

# Modeling the Statistics of Image Features and Associated Text

Kobus Barnard, Pinar Duygulu, and David Forsyth  
Computer Division, University of California, Berkeley  
{kobus, duygulu, daf}@cs.berkeley.edu

## ABSTRACT

We present a methodology for modeling the statistics of image features and associated text in large datasets. The models used also serve to cluster the images, as images are modeled as being produced by sampling from a limited number of combinations of mixing components. Furthermore, because our approach models the joint occurrence image features and associated text, it can be used to predict the occurrence of either, based on observations or queries. This supports an attractive approach to image search as well as novel applications such as suggesting illustrations for blocks of text (auto-illustrate) and generating words for images outside the training set (auto-annotate).

In this paper we illustrate the approach on 10,000 images of work from the Fine Arts Museum of San Francisco. The images include line drawings, paintings, and pictures of sculpture and ceramics. Many of the images have associated free text whose nature varies greatly, from physical description to interpretation and mood. We incorporate statistical natural language processing in order to deal with free text. We use WordNet to provide semantic grouping information and to help disambiguate word senses, as well as emphasize the hierarchical nature of semantic relationships.

**Keywords:** Image retrieval, recognition, learning image semantics, hierarchical clustering, aspect model

## 1. INTRODUCTION

It is a remarkable fact that, while text and images are separately ambiguous, jointly they tend not to be; this is probably because the writers of text descriptions of images tend to leave out what is visually obvious (the colour of flowers, etc.) and to mention properties that are very difficult to infer using vision (the species of the flower, say). We exploit this phenomenon to organize image databases using both image features and associated text using a probabilistic model inspired by one developed for text<sup>1</sup>). By integrating the two kinds of information during model construction, the system learns links between the image features and semantics which can be exploited for better browsing (§6.1), better search (§6.2), and novel applications such as associating images with words (“auto-illustrate”, §6.3) and words with pictures (“auto-annotate”, §6.4), the latter having clear ties to object recognition.

The system works by modeling the statistics of word and feature occurrence and co-occurrence. We use a hierarchical structure which further encourages semantics through levels of generalization, as well as being a natural choice for browsing applications. An additional advantage of our approach is that since it contains processes for predicting image components—words and features—from observed image components, we can use the prediction performance on a held out set to compare model variants. In general we can measure the performance of the model in ways not typically available for image retrieval systems. This is exciting because an effective performance measure is an important tool for further improving the model.

A number of other researchers have introduced systems for searching image databases. There are reviews in<sup>2,3</sup>. A few systems combine text and image data. Search using a simple conjunction of keywords and image features is provided in Blobworld<sup>4</sup>. Webseer<sup>5</sup> uses similar ideas for query of images on the web, but also indexes the results of a few automatically estimated image features. These include whether the image is a photograph or a sketch and notably the output of a face finder. Going further, Cascia et al integrate some text and histogram data in the indexing<sup>6</sup>. Others have also experimented with using image features as part of a query refinement process<sup>7</sup>. Enser and others have studied the nature of the image database query task<sup>8-10</sup>. Srihari and others have used text information to disambiguate image features, particularly in face finding applications<sup>11-15</sup>.

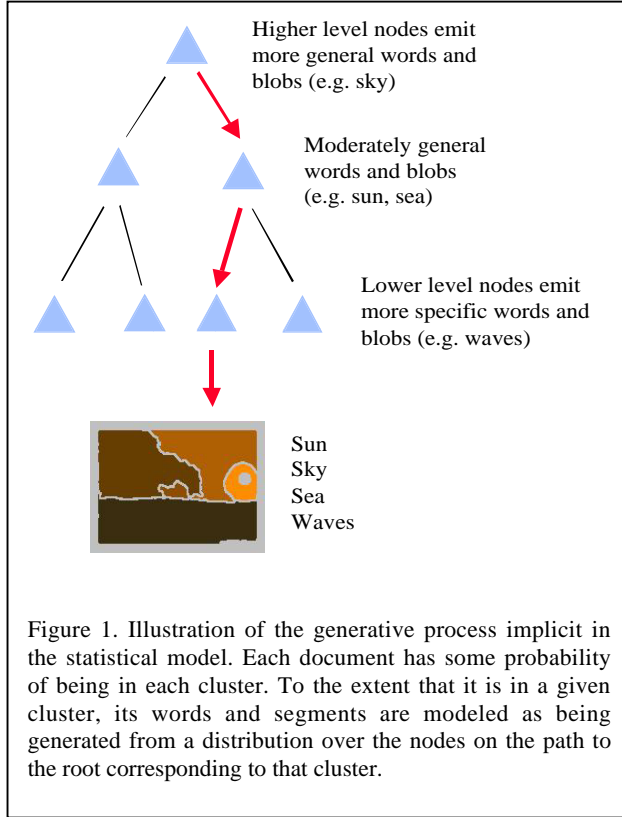
## 2. MODELS

We consider two models, both inspired by one proposed for text by Hofmann<sup>1,16</sup>, and first applied to multiple data sources (text and image features) in<sup>2</sup> and<sup>17</sup> (respectively). The first model is a hierarchical combination of the assymmetric clustering model which maps documents into clusters, and the symmetric clustering model which models the joint distribution of documents and features (the “aspect” model). The data is modeled as being generated by a fixed hierarchy of nodes, with the leaves of the hierarchy corresponding to clusters. Each node in the tree has some probability of generating each word, and similarly, each node has some probability of generating an image segment with given features. The documents belonging to a given cluster are modeled as being generated by the nodes along the path from the leaf corresponding to the cluster, up to the root node, with each node being weighted on a document and cluster basis. Conceptually a document belongs to a specific cluster, but given finite data we can only model the probability that a document belongs to a cluster, which essentially makes the clusters soft. We note also that clusters which have insufficient membership are extinguished, and therefore, some of the branches down from the root may end prematurely.

