Robust Object Recognition with Cortex-Like Mechanisms

Thomas Serre, Lior Wolf, Stanley Bileschi, Maximilian Riesenhuber, and Tomaso Poggio, *Member*, *IEEE*

Abstract—We introduce a new general framework for the recognition of complex visual scenes, which is motivated by biology: We describe a hierarchical system that closely follows the organization of visual cortex and builds an increasingly complex and invariant feature representation by alternating between a template matching and a maximum pooling operation. We demonstrate the strength of the approach on a range of recognition tasks: From invariant single object recognition in clutter to multiclass categorization problems and complex scene understanding tasks that rely on the recognition of both shape-based as well as texture-based objects. Given the biological constraints that the system had to satisfy, the approach performs surprisingly well: It has the capability of learning from only a few training examples and competes with state-of-the-art systems. We also discuss the existence of a *universal*, redundant dictionary of features that could handle the recognition of most object categories. In addition to its relevance for computer vision, the success of this approach suggests a plausibility proof for a class of feedforward models of object recognition in cortex.

Index Terms—Object recognition, model, visual cortex, scene understanding, neural network.

1 Introduction

Understanding how visual cortex recognizes objects is a critical question for neuroscience. Because humans and primates outperform the best machine vision systems with respect to almost any measure, building a system that emulates object recognition in cortex has always been an attractive but elusive goal. For the most part, the use of visual neuroscience in computer vision has been limited to early vision for deriving stereo algorithms (e.g., [1]) and to justify the use of DoG (derivative-of-Gaussian) filters and more recently of Gabor filters [2], [3]. No real attention has been given to biologically plausible features of higher complexity. While mainstream computer vision has always been inspired and challenged by human vision, it seems to never have advanced past the very first stages of processing in the simple (and binocular) cells in $V1$ and $V2$. Although some of the systems inspired—to various degrees—by neuroscience [4], [5], [6], [7], [8], [9], [10] have been tested on at least some natural images, neurobiological models of object recognition in cortex have not yet been extended to deal with real-world image databases [11], [12], [13], [14].

We present a system that is based on a quantitative theory of the ventral stream of visual cortex [14], [15]. A key element in the approach is a new set of scale and position-tolerant feature detectors, which agree quantitatively with the tuning properties of cells along the ventral stream of visual cortex. These features are adaptive to the training set, though we also show that a *universal* feature set, learned from a set of natural images unrelated to any categorization task, likewise achieves good performance. To exemplify the strength of this feature-based representation, we demonstrate classification results with simple linear classifiers. We show that the approach performs well on the recognition of isolated objects in clutter for both binary and multiclass classification problems on publicly available data sets. Our approach also demonstrates good classification results on a challenging (street) scene understanding application that requires the recognition of both shape-based as well as texture-based objects.

Both the source code of our system and the *StreetScenes* data set used in our experiments are readily available [16].

1.1 Related Work

Hierarchical architectures have been shown to outperform single-template (flat) object recognition systems on a variety of object recognition tasks (e.g., face detection [17] and car detection [18]). In particular, constellation models [19], [20], [21] have been shown to be able to learn to recognize many objects (one at a time) using an unsegmented training set from just a few examples [20], [21]. Multilayered convolutional networks were shown to perform extremely well in the domain of digit recognition [4], [5] and, more recently, generic object recognition [10] and face identification [22].

The simplest and one of the most popular appearance-based feature descriptors corresponds to a small gray value patch [23] of an image, also called component [17], [24], part [19], [25], or fragment [26]. Such patch-based descriptors are, however, limited in their ability to capture variations in the

- *T. Serre is with the Massachusetts Institute of Technology, the Center for Biological and Computational Learning, McGovern Institute for Brain Research and Brain & Cognitive Sciences Department, 77 Massachusetts Avenue, Bldg 46-5155B, Cambridge, MA 02139. E-mail: serre@mit.edu.*
- *L. Wolf and S. Bileschi are with the Massachusetts Institute of Technology, the Center for Biological and Computational Learning, McGovern Institute for Brain Research and Brain & Cognitive Sciences Department, 77 Massachusetts Avenue, Bldg 46-5155C, Cambridge, MA 02139. E-mail: {liorwolf, bileschi}@mit.edu.*
- *M. Riesenhuber is with Georgetown University Medical Center, Research Building Room WP-12 3970, Reservoir Rd., NW, Washington, DC 20007. E-mail: mr287@georgetown.edu.*
- *T. Poggio is with the Massachusetts Institute of Technology, the Center for Biological and Computational Learning, McGovern Institute for Brain Research and Brain & Cognitive Sciences Department, 77 Massachusetts Avenue, Bldg 46-5177B, Cambridge, MA 02139. E-mail: tp@ai.mit.edu.*

Manuscript received 8 Sept. 2005; revised 14 May 2006; accepted 27 June 2006; published online 15 Jan. 2007.
Recommended for acceptance by M. Srinivasan.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0488-0905.

object appearance: They are very selective for a target shape but lack invariance with respect to object transformations.

At the other extreme, histogram-based descriptors [27], [28], [29] have been shown to be very robust with respect to object transformations. Perhaps the most popular features are the SIFT features [27], which excel in the redetection of a previously seen object under new image transformations and have been shown to outperform other descriptors [30]. However, as we confirmed experimentally (see Section 3.1.2), with such a degree of invariance, it is very unlikely that these features could perform well on a generic object recognition task.

The new appearance-based feature descriptors described here exhibit a balanced trade-off between invariance and selectivity. They are more flexible than image patches and more selective than local histogram-based descriptors. Though they are not strictly invariant to rotation, invariance to rotation could, in principle, be introduced via the training set (e.g., by introducing rotated versions of the original input).

1.2 The Standard Model of Visual Cortex

Our system follows a recent theory of the feedforward path of object recognition in cortex that accounts for the first 100-200 milliseconds of processing in the ventral stream of primate visual cortex [14], [15]. The model itself attempts to summarize—in a quantitative way—a core of well-accepted facts about the ventral stream in the visual cortex (see [15] for a review):

1. Visual processing is hierarchical, aiming to build invariance to position and scale first and then to viewpoint and other transformations.
2. Along the hierarchy, the receptive fields of the neurons (i.e., the part of the visual field that could potentially elicit a response from the neuron) as well as the complexity of their optimal stimuli (i.e., the set of stimuli that elicit a response of the neuron) increases.
3. The initial processing of information is feedforward (for *immediate recognition* tasks, i.e., when the image presentation is rapid and there is no time for eye movements or shifts of attention).
4. Plasticity and learning probably occurs at all stages and certainly at the level of inferotemporal (IT) cortex and prefrontal cortex (PFC), the top-most layers of the hierarchy.

In its simplest form, the model consists of four layers of computational units, where *simple* S units alternate with *complex* C units. The S units combine their inputs with a bell-shaped tuning function to increase selectivity. The C units pool their inputs through a maximum (MAX) operation, thereby increasing invariance.[1] Evidence for the two key operations as well as biophysically plausible circuits can be found in [15]. The model is qualitatively and quantitatively consistent with (and, in some cases, actually predicts) several properties of cells along the ventral stream of visual cortex (see [15] for an overview). For instance, the model predicts, at the C_1 and C_2 levels (see Fig. 1), respectively, the max-like behavior of a subclass of complex cells in V1 [31] and cells in V4 [32]. Read-out from units similar to the C_2 units in Fig. 1 predicted recent read-out experiments in monkey IT cortex [33], showing very similar selectivity and invariance for the same set of stimuli.

The model in its initial version [14] used a very simple *static* dictionary of handcrafted features. It was suggested that features from intermediate and higher layers in the model should instead be learned from visual experience. Here, we extend the model by showing how to learn a vocabulary of visual features from images and applying it to the recognition of real-world object-categories. Preliminary results previously appeared in several conference proceedings [34], [35], [36].

2 Detailed Implementation

S_1 *units*: Our system is summarized in Fig. 1. A gray-value input image is first analyzed by a multidimensional array of simple S_1 units which correspond to the classical simple cells of Hubel and Wiesel found in the primary visual cortex (V1) [11]. S_1 units take the form of Gabor functions [2], which have been shown to provide a good model of cortical simple cell receptive fields [3] and are described by the following equation:

$$F(x,y) = \exp\left(-\frac{(x_o^2+\gamma^2 y_o^2)}{2\sigma^2}\right) \times \cos\left(\frac{2\pi}{\lambda}x_o\right), \quad \text{s.t.} \qquad (1)$$

$$x_o = x\cos\theta + y\sin\theta \text{ and } y_o = -x\sin\theta + y\cos\theta. \qquad (2)$$

All filter parameters, i.e., the aspect ratio, $\gamma = 0.3$, the orientation θ, the effective width σ, the wavelength λ as well as the filter sizes s were adjusted so that the tuning properties of the corresponding S_1 units match the bulk of V1 parafoveal simple cells based on data from two groups [37], [38], [39], [40]. This was done by sampling the parameter space, applying the corresponding filters to stimuli commonly used to probe cortical cells (i.e., gratings, bars, and edges) and selecting the parameter values that capture the tuning properties of the bulk of V1 simple cells (see Table 1 and [41] for details). We arranged the S_1 filters to form a pyramid of scales, spanning a range of sizes from 7×7 to 37×37 pixels in steps of two pixels. To keep the number of units tractable, we considered four orientations (0°, 45°, 90°, and 135°), thus leading to 64 different S_1 receptive field types total (16 scales $\times$ 4 orientations).

C_1 *units*: The next, C_1, stage corresponds to cortical complex cells which show some tolerance to shift and size: Complex cells tend to have larger receptive fields (twice as large as simple cells), respond to oriented bars or edges anywhere within their receptive fields (tolerance to position), and tend to be more broadly tuned than simple cells (tolerance to size) [11]. C_1 units pool over retinotopically organized afferent S_1 units from the previous layer with the same orientation and from the same *scale band* (see Table 1). Each scale band contains two adjacent filter sizes (there are eight scale bands for a total of 16 S_1 filter sizes). For instance, scale band 1 contains S_1 filters with sizes 7×7 and 9×9. The scale band index of the S_1 units also determines the size of the S_1 neighborhood $N_S \times N_S$ over which the C_1 units pool. Again, this process is done for each of the four orientations and each scale band independently.

This pooling increases the tolerance to 2D transformations from layer S_1 to C_1. The corresponding pooling operation is a

1. In this paper, we used a Gaussian function but, as discussed in [15], a bell-shaped tuning function could also be approximated via a normalized dot-product.

