The Effects of Segmentation and Feature Choice in a Translation Model of Object Recognition

Kobus Barnard[1], Pinar Duygulu[2], Raghavendra Guru[1], Prasad Gabbur[1], and David Forsyth[3]

[1] *Department of Computing Science,*
University of Arizona
{kobus, graghave}@cs.arizona.edu
pgsangam@ece.arizona.edu

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

[3] *Computer Science*
Division, University of
California, Berkeley
daf@cs.berkeley.edu

Abstract

We work with a model of object recognition where words must be placed on image regions. This approach means that large scale experiments are relatively easy, so we can evaluate the effects of various early and mid-level vision algorithms on recognition performance.

We evaluate various image segmentation algorithms by determining word prediction accuracy for images segmented in various ways and represented by various features. We take the view that good segmentations respect object boundaries, and so word prediction should be better for a better segmentation. However, it is usually very difficult in practice to obtain segmentations that do not break up objects, so most practitioners attempt to merge segments to get better putative object representations. We demonstrate that our paradigm of word prediction easily allows us to predict potentially useful segment merges, even for segments that do not look similar (for example, merging the black and white halves of a penguin is not possible with feature-based segmentation; the main cue must be "familiar configuration").

These studies focus on unsupervised learning of recognition. However, we show that word prediction can be markedly improved by providing supervised information for a relatively small number of regions together with large quantities of unsupervised information. This supervisory information allows a better and more discriminative choice of features and breaks possible symmetries.

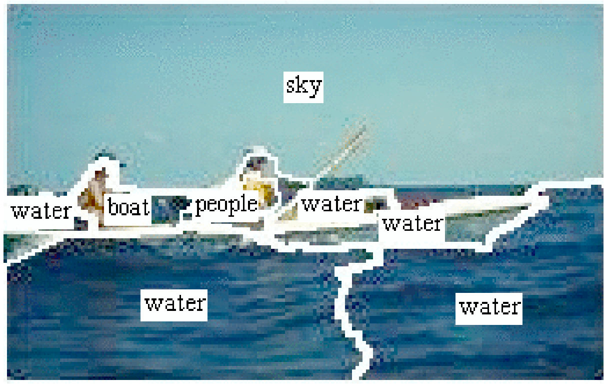

Figure 1. Illustration of labeling. Each region is labeled with the maximally probable word, but a probability distribution over all words is available for each region.

1. Introduction

We adopt a model of object recognition where words must be placed on image regions [1-3]. This is achieved in practice by exploiting large image data sets with associated text. Critically, we do not require that the text be associated with the image regions, as such data is rare. Considering processes which translate from images (visual representation) to words (semantics) gives a handle on a number of difficult computer vision problems. In part, this is because translation performance can be measured on a large scale, by comparing at the proposed translation (predicted words) with the actual translation (associated text).

We use word prediction performance to evaluate image segmentation and feature choices. It is widely agreed that segmentation measures should be task oriented (see [4] for related work). We argue that word prediction is an excellent task because it is associated with higher level image semantics and recognition. An orthogonal recent approach is to link segmentation performance to those provided by human subjects [5, 6].

Perfect segmentation is not available, and even if contours are perfectly followed, low level approaches cannot consistently deliver groupings reflecting semantics. For example, low level segmenters cannot merge the black and white halves of a penguin. We propose using region word prediction as a vehicle for integrating the higher level processes with the lower level ones. Specifically merges are proposed between regions with similar word posteriors. Such a mechanism should facilitate the learning of familiar configuration, and possibly shape, as shape descriptions should become more pertinent as appropriate regions are merged.

While the system we are building is purposely designed to learn from data with minimal structure, there are limits to what can be done without supervision. For example, it can be difficult to learn to distinguish items that tend to co-occur. However, supervised data is difficult and expensive to collect, so in §7 we study methods that improve performance using a small amount of labeled data.

The translation model for machine vision has many elements from a long history of computer vision research. Early work along the lines of traditions artificial intelligence (summarized in [7]) proposed paradigms for reasoning about image pieces with labels. There is also previous work on learning region classification from labeled [8, 9] and semi-labeled data [10, 11] Recent work on integrating words and text to improve image retrieval is also relevant [12-14].

