International Journal of Computer Vision
© 2006 Springer Science + Business Media, LLC. Manufactured in The Netherlands.
DOI: 10.1007/s11263-006-7538-0

Short Paper
A Critical View of Context

LIOR WOLF AND STANLEY BILESCHI*
The Center for Biological and Computational Learning, Massachusetts Institute of Technology, Cambridge, MA 02139
liorwolf@mit.edu
bileschi@mit.edu

Received July 14, 2005; Revised January 17, 2006; Accepted January 31, 2006

First online version published in April, 2006

Abstract. In this study, a discriminative detector for object context is designed and tested. The context-feature is simple to implement, feed-forward, and effective across multiple object types in a street-scenes environment.

Using context alone, we demonstrate robust detection of locations likely to contain **bicycles**, **cars**, and **pedestrians**. Furthermore, experiments are conducted so as to address several open questions regarding visual context. Specifically, it is demonstrated that context may be determined from low level visual features (simple color and texture descriptors) sampled over a wide receptive field. At least for the framework tested, high level semantic knowledge, e.g, the nature of the surrounding objects, is superfluous. Finally, it is shown that when the target object is unambiguously visible, context is only marginally useful.

Keywords: context, learning, streetscenes, object detection, scene understanding

1. Introduction and Related Work

In object detection, the goal is to locate and identify all instances of a particular object class within an image, i.e., finding all the cars in a snapshot of a street. There exists a dichotomy of useful information sources for this task: appearance and context. Appearance information includes patterns of brightness, edge responses, color histograms, texture cues, and other features commonly used for object detection. The notion of *contextual-features* is somewhat loosely defined, encapsulating at once the nature of nearby objects (Murphy et al., 2003; Strat and Fischler, 1991; Hanson and Riseman, 1978; Carbonetto et al., 2004), the relative position and scale of those objects (Singhal et al., 2003; Kruppa and Schiele, 2003; Fink and Perona, 2003; Bergboer et al., 2003; Haralick, 1983; Bileschi and Heisele, 2003), as well as statistics of low level visual features of the scene as a whole (Torralba and Sinha, 2001). A good definition of context might be *information relevant to the detection task but not directly due to the physical appearance of the object*. For example, knowledge of the location of a road may influence the detection of cars. In general, in natural images, objects are strongly expected to fit into a certain relationship with the scene, and context gives access to that relationship.

Context is presumed to be an important cue for object detection in humans, believed to reduce processing time and to help disambiguate low quality inputs by mitigating the effect of clutter, noise and ambiguous inputs. Biederman et al. (1982) shows that humans detecting objects that violate their standard context take longer and make more errors. Functional MRI evidence

*Both authors contributed equally.

of humans using contextual cues was provided by Cox et al. (2004) and Bar et al. (2005), among others. If the human organism has evolved structures for processing visual context, then it is likely that context information is a path to efficient understanding of the natural visual world. Therefore, *synthetic* vision systems may also benefit from context.

Previous context-enabled systems may be grouped into three sets: systems which share information via a network of object-detectors, systems which classify the scene as a whole, and systems which employ a graphical model over a segmentation of the image. As an example of a system from the first set, Torralaba et al. (2004) employ boosting and graphical networks to learn associations between the likely co-occurrence and relative position of objects. Fink describes a similar system (Fink and Perona, 2003) for which detections of objects' parts, as well as detections of other objects in a scene, are employed in a modification of the Viola-Jones cascaded object detection architecture (Viola and Jones, 2001). This type of dense composition leaves no information source untapped, but the downside of such structures is that any mutual dependencies must be computed in an iterative fashion, first updating one object then the other. Moreover, if the target object is the only labeled object in the database then there are no sources of contextual information. Systems which pre-segment the image and then model the relationships between neighboring segments, i.e. (Kumar and Hebert, 2003; Carbonetto et al., 2004), suffer from similar issues.

Mutual dependance is not a problem for systems which use context by processing the scene as a whole, without first detecting other objects. Murphy et al. (2003) employs context as a scene 'gist', which influences priors of object existence and global location within that scene. The disadvantage here is that the scene must be taken as one complete unit and spatially localized processing can not take place.

Some researchers believe that context only makes sense in a generative framework, using for example random fields or graphical models of context and focusing on the expected spatial relationships between objects. Even assuming that the world is best described by such a hierarchal generative process, there is no reason to believe that accurate and useful classifiers can not be built using a discriminative framework. This is one of the main results of Vapnik's statistical learning theory (Vapnik, 1999).

The system for context recognition described in this work is simple to implement and feed-forward. It uses the relative positions of other detected objects in the scene as well as low-level cues such as global positions, colors and textures to build a map of the contextual support for the target object. The internals of this algorithm are detailed in Section 3. For the sake of clarity and ease of understanding we have opted to discuss the architecture of the system in the context of our particular implementation, rather than in the general form. The data for our experiments will be drawn from our StreetScenes database, a database of hand-labeled images taken from the streets of Boston and the surrounding areas, as shown in Fig. 1.

2. Goals

The primary goal of this work is to suggest a simple feed-forward context feature and to demonstrate its effectiveness across a variety of object types. By coupling a detector for each such object's appearance with a detector for that same object's *context*, we will support previous studies (Torralaba et al., 2004; Carbonetto et al., 2004) which show that, for at least the objects tested in our street-scenes database, context does aid in detection performance, especially when the appearance information is weak.