The model is illustrated further in Figure 1. To the extent that the sunset image illustrated is in the third cluster, as indicated in the figure, its words and segments are modeled by the nodes along the path shown. Taking all clusters into consideration, the document is modeled by a sum over the clusters, weighted by the probability that the document is in the cluster. Mathematically, the process for generating the set of observations  $D$  associated with a document  $d$  can be described by

$$P(D | d) = \sum_c P(c) \prod_{i \in D} \left( \sum_l P(i | l, c) P(l | c, d) \right) \quad (1)$$

where  $c$  indexes clusters,  $i$  indexes items (words or image segments), and  $l$  indexes levels. Notice that  $D$  is a set of observations that includes both words and image segments.



In (1) there is a separate probability distribution over the nodes for each document. This is an advantage for some applications such as search as each document is individually characterized. However this model is expensive in space, and documents belonging mostly to the same cluster can be quite different because their distribution over nodes can differ substantially. Finally, when a new document is considered, as in the case with the “auto-annotate” application described below, the distribution over the nodes must be computed using an iterative process. This is because, as argued in<sup>18</sup>, the aspect model is not a truly generative model. Thus for some applications we propose a simpler variant of the model which uses a cluster dependent, rather than document dependent, distribution over the nodes. Documents are generated with this model according to

$$P(D) = \sum_c P(c) \prod_{i \in D} \left( \sum_l P(i | l, c) P(l | c) \right) \quad (2)$$

In training the average distribution,  $P(l | c)$ , is maintained in place of a document specific one; otherwise things are similar. We will refer to the standard model in (1) as Model I, and the model in (2) as Model II. Both models provide a joint distribution for words and image segments.

The probability for an item,  $P(i | l, c)$ , is conditionally independent, given a node in the tree. A node is uniquely specified by cluster and level. In the case of a word,  $P(i | l, c)$  is simply tabulated, being determined by the appropriate word counts during training. For image segments, we use Gaussian distributions over a number of features capturing some aspects of size, position, colour, texture, and shape. These features taken together form a feature vector  $X$ . Each node, subscripted by cluster  $c$ , and level  $l$ , specifies a probability distribution over image segments by the usual formula. In this work we assume independence of the features, as learning the full covariance matrix leads to precision problems. A reasonable compromise would be to enforce a block diagonal structure for the covariance matrix to capture the most important dependencies.

To train the model we use the Expectation-Maximization algorithm<sup>19</sup>. This involves introducing hidden variables  $H_{d,c}$  indicating that training document  $d$  is in cluster  $c$ , and  $V_{d,i,l}$  indicating that item  $i$  of document  $d$  was generated at level  $l$ . Additional details on the EM equations can be found in<sup>1</sup>.

Both the vertical and horizontal structure of the nodes are important. The vertical structure allows the generation of images of a variety of combinations of components (aspects, topics) without encoding all possibilities. The horizontal structure provides the clustering and modeling economy, which also facilitates training. The hierarchical structure provides further economy, but more importantly, can exploit the expected hierarchical nature of data, and similarly can learn some of that structure. Specifically, more general terms and more generic image segment descriptions will occur in the higher level nodes because they occur more often.

### 3. IMPLEMENTATION

We incorporate simple natural language processing in order to deal with free text and to take advantage of additional semantics available using natural language tools (see §4). For segmentation we use Normalized Cuts<sup>20,21</sup>. For this work we use a modest set of features, specifically region color and standard deviation, region average orientation energy (12 filters), and region size, location, convexity, first moment, and ratio of region area to boundary length squared.

#### 3.1 Data Set

We provide results for a database consisting of 10,000 images of work from the Fine Arts Museum of San Francisco. The images are extremely diverse, and include line drawings, paintings, sculpture, ceramics, antiques, and so on. Many of the images have associated free text provided by volunteers. The nature of this text varies greatly, from physical description to interpretation and mood. Descriptions can run from a short sentence to several hundred words, and were not written with machine interpretation in mind.

