Recognition as Translating Images into Text

Kobus Barnard[a] , Pinar Duygulu[b], and David Forsyth[c]

[a]Department of Computer Science, University of Arizona
kobus@cs.arizona.edu

[b]Computer Engineering Department, Middle East Technical University, Turkey
duygulu@ceng.metu.edu.tr

[c]Computer Division, University of California, Berkeley
daf@cs.berkeley.edu

ABSTRACT

We present an overview of a new paradigm for tackling long standing computer vision problems. Specifically our approach is to build statistical models which translate from a visual representations (images) to semantic ones (associated text). As providing optimal text for training is difficult at best, we propose working with whatever associated text is available in large quantities. Examples include large image collections with keywords, museum image collections with descriptive text, news photos, and images on the web.

In this paper we discuss how the translation approach can give a handle on difficult questions such as: What counts as an object? Which objects are easy to recognize and which are hard? Which objects are indistinguishable using our features? How to integrate low level vision processes such as feature based segmentation, with high level processes such as grouping. We also summarize some of the models proposed for translating from visual information to text, and some of the methods used to evaluate their performance.

Keywords: Object recognition, machine translation, learning image semantics, hierarchical clustering, aspect model

1. INTRODUCTION

What is vision? One answer provided in the first page of David Marr's book, "Vision", is that it is a *process* of going from visual information to a representation of what is "present in the world and where it is" [1]. We paraphrase this even more to say that it is a process of translating from visual information to semantic information. The interesting thing about emphasizing this view is that it is amenable to attack by computer programs, provided that we accept language as a semantic representation. The key is to exploit the large quantities of images with associated text which is now available as training data.

In addition to shedding some light on the vision problem, we further argue that utility lies in going from one representation to the another. For example, going from an image of a tiger in the grass, to an action (run away) can be useful to a prey species. For modern humans, going from that image to the phrase " tiger in the grass" can be exploited in image retrieval and browsing applications. A preliminary analysis of the role of image semantics as represented by language in content based image retrieval is available[2], and a few innovative systems have been built or proposed which can use text to improve image retrieval[3-5 6,7].

In this paper we give an overview of the translation paradigm for computer vision. The approach is to learn statistical translation models from large data sets of images with associated text. Currently we segment the images into regions and represent each region by a collection of its features. As discussed in depth elsewhere [8,9], the words may also be preprocessed for better suitability for the task. Learning translation models for the relationships between the image regions and the words suggests that computer vision can be viewed as a form of data mining. Explicitly, if our data contains many images with orange stripy regions on a variety of backgrounds with "tiger" among associated words, then the data set may very well implicitly contain the information that orange stripy regions should be statistically linked with the word "tiger". Even more explicitly, consider two images: An orange stripy region surrounded by blue, with associated words "tiger, water", and a similar region surrounded by green with associated words "tiger grass". The fact that the orange stripy region and the word "tiger" are the only constants, is a strong hint that these should be linked. Of

course, learning a translation model on a real data set involves dealing with many of forms of noise including incomplete, erroneous, and irrelevant words, as well as segmentation errors. This reality is one of the reasons why a statistical approach is most suitable.

In our approach, we do not need to specify in advance what is to be recognized. Some objects will not be identifiable/recognizable given our features and these ambiguities are identified. This provides a platform to propose additional features or required data (either by consulting a search engine, or human input). Furthermore we learn to label image regions without labeled training data. This is significant because labeled training data is expensive to obtain in quantity. This is in contrast with text loosely associated with images which is now available in substantial quantities. Examples include the Corel image collection (4-5 keywords for 40,000 images), a data set that we have been studying courtesy of the Fine Arts Museum of San Francisco (meta data as well as descriptive paragraphs for many of 83,000 images), news photos with captions (of the order of 1,000 a day available on line), and, of course, the web at large, with millions of images which have associated text, a workable part of which can be extracted using a variety of heuristics [3,10].

1.1 Practical Applications

Very large collections of images are widespread. As we see below (§1.2), there is extensive evidence from the user studies literature that users would like to search for images on the level of (at least) object semantics. This is difficult to do with current computer vision methods; linking image information with text annotations might improve such searches. There are numerous other practical applications for methods that can link text and images, however imperfectly:

Automated image annotation: Numerous organizations manage collections of images for internal use. A typical workflow is described by the work of Markkula and Sormunen [11] who studied the image archive of a Finnish newspaper (see also Enser's work [12-14] on various image archives, which use roughly the same procedure). Archivists receive pictures and annotate them with words that are likely to be useful keys for retrieving the pictures; journalists then search the collection using these keywords. Annotation is often difficult and uncertain; it would be attractive to have a procedure that annotated images automatically. One might auto-annotate by predicting words with high posterior probability given an image. Examples of automated annotation appear in [9,15] and below.