2. Predicting words from images

A number of methods have recently been described for predicting words from segmented images [1-3, 15]. For most of the results reported in this paper we use a special case of one of the models in [3]. Specifically, we model the joint probability of words and images regions as being generated by a collection of nodes, each of which has a probability distribution over both words and regions. The word probabilities are provided by simple frequency tables, and the region probability distribution are Gaussians over feature vectors. We restrict the Gaussians to have diagonal covariance (features are modeled as being independent).

Given an image region, its features imply a probability of being generated from each node. These probabilities are then used to weight the nodes for word emission. Thus words are emitted conditioned on image regions. In order to emit words for an entire image (auto-annotation), we simply sum the distributions for the N largest regions. Thus each region is given equal weight, and the image words are forced to be generated through region labeling.

To be consistent with the more general models referenced above, we index the nodes by "levels", l. Given a region ("blob"), b, and a word w, we have

$$P(w \mid b) = \sum_{l} P(w \mid l) P(b \mid l) P(l) \big/ P(b) \qquad (1)$$

where P(l) is the level prior, P(w|l) is a frequency table, and P(b|l) is a Gaussian over features. To estimate the conditional density of words given blobs for the entire image these probabilities are summed over the N largest blobs. Parameters for the conditional probabilities linking words and blobs are estimated from the word-blob co-occurrence data using Expectation Maximization [16]. For all experiments reported in this paper we use 500 nodes.

For the work on supervision (§7) we use a simpler model [2] where the image regions are first discretized using K-means clustering, and then a machine translation algorithm [17, 18] is used to simultaneously learn the translation table and the correspondences.

3. Experimental Protocol

For the bulk of 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 test (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 evaluated on the basis of at least 1000 images. The results of 10 such samples were further averaged. This controls for both the input data and EM initialization. Words occurring less than 20 times in the training set were excluded. The number of words in the vocabulary varied from 153 to 174 over the 10 runs.

For the segmentation evaluation and segment merging experiments we used a modest selection of features for each segment, including size, position, color, oriented energy (12 filters), differential response of 2 different Gaussian filters, a few simple shape features—essentially consistent with recent work on linking words with images [2, 3, 15]. We normalize all features so that in the training data each has mean zero and variance one.

For the feature evaluation experiments images were segmented using Normalized Cuts [19].

Performance measures. Several ways to quantify word prediction performance have been proposed [3]. Here we use the simplest measure. Specifically, we allow the model to predict M words, where M is the number of words available for the given test image. In our data M varies from 1 to 5. The number correct divided by M is the score.

In all results reported for segmentation, feature choice, and region merging, we express word prediction relative to that for the empirical word distribution—i.e., the frequency table for the words in the training set. This reduces variance due to varied test sample difficulty. Exceeding the empirical density performance 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.

4. Semantic based segmentation evaluation

We evaluate six variants from three classes of segmentation methods: the expectation-maximization segmenter used for Blobworld [20-22], Normalized Cuts [19], and the mean shift algorithm [23]. The implementation of Normalized Cuts available to us provides both over-segmented initial output (“preseg”) as well as the finished results (“ncuts”). Similarly, the mean shift implementation, kindly made available on-line [24], gives three options (over segmentation, under segmentation, and quantization). Example segmentations from the three classes is shown in Figure 2.

A possible confound in our process is the number of segments used for word prediction and thus in Figure 3 we plot performance as a function of using the largest 2, 4, 6, 8, 10, and 12 regions. The large scale of our experiments—results for 10,000 images are used for each data point—means we can estimate errors for each plotted value (indicated by error bars).