#### 3.2 Scale

Training on an large image collection requires sensitivity to scalability issues. A naive implementation of the method described in<sup>1</sup> requires a data structure for the vertical indicator variables which increases linearly with four parameters: the number of images, the number of clusters, the number of levels, and the number of items (words and image segments). The dependence on the number of images can be removed at the expense of programming complexity by careful updates in the EM algorithm as described here. In the naive implementation, an entire E step is completed before the M step is begun (or vice versa). However, since the vertical indicators are used only to weight sums in the M step on an image by images bases, the part of the E step which computes the vertical indicators can be interleaved with the part of the M step which updates sums based on those indicators. This means that the storage for the vertical indicators can be recycled, removing the dependency on the number of images. This requires some additional initialization and cleanup of the loop over points (which contains a mix of both E and M parts). Weighted sums must be converted to means after all images have been visited, but before the next iteration. The storage reduction also applies to the horizontal indicator variables (which has a smaller data structure). Unlike the naive implementation, our version requires having both a "new" and "current" copy of the model (e.g. means, variances, and word emission probabilities), but this extra storage is small compared with the overall savings.

## 4. LANGUAGE MODELS

We use WordNet<sup>22</sup> (an on-line lexical reference system, developed by the Cognitive Science Laboratory at Princeton University), to determine word senses and semantic hierarchies. Every word in WordNet has one or more senses each of which has a distinct set of words related through other relationships such as hyper- or hyponyms (IS\_A), holonyms (MEMBER\_OF) and meronyms (PART\_OF). Most words have more than one sense. Our current clustering model requires that the sense of each word be established. Word sense disambiguation is a long standing problem in Natural Language Processing and there are several methods proposed in the literature<sup>23-25</sup>. We use WordNet hypernyms to disambiguate the senses.

For example, in the Corel database, sometimes it is possible that one keyword is a hypernym of one sense of another keyword. In such cases, we always choose the sense that has this property. This method is less helpful for free text, where there are more, less carefully chosen, words. For free text, we use shared parentage to identify sense, because we assume that senses are shared for text associated with a given picture (as in Gale et. al's one sense per discourse hypothesis<sup>26</sup>).

Thus, for each word we use the sense which has the largest hypernym sense in common with the neighboring words. For example, figure 2 shows four available senses of the word path. Corel figure no. 187011 has keywords path, stone, trees and mountains. The sense chosen is path<-way<-artifact<-object.

The free text associated with the museum data varies greatly, from physical descriptions to interpretations and descriptions of mood. We used Brill's part of speech tagger<sup>27</sup> to tag the words; we retained only nouns, verbs, adjectives and adverbs, and only the hypernym synsets for nouns. We used only the six closest words for each occurrence of a word to disambiguate its sense. Figure 3 shows a typical record; we use WordNet only on descriptions and titles. In this case, the word "vanity" is assigned the furniture sense.

For the Corel database where each image has 4-5 keywords, our strategy assigns the correct sense to almost all keywords. Disambiguation is more difficult for the museum data. For example, even though "doctor" and "hospital" are in the same concept, they have no common hypernym synsets in WordNet and if there are no other words helping for disambiguation it may not be possible to obtain the correct sense.

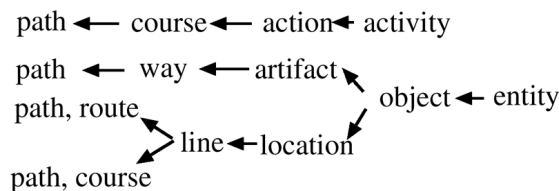


Figure 2: Four possible senses of the word "path"

web number: 4359202410830012	Description: serving woman stands in a dressing room, in front of vanity with chair, mirror and mantle, holding a tray with tea and toast.
rec number: 2	
Artist: Tissot	
Primary Class: print	Display date: 1886
Country: France	

Figure 3: a typical record associated with an image in the Fine Arts Museum of San Francisco collection.

## 5. QUALITY OF CLUSTERS

We applied our method to 8405 museum images, with an additional 1504 used as held out data for the annotation experiments. The augmented vocabulary for this data had 3319 words (2439 were from the associated text, and the remainder were from WordNet). We used a 5 level quad tree giving 256 clusters. Sample clusters are shown in Figure 5. These were generated using Model I. Using Model II to fit the data yielded clusters which were qualitatively at least as coherent.

Our primary goal in this work is to expose structure in a collection of image information. Ideally, this structure would be used to support browsing. An important goal is that users can quickly build an internal model of the collection,

so that they know what kind of images can be expected in the collection, where to look for them. It is difficult to tell directly whether this goal is met.

However, we can obtain some useful indirect information. In a good structure, clusters would “make sense” to the user. If the user finds the clusters coherent, then they can begin to internalize the kind of structure they represent. Furthermore, a small portion of the cluster can be used to represent the whole, and will accurately suggest the kinds of pictures that will be found by exploring that cluster further.

In <sup>2</sup> clusters were verified to have coherence by having a subject identify random clusters versus actual clusters. This was possible at roughly 95% accuracy. This is a fairly basic test; in fact, we want clusters to “make sense” to human observers. To test this property, we showed 16 clusters to a total of 15 naïve human observers, who were instructed to write down a small number of words that captured the sense of the cluster for each of these clusters. Observers did not discuss the task or the clusters with one another. The raw words appear coherent, but a better test is possible. For each cluster, we took all words used by the observers, and scored these words with the number of WordNet hypernyms they had in common with other words (so if one observer used “horse”, and another “pony”, the score would reflect this coherence). Words with large scores tend to suggest that clusters are “make sense” to viewers. Most of our clusters had words with scores of eight or more, meaning that over half our observers used a word with similar semantics in describing the cluster. In figure 4, we show a histogram of these scores for all sixteen clusters; clearly, these observers tend to agree quite strongly on what the clusters are “about”.