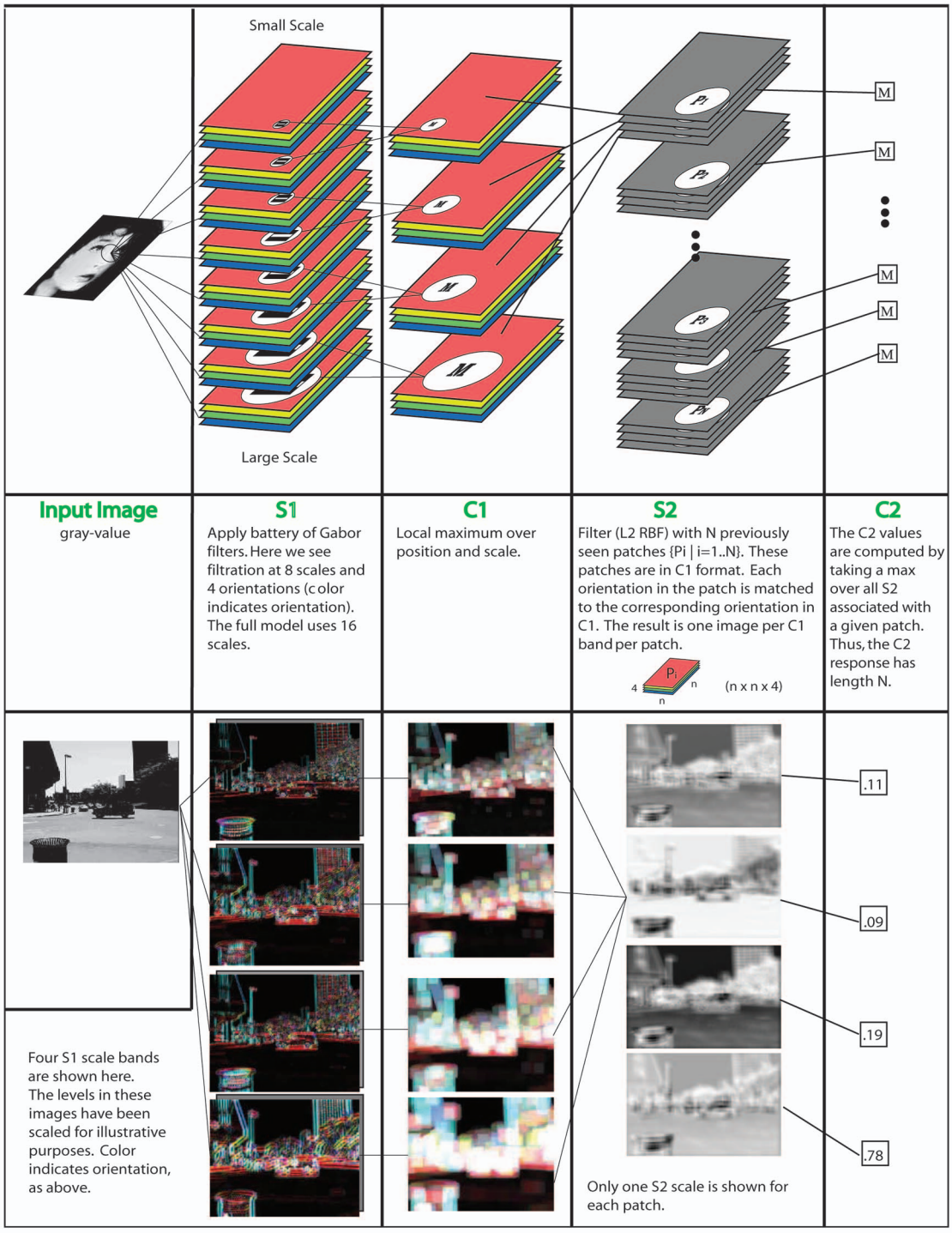

Fig. 1. System overview: A gray-value input image is first analyzed by an array of S_1 units at four different orientations and 16 scales. At the next C_1 layer, the image is subsampled through a local MAX (M) pooling operation over a neighborhood of S_1 units in both space and scale, but with the same preferred orientation. In the next stage, S_2 units are essentially RBF units, each having a different preferred stimulus. Note that S_2 units are tiled across all positions and scales. A MAX pooling operation is performed over S_2 units with the same selectivity to yield the C_2 unit responses.

MAX operation. That is, the response r of a complex unit corresponds to the response of the strongest of its m afferents $(x_1, \ldots, x_m)$ from the previous S_1 layer such that:

$$r = \max_{j=1\ldots m} x_j. \tag{3}$$

Consider, for instance, the first band: $\mathcal{S} = 1$. For each orientation, it contains two S_1 maps: The one obtained using a filter of size 7×7 and the one obtained using a filter of size 9×9 (see Table 1). The maps have the same dimensionality but they are the outputs of different filters. The C_1 unit responses are computed by subsampling these maps using a cell grid of size $N_{\mathcal{S}} \times N_{\mathcal{S}} = 8 \times 8$. From each grid cell, one single measurement is obtained by taking the maximum of all 64 elements. As a last stage, we take a max over the two scales from within the same spatial neighborhood, by recording

TABLE 1
Summary of the S_1 and C_1 *SMFs* Parameters

C_1 layer			S_1 layer		
Scale band $\mathcal{S}$	Spatial pooling grid ($N_S \times N_S$)	Overlap $\Delta_{\mathcal{S}}$	filter size s	Gabor σ	Gabor λ
Band 1	8×8	4	7×7	2.8	3.5
			9×9	3.6	4.6
Band 2	10×10	5	11×11	4.5	5.6
			13×13	5.4	6.8
Band 3	12×12	6	15×15	6.3	7.9
			17×17	7.3	9.1
Band 4	14×14	7	19×19	8.2	10.3
			21×21	9.2	11.5
Band 5	16×16	8	23×23	10.2	12.7
			25×25	11.3	14.1
Band 6	18×18	9	27×27	12.3	15.4
			29×29	13.4	16.8
Band 7	20×20	10	31×31	14.6	18.2
			33×33	15.8	19.7
Band 8	22×22	11	35×35	17.0	21.2
			37×37	18.2	22.8

only the maximum value from the two maps. Note that C_1 responses are not computed at every possible locations and that C_1 units only overlap by an amount $\Delta_{\mathcal{S}}$. This makes the computations at the next stage more efficient. Again, parameters (see Table 1) governing this pooling operation were adjusted such that the tuning of the C_1 units match the tuning of complex cells as measured experimentally (see [41] for details).

S_2 units: In the S_2 layer, units pool over afferent C_1 units from a local spatial neighborhood across all four orientations. S_2 units behave as radial basis function (RBF) units.[2] Each S_2 unit response depends in a Gaussian-like way on the Euclidean distance between a new input and a stored prototype. That is, for an image patch $\mathbf{X}$ from the previous C_1 layer at a particular scale S, the response r of the corresponding S_2 unit is given by:

$$r = \exp\left(-\beta \|\mathbf{X} - \mathbf{P_i}\|^2\right), \tag{4}$$

where β defines the sharpness of the TUNING and $\mathbf{P}_i$ is one of the N features (center of the RBF units) learned during training (see below). At runtime, S_2 maps are computed across all positions for each of the eight scale bands. One such multiple scale map is computed for each one of the $(N \sim 1,000)$ prototypes $\mathbf{P}_i$.

C_2 units: Our final set of shift- and scale-invariant C_2 responses is computed by taking a global maximum ((3)) over all scales and positions for each S_2 type over the entire S_2 lattice, i.e., the S_2 measures the match between a stored prototype $\mathbf{P_i}$ and the input image at every position and scale; we only keep the value of the best match and discard the rest. The result is a vector of N C_2 values, where N corresponds to the number of prototypes extracted during the learning stage.

The learning stage: The learning process corresponds to selecting a set of N prototypes $\mathbf{P}_i$ (or features) for the S_2 units. This is done using a simple sampling process such that, during training, a large pool of prototypes of various sizes and at random positions are extracted from a target set of images. These prototypes are extracted at the level of the C_1 layer across all four orientations, i.e., a patch P_o of size $n \times n$ contains $n \times n \times 4$ elements. In the following, we extracted patches of four different sizes $(n = 4, 8, 12, 16)$. An important question for both neuroscience and computer vision regards the choice of the unlabeled target set from which to learn—in an unsupervised way—this vocabulary of visual features. In the following, features are learned from the positive training set for each object independently, but, in Section 3.1.2, we show how a *universal* dictionary of features can be learned from a random set of natural images and shared between multiple object classes.

The Classification Stage: At runtime, each image is propagated through the architecture described in Fig. 1. The C_1 and C_2 standard model features (*SMFs*) are then extracted and further passed to a simple linear classifier (we experimented with both SVM and boosting).

3 Empirical Evaluation

We evaluate the performance of the *SMFs* in several object detection tasks. In Section 3.1, we show results for detection *in clutter* (sometimes referred to as weakly supervised) for which the target object in both the training and test sets appears at variable scales and positions within an unsegmented image, such as in the *CalTech101* object database [21]. For such applications, because 1) the size of the image to be classified may vary and 2) because of the large variations in appearance, we use the scale and position-invariant C_2 *SMFs* (the number N of which is independent of the image size and only depends on the number of prototypes learned during training) that we pass to a linear classifier trained to perform a simple object present/absent recognition task.

In Section 3.2, we evaluate the performance of the *SMFs* in conjunction with a *windowing* approach. That is, we extract a large number of fixed-size image windows from an input image at various scales and positions, which each have to be classified for a target object to be present or absent. In this task, the target object in both the training and test images exhibits a limited variability to scale and position (lighting and within-class appearance variability remain) which is accounted for by the scanning process. For this task, the presence of clutter within each image window to be classified is also limited. Because the size of the image windows is fixed, both C_1 and C_2 *SMFs* can be used for classification. We show that, for such an application, due to the limited variability of the target object in position and scale and the absence of clutter, C_1 *SMFs* appear quite competitive.

In Section 3.3, we show results using the *SMFs* for the recognition of *texture-based* objects like trees and roads. For this application, the performance of the *SMFs* is evaluated at every pixel locations from images containing the target object which is appropriate for detecting amorphous objects in a scene, where drawing a closely cropped bounding box is often impossible. For this task, the C_2 *SMFs* outperform the C_1 *SMFs*.

3.1 Object Recognition in Clutter

Because of their invariance to scale and position, the C_2 *SMFs* can be used for weakly supervised learning tasks for which a labeled training set is available but for which the training set is not normalized or segmented. That is, the target object is presented in clutter and may undergo large

2. This is consistent with well-known response properties of neurons in primate inferotemporal cortex and seems to be the key property for learning to generalize in the visual and motor systems [42].

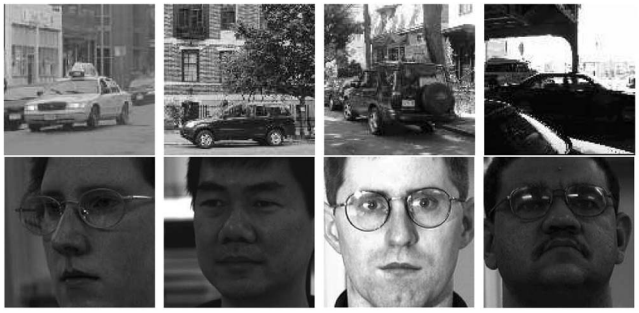

Fig. 2. Sample images from the MIT-CBCL multiview car [18] and face [17] data sets.

changes in position and scales. Importantly, the number of C_2 features depends only on the number of patches extracted during training and is independent of the size of the input image. Here, to perform different categorization tasks, the C_2 responses computed over a new input image are simply passed to a linear classifier (linear SVM or boosting).[3]

Below, we compare the performance of the scale and translation-invariant C_2 features when used as inputs to simple linear classifiers with other benchmark systems for the recognition of objects in clutter (i.e., both training and testing are performed on unsegmented images). We consider three data sets, denoted *CalTech5*, *CalTech101*, and *MIT-CBCL*, to evaluate our system performance.

