

Exploring the Relationship Between Context and Pose: A Case Study

Itai Katz, Hamid Aghajan
Stanford University
Stanford, CA 94305
{itai, aghajan@stanford.edu}

Abstract

While context has received little attention in the visual object classification literature, it nevertheless plays a vital role in the ability to identify objects in a scene. This paper seeks to improve the performance of object classifiers by incorporating contextual information. Our method uses probability maps to guide classifiers to image regions likely to contain the object in question, based on the object's past positions and the positions of surrounding objects. We contrast our method with a baseline unguided classifier and show that using probability maps as a preprocessing step significantly reduces the number of positions a classifier needs to evaluate. The structures presented here can be used with any classification algorithm that evaluates windowed image regions.

Introduction

The ability to automatically identify objects in an image is one of the fundamental topics driving computer vision research. While classifiers have demonstrated remarkable success in constrained domains such as faces ([Viola], [Schniderman], [Gu], [Heisele], [Hjelmas], [Yang]), their performance still lags far behind human capabilities when given arbitrary scenes. We believe one reason for this discrepancy is due to the lack of context in the traditional classification framework.

Until recently, computer vision has drawn heavily from signal theory. In general, classifiers have relied on local, low-level image statistics without considering surrounding objects. Natural images (i.e., those without content constraints) have a great deal of context which can be used to infer particular characteristics of objects in the scene. When multiple classifiers are run sequentially on the same image, they operate independently of one another. Intuitively however, we know that certain objects tend to be present together (e.g., spoons and forks, or tables and chairs), and are linked by well-defined spatial relationships. In this paper we propose a framework for incorporating knowledge of previously detected objects into object classifiers.

Background

The general structure for an object classifier is as follows: In the learning phase, a collection of positive and negative training examples is used to

generate a series of representative features. Features can consist of values taken directly from an image (such as a histogram of pixel intensities) or an abstraction (such as Haar-like wavelets [Papageorgiou]). During the test phase, a subwindow is selected from the query image. This subwindowed region is assigned a score in proportion to its correlation with the learned features. This region is then shifted and the process is repeated until the entire image has been evaluated. Image regions with a score above a given threshold are said to contain the object in question.

The problem with this sort of exhaustive search is twofold:

- 1) When the distribution of objects in an image is known to be non-uniform, searching in low-probability regions is extraneous. For applications requiring multiple classifiers or applications using low-throughput, embedded hardware, classification speed rapidly becomes a computational bottleneck.
- 2) An exhaustive search creates opportunities for false detections in regions that are unlikely to contain the object of interest.

An exploration of the biological basis of object detection in [Itti] proposes that humans employ a hybrid bottom-up/top-down attention model. Input images are broken down into feature maps which encode low-level observations such as color, intensity, and gradient orientation. The feature maps are then combined into a single saliency map. In parallel, a database of acquired knowledge directs the focus towards areas that provide maximum information gain.

Empirical results from [Davenport] find that humans can identify objects more accurately when placed in a semantically consistent setting. In functional MRI (fMRI) studies, cortical regions were found that regulate processing of contextual relationships between pairs of objects [Bar].

Recent implementations inspired by physiological models have focused on the attention mechanism. [Gould] presents a system that combines wide-angle and telephoto cameras to simulate peripheral and foveal vision. A fixed, wide-angle camera provides a low-resolution overview of

the scene, while a PTZ-mounted telephoto lens hops between regions of interest. [Orabona] combines feature maps to form “salient regions” which detect proto-objects for intelligently guided object classifiers.

[Torralba] describes a method to improve classification accuracy which models three terms: object appearance, object spatial distribution, and the likelihood of an object given a particular scene category. The first of these corresponds to the well-known low-level features, while the latter two comprise the higher-order knowledge. The three terms are multiplied to form a single probability function. [Hotz] uses a different approach to incorporate context. Low- and high-level analyses are performed separately and communicate through a feedback loop. A hypothesis module generates predictions on object positions, which is passed into an AdaBoost-trained classifier. These results are in turn used to update the hypothesis.

Approach

The scanning method used by the majority of classifiers calls for sliding the subwindow through the image in a linear (left-to-right, top-to-bottom) pattern. As the two problems outlined above suggest, the order that image regions are evaluated could instead be guided by the spatial distributions of the objects being searched for. Scanning the most likely regions first allows the classifier to simultaneously reduce the number of evaluations and the number of false positives. To accomplish this we guide classifiers with the aid of probability maps, two-dimensional structures that encode the likelihood of finding an object at a given image coordinate. After computing the maps, an ordered list of image coordinates is generated, sorted by probability. The classifier first evaluates a subwindow in the most probable coordinate and works down the list, possibly terminating early if the probability drops below a predetermined threshold.