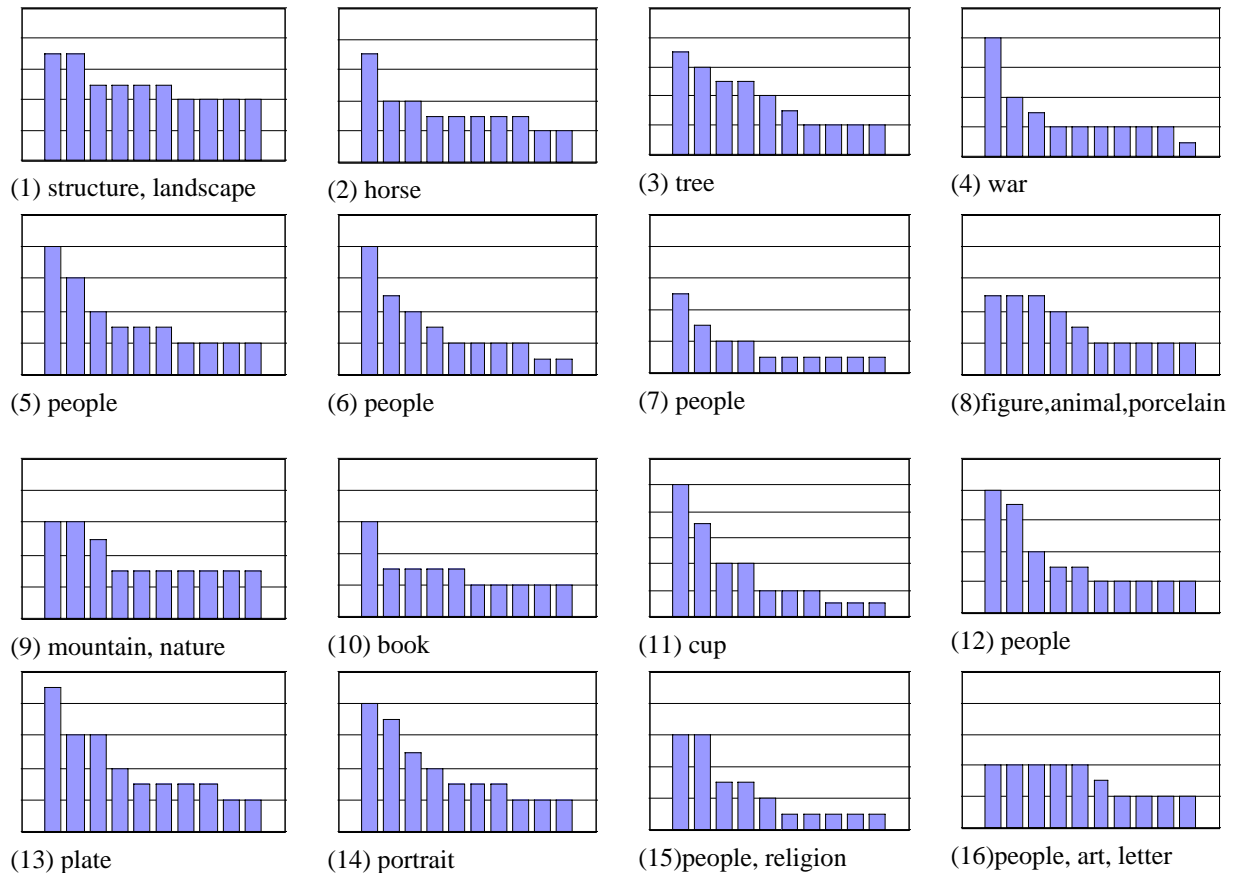


Figure 4. Each histogram corresponds to a cluster and shows the score (described in the text) for the 10 words with highest score used to describe that cluster by human observer in that cluster. The scales for the histograms are the same, and go in steps of 2; note that most clusters have words with scores of eight or above, meaning that about half of our 15 observers used that or word with similar semantics to describe the cluster. The number of total words for each cluster varies between 15-35.

## 6. APPLICATIONS

### 6.1. Browsing

Most image retrieval systems do not support browsing, likely because it is difficult to define and implement. Rather these systems force the user to specify what they are looking for with a query. This does not help the user learn what kind of images can be found. Setting up image databases so that their content is easy to internalize and thus navigate is difficult, and normally involves much human input. Our clustering method can achieve this because coherent clusters can be represented by a single image. This can suggest to the user that images with similar semantics, appearance, or both, are available, possibly via a mouse click, without taking up too much space on a display (see Figure 6).

### 6.2. Search

A second important facility for image databases is retrieval based on user queries. We wish to support queries based on text, image features, or both. We also would like the queries to be soft in the sense that the combinations of items is taken into consideration, but documents which do not have a given item should still be considered. Finally, we would like the queries to be easily specified without reference to images already found—in other words, we do not want to rely on “query by example” where the search is based on finding images similar to an exemplar image.