Furthermore, using this model, we explore some relevant issues regarding context. For instance, we address whether low-level context (derived from visual early features like wavelet values) is as powerful as high-level context (derived from semantic information like the presence of other objects). Previous systems have used one or the other, but this is the first direct comparison of the two.

The implications of this research question may be far reaching. If little or no benefit is gained by designing a system based on high level information, then context is nothing more than an additional classification feature, perhaps with a larger *receptive field,*[1] and can be computed in a feed forward manner. If, instead, context is heavily dependant on high level information, then robust context-enabled object detection systems may be limited to computation structures involving some form of feedback.

We will also show that the utility of context information is related to the difficulty of the detection problem. If it is very difficult to discern the target object from the background, perhaps due to occlusion or low

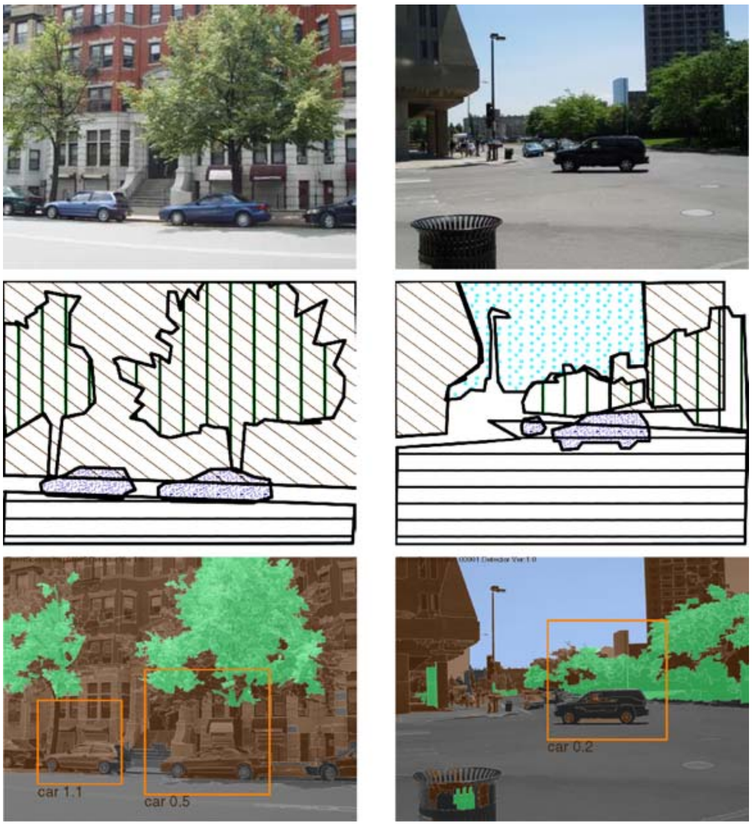

Figure 1. Top: StreetScenes database examples. Middle: Hand labeled semantic layer for the images above. Each StreetScenes example includes polygons and labels for 9 object categories. In these images different fill patterns denote different object labels. 5 object types are shown: cars, buildings, trees, roads, and skies. Bottom: Output of our system detecting amorphous objects and rigid objects using the context.

resolution images, then visual context can be helpful, but when the target object is unambiguously visible the context is only marginally useful, suggesting that the context information is highly redundant to the appearance information.

Note that, regarding these investigations, we make several assumptions. Firstly, in the experimental investigation, all tests use a database of images of street scenes, and the utility of context is measured via a detection task using three separate objects within this database. We assume that the contextual relationships of these objects in these types of scenes extend different objects to other scenarios. Furthermore, all studies are performed using the same general framework for the contextual feature. The context feature captures semantic and/or low-level visual information sampled in a pre-determined spatial pattern. This feature is very similar in style to features which have been successful for describing object appearances for detection algorithms, and is an obvious extension thereof. All claims we make about context can only be supported in so far that this feature captures well the available context information. If there is pertinent, computable context information which this feature does not express to the subsequent learning machine, then the results of these experiments do not adequately answer our stated inquiries into the nature of context. It is hoped that this work will serve to tie together some of the disparate notions of context and serve as a resource to those who are interested in perhaps adding contextual level understanding to an object detection system.

3. A Feed-Forward Context Feature

Our system eschews complex models in lieu of a fast, simple, feed-forward classifier. We favor the discriminative approach to the learning problem, and as such are compelled to learn both relative and absolute geometric models in such a framework. We show that this is an easy problem when using the right type of features.

The construction of the context feature is best understood as a two stage process. 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.

3.1. *Computing Low Level Image Features*

To produce low-level features we first downsize the input image to 60×80 pixels and compute color and texture spaces as in the Blobworld system of Carson et al. (1998). This resolution was selected as it was large enough to capture well the gross pattern of texture-objects (buildings, trees, roads, etc.). Note that we do not employ the subsequent clustering-based segmentation stage of Blobworld. Blobworld returns a new image represented with six layers of information, three for color, and three for texture. The color space used is the well documented CIE LAB space. LAB color is designed to model color dissimilarity as seen by humans as euclidian distance in the color space. The texture layers capture information about the local structure of the brightness gradient. The first texture layer is referred to as the polarity, and measures the likelihood of the gradient to switch direction. In a sense it discriminates between boundaries of brightness and distributed textures. The second layer is the anisotropy, which is roughly a measure of the relative strength of the gradient in orthogonal directions. The third layer of texture information is the texture contrast, which can be seen as a way to measure the roughness or harshness of the region.

