Computer Science and Artificial Intelligence Laboratory
Technical Report

MIT-CSAIL-TR-2008-019 April 9, 2008
CBCL-271

A Multi-Scale Generalization of the HoG and HMAX Image Descriptors for Object Detection

Stanley M Bileschi

massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

A Multi-Scale Generalization of the HoG and HMAX Image Descriptors for Object Detection

Stanley Bileschi
Massachusetts Institute of Technology
Cambridge MA
bileschi@mit.edu

Abstract

Recently, several powerful image features have been proposed which can be described as spatial histograms of oriented energy. For instance, the HoG [5], HMAX C1 [14], SIFT [13], and Shape Context feature [4] all represent an input image using with a discrete set of bins which accumulate evidence for oriented structures over a spatial region and a range of orientations. In this work, we generalize these techniques to allow for a foveated input image, rather than a rectilinear raster. It will be shown that improved object detection accuracy can be achieved via inputting a spectrum of image measurements, from sharp, fine-scale image sampling within a small spatial region within the target to coarse-scale sampling of a wide field of view around the target. Several alternative feature generation algorithms are proposed and tested which suitably make use of foveated image inputs. In the experiments we show that features generated from the foveated input format produce detectors of greater accuracy, as measured for four object types from commonly available data-sets. Finally, a flexible algorithm for generating features is described and tested which is independent of input topology and uses ICA to learn appropriate filters.

1. Introduction

In the field of object detection, often it is the features used to represent the input, rather than the statistical techniques used to learn patterns of those features, that are the key to accurate performance. Whereas the earliest detectors used simple cues such as gray-scale values, wavelet coefficients, or histograms of RGB values, modern techniques can attribute much of their success to features such as Lowe's SIFT [13], Dalal's Histogram of Oriented Gradients (HoG) [5], the visual Bag-Of-Words [18], and hierarchical networks of selectivity and invariance, such as Poggio's HMAX network, LeCun's convolutional network, among notable others [12, 14, 10, 9, 8].

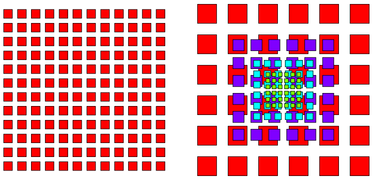


Figure 1. An illustration of the difference between rectilinear *left* and foveal *right* image sampling. Image features normally input rectilinear image samples. This work explores the value of foveal sampling.

The aim of this work is to improve upon two existing image representations via adaptation to a foveated input. Whereas most image features begin by ingesting image data sampled at regular intervals, the aim of this work is to adapt the HoG and HMAX feature to input a spectrum of brightness values, sampled densely at a fine scale at the center of the target, and coarsely further away, as illustrated in Fig. 1.

The motivation behind such an approach is twofold. Firstly, biologically, the sensors of the vertebrate eye are arranged in such a log-polar fashion. There are many cells with high acuity and small receptive fields at the center of the focal location, and there are cells with wide receptive fields which are sensitive to light far from the attended location. The second motivation, computationally, is that recent experiments exploring visual gist (e.g., [19]) suggest that object detection may often be a matter of context as much as appearance. By sampling in such a pattern, the classifier has access to information about the surroundings of the object, the appearance of the object itself, and the texture within the object.

There are three major experimental efforts within this work. First we will explore how the scale of the input image, relative to the size of the target, affects the performance of an object detector. This experiment will uncover the relative utility of differing scales. Secondly, multiple scales will be included into the same classifier in a simple early-fusion framework. This experiment will expose whether or not features from different scales can be combined to build a more accurate detector than any one scale, i.e., whether or not the information contained in separate scales is redundant. Finally, more sophisticated features will be discussed, designed to leverage information across many scales. Adaptations to both HMAX and HoG are described and tested.

2. The HoG and HMAX Features

In order to best understand the experiments conducted in this work, and to get a feel for the motivation underlying the design of the multi-scale features, it is important to understand the features and models they were extended from. The Histogram of Oriented Gradients (HoG) algorithm and the Hierarchal Maximization Architecture (HMAX) algorithm are both well-known, successful methods for converting an input image into a mathematical description, or feature vector [5, 14]. HoG was designed as an adaptation to SIFT [13] to maximize the detection accuracy on a pedestrian detection task, while HMAX was designed primarily to mimic the behavior of the early stages of the mammalian ventral visual stream. For the purposes of this work, we will only be using the first layers of HMAX, S1 and C1.

Recently these features, and other features like them, have been used to accurately detect a wide variety of objects in natural scenes [6, 17]. Along with SIFT and Shape Context [4], they produce feature vectors which are accumulations of evidences for spatial patterns of oriented structures within the input.

