Object Indexing using an Iconic Sparse Distributed Memory*

Rajesh P.N. Rao and Dana H. Ballard
Department of Computer Science, University of Rochester
Rochester, NY 14627, USA
{rao,dana}@cs.rochester.edu

Abstract

A general-purpose object indexing technique is described that combines the virtues of principal component analysis with the favorable matching properties of high-dimensional spaces to achieve high precision recognition. An object is represented by a set of high-dimensional iconic feature vectors comprised of the responses of derivative of Gaussian filters at a range of orientations and scales. Since these filters can be shown to form the eigenvectors of arbitrary images containing both natural and man-made structures, they are well-suited for indexing in disparate domains. The indexing algorithm uses an active vision system in conjunction with a modified form of Kanerva's sparse distributed memory which facilitates interpolation between views and provides a convenient platform for learning the association between an object's appearance and its identity. The robustness of the indexing method was experimentally confirmed by subjecting the method to a range of viewing conditions and the accuracy was verified using a well-known model database containing a number of complex 3D objects under varying pose.

1 Introduction

The earliest models of objects for computer vision emphasized geometrical descriptions based on shape [20, 5]. Such descriptions are attractive as they are easily adapted for the manipulation requirements of robotic assembly tasks. However, they have proved very difficult to extract from the image owing to the fact that geometric and photometric properties are relatively uncorrelated. Insights gained from work on active/animate vision [1, 2, 4] seem to suggest that simpler iconic descriptions of objects based on their photometric properties may often suffice for many visual tasks [18].

This paper investigates the use of an iconic description comprised of photometric features at a local image patch as a medium for efficient object indexing in active vision systems. The photometric features are obtained by taking the responses of nine derivative-of-Gaussian filters at various orientations, each at five different scales. The derivative-of-Gaussian filters can be shown to arise as a result of unsupervised Hebbian learning by a neural network that performs principal component analysis on natural image patches during an initial "development" phase. An object can then be represented by a set of filter response vectors from different loci within the object for a small number of views sampled from the viewing sphere.

The process of object indexing itself is realized within the framework of an active vision system used in conjunction with a modified form of Kanerva's sparse distributed memory [10]; the memory facilitates interpolation between different views of an object and provides a convenient platform for learning the association between an object's appearance and its identity. Real-time performance is achieved by implementing both visual preprocessing *and* associative memory within a pipeline image processor and exploiting its ability to perform convolutions at frame-rate (Section 5).

Experimental results as presented in Section 6 indicate that the indexing scheme is remarkably tolerant to moderate changes in viewing conditions caused by occlusions, illumination changes, scale changes and rotations in 3D. The accuracy of the indexing method was verified on the well-known Columbia object database containing a number of arbitrary 3D objects with complex appearance characteristics; the method was able to attain a 100% recognition rate with a small number of iconic indexes per object.

2 Unsupervised Learning of Spatial Filters for Recognition

Typical natural stimuli are highly redundant containing statistical regularities that can be exploited for the purposes of visual coding. For example, in most images, nearby pixels tend to be highly correlated due to the morphological consistency of objects. Thus, some form of recoding into a more efficient representation is highly desirable. An optimal linear method for reducing redundancy is the Karhunen-Loéve transform or eigenvector expansion via Principal Component Analysis (PCA). Briefly, PCA generates a set of eigenvectors or *principal components* (orthogonal axes of projections) of a set of input images in the order of decreasing variance. Thus, by projecting new input only along the directions given by the dominant eigenvectors (i.e. those associated with the highest variance), significant data-compression can be achieved.

In recent years, there has been considerable interest in the use of PCA for both synthesis and analysis. For example, PCA has recently been applied quite successfully to synthesize basis functions for recognition of faces [23] and arbitrary 3D objects [15]. Researchers analyzing the human visual pathway have found PCA to be the crucial link between the profiles of cortical receptive fields and the statistics of natural images. Oja [16] first noted that a simple one-layer feedforward neural-network employing a form of the *Hebbian learning rule* acted as a principal component analyzer.

*This work is supported by NSF research grant no. CDA-8822724, NIH/PHS research grant no. 1 R24 RRO6853, and a grant from the Human Science Frontiers Program.