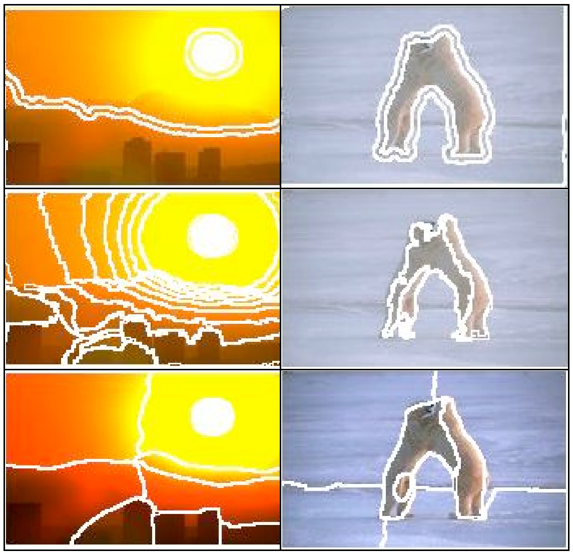

Figure 2. Examples of segments from Blobworld (top), mean shift quantized (middle) and normalized cuts (bottom).

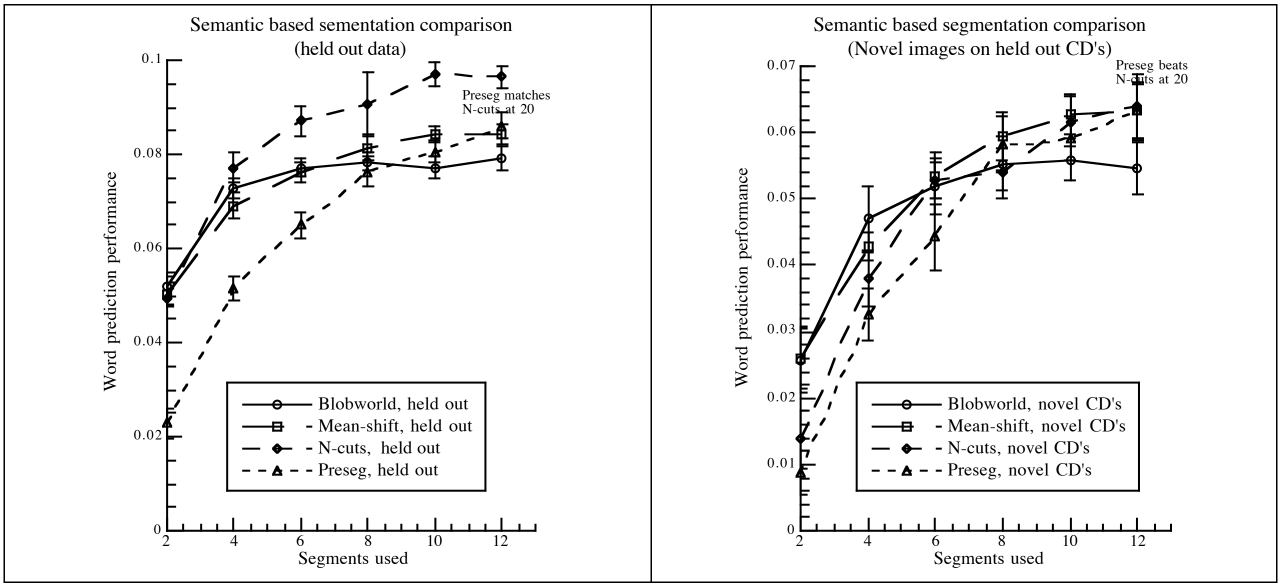

Figure 3. Segmentation methods compared using word prediction performance, evaluated on held out date (left), and novel data (right). All values plotted are positive, which means that performance always exceeds that using the empirical distribution. We restrict the plots to the “over-segment” version of mean-shift to avoid clutter—the other versions give results close to this one. Some of the segmentation approaches are shown to be significantly different, given the error estimates indicated by the bars around the points. The errors were estimated from the variance of the word prediction process over 1000 different images over 10 input sets.

We find that ncuts provides distinctly better support (well outside of error) for word prediction compared with the Blobworld EM segmenter. The mean-shift algorithm is somewhere between the two, again significant given the error estimates in the case of the first held out set. For the novel images, the order remains the same but there is more variance. Interestingly, preseg seems to be comparable to ncuts, provided that we increase the number of segments to 20 (not plotted). Additional experiments are needed before we can say whether there is a real difference.

