

---

## Content-based indexing of images and video

Alex Pentland

*Phil. Trans. R. Soc. Lond. B* 1997 **352**, 1283-1290  
doi: 10.1098/rstb.1997.0111

---

### Email alerting service

Receive free email alerts when new articles cite this article - sign up in the box at the top right-hand corner of the article or click [here](#)

---

To subscribe to *Phil. Trans. R. Soc. Lond. B* go to: <http://rstb.royalsocietypublishing.org/subscriptions>

---

# Content-based indexing of images and video

ALEX PENTLAND

*The Media Laboratory, Massachusetts Institute of Technology, 20 Ames Street, Cambridge, MA 02139, USA*  
(sandy@media.mit.edu)

## SUMMARY

By representing image content using probabilistic models of an object's appearance we can obtain semantics-preserving compression of the image data. Such compact representations of an image's salient features allow rapid computer searches of even large image databases. Examples are shown for databases of face images, a video of American sign language (ASL), and a video of facial expressions.

## 1. INTRODUCTION: THE PROBLEM

The traditional model for the search of stored images and video, both as a model of human memory and as a model for computerized search, has been to create propositional annotations that describe the content of the image, and then enter these annotations into a standard database or semantic net. The images themselves are not really part of the memory or database; they are only referenced by computer text strings or mental propositions.

The problem with this approach is that the old saying 'a picture is worth 1000 words' is an understatement. In most images there are literally hundreds of objects that could be referenced, and each imaged object has a long list of attributes. Even worse, spatial relationships are important in understanding image content, so that complete annotation of an image with  $n$  objects each with  $m$  attributes requires  $O(n^2m^2)$  database entries. And if we must also consider relations among images, such a memory indexing model quickly becomes intractable.

During the last few years, however, there has been a major change in thinking about image databases. The key conceptual breakthrough was the idea of *content-based indexing*, allowing images and audio to be searched *directly* instead of relying on keyword (or propositional) annotation. The Massachusetts Institute of Technology (MIT) 'Photobook' system (Pentland & Picard 1994; Pentland *et al.* 1994) together with efforts such as the IBM 'query-by-image-content' (Faloutsos *et al.* 1994) system or the ISS project in Singapore (Smoliar & Zhang 1994), have demonstrated that such a search is both possible and useful. The idea has since become the focus of dozens of special journal issues, workshops, and the like. There have also been considerable commercial successes with this technology. Photobook-like technology is now available as an IBM Ultimedia Manager system and as the Virage Engine (Virage). Consequently simple examples of content-based indexing is now widely available on a variety of platforms and as an option within several traditional database systems.

In this paper I will review the Photobook concept of content-based indexing, as developed by Professors Picard, Sclaroff, and myself and originally described in an earlier paper (Pentland *et al.* 1996), and give several examples drawn from recent work by my students and myself. For a full account of our research, including current papers, some computer code, and on-line demonstrations, see our web site at (<http://www-white.media.mit.edu/vismod>).

### (a) *The problem: semantic indexing of image content*

The problem is that to make a user- and purpose-independent image database we must annotate everything in the images and all the relations between them. Text databases avoid this problem by using strings of characters e.g., words that are a consistent encoding of the database's semantic content. Thus, questions about the database's semantic content can be answered by simply comparing sets of text strings. Because this search is efficient, users can search for their answers at query time rather than having to pre-annotate everything.

To accomplish the same thing for image databases, we must be able to efficiently compare the images themselves, to see if they have the same, or more generally similar, semantic content. There is, of course, a trade-off between how much work you do at input time and how much you do at query time. For instance, one could try to precompute the answers to all possible queries, so that no search would be required. Alternatively, one could search the raw images themselves, repeating all of the low-level image processing tasks for each query.

For image databases there is a compelling argument for employing a pre-purposive 'iconic' level of representation. It does not make sense to try to precompute a 'general purpose'—a completely symbolic representation of image content—because the number of possibly interesting geometric relations is combinatorially explosive. Consequently, the output of our precomputation

must be image-like data structures where the geometric relationships remain implicit. On the other hand, it does make sense to precompute as much as is possible because low-level image operations are so expensive.