The motivation for incorporating object statistics as a preprocessing step in the form of maps is to leverage the existing power of classifiers. Since our method makes use of higher-order knowledge, it is natural to apply this contextual information as a layer above the low-level classifier, rather than proposing a new, monolithic algorithm. A corollary benefit is that our method works independently of the underlying classification algorithm and thus can remain useful as more powerful algorithms are developed.

Object-based Probability Maps

In its simplest form, a probability map for an object o is merely the PDF of the object’s spatial location. We call this function an object-based probability map, since it is dependent only on

statistics of the object alone, without considering additional contextual cues. This distribution is modelled as a sum of two-dimensional conditional Gaussians distributions:

$$P(\bar{x} | o) = \sum_{i=1}^N b_i G(\bar{x} | o; \mu_i, \sigma_i)$$

where \bar{x} is a vector indicating an image location $\{x,y\}$ in pixels, o is the object class, and N is the number of components. To train the model, we collect images containing object o and hand-label the objects’ centers in each image. These points are passed into a standard EM (Expectation-Maximization) algorithm which finds the Gaussian parameters (mean vector μ_i , covariance matrix σ_i and weighting coefficients b_i) that best describe the training data. For a comprehensive description of the procedure, the reader is directed to [Gershenfeld]. The algorithm is repeated 1000 times for $N = \{1\dots5\}$, each time with a new set of randomized initial parameters. The iteration with highest log-likelihood is retained.

An example of a probability map for pedestrians is shown in fig. 1. To generate the map, we took 800 hand-segmented images from the MIT LabelMe database [Russell] that contains varying numbers of pedestrians. These images represent a diversity of natural images in multiple scales and camera angles. The dataset was split evenly amongst training and testing sets. For the purposes of normalization, images and annotation data were scaled to 100^2 pixels.

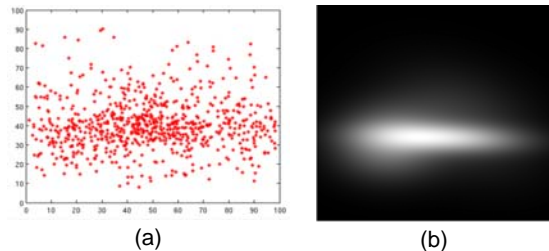


Figure 1: Pedestrians exhibit a characteristic distribution. (a) centroids of pedestrians from a dataset containing 800 images. Units are in image pixels. (b) the resulting object-based probability map. For this example, the log-likelihood is maximized for $N=3$.

It is interesting to note that even without any constraints on the scene type or object distance, significant structure is apparent in an object’s distribution. For example, pedestrians tend to be clustered in a narrow band towards the bottom of the image. Presumably this is a result of the images being taken from similar vantage points: at an average adult height with the camera held level.

Although the method described here is straightforward, it encodes global characteristics about objects that would be difficult for an operator to input manually.

A rough estimation of efficacy can be found by noting the percentage of the search space needed to find a given percentage of targets in the test set. Table 1 below shows how well the pedestrian probability map characterizes the dataset. The results for the object-based scanning path are compared with the standard linear scanning path. More detailed results are shown in fig. 2.

Object-based Probability Map Efficacy

% Targets found	% Image evaluated	
	Object-based	Standard (linear)
0.50	0.13	0.61
0.75	0.27	0.68
0.90	0.44	0.75
0.99	0.74	0.88

Table 1: A comparison of object-based and standard scanning patterns.

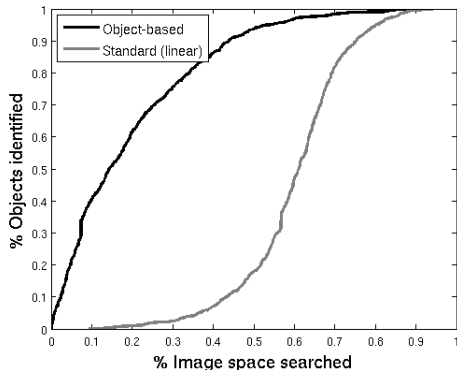


Figure 2: Efficacy curves for two scanning patterns.

Scene-based Probability Maps

The method described above relies exclusively on object models. This approach is well suited for targets that have highly constrained positions with few or no outliers. Most real-world objects, however, exhibit enough positional variance so that an object-based probability map fails to characterize atypical positions (fig. 3[c]). A naïve solution would be to simply multiply the covariances of the constituent Gaussian distributions by a constant factor, thereby increasing the size of the likely region to incorporate improbable, but possible positions (fig. 3[d]). This solution is undesirable, however, since the effectiveness of guided classifiers decreases as the size of the likely region increases. As an extreme

example, consider the degenerate case where the probability map is uniform—the resulting scanning path would be linear.