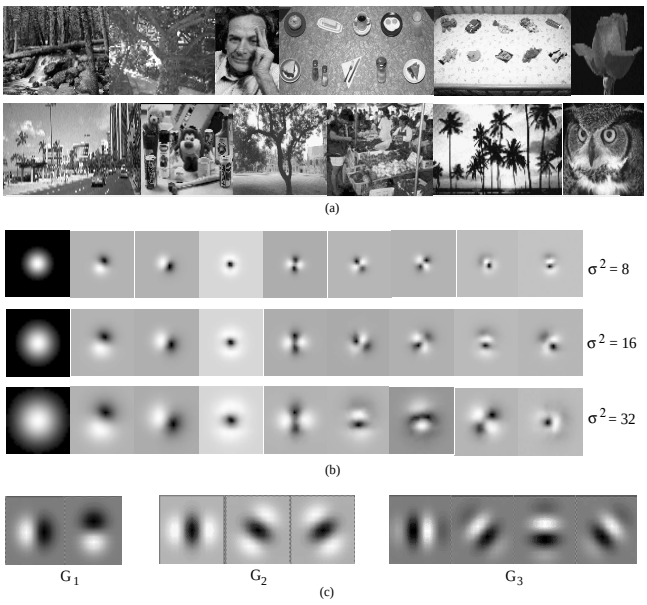


Figure 1: (a) Twelve of the 20 images that we used for training Sanger's PCA network. The network adapted its weights according to a form of the Hebbian learning rule in response to 12000 32×32 image patches obtained by scanning across the images. (b) First nine dominant eigenvectors that the weights of the network converged to, shown here for different scales (σ) of the Gaussian window (intensity is proportional to magnitude). (c) The Gaussian derivative basis functions of up to the third-order used in our iconic representations. The first few dominant eigenvectors of natural images shown in (b) closely resemble these analytically derived function profiles. Note however that we do not use the first eigenvector to avoid illumination dependence and additionally incorporate some non-orthogonal basis functions at the higher orders in order to achieve rotational invariance using the property of steerability. This choice also obviates using mixed derivatives (as in (b)) since the other oriented filters yield a complete basis.

Sanger [21] extended this work to obtain the first k principal components and noted that when iteratively applied to natural image patches, his network converged to approximations of oriented first- and second-derivative operators. Hancock et al. [9] used Sanger's network to extract the first few principal components of an ensemble of natural images windowed by a Gaussian in order to avoid the distortions that may have been caused by the use of square windows in Sanger's work. They observed that the eigenvectors that the network converged to were very close approximations of the different oriented derivative-of-Gaussian operators that have been shown to provide the best fit to primate cortical receptive field profiles among the different mathematical profiles suggested in the literature [24]. We employed Sanger's network to ascertain whether the results of Hancock et al. remained true for collections of images containing equal proportions of natural and man-made stimuli. The results, parts of which are shown in Figure 1 (b), confirmed that *regardless of the scale of analysis, the weight vectors of the network eventually converged to approximations of different Gaussian derivative operators.*

The oriented derivative-of-Gaussian operators can be regarded as an ideal set of *natural basis functions* for general-purpose recognition. Part of the rationale for this belief stems from the fact that these functions are obtained as a result of applying the principle of dimensionality-reduction to arbitrary collections of images containing a plethora of features from natural as well as man-made structures rather than just the images of particular objects or faces. By sacrificing specialization for a particular class of objects, we achieve wider applicability and by using fixed basis functions which were learned during an initial "development" phase, we avoid the high computational overhead involved in recomputing new basis functions upon the introduction of new objects as necessitated by previous methods [15, 23]. Further support for using the oriented derivative-of-Gaussian operators comes from the observation that correlation filters generated by principal component expansion maximize signal-to-noise ratio and yield much sharper correlation peaks than traditional raw image cross-correlation techniques (see, for instance, [13]). Finally, while it is relatively well-known that the class of functions that simultaneously minimize the product of the standard deviation of the spatial position sensitivity and spatial frequency sensitivity (as given by the uncertainty principle from Fourier theory) are the complex-Gabor elementary functions [8], a relatively lesser known fact is that the class of *real-valued* functions that minimize the above conjoint localization metric are in fact the Gaussian derivative functions as first noted by Gabor himself ([8] p. 441; see also [22]).

3 The Multiscale Iconic Index

Our iconic representation for objects is inspired by the existence of "natural basis functions" as outlined in the previous section. The current implementation uses nine Gaussian derivative basis filters denoted by:

$$G_n^{\theta_n}, n = 1, 2, 3, \theta_n = 0, \ldots, k\pi/(n+1), k = 1, \ldots, n \quad (1)$$

where n denotes the order of the filter and θ_n the orientation of the filter. Figure 1 (c) shows the basis filters for a particular scale.

The response of an image patch I centered at (x_0, y_0) to a particular basis filter $G_i^{\theta_j}$ can be obtained by convolving the image patch with the filter:

$$r_{i,j}(x_0, y_0) = \iint G_i^{\theta_j}(x_0 - x, y_0 - y) I(x, y) dx\, dy \quad (2)$$

The iconic index for a local image patch on an object can then be formed by combining into a single high-dimensional vector the responses of each of the nine basis filters at different scales:

$$\mathbf{r} = \left(r_{i,j,s}\right), i = 1, 2, 3; j = 1, .., i+1; s = s_{min}, .., s_{max} \quad (3)$$

where $r_{i,j,s}$ denotes the response of a filter with the index i denoting the order of the filter, j denoting the number of filters per order, and s denoting the number of different scales. In our experiments, we used five octave-separated scales.