These precomputed image primitives must play a role similar to that of letters and words in a database query sentence. The user can use them to describe 'interesting' or 'significant' visual events, and then let the computer search for instances of similar events. For instance, the user should be able to select a video clip of a lush waterfall, and be able to ask for other video sequences in which more of the same 'stuff' occurs. The computer would then examine the precomputed decomposition of the waterfall sequence, and characterize it in terms of texture-like primitives such as spatial and temporal energy. It could then search the precomputed decomposition of other video clips to find places where there is a similar distribution of primitives.

Alternatively, the user might circle a 'thing' like a person's face, and ask the computer to track that person within the video clip, or ask the computer to find other images where the same person appears. In this case the computer would characterize the person's two-dimensional image appearance in terms of primitives such as edge geometry and the distribution of normalized intensity, and then either track this configuration of features over time or search other images for similarly-arranged conjunctions of the same features.

These two types of semantic indexing, using texture-like descriptions of 'stuff' and object-like descriptions of 'things' constitute the two basic types of image search operation in our system. These two types of description seem to be fundamentally different in human vision, and correspond roughly to the distinction between mass nouns and count nouns in language. Note that both types of image query can operate on the same image primitives, e.g., the energy in different band-pass filters, but they differ in how they group these primitives for comparison. The 'stuff' comparison method pools the primitives without regard to detailed local geometry, while the 'things' method preserves local geometry.

## 2. SEMANTICS-PRESERVING IMAGE COMPRESSION

The ability to search at query time for instances of the same or similar image events depends on the two conditions discussed below.

(i) There must be a similarity metric for comparing objects or image properties, e.g. shape, texture, colour, object relationships, that matches human judgements of similarity. This is not to say that the computation must somehow mimic the human visual system, but rather that computer and human judgements of similarity must be generally correlated. Without this, the images that the computer finds will not be those desired by the human user.

(ii) The search must be efficient enough to be interactive. A search that requires minutes per image is simply not useful in a database with millions of images. Furthermore, interactive search speed makes it possible

for users to recursively refine a search by selecting examples from the currently retrieved images and using these to initiate a new select-sort-display cycle. Thus, users can iterate a search to quickly 'zero in on' what they are looking for.

Consequently, we believe that the key to solving the image database problem is semantics-preserving image compression, compact representations that preserve essential image similarities. This concept is related to some of the 'semantic bandwidth compression' ideas put forth in the context of image compression (Picard 1992). Image coding has utilized semantics primarily through efforts to compute a compact image representation by exploiting knowledge about the content of the image. A simple example of semantic bandwidth compression is coding the people in a scene using a model specialized for people, and then using a different model to code the background.

In the image database application, compression is no longer the singular goal. Instead, it is important that the coding representation be (i) 'perceptually complete', and (ii) 'semantically meaningful'. The first criterion will typically require a measure of perceptual similarity. Measures of similarity on the coefficients of the coded representation should roughly correlate with human judgements of similarity on the original images.

The definition of 'semantically meaningful' is that the representation gives the user direct access to the parts of the image content that are important for their application. That is, it should be easy to map the coefficients that represent the image to 'control knobs' that the user finds important. For instance, if the user wishes to search among faces, it should be easy to provide control knobs that allow selection of facial expressions or selection of features such as moustaches or glasses. If the user wishes to search among textures, then it should be easy to select features such as periodicity, orientation, or roughness.

Having a semantics-preserving image compression method allows you to quickly search through a large number of images because the representations are compact. It also allows you to find those images that have perceptually similar content by simply comparing the coefficients of the compressed image code. Thus, in our view the image database problem requires development of semantics-preserving image compression methods.

### (a) *Algorithm design*

How can one design 'semantics-preserving image compression' algorithms for particular objects or textures? The general theoretical approach is to use probabilistic modelling of low-level, two-dimensional representations of regions of the image data that correspond to objects or textures of interest.

To perform such modelling, one first transforms portions of the image into a low-dimensional coordinate system that preserves the general perceptual quality of the target object's image, and then use standard statistical methods, such as expectation maximization of a Gaussian mixture model, to learn the range of

appearance that the target exhibits in that new coordinate system. The result is a very simple, neural-net-like representation of the target class's appearance, which can be used to detect occurrences of the class, to compactly describe its appearance, and to efficiently compare different examples from the same class.