Browsing support: Museums release parts of their collections onto the web to attract visitors by giving them a sense of what they would see if they visited. Typically users who know a collection well wish to search it, and those who don't, prefer to browse [16]. This means it would be attractive to organize the collection in a way that made sense to visitors, and so supported browsing. Collecting together images that looked similar and were similarly annotated would be a good start. Fitting a probability model with an appropriate structure yields quite useful clusters, as described in [9].

Auto-illustrate: Commercial image collections can't supply an attractive service to a casual user, because searching the collection is typically difficult and expensive. A tool that could automatically suggest images to illustrate blocks of text might expose value in the collection by making it possible for casual users to get reasonable results cheaply. Auto-illustration is possible if one can obtain images with high probability given text ([9,15]).

1.2 How people use image collections

A broad range of computer vision methods have been used to search collections of images. Typically, images are matched based on features computed from the entire image or from image regions. The literature is too broad to review here; there are reviews in [17,18]. With the exception of systems that can identify faces [19], naked people [20], pedestrians [21] or cars [19], matching is not usually directed toward object semantics.

Typically, users query images on semantics [22] [23] [2]. Recent work of Enser's [12-14] deals with the disparity between user needs and what technology supplies. The paper makes hair-raising reading; for example, he cites a request to a stock photo library for

> "Pretty girl doing something active, sporty in a summery setting, beach - not wearing lycra, exercise clothes - more relaxed in tee-shirt. Feature is about deodorant so girl should look active - not sweaty but happy, healthy, carefree - nothing too posed or set up - nice and natural looking".

Other user studies include the work of [24], who studied practice at a manually operated newspaper photo archive and Markkula and Sormunen [11] who study practice at a Finnish newspaper's digital photo archive. Keister studied requests received by the National Library of Medicine's Archive [25].

In the user studies literature, authors break out the semantics of the images requested in different ways, but from our perspective the important points are:

- that users request images both by object kinds (i.e. a princess) and identities (i.e. the princess of Wales);
- that users request images both by what they depict (i.e. things visible in the picture) and by what they are about (i.e. concepts evoked by what is visible in the picture);
- that queries based on image histograms, texture, overall appearance, etc. are vanishingly uncommon;

and that text associated with images is extremely useful in practice—for example, newspaper archivists index largely on captions [11].

1.3 Representing collections using images and words

Combining text and images is currently uncommon. A few systems combine text and image data. Search using a simple conjunction of keywords and image region features is provided in Blobworld [6]. Webseer [26] 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 [3]. Others have also experimented with using image features as part of a query refinement process [7,27]. Srihari and others have used text information to disambiguate image features, particularly in face finding applications [4,5,28]. In [29,30], Maron et al. study automatic annotation of images, but work one word at a time, and offer no method of finding the correspondence between words and regions. Finally, perhaps closest to our work on predicting words for regions is the work of Mori et al. [31], where co-occurrence statistics are collected for words and image areas defined by a fixed grid.

1.4 Annotation, correspondence and recognition

Predicting images using text is conceptually straightforward, if difficult in practice. Predicting text using images is not, because there are two possible tasks one could attack. Firstly, one might attempt to predict annotations of entire images using all information present. We refer to this task as **annotation**. Secondly, one might attempt to associate particular words with particular image substructures—that is, to infer **correspondence**.

Correspondence is a peculiar feature of object recognition. Current theories of object recognition reason either in terms of geometric correspondence and pose consistency; in terms of template matching via classifiers; or by correspondence search to establish the presence of suggestive relations between templates. A detailed review of these strategies appears in [17]. These types of theory are at the wrong scale to address core issues: in particular, **what counts as an object?** (usually addressed by choosing by hand objects that can be recognized using the strategy propounded); **which objects are easy to recognize and which are hard?** (not usually addressed explicitly); and **which objects are indistinguishable using our features?** (current theories typically cannot predict the equivalence relation imposed on objects by the use of a particular set of features).

If we view recognition as a statistical process that attaches words to image regions, then these problems are amenable to attack. In this model, we can attack: **what counts as an object?** by saying that all words (or all nouns, etc.) count as objects; **which objects are easy to recognize?** by saying that words that can be reliably attached to image regions are easy to recognize and those that cannot, are not; and **which objects are indistinguishable using our features?** by finding words that are predicted with about the same posterior probability given any image group—such objects are indistinguishable given the current feature set. if one could predict these annotations, one could save considerable work). While none of these questions are easy or resolved in this framework, it does offer a way to talk about them.