Our approach to searching is to compute the probability of each candidate image of emitting the query items. Defining search by the computation of probabilities very naturally yields an appealing soft query system. Given a set of query items,  $Q$ , and a candidate document  $d$ , we can express the probability that the document produces the query using model I by:

$$P(Q|D, d) = \frac{P(Q, D)}{P(D)} = \sum_c P(Q|c, d) \frac{P(D|c, d)P(c)}{P(D)} = \sum_c \left\{ \prod_{i \in Q} \left( \sum_l P(i|l, c)P(l|c, d) \right) P(c|D, d) \right\} \quad (3)$$

and using model II by:

$$P(Q|D) = \sum_c \left\{ \prod_{i \in Q} \left( \sum_l P(i|l, c)P(l|c) \right) P(c|D) \right\} \quad (4)$$

Documents with a high score are then returned to the user. A second approach is to find the probabilities that the query is generated by each cluster, and then sample from the clusters, weighted by the probability that the cluster emits the query. This often works reasonably well because cluster membership plays the dominant role in generating the documents, which simply reflects the fact that the clusters are coherent. Nonetheless, we have found that the results using (3) are sometimes significantly better, and rarely worse.

We have implemented search for arbitrary combinations of words and image features. For the purposes of experimentation we specify features simply by selecting segments from our available data, and using the features of the selected segments. Clearly providing a more flexible method of specifying image features is an important next step. Part of such a system could employ a user’s selection from features suggested by a returned result set. This is as explored in the many “query by example” image retrieval systems. However, this is only one strategy, and we emphasize that our system can retrieve images without starting from an example image. This captures the users needs in some ways, but not in others. Put differently, we can query for a dog by the word “dog”, and if we want blue sky above the dog, then we can add the appropriate segment feature to the query. Working with pieces of other images is not required.

### 6.3. Auto-illustration

An extreme example of such search is auto-illustration, where the database is queried based on, for example, a paragraph of text. We tried this on text passages from some classic literary works available on the web free of copyright problems. One result is shown in Figure 7.

### 6.4. Auto-annotation

The previous two applications take advantage of the fact that model I is especially good at characterizing the data. However, some of what is learnt is applicable to images outside of the training set. For example, if there are a number of images of tigers in the training data, then the system can learn that an image consisting of an orange stripy region surrounded by green has a significantly higher probability than chance of being associated with the word “tiger”. Thus,

the statistical model can be used to predict words based on image segments. Specifically, if  $B$  is the set of segments in the image under consideration, then

$$P(w | B, d) \propto P(B, w | d) = \sum_c P(w | c) P(B | c, d) P(c) \quad (5)$$

In the case of Model I completing the above calculation is not completely straightforward because the aspect model is properly expressed in terms in of the documents of the training set, and we are now considering a new document. The unknown quantity is the distribution over the vertical nodes for the new document. We have experimented with 4 strategies to deal with this problem: 1) Use an average value for the vertical weights (much like Model II); 2) Fit the weights based on the observed segments,  $B$ , (somewhat expensive), 3) Fit the weights based on both the overall segments and the words (very expensive), 4) Marginalize over  $d$  from the training set (very expensive). Interestingly, in our (somewhat limited) testing done so far, the methods are hard to distinguish, with the inexpensive approach, 1, doing essentially as well as the others, despite its simplicity. Figure 8 shows some annotation examples.

## 7. DISCUSSION

Both text and image features are important in the clustering process. For example, in the cluster of human figures on the top left of figure 5, the fact that most elements contain people is attributable to text, but the fact that most are vertical is attributable to image features; similarly, the cluster of pottery on the bottom left exhibits a degree of coherence in its decoration (due to the image features; there are other clusters where the decoration is more geometric) and the fact that it is pottery (ditto text). Furthermore, by using both text and image features we obtain a joint probability model linking words and images, which can be used both to suggest images for blocks of text, and to annotate images. Our clustering process is remarkably successful for a very large collection of very diverse images and free text annotations. This is probably because the text associated with images typically emphasizes properties that are very hard to determine with computer vision techniques, but omits the “visually obvious”, and so the text and the images are complementary.

We mention some of many loose ends. Firstly, the topology of our model is too rigid, and it would be pleasing to have a method that could search topologies. Secondly, it is still hard to demonstrate that the hierarchy of clusters represents a semantic *hierarchy*. Our current strategy of illustrating (resp. annotating) by regarding text (resp. images) as conjunctive queries of words (resp. blobs) is clearly sub-optimal, as the elements of the conjunction may be internally contradictory; a better model is to think in terms of robust fitting. Our system produces a joint probability distribution linking image features and words. As a result, we can use images to predict words, and words to predict images. The quality of these predictions is affected by (a) the mutual information between image features and words under the model chosen and (b) the deviance between the fit obtained with the data set, and the best fit. We do not currently have good estimates of these parameters. Finally, it would be pleasing to use mutual information criteria to prune the clustering model.

Annotation should be seen as a form of object recognition. In particular, a joint probability distribution for images and words is a device for object recognition. The mutual information between the image data and the words gives a measure of the performance of this device. Our work suggests that unsupervised learning may be a viable strategy for learning to recognize very large collections of objects.