As different parts of the image have different characteristics, we must use a variety of representations, each tuned for a specific type of image content. For instance, to represent images of 'things', which requires preservation of detailed geometric relations, we use the Karhunen–Loève transform (KLT), which is also called the principal components analysis (PCA). It is a classic mathematical result that the KLT provides an optimally-compact linear basis with respect to root mean square error for a given class of signal. For characterization of texture classes, we use an approach based on the Wold decomposition. This transform separates 'structured' and 'random' texture components, allowing extremely efficient encoding of textured regions, e.g., by using the KLT on the separated components, while preserving their perceptual qualities (Picard & Kabir 1993).

Given several examples of a target class,  $\Omega$ , in such a low-dimensional representation, it is straightforward to model the probability distribution function,  $p(x|\Omega)$ , of its image-level features ( $x$ ), as a mixture of Gaussians, thus obtaining a low-dimensional, parametric appearance model for the target class (Moghaddam & Pentland 1995). Once the target class's probability distribution function (PDF) has been learned, we can use Bayes' rule to perform 'maximum *a posteriori*' (MAP) detection and recognition.

The use of parametric appearance models to characterize the PDF of an object's appearance in the image is a generalization of the idea of view-based representation, as advocated by Ullman & Basri (1991) and Poggio & Edelman (1990). As originally developed, the idea of view-based recognition was to accurately describe the spatial structure of the target object by interpolating between various views. However, in order to describe natural objects such as faces or hands, we have found it necessary to extend the notion of 'view' to include characterizing the range of geometric and feature variation, as well as the likelihoods associated with such variation.

This approach is typified by our face recognition research (Turk & Pentland 1991; Moghaddam & Pentland 1995) which uses linear combinations of eigenvectors to describe a space of target appearances, and then characterizes the PDF of the target's appearance within that space. This method has been shown to be very powerful for detection and recognition of human faces, hands, and facial expressions (Moghaddam & Pentland 1995). Other researchers have used extensions of this basic method to recognize industrial objects and household items (Murase & Nayar 1994).

### (b) Comparison with other approaches

During the last few years many researchers have proposed a variety of image indexing methods, based on shape, colour, or combinations of such indices (Faloutsos *et al.* 1994; Smoliar & Zhang 1994). The general approach is to calculate an approximately invariant statistic, such as colour histogram or invariants of shape moments, and use that to stratify or partition the image database. Such partitioning allows users to limit the search space when looking for a particular image, and has proven to be quite useful for small image databases.

The difference between these methods and ours is that they emphasize computing a discriminant that can reject many false matches, whereas ours can encode the image data to the accuracy required to retain 'all' of its perceptually salient aspects. Generally speaking, the coefficients these earlier efforts have produced are not sufficiently meaningful to reconstruct the perceptually salient features of the image. For instance, one cannot reconstruct an image region from its moment invariants or its colour histogram. In contrast, the models we present use coefficients which allow reconstruction. Figure 1 shows three reconstructions using appearance, shape, and texture descriptions of image content.

In our view, the problem with using invariants or discriminants is that significant semantic information is irretrievably lost. For instance, do we really want our database to think that apples, Ferrarris, and tongues are 'the same' just because they have the same colour histogram? Discriminants give a way to limit search space, but do not answer 'looks like' questions except within constrained data sets. In contrast, when the coefficients

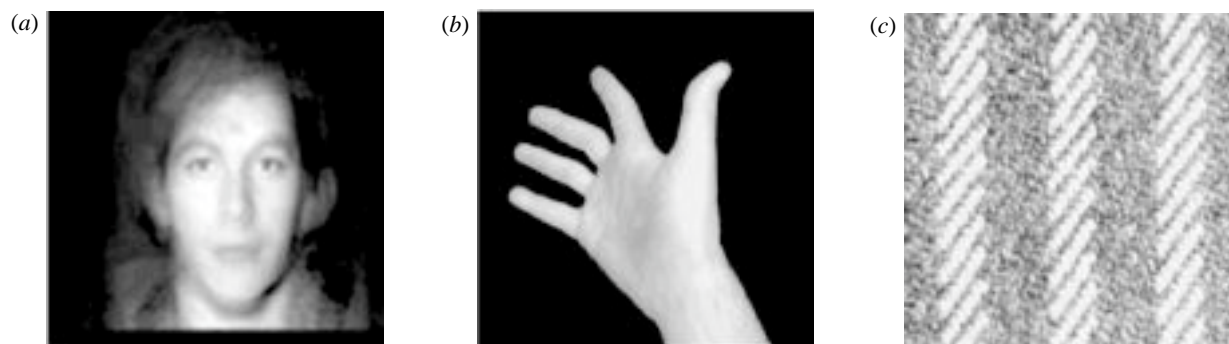


Figure 1. Images reconstructed from coefficients used for database search: (a) 30 coefficients, (b) 100 coefficients, and (c) 60 coefficients. From Pentland *et al.* (1996).

provide a perceptually complete representation of the image information, then things the database thinks are 'the same' actually look the same.

