Object Recognition from Local Scale-Invariant Features

David G. Lowe
Computer Science Department
University of British Columbia
Vancouver, B.C., V6T 1Z4, Canada
lowe@cs.ubc.ca

Abstract

An object recognition system has been developed that uses a new class of local image features. The features are invariant to image scaling, translation, and rotation, and partially invariant to illumination changes and affine or 3D projection. These features share similar properties with neurons in inferior temporal cortex that are used for object recognition in primate vision. Features are efficiently detected through a staged filtering approach that identifies stable points in scale space. Image keys are created that allow for local geometric deformations by representing blurred image gradients in multiple orientation planes and at multiple scales. The keys are used as input to a nearest-neighbor indexing method that identifies candidate object matches. Final verification of each match is achieved by finding a low-residual least-squares solution for the unknown model parameters. Experimental results show that robust object recognition can be achieved in cluttered partially-occluded images with a computation time of under 2 seconds.

1. Introduction

Object recognition in cluttered real-world scenes requires local image features that are unaffected by nearby clutter or partial occlusion. The features must be at least partially invariant to illumination, 3D projective transforms, and common object variations. On the other hand, the features must also be sufficiently distinctive to identify specific objects among many alternatives. The difficulty of the object recognition problem is due in large part to the lack of success in finding such image features. However, recent research on the use of dense local features (e.g., Schmid & Mohr [19]) has shown that efficient recognition can often be achieved by using local image descriptors sampled at a large number of repeatable locations.

This paper presents a new method for image feature generation called the Scale Invariant Feature Transform (SIFT). This approach transforms an image into a large collection of local feature vectors, each of which is invariant to image translation, scaling, and rotation, and partially invariant to illumination changes and affine or 3D projection. Previous approaches to local feature generation lacked invariance to scale and were more sensitive to projective distortion and illumination change. The SIFT features share a number of properties in common with the responses of neurons in inferior temporal (IT) cortex in primate vision. This paper also describes improved approaches to indexing and model verification.

The scale-invariant features are efficiently identified by using a staged filtering approach. The first stage identifies key locations in scale space by looking for locations that are maxima or minima of a difference-of-Gaussian function. Each point is used to generate a feature vector that describes the local image region sampled relative to its scale-space coordinate frame. The features achieve partial invariance to local variations, such as affine or 3D projections, by blurring image gradient locations. This approach is based on a model of the behavior of complex cells in the cerebral cortex of mammalian vision. The resulting feature vectors are called SIFT keys. In the current implementation, each image generates on the order of 1000 SIFT keys, a process that requires less than 1 second of computation time.

The SIFT keys derived from an image are used in a nearest-neighbour approach to indexing to identify candidate object models. Collections of keys that agree on a potential model pose are first identified through a Hough transform hash table, and then through a least-squares fit to a final estimate of model parameters. When at least 3 keys agree on the model parameters with low residual, there is strong evidence for the presence of the object. Since there may be dozens of SIFT keys in the image of a typical object, it is possible to have substantial levels of occlusion in the image and yet retain high levels of reliability.

The current object models are represented as 2D locations of SIFT keys that can undergo affine projection. Sufficient variation in feature location is allowed to recognize perspective projection of planar shapes at up to a 60 degree rotation away from the camera or to allow up to a 20 degree rotation of a 3D object.

Proc. of the International Conference on Computer Vision, Corfu (Sept. 1999)

2. Related research

Object recognition is widely used in the machine vision industry for the purposes of inspection, registration, and manipulation. However, current commercial systems for object recognition depend almost exclusively on correlation-based template matching. While very effective for certain engineered environments, where object pose and illumination are tightly controlled, template matching becomes computationally infeasible when object rotation, scale, illumination, and 3D pose are allowed to vary, and even more so when dealing with partial visibility and large model databases.