## ACKNOWLEDGEMENTS

This project is part of the Digital Libraries Initiative sponsored by NSF and many others. Kobus Barnard also receives funding from NSERC (Canada), and Pinar Duygulu is funded by TUBITAK (Turkey).

## REFERENCES

1. T. Hofmann, Learning and representing topic. A hierarchical mixture model for word occurrence in document databases, *Workshop on learning from text and the web* (1998).
2. K. Barnard and D. Forsyth, Learning the Semantics of Words and Pictures, *International Conference on Computer Vision*, II:408-415 (2001).

3. D. A. Forsyth, Computer Vision Tools for Finding Images and Video Sequences, *Library Trends*, **48**, 326-355 (1999).
4. C. Carson, S. Belongie, H. Greenspan, and J. Malik, Blobworld: Image segmentation using Expectation-Maximization and its application to image querying, submitted for publication to IEEE Transactions on Pattern Analysis and Machine Intelligence. Available in the interim from <http://HTTP.CS.Berkeley.EDU/~carson/papers/pami.html>.
5. C. Frankel, M. J. Swain, and V. Athitsos, Webseer: An Image Search Engine for the World Wide Web, U. Chicago TR-96-14, available from (1996).
6. M. La Cascia, S. Sethi, and S. Sclaroff, Combining Textual and Visual Cues for Content-based Image Retrieval on the World Wide Web, *IEEE Workshop on Content-Based Access of Image and Video Libraries* (1998).
7. F. Chen, U. Gargi, L. Niles, and H. Schütze, Multi-modal browsing of images in web documents, *SPIE Document Recognition and Retrieval* (1999).
8. P. G. B. Enser, Query analysis in a visual information retrieval context, *Journal of Document and Text Management*, **1**, 25-39 (1993).
9. P. G. B. Enser, Progress in documentation pictorial information retrieval, *Journal of Documentation*, **51**, 126-170 (1995).
10. L. H. Armitage and P. G. B. Enser, Analysis of user need in image archives, *Journal of Information Science*, **23**, 287-299 (1997).
11. R. Srihari, Extracting Visual Information from Text: Using Captions to Label Human Faces in Newspaper Photographs, SUNY at Buffalo, Ph.D., (1991).
12. V. Govindaraju, A Computational Theory for Locating Human Faces in Photographs, SUNY at Buffalo, Ph.D., (1992).
13. R. K. Srihari, R. Chopra, D. Burhans, M. Venkataraman, and V. Govindaraju, Use of Collateral Text in Image Interpretation, *ARPA Image Understanding Workshop* (1994).
14. R. K. Srihari and D. T. Burhans, Visual Semantics: Extracting Visual Information from Text Accompanying Pictures, *AAAI '94* (1994).
15. R. Chopra and R. K. Srihari, Control Structures for Incorporating Picture-Specific Context in Image Interpretation, *IJCAI '95* (1995).
16. T. Hofmann and J. Puzicha, Statistical models for co-occurrence data, Massachusetts Institute of Technology, A.I. Memo 1635, available from (1998).
17. K. Barnard, P. Duygulu, and D. Forsyth, Clustering Art, *Conference on Computer Vision and Pattern Recognition* (2001).
18. D. Blei, A. Ng, and M. Jordan, Dirichlet Allocation Models, *NIPS* (2001).
19. A. P. Dempster, N. M. Laird, and D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society. Series B (Methodological)*, **39**, 1-38 (1977).
20. J. Shi and J. Malik., Normalized Cuts and Image Segmentation, *IEEE Trans. Patt. Anal. Mach. Intell.*, **22**, 888-905 (2000).
21. Available from <http://dlp.CS.Berkeley.EDU/~doron/software/ncuts/>
22. G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, Introduction to WordNet: an on-line lexical database, *International Journal of Lexicography*, **3**, 235 - 244 (1990).
23. D. Yarowski, Unsupervised Word Sense Disambiguation Rivaling Supervised Methods, *33rd Conference on Applied Natural Language Processing*, ACL (1995).
24. R. Mihalcea and D. Moldovan., Word sense disambiguation based on semantic density, *COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems* (1998).
25. E. Agirre and G. Rigau, A proposal for word sense disambiguation using conceptual distance, *1st International Conference on Recent Advances in Natural Language Processing* (1995).
26. W. Gale, K. Church, and D. Yarowski, One Sense Per Discourse, *DARPA Workshop on Speech and Natural Language*, 233-237 (1992).
27. E. Brill, A simple rule-based part of speech tagger, *Third Conference on Applied Natural Language Processing*, ACL (1992).



