Evaluation of localized semantics: Data, methodology, and experiments

KOBUS BARNARD, QUANFU FAN, RANJINI SWAMINATHAN, ANTHONY HOOGS,
RODERIC COLLINS, PASCALE RONDOT, JOHN KAUFHOLD

Computer Science Department, The University of Arizona

GE Global Research, One Research Circle, Schenectady, New York, 12309

{kobus, quanfu, ranjini}@cs.arizona.edu

hoogs@crd.ge.com, collins@research.ge.com

University of Arizona, Computing Science,
Technical Report, TR-05-08,
September 12, 2005
(Significanly revised October 26, 2006)

Abstract. We present a new data of 1014 images with manual segmentations and semantic labels for each segment, together with a methodology for using this kind of data for recognition evaluation. The images and segmentations are from the UCB segmentation benchmark database (Martin et al., 2001). The database is extended by manually labeling each segment with its most specific semantic concept in WordNet (Miller et al., 1990). The evaluation methodology establishes protocols for mapping algorithm specific localization (e.g., segmentations) to our data, handling synonyms, scoring matches at different levels of specificity, dealing with vocabularies with sense ambiguity (the usual case), and handling ground truth regions with multiple labels. Given these protocols, we develop two evaluation approaches. The first measures the range of semantics that an algorithm can recognize, and the second measures the frequency that an algorithm recognizes semantics correctly. The data, the image labeling tool, and programs implementing our evaluation strategy are all available on-line (kobus.ca//research/data/IJCV).

We apply this infrastructure to evaluate four algorithms which learn to label image regions from weakly labeled data. The algorithms tested include two variants of multiple instance learning (MIL), and

two generative multi-modal mixture models. These experiments are on a significantly larger scale than previously reported, especially in the case of MIL methods. More specifically, we used training data sets up to 37,000 images and training vocabularies of up to 650 words.

We found that one of the mixture models performed best on image annotation and the frequency correct measure, and that variants of MIL gave the best semantic range performance. We were able to substantively improve the performance of MIL methods on the other tasks (image annotation and frequency correct region labeling) by providing an appropriate prior.

1 Introduction

Demonstrating recognition requires specifying where an entity is in the image, in addition to whether or not it is present. Intuitively, if a program "recognizes" a horse in an image, but attaches the location of the horse to the grass that the horse is standing on (Figure 1), then true recognition performance is limited, and certainly does not match that of a program that can specify where the horse is. Thus to properly evaluate recognition approaches, we need to consider localization.

In this paper we quantify performance on what we see as the main challenge for general purpose recognition algorithms, namely inferring detailed, localized, semantics from image data. Systems developed for general recognition will express localization and semantics in various ways. Semantics are typically encoded in words or labels often inherited from training data that can range from fixed labels from standard test data sets to words associated with images on the web. Further, as recognition systems take advantage of more sources of information, they will be able to make choices in learning and expressing semantics. For example, a system trained on tigers, leopards, and lions may choose to output "cat" in ambiguous cases, and may even be able to do so for new cats such as cheetahs. Similarly, an algorithm trained on free form text that makes use of language tools may be able to distinguish some word senses (e.g., bank as in "river bank" versus "money bank"), and thus choose to specify the sense to output more precise semantics.

Our goal with this research is to provide publicly available infrastructure for automated evaluation of the localization of image semantics that is applicable to a wide range of algorithms as exemplified above. Since specifying detailed localized image semantics is a time consuming task, it should be done so that it

serves diverse evaluation endeavors, thereby maximizing the gain for the effort. This means that image semantics need to be characterized independently of proposed algorithms. Our approach is to provide general purpose tools that can map results from specific experiments into the semantics specified by a ground-truth data set.

To do this we begin with human image segmentations from the UCB segmentation benchmarking project (Martin et al., 2001) which provides the localization of image semantics. To label the entities we use the WordNet system (Fellbaum et al., ; Miller et al., 1990) for maximal accuracy and flexibility. We then develop a methodology for the evaluation of algorithms with disparate vocabularies and localizations with respect to the same ground truth data. To do this we map algorithm localizations onto ground truth localizations and algorithm vocabularies to the ground truth vocabulary. This is important because we wish to support a wide range of vocabularies, from object categories to free form text associated with images gathered from the web. While some practitioners may choose to work directly with the ground truth vocabulary, others may not, and we want to be able to support the characterization of all relevant algorithms.