3.1.1 Image Data Sets

CalTech5: We consider five of the databases,[4] i.e., the frontal-face, motorcycle, rear-car, and airplane data sets from [20], as well as the leaf data set from [19]. On these data sets, we used the same fixed splits as in the corresponding studies whenever applicable and otherwise generated random splits. All images were rescaled to be 140 pixels in height (width was rescaled accordingly so that the image aspect ratio was preserved) and converted to gray scale.

CalTech101: It contains 101 object classes plus a background class (see [21] for details). All results reported were generated with 10 random splits. In the binary experiments, we used 50 negative training examples and a variable number of positive training examples (1, 3, 15, 30, and 40). For testing, we selected 50 negative examples and 50 positive examples from the remaining images (or as many left if less than 50 were available). In the multiclass experiment, we used 15 or 30 training images per class. This includes the background class and the "faces" and "faces-easy" as three of the classes. We used as many as 50 testing examples per class, less if there were not enough examples left after training. If less than 50 examples were used, the results were normalized to reflect equal contributions for each class. We report the mean and standard deviation of the performance across all classes. All images were rescaled to be 140 pixels in height (width was rescaled accordingly so that the image aspect ratio was preserved) and converted to gray scale.

MIT-CBCL: This includes a near-frontal ($\pm 30^\circ$) face data set [17] and a multiview car data set from [18] (see Fig. 2). The face data set contains about 6,900 positive and 13,700 negative images for training and 427 positive and 5,000 negative images for testing. The car data set contains 4,000 positive and 1,600 negative training examples and 1,700 test examples (both positive and negative). Although the *benchmark* algorithms were trained on the full sets and the results reported accordingly, our system only used a subset of the training sets (500 examples of each class only).

These two MIT-CBCL data sets are challenging: The face patterns used for testing are a subset of the CMU PIE database [44], which contains a large variety of faces under extreme illumination conditions (see [17]). The test nonface patterns were selected by a low-resolution LDA classifier as the most similar to faces (the LDA classifier was trained on an independent 19×19 low-resolution training set). The car database includes a wide variety of vehicles, including SUVs, trucks, buses, etc., under wide pose and lighting variations. Random image patterns at various scales that were not labeled as vehicles were extracted and used as a negative test set.

TABLE 2
Results Obtained with 1,000 C_2 Features Combined with SVM or GentleBoost (*boost*) Classifiers and Comparison with Existing Systems (*Benchmark*)

Datasets	Benchmark	C_2 features	
		boost	SVM
Leaves [19]	84.0	**97.0**	95.9
Cars [20]	84.8	99.7	**99.8**
Faces [20]	96.4	**98.2**	98.1
Airplanes [20]	94.0	**96.7**	94.9
Motorcycles [20]	95.0	**98.0**	97.4
Faces [17]	90.4	**95.9**	95.3
Cars [18]	75.4	**95.1**	93.3

3.1.2 Results

Comparison with benchmark systems: Table 2 summarizes the performance of the C_2 *SMFs* compared with other published results from benchmark systems: the constellation models by Perona et al. [19], [20], the hierarchical SVM-based face-detection system by Heisele et al. [17] and a standard system [18] that uses Ullman et al.'s fragments [26] and gentleBoost as in [45]. The performance measure reported is the accuracy at the equilibrium point, i.e., the accuracy point such that the false positive rate equals the miss rate. Results obtained with the C_2 *SMFs* are superior to previous approaches [17], [18] on the MIT-CBCL data sets and comparable to the best systems [46], [47] on the *CalTech5* data sets.[5]

Comparison with SIFT features: We also compared the C_2 *SMFs* to a system based on Lowe's SIFT features [27]. To perform this comparison at the feature level and ensure a fair comparison between the two systems, we neglected all position information recovered by Lowe's algorithm. It was recently suggested in [47] that structural information does not seem to help improve recognition performance. We selected 1,000 random reference key-points from the training set. Given a new image, we measured the minimum distance between all its key-points and the 1,000 reference key-points, thus obtaining a feature vector of size 1,000.[6]

3. More biologically plausible classifiers are described in [43]. Such classifiers are likely to correspond to the task-specific circuits in the cortex from IT to PFC (see [15], [43]).

4. Available at http://www.robots.ox.ac.uk/vgg/data3.html.

5. Experimental procedures may vary from one group to another (e.g., splits used, preprocessing, scale normalization, etc.). Comparisons should therefore be taken cautiously.

6. Lowe recommends using the ratio of the distances between the nearest and the second closest key-point as a similarity measure. We found instead that the minimum distance leads to better performance than the ratio.

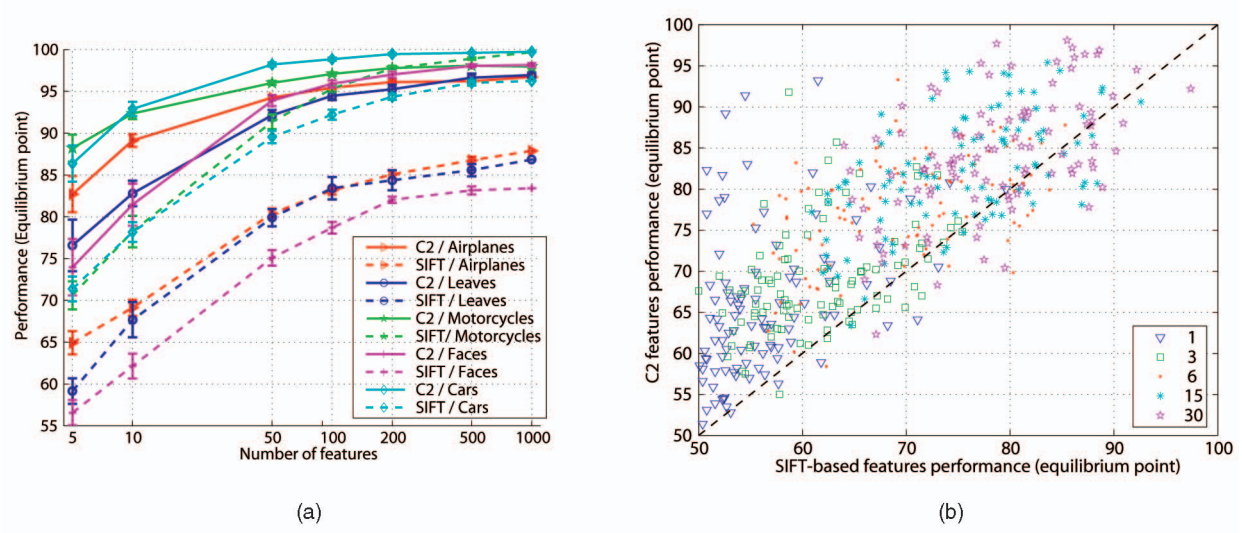

Fig. 3. Comparison between the SIFT and the C_2 features on the *CalTech5* for (a) different numbers of features and on the (b) *CalTech101* for a different number of training examples.

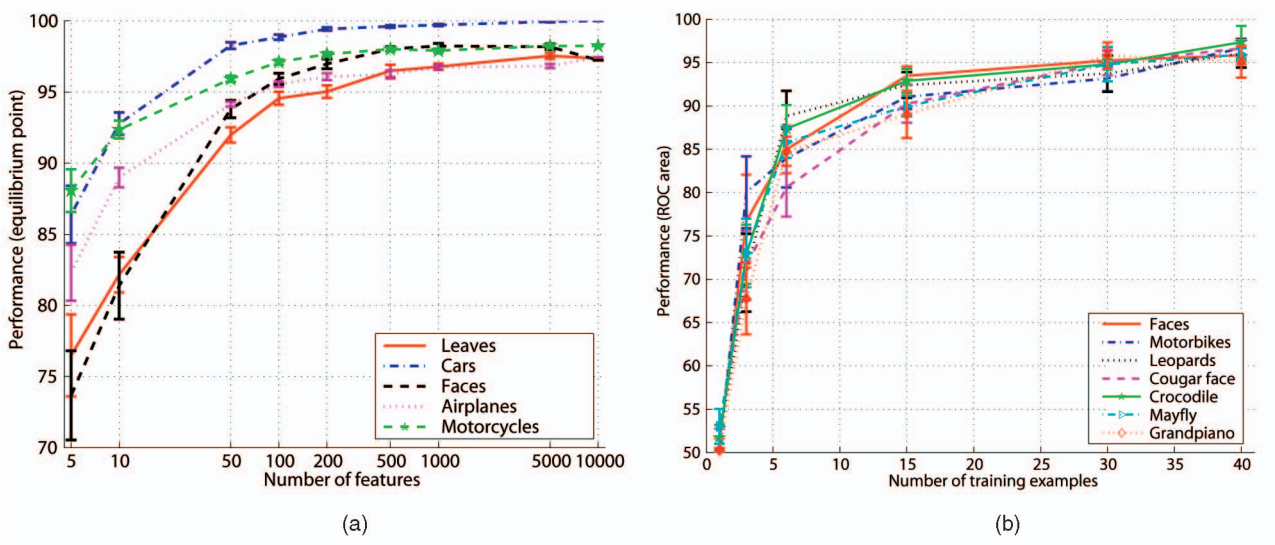

Fig. 4. Performance obtained with gentleBoost and different numbers of C_2 features on the (a) *CalTech5* and on sample categories from the (b) *CalTech101* for a different number of training examples.

Fig. 3 shows a comparison between the performance of the SIFT and the C_2 *SMFs* (both with gentleBoost; similar results were obtained with a linear SVM). Fig. 3a shows a comparison on the *CalTech5* database for different numbers of features (obtained by selecting a random number of them from the 1,000 available) and Fig. 3b on the *CalTech101* database for different number of training examples. In both cases, the C_2 features outperform the SIFT features significantly. SIFT features excel in the redetection of a transformed version of a previously seen example, but may lack selectivity for a more general categorization task at the basic level.

Number of features and training examples: To investigate the contribution of the number of features on performance, we first created a set of 10,000 C_2 *SMFs* and then randomly selected subsets of various sizes. The results reported are averaged over 10 independent runs. As Fig. 4a shows, while the performance of the system can be improved with more features (e.g., the whole set of 10,000 features), reasonable performance can already be obtained with 50-100 features. Features needed to reach the plateau (about 1,000-5,000 features) is much larger than the number used by current systems (on the order of 10-100 for [17], [26], [45] and 4-8 for constellation approaches [19], [20], [21]). This may come from the fact that we only sample the space of features and do not perform any clustering step like other approaches (including an earlier version of this system [34]). We found clustering to be sensitive to the choice of parameters and initializations, leading to poorer results.

We also studied the influence of the number of training examples on the performance of the system on the *CalTech101* database. For each object category, we generated different positive training sets of size 1, 3, 6, 15, and 30 as in [21] (see Section 3.1.1). As shown in Fig. 4b, the system achieves error rates comparable to [21] on a few training examples (less than 15), but its performance still improves with more examples (where the system by Fei-Fei et al. seems to be reaching a plateau, see [21]). Results with an SVM (not shown) are similar, although the performance tended to be higher on very few training examples (as SVM seems to avoid overfitting even for one example).