Another important consequence of representational completeness is that we can ask a wide range of questions about the image, rather than being limited to only a few predefined questions. For instance, it requires only a few matrix multiples per image to calculate indices such as colour histograms or moment invariants from our coefficients. The point is that if you start with a relatively complete representation, then you are not limited in the types of questions you can ask; whereas, if you start by calculating discriminants, then you are limited to queries about those particular measures only.

### 3. THE PHOTOBOK SYSTEM

Photobook is a computer system that allows the user to browse large image databases quickly and efficiently, using both text annotation information in an artificial intelligence database and by having the computer search the images directly based on their content (Pentland & Picard 1994; Pentland *et al.* 1996). This allows people to search in a flexible and intuitive manner, using semantic categories and analogies, e.g., 'show me images with text annotations similar to those of this image but shot in Boston', or visual similarities, e.g., 'show me images that have the same general appearance as this one'.

Interactive image browsing is accomplished using a Motif interface. This interface allows the user to first select the category of images they wish to examine; e.g., pictures of white males over 40 years of age, or images of mechanic's tools, or cloth samples for curtains. Photobook then presents the user with the first screenful of these images (see figure 4); the rest of the images can be viewed by 'paging' through them one screen at a time.

Users most frequently employ Photobook by selecting one (or several) of the currently-displayed images, and asking it to sort the entire set of images in terms of their similarity to the selected image or set of images. By selecting several example images the user is providing information about the distribution of visual parameters that characterize items of interest. Photobook uses such multiple examples to make an improved estimate of the search parameters. Photobook then re-presents the images to the user, now sorted by similarity to the selected images. The select-sort-redisplay cycle typically takes less than 1 s. When searching for a particular item, users quickly scan the newly-displayed images, and initiate a new select-sort-redisplay cycle every 2 or 3 s.

#### (a) *An image database example: faces*

One of the clearest examples of content-based indexing is face recognition, and the Photobook face recognition system was certified by the US Army as being extremely accurate (Phillips *et al.* 1997). Face recognition within this system is accomplished by first determining the PDF for face images within a low-



Figure 2. The first eight eigenfaces.

dimensional eigenspace calculated directly from contrast-normalized image data. Knowledge of this distribution then allows the face and facial features to be precisely located, and compared along meaningful dimensions. The following gives a brief description of how this is accomplished within the Photobook system (for additional detail see Moghaddam & Pentland (1995)).

#### (b) *Face and feature detection*

The standard detection paradigm in image processing is that of normalized correlation or template matching. However, this approach is only optimal in the case of a *deterministic* signal embedded in white Gaussian noise. When we begin to consider a target class detection problem—e.g., finding a generic human face or a human hand in a scene—we must incorporate the underlying probability distribution of the object of interest. Subspace or eigenspace methods, such as the KLT and PCA, are particularly well-suited to such a task since they provide a compact and parametric description of the object's appearance and also automatically identify the essential components of the underlying statistical variability.

In particular, the eigenspace formulation leads to a powerful alternative to standard detection techniques such as template matching or normalized correlation. The reconstruction error, or residual, of the KLT expansion is an effective indicator of a match. The residual error is easily computed using the projection coefficients and the original signal energy. This detection strategy is equivalent to matching with a linear combination of *eigentemplates* and allows for a greater range of distortions in the input signal (including lighting, and moderate rotation and scale). Some of the low-order eigentemplates for a human face are shown in figure 2. In a statistical signal detection framework, the use of eigentemplates has been shown to be orders of magnitude better than standard matched filtering (Moghaddam & Pentland 1995).

Using this approach the target detection problem can be reformulated from the point of view of a MAP estimation problem. In particular, given the visual field, estimate the position (and scale) of the subimage which is most representative of a specific target class  $\Omega$ . Computationally this is achieved by sliding an  $m$ -by- $n$  observation window throughout the image and at each location computing the likelihood that the given observation  $x$  is an instance of the target class  $\Omega$ —i.e.,  $p(x|\Omega)$ . After this probability map is computed, the location corresponding to the highest likelihood can be selected as the MAP estimate of the target location.

