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 Δ_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 Δ_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:

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

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].