Given these protocols, we propose two evaluation measures. The first one measures the range of semantics that an algorithm can recognize, and the second one measures the frequency that an algorithm recognizes semantics correctly. The first considers all semantic entities as equal, and aggregates the performance on each entity over the entire data set. It is immune to how common the entity is. The second takes the opposite approach, effectively counting recognition successes. It thus rewards good performance on common entities. Published results tend to be along the lines of the second strategy, or for experimental conditions where the two measures are similar, but both measures provide different useful characterizations of recognition.

1.1 Benefits of evaluating semantics with localization

Pragmatically, we need to characterize localization performance in order to understand when performance is the consequence of the visual characteristics of relevant semantic entities, and when it is due to correlations with other entities. If we can separate the effect of these two sources of information, then we can better integrate them, and design methods that exploit context, but are not overly fooled when it is not

informative. Further, while we expect that the performance of algorithms that make excessive use of the background will decrease rapidly as testing conditions depart from training conditions, characterizing localization performance exposes this more explicitly and effectively.

Localization performance helps characterize correspondence ambiguity in methods that attempt to learn from loosely labeled data. All algorithms evaluated as part of this work fall into this genre. These methods attempt to learn how to identify regions in images based on labels which apply to the image as a whole. Given a single image with multiple labels and regions, it is clearly not possible to resolve the ambiguity as to which region(s) should be linked to which word(s). Region-word consistency over multiple images, can, however, be used to resolve the ambiguity, provided that there is sufficient variety in the images and the labels. However, if words always co-occur, or effectively co-occur in patterns that are hard to characterize, then the ambiguity cannot be resolved (see Figure 1). It is thus important to be able to measure the reduction in correspondence ambiguity possible by various approaches.

Semantically segmented and labeled data is also a source of high quality training data for learning oriented algorithms. In addition to supporting standard learning approaches to vision, such data is especially useful in understanding how a small amount of supervisory data can help augment loosely labeled data which is available in large quantities.

1.2 Related work

An early region level semantically labeled image data set is the Sowerby image data base[1] (Vivarelli and Williams, 1997). Here roughly 200 images relevant to driving on British roadways were hand segmented, and given one of 54 labels from a hierarchy. This data is still useful, despite the limited domain.

More recently, the development of algorithms that use large scale weakly labeled data has produced a need for evaluating region performance. In this domain, images are assumed to have associated text such as keywords, but the part of the image that the words refer to is not known. Thus the goal of localizing the semantics is indirectly related to what is available in the data, and measuring performance directly requires region level evaluation. Due to the effort involved in gathering such data, the bulk of the evaluation of these methods has been on the proximal measure provided by the weak labels which measure how well region understanding supports predicting words relevant to the image as whole (image-

annotation). Despite the hazards of this approach, it has the advantage of supporting large scale evaluation (Barnard et al., 2003b; Barnard and Forsyth, 2001). Nonetheless, results of region labeling for a number of algorithms for 500 hand labeled regions are available (Barnard et al., 2003a). Unfortunately, this data is specific to both the segmentations and the vocabulary used in the experiments. Additional region labeling results were reported in (Carbonetto et al., 2004).

Recent progress in recognizing object categories (Agarwal et al., 2004; Berg et al., 2005; Fei-Fei et al., 2004; Fergus et al., 2003; Torralba et al., 2004; Weber et al., 2000) has prompted the gathering of a variety of image data which have been grouped together for the PASCAL Object Recognition challenge (www.pascal-network.org/challenges/VOC/). The CalTech 101 database (Fei-Fei et al., 2004) provides category data at an image level. Images are assumed to be an object from the category and some background. Despite the lack of localization, the second data set is challenging because of the large number (101) of relatively diverse categories. Data sets which have some localization data include the TU Darmstadt Database (Leibe and Schiele) the UIUC Image Database for Car Detection (Agarwal et al.), the Caltech Database (Fergus and Perona), the TU Graz-02 Database (Opelt and Pinz), and the MIT-CSAIL Database of Objects and Scenes (Torralba et al.).

While the above collections goes some distance in providing the community's need, there are several properties of the data set described here that are not available in the existing data. In particular, all regions of reasonable size are labeled (there is no generic concept of background), object contours are of high quality (many of the existing localizations are of the form of bounding boxes), labels link into a semantic structure (WordNet), and the extent of the semantic space is large (over 1,000 WordNet words).