**(c) The face processor**

This MAP-based face finder has been employed as the basic building block of an automatic face recognition system. The block diagram of the face finder system is shown in figure 3. It consists of object detection and alignment, contrast normalization, feature extraction, followed by recognition and, optionally, facial coding. Figure 3*b–e* illustrates the operation of the detection and alignment stage on a natural image containing a human face.

The first step in this process is illustrated in figure 3*c* where the MAP estimate of the position and scale of the face are indicated by the cross-hairs and bounding box. Once these regions have been identified, the estimated scale and position are used to normalize for translation and scale, yielding a standard 'head-in-the-box' format image, figure 3*d*. A second feature detection stage operates at this fixed scale to estimate the position of four facial features: the left and right eyes, the tip of the nose and the centre of the mouth, figure 3*e*. Once the facial features have been detected, the face image is warped to align the geometry and shape of the face with that of a canonical model. Then the facial region is extracted, by applying a fixed mask, and subsequently normalized for contrast. This geometrically aligned and normalized image is then projected onto the set of eigenfaces shown in figure 2.

The projection coefficients obtained by comparison of the normalized face and the eigenfaces form a feature vector which accurately describes the appearance of the face. This feature vector can therefore be used for facial recognition, as well as for facial image coding. Figure 4 shows a typical result when using the eigenface feature vector for face recognition. The image in the upper left is the one to be recognized and the remainder are the most similar faces in the database, ranked by facial similarity left to right, and top to bottom. The top three matches in this case, are images

of the same person taken a month apart and at different scales. The recognition accuracy of this system, defined as the percent correct rank-one matches, is 99% (Moghaddam & Pentland 1995).

#### 4. VIDEO DATABASE EXAMPLES: HUMAN EXPRESSION AND GESTURE

Given the rapid progress that has occurred with databases of static images, researchers are now beginning to turn their attention to the problem of video databases. Interestingly, from a practical point of view the problem of video databases is largely equivalent to the problem of interpreting human behaviour in video. That is, most video is about people, and the identity and behaviour of the people in the video are the most important element of its content.

Again, the general theoretical approach we have taken to interpretation of video is that of MAP interpretation on low-level, two-dimensional representations of regions of the image data (Pentland 1996). As illustrated by the face database example above, the appearance of a target class  $\Omega$ , e.g., the probability distribution function  $p(x|\Omega)$  of its image-level features  $x$ , can be characterized by use of a low-dimensional parametric appearance model. Once such a PDF has been learned, it is straightforward to use it in a MAP estimator in order to detect and recognize target classes. Behaviour recognition is accomplished in a similar manner; these parametric appearance models are tracked over time, and their time evolution  $p(x(t)|\Omega)$ , characterized probabilistically to obtain a spatio-temporal behaviour model (Pentland 1996). Incoming spatio-temporal data can then be compared to the spatio-temporal PDF of each of the various behaviour models using elastic matching methods, such as dynamic time warping (Darrell & Pentland 1993) or hidden Markov modelling (Starner & Pentland 1995).