Typical current applications of machine learning in object recognition involve a high degree of supervisory input (e.g. the examples of finding faces, people, cars or digits described above and the summary in [17]). Manually identifying every object in a large training set becomes very difficult to do when the object vocabulary is large. The analogy

between learning a correspondence model that can associate words with image regions and learning a lexicon, suggests it is possible to build a process that uses rather less supervisory input. In effect, one builds a model using unsupervised methods, marks up the model's output, and refits. This is a standard process in the machine translation literature (a good guide is Melamed's thesis [32]; see also [33,34]).

2. INPUT REPRESENTATION AND PRE-PROCESSING

Each image is segmented using normalized cuts[35]. This segmenter shares with most others the occasional tendency to produce small, typically unstable regions. We represent the 8 largest regions in each image by computing, for each region, a set of 40 features. The features represent, rather roughly, major visual properties:

- **Size** is represented by the portion of the image covered by the region
- **Position** is represented using the coordinates of the region center of mass normalized by the image dimensions
- **Color** is (redundantly) represented using the average and standard deviation of (R,G,B), (L,a,b) and (r=R/(R+G+B), g=G/ (R+G+B)) over the region.
- **Texture** is represented using the average and variance of 16 filter responses. We use 4 difference of Gaussian filters with different sigmas, and 12 oriented filters, aligned in 30 degree increments.
- **Shape** is represented by the ratio of the area to the perimeter squared, the moment of inertia (about the center of mass), and the ratio of the region area to that of its convex hull.

3. MODEL OVERVIEW

A number of models for serving the annotation and recognition tasks have been developed [31 9,15,36]. Here we give a condensed overview, focusing on translation models. Details and the results of comprehensive tests are available elsewhere[36].

3.1 Discrete data translation

We begin with a model inspired by statistical machine translation. In machine translation a lexicon links discrete objects (words in one language) to discrete objects (words in the other language). We must come up with a lexicon given an aligned bitext, which consists of many small blocks of text in both languages, which are known to correspond in meaning. A traditional example is Hansard for the Canadian parliament, where each speaker's remarks in French and in English correspond in meaning. Assuming an unknown one-one correspondence between words, coming up with a joint probability distribution linking words in the two languages is a straightforward missing data problem (Brown et al. [37]). It is straightforward to create analogous image data. We use K-means to vector-quantize the set of features representing an image region. Each region then gets a single label (blob token).

We now have an aligned bitext consisting of the blobs and the words for each image. We must construct a joint probability table linking word tokens (the abstract model of a word, as opposed to an instance) to blob tokens. In the current work, we use all keywords associated with each image. Because the data set does not provide explicit correspondences, we have a missing data problem which is easily dealt with as an application of EM [38]. We refer to this method as **discrete-translation.** The details of the algorithm are provided in Duygulu et al[39]. In our implementation we follow Brown et al.[37] and initialize the model with the co-occurrence data. This initialization point is essentially the algorithm proposed by Mori et al.[31] in the case of image parts being simple blocks rather than regions.

3.2 Correspondence from a hierarchical clustering model

The hierarchical combination of clustering and aspect models[40,41] modified for learning the joint probability of words and blobs [15] can be used both for image annotation and for region labeling. This is because the vertical structure of the hierarchical model (aspects) emphasizes that documents (images) are composed of parts. The nodes of the hierarchy are used to represent the joint probability of image regions and associated text, with images in a given cluster being considered to be collections of blobs and words generated by independent draws from the same set of nodes. The node parameters learnt during model fitting can then be used to predict words for new image regions using probabilistic

inference. However, the independent emission of regions and words makes the model less than ideal for recognition, and modifications have been proposed to tighten the relationship between the two[36].

The first approach is to emit words conditioned on the blobs. We label variants of this model with the prefix "dependent". The second approach further tightens the relationship and insists that words and blobs are emitted as pairs. To deal with the fact that the number of words and blobs is typically different, we either leave some unassigned, or we duplicate the emission of some of the words or blobs as need be. This method requires that correspondence between the blobs and words be estimated during training, which is done using graph matching[42].

There are two reasons for considering these more complex approaches as alternatives to the discrete translation model in the previous section. The first advantage is that the learning of the blob feature characterizations is integrated into the training instead of quantizing before learning the translation model. Currently we use Gaussian distributions with diagonal covariance matrices for the features, and the parameters of these are learnt during model training along with the other parameters.