We seek to improve on the object-based model by reducing the size of the likely region while simultaneously providing better localization. To accomplish this, we augment the object-based probability map with semantic scene constraints. Many object pairs are often co-present in a scene and typically these pairs are constrained to a particular spatial relationship. Trees are usually above pedestrians, for example. If the position of one object is known a priori (the “source”), we can delineate a region that is likely to contain its pair (the “target”). Mathematically, this is equivalent to evaluating the expression:

$$P(\bar{x}_t | \bar{x}_s, o) = \sum_{i=1}^N b_i G(\bar{x}_t | \bar{x}_s, o; \mu_i, \sigma_i)$$

for target position \bar{x}_t , source position \bar{x}_s and target class o . We call this structure a scene-based probability map, since it considers object history as well as its context in the current image. To generate this model, a database of images containing the source and target is collected, and the object centers are recorded. When given a query image, computing the probability maps consists of the following steps:

- 1) Find \bar{x}_s using object-based probability maps or a secondary method
- 2) Construct a constraint vector \bar{c} whose elements correspond to the features in \bar{x}_s
- 3) Identify a subset of source objects \bar{x}_{si} from the database that are close in feature-space to \bar{x}_s :
$$\left(\bar{x}_s - \frac{1}{2} \cdot \bar{c} \right) < \bar{x}_{si} < \left(\bar{x}_s + \frac{1}{2} \cdot \bar{c} \right)$$
- 4) Map each point \bar{x}_{si} into its corresponding target point, \bar{x}_{ti}
- 5) Using the EM algorithm, find a best-fit distribution $P(\bar{x}_t | \bar{x}_s, o)$ over the points \bar{x}_{ti}

The resulting distribution is the scene-based probability map.

Of possible concern is that this method is dependent on having at least one object, \bar{x}_s , already identified. One solution, mentioned above, is to locate \bar{x}_s without context, using an object-based probability map. As each identified object in a

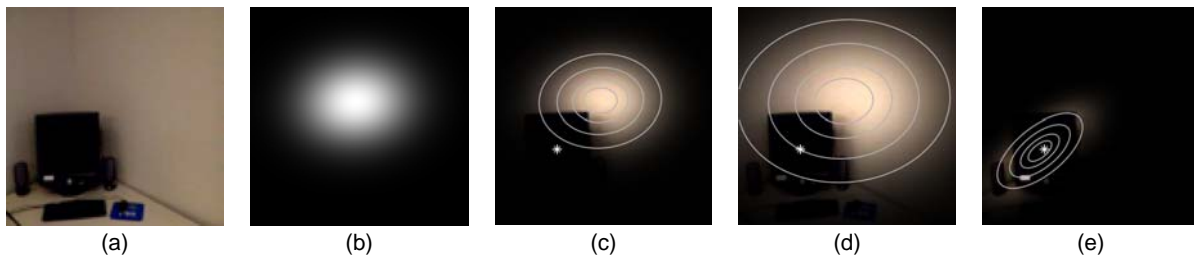


Figure 3: In unconstrained, natural images, the positions of some objects are poorly characterized by a single distribution. (a) Monitors are one such example where the degree of positional variance is high. When an object-based probability map is computed (b) and overlaid atop the monitor image (c), the mean of the distribution lies far from the monitor center (denoted by an astrisk). Simply expanding the distribution to include this outlier (d) is inefficient as it results in a higher number of high-probability coordinates to search. Computing a scene-based probability map based on the keyboard position (e) localizes the monitor well without increasing the distribution size

scene provides additional information, only the first few objects will have to be located without the benefit of context. It should also be noted that some objects are significantly easier to classify than others. Objects which are viewpoint-invariant or have constant color distributions can often be found through simpler means. These can in turn be used as source objects paired with more difficult to classify targets. Additionally, in applications where the camera is fixed and the source objects immobile (sidewalks, trees, tables, etc.), the positions of source objects can be specified manually.

We investigate this method by choosing a motivating example of a keyboard and monitor for the source and target objects, respectively. We choose these objects for this example because of the rigidity of their relationship. The success of using spatial relationships to constrain and localize the likely region depends on the repeatability of the objects' joint pose. From experience, we know that monitors are typically found directly above their corresponding keyboards (see fig. 4).