Both HoG and HMAX can also be described within the context of Hubel and Wiesel's simple and complex cell model of the human visual process, wherein simple cells (or "S units") respond to input patterns at specific positions and scales within the input, and complex cells accumulate the results from multiple simple cells, so as to build invariance to irrelevant input transforms [10]. For HMAX, the first S layer is computed via Gabor wavelet filtration. Complex cells then compute the maximum response over a small set of S units with similar position and orientation characteristics. It has been shown that this first layer of HMAX C cell outputs is sufficient for highly accurate detection of some objects [17].

Similarly, the HoG and SIFT features begin by calculating a gradient magnitude and direction at each pixel. It is easy to see that this could correspond to oriented filtration followed by a winner-takes-all lateral inhibition. The complex layer analog in HoG computes a weighted sum of the input S units, pooling inputs of similar position and orientation over space. It should be noted that the HoG feature also includes in its definition a normalization stage, which post-processes the output bins.

3. Data

In order to test the role of scale in object detection, it was necessary to find a suitable database of labeled visual objects within their natural contexts. It was important that the data included a wide background field around the objects, in order to explore larger scales. Furthermore, many labeled examples of each target object were necessary in order to have enough data to train a classifier and still perform statistically significant tests. Two objects in the LabelMe database [15] and two more in the StreetScenes database [1] were found to meet thede constraints. Both of these databases are available online for download[1].

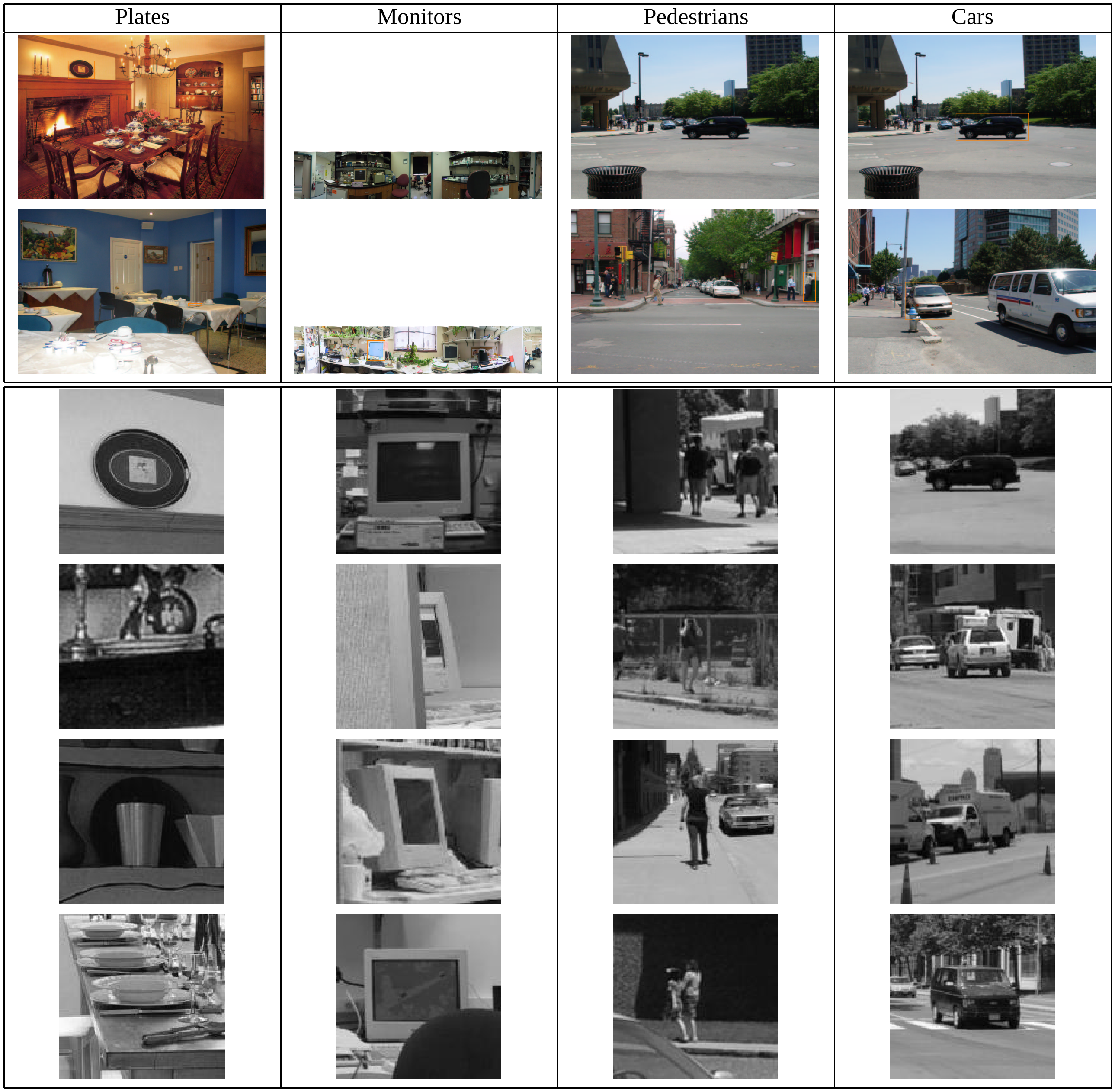
Within the LabelMe database, the *monitors* object from the `'fink-static-indoor-officepanorama-small'` subset and the *plates* object from the `'static-indoor-database-by-aude-oliva'` subset were chosen. In the StreetScenes database, the *cars* and *pedestrians* objects were selected. These objects were chosen because of the number of examples, the resolution of the scenes, and the relative size of the scene surrounding the object, i.e., small objects with a great deal of surrounding background were preferred. Fig. 2 illustrates some typical examples of these data and relates each object to the number of labeled examples, the average size of those examples, and the total number of images.