An attractive property of the index is that it can be made rotation-invariant about the viewing axis when scale is unchanged. This can be done by exploiting the *steerability* [7] of the basis functions. First, a canonical orientation (say, horizontal) is assumed. Then, the orientation for a given vector of responses $\mathbf{r}$ can be computed from the two first-order responses as

$$\alpha = atan2(r_{1,1,s_{max}}, r_{1,2,s_{max}}) \quad (4)$$

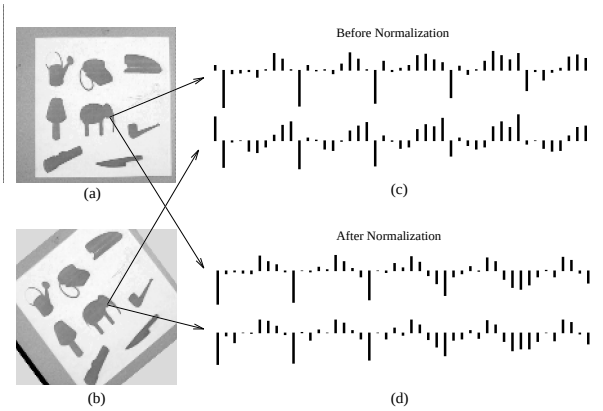


Figure 2: **Rotation Normalization**. (a) A test image; (b) The same image rotated 38° counterclockwise; (c) The response vectors for corresponding points near the elephant's mouth in the two images before normalization; (d) the response vectors after normalization (Positive responses are represented by upward bars proportional to the response magnitude and negative ones by downward bars with the nine smallest scale responses at the beginning and the nine largest ones at the end).

For normalization, the entire set of filter responses can be "rotated" to the canonical orientation using a set of interpolation functions as derived by Freeman and Adelson [7]

$$r'_{i,j,s} = \sum_{j'=1}^{i+1} r_{i,j',s} k_{j'i}(\alpha), \tag{5}$$

where $i = 1, 2, 3; j = 1, \ldots, i+1; s = s_{min}, \ldots, s_{max}$, and

$$k_{j'1}(\theta) = \frac{1}{2}\Big[2\cos(\theta - (j'-1)\pi/2)\Big], j' = 1, 2 \tag{6}$$

$$k_{j'2}(\theta) = \frac{1}{3}\Big[1 + 2\cos(2(\theta - (j'-1)\pi/3))\Big], j' = 1, 2, 3 \tag{7}$$

and

$$k_{j'3}(\theta) = \frac{1}{4}\Big[2\cos(\theta-(j'-1)\pi/4)+2\cos(3(\theta-(j'-1)\pi/4))\Big] \tag{8}$$

where $j' = 1, 2, 3, 4$. Figure 2 illustrates the rotation normalization procedure. It is can be seen that the two previously uncorrelated response vectors of the same point have been rendered almost identical after normalization.

4 Sparse Distributed Memory

For object indexing, the response vectors obtained from various objects need to be stored along with their associated labels. One way of accomplishing this is to use an associative memory. A model of associative memory that is specifically geared towards storage and retrieval of high-dimensional vectors is Kanerva's Sparse Distributed Memory (SDM) [10].

SDM was developed by Kanerva in an attempt to model human long-term memory. The model is based on the crucial observation that if concepts or objects of interest are represented by high-dimensional vectors, they can benefit

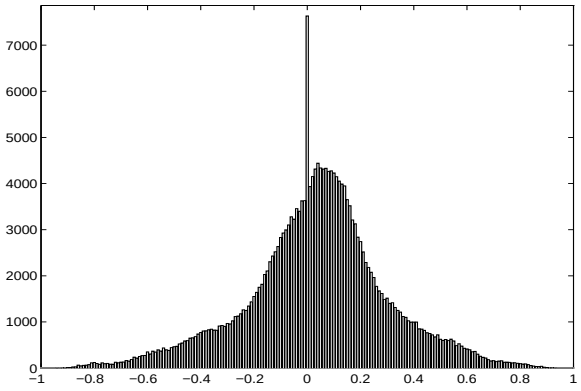


Figure 3: Distribution of distances (correlations) between response vectors for a given point and 220268 other unrelated points in a cluttered scene. A vast majority of the vectors lie near the mean distance $\mu = 0.037$ and are thus relatively uncorrelated with the response vector for the given point.

from the very favorable matching properties caused by the inherent tendency toward orthogonality in high-dimensional spaces. For example, consider the space $\{0, 1\}^n$ for large n ($n \geq 100$). If Hamming distance is used as the distance metric between points in this space, then the number of points that are within a distance of D bits from an arbitrary point follows a binomial distribution which, for large n, can be approximated by the normal distribution with mean $n/2$ and standard deviation $\sqrt{n}/2$. In other words,

$$N(D) \simeq \Phi\Big(\frac{D - n/2}{\sqrt{n}/2}\Big) \tag{9}$$

where $\Phi(z)$ denotes the standard normal distribution function with zero mean and unit deviation. Then,

$$Pr[|D - n/2| \geq t\sqrt{n}/2] \leq 2(1 - \Phi(t)) \tag{10}$$

The important observation is that most of the space is *orthogonal* (or "indifferent") to any given point. For example, with $n = 360$, the mean distance is 180 with a standard deviation of 9.5. Using $\Phi(4) = 0.99997$, we see that most of the space (99.994%) is approximately at the mean distance of 180 from a given point; less than 0.00006th of the vector space is closer to the point than 142 bits or further from it than 218 bits. Thus, *an object of interest can be represented by a high-dimensional vector that can be subjected to considerable noise before it is confused with other objects*. The same argument also applies to high-dimensional vectors whose components are non-binary such as the iconic feature vectors. Figure 3 shows the distribution of distances (computed as normalized dot-products or correlations) between the feature vector for a given model point and 220268 other unrelated points in a cluttered scene. The distribution of the distances has a mean $\mu = 0.037$ with a standard deviation $\sigma = 0.263$. It is clear most of the space is indifferent (correlation $\simeq 0.0$) to the given model point. Only 0.018% of the points had a correlation greater than 0.90, most of these points being located close to the model point.