In addition to the 6 color and texture features, we also include 10 features to represent the global position. These position features are calculated at every pixel p_i by recording the distance from p_i to a set of 10 predefined locations, roughly evenly distributed over the image. This representation was chosen in order to make it possible for a classifier, even a simple linear one, to learn a wide variety of position priors. For instance, were just the x and y recorded in the feature, it would be impossible for a linear classifier to prefer points near the center of the image over points near the borders.

3.2. *Semantic Image Features*

In order to investigate the importance of high-level features in constructing context cues, we include several semantic image features. For our task of detecting cars, pedestrians, and bicycles, we add four semantic layers indicating the presence of buildings, trees, roads, and skies. For instance, in the building feature, a pixel with a value of 1 indicates that this pixel is over a building, and a value of 0 indicates that it is not. Because of the coarse labeling and ambiguous border cases, a pixel may be given multiple labels, i.e., it may be both building and tree, or it may have a null label. The ground truth for these four layers is available from the hand labeled StreetScenes images. Figure 2 illustrates some examples of these labels.

Since the ground truth semantic information is not available in test images, four binary support vector machine (SVM) classifiers were trained to automatically detect these categories. Their training set of 10,000 samples per category was extracted from 100 training images. Generating the semantic features for a novel test image involves first computing the low-level feature image, and then feeding this data into the four SVMs. See Fig. 2 for learned semantic labeling. Measures of the performance of these four classifiers are available in Fig. 4.

3.3. *Building the Context Feature*

At this point, the original image has been converted into an image with 20 layers of information. Each pixel p_i is labeled with 4 binary semantic features, 3 color features, 3 texture features, and 10 global position features. However, the context feature for p_i must hold information representing not only the immediate proximity, but also the larger neighborhood. This information is captured by sampling the data at 40 predetermined locations relative to p_i, as shown in Fig. 3. The relative positions are arranged in a quasi-log-polar fashion so as to sample the local neighborhood more densely than distant regions. This is similar to biological systems, such as the mammalian retina, and several computer vision systems, e.g., (Belongie et al., 2002). Specifically, data is sampled at 5 radii of 3, 5, 10, 15, and 20 pixels, and at 8 orientations. These 40 samples are concatenated to generate an 800 dimensional context feature vector for each pixel. Note that the 20 dimensional image is smoothed first by averaging over a 5×5 window.

While it may seem that computing semantic-level information from the low-level information, and then including both is an exercise in redundancy, we should point out that this is not the case. Reductio ad absurdum,

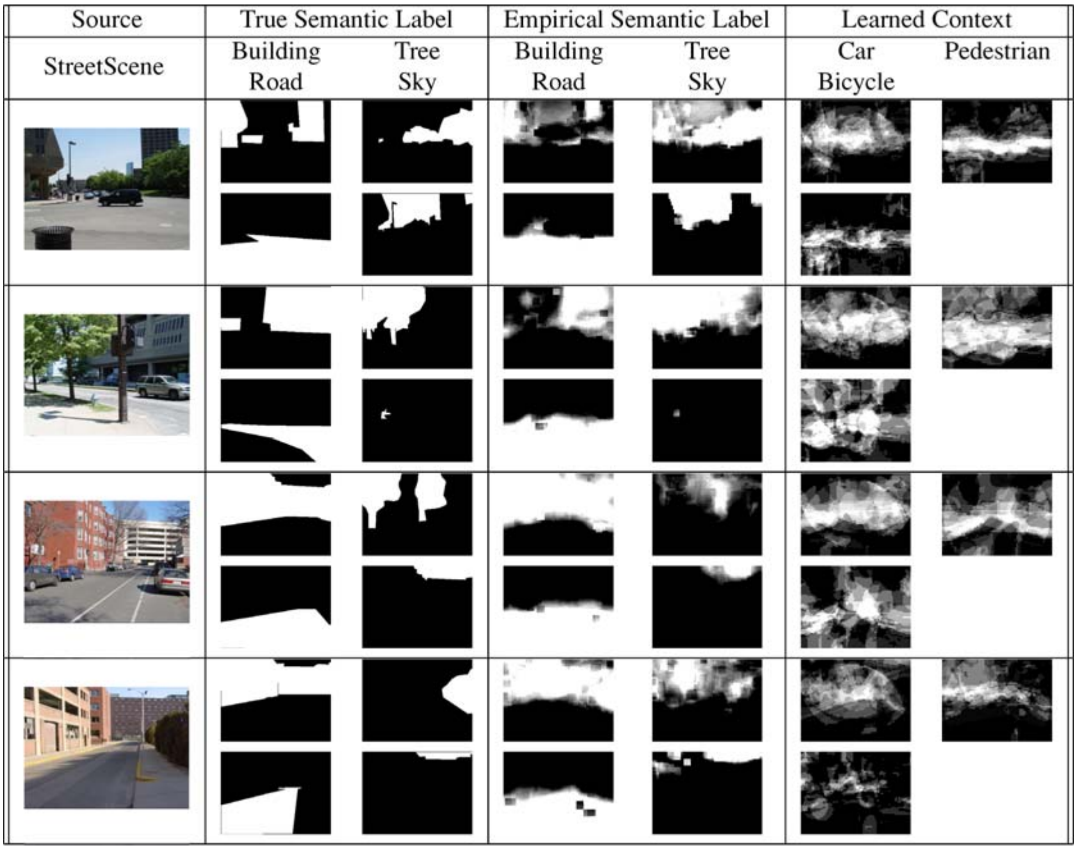

Figure 2. Column 1: Test images from the StreetScenes database. Column 2: True hand-labeled semantic data for the building, tree, road, and sky class. Column 3: Automatically classified semantic data. (Locations with larger positive distance from hyperplane shown brighter). Column 4: Learned context images for the three object classes: car, pedestrian, and bicycle. Brighter regions indicate context suggests objects presence.