For negative examples, it was necessary to chose from a distribution similar to that of the positive data, to prevent learning spurious statistics unrelated to the presence of the object. Locations and images were chosen from the same marginal distributions as the positive data. Any candidate negative whose minimum bounding square intersected the bounding square of a positive example with an intersect to union ratio greater than .25 was rejected. In this way the negatives were drawn from the same images as the positives, and at the same expected locations and scales. Note that for each positive object class, an independent set of negatives were chosen. Note also that while care was taken not to include labeled positives in the negative set, unlabeled examples can sometimes be included due to imperfections in the ground truth. These represent a very small minority of the actual negative examples. Some example data is illustrated in Figure 2.

4. Accuracy as a Function of Scale

In this section an experiment is described in which an object detector is trained and tested using an existing image feature, but the input image is varied in scale, relative to the size of the ground truth hand-drawn label. In an effort to have the broadest possible applicability, this exper-

[1] http://cbcl.mit.edu/software-datasets/streetscenes and http://labelme.csail.mit.edu/



	Plates	Monitors	Pedestrians	Cars
n Labeled Objects	234	395	1449	5799
n Scenes with at least one labeled object	46	195	852	3090
median object size (pixels $x \times y$)	[60 × 42]	[142 × 145]	[72 × 160]	[295 × 157]
median scene size (pixels $x \times y$)	[1200 × 900]	[4092 × 621]	[1280 × 960]	[1280 × 960]

Figure 2. Illustration of the object-detection data used from the LabelMe and StreetScenes databases. *Top:* Two full scenes for each of the four object types. The target object is annotated with an orange bounding box. Note that the data used for monitor detection comes from wide panoramas of office scenes. *Middle:* Four sample extractions for each object, extracted slightly larger than the label and resized to 128×128 pixels. *Bottom:* Statistics of the data. It is easy to see that for all four object types there are a significant number of samples and source scenes, and that the target is generally significantly smaller than the scene as a whole.

iment is repeated for two choices of image feature (HoG and HMAX), using two different classifiers (gentleBoost [7], and a linear-kernel SVM), on the four object databases (Pedestrian, Car, Plate and Monitor), as described in Section 3.

Each experimental condition is executed as follows. First a set of positive and negative images were cropped from the database. The crop region was selected by first finding the minimum square bounding box around the object, and then scaling that box by some scale factor. The scale factors ranged from a minimum of $\frac{1}{2}$ to a maximum of 16 times the size of the original box. Figure 3 illustrates a the set of bounding boxes extracted for an example pedestrian. The small scales are indeed smaller than the target object, and, depending on the shape of the target, may be completely within the object. The largest scales leave the target object as a very small part of the window, most of the window is background or clutter. The positive and negative crops are all converted to grayscale and resized to 128×128 pixels using MATLAB's bilinear `imresize` function. This size was chosen to match the experiments of [17, 5] as closely as possible. The images were then converted into the target feature format, and a classifier was trained and tested using 5 random training and testing splits. In these experiments 75% of the data was used for training, and the remaining 25% for testing.

Figure 4 illustrates the output of this experiment, as measured via the average equal-error-rate (EER) of the resulting ROC curves. Each of the 8 graphs illustrates how the system accuracy behaves as a function of scale for one of the four objects, and one of the two image features. The blue circles indicate systems trained and testing using gentleBoost, and the red $\times$s plot the system using linear kernel SVM's. It is easy to see that the results from the two classifiers are not significantly different. For reference, scale index 4 is the scale factor where the extraction boundaries are equal to the minimum square bounding box enclosing the ground-truth polygon.