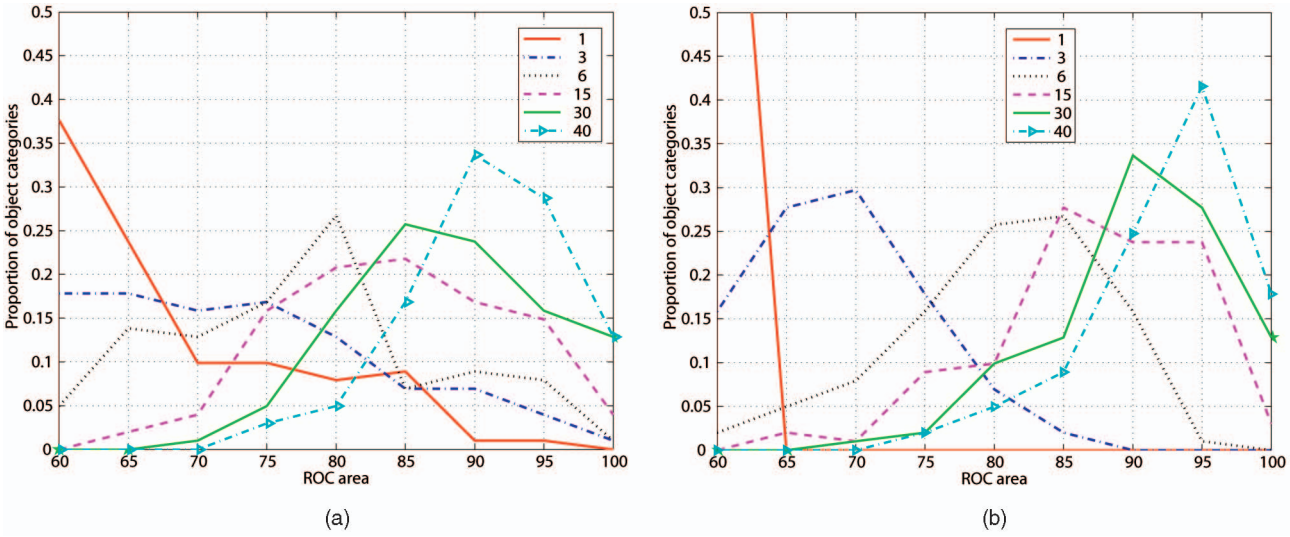

Fig. 5. Histogram of the C_2 features performance for all 101-object categories with (a) linear SVM and (b) gentleBoost for different numbers of positive training examples.

However, since SVM does not *select* the relevant features, its performance tends to be lower than gentleBoost as the number of training examples increases. Fig. 5 shows the performance of the gentleBoost and SVM classifiers used with the C_2 *SMFs* on all categories and for various numbers of training examples (each result is an average of 10 different random splits). Each plot is a single histogram of all 101 scores, obtained using a fixed number of training examples, e.g., with 40 examples, the gentleBoost-based system gets around 95 percent ROC area for 42 percent of the object categories.

Toward a universal dictionary of features: We here describe experiments that suggest that it is possible to perform robust object recognition with C_2 *SMFs* learned from a separate set of randomly selected natural images. In Fig. 6, we compare the performance of two sets of features on the *CalTech101* database: 1) a standard set of *object-specific* features that were learned from a training set of images from the target object category (200 features per training image) and 2) a *universal* set of 10,000 features learned independently from a set of random natural images (downloaded from the Web). While the *object-specific* set performs significantly better with enough training examples, the universal set appears to be competitive for smaller training sets.

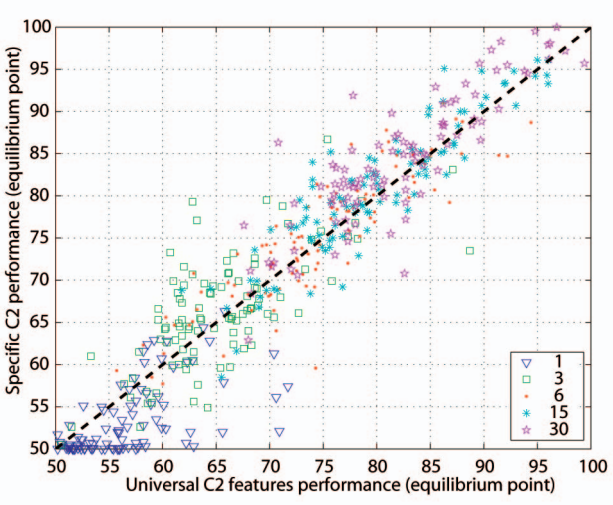

Fig. 6. Object-specific versus universal C_2 features.

Indeed the *universal* feature set is less prone to overfitting with few training examples (both the learning of the features and the training of the final classifier are performed on the same set with the *object-specific* set). In addition, contrary to the *object-specific* set, the size of the *universal* set is constant regardless of the number of training examples (10,000). As a result, with small training data sets, fewer features can be used with the object-specific set (we found that extracting more than 200 features per training image had very little effect on performance). This may constitute a relevant and intriguing result on its own. Our results also suggest that it should be possible for biological organisms to acquire a *basic vocabulary of features early in development while refining it with more specific features later on*. The latter point is consistent with reports of plasticity in inferotemporal cortex from adult monkey (the complexity and sizes of the largest C_2 features are consistent with the receptive fields of posterior IT neurons).

Multiclass results on the CalTech101: Finally, we report results on multiclass classification on the *CalTech101* database. To conduct this experiment, we use the *universal* dictionary of 1,000 features similar to the one described earlier. This offers a significant gain in speed in a multiclass setting compared to the standard *object-specific* set. The classifier is a multiclass linear SVM that applied the all-pairs method and is trained on 102 labels (101 categories plus the background category). The performance of the system reaches above 44 $\pm$ 1.14 percent correct classification rate when using 15 training examples per class averaged over 10 repetitions (see Section 3.1.1). Using only five training images per class, the performance degrades to $\sim$ 30 percent.

By considering *gestalt*-like features (e.g., good-continuity detectors, circularity detectors, and symmetry detectors) within the same framework in addition to the C_2 *SMFs*, Wolf et al. obtained 51.2 percent $\pm$ 1.2 percent correct [48], [49] and recently incorporated some changes with Sharat Chikkerur to get 55.0 percent $\pm$ 0.9 percent (all these results are for 15 training images). At press time, some of the best

TABLE 3
The *StreetScenes* Database

Type	shape-based			texture-based			
Object	car	ped.	bic.	bldg.	tree	road	sky
# labeled ex.	5799	1449	209	5067	4932	3400	2562

systems include the system in [50] ($\approx$ 44 percent correct) and in [51] (45 percent correct).

3.2 Object Recognition without Clutter

3.2.1 The StreetScenes Database

In order to test the *SMFs* on a challenging real-world object detection problem, we have built training and test data from the *StreetScenes* scene-understanding data set. This database consists of more than 3,000 labeled images of the streets around Boston and Cambridge. Sample images and their hand labelings are illustrated in Fig. 10; some statistics of the content of the data set are given in Table 3. The accurate detection of many of these object categories is made difficult by the wide internal variability in their appearance. For example, the object class *cars* includes examples of many diverse models, at many poses, and in various types of occlusion and lighting, *trees* appear very different in summer and winter, and the class of *buildings* includes skyscrapers as well as suburban houses. Capturing this wide variability while maintaining high accuracy is part of the challenge of the scene-understanding problem. The database is available online at [16].

3.2.2 Training the SMFs-Based Systems

Using data extracted from our *StreetScenes* database, we trained object detectors for the classes *car*, *pedestrian*, and *bicycle*. This data was extracted by cropping out labeled examples of these object classes. Negative examples were extracted similarly by finding locations and scales which matched the positive data, but did not overlap the labeled positives. Each example, positive and negative, was resized to 128×128 pixels and converted to gray scale. This image was then converted into C_1 space using the method of Section 2. For a 128×128 gray-scale image and our parameter values, this resulted in a feature vector of 13,362 features that provided the input to the C_1-based classifier. The C_2 representation was built as in Section 3.1 for the recognition in clutter. Classifiers for these objects were trained using gentleBoost For these experiments, all labeled positive examples and 10 times as many negative examples were extracted. The systems evaluation was performed using randomized training (1/3) and testing (2/3) splits..

3.2.3 Benchmark Systems

For comparison, we also implemented four other benchmark systems. Our most simple baseline detector is a single-template *Grayscale* system: Each image is normalized in size and histogram equalized before the gray-values are passed to a linear classifier (gentleBoost). Another baseline detector, *Local Patch Correlation*, is built using patch-based features similar to [45]. Each feature f_i is associated with a particular image patch p_i, extracted randomly from the training set. Each feature f_i is calculated in a test image as the maximum normalized cross correlation of p_i within a subwindow of the image. This window of support is equal to a rectangle three times the size of p_i and centered in the image at the same relative location from which p_i was originally extracted. The advantage of the patch-based features over the single-template approach is that local patches can be highly selective while maintaining a degree of position invariance. The system was implemented with $N = 1,024$ features and with patches of size 12×12 in images of size 128×128. The third benchmark system is a *Part-based system* as described in [25]. Briefly, both object parts and a geometric model are learned via image patch clustering. The detection stage is performed by redetecting these parts and allowing them to vote for objects-at-poses in a generalized Hough transform framework. Finally, we compare to an implementation of the Histogram of Gradients (HoG) feature of [52], which has shown excellent performance on these types of objects. All benchmark systems were trained and tested on the same data sets as the *SMFs*-based system. They all use gentleBoost except [25].

3.2.4 Results

The ROC results of this experiment are illustrated in Fig. 7. For the two (C_1 and C_2) *SMFs*-based systems, the *Grayscale* as well as the *Local Patch Correlation* system, the classifier is gentleBoost, but we found very similar results with both a linear and a polynomial-kernel SVM. Overall, for all three object categories tested, the *SMFs*-based system performs best on cars and bicycles and second behind HoG on pedestrians (the HoG system was parameter-tuned in [52] to achieve maximal performance on this one class). Finally, for this recognition task, i.e., with a windowing framework, the C_1 *SMFs* seem to be superior to the C_2 *SMFs*. Indeed, the C_1 *SMFs* are adept at representing the object boundaries of these *shape-based* objects, which have strong interexample correspondence.

3.3 Object Recognition of Texture-Based Objects

Here, we demonstrate the utility of the *SMFs* in a texture-based object recognition task. Performance is measured by considering each pixel, rather than each instance of an object, to be a separate example. We consider four texture-based objects: buildings, trees, roads, and skies.

3.3.1 Training the SMFs-Based Systems

In building a database of labeled texture examples, we were careful to avoid errors due to overlap and loose polygonal labeling in the *StreetScenes* database. Because of object occlusions, some pixels in the database are labeled as one object, i.e., *building*, but their actual appearance is due to another object, i.e., *tree*. We addressed this by removing pixels with either multiple labels or no labe, from the test. Additionally, training samples were never drawn from within 15 pixels of any object's border. The same training and test locations were used for both the *SMFs*-based and the benchmark systems.

To build the C_1 *SMFs*-based system, C_1 maps were computed for each image and, for each sample point, feature vector elements were collected by sampling the resulting C_1 maps at a set of relative locations and scale-bands. A C_2 *SMF*-based system was also built as in Section 3.1 except for the maximum over position at the S_2 level that was taken over a local neighborhood instead of the whole image. This local area corresponded to a 60×60 pixel window in the original $960 \times 1,280$ pixel image.