An alternative to searching all image locations for matches is to extract features from the image that are at least partially invariant to the image formation process and matching only to those features. Many candidate feature types have been proposed and explored, including line segments [6], groupings of edges [11, 14], and regions [2], among many other proposals. While these features have worked well for certain object classes, they are often not detected frequently enough or with sufficient stability to form a basis for reliable recognition.

There has been recent work on developing much denser collections of image features. One approach has been to use a corner detector (more accurately, a detector of peaks in local image variation) to identify repeatable image locations, around which local image properties can be measured. Zhang *et al.* [23] used the Harris corner detector to identify feature locations for epipolar alignment of images taken from differing viewpoints. Rather than attempting to correlate regions from one image against all possible regions in a second image, large savings in computation time were achieved by only matching regions centered at corner points in each image.

For the object recognition problem, Schmid & Mohr [19] also used the Harris corner detector to identify interest points, and then created a local image descriptor at each interest point from an orientation-invariant vector of derivative-of-Gaussian image measurements. These image descriptors were used for robust object recognition by looking for multiple matching descriptors that satisfied object-based orientation and location constraints. This work was impressive both for the speed of recognition in a large database and the ability to handle cluttered images.

The corner detectors used in these previous approaches have a major failing, which is that they examine an image at only a single scale. As the change in scale becomes significant, these detectors respond to different image points. Also, since the detector does not provide an indication of the object scale, it is necessary to create image descriptors and attempt matching at a large number of scales. This paper describes an efficient method to identify stable key locations in scale space. This means that different scalings of an image will have no effect on the set of key locations selected. Furthermore, an explicit scale is determined for each point, which allows the image description vector for that point to be sampled at an equivalent scale in each image. A canonical orientation is determined at each location, so that matching can be performed relative to a consistent local 2D coordinate frame. This allows for the use of more distinctive image descriptors than the rotation-invariant ones used by Schmid and Mohr, and the descriptor is further modified to improve its stability to changes in affine projection and illumination.

Other approaches to appearance-based recognition include eigenspace matching [13], color histograms [20], and receptive field histograms [18]. These approaches have all been demonstrated successfully on isolated objects or pre-segmented images, but due to their more global features it has been difficult to extend them to cluttered and partially occluded images. Ohba & Ikeuchi [15] successfully apply the eigenspace approach to cluttered images by using many small local eigen-windows, but this then requires expensive search for each window in a new image, as with template matching.

3. Key localization

We wish to identify locations in image scale space that are invariant with respect to image translation, scaling, and rotation, and are minimally affected by noise and small distortions. Lindeberg [8] has shown that under some rather general assumptions on scale invariance, the Gaussian kernel and its derivatives are the only possible smoothing kernels for scale space analysis.

To achieve rotation invariance and a high level of efficiency, we have chosen to select key locations at maxima and minima of a difference of Gaussian function applied in scale space. This can be computed very efficiently by building an image pyramid with resampling between each level. Furthermore, it locates key points at regions and scales of high variation, making these locations particularly stable for characterizing the image. Crowley & Parker [4] and Lindeberg [9] have previously used the difference-of-Gaussian in scale space for other purposes. In the following, we describe a particularly efficient and stable method to detect and characterize the maxima and minima of this function.

As the 2D Gaussian function is separable, its convolution with the input image can be efficiently computed by applying two passes of the 1D Gaussian function in the horizontal and vertical directions:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$$

For key localization, all smoothing operations are done using $\sigma = \sqrt{2}$, which can be approximated with sufficient accuracy using a 1D kernel with 7 sample points.

The input image is first convolved with the Gaussian function using $\sigma = \sqrt{2}$ to give an image A. This is then repeated a second time with a further incremental smoothing of $\sigma = \sqrt{2}$ to give a new image, B, which now has an effective smoothing of $\sigma = 2$. The difference of Gaussian function is obtained by subtracting image B from A, resulting in a ratio of $2/\sqrt{2} = \sqrt{2}$ between the two Gaussians.