4.1 Description of SDM

Simply put, SDM is a generalized random-access memory wherein the memory addresses and data words come

from high-dimensional vector spaces. As in a conventional random-access memory, there exists an array of storage locations, each identified by a number (the address of the location) with associated data being stored in these locations. However, due to the astronomical size of the vector space spanned by the address vectors, *only a sparse subset of the address space is used for identifying data locations and input addresses are not required to match stored addresses exactly but to only lie within a specified distance of an address to activate that address*.

The basic operation of SDM[1] as proposed by Kanerva can be summarized as follows :

- **Initialization**: The physical locations in SDM correspond to the rows of an $m \times k$ contents matrix **C** (initially filled with zeroes) in which data vectors $\in \{-1, 1\}^k$ are to be stored (see Figure 4). Pick m unique addresses (n-element binary vectors) at random for each of these locations.

- **Data Storage**: Given an n-element binary address vector **a** and a k-element data vector d for storage, select all storage locations whose addresses lie within a Hamming distance of D from **a**. *Add* the data vector d to the previous contents of each of the selected row vectors of **C**. Note that this is different from a conventional memory where addresses need to exactly match and previous contents are overwritten with new data.

- **Data Retrieval**: Given an n-element binary address vector **a**, select all storage locations whose addresses lie within a Hamming distance of D from **a**. Add the values of these selected locations in parallel (i.e. vector addition) to yield a sum vector **s** containing the k sums. Threshold these k sums at 0 to obtain the data vector $\mathbf{d}'$ i.e. $d_i = 1$ if $s_i > 0$ and $d_i = -1$ otherwise.

The statistically reconstructed data vector $\mathbf{d}'$ should be the same as the original data vector provided the *capacity* of the SDM [11] has not been exceeded. The intuitive reason for this is as follows: When storing a data vector **d** using an n-dimensional address vector **a**, each of the selected locations receives one copy of the data. During retrieval with an address close to **a**, say $\mathbf{a}'$, *most* of the locations that were selected with **a** are also selected with $\mathbf{a}'$. Thus, the sum vector contains most of the copies of **d**, plus copies of other different words; however, due to the orthogonality of the address space for large n, these extraneous copies are much fewer than the number of copies of **d**. This biases the sum vector in the direction of **d** and hence, **d** is output with high probability. A more rigorous argument can be found in [11].

4.2 Using SDM for Visual Recognition

The model of SDM used in our method differs from the one proposed by Kanerva in the following ways:

- The addresses are no longer binary but correspond to multivalued response vectors whose range is determined by the range of filter outputs.

[1]The SDM model can be realized as a three-layer feedforward neural network. In fact, the organization of SDM is strikingly similar to the organization of the human cerebellum. In particular, the cerebellar model proposed by the late David Marr [14] (and also the CMAC of James Albus) are closely related to generalized forms of the SDM as discussed in [11].

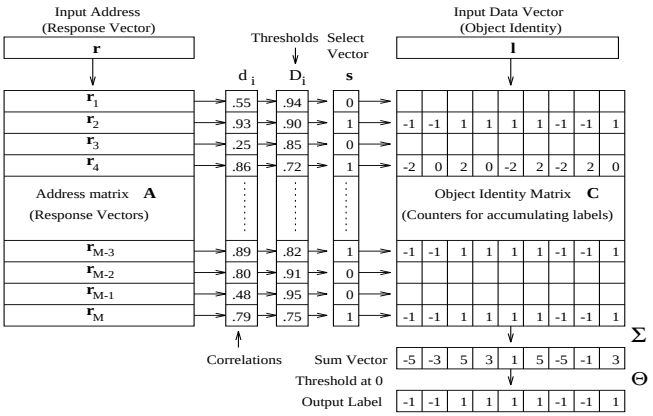


Figure 4: The modified Sparse Distributed Memory (SDM) model for learning associations between object appearance and object identity.

- The normalized dot product is used as the distance metric instead of the Hamming distance. In other words, the distance between response vectors $\mathbf{r^1}$ and $\mathbf{r^2}$ is computed as:

$$d(\mathbf{r^1}, \mathbf{r^2}) = \frac{\mathbf{r^1} \cdot \mathbf{r^2}}{||\mathbf{r^1}|| \ ||\mathbf{r^2}||} \tag{11}$$

- The set of response vectors will be clustered in many correlated groups distributed over a large portion of the response vector space. Therefore, if addresses are picked randomly, a large number of locations will never be activated while a number of locations will be selected so often that their contents will resemble noise. The way out of this dilemma is to *pick addresses according to the distribution of the data* [12]. In our case, we simply use an initial subset of the training response vectors. When all address locations have subsequently been filled, the address space can be allowed to *self-organize* using the well-known competitive Hebbian learning rule (or the Kohonen rule) as suggested by Keeler in [12].