2 Developing an image data set with localized semantic labels

Our goal is to specify localized image semantics to provide ground truth for a wide variety of evaluation experiments. To have this generality, we first focus on semantically labeling the images independently of any particular experiment. The data can then be automatically distilled for a given experiment.

We begin with images segmented into semantically coherent regions. Of course, current segmentation algorithms are not able to deliver accurate, semantically sensitive, segmentations. Thus we exploit the human segmentations for 1014 images from the Corel™ data set produced for a different study

(Martin et al., 2001). In that work, multiple segmentations were produced for each image. For this data set we used the segmentation for each image with the median number of segments.

The images in this data set are very diverse, covering a wide range of semantics. In addition to integrating with the existing framework, working with the Corel™ data is warranted because much previous training and evaluation has been done on these and similar images. However, the images themselves are not in the public domain, and must be acquired by those that do not already have them. Due to copyright restrictions, we are only able to provide derived information such as segmentation masks, label files, and feature vectors.

To accurately capture the semantics of each region, human labelers had access to the full WordNet (Fellbaum et al., ; Miller et al., 1990) vocabulary. With our labeling tool (§2.2), incorporating the rigor of WordNet does not cost much extra time, and easily justifies the benefits for current and future uses. WordNet terms are sense disambiguated which is important for encoding the semantics of images because some sense-free words (e.g. "bank") can indicate widely different semantics ("river bank" versus "money bank"). Further, WordNet terms are organized into semantic hierarchies, which are key for linking the labeling to other vocabularies (§3.6).

2.1 Labeling guidelines

To encourage consistency among labelers, we developed a set of labeling guidelines. We remark that some of the information collected according to these rules (e.g., synonyms) is effectively ignored by the specific processing strategies described below (§3), as the rules were developed in anticipation of additional uses. Our labeling rules are:

1) Words should correspond to their WordNet definition.
2) Words should be lowercase.
3) Words should be singular.
4) The sense in WordNet (if multiple) should be mentioned as word (i), where i is the sense number in Word net except if i=1. (e.g. tiger (2)).
5) Vegetation should be used for any group of plants.
6) Indiscernible objects, which clearly belong to the background, should be labeled background.
7) Add the first synonym given in WordNet as an additional entry. (e.g. building edifice).
8) Words that restrict to a part or class of an object should be word_type, like bobcat_head, the word itself should appear as another entry for that part, (e.g. bobcat bobcat_head). Wherever possible the sense of the word should be incorporated in the part also (e.g. carriage (2) rig (6) carriage (2)_shelter (2)). It is not required to use the additional synonym entry for labeling a part or class.

9) If a word is a compound word and another word can describe it as well, then add the single word (e.g. “birch tree”, birch).
10) If the same object can be described in a different way, but not necessarily synonym, label it in all ways (e.g. grass, ground).
11) If no objects are discernable in a segment it will be called “background”.
12) If several objects are discernable in a segment then all components will be labeled.
13) If an object is subject to human interpretation, add a question mark at the end of the word (e.g. human?)

We do not attempt to label ground truth regions which are smaller than 1% of image area. We have also developed additional rules specific to humans which are listed in Appendix A. While much of the additional data specific due to humans is not relevant to the experiments presented below, it is very useful to specialized applications. Figure 2 shows a labeling decision tree for the process, and Figure 3 shows two labeling examples.

2.2 Labeling tool and process

We implemented a labeling tool in Java to make the execution of the above approach as efficient as possible. The image being labeled is displayed in a window with the segment being labeled identified by a red outline (see Figure 4). The human labeler enters appropriate words either by typing them or selecting from lists. During this process, an on-line interface to WordNet is also at hand, and often words are cut and pasted from WordNet. Words that have been used recently, or with chosen similar images, and any available keywords (senses have to be added), are all available for selection. The tool can read previous labelings which is critical for checking and iterative refinement.

For this study four people contributed to the labeling of 1014 images. We estimate that a serviceable preliminary labeling took about 10 minutes per image. After all images were labeled, one labeler went over all images to check for consistency and errors.