To generate the next pyramid level, we resample the already smoothed image B using bilinear interpolation with a pixel spacing of 1.5 in each direction. While it may seem more natural to resample with a relative scale of $\sqrt{2}$, the only constraint is that sampling be frequent enough to detect peaks. The 1.5 spacing means that each new sample will be a constant linear combination of 4 adjacent pixels. This is efficient to compute and minimizes aliasing artifacts that would arise from changing the resampling coefficients.

Maxima and minima of this scale-space function are determined by comparing each pixel in the pyramid to its neighbours. First, a pixel is compared to its 8 neighbours at the same level of the pyramid. If it is a maxima or minima at this level, then the closest pixel location is calculated at the next lowest level of the pyramid, taking account of the 1.5 times resampling. If the pixel remains higher (or lower) than this closest pixel and its 8 neighbours, then the test is repeated for the level above. Since most pixels will be eliminated within a few comparisons, the cost of this detection is small and much lower than that of building the pyramid.

If the first level of the pyramid is sampled at the same rate as the input image, the highest spatial frequencies will be ignored. This is due to the initial smoothing, which is needed to provide separation of peaks for robust detection. Therefore, we expand the input image by a factor of 2, using bilinear interpolation, prior to building the pyramid. This gives on the order of 1000 key points for a typical 512×512 pixel image, compared to only a quarter as many without the initial expansion.

3.1. SIFT key stability

To characterize the image at each key location, the smoothed image A at each level of the pyramid is processed to extract image gradients and orientations. At each pixel, A_{ij}, the image gradient magnitude, M_{ij}, and orientation, R_{ij}, are computed using pixel differences:

$$M_{ij} = \sqrt{(A_{ij} - A_{i+1,j})^2 + (A_{ij} - A_{i,j+1})^2}$$

$$R_{ij} = \operatorname{atan2}\left(A_{ij} - A_{i+1,j}, A_{i,j+1} - A_{ij}\right)$$

The pixel differences are efficient to compute and provide sufficient accuracy due to the substantial level of previous smoothing. The effective half-pixel shift in position is compensated for when determining key location.

Robustness to illumination change is enhanced by thresholding the gradient magnitudes at a value of 0.1 times the maximum possible gradient value. This reduces the effect of a change in illumination direction for a surface with 3D relief, as an illumination change may result in large changes to gradient magnitude but is likely to have less influence on gradient orientation.

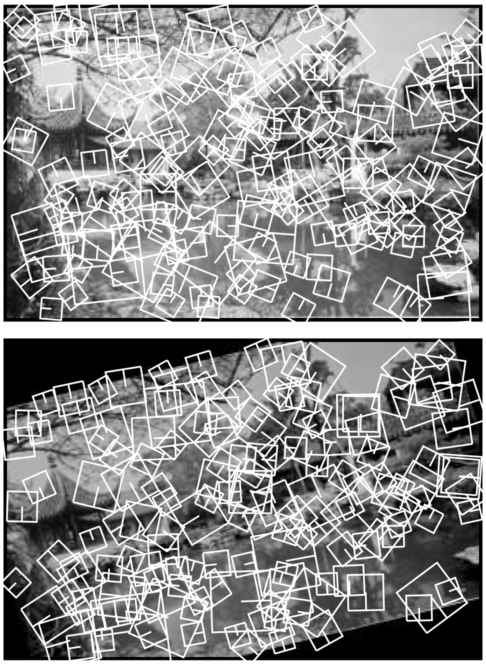

Figure 1: The second image was generated from the first by rotation, scaling, stretching, change of brightness and contrast, and addition of pixel noise. In spite of these changes, 78% of the keys from the first image have a closely matching key in the second image. These examples show only a subset of the keys to reduce clutter.