5. Semantic based feature evaluation

We apply a similar strategy to evaluating features. Here we keep the segmenter and the number of regions fixed (normalized cuts, 8 regions), and investigate word prediction performance as a function of features. In addition to the feature sets used in previous work, we experiment with several others, including a more comprehensive shape descriptor and color context as described below.

Since it is impractical to evaluate all combinations of features we break them into groups. We consider a "base" set of features which consists of region size, location, and two simple shape features, namely the first moment of the region, and the area divided by the square of the outer boundary length.

We consider adding color as encoded in three different ways—straight RGB, L*a*b, and chromaticity with brightness, specifically, S=R+G+B, r=R/S, and g=G/S. In all case both the average color and its variance over the region is used. Thus color adds 6 numbers to our feature vector.

Texture is represented by a combination of the average energy response to 12 filters with different orientations, and the average response to the difference of 4 different combinations of 2 Gaussian filters.

Our base features include minimal shape information. It is not clear whether our segmentations of thumbnail sized images contains usable shape information. In this work we attempt to test this. Shape is difficult to characterize using a modest sized feature vector, but we wanted to keep the number of components roughly the order of what is known to be manageable by our learning procedure. Thus we choose to encode a limited amount of shape information in 30 numbers. We considered only the outer boundary of the each region, normalized for the length of the boundary, and parameterized the distance from the center of mass by arc-length. The result was then smoothed and sampled at 30 points. The first point was taken to be the top left corner. We specifically did not make the shape descriptor invariant to rotation on the assumption that the photographer bias for upright images means that the orientation of the shape carries usable information. (We have yet to test that assumption.)

By color context we mean the average color adjacent to regions in various directions. It is intuitively reasonable as a feature to try for improved word prediction. For example, a brown blob is more likely to be a bird, and less likely to be dirt, if it is surrounded by light blue. To compute color context we start be computing the average distance of the outer boundary of a region from its center mass. Then we consider all points within twice this distance in 4 quadrants aligned at 45 degrees to the image axis. For each of the four wedges (top, bottom, left, right), we average the colors in the wedge but not in the region, provided that there are more than 100 such points. Otherwise the average color of the region itself is used. This gives 12 numbers for each region.

In Table 1 we give word prediction performance for a number of combinations of features. Not surprisingly given the nature of the Corel data, color is most important. Interestingly, color space makes a significant difference (more than we expected). Chromaticity plus brightness does the best, and both it and L*a*b do significantly better than RGB. This ranking suggests that correlation among the color components is a likely source of trouble (recall that we treat features as independent). This also suggests that steps should be taken to reduce the correlation among other features.

Color context helps, but not as much as we hoped. Color context was conveniently computed in terms of RGB. The above finding on the effect of color space suggests that we should test color context expressed in the chromaticity plus brightness space.

Table 1. Word prediction performance for a variety of feature sets. More features is certainly not better, likely due to over-training and noise. Color is the the best single cue, followed by texture.

Feature set	Word prediction performance on the various data sets (error is roughly 0.003)		
	Training	Held out	Novel
Base set	0.019	0.020	0.018
Base set, RGB	0.076	0.057	0.044
Base set, L*a*b	0.097	0.085	0.061
Base set, rgS	0.109	0.092	0.065
Base, rgS, color context	0.134	0.094	0.055
Base set, texture	0792	0.048	0.041
Base, rgS, texture	0.109	0.072	0.059
Base set, shape	0.053	0.016	0.017
Base set, rgS, shape	0.065	0.029	0.027
Base,rgS, texture, shape	0.083	0.043	0.038
Everything	0.097	0.055	0.039

Table 2. The relationship between a histogram of machine proposed merges and the human evaluation thereof. We are encouraged that as the strength of the proposal rises, it is more likely to be judged as appropriate by the human evaluator. The error estimates shown in parentheses indicate that the proposed merges are significantly different from chance.