2.3 Global statistics of the ground-truth data

Our ground-truth has 1297 labels, as well as the two special labels “unknown” and “background” which occur with 4810 and 2121 regions respectively. The “unknown” label accounts for 1.6% of the total image area, and “background” accounts for 5%, and appears as a co-label for another 8%. The most common labels (with counts in parenthesis) are: sky_1 (980), ground_1 (827), tree_1 (818), grass_1

(810), man_1 (768), woman_1 (747), rock_1 (588), stone_1 (586), water_2 (522), and building_1 (522). There are 42 labels that occur at least 100 times, and 74 labels that occur at least 50 times. Figure 4 gives an indication of the entire distribution. Naturally, many words are relatively rare. However, it is clear that there is enough data to help train classifiers for many entities.

3 General purpose evaluation methodology

Our labeling system is designed to very generally capture image semantics with locality, and thus can support a number of different methodologies for the evaluation of inferring semantics from image data. No single evaluation strategy optimally measures all interpretations and applications of this task, and we expect that our data will be used in a variety of ways. However, as an important part of this work, we examine the issues in developing an evaluation methodology, and propose strategies consistent with specific preferences regarding what should be measured. It is critical to consider what different preferences entail, choose those that are appropriate for the task at hand, and then ensure that the computations are consistent with the intention. Otherwise, algorithm characterization will be haphazard.

We find that the most critical preference is whether to focus on the *range* of semantics that can be correctly identified, or, alternatively, how *often* semantics can be correctly identified. A simple example will clarify the difference. Consider two algorithms, one which reliably identifies tigers, but nothing else, and a second one which reliably identifies sky, but nothing else. By the first notion, these two algorithms have the same performance (one semantic entity). By the second notion, the second algorithm performs better because sky is much more common, and thus a count of correctly identified entities over a reasonable test set will be higher.

Notice that we do not attempt to control for the difficulty of recognizing a particular entity, which is generally not known. We simply suggest that success should be weighted consistently with the evaluation objective. We further comment that adding scores over disparate entities is most warranted when evaluating general recognition approaches. However, the data can be used to compare performance of more specific algorithms on any subset of the ground truth entities that have sufficient examples.

3.1 General framework

We assume a very general framework where algorithms for inferring semantics applied to the ground truth images output a weighting over algorithm dependent labels for algorithm dependent localizations. In this work we further assume that the localizations are regions, although it is not difficult to arrange for other kinds of localizations. Under these assumptions, the methodology distills down to specifying the score warranted for each algorithm vocabulary word for each algorithm region. In other words, we compute a different score matrix for each algorithm. Various choices, such as semantic range versus frequency correct, are reflected in the values in the score matrices.

3.2 Vocabulary and localization mapping

To implement our scoring strategies, we map each algorithm's native semantics (vocabulary) into the ground truth semantic space, and each algorithm's localization into the ground truth segmentation. As an example, consider evaluating the region labeling in Figure 5. The segmentation is not semantically accurate and does not correspond to the one used in the evaluation infrastructure. The words are not necessarily in the ground truth vocabulary, and, unfortunately, are not sense-disambiguated. Ideally, instead of "tiger", we would have tiger(2) indicating the animal meaning of "tiger" in the WordNet system, but computer vision data typically does not have sense specific labels. Further, while some words are either clearly correct ("tiger") or incorrect ("buildings"), some, like "cat", are in-between, and perhaps should be attributed an in-between score.

Providing vocabulary mapping at evaluation time supports a wide range of approaches, and in particular learning approaches that prefer the native vocabulary accompanying training data. To further see the advantages of the generality provided by mapping, consider some of the classes of approaches that one may want to evaluate on the *same* task. One approach is to gather a number of images for some ground truth entities and hand label the particular regions. A second approach is to use images with a ground truth entity somewhere in the image, but without distinguishing the background (e.g. Caltech 101). These two approaches are tailored to the ground truth vocabulary, and do not require vocabulary adjustment. A third approach is to learn the semantics of the backgrounds, regardless of whether they occur in the ground truth, using additional labels available with the images. This may be helpful as it can

push the loosely labeled data towards more tightly labeled data, and help disambiguate the visual information that is relevant to identifying the ground truth entity. Notice also that such a method may also allow the data to dictate the level of specificity that can be supported, and thus, for example, may demonstrate non-trivial behavior by labeling tigers as cats. Finally, all methods can choose to learn levels of specificity on the assumption that semantic hierarchy sometimes align with visual ones, by managing and/or augmenting their vocabularies using WordNet.

In what follows we will present in detail the process to compute the score matrices that implement proposed evaluation strategies. The reader may find it helpful to refer to the summary in Figure 7. To make it easier for others to use our scoring methodology, we have made a program which implements these computation available on-line (see Appendix B).