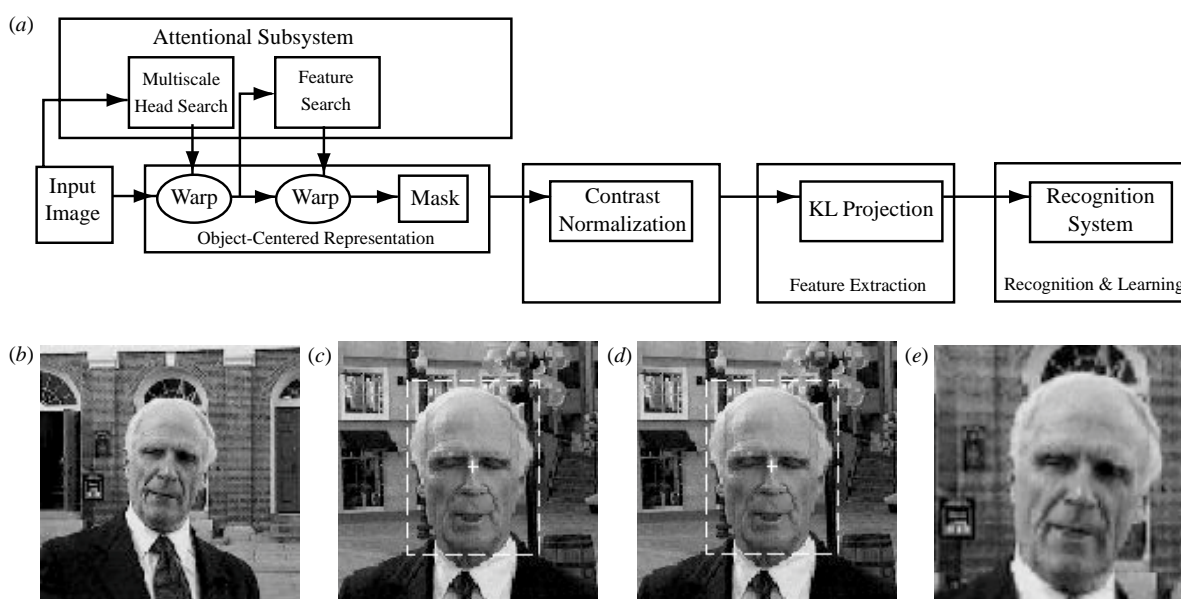


Figure 3. (a) The face processing system, (b) original image, (c) position and scale estimate, (d) normalized head image, (e) position of facial features.

