at that position. Overall, we find the results encouraging, despite the small nature of the training set.

5. Summary and Conclusions

We have shown how to exploit an holistic, low-dimensional representation of the image to perform robust place recognition, categorization of novel places, and object priming. A summary of the system is shown in Figure 11.

In current work, we are extending this system by combining the top-down prior provided by context with the results of a more standard bottom-up object recognition phase [5].

Acknowledgments

We thank Egon Pasztor (MERL) for implementing the image annotation tool, and Aude Oliva (MSU) for discussions. This work was sponsored by the Air Force under Air Force Contract F19628-00-C-0002, by the U.S. Department of Interior-Fort Huachuca under DARPA/MARS contract DABT 63-00-C-1012, and by the Nippon Telegraph and Telephone Corporation as part of the NTT/MIT Collaboration Agreement. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the U.S. Government.

References

[1] Y. Bengio. Markovian models for sequential data. *Neural Computing Surveys*, 2:129–162, 1999.

[2] M. O. Franz, B. Scholkopf, H. A. Mallot, and H. H. Bulthoff. Where did I take that snapshot? Scene-based homing by image matching. *Biological Cybernetics*, 79:191–202, 1998.

[3] J. Kosecka, L. Zhou, P. Barber, and Z. Duric. Qualitative image-based localization in indoors environments. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003.

[4] H. Murase and S. Nayar. Visual learning and recognition of 3-d objects from appearance. *Intl. J. Computer Vision*, 14:5–24, 1995.

[5] K. Murphy, A. Torralba, and W. Freeman. Using the forest to see the trees: a graphical model relating features, objects and scenes, 2003. AIM-2003.

[6] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *Intl. J. Computer Vision*, 42(3):145–175, 2001.

[7] C. Papageorgiou and T. Poggio. A trainable system for object detection. *Intl. J. Computer Vision*, 38(1):15–33, 2000.

[8] J. Portilla and E. P. Simoncelli. A parametric texture model based on joint statistics of complex wavelets coefficients. *Intl. J of Computer Vision*, 40:49–71, 2000.

[9] B. Schiele and J. L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *Intl. J. Computer Vision*, 36(1):31–50, 2000.

[10] H. Schneiderman and T. Kanade. A statistical model for 3D object detection applied to faces and cars. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2000.

[11] E. P. Simoncelli and W. T. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *2nd IEEE Intl. Conf. on Image Processing*, 1995.

[12] T. Starner, B. Schiele, and A. Pentland. Visual contextual awareness in wearable computing. In *Intl. Symposium on Wearable Computing*, pages 50–57, 1998.

[13] T. M. Strat and M. A. Fischler. Context-based vision: recognizing objects using information from both 2-D and 3-D imagery. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(10):1050–1065, 1991.

[14] A. Torralba and P. Sinha. Recognizing indoor scenes. Technical report, MIT AI lab, 2001.

[15] A. Torralba and P. Sinha. Statistical context priming for object detection. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 763–770, 2001.

[16] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision - to appear*, 2002.

[17] J. Wolf, W. Burgard, and H. Burkhardt. Robust vision-based localization for mobile robots using an image retrieval system based on invariant features. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2002.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03)
0-7695-1950-4/03 $ 17.00 © 2003 IEEE