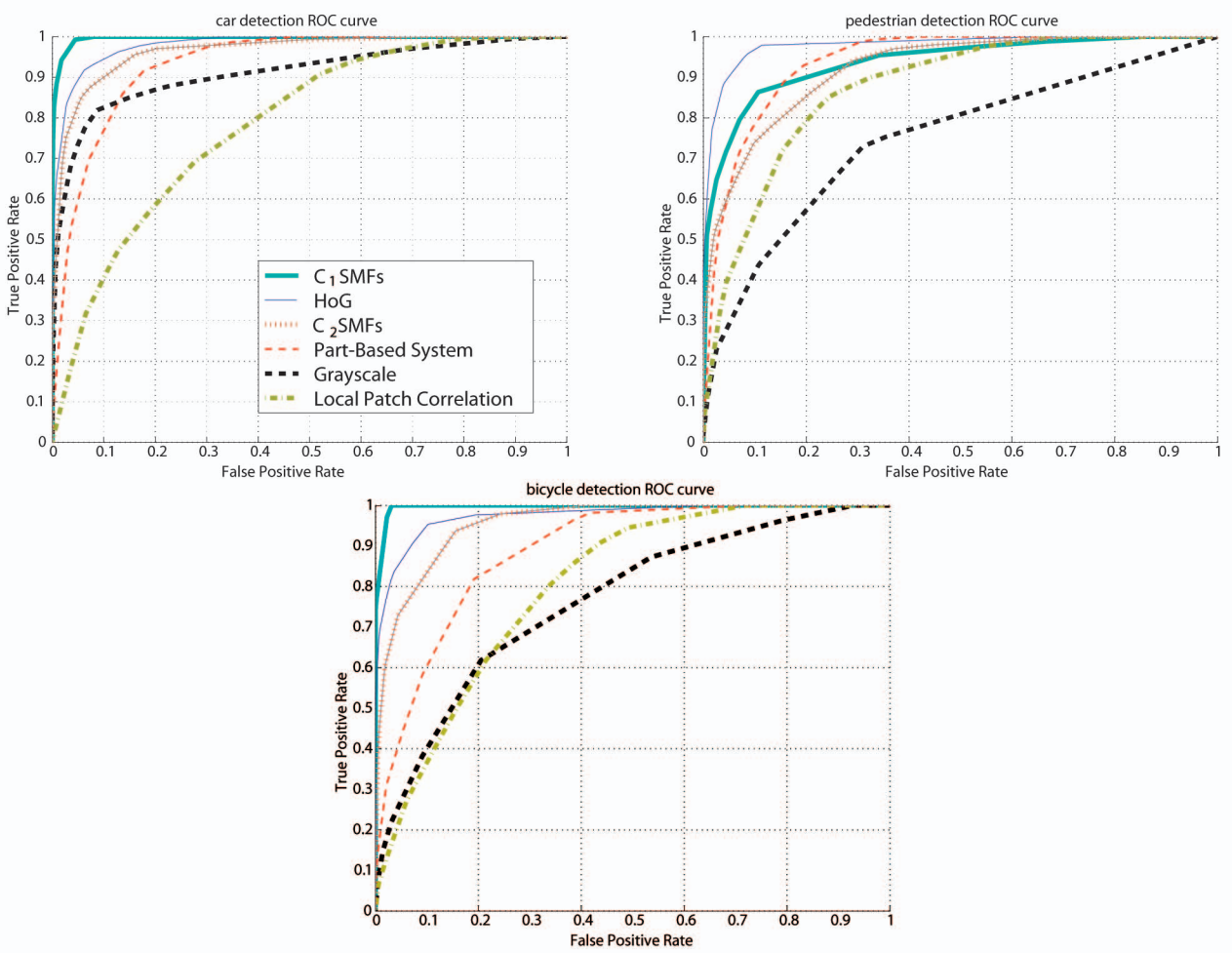

Fig. 7. ROC curves illustrating the performance of the standard-model object-detectors compared to four baseline systems (see Section 3.2). Note that, in this test, the amount of clutter is limited by the windowing process, creating better interexample correspondence and thereby allowing the direct application of the C_1 *SMFs*.

3.3.2 Benchmark Systems

We implemented four benchmark texture classification systems. The *Blobworld* (*BW*) system was constructed as described in [53]. Briefly, the Blobworld feature, originally designed for image segmentation, is a six-dimensional vector at each pixel location; three dimensions encode color in the Lab color space and three dimensions encode texture using the local spectrum of gradient responses. We did not include the color information for a fair comparison between all the various texture detection methods.

The systems labeled $T1$ and $T2$ are based on [29]. In these systems, the test image is first processed with a number of predefined filters. $T1$ uses 36 oriented edge-filters arranged in five degrees increments from 0 degrees to 180 degrees. $T2$ follows [29] exactly by using 36 Gabor filters at six orientations, three scales, and two phases. For both systems independently, a large number of random samples of the 36-dimensional edge response images were taken and subsequently clustered using k-means to find 100 cluster centroids (i.e., the *textons*). The *texton image* was then calculated by finding the index of the nearest texton to the filter response vector at each pixel in the response images. A 100-dimensional texton feature vector was then built by calculating the local 10×10 histogram of nearest texton indexes.

Finally, the Histogram of edges (*HoE*) system was built by simply using the same type of histogram framework, but over the local 36-dimensional directional filter responses (using the filters of $T1$) rather than the texton identity. Here, as well, learning was done using the gentleBoost algorithm (again a linear SVM produced very similar results). The within-class variability of the texture-objects in this test is considerably larger than that of the texture classes usually used to test texture-detection systems, making this task somewhat different. This may explain the relatively poor performance of some of these systems on certain objects.

3.3.3 Results

As shown in Fig. 8, the SMFs-based texture system seems to consistently outperform the benchmarks (*BW*, $T1$, $T2$, and *HoE*). C_2 compared to C_1 *SMFs* may be better suited to this task because of their increased invariance properties and complexity.

3.4 Toward a Full System for Scene Understanding

The *SMFs*-based object detection systems described previously were combined into a complete system for scene understanding. The objects to be detected are divided into two distinct categories, *texture-based* objects and *shape-based* objects, which are handled using different recognition

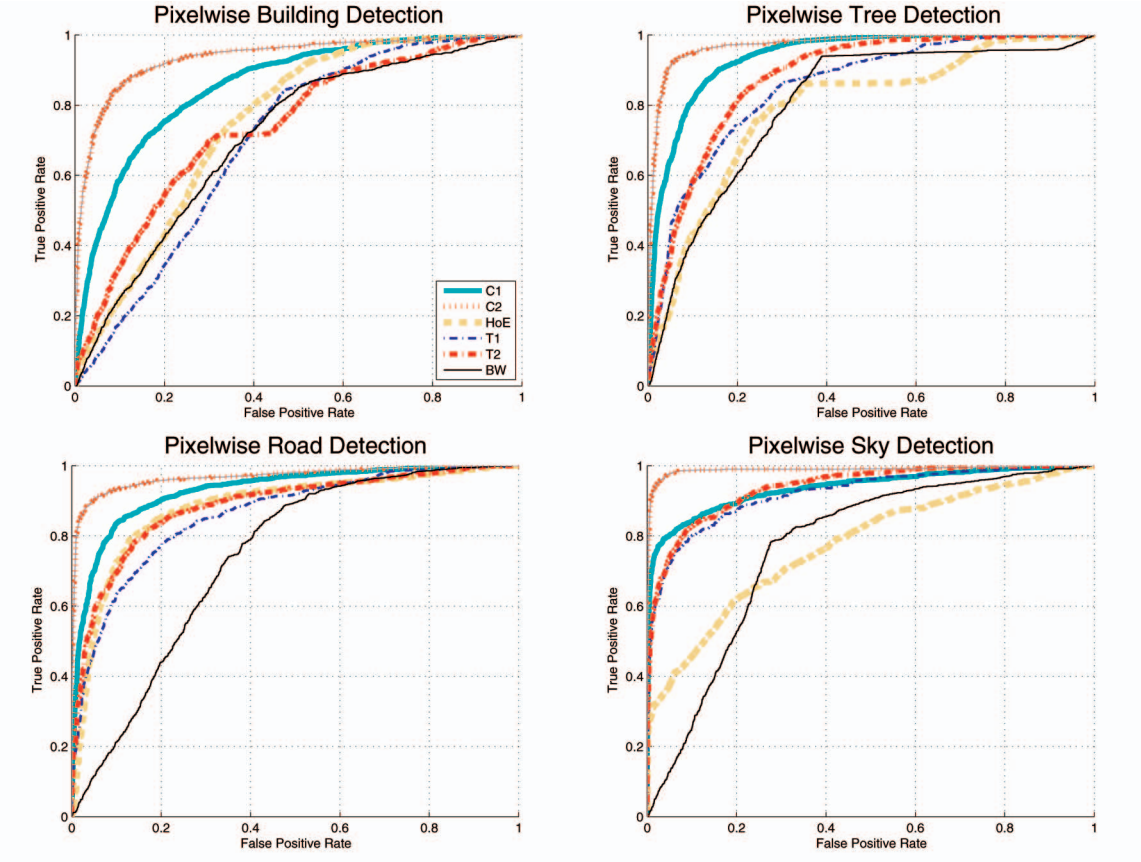

Fig. 8. Performance (ROC curves) of five texture classification algorithms for the detection of buildings, trees, skies, and roads. This texture classification task requires reliable recognition of texture classes with wide intraclass variability. This difficult test may in part explain the inferior performance of the benchmark algorithms, which have been previously used to detect object boundaries and classify materials, but not for object recognition.

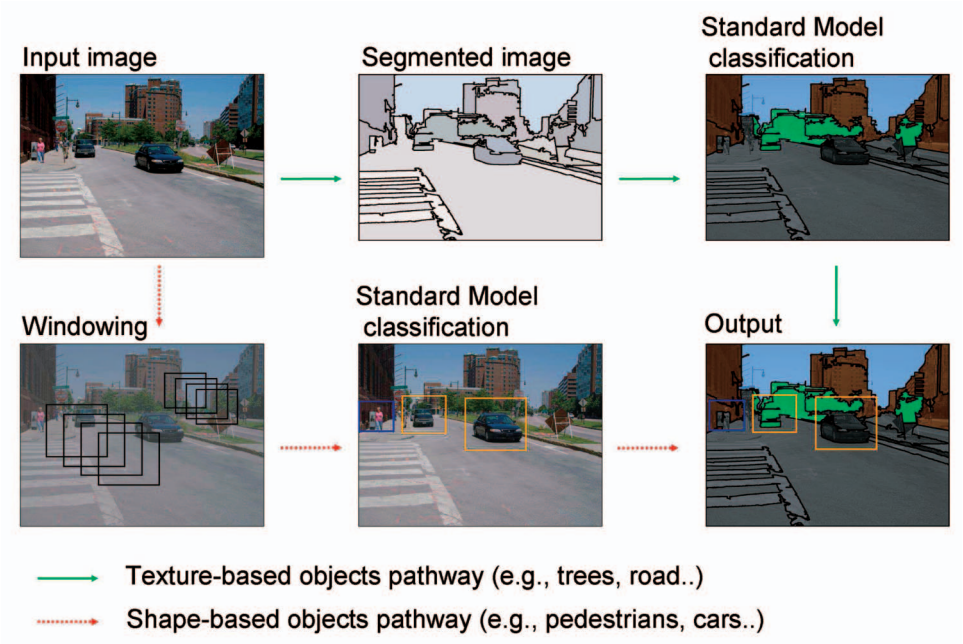

Fig. 9. Data flow diagram of the scene-understanding system (see text for details).

strategies. Fig. 9 illustrates the architecture of the data flow diagram, specifically highlighting the two pathways for the detection of the texture-based and shape-based objects.