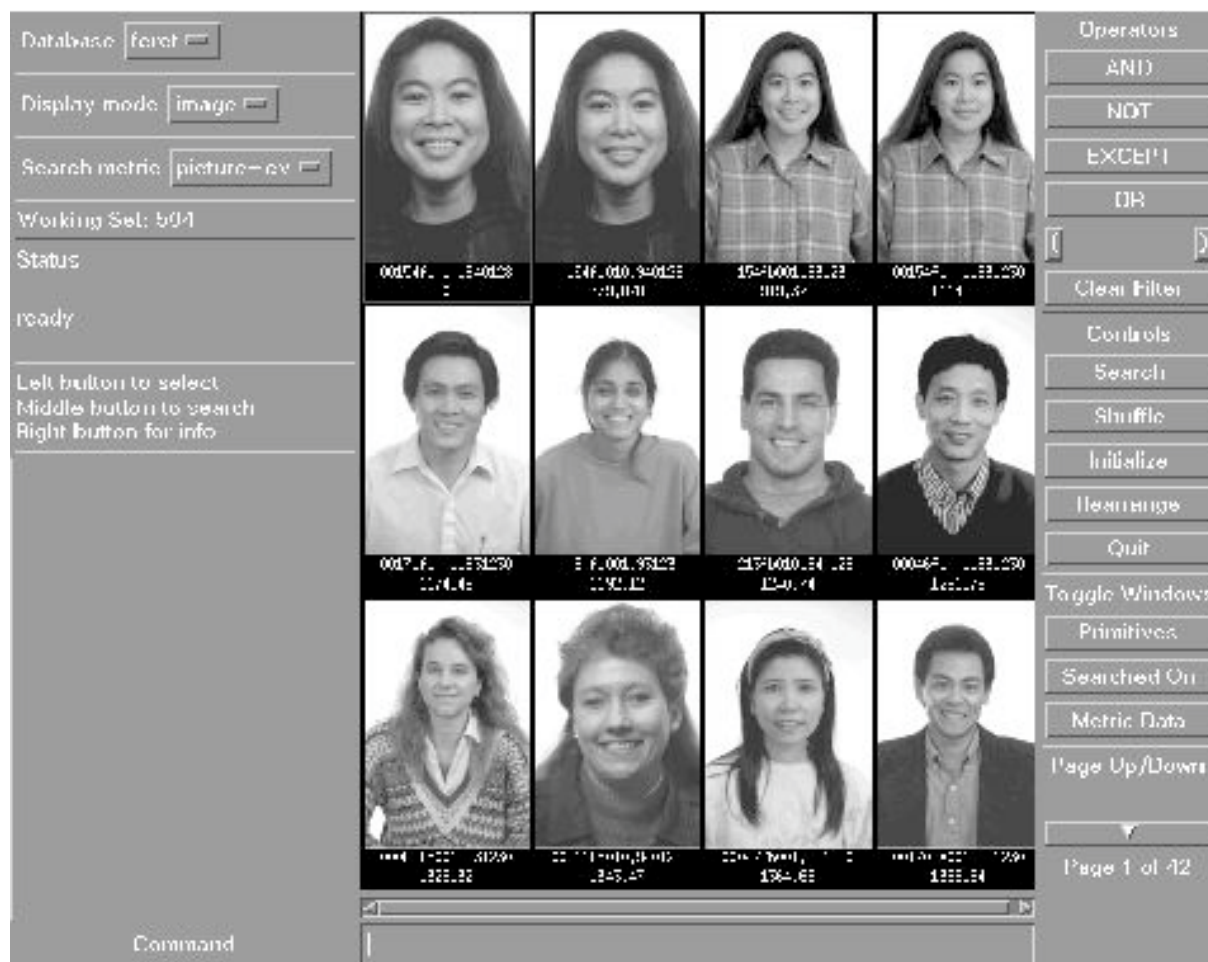


Figure 4. Searching for similar faces in a database, using the photobook image database tool (Pentland *et al.* 1996).

**(a) Example: gesture recognition**

The first example of this approach is recognition of human hand gestures. To address this problem we modelled the person's hand motion as a Markov device, with internal states that have their own particular distribution of appearance and interstate transition probabilities. Because the internal states of a human are not directly observable, they must be determined through an indirect estimation process, using the person's movements as measurements. One efficient and robust method of accomplishing this is to use the Viterbi recognition methods developed for use with hidden Markov models (HMMs) (Rabiner & Juang 1996).

This general approach is similar to that taken by the speech recognition community. The difference in our approach is that internal state is not thought of as being just words or sentences; the internal states can also be actions or even intentions. Moreover, the input is not just audio filter banks but also facial appearance, body movement, and vocal characteristics (such as pitch) are used to infer the user's internal state. One good example that employs this approach to behaviour recognition is our system for reading American sign language (ASL).

The ASL reader is a real-time system that performs 99% accurate classification of a forty-word subset of

ASL. Thad Starner is shown using this system in figure 5. The accurate classification performance of this system is particularly impressive because in ASL the hand movements are rapid and continuous, and exhibit large coarticulation effect. (For additional detail see Starner & Pentland (1995).)



Figure 5. Real-time reading of American sign language, with Thad Starner doing the signing






expressions	smile	surprise	anger	disgust	raise brow
template					
smile	12	0	0	0	0
surprise	0	10	0	0	0
anger	0	0	9	0	0
disgust	0	0	1	10	0
raise brow	0	0	0	0	8
success	100%	100%	90%	100%	100%

Figure 6. Results of facial expression recognition using spatio-temporal motion energy models. This result is on based on 12 image sequences of smile, 10 image sequences of surprise, anger, disgust, and raise eyebrow. Success rate for each expression is shown in the bottom row. Overall recognition rate is 98.0%.

(b) *Example: expression recognition*

A second example is the recognition of facial expression. We have addressed this problem by learning spatio-temporal motion-energy appearance models for the face. That is, for each facial expression we have created a motion-energy appearance model that expresses the amount and direction of motion that one would expect to see at each point on the face. These simple, biologically-plausible motion-energy models can be used for expression recognition by comparing the motion energy observed for a particular face to an 'average' motion-energy model for each expression. To classify an expression one compares the observed facial motion energy with each of these models, and then picks the expression with the most similar pattern of motion.

This method of expression recognition has been applied to a database of 52 image sequences of eight subjects making various expressions. In each image sequence the motion energy was measured, compared to each of the models, and the expression classified, generating the confusion matrix shown in figure 6. This figure shows just one incorrect classification, giving an overall recognition rate of 98.0%. (For additional details see Essa & Pentland (1994, 1995).)

## 5. CONCLUSION

The Photobook system is an interactive tool for browsing and searching images and image sequences. The key idea behind this suite of tools is *semantics-preserving image compression*, which reduces images to a small set of perceptually significant coefficients. Compact semantics-preserving representations of image and video content can be created by learning an appearance model of a target class,  $\Omega$ , in a low-dimensional representation, such as is produced by the KLT. Once the

probability distribution function  $p(x|\Omega)$  of the target class has been learned, we can use Bayes's rule to perform MAP detection and recognition. The same approach can be extended to time by use of methods such as dynamic time warping and HMM.