3.3 Ground truth vocabulary pre-processing

Due to the general and flexible data collection methodology described above, many ground truth labels have multiple labels. As a first pre-processing step we remove words that are ancestors of others in the label set. For example, if a ground truth region has both “cat” and “tiger”, then “cat” is dropped. We also remove synonyms, so that there is at most one word from any given WordNet synset (synonym set). Multiple labels are still possible after these pruning processes if there are multiple diverse entities in a region, reflecting a segmentation that was not fine enough. We retain these and later distribute the score available for the region equally among the multiple labels (§3.7).

3.4 Ground truth region weights

We need to compute localized scores with respect to the ground truth regions, but algorithms specify semantics relative to their own system of localization which may be regions from a segmentation, pixel grid blocs, or localized descriptors. Here we only provide details for localization using regions.

The first complication is that algorithms omit parts of the image for various reasons. For example, in the experiments below we only use up to the 16 largest segments from our machine based segmentation. As a second example, consider building an ROC curve for performance based on a threshold on labeling confidence. The regions which an algorithm refuses to label at a given confidence level similarly need to be excluded.

We consider that each ground truth region owns an equal share of the image semantics, but they only contribute to the scoring to the extent that they are covered by evaluation regions. Hence we begin by computing an initial weight (mass), $m_o(G)$, for each ground truth segment, G, which is simply the fraction of it that overlaps with all evaluation regions,

$$m_o(G) = \sum_R \frac{|R \cap G|}{|G|} \tag{1}$$

where R indexes over algorithm regions, and | | denotes the area of a region. Notice that some ground truth segments may become irrelevant by this process. We then normalize the ground truth region weights so that they sum to one for each image. The normalized weights, $m(G)$, implement the choice that each image contributes equally to the score. This has the side effect that similar regions can score differently in different images. On the other hand, without normalization, images with more ground truth regions contribute more to the score. Both alternatives are likely closer to ideal for different evaluation experiments. Here we exclusively use the image normalized region weights.

3.5 Inverse word frequency weights with level of specificity

The next step is to use the ground truth region weights to compute values of the ground truth words based on the frequency of occurrence. This is necessary to ensure that all words are equivalent without consulting visual information, which corresponds to the notion that all words give the same expected score if simply guessed. Further, the frequency counts need to be properly calibrated for levels of specificity.

To weight words with different degree of specificity, we first construct a semantic directed acyclic graph (DAG) derived from the WordNet hierarchy. WordNet can be viewed as a directed graph in which nodes are words and links are semantic relations between words. WordNet supports many semantic relations. We used four of them in the *nouns* category: *is-a*, *part-of*, *member-of* and *instance-of*. A word w_j is an *ancestor* of another word w_j if there is one path from w_i to w_j through one of these relations. Let W be the ground truth vocabulary. We construct a DAG by adding edges, $E(i, j)$, from $w_i \in W$ to $w_j \in W$ if w_j is the nearest ancestor of w_i among the words in W according to WordNet. Figure 8(b) shows an example of such a DAG with 7 words. Note that for any word $w_i \in W$ in the DAG, we can

populate a breadth first search (BFS) tree $T(w_i)$ that is rooted at w_i which encodes the shortest path from w_i to all of its ancestors (see Figure 8(c)).

To set the node weights, we construct the BFS tree $T(w_i)$ for each ground truth word, w_i, and then add $m(G)/l(G)$ for each ground truth region that has word w_i to all nodes in $T(w_i)$. Here $l(G)$ is the number of distinct labels associated with G. Once all words have been processed, we set the weight for each node to be the reciprocal of the sum of all the weight deposited at that node for all regions for all images. This means that the total amount of weight available to each entity over all the regions is the same. Notice that higher level nodes such as "cat", will receive significant weight due to more specific terms ("tiger", "lion") even if they occur infrequently themselves. Although most learning algorithms would avoid "cat" if it did not occur often in training, we assume that any algorithm can consult WordNet and consider "cat" instead of "tiger". Our approach assures that there is no advantage to doing so without intelligent processing. Finally, we remark that the node weights will be slightly dependent on the segmentation if parts of the image were excluded as this can affect $m(G)$. One simplification is to approximate the node weights for all segmentations assuming that segments are never excluded.