The second advantage is that image clusters can provide context for translation. For example, consider a grey patch. It could be translated into "pavement" or "mud". If the images cluster nicely into the appropriate clusters based on all the blobs taken as a group, then cluster membership can be used to disambiguate certain blobs. Performance naturally depends on whether the training data is such that the clusters make sense in the testing world. We hasten to add that our implementation of all the hierarchical models is such that a tree with no branches is allowed. In this case, there is only one cluster and documents are all generated from the same pool of "aspects" (following the terminology of Hofmann et al. [40,41]). In this case the dependent model, the result is a model which is much like the discrete-trans model in the previous section, except of course that the features are modeled using a Gaussian distribution for each node, instead of being discretized.

3.3 Correspondence models, NULL, fertility and refuse to predict

Correspondence comes with a variety of annoying difficulties which we have skated over above. The primary issue is the choice of correspondence model. Should there be a one-one map between regions and words (usually impossible, because of the numbers are different)? In the work described here, we require regions to be linked to words; there is no option of deciding that a region corresponds to no word. This forces models of image regions corresponding to particular words to cope with a large pool of outliers. This problem could, in principle, be handled by appending a special word, NULL, to the text of each data item and a special image region, NULL, to the image regions of each data item. This is a traditional solution in the machine translation literature; the tendency of single words in some languages to generate more than one word in others (a property referred to as "fertility") can be modeled explicitly in this framework [32,37]. In our limited experience, such models are not easy to fit to our datasets, because of a tendency to fit a model where every word is generated by the NULL image regions and every image region generates the NULL word. This is clearly a matter to be resolved by a prior model of deletion of words or image regions, respectively. One complication is that the probability that an annotation is absent is not independent of the annotation—annotators always mention "tigers", but only sometimes mention "people".

A crude strategy that offers some benefits of directly modeling NULL words is to refuse to predict an annotation when the annotation with the highest probability given the region has too low a probability; this discourages predictions by regions whose identity is moot. This is crude, because it doesn't mitigate the effect of all the outliers in the fitting process.

4. EVALUATION METHODS

There are two ways our models could be used for annotation. In the first, a model is used to annotate images drawn from a collection well represented by the original training data (for example, in an application where the model must annotate images arriving at an archive). In this case, we would like the model to approximate the joint probability of words and images well. In the second, we will use the model to predict words for images from a collection not well represented by the original training data; we might reasonably expect object recognition to be a case like this. In this case, we care mainly about the conditional probability of words given images.

Correspondence models present further difficulties. The issue now is how will we predict appropriate words for each particular region. Typically, the only way to obtain an accurate answer to this question is to look at the picture. This form of manual evaluation is very difficult to do for a satisfactory number of images. A less strict, but nonetheless informative, test is to determine the annotation performance for a correspondence model, on the grounds that poor annotation performance implies poor correspondence performance (crucially, the contrapositive is not necessarily true).

4.1 Measuring Annotation Performance

We can measure annotation performance by comparing the words predicted by various models with words actually present for held-out data. In most data sets, including ours, image annotations typically omit some obviously appropriate words. However, since our purpose is to compare methods this is not a significant problem as each model must cope with the same set of missing annotations. Performance comparisons can be carried out automatically and therefore on a substantial scale. We express prediction performance relative to predictions obtained using the empirical word frequency of the training set. Matching the performance empirical density is required to demonstrate non-trivial learning. Doing substantially better than this on the Corel data is difficult. The annotators typically provide several common words (e.g. “sky”, “water”. “people”), and fewer less common words (e.g. “tiger”). This means that annotating all images with, say, “sky”, “water”, and “people” is quite a successful strategy. Performance using the empirical word frequency would be reduced if the empirical density was flatter. Thus for this data set, the increment of performance over the empirical density is a sensible indicator. In the paper detailing the models[36] we propose three different measures for word prediction performance: One based on the Kullback-Leibler (KL) divergence between the computed predictive distribution and an estimate of the target distribution, one based on a normalized version of subtracting incorrect predictions from correct ones, and one based on observing the number of correct predictions out of n guesses, where n is the number of words available in the test image annotation. Thus if there are three keywords, “sky”, “water”, and “sun”, then n=3, and we allow the models to predict 3 words for that image.

4.2 Measuring Correspondence Performance

Measuring the performance of methods that predict a specific correspondence between regions and words is difficult, because images must be checked by hand. This limits the size of the pool that can be used, and also means that measurements may contain significant noise (it is surprisingly difficult to establish, and stick to, an exact policy about what regions should carry, say, the label “people”). However, we can use a region based method for annotation by summing over the word posteriors for all the regions. Furthermore, we can reasonably expect that a method that cannot predict annotations accurately is unlikely to predict correspondence well. This means that annotation measures offer a plausible proxy.