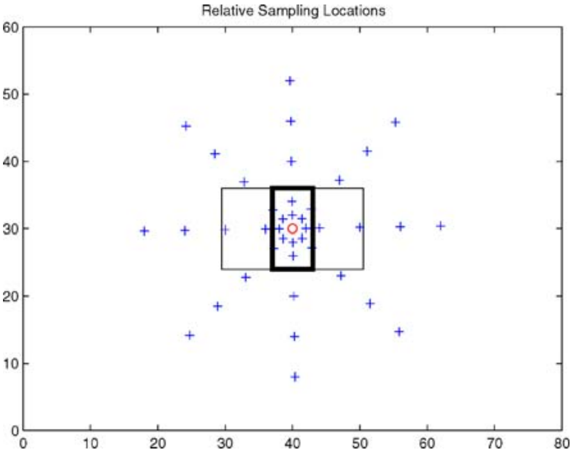

Figure 3. An illustration of the 40 relative pooling locations, plotted as blue '+' signs, relative to the red ○ . The thin black rectangle represents the average size of the cars in the database, and the thick black rectangle represents the average size of pedestrians.

this argument would support the claim that one should only include the original pixel-level information, since all visual features can be computed directly from these. Since current classifiers are incapable of automatically learning appropriate data representations, it makes sense to include all useful representations of the input.

4. Experiments and Results

4.1. Fidelity of Semantic Information

Empirical semantic features are learned via four SVMs trained to discriminate between positive and negative examples of the four classes: building, tree, road, and sky. The features used to learn these classes are the color, texture, and global position information described in Section 3.1. By splitting the training data and using cross validation, we obtain the ROC curves

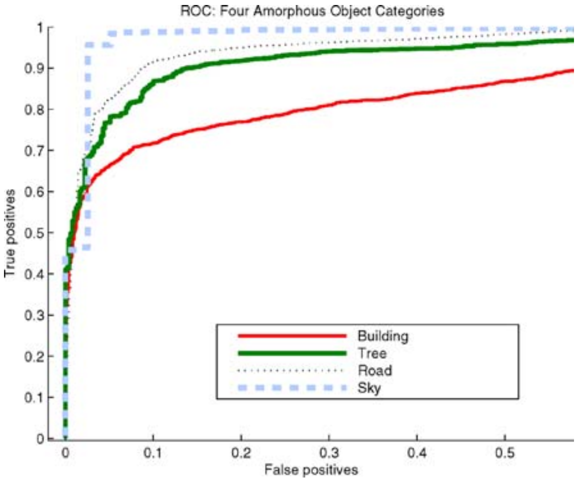

Figure 4. ROC curves for the four semantic classifiers; building, tree, road, and sky.

illustrated in Fig. 4. These training and testing examples were drawn randomly from those pixels with exactly one label from the set {building, tree, road, sky}, and negative examples of each category consisted of the positive examples from the other three categories. While the learned semantic classifiers are not perfect, they are operating at a level much better than chance. See Fig. 2 for some examples of learned semantic labeling.

4.2. *Performance of the Context Detector*

Once the 100 images selected for the training of the semantic classifiers are removed, 3,447 labeled images remain in the StreetScenes database. This corpus is split evenly into context-training and context-testing sets. To learn a model of object context, it is necessary to collect a database of samples of positive and negative context. One sample of positive context is taken per labeled target object in the training database. For instance, for each labeled car example in the training database, one 800 dimensional sample of positive context is extracted at the approximate center of the car. Additionally, ten times as many locations of *negative* context are recorded from locations at least 7 pixels away from our target object. In this way, 3,002 car, 209 bicycle, and 1,449 pedestrian examples of positive context are recorded.

Models of context are built by training a boosting classifier until convergence on these corpora, and the performance is evaluated on the testing data (which is extracted analogously to the training data). The results are plotted in Fig. 5. For comparison, we include results for a similar context detection system where the SVM-estimated semantic features have been replaced with true hand-labeled semantic features for training and testing. We also include results for a detector of object *appearance* trained from the same object examples. A description of the structure of this appearance detector is available in the Appendix.