3.6 Scoring the vocabulary words

To score the match between a vocabulary word and a ground truth word, we consider the BFS trees of the two words. For a non-zero score, the vocabulary word must be in the BFS tree for the ground truth word, meaning that there must exist a WordNet path from the ground truth word to the WordNet root that includes the vocabulary word. For semantic range, the word match score, $s_w(w,l)$, is then simply the weight of the most specific common node of the vocabulary word, *w*, and the ground truth label, *l*. Thus if the vocabulary word is "tiger_2", and the ground truth word is "cat_1", then the score is the value of the node for "cat_1".

In the case that the vocabulary word is a sibling of the ground truth word, we argue that the score should be zero. It might seem reasonable that a sibling ("lion" for "tiger") should justify some reward. However, the potential gain for being correct for "lion", say, compared with "cat", has to be offset by the risk of being wrong. If "lion" was rewarded for being a sibling to "tiger", then it would be better to guess "lion" than "cat" despite a lack of evidence that the cat under consideration is a lion.

The same reasoning applies in the child case, but here we remark that such examples are a symptom that that ground truth data is not specific enough. As the data is refined, and more general labels are replaced by more specific ones, inaccuracies due to this problem will be removed. On the other hand, if the label is general because more specific semantics are not discernable, then a program that attempts to increase its score by "guessing" a more specific label should score less than the program that decides that the more general term is the best that can be done (i.e., agreeing with the human labelers).

By construction, this system for semantic range has the nice property that randomly guessing any word, at any level of specificity will lead to the same score. Thus labeling everything as "tiger" will have the same score as labeling everything as "entity". Thus there is no advantage to consulting WordNet, and better performance requires considering the image.

3.6.1 Scoring frequency correct

Scoring frequency correct also requires proper accounting of levels of specificity. If we ignore levels of specificity, then a perfect score might be achievable by simply labeling everything by "entity". Fortunately, we can easily modify the above methodology to score frequency correct. Instead of scoring word matches by the node weight of the most specific sense common to the two paths, we use the ratio of that node weight to the node weight for the ground truth term. Notice that if the vocabulary word is the ground truth word, then the score is one, and the method reduces to simply counting correct labels. If the algorithm chooses to use a less specific term, ("cat" for "tiger"), then the score for being correct is less, but the chances that the term is correct are greater. Importantly, there is no clear advantage to using WordNet to replace a more specific term with a more general one.

To further illustrate the effect of choosing a more general term over a more specific one, we compare the case of labeling all regions by either a ground truth word, w_i, or its parent, $p(w_i)$. We simplify the discussion by assuming that all ground truth regions have the same weight ($m(G) = \text{constant}$), which means that we can replace the sum of ground truth weights for a label, *l*, with counts of that label, $c(l)$.

In the case of frequency correct, the score for labeling a region with ground truth w_i with $p(w_i)$ is the ratio of the node weight for $p(w_i)$ to that for the ground truth word:

$$s_w(p(w_i), w_i) = \frac{n(p(w_i))}{n(w_i)} = \frac{c(w_i)}{\sum_{j \ni a = p(w_j)} c(w_j)} \quad (2)$$

The score achieved by labeling all regions by *a* is then:

$$\sum_{i \ni a = p(w_i)} c(w_i) s_w(p(w_i), w_i) = \frac{\sum_{i \ni a = p(w_i)} c(w_i)^2}{\sum_{j \ni a = p(w_j)} c(w_j)} \quad (3)$$

Alternatively, the score of being more specific, and labeling all regions by a particular w_i is:

$$c(w_i) s_w(w_i, w_i) = c(w_i) \quad (4)$$

Elementary considerations reveal that (3) never exceeds (4) evaluated for the w_i with largest $c(w_i)$. If the $c(w_i)$ are uniformly distributed, then (3) and (4) are equal. Equality is also approached at the other extreme when the maximum $c(w_i)$ proportionally dominates the overall count. This corresponds to the case that there is only one relevant ancestor, and the more specific and more general terms become equivalent. For distributions of $c(w_i)$ which are in-between the two extremes, there is some advantage to being more specific and choosing the most common w_i, which reflects the overall bias of the measure towards common terms.

3.6.2 Vocabularies without senses

So far we have assumed the ideal case that algorithm vocabularies are sense disambiguated. However, typical word prediction vocabularies, such as Corel™ keywords, are not sense specific. Thus we need to be able to compute the matching score to account for multiple senses in the ground truth vocabulary. In particular, our methodology should reflect that a sense-free, but otherwise correct word, should receive substantive score, but less score than the sense-specific form. Put differently, labeling a region with "bank" which has two main senses, is similar to labeling a region as "either bank_1 or bank_2". These are potentially very different semantic labels. We introduce an ambiguity factor for a label, *l*, *a(l)*, which penalizes an algorithm's unwillingness to specify the sense.