From these results we see that for each object, and for both features, there is a preferred scale which reliably produces the most accurate detections. As the crop region grows larger or shrinks smaller than this preferred scale, the performance suffers. Furthermore, we can see that for all four objects, the preferred scale is larger than the minimum bounding square. It is presumed that this is because both HoG and HMAX are representing a distribution of image edges (gradients), which are most stable at the object boundaries. Within the object we can not see those boundaries, and too far from the object the boundaries are not visible due to low resolution. Notice from these results that the performance of the detector even very close or very far from the object is significantly above random chance (EER $< .5$). This suggests that there is discriminative information in these measurements, and that perhaps this information is not available at other scales. A simple strategy of combining multiple scales into one detector will be explored in the next section, so as to see if performance can be improved over the the single optimum scale.

5. Multiple Scales in One Feature

In the previous experiment it was shown experimentally that object detection can be performed with some level of accuracy over a very wide range of scales, i.e., there is information about target presence at fine, near scales within the object as well as coarse scales with a wide range of view around the object. Still, it remains unclear whether this information is redundant across scales, or if it is complementary. In this experiment, information from multiple scales is included into a single classifier. The goal is to determine whether a multi-scale approach will outperform a single, optimized scale. The experimental design will be very simple, it will use the same setup as in the first experiment, but will input features from multiple scales into the classifier, instead of one scale at a time. Because our tools are limited to using data with fewer than about 4000 features, a simple feature selection approach will be used.

The experimental design proceeds as follows; first HoG or HMAX features were calculated from three scales independently as in the previous experiment. Scale factors 2, 6, and 10 were selected, corresponding to scales smaller than the object, slightly larger than the object, and much larger than the object (scale-factors 0.63, 1.59 ,and 4). These scales were chosen since they represent a wide range, but not so small or or so large as to severely impair performance, as can be seen from Fig. 4. For feature selection, a boosting classifier is trained on each of the three sets of features *independently*, noting the features from which the stumps were derived. Then a single monolithic boosting classifier is trained on the union of those three selected sets of features. Figure 5 illustrates the results of this experiment for the four objects tested, again in terms of equal error rate. In each graph, the results of the 13 single scale classifiers are plotted. Within the same plot, a horizontal line indicates the mean EER of the classifier trained with features from multiple scales (dotted lines indicate the standard deviation of the 10 trials). The red line shows the classifier trained with HMAX features and the blue line HoG features, though they are not statistically different.

For each object tested, the classification score from the multi-scale approach outperforms the best score from a classifier trained at any single scale. These results support the assertion that information from different scales isn't necessarily totally redundant, complementary information from different scales can be leveraged to improve system-wide performance, even when the underlying image feature and statistical learning method are unchanged.