Assume that the number of response vectors (each n-element long) currently stored is m. Let **A** represent the $m \times n$ matrix of magnitude-normalized (i.e. $\mathbf{r^i}/||\mathbf{r^i}||$) response vectors from the objects. Assume that we have stored response vectors for p objects. Each object is assigned an identity vector which can be viewed as the response of the system to the visual stimulus provided by the object; for instance, the identity vector could specify a name, a motor command, or even the response vector itself. For the current purposes, we associate the identity vectors with object labels, each object being defined by *a fixed range of values giving an indication of the pose of the object*. The identity vectors are assumed to belong to the set $\{-1, 1\}^k$, where k is chosen large enough to allow distinct labels for the various objects in the domain. Let **C** represent the $m \times k$ counter (or object identity) matrix whose rows will hold summations of object labels and whose entries fall within the set $\{-c, \ldots, (c-1)\}$ for some positive integer c. Figure 4 illustrates this organization.

4.2.1 Visual Learning of Object Identity

During the training phase, objects are presented to the active vision system which extracts the response vectors from the image region lying within the fovea. Each response vector **r** extracted from an object with a label **l** is stored in the SDM as follows. Let D_i denote the threshold for the ith address location and let T denote the nonlinear threshold function defined on m-element vectors whose ith component is given by :

$$T(\mathbf{x})_i = \begin{cases} 1 & \text{if } x_i \geq D_i \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

Note that T can in general be an arbitrary *radial basis function* [17]. The select vector

$$\mathbf{s} = T(\mathbf{A} \cdot \frac{\mathbf{r}}{||\mathbf{r}||}) \quad (13)$$

is then simply the vector containing ones in the locations i that have a correlation of at least D_i with **r**.[2] The object identity label **l** is then stored in the counter matrix **C** by simply adding it to the rows of **C** that were selected by **s** :

$$\mathbf{C} := \mathbf{C} + \mathbf{s} \square \mathbf{l} \quad (14)$$

where $\square$ represents the outer product operation. This in fact corresponds to a generalized *Hebbian learning rule* as noted in [12].

4.2.2 Retrieving Object Identity

Let **r** be a response vector obtained from one of the points in the current foveal region. Then the identity label $\mathbf{l}'$ corresponding to **r** is computed by summing all the vectors selected by **s** and thresholding the sum vector thus obtained at 0 :

$$\mathbf{l}' = \Theta(\mathbf{C}^T \mathbf{s}) \quad (15)$$

where $\Theta(\mathbf{x}) = \mathbf{u}$ where $u_i = 1$ if $x_i > 0$ and $u_i = -1$ otherwise. When more than one vector is used per object, the output label is obtained by thresholding the cumulative sum vector over the different object vectors. An alternative here is to use separate SDMs for the different foveal locations, thereby yielding a *topographic memory* [19].

5 Implementation

The algorithms described in the previous section have been implemented on an active vision system comprised of a binocular head with two color CCD television cameras that provide input to a $Datacube\ MaxVideo^{TM}$ MV200 pipeline image-processing system. The MV200 is a single integrated 6U VME circuit board with a wide range of frame-rate image analysis capabilities. Of particular interest to our work is its ability to perform convolutions at frame-rate ($30/sec$).

There are clearly three distinct phases in the algorithms of the previous section during either storage or retrieval : (a) Figure-ground segmentation, (b) Visual preprocessing to extract filter responses, and (c) Memory access.

[2] Note that **s** is a new representation in an m-dimensional space and corresponds to the *codon representation* of input in Marr's cerebellar model [14]. This transformation from an n-dimensional to an m-dimensional space ($m >> n$) adds further orthogonality to the matching process by amplifying any differences between input response vectors.

5.1 Figure-Ground Segmentation

The problem of figure-ground segmentation is much simpler than the general segmentation problem and can be solved in a number of different ways, most notably by the use of stereo. We have previously shown [3] that the use of an active binocular head allows stereo to be used for segmenting an occluder by using *zero disparity filtering* [6]. The zero disparity filter is a simple non-linear image filter that suppresses features that have non-zero disparity; in other words, it only passes image energy in the horopter. Such a filter is well-suited to perform a crude figure-ground segmentation of an object amidst a cluttered background.

5.2 Visual Preprocessing

Once the approximate boundary of the face is determined, the fovea can be directed to the centroid of the object. The MV200 executes nine convolutions with the different 8×8 Gaussian derivative kernels on a low-pass filtered five-level pyramid of the input image and filter responses are extracted for each of the sparse number of points in the foveal region. For the experiments, an object was represented by *response vectors from the centroid and each of the points lying on the intersections of radial lines with concentric circles of exponentially increasing radii centered on the centroid* as shown in Figure 6 (c). Note that this corresponds to an *implicit representation by parts*.

5.3 Memory Access

Our implementation optimizes the traditionally time-consuming step of memory access by implementing memory directly within the MV200 image processing system itself and *using convolutions for distance computations*. The modified SDM described in Section 4.2 can be implemented by using one (or more) of the memory banks of the MV200 for storing the matrix **A** as a "memory surface." During indexing, an input response vector is loaded into the 8×8 convolution kernel and convolved with the memory surface **A**; the closest vectors can be selected by simply thresholding the results of the convolution.