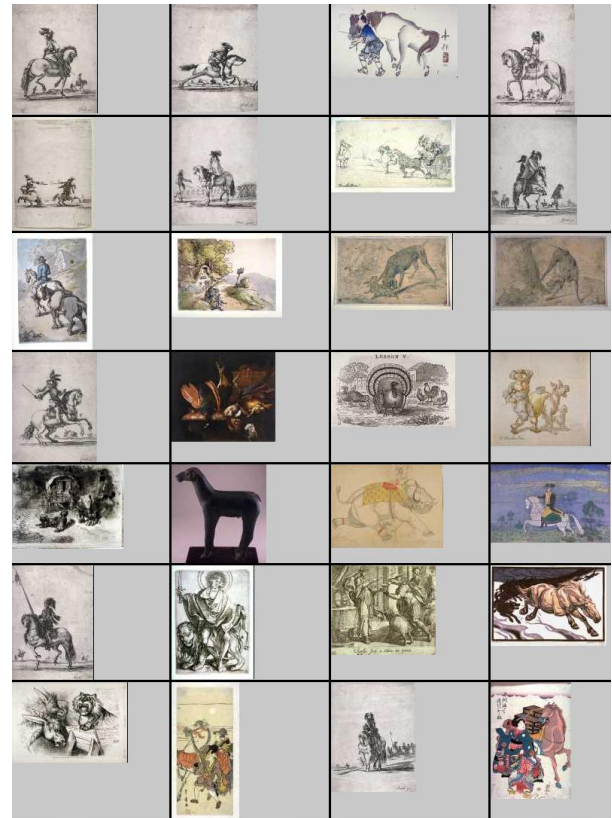
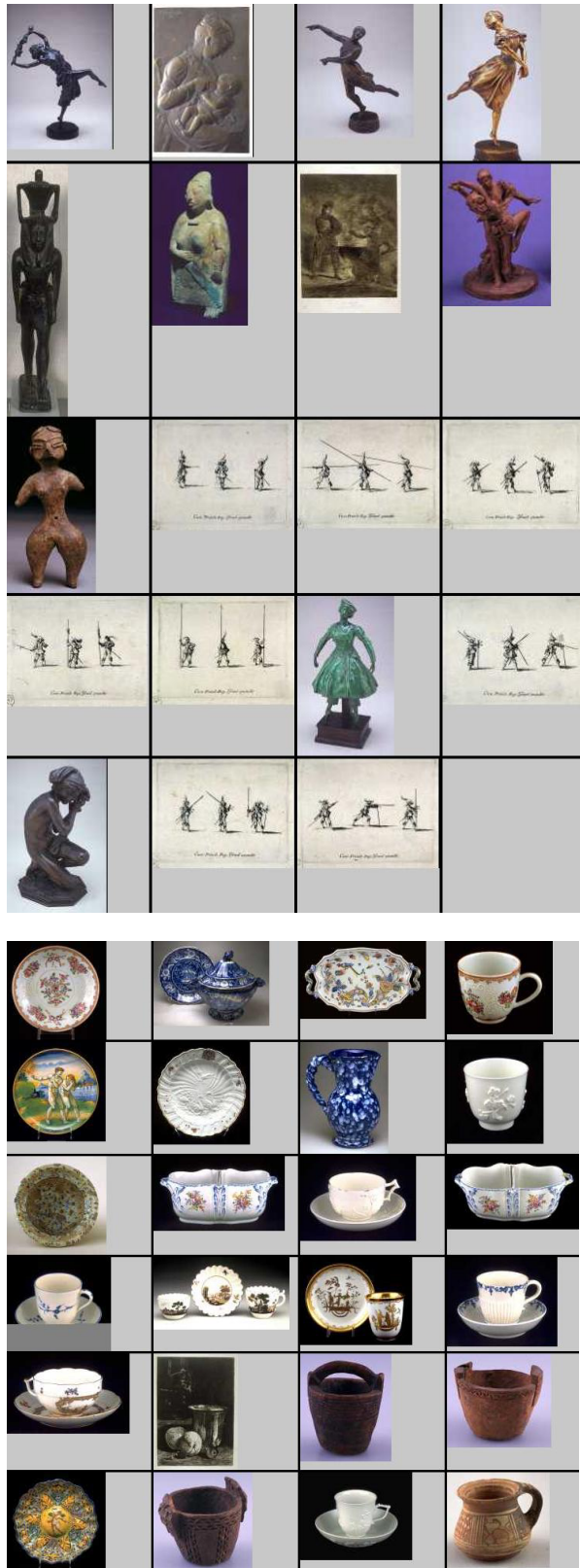


Figure 5. Some sample clusters from the museum data. The theme of the upper left cluster is clearly female figurines, the upper right contains a variety of horse images, and the lower left is a sampling of the ceramics collection. Some clusters are less perfect, as illustrated by the lower right cluster where a variety of images are blended with seven images of fruit.