Each key location is assigned a canonical orientation so that the image descriptors are invariant to rotation. In order to make this as stable as possible against lighting or contrast changes, the orientation is determined by the peak in a histogram of local image gradient orientations. The orientation histogram is created using a Gaussian-weighted window with σ of 3 times that of the current smoothing scale. These weights are multiplied by the thresholded gradient values and accumulated in the histogram at locations corresponding to the orientation, R_{ij}. The histogram has 36 bins covering the 360 degree range of rotations, and is smoothed prior to peak selection.

The stability of the resulting keys can be tested by subjecting natural images to affine projection, contrast and brightness changes, and addition of noise. The location of each key detected in the first image can be predicted in the transformed image from knowledge of the transform parameters. This framework was used to select the various sampling and smoothing parameters given above, so that max-

Image transformation	Match %	Ori %
A. Increase contrast by 1.2	89.0	86.6
B. Decrease intensity by 0.2	88.5	85.9
C. Rotate by 20 degrees	85.4	81.0
D. Scale by 0.7	85.1	80.3
E. Stretch by 1.2	83.5	76.1
F. Stretch by 1.5	77.7	65.0
G. Add 10% pixel noise	90.3	88.4
H. All of A,B,C,D,E,G.	78.6	71.8

Figure 2: For various image transformations applied to a sample of 20 images, this table gives the percent of keys that are found at matching locations and scales (Match %) and that also match in orientation (Ori %).

imum efficiency could be obtained while retaining stability to changes.

Figure 1 shows a relatively small number of keys detected over a 2 octave range of only the larger scales (to avoid excessive clutter). Each key is shown as a square, with a line from the center to one side of the square indicating orientation. In the second half of this figure, the image is rotated by 15 degrees, scaled by a factor of 0.9, and stretched by a factor of 1.1 in the horizontal direction. The pixel intensities, in the range of 0 to 1, have 0.1 subtracted from their brightness values and the contrast reduced by multiplication by 0.9. Random pixel noise is then added to give less than 5 bits/pixel of signal. In spite of these transformations, 78% of the keys in the first image had closely matching keys in the second image at the predicted locations, scales, and orientations

The overall stability of the keys to image transformations can be judged from Table 2. Each entry in this table is generated from combining the results of 20 diverse test images and summarizes the matching of about 15,000 keys. Each line of the table shows a particular image transformation. The first figure gives the percent of keys that have a matching key in the transformed image within σ in location (relative to scale for that key) and a factor of 1.5 in scale. The second column gives the percent that match these criteria as well as having an orientation within 20 degrees of the prediction.

4. Local image description

Given a stable location, scale, and orientation for each key, it is now possible to describe the local image region in a manner invariant to these transformations. In addition, it is desirable to make this representation robust against small shifts in local geometry, such as arise from affine or 3D projection. One approach to this is suggested by the response properties of complex neurons in the visual cortex, in which a feature position is allowed to vary over a small region while orientation and spatial frequency specificity are maintained. Edelman, Intrator & Poggio [5] have performed experiments that simulated the responses of complex neurons to different 3D views of computer graphic models, and found that the complex cell outputs provided much better discrimination than simple correlation-based matching. This can be seen, for example, if an affine projection stretches an image in one direction relative to another, which changes the relative locations of gradient features while having a smaller effect on their orientations and spatial frequencies.

This robustness to local geometric distortion can be obtained by representing the local image region with multiple images representing each of a number of orientations (referred to as orientation planes). Each orientation plane contains only the gradients corresponding to that orientation, with linear interpolation used for intermediate orientations. Each orientation plane is blurred and resampled to allow for larger shifts in positions of the gradients.