3.4.1 *Shape-Based Object Detection in StreetScenes*

Shape-based objects are those objects for which there exists a strong part-to-part correspondence between examples,

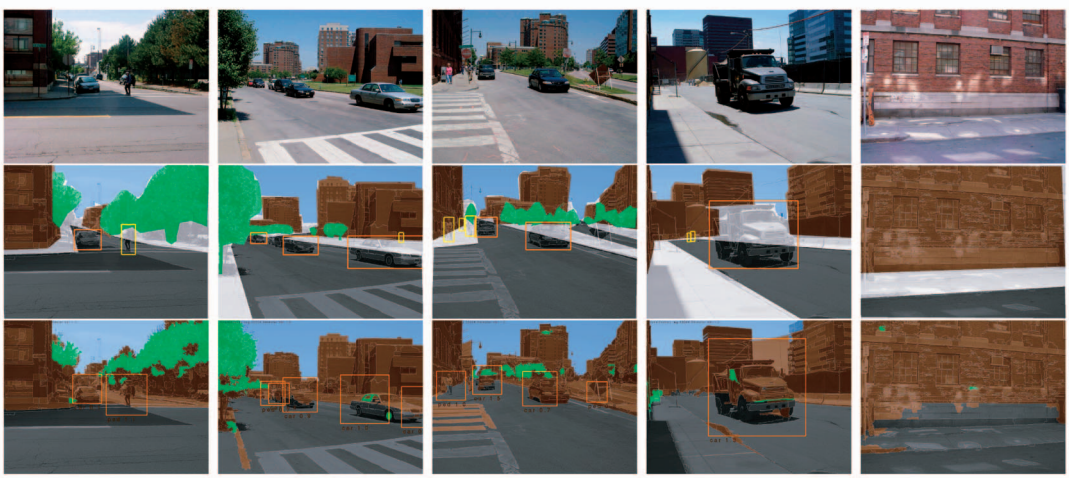

Fig. 10. Top row: Sample *StreetScenes* examples. Middle row: True hand-labeling; color overlay indicates texture-based objects and bounding rectangles indicate shape-based objects. Note that pixels may have multiple labels due to overlapping objects or no label at all (indicated in white). Bottom row: Results obtained with a system trained on examples like (but not including) those in the second row.

including pedestrians, cars, and bicycles. In order to detect shape-based objects, a standard windowing technique is used. This contrasts with the approach presented in Section 3.1, wherein objects in clutter are detected using scale and translation-invariant C_2 *SMFs*, rather than testing for object presence at each position and scale independently. While the C_2 approach is appropriate for fast decisions of object presence or absence, it would be impractical for this scene-understanding application as the locations of individual objects would be lost.

In conjunction with this windowing approach, we use the C_1 SMFs. Since the window crops away much of the clutter, leaving the potential object nearly centered, the additional invariance from the C_2 features is not necessary. It is important to note that the good performance of the C_1 *SMFs* is dependent upon training data with accurate descriptions of the position and scale of the target object. Performance metrics for both C_1 and C_2 *SMFs* were shown in Section 3.2, as well as those for a number of benchmark systems. In the final system, the classifiers output is thresholded and a standard local neighborhood suppression technique is used in which the maximum detection is recorded and the response within a neighborhood in scale space is suppressed. In Fig. 10, we present some sample results obtained with the system.

3.4.2 Pixel-Wise Detection of Texture-Based Objects

Texture-based objects are those objects for which, unlike shape-based objects, there is no obvious visible interobject part-wise correspondence. These objects are better described by their texture rather than the geometric structure of reliably detectable parts. For the *StreetScenes* database, these currently include buildings, roads, trees, and skies.

Using the models trained in Section 3.3 and applying them to each pixel within the image, one obtains a detection confidence map of the size of the original image for each object. This map is used to judge which pixel belongs to which texture-object category. Simply taking the value with maximum response strength results in unsatisfactory results, as it was found that, when the receptive field of a unit overlaps a texture-boundary, the response becomes unreliable. This was addressed by smoothing the anomalous responses by segmenting the input image and averaging the responses of the detectors over each segment. As a result, uncertain responses at the object borders are compensated for by the more numerous responses within the object boundaries. This was accomplished using the segmentation software *Edison* [54]. Sample results of our texture recognition system can be seen in the bottom row of Fig. 10.

4 Discussion

4.1 A Computer Vision Perspective on the Model

The computer vision system described in this work was constructed from a neuroscience model of the primate visual cortex which is a rather unusual approach. The model itself is based on a consensus among neuroscientists and on fitting available experimental data. Still, one may wonder about the relationships between the *SMFs* and other computer vision algorithms: Because of the hierarchical and nonlinear nature of the architecture described in Fig. 1, there is little hope in finding a simple general cost function that the system would minimize. These types of functions are seldom available for hierarchical systems which are not probabilistic in nature or explicitly set out to minimize an energy function. Instead, we next study each layer of the system separately.

The first layer (S_1) consists of applying Gabor filters to the input image, which mimics the processing by simple cells in the primary visual cortex (V1). Gabor filters have been around in computer vision for decades, starting with Gabor's demonstration [2] that these elementary functions minimize the uncertainty of their product and Daugman's extension [55] to 2D. They are also very similar to DoG filters used since the 1960s to model receptive fields in the retina and primary visual cortex and to perform edge detection in computer vision (see [56], [57]). Bovik et al. [58] used Gabor filters for texture segmentation and Sanger [59] for the computation of disparity in stereovision. In biometrics, it has been used for face recognition (e.g., [60]), iris recognition as well as fingerprint recognition. Olshausen and Fields demonstrated that optimizing a sparse coding scheme over a set of natural images produces a set of edge filters similar to Gabor filters [61]. Hence, it was expected that the output of Gabor filters on natural images would be sparse. This

Fig. 11. Max scale space images of Lena (top row) and of the Gabor filtered version of Lena (bottom row). While the gray-value image gets distorted, the information in the sparse edge image is enhanced.

Fig. 12. Gaussian scale space images of Lena (top row) and of the Gabor filtered version of Lena (bottom row). While the gray-value image degrades gracefully, revealing structures at different scales, the sparse edge image fades away.

result comes from the fact that Gabor filters, as edge detecting filters, are activated only near image edges. Any further analysis step done on top of Gabor filters should take this sparseness property into account.

The next layer (C_1) does something which is unorthodox for computer vision systems—it maximizes the output of the filters locally. Each C_1 unit computes the maximum over a small pool of S_1 outputs. While many systems maximize the output of a detector over the entire image, local maximization has only been done recently. For part-based object detection [17], [26], [45], local detectors of each part are learned independently, and are then applied to local regions where the parts are expected to appear.

Our work seems novel in that general purpose filters are being maximized over uniformly distributed local regions in the image. In order to understand this stage, we can invoke some scale space terminology (see [62] for an overview). Scale space theory was mostly concerned at first with the Gaussian scale space. This scale space has many desirable properties such as separability, linearity, shift invariance, isotropy, homogeneity, and causality. The last property is an important one: Causality means that no new level sets are created by going into coarser scales. A related property is to demand the noncreation of local extrema in coarser scales. In our application, a local maximization (instead of Gaussian blurring) is used to go from a fine to a coarser scale in order to make the C_1 layer invariant to small local translations. As a pseudoscale space, local maximization has some desirable properties: It is separable (one can apply it over the rows and then over the columns), it is shift invariant, and it is homogeneous (it can be applied in the same way to each scale; applying it repeatedly corresponds to moving into coarser and coarser scales). However, in general it is not an appropriate scale space— among other problems, when applying it to an image, new local extrema are being created. This can be seen in the top row of Fig. 11, where applying the max scale space to the Lena image creates block-like structures, which are new level sets, and where the corners are new local maxima.

However, our application of the local maximum operator is on the Gabor filtered image, which is a sparse representation of the original image. For such an input, the Gaussian scale space results in a diluted image (see bottom row of Fig. 12). The max scale space, on the other hand, is successful in keeping the sparse inputs through the consecutive applications of the max filter. Put differently, for the analysis of gray-level images, it is important not to create new structures while moving to coarser scales: In this, a Gaussian scale space is appropriate and a local maximum type of analysis is not. For the analysis of sparse coding, it is important to conserve the local maxima, which is precisely what the maximum operator does (the Gaussian scale space on the other hand flattens the input).

The next two levels in our system involve the combination of C_1 outputs using a template matching approach. Prototype templates (*patches*) are extracted from the training images and the best match with these serves as an image representation. The first template-based stage (S_2) measures the "correlation" (Euclidean distance) of the C_1 maps with many small crops obtained from such maps.

The correlation is measured for the four orientations together, thus making our algorithm sensitive to large rotations of the image. Small rotations can be approximated by small translations, which are handled by the maximization at the C_1 level. Note that this stage is done at multiple scales such that a given template taken from a C_1 map at a

certain scale during the prototype templates collection stage of training is matched across all C_1 maps when constructing the C_2 feature vector. The last stage of our system (C_2) is a standard maximization over the entire image (in principle and more biologically, this would be over an area of the size of the fovea but not the whole visual field, see [15]). This is equivalent to scanning over all locations and scales for the maximum correlation with each C_1 template selected in training.

4.2 What SMFs to Use for Which Tasks?

To summarize our main results: In Section 3.1, we have shown an application to the C_2 *SMFs* to the semisupervised recognition of objects in clutter. For such tasks, the training images are unsegmented: The target object is embedded in clutter and undergo changes in scale and position. Additionally, because the training images come in different sizes, only a global representation based on a fixed-length scale and position-invariant feature vector such as the C_2 *SMFs* is suitable.

As described in Section 3.2 for the recognition of shape-based objects in conjunction with a scanning approach (the images to be classified are segmented and normalized), a more "holistic" representation based on C_1 *SMFs* which are adept at detecting object boundaries tend to be superior. For such tasks, the variations in scale and position are limited and clutter is almost completely absent. As a result, the scale and position-invariance of the C_2 *SMFs* does not bring any extra computational benefit.

Finally, in Section 3.3, we showed that the C_2 *SMFs* excel at the recognition of texture-based objects which lack a geometric structure of reliably detectable parts in comparison to the C_1 *SMFs* as well as other benchmark systems.

4.3 Object Recognition in Cortex: Remarks

Our system belongs to a family of feedforward models of object recognition in the cortex that have been shown to be able to duplicate the tuning properties of neurons in several visual cortical areas [14]. In particular, Riesenhuber and Poggio showed that such a class of models accounts quantitatively for the tuning properties of view-tuned units in IT cortex which respond to images of the learned object more strongly than to distractor objects, despite significant changes in position and size [63]. Model performance was so far only reported for simple artificial stimuli such as paperclips on a uniform background [14], with no real-world image degradations such as change in illumination, clutter, etc. The success of our extension of the original model on a variety of large-scale real-world's object recognition databases provides a compelling plausibility proof for this class of feed-forward models.