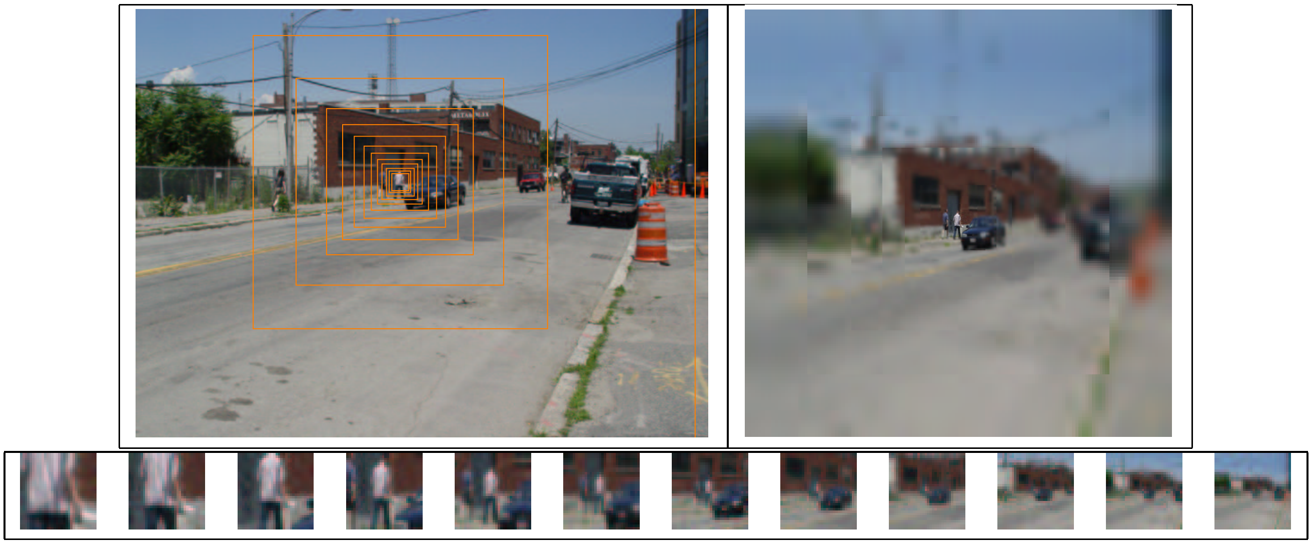


Figure 3. An illustration of the set of bounding boxes used to test object detection accuracy as a function of scale. *Top Left:* A sample scene and a set of 12 square bounding boxes, calculated as a function of the original bounding polygon of the pedestrian. *Bottom:* Extractions from those 12 boxes, scaled to 32 × 32 pixels each. *Top Right:* A reconstruction of the original scene using the small scaled extractions.

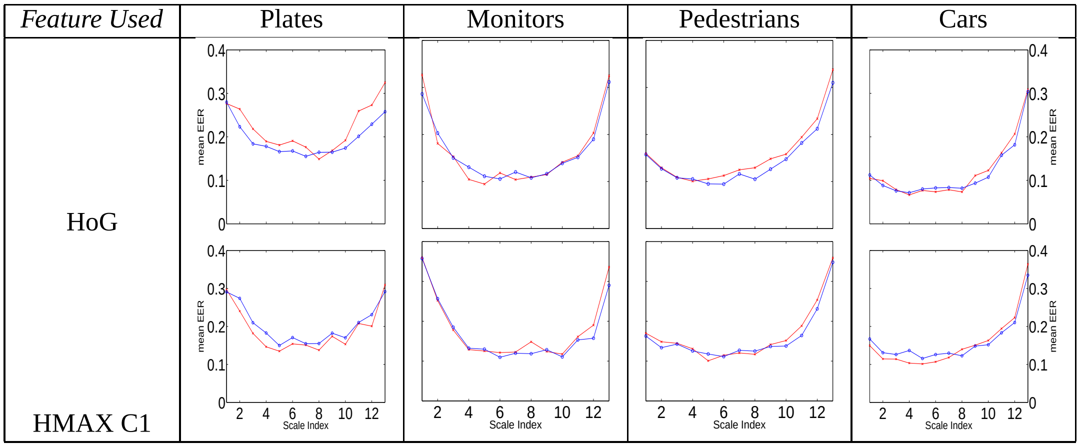


Figure 4. **Detection performance as a function of input scale.** Detailed in Sec. 4. Scales are indexed from factor .5 to 16 (too small to too large). Results are illustrated in the form of ROC Equal Error Rate (EER), averaged over 5 independent trials. These results show that for the parameters tested, independent of the object, the statistical learning machinery, or the input feature, the scale of the input image affects the detection rate in a predictable way. If the scale is too small or too large, the classification problem is more difficult.

6. Two Features Designed Natively Multiscale

In the previous section it was demonstrated that a classifier with access to features from multiple scales can outperform the same statistical learning machine with features from only the best single scale. In this section we outline the design of two features similar to the HMAX or HoG feature, but designed with a multi-scale foveated inputs in mind. While the input to HMAX or HoG is often a 128×128 grayscale image, the input used in for these two features is a set of 8 32×32 images, centered at the same location, but with scales ranging over 4 octaves. Compared to a single 128×128 input image, this is only half as many samples (8192), but arranged in a foveated manner as illustrated in figure 3.

Two features for foveated inputs will be described. The first simply applies a suitable HoG-like algorithm to each 32×32 image independently, concatenating the values from each scale. It will be shown that this feature is fast and accurate, but depends on the input being arranged as a set of $2D$ images. The second feature is free from that constraint, but not as accurate. It is an attempt at producing a feature which is independent of the input image structure, so as to pool and compare information across several scales. It's architecture is inspired by that described by that of HMAX