This approach can be efficiently implemented by using the same precomputed gradients and orientations for each level of the pyramid that were used for orientation selection. For each keypoint, we use the pixel sampling from the pyramid level at which the key was detected. The pixels that fall in a circle of radius 8 pixels around the key location are inserted into the orientation planes. The orientation is measured relative to that of the key by subtracting the key's orientation. For our experiments we used 8 orientation planes, each sampled over a 4×4 grid of locations, with a sample spacing 4 times that of the pixel spacing used for gradient detection. The blurring is achieved by allocating the gradient of each pixel among its 8 closest neighbors in the sample grid, using linear interpolation in orientation and the two spatial dimensions. This implementation is much more efficient than performing explicit blurring and resampling, yet gives almost equivalent results.

In order to sample the image at a larger scale, the same process is repeated for a second level of the pyramid one octave higher. However, this time a 2×2 rather than a 4×4 sample region is used. This means that approximately the same image region will be examined at both scales, so that any nearby occlusions will not affect one scale more than the other. Therefore, the total number of samples in the SIFT key vector, from both scales, is $8 \times 4 \times 4 + 8 \times 2 \times 2$ or 160 elements, giving enough measurements for high specificity.

5. Indexing and matching

For indexing, we need to store the SIFT keys for sample images and then identify matching keys from new images. The problem of identifying the most similar keys for high dimensional vectors is known to have high complexity if an exact solution is required. However, a modification of the k-d tree algorithm called the best-bin-first search method (Beis & Lowe [3]) can identify the nearest neighbors with high probability using only a limited amount of computation. To further improve the efficiency of the best-bin-first algorithm, the SIFT key samples generated at the larger scale are given twice the weight of those at the smaller scale. This means that the larger scale is in effect able to filter the most likely neighbours for checking at the smaller scale. This also improves recognition performance by giving more weight to the least-noisy scale. In our experiments, it is possible to have a cut-off for examining at most 200 neighbors in a probabilistic best-bin-first search of 30,000 key vectors with almost no loss of performance compared to finding an exact solution.

An efficient way to cluster reliable model hypotheses is to use the Hough transform [1] to search for keys that agree upon a particular model pose. Each model key in the database contains a record of the key's parameters relative to the model coordinate system. Therefore, we can create an entry in a hash table predicting the model location, orientation, and scale from the match hypothesis. We use a bin size of 30 degrees for orientation, a factor of 2 for scale, and 0.25 times the maximum model dimension for location. These rather broad bin sizes allow for clustering even in the presence of substantial geometric distortion, such as due to a change in 3D viewpoint. To avoid the problem of boundary effects in hashing, each hypothesis is hashed into the 2 closest bins in each dimension, giving a total of 16 hash table entries for each hypothesis.

Figure 3: Model images of planar objects are shown in the top row. Recognition results below show model outlines and image keys used for matching.

6. Solution for affine parameters

The hash table is searched to identify all clusters of at least 3 entries in a bin, and the bins are sorted into decreasing order of size. Each such cluster is then subject to a verification procedure in which a least-squares solution is performed for the affine projection parameters relating the model to the image.

The affine transformation of a model point $[x\ y]^T$ to an image point $[u\ v]^T$ can be written as

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

where the model translation is $[t_x\ t_y]^T$ and the affine rotation, scale, and stretch are represented by the m_i parameters.

We wish to solve for the transformation parameters, so the equation above can be rewritten as

$$\begin{bmatrix} x & y & 0 & 0 & 1 & 0 \\ 0 & 0 & x & y & 0 & 1 \\ & & \cdots & & & \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u \\ v \\ \vdots \end{bmatrix}$$

This equation shows a single match, but any number of further matches can be added, with each match contributing two more rows to the first and last matrix. At least 3 matches are needed to provide a solution.

We can write this linear system as

$$\mathbf{Ax} = \mathbf{b}$$

The least-squares solution for the parameters **x** can be deter-

Figure 4: Top row shows model images for 3D objects with outlines found by background segmentation. Bottom image shows recognition results for 3D objects with model outlines and image keys used for matching.

mined by solving the corresponding normal equations,

$$\mathbf{x} = [\mathbf{A}^{\mathbf{T}}\mathbf{A}]^{-1}\mathbf{A}^{\mathbf{T}}\mathbf{b}$$

which minimizes the sum of the squares of the distances from the projected model locations to the corresponding image locations. This least-squares approach could readily be extended to solving for 3D pose and internal parameters of articulated and flexible objects [12].