A long-time goal for computer vision has been to build a system that achieves human-level recognition performance. Until now, biology had not suggested a good solution. In fact, the superiority of human performance over the best artificial recognition systems has continuously lacked a satisfactory explanation. The computer vision approaches had also diverged from biology: For instance, some of the best existing computer vision systems use geometrical information about objects' constitutive parts (the constellation approaches [19], [20], [21] rely on a probabilistic shape model; in [17], the position of the facial components is passed to a combination classifier (along with their associated detection values) whereas biology is unlikely to be able to use it—at least in the cortical stream dedicated to shape processing and object recognition). The system described in this paper may be the first counterexample to this situation: It is based on a model of object recognition in cortex [14], [15], it respects the properties of cortical processing (including the absence of geometrical information) while showing performance at least comparable to the best computer vision systems.

It has been suggested that "immediate recognition" during scene categorization tasks may rely on partial processing by the visual system based on a rapid and parallel detection of disjunctive sets of unbound features of the target category [64], [65]. Interestingly a recent psychophysical experiment [66] suggested that spatial information about the objects location may be absent during "immediate recognition." That is, even though human observers correctly detect a target object within a frame embedded in a rapid sequence of images, they are, however, not able to recover even its approximate location [66]. Such an observation is in good agreement with the experiment described in Section 3.1 in which the recognition of objects in clutter is based on a *bag* of translation and scale-invariant C_2 features computed over the entire image for which spatial information is lost. Indeed, we recently showed that an extension of the model described in this paper accounts for the level and the pattern of performance of human observers [43] on a rapid animal versus nonanimal categorization task [67]. This may be the first time that a neurobiological model, faithful to the physiology and the anatomy of the visual cortex, provides a realistic alternative to engineered artificial vision systems.

4.4 Open Questions, Limitations, and Possible Improvements

4.4.1 *Have We Reached the Limit of What a/this Feedforward Architecture Can Achieve in Terms of Performance?*

There seem to be at least three directions that could be followed to further improve the performance of the architecture described here: First, very recent experiments [43] suggests that the addition of extra layers (e.g., S_3, C_3, S_4, etc.), in agreement with the anatomy and physiology of the visual cortex, may provide a significant gain in performance. Additionally, we also found that loosening the hierarchy described in Fig. 1 may also provide some significant computational benefits. As already suggested by the results of our experimental simulations in Section 3, not all tasks are equal. Depending on the amount of clutter and 2D transformations involved, it is sometimes beneficial to use the fine information from low-level *SMFs* and some other times to use more invariant high-level *SMFs*. We found that passing different types of *SMFs* to the final classifier and letting the classifier choose for the optimal features may further improve performance (for instance, passing both C_1 and C_2 *SMFs*) [43], [48].

Second, the sampling procedure we used here to learn features is very simple. It is likely that not all features are useful for recognition. Applying a standard feature selection technique may give further improvement in performance. Indeed, a very recent study showed that selecting the subset of the C_2 *SMFs* that are highly weighted by the SVM classifier provide a substantial increase in performance [68].

Third, for all the tests reported here, we did not tune a single parameter to get optimal performance. Instead, model parameters were set to match what is known about the primate visual system. Further improvements could likely be obtained by tuning some of the model parameters [69] (see Table 1)—perhaps through learning.

4.4.2 Beyond Feedforward Architectures

As a feedforward model of the ventral stream pathway, the architecture of Fig. 1 cannot account for all aspects of our everyday vision which involve eye movements and top-down effects, which are mediated by higher brain centers and the extensive anatomical back-projections found throughout the visual cortex and are not implemented in the present feedforward model. While our system exhibits competitive performance compared to other benchmark systems, it remains limited compared to biological visual systems: The model seems to be able to account for the level of performance of human observers on a rapid categorization task [67] when the stimulus presentation times are short and back-projections are inactive [43]. Yet the performance of the model remains far behind the performance of human observers for long presentation times.

It is important to point out that this recognition with a *glimpse* only constitutes the initial processing step in natural vision. In particular, the model—in its present form—does not capture Gestalt-like properties such as continuity and parallelism or figure-ground segmentation, which probably involves lateral and feedback connections, yet to be inserted in the model. A feedforward system (like the one we presented here) could, in principle, be used as the front-end of a visual system as part of a prediction-verification loop [70]. The feedforward path would provide an initial hypothesis about what object is presented in the visual field, yet to be verified through feedback loops.

4.4.3 Future Work

Perhaps the major limitation of our system in a real-world applications setting remains its processing speed (limited by the S_1 and C_1 stages)—typically, tens of seconds, depending on the size of the input image—which is too slow for a real-time application. Another important question, yet to be addressed, is whether the recognition results obtained with bags of C_2 features could be extended to other tasks, such as face and gesture recognition or the analysis of video.

4.5 Conclusion

In this paper, we have described a new framework for robust object recognition, which we have applied to two different recognition scenarios: First, we have shown an application to the problem of semisupervised recognition of objects in clutter that does not involve image scanning. The system first computes a set of scale and translation-invariant C_2 features from a training set of images, which is then passed to a standard classifier on the vector of features obtained from the input image. The system was tested on several object databases and shown to outperform several more complex benchmark systems (e.g., the systems in [19], [20], [21] involve the estimation of probability distributions; [17] uses a hierarchy of SVMs and requires accurate correspondences between positive training images, i.e., 3D head models). Interestingly, the approach was shown to be able to learn from a few examples and could compete with generative models that use prior category information [21].

Second, we have described a new approach to scene understanding with an application to a *StreetScenes* database involving different types of rigid objects as well as texture-based objects. We found that the Standard Model Features (SMFs) constitute a flexible framework that can be used in conjunction with standard computer vision techniques, i.e., image scanning for the detection and localization of several target objects at multiple scales and image segmentation for the recognition of nonrigid texture-based objects.

Acknowledgments

The authors would like to thank Sharat Chikkerur and Timothee Masquelier for useful comments on this manuscript. This report describes research done at the Center for Biological & Computational Learning, which is in the McGovern Institute for Brain Research at MIT, the Department of Brain & Cognitive Sciences, and the Computer Sciences & Artificial Intelligence Laboratory (CSAIL). This research was sponsored by grants from: the US Defense Advanced Research Projects Agency (B. Yoon), US Office of Naval Research (DARPA), US National Science Foundation-National Institutes of Health (CRCNS). Additional support was provided by: Daimler-Chrysler AG, Eastman Kodak Company, Honda Research Institute USA, Inc., Komatsu Ltd., Oxygen, Siemens Corporate Research, Inc., Sony, Sumitomo Metal Industries, Toyota Motor Corporation, and the Eugene McDermott Foundation.

References

[1] D. Marr and T. Poggio, "A Computational Theory of Human Stereo Vision," *Proc. Royal Soc. London B,* vol. 204, pp. 301-328, 1979.
[2] D. Gabor, "Theory of Communication," *J. IEE,* vol. 93, pp. 429-459, 1946.
[3] J.P. Jones and L.A. Palmer, "An Evaluation of the Two-Dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex," *J. Neurophysiology,* vol. 58, pp. 1233-1258, 1987.
[4] K. Fukushima, "Neocognitron: A Self Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position," *Biological Cybernetics,* vol. 36, pp. 193-201, 1980.
[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE,* vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
[6] B.W. Mel, "SEEMORE: Combining Color, Shape, and Texture Histogramming in a Neurally Inspired Approach to Visual Object Recognition," *Neural Computation,* vol. 9, pp. 777-804, 1997.
[7] S. Thorpe, "Ultra-Rapid Scene Categorisation with a Wave of Spikes," *Proc. Biologically Motivated Computer Vision,* pp. 1-15, 2002.
[8] H. Wersing and E. Koerner, "Learning Optimized Features for Hierarchical Models of Invariant Recognition," *Neural Computation,* vol. 15, no. 7, pp. 1559-1588, 2003.
[9] Y. Amit and M. Mascaro, "An Integrated Network for Invariant Visual Detection and Recognition," *Vision Research,* vol. 43, no. 19, pp. 2073-2088, 2003.
[10] Y. LeCun, F.J. Huang, and L. Bottou, "Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.
[11] D.H. Hubel and T.N. Wiesel, "Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex," *J. Physiology,* vol. 160, pp. 106-154, 1962.
[12] D. Perrett and M. Oram, "Neurophysiology of Shape Processing," *Image and Vision Computing,* vol. 11, pp. 317-333, 1993.
[13] G. Wallis and E. Rolls, "A Model of Invariant Object Recognition in the Visual System," *Progress in Neurobiology,* vol. 51, pp. 167-194, 1997.

[14] M. Riesenhuber and T. Poggio, "Hierarchical Models of Object Recognition in Cortex," *Nature Neuroscience,* vol. 2, no. 11, pp. 1019-1025, 1999.

[15] T. Serre, M. Kouh, C. Cadieu, U. Knoblich, G. Kreiman, and T. Poggio, "A Theory of Object Recognition: Computations and Circuits in the Feedforward Path of the Ventral Stream in Primate Visual Cortex," AI Memo 2005-036/CBCL Memo 259, Massachusetts Inst. of Technology, Cambridge, 2005.

[16] A software implementation of the system as well as the StreetScenes data set, http://cbcl.mit.edu/software-datasets, 2006.

[17] B. Heisele, T. Serre, M. Pontil, T. Vetter, and T. Poggio, "Categorization by Learning and Combining Object Parts," *Advances in Neural Information Processing Systems,* vol. 14, 2002.

[18] B. Leung, "Component-Based Car Detection in Street Scene Images," master's thesis, Dept. of Electrical Eng. and Computer Science, Massachusetts Inst. of Technology, 2004.

[19] M. Weber, W. Welling, and P. Perona, "Unsupervised Learning of Models of Recognition," *Proc. European Conf. Computer Vision,* vol. 2, pp. 1001-1108, 2000.

[20] R. Fergus, P. Perona, and A. Zisserman, "Object Class Recognition by Unsupervised Scale-Invariant Learning," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 264-271, 2003.

[21] L. Fei-Fei, R. Fergus, and P. Perona, "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories," *Proc. IEEE Conf. Computer Vision and Pattern Recognition, Workshop Generative-Model Based Vision,* 2004.

[22] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a Similarity Metric Discriminatively, with Application to Face Verification," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[23] B. Leibe and B. Schiele, *Interleaved Object Categorization and Segmentation,* pp. 759-768, Sept. 2003, citeseer.csail.mit.edu/leibe04interleaved.html.

[24] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection in Images by Components," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, pp. 349-361, 2001.

[25] B. Leibe, A. Leonardis, and B. Schiele, "Combined Object Categorization and Segmentation with an Implicit Shape Model," *Proc. SLCP '04 Workshop Statistical Learning in Computer Vision,* 2004.

[26] S. Ullman, M. Vidal-Naquet, and E. Sali, "Visual Features of Intermediate Complexity and Their Use in Classification," *Nature Neuroscience,* vol. 5, no. 7, pp. 682-687, 2002.