Range of merge proposal scores based on word prediction. Each group consists of 982 or 983 merges	Average human evaluation score
0-20%	0.420 (0.016)
20-40%	0.436 (0.016)
40-60%	0.471 (0.016)
60-80%	0.484 (0.016)
80-100%	0.525 (0.016)

Texture also carries some usable information—using it with only the base set gives significant improvement, but when used in conjunction with color the increment is not that large. This may be due to the fact that the variance we include with color carries some texture information. Additional experiments will sort this out.

Utilizing shape proved to be problematic. It is clear from the results on the training data that our shape feature carries usable information but the results on the held out data reveal that what was captured does not generalize well. We are not particularly surprised by this result given the nature of our segment boundaries obtained from small images. However, the results also indicate that longer feature vectors can make things worse—even though they provide more information—which indicates that over-training needs to be investigated as a source of difficulty.

Assuming that we can overcome these problems, we are still left with the reality that learning shape from data is confounded by the fact that objects are often split up by segmenation processes, often because making the pieces into an object requires higher level information. To tackle learning the appropriate shape templates from familiar configurations we propose the novel strategy of using word prediction posterior probabilities to merge image regions

6. Merging regions using word prediction

It is generally assumed that segmentations based on low level features are not entirely appropriate for recognition. For example, merging the white and black halves of a penguin requires deference to some other processes. We propose that associated text is a possible broker for these processes, providing the tokens upon which salient familiar configurations can be learnt. As a first step in this direction we offer a simple region proposal strategy based on word prediction.

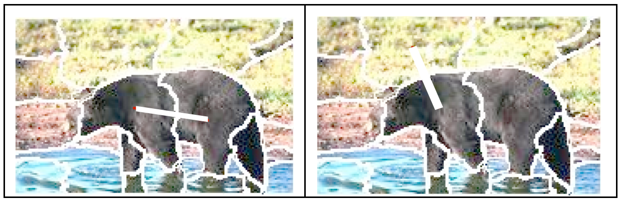

Figure 4. Example proposed merges illustrated by the line connecting two regions in each of the images. The example on the left is evaluated as good, and the example on the right is evaluated as bad. To evaluate the automatic merge proposal process, we examined 5911 images like these.

Specifically we look at the correlation between the word posteriors for adjacent regions. The higher the dot product of the two vectors representing the word posterior, the more attractive the merge.

To evaluate our region merging mechanism we examined all possible merges for 50 held out images for each of our 10 samples. A total of 5911 images like the ones in Figure 4 were rated. Two evaluators did approximately half of the scoring apiece. The evaluators were blind to the machine merge score, Each merge was rated as accept or reject. Roughly 10% (590) were rated as "undecided" and thus ignored. We wish to test for the existence of a relationship between the merge score above and human rating, but of course the data is very noisy. Thus we rank the machine scores, and put them into bins representing the worst 20%, the next best 20%, and so on. For each bin, we compute the average acceptance score. The results in Table 2 demonstrate that the merges with higher scores are more likely to be accepted by the human evaluators.

7. Using Labeled Data

For our purposes, supervised recognition data is obtained by manually attaching word labels to image regions. Such data is never going to be available in large quantities, and needs to be used jointly with unsupervised data. However, quite small quantities of supervised data can help the recognition process in two ways: selecting appropriate representations of image information, and breaking symmetries present in unsupervised data.

7.1. Learning to Match Regions and Words

It is easier to study the effects of labeled data using a somewhat simpler process to attach labels to image regions than that used to study segmentation. In [2], an unsupervised process learns to attach words to image regions by vector quantizing a representation of the image region, and then building a joint probability table using EM. The missing variables are the correspondence between image regions and annotation words. This process is exactly analogous with that used to build lexicons in statistical machine translation [25]. The difficulty with this approach is that the representation of the image regions is obtained completely independent of the words. Ideally, one would want have clusters of image regions that tend to result in few distinct words. Supervised data can achieve this.