Outliers can now be removed by checking for agreement between each image feature and the model, given the parameter solution. Each match must agree within 15 degrees orientation, $\sqrt{2}$ change in scale, and 0.2 times maximum model size in terms of location. If fewer than 3 points remain after discarding outliers, then the match is rejected. If any outliers are discarded, the least-squares solution is re-solved with the remaining points.

Figure 5: Examples of 3D object recognition with occlusion.

7. Experiments

The affine solution provides a good approximation to perspective projection of planar objects, so planar models provide a good initial test of the approach. The top row of Figure 3 shows three model images of rectangular planar faces of objects. The figure also shows a cluttered image containing the planar objects, and the same image is shown overlayed with the models following recognition. The model keys that are displayed are the ones used for recognition and final least-squares solution. Since only 3 keys are needed for robust recognition, it can be seen that the solutions are highly redundant and would survive substantial occlusion. Also shown are the rectangular borders of the model images, projected using the affine transform from the least-square solution. These closely agree with the true borders of the planar regions in the image, except for small errors introduced by the perspective projection. Similar experiments have been performed for many images of planar objects, and the recognition has proven to be robust to at least a 60 degree rotation of the object in any direction away from the camera.

Although the model images and affine parameters do not account for rotation in depth of 3D objects, they are still sufficient to perform robust recognition of 3D objects over about a 20 degree range of rotation in depth away from each model view. An example of three model images is shown in

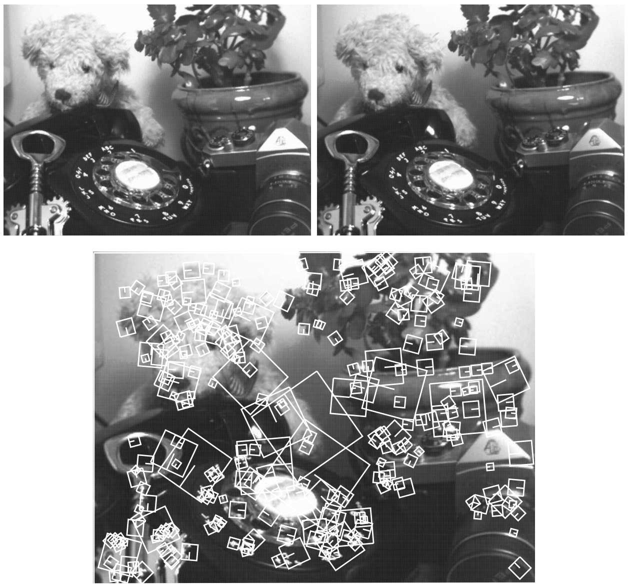

Figure 6: Stability of image keys is tested under differing illumination. The first image is illuminated from upper left and the second from center right. Keys shown in the bottom image were those used to match second image to first.

the top row of Figure 4. The models were photographed on a black background, and object outlines extracted by segmenting out the background region. An example of recognition is shown in the same figure, again showing the SIFT keys used for recognition. The object outlines are projected using the affine parameter solution, but this time the agreement is not as close because the solution does not account for rotation in depth. Figure 5 shows more examples in which there is significant partial occlusion.

The images in these examples are of size 384×512 pixels. The computation times for recognition of all objects in each image are about 1.5 seconds on a Sun Sparc 10 processor, with about 0.9 seconds required to build the scale-space pyramid and identify the SIFT keys, and about 0.6 seconds to perform indexing and least-squares verification. This does not include time to pre-process each model image, which would be about 1 second per image, but would only need to be done once for initial entry into a model database.