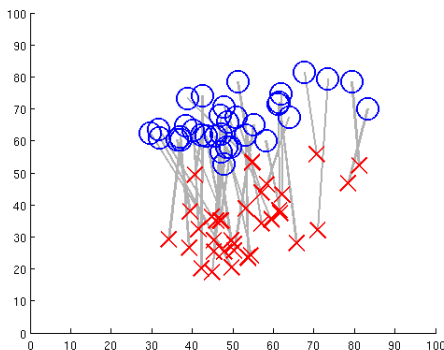


Figure 4: Keyboards (crosses) and monitors (circles) have a consistent spatial relationship. For clarity, only a subset of observations are shown.

We collected 665 images from the LabelMe database that contain a single instance of a keyboard and monitor pair, normalized to 100^2 pixels. The dataset was split equally amongst training and testing sets. The constraint vector was set to:

$$\bar{c} = \begin{bmatrix} width \\ height \end{bmatrix} = \begin{bmatrix} 10 \\ 10 \end{bmatrix}$$

Examples of the resulting maps are shown in figure 5.

Scene-based probability maps generate smaller, more localized distributions than object-based probability maps. To measure distribution size, we find the percentage of the map that contains 90% of the distribution. Note that object-based probability maps are static, whereas scene-based probability maps are different for each test image. In this case an average distribution size is taken. Localization error is measured by computing the squared distance between the monitor centers and the centroid of the distribution, averaged over the set of images in the test set. A theoretical score of zero indicates the likely region is perfectly centered on the monitor for each image.

Probability Map Performance

Type	Localization error (pixels ²)	Distribution size (% of image)
Object-based	323.4	0.31
Scene-based	166.9	0.11

Table 2: A comparison of object-based and scene-based probability maps

Additional Contextual Cues

Source position can be combined with other source object attributes to further reduce the likely region. This is achieved simply by adding an additional feature to the source object \bar{x}_s and an additional constraint to \bar{c} . For the keyboard-monitor pair, we

tested the effect of pairing the keyboard position with the area of its bounding polygon. Experimentation found that the best results were achieved when the area of database source objects were restricted to being +/- 50% the area of the query source object. The result of including area as a contextual cue reduced localization error to 125.4 and reduced the distribution size to 0.066. Including area is reasonable, since due to the perspective effect, source-target distance is proportional to the source area.

Conclusion and Future Work

A great deal of context can be extracted from natural scenes, even when no constraints are placed on the scene type or viewing angle. In this paper we have presented a framework for incorporating this context to guide classifiers using object distributions. In addition to performance benefits, our method has the advantage of conceptual simplicity and independence from the underlying classifier.

We can improve on our method in a

number of ways: 1) Incorporate semantic scene type as an additional contextual cue. Are we looking at a kitchen? A construction site? 2) Consider multiple related objects. A keyboard and mouse together, for instance, may be better able to localize a monitor than a keyboard alone. 3) Allow negative relationships between objects. For many pairs of objects, their presence is mutually exclusive. 4) Take advantage of existing linguistic databases. Projects such as WordNet [Fellbaum] provide a wealth of information on how objects relate to one another with detail that cannot be extracted from images alone (e.g., semantic equivalence of two objects).

Acknowledgements

This research has been financially supported through the National Institute of Standards and Technology (NIST STRS project #08-86104-4102) with additional support through the Precourt Institute for Energy Efficiency. The authors would also like to thank Prof. John Haymaker, Dr. Kam Saidi, Alan Lytle, and Nick Scott for their valuable discussions.