6 Experimental Results

We first describe the results of varying viewing conditions on the iconic feature vectors of arbitrary objects. These experiments give an indication of the robustness of the indexing algorithm by showing that the response vectors often change only slightly (correlation with the model vector remains above 0.8) when subjected to different variations in viewing condition. The SDM uses thresholds in the range 0.80-0.95 as motivated by Figure 3 where only 0.26% of the points have correlations greater than 0.8. Ambiguities left unresolved by single vectors are countered by using more than one feature vector per object as described in Section 5.2.

For the first experiment, we extracted the response vector from a region near the centroid of an initially unoccluded model object and plotted the distance (correlation) between the model response vector and those for the same point in scenarios with increasing degrees of occlusion as shown in Figure 5 (a). Despite the distortions caused by the occluders, the new iconic feature vectors remain correlated with the original vector.

To test insensitivity to modest changes in view, we examined the effect of gradual clockwise 5° changes in pose on the response vectors for a fixed point for a simple 3D

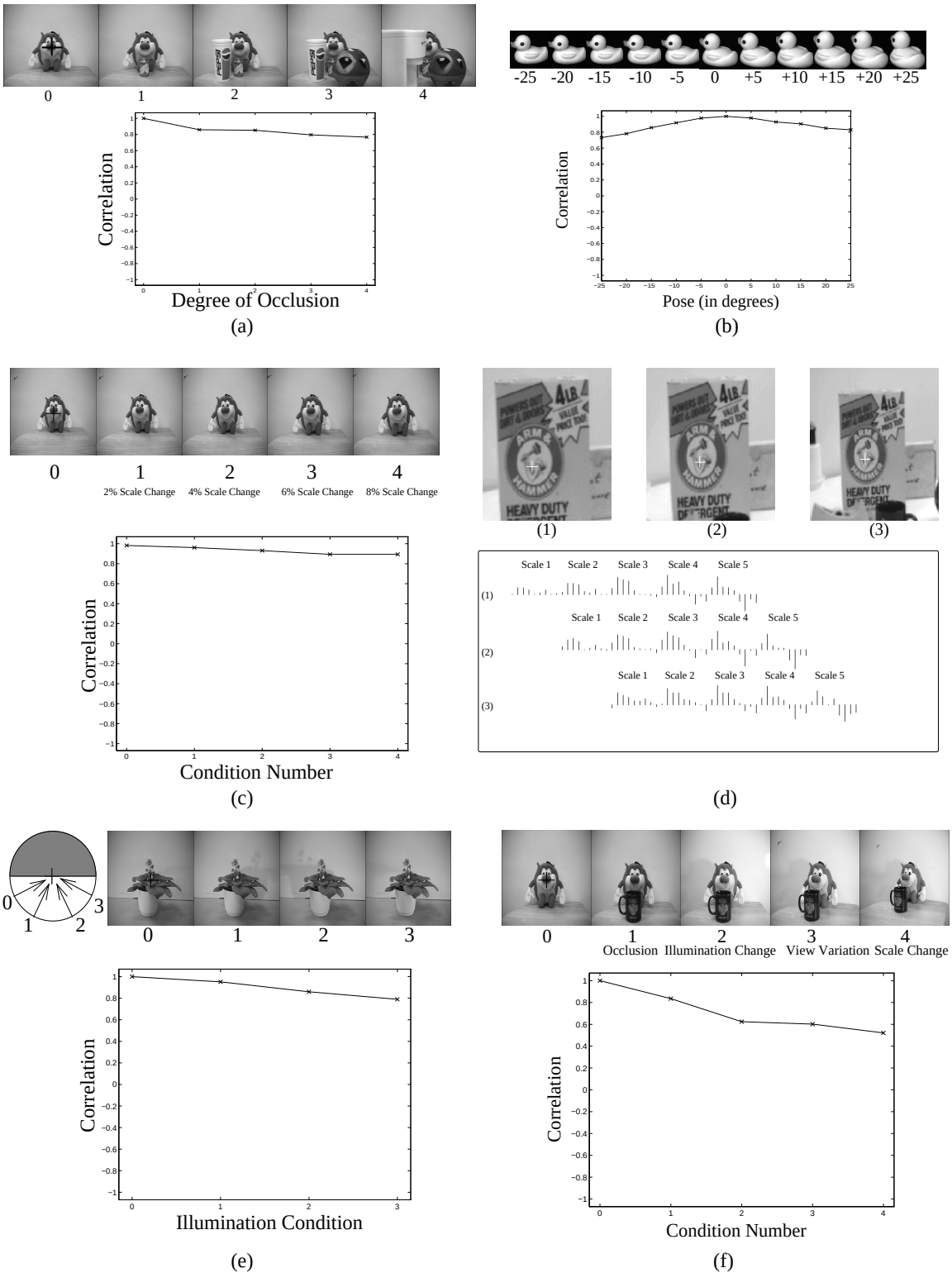


Figure 5: **Effects of Varying Viewing Conditions**. (a) Tolerance to partial occlusions; (b) Effect of Changes in 3D Pose; (c) Minor Scale Variations; (d) Handling larger scale variations by scale interpolation; (e) Changes in Illumination; (f) Combined effects of variations caused by multiple sources.