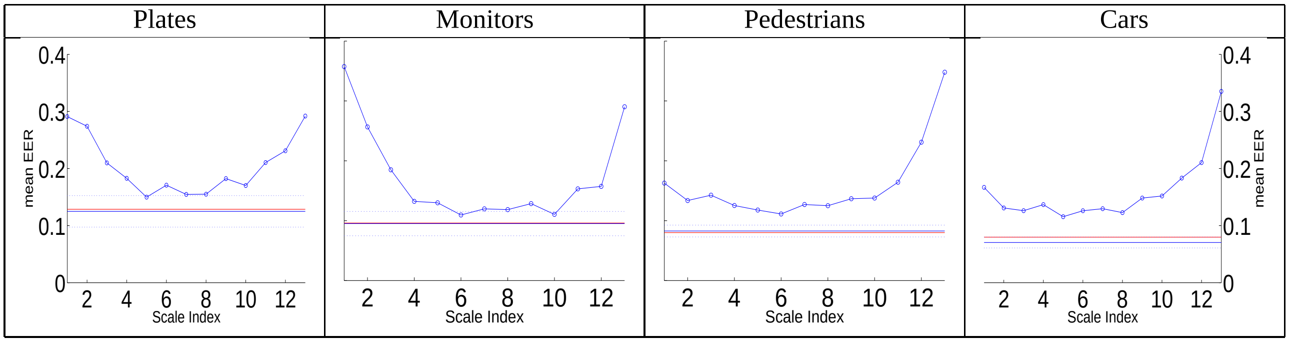


Figure 5. **Single Scale vs. Multi-Scale Detection**. As detailed in Sec. 5, the performance of classifiers with multi-scale input is compared to the 13 classifiers from the previous experiment (using HoG and boosting). The solid lines indicate the mean EER of the two classifiers trained with inputs from 3 different scales. The red line shows the multi-scale classifier trained with HMAX features, HoG features are shown in blue, though there is little difference. The dotted lines illustrate the boundaries of the standard deviation for the multi-scale classifier trained on HoG.

and the Hubel and Wiesel [10] architecture in general.

6.1. Foveated Histogram of Gradients

The first natively multi-scale feature uses a HoG feature independently on each scale. Since each image is only 32×32 pixels, the parameters were adjusted accordingly. As in the standard HoG feature, the first step estimates orientation magnitude and direction at each pixel. The second step bins these elements into a histogram, where each bin is 8 pixels across, and magnitudes are split trilinearly across space and orientation into in general 8 histogram bins. Blockwise normalization proceeds as in the original work, where each 2×2 spatial block of histogram bins is normalized to have unit length in L_1. Since each block has 4 locations and 9 orientations, there are 36 features per block. The small images only have 9 blocks each, meaning 324 features per scale. With the 8 scales used here, the Foveated Histogram of Gradients (FHoG) feature for this input produces 2592 total features.

The accuracy of the FHoG feature was compared to the single scale HoG and HMAX C1 feature from Sec. 4 using the same experimental setup. The results are illustrated in the box plot of Fig. 7 under condition **(C)**. For each of thefour objects, the FHoG feature out-performs the best associated single scale classifier **(A)** and matches the performance of the corresponding 3-scale classifier **(B)** described in Sec. 5, even though it receives $\frac{1}{6}$ as many brightness samples from the input.

6.2. Flexible Foveated Architecture

The final feature tested in this work implements a multi-scale feature considerably different from those described above. This feature, like those before, fits into the Hubel-Wiesel model, as described in Sec. 2, but differs in that connectivity and weights are learned, rather than hand designed or implied via weight sharing. The feature uses a simple three layer model, with the first layer representing the inputs, the second layer represents the filter-like outputs, and the third layer represents the features to be input to the statistical learning machine.

Like the HMAX feature, the first stage is a linear function of the inputs. And like the FHoG feature above, a set of 8 32×32 grayscale images shall be the input. However, rather than defining a filter and repeating it over space, as in traditional filtration, this feature, during a development phase, first selects a connectivity from the input to the second (S) layer, and then learns appropriate filter values using independent components analysis (ICA) [11]. ICA is a good choice to learn this transfer function because of its statistical properties[2] and for biological evidence supporting the existence of filters which are similar to those learned via ICA [3].

In a bit more detail, the function to convert the input into the S values was designed as following. First, a random set of 128 seed units were selected from the $8,192$ input units. For each seed unit, the set of the 63 most strongly correlated input units was determined. 128 sets of 64 inputs was chosen such that it would be possible (though unlikely), to cover the input space, leaving no input variable unused. Each pool of 64 brightness values becomes the input to a set of S units. In HoG, a small pool of input units feeds into 9 separate orientation tuned units, and in HMAX, pools of input units feed into 4 orientations, in this feature the same input units feed into 40 separate S level units, whose connection weights are learned via ICA [2]. It should be noted that learning patterns within groups of highly correlated inputs is not a new idea in the field of image features, it has been used successfully before, such as in [16].

Figure 6 illustrates the receptive fields learned for 8 S

[2] ICA produces a rotation of the input space in which the discovered directions are maximally independent.

units, 4 each from two input sets. These images were created by projecting the weights and receptive fields of the learned S units back into the input space. It is easy to see that ICA learns wavelet-like features from the multi-scale inputs. Note that these images are marginalized over scale, so S features which express differentials over scale are not visible here.