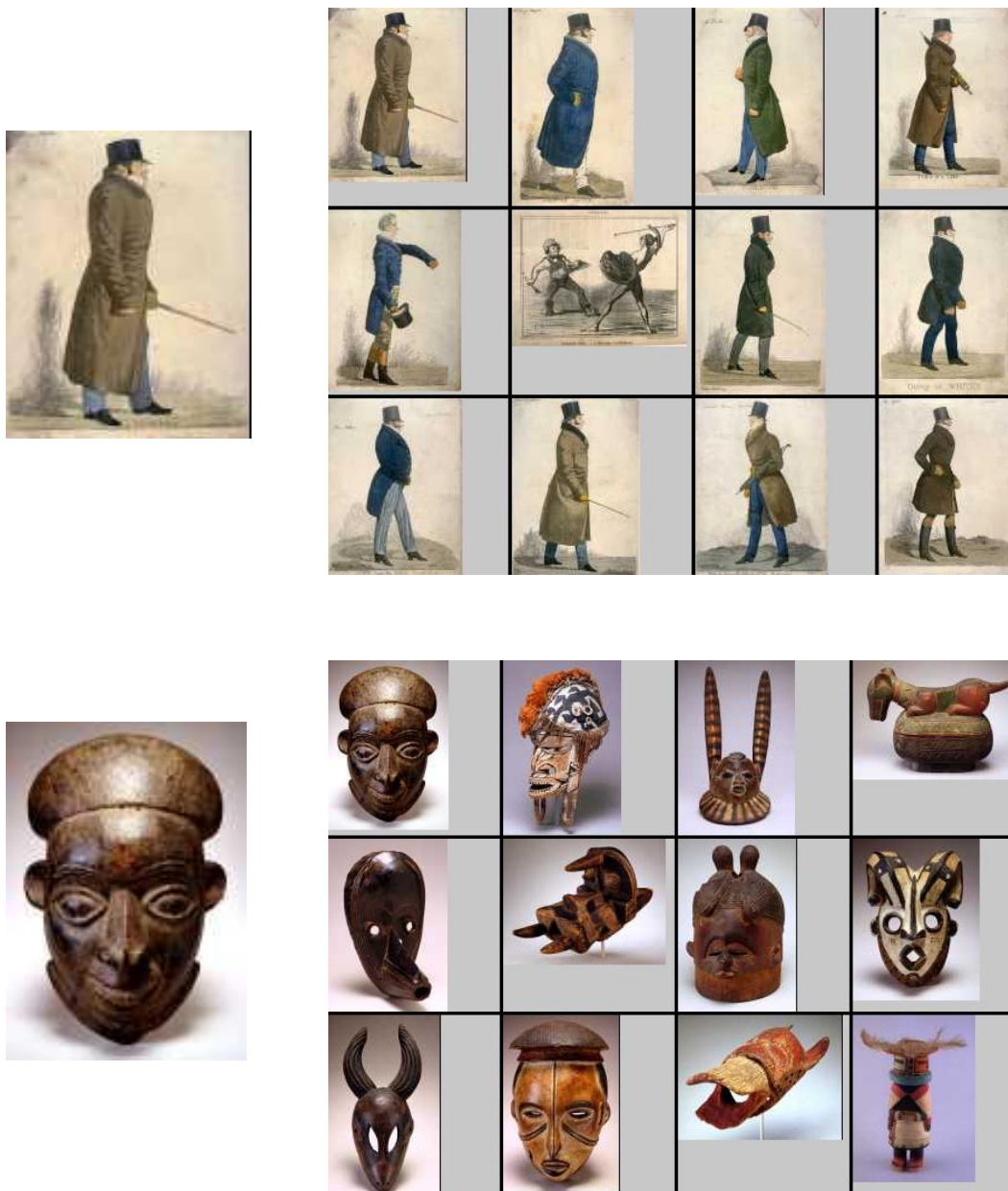


Figure 6. The clusters on the right can be adequately represented by the images on the left (only 12 images—roughly a third of the images in the two cluster are shown). In a browsing application, the single image quickly tells the user whether they would like to further explore that part of the collection.



	<p>“The large importance attached to the harpooneer's vocation is evinced by the fact, that originally in the old Dutch Fishery, two centuries and more ago, the command of a whale-ship was not wholly lodged in the person now called the captain, but was divided between him and an officer called the Specksynder. Literally this word means Fat-Cutter; usage, however, in time made it equivalent to Chief Harpooneer. In those days, the captain's authority was restricted to the navigation and general management of the vessel; while over the whale-hunting department and all its concerns, the Specksynder or Chief Harpooneer reigned supreme. In the British Greenland Fishery, under the corrupted title of Specksioneer, this old Dutch official is still retained, but his former dignity is ...”</p>
<p>large importance attached fact old dutch century more command whale ship was per son was divided officer word means fat cutter time made days was general vessel whale hunting concern british title old dutch official present rank such more good american officer boat night watch ground command ship deck grand political sea men mast way professional superior</p>	

Figure 7. Examples of auto-illustration using a passage from Moby Dick , half of which is reproduced to the right of the images. Below are the words extracted from the passage used as a conjunctive probabilistic query.







		<p>Associated Words KUSATSU SERIES STATION TOKAIDO TOKAIDO GOJUSANTSUGI PRINT HIROSHIGE Predicted Words (rank order) tokaido print hiroshige object artifact series ordering gojusantsugi station facility arrangement minakuchi sakanoshita maisaka</p>
		<p>Associated Words SYNTAX LORD PRINT ROWLANDSON Predicted Words (rank order) rowlandson print drawing life_form person object artifact expert art creation animal graphic_art painting structure view</p>
		<p>Associated Words DRAWING ROCKY SEA SHORE Predicted Words (rank order) print hokusai kunisada object artifact huge process natural_process district administrative_district state_capital rises</p>

Figure 8. Some annotation results showing the original image, the N-Cuts segmentation, the associated words, and the predicted words in rank order. The test images were not in the training set. Keywords in upper-case are in the vocabulary. The first two examples are excellent, and the third one is a typical failure. Some of the words make sense given the segments, but the semantics are incorrect.