When we use annotation as a proxy, we insist that the methods predict words for each region, even if there is a better inference method for annotation available for that model. However, when we compute region word posteriors using the image based annotation methods, the word posteriors do not necessarily sum to one because there is no requirement in these models that each region emits any word. We enforce this requirement by normalizing the posteriors for each region before summing them.

To corroborate the annotation proxy measure, we also score some correspondence results by hand. While this method directly looks at the correspondence, it does require human judgment. We hand labeled each region in a number of images with every appropriate word in the vocabulary. We insisted that the region has a plausible visual connection to the chosen words. Thus the word “ocean” for “coral” would be judged incorrectly because the ocean is transparent. Other difficulties include words like “landscape” and “valley” which normally apply to larger areas than our regions, and “pattern” which can arguably be designated as correct whenever it appears, but we scored it as incorrect because “pattern” recognition isn’t particularly helpful. Some regions could not be linked with any vocabulary term, and these regions were omitted from consideration in computing the scores. Producing the labeled dataset is clearly a time consuming and error prone process, and thus we are only able to use this ground truth for a modest number of images (50 images for each of ten test sets). With the hand labeled set, we are able to compute the same measures as for the image annotation case, although over a much smaller test set.

5. EXPERIMENTS

For our experiments we used images from 160 CD's from the Corel image data set. Each CD has 100 images on one relatively specific topic such as "aircraft". From the 160 CD's we drew samples of 80 CD's, and these sets were further divided up into training (75%) and "standard" held out (25%) sets. The images from the remaining CD's formed a more difficult "novel" held out set. Predicting words for these images is difficult, as we can only reasonably expect success on quite generic regions such as "sky" and "water"—everything else is noise.

Each such sample was given to each process under consideration, and the results of 10 of such samples were averaged. This controls for both the input data and EM initialization. Images were segmented using N-Cuts [35]. We excluded words which occurred less than 20 times in the test set, which yielded vocabularies of the order of 155 words. We used a modest selection of features for each segment, including size, position, color, oriented energy (12 filters), and a few simple shape features. For the discrete translation model, we used 500 clusters for vector quantization.

A fairly comprehensive set of results focusing on model comparison is already available [36]. Here we take the opportunity to provide a few additional results. We consider in more detail when the clustering context is being helpful given our data set and chosen topology. Thus we test versions of the hierarchical models for the independent aspect based model [15], both with a 511 node binary tree (labeled I-0 in[36] and independent-binary below), and with a vertical list of 500 nodes (labeled independent-vertical below). In this second configuration, this model is quite like the aspect model of Hofmann et al. [41], except, of course, the probability distributions for each node are joint distributions over both region features and words. Similarly, we test the dependent version of the model with binary tree topology (labeled D-0 in [36] and dependent-binary below), as well as without clusters (labeled dependent-vertical below) . Because we are interested in the implications for recognition, we predict the words using the image regions and summing the results, as opposed to using the image regions together. However, because we are interested in the effect of the cluster context, we use cluster information to bias the weights of the nodes based on cluster membership. Specifically, we use the "region-cluster" inference process (described in [36]).

Annotation results are provided in Table 1. Figure 1 shows region annotations for a few sample images. We labeled each region with the word with maximal probability, but it important to realize that the complete word posterior is computed and available to processes that can use it. In Table 2 we provide quantitative correspondence results computed over 50 images from each of the 10 held out sets.

These preliminary results indicate that we are not making as much use of the context as supplied by cluster membership as we would like. Despite a clear advantage of using cluster information when evaluating the training set, the results on the held out set are mixed, with a slight indication that we are better off not using the clusters. For the novel images, the results clearly show that the cluster information is a hindrance. Since these images are so different than the training images, it is perhaps not surprising that the clusters found in training do not make sense, but the degree of detriment was greater than expected.

Method	Training data	Held out data	Novel data
independent-vertical	0.129 (0.003)	0.109 (0.003)	0.056 (0.004)
independent -binary-tree	0.154 (0.003)	0.126 (0.003)	0.038 (0.004)
dependent-vertical	0.145 (0.004)	0.122 (0.004)	0.060 (0.005)
dependent-binary-tree	0.163 (0.004)	0.108 (0.004)	0.040 (0.007)
discrete-translation	0.122 (0.004)	0.073 (0.002)	0.028 (0.004)