[27] D.G. Lowe, "Object Recognition from Local Scale-Invariant Features," *Proc. Int'l Conf. Computer Vision,* pp. 1150-1157, 1999.

[28] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, 2002.

[29] L.W. Renninger and J. Malik, "When Is Scene Recognition Just Texture Recognition," *Vision Research,* vol. 44, pp. 2301-2311, 2002.

[30] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *Proc. Int'l Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 257-263, June 2003, http://lear.inrialpes.fr/pubs/2003/MS03.

[31] I. Lampl, D. Ferster, T. Poggio, and M. Riesenhuber, "Intracellular Measurements of Spatial Integration and the MAX Operation in Complex Cells of the Cat Primary Visual Cortex," *J. Neurophysiology,* vol. 92, pp. 2704-2713, 2004.

[32] T.J. Gawne and J.M. Martin, "Responses of Primate Visual Cortical V4 Neurons to Simultaneously Presented Stimuli," *J. Neurophysiology,* vol. 88, pp. 1128-1135, 2002.

[33] C. Hung, G. Kreiman, T. Poggio, and J. DiCarlo, "Fast Read-Out of Object Identity from Macaque Inferior Temporal Cortex," *Science,* vol. 310, pp. 863-866, Nov. 2005.

[34] T. Serre, J. Louie, M. Riesenhuber, and T. Poggio, "On the Role of Object-Specific Features for Real World Recognition in Biological Vision," *Proc. Workshop Biologically Motivated Computer Vision,* pp. 387-397, 2002.

[35] T. Serre, L. Wolf, and T. Poggio, "Object Recognition with Features Inspired by Visual Cortex," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[36] S. Bileschi and L. Wolf, "A Unified System for Object Detection, Texture Recognition, and Context Analysis Based on the Standard Model Feature Set," *Proc. British Machine Vision Conf.,* 2005.

[37] R. DeValois, D. Albrecht, and L. Thorell, "Spatial Frequency Selectivity of Cells in Macaque Visual Cortex," *Vision Research,* vol. 22, pp. 545-559, 1982.

[38] R. DeValois, E. Yund, and N. Hepler, "The Orientation and Direction Selectivity of Cells in Macaque Visual Cortex," *Vision Research,* vol. 22, pp. 531-544, 1982.

[39] P.H. Schiller, B.L. Finlay, and S.F. Volman, "Quantitative Studies of Single-Cell Properties in Monkey Striate Cortex III. Spatial Frequency," *J. Neurophysiology,* vol. 39, no. 6, pp. 1334-1351, 1976.

[40] P.H. Schiller, B.L. Finlay, and S.F. Volman, "Quantitative Studies of Single-Cell Properties in Monkey Striate Cortex II. Orientation Specificity and Ocular Dominance," *J. Neurophysiology,* vol. 39, no. 6, pp. 1334-1351, 1976.

[41] T. Serre and M. Riesenhuber, "Realistic Modeling of Simple and Complex Cell Tuning in the HMAX Model, and Implications for Invariant Object Recognition in Cortex," CBCL Paper 239/AI Memo 2004-017, Massachusetts Inst. of Technology, Cambridge, 2004.

[42] T. Poggio and E. Bizzi, "Generalization in Vision and Motor Control," *Nature,* vol. 431, pp. 768-774, 2004.

[43] T. Serre, "Learning a Dictionary of Shape-Components in Visual Cortex: Comparison with Neurons, Humans, and Machines," PhD dissertation, Massachusetts Inst. of Technology, Cambridge, Apr. 2006.

[44] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) Database of Human Faces," Computer Science Technical Report 01-02, Carnegie Mellon Univ., 2001.

[45] A. Torralba, K.P. Murphy, and W.T. Freeman, "Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[46] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, "Generic Object Recognition with Boosting," Technical Report TR-EMT-2004-01, Graz Univ. of Technology, 2004.

[47] S. Lazebnik, C. Schmid, and J. Ponce, "A Maximum Entropy Framework for Part-Based Texture and Object Recognition," *Proc. Int'l Conf. Computer Vision,* 2005.

[48] L. Wolf, S. Bileschi, and E. Meyers, "Perception Strategies in Hierarchical Vision Systems," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[49] S. Bileschi, "Streetscenes: Towards Scene Understanding in Still Images," PhD dissertation, Massachusettes Inst. of Technology, 2006.

[50] A. Holub, M. Welling, and P. Perona, "Exploiting Unlabelled Data for Hybrid Object Classification," *Proc. Neural Information Processing Systems, Workshop Inter-Class Transfer,* 2005.

[51] A. Berg, T. Berg, and J. Malik, "Shape Matching and Object Recognition Using Low Distortion Correspondence," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[52] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 886-893, June 2005, http://lear.inrialpes.fr/pubs/2005/DT05.

[53] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, and J. Malik, "Blobworld: A System for Region-Based Image Indexing and Retrieval," *Proc. Third Int'l Conf. Visual Information Systems,* 1999.

[54] C.M. Christoudias, B. Georgescu, and P. Meer, "Synergism in Low Level Vision," *Proc. Int'l Conf. Computer Vision,* vol. IV, pp. 150-155, Aug. 2002.

[55] J. Daugman, "Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two-Dimensional Visual Cortical Filters," *J. Optical Soc. Am. A,* vol. 2, pp. 1160-1169, 1985.

[56] D. Marr, S. Ullman, and T. Poggio, "Bandpass Channels, Zero-Crossings and Early Visual Information Processing," *J. Optical Soc. Am. A,* vol. 69, pp. 914-916, 1979.

[57] V. Torre and T. Poggio, "On Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, pp. 147-163, 1986.

[58] A. Bovik, M. Clark, and W. Geisler, "Multichannel Texture Analysis Using Localized Spatial Filters," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, pp. 55-73, 1990.

[59] T. Sanger, "Stereo Disparity Computation Using Gabor Filters," *Biological Cybernetics,* vol. 59, pp. 405-418, 1988.

[60] K. Okada, J. Steffens, T. Maurer, H. Hong, E. Elagin, H. Neven, and C. von der Malsburg, "The Bochum/USC Face Recognition System and How It Fared in the FERET Phase III Test," *Face Recognition: From Theory to Applications,* pp. 186-205, 1998.

[61] B. Olshausen and D. Field, "Emergence of Simple-Cell Receptive Field Properties by Learning a Sparse Code for Natural Images," *Nature,* vol. 381, pp. 607-609, 1996.

[62] T. Lindeberg, "Scale-Space Theory: A Framework for Handling Image Structures at Multiple Scales," *Proc. CERN School of Computing*, Egmond aan Zee, The Netherlands, 1996.
[63] N. Logothetis, J. Pauls, and T. Poggio, "Shape Representation in the Inferior Temporal Cortex of Monkeys," *Current Biology*, vol. 5, pp. 552-563, 1995.
[64] A.M. Treisman and G. Gelade, "A Feature-Integration Theory of Attention," *Cognitive Psychology*, vol. 12, pp. 97-136, 1980.
[65] J.M. Wolfe and S.C. Bennett, "Preattentive Object Files: Shapeless Bundles of Basic Features," *Vision Research*, vol. 37, no. 1, pp. 25-43, Jan. 1997.
[66] K. Evans and A. Treisman, "Perception of Objects in Natural Scenes: Is It Really Attention Free," *J. Experimental Psychology: Human Perception Performance*, vol. 31, no. 6, pp. 1476-1492, 2005.
[67] S. Thorpe, D. Fize, and C. Marlot, "Speed of Processing in the Human Visual System," *Nature*, vol. 381, pp. 520-522, 1996.
[68] J. Mutch and D. Lowe, "Multiclass Object Recognition Using Sparse, Localized Features," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006.
[69] M. Marin-Jimenez and N.P. de la Blanca, "Empirical Study of Multi-Scale Filter Banks for Object Categorization," *Proc. Int'l Conf. Pattern Recognition*, 2006.
[70] T. Lee and D. Mumford, "Hierarchical Bayesian Inference in the Visual Cortex," *J. Optical Soc. Am. A*, vol. 20, no. 7, pp. 1434-1448, 2003.

Thomas Serre received the diplome d'Ingénieur in telecommunications from the Ecole Nationale Supérieure des Télécommunications de Bretagne, Brest, France, in 2000, the DEA (MSc) degree from the University of Rennes, France, in 2000 and, in 2005, the PhD degree in neuroscience from the Massachusetts Institute of Technology (MIT). He is currently a postdoctoral associate in Professor Poggio's lab at MIT in Cambridge. His main research focuses on object recognition with both brains and machines.

Lior Wolf graduated from the Hebrew University, Jerusalem, where he worked under the supervision of Professor Shashua. He is a postdoctoral associate in Professor Poggio's lab at Massachusetts Institute of Technology (MIT). He won the Max Shlumiuk award for 2004 and the Rothchild fellowship for 2004. His joint work with Professor Shashua for ECCV 2000 received the best paper award, and their work for ICCV 2001 received the Marr prize honorable mention. His reseach interests include object-recognition, video-analysis, and structure-from-motion. In Spring 2006, he is expected to join the faculty of the Computer Science Department at Tel-Aviv University.

Stanley Bileschi is a postdoctoral associate in Professor Poggio's lab at Massachusetts Institute of Technology (MIT), where he received the graduate degree in 2006. Previously, he attended the State University of New York at Buffalo and earned degrees in computer science and electrical engineering. His graduate work was sponsored in part through a US National Science Foundation fellowship. His research interests include computer vision, contextual semantic understanding of video streams, and neurocomputation.

Maximilian Riesenhuber received the Diplom in physics from the University of Frankfurt, Germany, in 1995, and the PhD degree in computational neuroscience from the Massachusetts Institute of Technology in 2000. He is currently an assistant professor of neuroscience at Georgetown University Medical Center in Washington, D.C. His main research foci are the neural mechanisms underlying object recognition and plasticity in the normal brain and the translation to neural disorders, brain-machine interfaces, and machine vision. Dr. Riesenhuber has received several awards, including a McDonnell-Pew Award in Cognitive Neuroscience, Technology Review's TR100, and an US National Science Foundation CAREER Award.

Tomaso Poggio is the Eugene McDermott Professor in the Department of Brain and Cognitive Sciences, the Codirector, Center for Biological and Computational Learning, a member for the last 25 years of the Computer Science and Artificial Intelligence Laboratory at MIT, and, since 2000, a member of the faculty of the McGovern Institute for Brain Research. He is the author or coauthor of more than 400 papers in the fields of learning theory, computer science, computational neuroscience, and nonlinear systems theory and he belongs to the editorial board of several scientific journals. He is an honorary member of the Neuroscience Research Program, a member of the American Academy of Arts and Sciences, and a Founding Fellow of the AAAI. He received several awards such as the Otto-Hahn-Medaille Award of the Max-Planck-Society, the Max Planck Research Award (with M. Fahle), from the Alexander von Humboldt Foundation, the MIT 50K Entrepreneurship Competition Award, the Laurea Honoris Causa in Ingegneria Informatica for the Bicentenario dell'Invenzione della Pila from the University of Pavia, and the 2003 Gabor Award. His current research is focused on the development of the theory and on the application of novel learning techniques to computer vision, bioinformatics, computer graphics, and especially neuroscience. He is a member of the IEEE.