The illumination invariance of the SIFT keys is demonstrated in Figure 6. The two images are of the same scene from the same viewpoint, except that the first image is illuminated from the upper left and the second from the center right. The full recognition system is run to identify the second image using the first image as the model, and the second image is correctly recognized as matching the first. Only SIFT keys that were part of the recognition are shown. There were 273 keys that were verified as part of the final match, which means that in each case not only was the same key detected at the same location, but it also was the closest match to the correct corresponding key in the second image. Any 3 of these keys would be sufficient for recognition. While matching keys are not found in some regions where highlights or shadows change (for example on the shiny top of the camera) in general the keys show good invariance to illumination change.

8. Connections to biological vision

The performance of human vision is obviously far superior to that of current computer vision systems, so there is potentially much to be gained by emulating biological processes. Fortunately, there have been dramatic improvements within the past few years in understanding how object recognition is accomplished in animals and humans.

Recent research in neuroscience has shown that object recognition in primates makes use of features of intermediate complexity that are largely invariant to changes in scale, location, and illumination (Tanaka [21], Perrett & Oram [16]). Some examples of such intermediate features found in inferior temporal cortex (IT) are neurons that respond to a dark five-sided star shape, a circle with a thin protruding element, or a horizontal textured region within a triangular boundary. These neurons maintain highly specific responses to shape features that appear anywhere within a large portion of the visual field and over a several octave range of scales (Ito *et. al* [7]). The complexity of many of these features appears to be roughly the same as for the current SIFT features, although there are also some neurons that respond to more complex shapes, such as faces. Many of the neurons respond to color and texture properties in addition to shape. The feature responses have been shown to depend on previous visual learning from exposure to specific objects containing the features (Logothetis, Pauls & Poggio [10]). These features appear to be derived in the brain by a highly computation-intensive parallel process, which is quite different from the staged filtering approach given in this paper. However, the results are much the same: an image is transformed into a large set of local features that each match a small fraction of potential objects yet are largely invariant to common viewing transformations.

It is also known that object recognition in the brain depends on a serial process of attention to bind features to object interpretations, determine pose, and segment an object from a cluttered background [22]. This process is presumably playing the same role in verification as the parameter solving and outlier detection used in this paper, since the accuracy of interpretations can often depend on enforcing a single viewpoint constraint [11].

9. Conclusions and comments

The SIFT features improve on previous approaches by being largely invariant to changes in scale, illumination, and local affine distortions. The large number of features in a typical image allow for robust recognition under partial occlusion in cluttered images. A final stage that solves for affine model parameters allows for more accurate verification and pose determination than in approaches that rely only on indexing.

An important area for further research is to build models from multiple views that represent the 3D structure of objects. This would have the further advantage that keys from multiple viewing conditions could be combined into a single model, thereby increasing the probability of finding matches in new views. The models could be true 3D representations based on structure-from-motion solutions, or could represent the space of appearance in terms of automated clustering and interpolation (Pope & Lowe [17]). An advantage of the latter approach is that it could also model non-rigid deformations.

The recognition performance could be further improved by adding new SIFT feature types to incorporate color, texture, and edge groupings, as well as varying feature sizes and offsets. Scale-invariant edge groupings that make local figure-ground discriminations would be particularly useful at object boundaries where background clutter can interfere with other features. The indexing and verification framework allows for all types of scale and rotation invariant features to be incorporated into a single model representation. Maximum robustness would be achieved by detecting many different feature types and relying on the indexing and clustering to select those that are most useful in a particular image.

References

[1] Ballard, D.H., "Generalizing the Hough transform to detect arbitrary patterns," *Pattern Recognition*, **13,** 2 (1981), pp. 111-122.

[2] Basri, Ronen, and David. W. Jacobs, "Recognition using region correspondences," *International Journal of Computer Vision*, **25**, 2 (1996), pp. 141–162.