Table 1. Representative annotation "proxy" evaluation of recognition results showing the effect of cluster context, as well as the effect of using models which simultaneously learn the feature distribution and the translation model, as compared to the discrete translation model. Word prediction is computed as the aggregate normalized prediction from the regions, even if a better strategy for annotation exists for the particular model. The models with binary tree topology are clustering models, and for these, cluster membership was used to weight the contributions of the nodes. The error measure the increase in performance over that of using the empirical word distribution (about 0.19). The specific word prediction measure is the ratio of correct words to available words in the test image annotation. There are on average about 3 words to predict, so a value of 0.11 corresponds to predicting about 0.9 of them (3 * (1.-, as opposed to under 0.6 with the empirical distribution. Predicting words for images from the novel CD's is very difficult, but all

methods consistently do a little better than the empirical distribution on this task. Errors (shown in parentheses) were estimated from the variance of the word prediction process over 10 different test sets, with at least 1000 samples in each set being averaged for the result for each set.

Method	Score
independent-vertical	0.108 (0.009)
independent -binary-tree	0.100 (0.009)
dependent-vertical	0.105 (0.009)
dependent-binary-tree	0.96 (0.009)
discrete-translation	0.91 (0.009)

Table 2. Correspondence performance as measured over 10 sets of 50 manually annotated images from the held out set. All values are relative to the performance using the empirical distribution (about 0.089).

8. DISCUSSION

We have provided an overview a variety of methods for predicting words from pictures. Each of these methods can predict some words rather well, and some can predict correspondence well for some words, too. There are practical applications for such methods. Furthermore, they offer an intriguing way to think about object recognition. A great deal remains to be done.

Currently we are working on a number of interesting and promising lines of attack. First, since we have developed a principled testing paradigm, we can use word prediction performance to evaluate vision tools such as segmentation algorithms and features. These results should be of general use since our task gets at the heart of vision. Second, we are working on using the region word posteriors to propose region merges, and help learn object shape. A typical low level image segmentation program cannot merge the black and white halves of a penguin, as they are as different in color as they can be. However, it the two regions have similar word posteriors, then a merge based on high level information can be proposed.

We currently have little information about the effect of supervision, but we expect that quite small supervisory input might lead to significant changes in the model. This is because missing correspondence information can generate symmetries in the incomplete data log-likelihood. For example, if “tiger” and “grass” always appear together, there is no way to determine which is which; but annotating a small number of images will break this symmetry, and could cause a substantial change in the model. A reasonable measure of performance of a model (and an associated fitting algorithm) is the quantity of supervisory input required to achieve a particular level of performance on some reference collection.

Large scale evaluation of correspondence models is genuinely difficult. The problem is important. In the not-too-distant future, there will be recognition systems that can manage vocabularies that are large enough that manual checking of labeled images is an unsatisfactory test. How can one tell how well such a system works? Our current strategy is to investigate methods that obtain extrapolated estimates of correspondence performance from proxies applied to test sets with carefully chosen properties. The key issue seems to be the entropy of the labels; if it is hard to predict the second word from the first word for each data item in the test collection, then annotation performance is likely to predict correspondence performance.

ACKNOWLEDGEMENTS

This project was largely completed while Kobus Barnard and Pinar Duygulu were at the University of California, Berkeley. The project was part of the Digital Libraries Initiative sponsored by NSF and many others. Kobus Barnard also received funding from NSERC (Canada), and Pinar Duygulu was funded by TUBITAK (Turkey). We are grateful to Jitendra Malik and Doron Tal for normalized cuts software, and Robert Wilensky for helpful conversations.