In order to simulate the complex (C) layer of the feature, it was necessary to choose a set of C units, and define their connectivity to the S layer. The strategy chosen was, for each C unit, begin with a random seed unit from the S layer, and choose a set units from the S layer with properties similar to that seed. The properties used to define S unit affinity were defined as functions of the units' receptive fields. The center of the receptive field was determined by calculating the center of mass of the projected afferent weights. The size of the receptive field was similarly determined. The frequency response was determined from the FFT of the receptive field. The similarity of two S units was calculated via a simple function of the distances between their centers, the difference between their sizes, and the difference between their orientation selectivity. For each C unit, the 16 units with affinity strongest for its seed became its inputs. The values of the C layer were calculated by simply taking the maximum for each unit over its afferent S values.

Using the C values calculated in this way, the object detection experiment was repeated for the four object types. The results of this experiment are summarized in Fig. 7, under heading **(D)**. The results suggest that while there is value in this method in terms of its novel architecture and the potential for tuning it, it is less accurate than simply calculating the HMAX or HoG feature on the image scales independently. It should be noted that this feature would perform identically should the input units be permuted, while the other features would need some sort of preprocessing to re-learn the mapping.

7. Summary and Next Steps

The contribution of this work is to clearly demonstrate the value of a multi-scale approach to object detection. It was first shown that object detection accuracy is dependant upon scale, for four separate object detection problems. Two important findings, consistent with the intuitive beliefs of the community, are that targets can be detected across a broad range of scales, and that there is a preferred scale which is slightly larger the size of the object itself.

After demonstrating the effect of scale on detection accuracy, we explored a small set of systems which included image features from multiple scales. The hypothesis was that a system privy to information from several scales could outperform the same system with access to only the most useful scale. This hypothesis was supported in Sec. 5 by training a more accurate classifier using features from several different scales. The hypothesis was bolstered further in Sec. 6 by using lower resolution images from each scale, and maintaining high levels of accuracy.

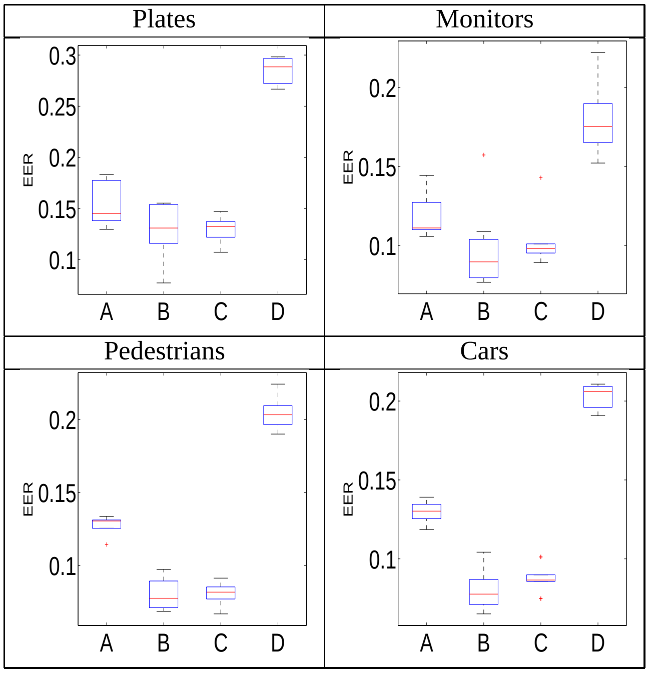


Figure 7. **Performance of Multi-scale features**. These box and whisker plots illustrate the object detection performance (in terms of EER) of the two multi-scale features described in Sec. 6. For comparison, the best single-scale classifier **(A)**, and the 3 scale classifier **(B)** are presented along side the results from the FHoG feature **(C)** described in Sec. 6.1 and the results from the flexible HMAX-like architecture **(D)** described in Sec. 6.2. It can be seen that the FHoG feature performs as well as the 3 scale classifier, even though it has far fewer inputs. Though **(D)** did not perform as well as the best single-scale classifier, it does significantly outperform a classifier trained directly on the inputs

Finally, we presented some preliminary work in the design of an image feature which natively ingests a multi-scale input. This extension of the histogram of oriented energy-like features shows promise in its flexibility to leverage multi-scale cues for target detection.

Our next steps are to continue to critically explore the space of multi-scale image features, so as to design features which are both discriminative for a wide variety of object types, and computationally inexpensive. Specifically, we will explore other methods of learning the connectivity and weights of the flexible network, beginning by replicating the success of calculating HMAX on each scale independently, and then slowly adapting the weights and connectivity to improve accuracy. Principles such as *slowness* [20] in a video framework, mutual information between units, and lateral inhibition within a layer will also be explored.