7.2 Selecting Appropriate Representations

In particular, we use the following procedure. Supervised data supplies a small set of image regions where we know both the label and features for the region. As in §3, each image region is originally represented by 30 features; these features are shifted and scaled to have zero mean and unit variance. We perform principal components analysis on the result, to reduce the dimension to 11 for stability, and then obtain linear discriminant features. This yields a feature space, within which we have a small number of labeled elements. We construct one cluster per appearance type. The vocabulary is reduced to only the label words required to describe the blobs in the selected CDs. We add the word `null` to the vocabulary, and to the annotations of each image. Unsupervised data is now vector-quantized by nearest neighbours in the feature space—this means that an unlabelled image region is assigned to the cluster belonging to the closest labeled image region.

There are now two possibilities. First, we could build a nearest neighbour classifier, where an unlabelled image region is given the *label* of the closest labeled image region. As the results show, this method suffers from the relatively small amount of labeled data available. Second, we could assign unlabeled image regions to the *cluster* of the closest labeled image region, but still learn the joint probability of image clusters and words from data using EM. This has the considerable advantage that we can still use the unsupervised word and image data.

7.3. Breaking Symmetries

It can be difficult to learn region-word correspondences from annotated images if the entropy of the annotations is not high (the usual case). In the extreme case, two words always appear together in annotations, and so the incomplete data log-likelihood has a symmetry—the `horses` could be green and the `grass` brown, or the other way round. Even small amounts of labeled data should break this symmetry. Manual labeling is easily incorporated into the method of [2]; one fixes the correspondences that are known between image regions and words, and fills in missing correspondences with EM, as before.

8. Experimental Results on Supervised Data

We expect that labeled data will primarily affect *correspondence*—which region in the image is annotated with which term. Here the annotation measure is clearly only a proxy. It seems reasonable to assume that arranging word prediction to occur only through individual regions (and not partly through image clusters as in [1]) makes the proxy measure more appropriate. However, we still need to check the correspondences, and this is done by hand; what this means is the scale of experiments must be somewhat smaller.

Data sets: We used 6 CDs from the Corel dataset to test the effect of labeled data. Each CD contains 100 images. Segmented versions of ten images from each CD are labeled by hand. In this collection, each CD represents a specific topic (“tigers”, “planes”, etc.) and so only a few keywords are sufficient to describe a CD.
Each CD is split into a 70 image unlabeled training set, a 10 image labeled training set (where word-region correspondences are manually identified) and a 20 image test set.

Strategies compared are:
1) The method of [2], but with only unsupervised data .
2) A method where labeled data are used to produce clusters of image regions as in §7.3, but where the joint probability table between clusters of image regions and words is learned using unsupervised data.
3) A method where labeled data are used to produce clusters of image regions and where the joint probability table between clusters and words is learned with a combination of supervised and unsupervised data.
4) A method where labeled data is used to produce a nearest neighbour classifier—image regions are assigned the label of the nearest labeled example.

Evaluation is difficult for this task, because to check correspondence between image regions and words, one must check each label assignment by hand. We use annotation performance as a proxy, and also check labelings of image regions by hand. Table 3 compares (a) annotation performance (the extent to which the method annotates images with words that are the same as those supplied) and (b) correspondence performance (the total number of regions that have words on them that are correct) for each of the four methods described above. Method 3 is better than method 2, and method 2 is significantly better than method 1; method 4 is better at correspondences than any other method, but does somewhat worse at annotation. This appears to be because there is relatively little data with which to train a nearest neighbour method; the unsupervised data improves performance by being available in bulk.

For methods 1, 2, and 3 one can compute the mutual information in the learned joint probability distribution coupling words and region descriptors. There are 15 words, and 22 blobs. The mutual information for joint probability tables linking words and blobs are, in bits, 1.25, 1.24 and 1.32 for methods 1,2 and 3 respectively. The maximum possible value is 3.72. Notice that supervisory information on image region clusters alone appears to make little difference, but supervising both clustering and correspondence results in a significant difference. While the value is low, it clearly shows the impact of supervisory information.

Nearest neighbours (method 4) has another significant flaw: it tends to over-predict words, as table 4 indicates.