REFERENCES

1. D. Marr, *Vision*, Freeman, 1982.
2. S. Santini, Semantic Modalities in Content-Based Retrieval, *IEEE International Conference on Multimedia and Expo* (2000).
3. 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).
4. R. K. Srihari and D. T. Burhans, Visual Semantics: Extracting Visual Information from Text Accompanying Pictures, *AAAI '94* (1994).
5. R. K. Srihari, R. Chopra, D. Burhans, M. Venkataraman, and V. Govindaraju, Use of Collateral Text in Image Interpretation, *ARPA Image Understanding Workshop* (1994).
6. 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 PAMI. Available in the interim from http://HTTP.CS.Berkeley.EDU/~carson/papers/pami.html.
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. K. Barnard, P. Duygulu, and D. Forsyth, Modeling the Statistics of Image Features and Associated Text, *Document Recognition and Retrieval IX - Electronic Imaging* (2002).
9. K. Barnard, P. Duygulu, and D. Forsyth, Clustering Art, *IEEE Conference on Computer Vision and Pattern Recognition*, II:434-441 (2001).
10. N. C. Rowe, Marie-4: A high-recall self-improving web crawler that finds images using captions, *IEEE Intelligent Systems*, **July/August**, 8-14 (2002).
11. M. Markkula and E. Sormunen, End-user searching challenges indexing practices in the digital newspaper photo archive, *Information retrieval*, **1**, 259-285 (2000).
12. P. G. B. Enser, Progress in documentation pictorial information retrieval, *Journal of Documentation*, **51**, 126-170 (1995).
13. P. G. B. Enser, Query analysis in a visual information retrieval context, *Journal of Document and Text Management*, **1**, 25-39 (1993).
14. L. H. Armitage and P. G. B. Enser, Analysis of user need in image archives, *Journal of Information Science*, **23**, 287-299 (1997).
15. K. Barnard and D. Forsyth, Learning the Semantics of Words and Pictures, *International Conference on Computer Vision*, II:408-415 (2001).
16. C. O. Frost, B. Taylor, A. Noakes, S. Markel, D. Torres, and K. M. Drabenstott, Browse and search patterns in a digital image database, *Information retrieval*, **1**, 287-313 (2000).
17. D. A. Forsyth and J. Ponce, *Computer Vision - A Modern Approach*, in press.
18. D. A. Forsyth, Computer Vision Tools for Finding Images and Video Sequences, *Library Trends*, **48**, 326-355 (1999).
19. H. Schneiderman and T. Kanade, A Statistical Approach to 3D Object Recognition Applied to Faces and Cars, *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 100 (2000).
20. M. M. Fleck, D. A. Forsyth, and C. Bregler, Finding Naked People, *4th European Conference on Computer Vision*, Springer, II:591-602 (1996).
21. M. Oren, C. Papageorgiou, P. Sinha, and E. Osuna, Pedestrian detection using wavelet templates, *Computer vision and pattern recognition*, 193-9 (1997).
22. J. P. Eakins, Towards intelligent image retrieval, *Pattern Recognition*, **35**, 3-14 (2002).
23. S. Santini, A. Gupta, and R. Jain, Emergent semantics through interaction in Image Databases, *IEEE Transactions on Knowledge and Data Engineering*, (2001).
24. S. Ornager, View a picture. Theoretical image analysis and empirical user studies on indexing and retrieval, *Swedis Library Research*, **2**, 31-41 (1996).
25. L. H. Keister, User types and queries: impact on image access systems, in *Challenges in indexing electronic text and images*. Learned Information (1994).
26. M. J. Swain, C. Frankel, and V. Athitsos, WebSeer: An Image Search Engine for the World Wide Web, Computer Science Department, University of Chicago TR-96-14, available from (1996).

27. J.-Y. Chen, C. A. Bouman, and J. C. Dalton, Hierarchical Browsing and Search of Large Image Databases, *IEEE Transactions on Image Processing*, **9**, 442-455 (2000).
28. R. Srihari, Extracting Visual Information from Text: Using Captions to Label Human Faces in Newspaper Photographs, SUNY at Buffalo, Ph.D. thesis, (1991).
29. O. Maron, Learning from Ambiguity, Massachusetts Institute of Technology, Ph.D., (1998).
30. O. Maron and A. L. Ratan, Multiple-Instance Learning for Natural Scene Classification, *The Fifteenth International Conference on Machine Learning* (1998).
31. Y. Mori, H. Takahashi, and R. Oka, Image-to-word transformation based on dividing and vector quantizing images with words, *First International Workshop on Multimedia Intelligent Storage and Retrieval Management (in conjunction with ACM Multimedia Conference 1999)* (1999).
32. D. Melamed, *Empirical methods for exploiting parallel texts*, MIT Press, Cambridge, Massachusetts, 2001.
33. D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, Prentice-Hall, 2000.
34. C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press. Cambridge, MA, 1999.
35. J. Shi and J. Malik., Normalized Cuts and Image Segmentation, *IEEE Trans. Patt. Anal. Mach. Intell.*, **22**, 888-905 (2000).
36. K. Barnard, P. Duygulu, N. D. Freitas, D. Forsyth, D. Blei, and M. I. Jordan, Matching Words and Pictures, *Journal of Machine Learning Research* (in press).
37. P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer, The mathematics of machine translation: parameter estimation, *Computational Linguistics*, **19**, 263-311 (1993).
38. 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).
39. P. Duygulu, K. Barnard, J. F. G. D. Freitas, and D. A. Forsyth, Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, *The Seventh European Conference on Computer Vision*, IV:97-112 (2002).
40. 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).
41. T. Hofmann and J. Puzicha, Statistical models for co-occurrence data, Massachusetts Institute of Technology, A.I. Memo 1635, available from (1998).
42. R. Jonker and A. Volgenant, A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems, *Computing*, **38**, 325-340 (1987).