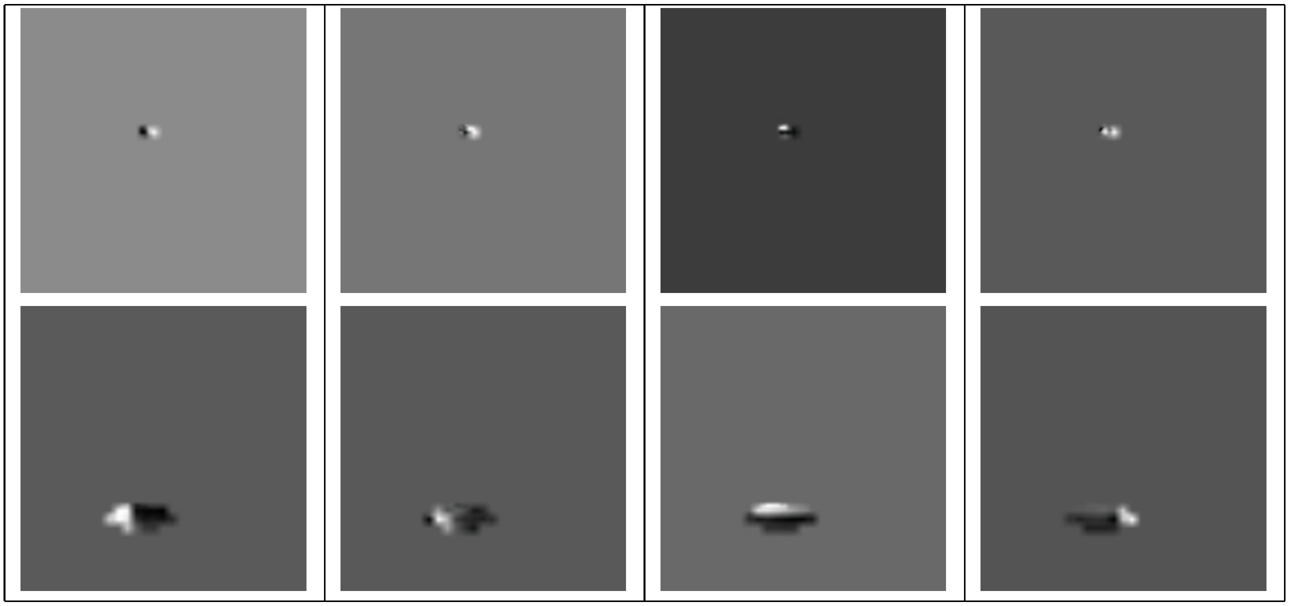


Figure 6. **Learned Receptive Fields**. Each image illustrates the receptive field learned for one of the S units in the flexible foveated feature. 4 S units from 2 sets are shown here. Learned connectivity and weights are projected back into the image space to produce patterns of strong excitation for these units.

References

[1] http://cbcl.mit.edu/software-datasets/streetscenes/. 2

[2] A.‘Hyvaarinen and A. Hoyer. Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, 12(7):1705–1720, 2000. 6

[3] A. J. Bell and T. J. Sejnowski. The ‘independent components’ of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997. 6

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 2002. 1, 2

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2:886–893, 2005. 1, 2, 4

[6] M. Everingham, L. V. Gool, C. K. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 2

[7] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Stanford University Technical Report*, 1998. 2

[8] K. Fukushima. Neocognitron capable of incremental learning. *Neural Networks*, 17(1):37–46, January 2004. 1

[9] G. Hinton, S. Osindero, and Y. W. Teh. A fast learning algorithm for deep belief nets. *Neural Comp.*, 18(7):1527–1554, July 2006. 1

[10] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction, and functional architecture in the cat’s visual cortex. *J. Physiology*, 160(1):106–154, 1962. 1, 2, 6

[11] A. Hyvrinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13:411–430, 2000. 6

[12] Y. LeCun, Huang, and Bottou. Learning methods for generic object recognition with invariance to pose and lighting. *CVPR*, 2004. 1

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 1, 2

[14] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2:1019–1025, 1999. 1, 2

[15] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 2007. 2

[16] H. Schneiderman and T. Kanade. A statistical model for 3d object detection applied to faces and cars. *CVPR*, 2000. 6

[17] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Object recognition with cortex-like mechanisms. *In: IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:411–426, 2007. 2, 4

[18] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their location in images. *ICCV*, 1:370–377, 2005. 1

[19] A. Torralba. Contextual priming for object detection, 2003. 1

[20] L. Wiskott. Slow feature analysis: a theoretical analysis of optimal free responses. *Neural Comput.*, 15(9):2147–2177, 2003. 7