The work described here was funded by British Telecom. I thank co-authors and collaborators Rosalind Picard, Stan Sclaroff, Irfan Essa, Fang Liu, Baback Moghaddam, Matthew Turk, Thad Starner, Bradley Horowitz, and Tom Minka for their contributions. Portions of this paper have appeared in Pentland (1996) and Pentland *et al.* (1996), and in the 1996 Image Understanding Workshop Proceedings (San Francisco: Morgan Kaufmann). Papers and technical reports on all aspects of this technology are available at <http://www-white.media.mit.edu/vismod> or by anonymous FTP from [whitechapel.media.mit.edu](http://whitechapel.media.mit.edu)

## REFERENCES

- Darrell, T. & Pentland, A. 1993 Space-time gestures. In *IEEE Conf. on Vision and Pattern Recognition, NY*, pp. 335–340.
- Essa, I. & Pentland, A. 1994 A vision system for observing and extracting facial action parameters. In *IEEE Conf. on Computer Vision and Pattern Recognition, Seattle, WA*, pp. 76–83.
- Essa, I. & Pentland, A. 1995 Facial expression recognition using a dynamic model and motion energy. In *IEEE Int. Conf. on Computer Vision, Cambridge, MA*, pp. 360–367.
- Faloutsos, C., Barber, R., Equitz, W., Flickner, M., Hafner, J., Niblack, W. & Petkovic, D. 1994 Efficient and effective querying by image content. *J. Intell. Inform. Syst.* **3**, 231–262.
- Jones, M., & Poggio, T. 1995 Model-based matching of line drawings by linear combinations of prototypes. In *IEEE Int. Conf. on Computer Vision, Cambridge, MA*, pp. 368–375.
- Moghaddam, B. & Pentland, A. 1995 Probabilistic visual learning for object detection. In *IEEE Int. Conf. on Computer Vision, Cambridge, MA*, pp. 786–793.
- Murase, H. & Nayar, S. 1994 Visual learning and recognition of 3D objects from appearance. *IEEE Int. J. Comp. Vision* **14**, 5–24.
- Pentland, A. 1996 Smart rooms, smart clothes. *Sci. Am.* **274**(4), 68–76.



- Pentland, A. & Picard, R. W. 1994 Video and image semantics: advanced tools for telecommunications. *IEEE Multimedia* **1**(2), 73–75.
- Pentland, A., Picard, R. W., Sclaroff, S. *et al.* 1996 Photobook: tools for content-based manipulation of image databases. *Int. J. Comp. Vision* **18**(3), 233–254.
- Phillips, P. J., Moon, H., Rauss, P. & Rizvi, S. A. 1997 The FERET September 1996 database and evaluation procedure. In *Proc. 1st Int. Conf. on Audio and Video-based Biometric Person Authentication, Crans-Montana, Switzerland, 12–14 March 1997*.
- Picard, R. W. 1992 Random field texture coding. In *Soc. for Information Display Int. Symp. Digest, vol. XXXIII*, pp. 685–688.
- Picard, R. W. & Kabir, T. 1993 Finding similar patterns in large image databases. *Proc. ICASSP, Minneapolis, MN, vol. V*, pp. 161–164.
- Picard, R. W. & Liu, F. 1994 A new Wold ordering for image similarity. In *Proc. ICASSP, Adelaide, Australia*.
- Poggio, T. & Edelman, S. 1990 A network that learns to recognize three-dimensional objects. *Nature* **33**, 263–266.
- Rabiner, L. & Juang, B. 1996 An introduction to hidden Markov models. In *IEEE ICASSP Magazine*, pp. 4–16.
- Smoliar, S. & Zhang, H. 1994 Content-based video indexing and retrieval. *IEEE Multimedia* **1**(2), 62–72.
- Starner, T. & Pentland, A. 1995 Visual recognition of American sign language using hidden Markov models. In *Proc. Int. Workshop on Automatic Face- and Gesture-Recognition, Zurich, Switzerland*.
- Turk, M. & Pentland, A. 1991 Eigenfaces for recognition. *J. Cognitive Neurosci.* **3**(1), 71–86.
- Ullman, S. & Basri, R. 1991 Recognition by linear combinations of models. *IEEE Trans. Pattern Analysis and Machine Vision* **13**, 992–1006.
- Virage <http://www.virage.com>.