By comparing the ROC curves it can be seen that the advantage of having true semantic information, as opposed to the empirical semantic information, is negligible. The appearance detector outperforms the context detector in the low-false-positive region. To surmise, however, from these plots that the appearance detector is better than the context detector is wrong for the following reason: the measure used here is a measure for object detection, not object context detection. If, for instance, the context detector responds strongly to pedestrian context over a crosswalk, a location likely to have pedestrians, and there are in fact no pedestrians in the image, then by this measure the

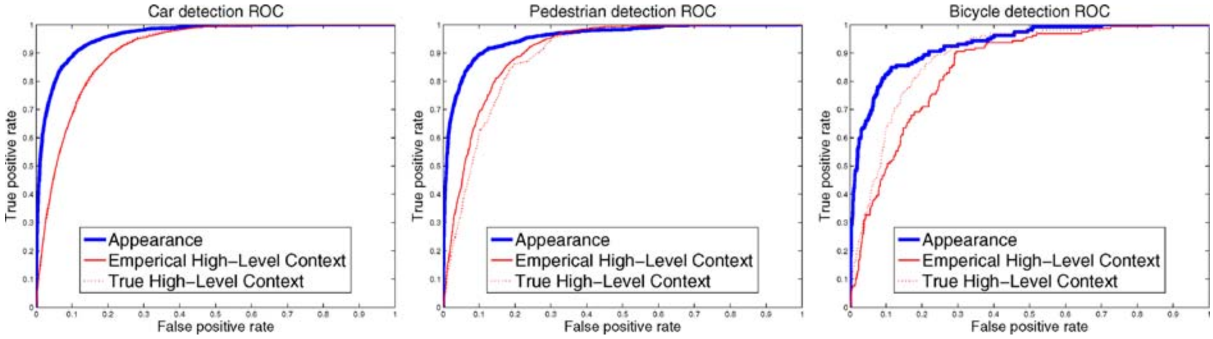

Figure 5. ROC curves for car, pedestrian and bicycle detection using (solid red): context with estimated semantic features, (dotted red): context with hand labeled semantic features, and (solid blue): appearance.

context detector has performed poorly, i.e., a false positive. The proper way to measure the context performance is to quantify how much aid is given to subsequent stages in the detection process. This is studied in Section 4.4

4.3. Relative Importance of Context Features

Previous systems employing context to aid in the object detection task have used either low-level image statistics or high-level object detections as their contextual clues. In this experiment we wish to answer whether the features input to our context classifier need to be of high level, or if, instead, the low level information is sufficient.

We train four context classifiers using the system outlined above in Section 3, the only difference between the four being the subset of the context features used. Classifier 'A' uses only position information and nothing else, classifier 'B' uses only semantic information and nothing else, classifier 'C' uses only color and texture information and nothing else, and finally classifier 'D' uses both color-texture features and semantic features, but no position information. The ROC curves of these classifiers are illustrated in Fig. 6 using the same measure as Fig. 5. We see from the figure that the position based classifier performs much better than chance, even though position information is identical for every test image. This performance is to be expected since most cars and pedestrians are near the bottom half of the image; the distribution is not uniform. The position-only-classifier can be considered to be calculating a sort of discriminative object prior for position. We see also that the semantic-information based classifier performs at about the same level as the position detector, even though this detector is not privy to the position information. We are surprised, however, to see that the low-level-feature based detector performs better than either of these. It was presumed that information about the relative positions of large objects within the image would be the best cue as to the likely location of the target objects. This would be of definite interest to practitioners, who may invest a great deal of time into complicated contextual classifiers and may be disappointed by the actual benefit over simpler methods.

Also of note is that the classifier which uses both semantic features and color-texture features does only marginally better than the classifier which uses only color-texture features. This suggests that almost all the relevant information available from these semantic features is also immediately available from the color-texture features. Results are not improved by using the true semantic information in place of the empirical semantic information.

One might notice that the system samples contextual information from some locations which may overlap the target object. It is possible that what is being learned is less a model of context and something more akin to a model of appearance. This would explain why the low level image description appears to be more influential than the high level information. In order to test this hypothesis, an experiment is performed further illuminating the relationship between the importance of a feature mode and its distance d from the point of interest. In this experiment, for each $d \in \{3, 5, 10, 15, 20\}$, a classifier is trained with only low-level or high-level features from distance d. No global position is included. Recall that the full image resolution at this stage is only 60×80, so these distances represent a wide receptive

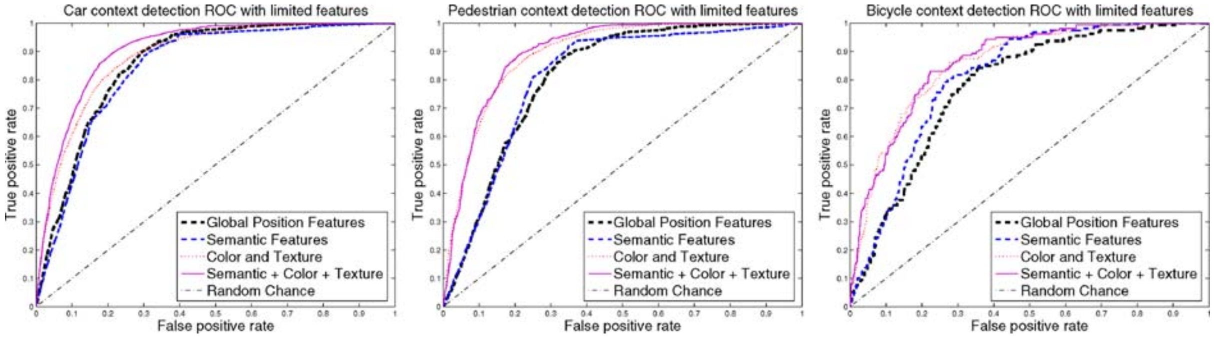

Figure 6. Different modes of contextual features have different utility to the detector. In all cases the low-level color and texture were the strongest cues. The performance of the system when using only semantic features is approximately equal to the performance when using only global position within the image.