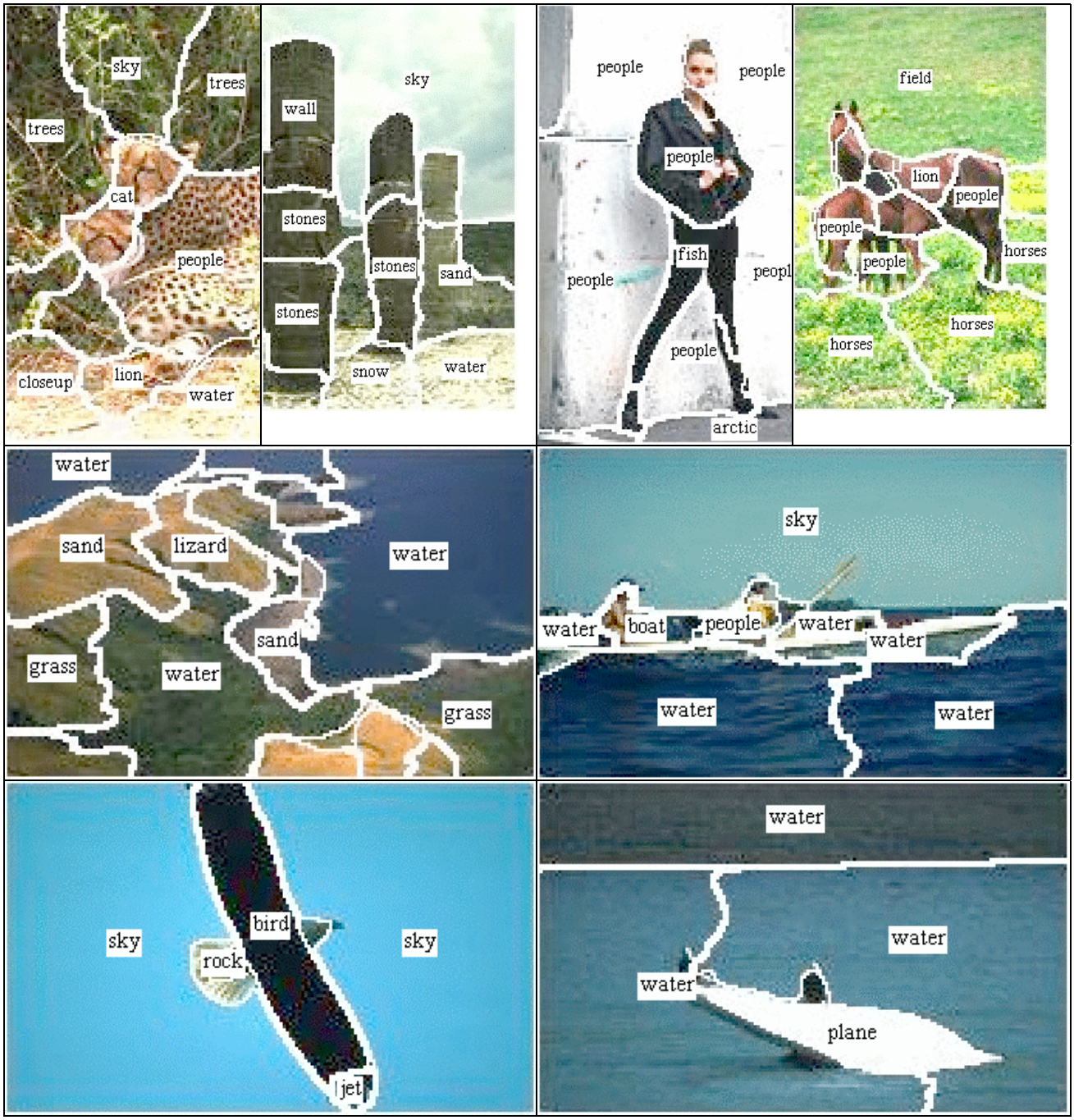

Figure 1. Examples of region based annotation showing a variety of results using the dependent method without cluster information on images from the held out set. Only the maximal probability word for each region is shown. The task here is very difficult. Some images illustrate some of the problems which we are currently working on. For example, the word horses on the green areas on the rightmost image on the top row illustrates the difference between annotation and recognition. The emission of horses would help this image to have a good annotation score, but the correspondence is incorrect. The reason is that in our data set, horses typically are on this kind of background, and this kind of background is rare in non-horse images. If two kinds of regions always co-occur, then the system cannot learn to tell them apart.