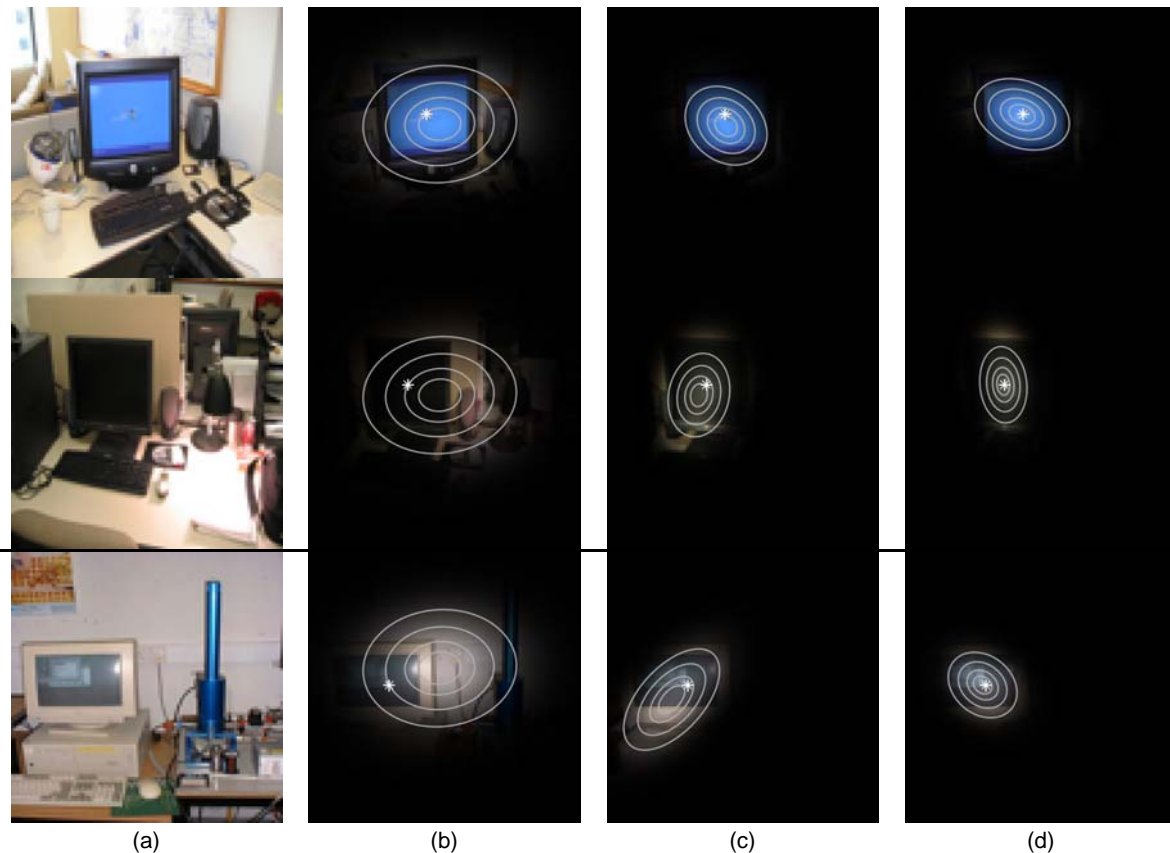


Figure 5: In office scenes, object-based probability maps have moderate success (b) in characterizing monitor centers (denoted by an asterisk). By incorporating knowledge of the keyboard position (c), the likely region is smaller and better localized. Including knowledge of keyboard area (d) further improves performance. The contour lines represent probabilities of 0.25, 0.50, 0.75, and 0.90, respectively.

References

- (1) Schneiderman, H. and Kanade, T. A statistical method for 3D object detection applied to faces and cars. *International Conference on Computer Vision*, 2000.
- (2) Viola, P. and Jones, M. Rapid object detection using a boosted cascade of simple features. *Proceedings, IEEE Conference on Computer Vision*, 2001.
- (3) Gi, L., Li, S. Z., and Zhang, H.J. Learning probabilistic distribution model for multiview face detection. *Proceedings, IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
- (4) Heisele, B., Serre, T., Pontil, M., and Poggio, T. Component-based face detection. *Proceedings, IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
- (5) Hjelmas, E. and Low, B.K. Face detection: A survey. *Comput. Vis. Image Understand.* 83. 236-274, 2001.
- (6) Yang, M.H., Kriegman, D., and Ahuja, N. 2002. Detecting face in images: A survey. *IEEE Trans. Patt. Anal. Mach. Intell.* 24, 34-58, 2002.
- (7) C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *International Conference on Computer Vision*, 1998.
- (8) Itti, Koch. Computational Modelling of visual attention, 2001. *Nature*.
- (9) Torralba, A. Contextual modulation of target saliency. *Advances in Neural Information Processing 14*, 2002.
- (10) Gould S., et al. Peripheral-foveal vision for real-time object recognition and tracking in video. *Intl. Joint Conference on Artificial Intelligence*, 2007.
- (11) Orabona, F., Metta, G., and Sandini, G. Object-based visual attention: A model for a behaving robot. *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
- (12) B. C. Russell, A. Torralba, K. P. Murphy, W. T. Freeman, LabelMe: a database and web-based tool for image annotation. MIT AI Lab Memo AIM-2005-025, September, 2005.
- (13) Gershenfeld, N. *The nature of mathematical modeling*. Cambridge University Press, 1999.
- (14) Fellbaum, C., *WordNet: An Electronic Lexical Database*. MIT Press, 1998.
- (15) Hotz, L.; Neumann, B.; Terzic, K.; Sochman, J. Feedback between Low-Level and High-Level Image Processing. TR FBI-B-278/07, Department of Informatics, University of Hamburg, 2007