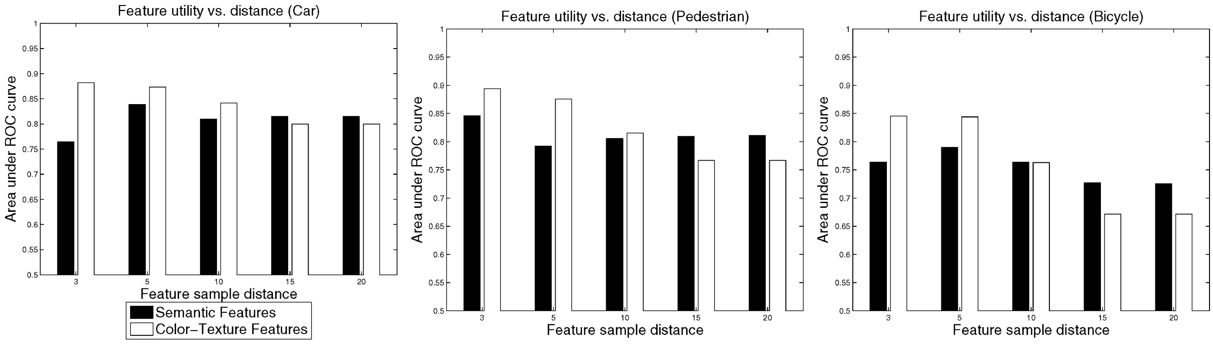

Figure 7. The quality of a detector of object context depends not only on the mode of features used, but also on the distance from which these features were sampled. For each object, context classifiers were trained and tested using either high or low-level features at $d \in \{3, 5, 10, 15, 20\}$. Each bar in this graph illustrates an area under an ROC curve of such a detector. As features are sampled from further away, the high-level information becomes more important than the low-level information.

field. The results of this experiment are illustrated in Fig. 7. Plotting the area under the ROC curve for these classifiers illustrates that for all three objects tested, as one takes relative locations further and further from the target object, the semantic features become more important than the color and texture features. However, the low-level features retain much of their discriminative power even at great distances from the target object. Paraphrased, knowing the color and texture information at a few points very distant from a point of interest is about equally useful as knowing whether those same distant points have skies, buildings, trees or road, at least for the task of deciding whether the point of interest is likely to have a car, pedestrian, or bicycle.

The impact of these studies is pertinent to anyone implementing a contextual system to aid in the detection of objects. If early visual features can be used in place of high-level semantic information, then tremendous time can be saved by not labeling data and training classifiers to detect the neighboring objects. Instead, all the relevant information is already available with simple image transformations and a larger receptive field.

4.4. Improving Object Detection with Context

In this final experiment it is demonstrated that context can be used to improve system performance. The architecture we will use is the rejection cascade illustrated in Fig. 8. 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. A pixel is judged to be an object detection only if the pixel passes both detectors. The context confidence threshold which maximizes the area under the ROC curve of the complete system is selected empirically using a validation set of 200 images. Figure 9 illustrates the effect that the TH_C has on the performance of the detector cascade for three different objects.

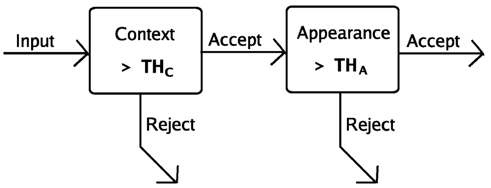

Figure 8. A data flow diagram of a rejection cascade combining both context and appearance information. Inputs are classified as positive only if they pass both classifiers. By tuning the confidence thresholds TH_C and TH_A, different points are achieved in the ROC plane.

ROCs of full system performance are illustrated in Fig. 10. These curves suggest that using context as a preliminary filter for an appearance detector may be a valid strategy, but, at least in this case the performance gain is marginal. The reason why the context cue was of so little assistance in this experiment can be understood by inspecting the distribution of the data. In Fig. 11 we plot the car examples in the plane where the x axis is the empirical appearance score, and the y axis is the empirical context score. A system trained to discriminate based on appearance alone would classify examples by setting some threshold along the appearance axis. In Fig. 11 the appearance classification

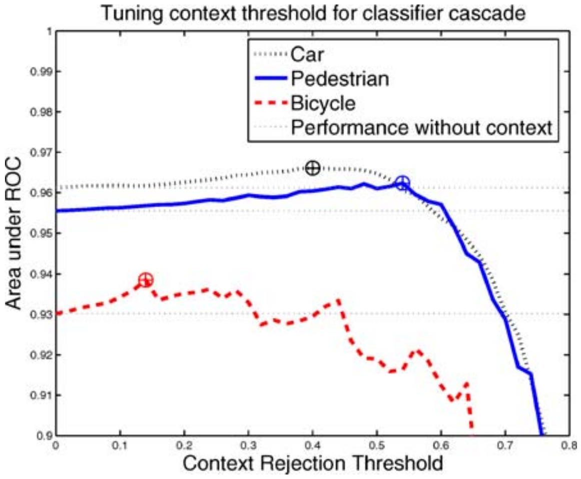

Figure 9. Area under the ROC curve of the rejection-cascade as a function of TH_C. Horizontal lines indicate performance with no context ($TH_C = -\infty$). The $\oplus$ marks the system's parameters selected via cross validation.