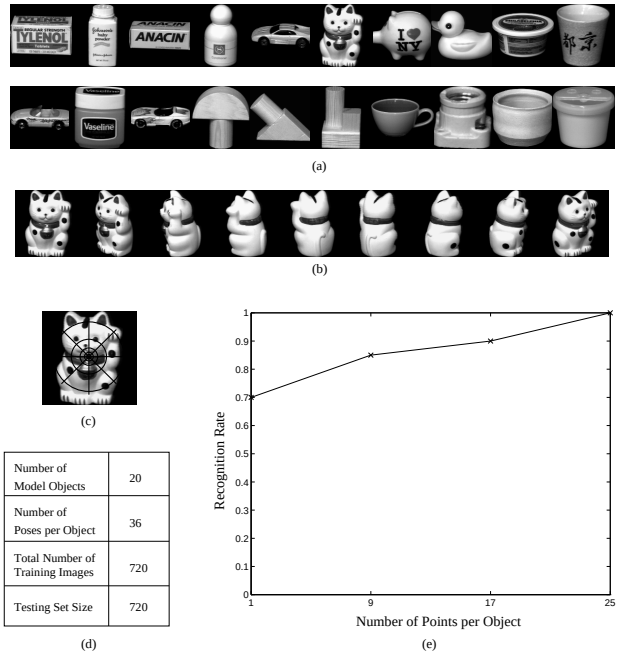
object. As shown in Figure 5 (b), the correlation remains above 0.8 for pose changes of upto $40°$.

The iconic object representations are tolerant to minor scale variations ($< 10\%$). This fact is illustrated in Figure 5 (c) which depicts the experimental results obtained by increasing scale in steps of 2%. Larger changes in scale are handled by a scale interpolation strategy which accounts for scale changes by interpolating with responses *across* scales as illustrated in Figure 5 (d) (see [18] for further details).

In the experiment shown in Figure 5 (e), we exposed a model object separately to illumination from a $60W$ bulb at a radial distance of 2 feet from four different directions (labeled 0, 1, 2, and 3). There is a noticeable decrease in correlation between the model vector and the new vectors, though it remains relatively high (> 0.8). Larger changes in illumination can be countered by using brightness normalization techniques in addition to possible active control of camera aperture.

The experiment in Figure 5 (f) shows the graceful degradation caused by incrementally adding (1) an occlusion, (2) an illumination change followed by (3) a view variation and finally, (4) a reduction in scale (brightness normalization and scale interpolation strategies were *not* used for this experiment). Despite the large distortions caused by these transformations, the new iconic feature vectors all have a correlation of 0.5 or more, which is still far from the indifference distance of 0.0 where the vast majority of the other vectors lie (Figure 3).

Finally, the 3D recognition performance of the indexing technique was tested on the Columbia object database that was originally used in [15] by Murase and Nayar. Figure 6 (a) shows the segmented images of 20 3D objects in the database for a given pose. During the training phase, 36 images of each object at $10°$ increments in pose were used to extract response vectors for storage in the SDM. For testing the indexing scheme, we randomly selected images of objects corresponding to poses that lie exactly in between the training poses. As indicated by Figure 6 (e), even when only one point was used per object, 70% of the test cases were still successfully recognized. Addition of more points within the fovea per object increased the recognition rate until 100% accuracy was achieved when 25 foveal points were used for indexing into the SDM.



Number of Model Objects	20
Number of Poses per Object	36
Total Number of Training Images	720
Testing Set Size	720

(d)

Figure 6: **Recognition Results**. (a) The 20 objects used in the experiment. (b) 9 of the 36 images of an object extracted at $10°$ rotational increments in pose to represent the entire pose space. (c) For storage in the SDM, response vectors from the points at the intersections of radial lines with concentric circles centered on the approximate object centroid were used. (d) summarizes the experimental parameters. (e) Recognition rate (fraction of test images correctly recognized) plotted as a function of number of vectors used per object in a given pose.

7 Discussion and Conclusions

This paper presents a new approach to the object indexing problem: using multiscale iconic feature vectors as components of a sparse distributed memory. This combination has a number of salient features which can be summarized as follows:

- **PCA-based Generalized Basis Functions**: The derivative-of-Gaussian basis functions used in our approach arise as a result of unsupervised learning in a neural network performing PCA on arbitrary images of natural scenes; they are thus well-suited for indexing in a wide variety of domains. The high-dimensionality of the response vectors derived from the basis functions further improves recognition accuracy.

- **Rotation and Scale Invariance**: The steerability of Gaussian derivative filters allows an efficient normalization procedure for rotations about the viewing axis. The incorporation of filter responses at different scales allows the use of simple interpolation strategies for achieving invariance in the presence of drastic changes in scale [18].

- **Tolerance to Changes in Viewing Conditions**: Minor occlusions[3] or modest perspective changes and interference caused by varying background lying in the receptive fields of the largest scale filters are tolerated because a large number of measurements are used per point; distortions in a few components act as noise to which the high-dimensional representation remains robust. Illumination changes are handled in two ways. First, none of the filters used have a DC response. Second, the use of normalized dot product as a distance metric additionally makes the matching process robust to global contrast changes.