[3] Beis, Jeff, and David G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," *Conference on Computer Vision and Pattern Recognition*, Puerto Rico (1997), pp. 1000–1006.

[4] Crowley, James L., and Alice C. Parker, "A representation for shape based on peaks and ridges in the difference of low-pass transform," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **6,** 2 (1984), pp. 156–170.

[5] Edelman, Shimon, Nathan Intrator, and Tomaso Poggio, "Complex cells and object recognition," Unpublished Manuscript, preprint at `http://www.ai.mit.edu/~edelman/mirror/nips97.ps.Z`

[6] Grimson, Eric, and Thomás Lozano-Pérez, "Localizing overlapping parts by searching the interpretation tree," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **9** (1987), pp. 469–482.

[7] Ito, Minami, Hiroshi Tamura, Ichiro Fujita, and Keiji Tanaka, "Size and position invariance of neuronal responses in monkey inferotemporal cortex," *Journal of Neurophysiology*, **73,** 1 (1995), pp. 218–226.

[8] Lindeberg, Tony, "Scale-space theory: A basic tool for analysing structures at different scales", *Journal of Applied Statistics*, **21,** 2 (1994), pp. 224–270.

[9] Lindeberg, Tony, "Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus-of-attention," *International Journal of Computer Vision*, **11,** 3 (1993), pp. 283–318.

[10] Logothetis, Nikos K., Jon Pauls, and Tomaso Poggio, "Shape representation in the inferior temporal cortex of monkeys," *Current Biology*, **5,** 5 (1995), pp. 552–563.

[11] Lowe, David G., "Three-dimensional object recognition from single two-dimensional images," *Artificial Intelligence*, **31,** 3 (1987), pp. 355–395.

[12] Lowe, David G., "Fitting parameterized three-dimensional models to images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **13,** 5 (1991), pp. 441–450.

[13] Murase, Hiroshi, and Shree K. Nayar, "Visual learning and recognition of 3-D objects from appearance," *International Journal of Computer Vision*, **14,** 1 (1995), pp. 5–24.

[14] Nelson, Randal C., and Andrea Selinger, "Large-scale tests of a keyed, appearance-based 3-D object recognition system," *Vision Research*, **38,** 15 (1998), pp. 2469–88.

[15] Ohba, Kohtaro, and Katsushi Ikeuchi, "Detectability, uniqueness, and reliability of eigen windows for stable verification of partially occluded objects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **19,** 9 (1997), pp. 1043–48.

[16] Perrett, David I., and Mike W. Oram, "Visual recognition based on temporal cortex cells: viewer-centered processing of pattern configuration," *Zeitschrift für Naturforschung C*, **53c** (1998), pp. 518–541.

[17] Pope, Arthur R. and David G. Lowe, "Learning probabilistic appearance models for object recognition," in *Early Visual Learning*, eds. Shree Nayar and Tomaso Poggio (Oxford University Press, 1996), pp. 67–97.

[18] Schiele, Bernt, and James L. Crowley, "Object recognition using multidimensional receptive field histograms," *Fourth European Conference on Computer Vision*, Cambridge, UK (1996), pp. 610–619.

[19] Schmid, C., and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE PAMI*, **19,** 5 (1997), pp. 530–534.

[20] Swain, M., and D. Ballard, "Color indexing," *International Journal of Computer Vision*, **7,** 1 (1991), pp. 11–32.

[21] Tanaka, Keiji, "Mechanisms of visual object recognition: monkey and human studies," *Current Opinion in Neurobiology*, **7** (1997), pp. 523–529.

[22] Treisman, Anne M., and Nancy G. Kanwisher, "Perceiving visually presented objects: recognition, awareness, and modularity," *Current Opinion in Neurobiology*, **8** (1998), pp. 218–226.

[23] Zhang, Z., R. Deriche, O. Faugeras, Q.T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Artificial Intelligence*, **78,** (1995), pp. 87-119.