boundary is illustrated with the vertical dashed line. The boundary for the context classifier is illustrated similarly as the solid horizontal line. Points which are classified positively by both systems lie in the upper-right quadrant of the diagram. The context classifier aids the system by rejecting negatives which are strong in appearance but weak in context, e.g., the negative points in the lower right quadrant. The important point to notice is that the distribution is curved such that these points are very rare. There are few samples, positive or negative, which appear like the target object and are simultaneously out of context. It is for this reason that the context cue did not give much performance gain. A superior appearance-based detector would achieve better horizontal separation of the positive and negative points, further marginalizing the importance of context. We attempted to use several other models of classifier combination, including training a linear model, but results were similar. Note that if boosting is used on the appearance and context together in one feature vector the classification performance is even worse than just using appearance information alone, suggesting that in this scenario the appearance information is much more relevant to the detection problem.

Further support of this thesis can be seen from the results published in Torralaba et al. (2004), where for the three target objects *computer mouse*, *keyboard*, and *monitor*, the context is marginally helpful for the detection of the monitor, somewhat helpful for the keyboard, and very helpful for the detection of the mouse. Since the mouse is physically small it is difficult to detect without the contextual cues, but the monitor is visually unambiguous, a conclusion not made in the original work. For our application there is very little appearance ambiguity.

5. Summary and Conclusions

The context system described in this work is simple enough for others to use in their own work and general enough to function across several object types. Experimental results demonstrate effective context detection for cars, pedestrians, and bicycles, and furthermore show that these context detections can be used in a rejection cascade architecture to improve detection accuracy. Our system's feed-forward design makes it possible to determine a map of object context at a resolution of 60 × 80 in under 10 seconds using a standard desktop computer.

It is commonly assumed that contextual cues can do much to improve the accuracy of an object detection

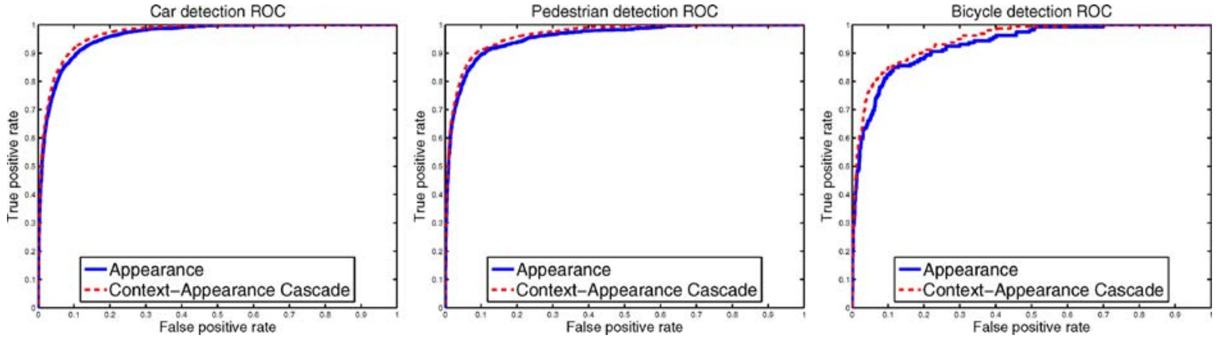

Figure 10. The car rejection cascade incorporating both context and appearance information outperforms the system using appearance alone.

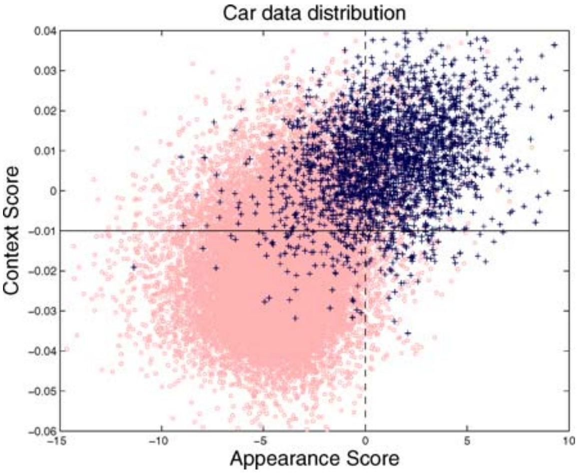

Figure 11. Empirical distribution of the car data in the appearance-context plane. Positive points are black '+' signs, negative points are the pink '○' signs. Selected thresholds for the context and appearance detectors are shown as solid and dashed lines, respectively. Note that few points are simultaneously strong in appearance but weak in context.

system by eliminating false positives that fall out of context. We have demonstrated that visual phenomena which bear strong visual resemblance to the target object while simultaneously being out of context may be very rare, and in this case the benefit to be gained by such a combination is marginal. Instead, we propose that context is a useful cue for robust object detection *only* when the appearance information is weak, such as in critically low-resolution or very noisy images. In retrospect, it shouldn't be surprising that context is useful in some situations and not useful in others.

Our investigation into the relative importance of different modalities of context features is the first of its kind. Common wisdom suggests that context must be computed at a high level by inferring likely target object locations from the locations of other related objects in the scene, but our experiments show that accurate context can be determined from the low-level early visual features both near and far from the location of interest. It is hoped that other practitioners will take note and attempt simple contextual methods before building detectors for related objects.