- **Sparse Distributed Memory**: An associative model of visual memory based on Kanerva's sparse distributed memory is used for storage and retrieval of object identity. This form of memory facilitates visual learning and allows interpolation between views besides offering the additional advantages of constant indexing time ($O(M) = O(1)$ where M is the number of address/storage locations) and the possibility of greater storage capacity over sequential memory due to

[3] A more sophisticated strategy for handling partial occlusions is described in [3].

the multiplexing inherent in the SDM combined with the use of more than one response vector per object.

- **Real-Time Recognition**: Iconic techniques such as the one proposed in this paper have been greeted with considerable skepticism in the past since they have been computation-intensive. However, the recent availability of pipeline image processors significantly ameliorates this drawback. In particular, the frame-rate convolution capability of these processors can be effectively exploited to make iconic techniques practical and efficient as demonstrated in this paper.

A possible cause for concern is the use of upto 25 vectors per object. A little reflection however reveals that this choice still results in considerable savings over the alternative of pixelwise storage of images (25×45 versus 128×128). Our view-based approach raises the question of scalability: will the method fail when extremely large model bases of objects are used with arbitrary $3D$ pose? It is however not hard to see that the use of more than one vector per object potentially allows an extremely large number of objects to be handled. Kanerva [11] estimates the capacity of the SDM to be about 5% of the number of storage locations; thus, with only 1000 storage locations, the number of potentially distinguishable objects is still $\binom{50}{25}$ which is an extremely large number, even after factoring out the number of different views for an object. The accuracy of the above naive estimate clearly depends on the extent to which response vectors are shared between different objects; while we have found noticeable overlap in general, we believe that the possible use of self-organization within the address space will significantly help in extending the capacity of the memory by allowing the stored response vectors to essentially act as higher-level basis functions for describing objects.

Ongoing work includes motion-based segmentation, saliency-based selection of object points, and augmentation of the feature vector with responses from a variety of color-opponent Gaussian center-surround filters derived from unsupervised learning along the RGB planes.

References

[1] J. Aloimonos, A. Bandopadhay, and I. Weiss. Active vision. *IJCV*, 1 (4):333–356, 1988.

[2] Ruzena Bajcsy. Active perception. In *Proceedings of the IEEE*, volume 76, pages 996–1005, August 1988.

[3] Dana H. Ballard and Rajesh P.N. Rao. Seeing behind occlusions. In *Proc. of ECCV*, pages 274–285, 1994.

[4] D.H. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[5] Roland T. Chin and Charles R. Dyer. Model-based recognition in robot vision. *ACM Computing Surveys*, 18(1):67–108, March 1986.

[6] David J. Coombs. *Real-Time Gaze Holding in Binocular Robot Vision*. PhD thesis, University of Rochester Computer Science Dept., 1992. Available as Technical Report 415.

[7] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(9):891–906, September 1991.

[8] D. Gabor. Theory of communication. *J IEE*, 93:429–459, 1946.

[9] Peter J.B. Hancock, Roland J. Baddeley, and Leslie S. Smith. The principal components of natural images. *Network*, 3:61–70, 1992.

[10] Pentti Kanerva. *Sparse Distributed Memory*. Bradford Books, Cambridge, MA, 1988.

[11] Pentti Kanerva. Sparse distributed memory and related models. In Mohamad H. Hassoun, editor, *Associative Neural Memories*, pages 50–76. New York : Oxford University Press, 1993.

[12] James D. Keeler. Comparison between Kanerva's SDM and Hopfield-type neural networks. *Cognitive Science*, 12:299–329, 1988.

[13] V.K. Kumar, D. Casasent, and H. Murakami. Principal-component imagery for statistical pattern recognition correlators. *Optical Engineering*, 21(1):43–47, 1982.

[14] David Marr. A theory of cerebellar cortex. *J. Physiol. (London)*, 202:437–470, 1969.

[15] Hiroshi Murase and Shree K. Nayar. Visual learning and recognition of 3D objects from appearance. *IJCV*, 14:5–24, 1995.

[16] Erkki Oja. A simplified neuron model as a principal component analyzer. *J. Math. Biology*, 15:267–273, 1982.

[17] T. Poggio and F. Girosi. Networks for approximation and learning. *Proc. IEEE*, 78:1481–1497, 1990.

[18] Rajesh P.N. Rao and Dana H. Ballard. An active vision architecture based on iconic representations. Technical Report 548, Department of Computer Science, University of Rochester, 1995.

[19] Rajesh P.N. Rao and Dana H. Ballard. Natural basis functions and topographic memory for face recognition. In *Proc. of IJCAI*, 1995. (To appear).

[20] L.G. Roberts. Machine perception of three-dimensional solids. In James T. Tippett et al., editor, *Optical and Electro-Optical Information Processing*. Cambridge: MIT Press, 1965.

[21] Terence David Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2:459–473, 1989.

[22] David G. Stork and Hugh R. Wilson. Do Gabor functions provide appropriate descriptions of visual cortical receptive fields? *J. Optical Society of America A*, 7(8):1362–1373, 1990.

[23] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[24] R.A. Young. The Gaussian derivative theory of spatial vision: Analysis of cortical cell receptive field line-weighting profiles. *General Motors Research Publication GMR-4920*, 1985.