Table 3. The effects of supervision on word prediction performance measured by comparing four different methods. The first three columns measure annotation prediction on unlabeled training, labeled training and test data respectively. For each image, the method predicts the number of annotations actually present in the image, and is scored based on the fraction of those predictions that are correct. The final column gives the total number of regions that *correspond* correctly, out of a maximum of 301.

method	unlabeled training	labeled training	test	correspondence
1	0.0692	0.1431	0.0597	15
2	0.0794	0.1736	0.0988	113
3	0.0782	0.1736	0.0811	202
4	0.0687	0.1486	0.1025	301

Table 4. The table shows correspondence results and false positive rates on a test data set for a set of words for methods 3 and 4. Correspondence results are obtained by predicting words using the method, and then checking each image by hand. We show the number of times the term was predicted correctly, the total number of predictions made, and the percentage of predictions that are correct. Notice that the two methods make correct predictions at about the same rate, but that method 4 predicts words very much more often than method 3. We also show the false positive results (fp) which are obtained using the annotation performance as a proxy. Nearest neighbor method creates more false postives.

	both supervised		nearest neighbor	
	correspondence	fp	correspondence	fp
eagle	0 / 0 (NAN)	0.00	4 / 63 (6%)	0.84
elephant	5 / 30 (17%)	0.77	4 / 30 (13%)	0.77
field	6 / 54 (11%)	0.85	6 / 54 (11%)	0.85
forest	0 / 0 (NAN)	0.00	0 / 5 (0%)	0.95
grass	10 /31(32%)	0.77	19 / 54 (35%)	0.78
horses	5 / 42 (12%)	0.82	5 / 37 (14%)	0.84
lion	2 / 35 (6%)	0.75	2 / 23 (9%)	0.76
plane	9 / 40 (23%)	0.70	9 / 40 (23%)	0.70

9. Discussion

Studying object recognition as a word prediction task has the attraction that it finesses the finer details of the recognition problem (what object representations to adopt; how to group shape representations; how to reason about pose) and makes it possible to do large scale experiments on the broader aspects of recognition. Such experiments give a rough but useful evaluation of different representations. We have shown that segmenters differ considerably in their ability to support word prediction—choosing a better segmenter will make a real difference in a practical problem. Furthermore, we have shown that the right choice of features will also improve performance. However, the great difficulty in the current word prediction paradigm is the lack of comprehensive shape representation. Shape representations are hard to incorporate, because the representation typically must be formed out of more than

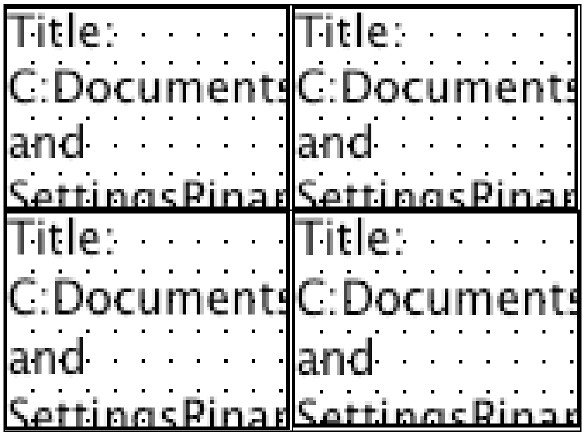

Figure 5. Each column shows images labeled using one of the four strategies (reading from the left): method 1 (k-means clustering of image regions); method 2 (supervised clustering of image regions, but unsupervised word-region correspondence learning); method 3 (supervised data used both to cluster image regions and to learn correspondences); and method 4 (nearest neighbours)

one region, and the criteria by which regions should be merged are obscure: "familiar configuration" simply means that the regions do better together than apart. We have shown that region merges suggested by a word prediction criterion are good; this suggests that it should be possible to take these merged regions and construct a shape representation that in turn enhances recognition. Finally, we have shown that supervised data, even in small amounts, can significantly improve word prediction rates by improving both the representation adopted for image regions and the correspondence established between regions and words.

Acknowledgements

This project is part of the Digital Libraries Initiative sponsored by NSF and many others. We are grateful to Jitendra Malik and Doron Tal for Normalized Cuts software. We also thank Fatos Yarman-Vural for many helpful discussions.