Appendix: The Appearance Detector

The appearance detectors for the three object classes are all constructed via linear kernel SVMs. Training examples for these classifiers were selected from the StreetScenes data using the same methodology as for the context detector, as described in Section 4.2. In brief, positive and negative samples of object appearance are extracted from the training images by selecting appropriate locations and sizes. For each training example, the minimum square bounding box of the object is calculated, and widened by a factor of $\frac{1}{6}$ on all sides. This bounding box is cropped from the image, converted to gray-scale, and resized to exactly 64×64 pixels. The resulting image is linearly filtered with 6 filters: four 3×3 Sobel filters at 45° intervals, one 3×3 Laplacian filter, and one identity filter. After taking the absolute value of the result, the resulting 6 images are submitted to the morphological gray-scale dilation operation using the 8-neighbor model of connectivity and a radius of 5 pixels. Finally, the images are downsampled to 16×16 using bilinear filtering. The resulting $1,536$ dimensional data ($6 \times 16 \times 16$) is used to train the SVM. In a windowing framework it is possible to filter and dilate the image before the actual windowing step, so as to save computation time.

Acknowledgments

This report describes research done at the Center for Biological & Computational Learning, which is in the McGovern Institute for Brain Research at MIT, as well as in the Dept. of Brain & Cognitive Sciences, and which is affiliated with the Computer Sciences & Artificial Intelligence Laboratory (CSAIL). This research was sponsored by grants from: Office of Naval Research (DARPA) Contract No. MDA972-04-1-0037, Office of Naval Research (DARPA) Contract No. N00014-02-1-0915, National Science Foundation-NIH (CRCNS) Contract No. EIA-0218506, and National Institutes of Health (Conte) Contract No. 1 P20 MH66239-01A1. Additional support was provided by: Central Research Institute of Electric Power Industry (CRIEPI), Daimler-Chrysler AG, Eastman Kodak Company, Honda Research Institute USA, Inc., Komatsu Ltd., Merrill-Lynch, NEC Fund, Oxygen, Siemens Corporate Research, Inc., Sony, Sumitomo Metal Industries, Toyota Motor Corporation, and the Eugene McDermott Foundation.

Note

1. We define receptive field of an image feature to mean all those pixels in the original image which may have influenced that feature.

References

Bar, M., Aminoff, E., Boshyan, J., Fenske, M., Gronauo, N., and Kassam, K. 2005. The contribution of context to visual object recognition. *Journal of Vision*, 5(8):88a.

Biederman, I., Mezzanotte, R.J., and Rabinowitz, J.C. 1982. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive Psychology*, l4.

Belongie, S., Malik, J., and Puzicha J. 2002. Shape Matching and Object Recognition Using Shape Contexts. PAMI.

Bergboer, N.H., Postma, E.O., and van den Herik, H.J. 2003. Context-enhanced object detection in natural images. In *Proc. of the Belgian-Netherlands AI Conference (BNAIC)*.

Bileschi, S.M. and Heisele, B. 2003. Advances in component based face detection. In *Int. Workshop on Analysis and Modeling of Faces and Gestures*.

Carbonetto, P., Freitas, N., and Barnard, K. 2004. A statistical model for general contextural object recognition. *ECCV*.

Carson, C., Belongie, S., Greenspan, H., and Malik, J. 1998. Color- and texture-based image segmentation using em and its application to content-based image retrieval, *ICCV*.

Cox, D., Meyers, E., and Sinha, P. 2004. Contextually evoked object-specific responses in human visual cortex. *Science*, 304:115–117.

Fink, M. and Perona, P. 2003. Mutual boosting for contextual inference. *NIPS*, 17.

Hanson, A.R. and Riseman, E.M. 1978. Visions: A computer system for interpreting scenes. In *Computer Vision Systems*, Academic Press.

Haralick, R.M. 1983. Decision making in context, PAMI.

Kruppa, H. and Schiele, B. 2003. Using context to improve face detection. *British machine Vision Conference (BMCV)*.

Kumar, S. and Hebert, M. 2003. Discriminative random fields: A discriminative framework for contextual interaction in classification, ICCV.

LeCun, Y. and Jie Huang, F. 2004. Learning methods for generic object recognition with invariance to pose and lighting. *CVPR*

Murphy, K., Torralba, A., and Freeman, W. 2003. Using the forest to see the trees: A graphical model relating features, objects, and scenes. *NIPS*, 16.

Schneiderman, H. 2004. Learning a restricted Bayesian network for object detection. *CVPR*.

Singhal, A., Luo, J., and Zhu, W. 2003. Probabilistic spatial context models for scene content understanding. *CVPR*.

Strat, T.M. and Fischler, M.A. 1991. Context-based vision: Recognizing objects using information from both 2D and 3D imagery. *PAMI*, 13:1050–1065.

Torralba, A., Murphy, K., and Freeman, W. Contextual models for object-detection using boosted random fields. *NIPS'*04.

Torralba, A., Murphy, K.P., and Freeman, W.T. 2004. Contextual models for object detection using boosted random fields. *NIPS*.

Torralba, A. and Sinha, P. 2001. Statistical context priming for object detection. *ICCV*, pp. 763–770.

Vapnik, V. 1999. *The Nature of Statistical Learning Theory*. Springer-Verlag.

Viola, P. and Jones, P. 2001. Rapid object detection using a boosted cascade of simple features. *CVPR*.