References

[1] K. Barnard and D. Forsyth, "Learning the Semantics of Words and Pictures," *Proc. International Conference on Computer Vision*, pp. II:408-415, 2001.

[2] 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," *Proc. The Seventh European Conference on Computer Vision*, Copenhagen, Denmark, pp. IV:97-112, 2002.

[3] K. Barnard, P. Duygulu, N. d. Freitas, D. Forsyth, D. Blei, and M. I. Jordan, "Matching Words and Pictures," *Journal of Machine Learning Research*, vol. 3, pp. 1107-1135, 2003.

[4] S. Borra and S. Sarkar, "A Framework for Performance Characterization of Intermediate Level Grouping Modules," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1306-1312, 1997.

[5] D. Martin, C. Fowlkes, D. Tai, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Proc. International Conference on Computer Vision*, pp. II:416-421, 2001.

[6] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using brightness and texture," *Proc. Neural Information Processing Systems*, pp. to appear, 2002.

[7] D. H. Ballard and C. M. Brown, *Computer Vision*: Prentice-Hall, 1982.

[8] N. W. Campbell, W. P. J. Mackeown, B. T. Thomas, and T. Troscianko, "Interpreting image databases by region classification," *Pattern Recognition*, vol. 30, pp. 555-563, 1997.

[9] N. W. Campbell, B. T. Thomas, and T. Troscianko, "Automatic segmentation and classification of outdoor images using neural networks.," *International Journal of Neural Systems*, vol. 8, pp. 137-144, 1997.

[10] O. Maron and A. L. Ratan, "Multiple-Instance Learning for Natural Scene Classification," *Proc. The Fifteenth International Conference on Machine Learning*, 1998.

[11] Y. Mori, H. Takahashi, and R. Oka, "Image-to-word transformation based on dividing and vector quantizing images with words," *Proc. First International Workshop on Multimedia Intelligent Storage and Retrieval Management (in conjunction with ACM Multimedia Conference 1999)*, Orlando, Florida, 1999.

[12] R. K. Srihari and D. T. Burhans, "Visual Semantics: Extracting Visual Information from Text Accompanying Pictures," *Proc. AAAI '94*, Seattle, WA, 1994.

[13] R. K. Srihari, R. Chopra, D. Burhans, M. Venkataraman, and V. Govindaraju, "Use of Collateral Text in Image Interpretation," *Proc. ARPA Image Understanding Workshop*, Monterey, CA, 1994.

[14] M. La Cascia, S. Sethi, and S. Sclaroff, "Combining Textual and Visual Cues for Content-based Image Retrieval on the World Wide Web," *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, 1998.

[15] K. Barnard, P. Duygulu, and D. Forsyth, "Clustering Art," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Hawaii, pp. II:434-441, 2001.

[16] 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)*, vol. 39, pp. 1-38, 1977.

[17] P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, and R. L. Mercer, "The mathematics of machine translation: parameter estimation," *Computational Linguistics*, vol. 19, pp. 263-311, 1993.

[18] P. F. Brown, J. Cocke, S. A. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin, "A statistical approach to machine translation," *Computational Linguistics*, vol. 16, pp. 79-85, 1990.

[19] J. Shi and J. Malik., "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888-905, 2000.

[20] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color-and Texture-based Image Segmentation Using the Expectation-Maximization Algorithm and Its Application to Content-Based Image Retrieval," *Proc. International Conference on Computer Vision*, 1998.

[21] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Color and Texture-Based Image Segmentation Using EM and Its Application to Image Querying and Classification," *IEEE PAMI*, vol. 24, pp. 1026-1038, 2002.

[22] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik, "Blobworld: Image segmentation using Expectation-Maximization and its application to image querying," *Proc. Third Int. Conf. on Visual Information Systems*, 1999.

[23] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis," *IEEE Trans. Patt. Analy. Mach. Intell.*, vol. 24, 2002.

[24] D. Comaniciu, "Mean Shift," 2002.

[25] D. Melamed, *Empirical methods for exploiting parallel texts*. Cambridge, Massachusetts: MIT Press, 2001.