We set *a(l)* so that there is no advantage due to knowing WordNet or the global statistics of the ground truth data. Specifically, set *a(l)* is the node weight for *l* normalized by the sum of the node weights for all the ground-truth labels that share the same spelling. If there are two senses for a word, then

the penalty over the entire data set is ½ (both for semantic range and frequency correct), regardless of the relative abundance of the two senses in the ground truth. In particular, with this method, there is no advantage to guessing a more common sense for a word, even for frequency correct where guessing more common words is better.

We emphasize that all sense processing is relative to the ground truth vocabulary. Senses not in the ground truth are ignored. The senses in WordNet reflect, in rough order of sense number, common usage. However, a particular image corpus, such as Corel™, will have a limited subset of senses, with differing statistics. For example, the first WordNet sense for "tiger" is a kind of person, reflecting usage in the reference corpora used to develop WordNet. This sense does not occur in Corel™, but there are many examples of the second sense (animal).

Dealing with senses is partly motivated because we want to be able to measure the effect of including knowledge about senses on inferring semantics. For example, an algorithm can attempt to infer the sense based on training text using one of the many algorithms developed for that task (Agirre and Rigau, 1995; Gale et al., 1992; Karov and Edelman, 1998; Mihalcea and Moldovan, 1998; Yarowsky, 1995), as we have done for data from the Fine Arts Museum of San Francisco (Barnard et al., 2001). However, since word sense disambiguation is still an unsolved problem, and performance beyond that of assuming the most common sense is difficult to achieve (Traupman and Wilensky, 2003), such strategies need to developed in the context of a serviceable evaluation methodology.

3.7 Mapping algorithm localization to ground truth

We are now able to specify the score, $s_r(w,R)$, for each vocabulary word, *w*, for each region, R. We gather contributions from the labels of each ground truth region that intersects R, in proportion to the amount of intersection. This gives the semantics of R as specified by the ground truth. However, it does not address localization within R. For example, if R contains a bird and some surrounding sky, the value of "bird" goes up as the proportion of the region that is covered by the bird increases. Hence we need a second factor. Taken all the above into account, $s_r(w,R)$, is given by:

$$s_r(w,R) = \sum_G m(G)\left(\frac{1}{l(G)}\sum_{l\in G} a(l)s_w(w,l)\right)\cdot\frac{|R\cap G|}{|G|}\cdot\frac{|R\cap G|}{|R|} \qquad (5)$$

where $l(G)$ is the number of diverse labels for G. The inner average accounts for the possibility that G has more than one label. Figure 9 illustrates the computation with a simple example.

This approach reflects the notion that scoring should be as independent as possible of segmentation errors. For example, a large sky area should score the same, whether it is properly segmented as one region, or whether it is two regions due to segmentation error, and whose scores are additive. Notice that if an algorithm labels two adjacent regions similarly, then it could merge the regions and thus remove the segmentation error (Barnard et al., 2003b; Carbonetto et al., 2004). On the other hand, if only one part of the split region was correctly identified, then the score would reflect the lesser combined labeling and segmentation performance.

4 Evaluation of region labeling algorithms

Our data and methodology enables significantly better evaluation of region labeling algorithms than previously available. Because we are interested in large scale experiments, we consider only learning approaches which can be trained on images with associated text and which can then produce labels for regions. We again draw the distinction between algorithms which provide words for images as a whole (auto-annotation), and those that label image regions. Every algorithm that labels regions can also provide image annotation by combining the region result in some way. For example, in the annotation results below, we assume that each region has equal weight, and use the sum of the normalized word prediction vectors over the regions.

Interestingly, many of the algorithms for image annotation which do not support region labeling nonetheless use image regions as the carriers of semantics (Barnard and Forsyth, 2001; Jeon et al., 2003; Lavrenko et al., 2003). This makes sense because the compositional nature of images means that an image annotation word is specific to a localized entity. We expect that most image annotation methods that use regions can be easily modified to expose the region information that is implicitly used to explicitly produce region labels. However, in this work we restrict our attention to existing region labeling approaches, specifically variants of two very different approaches: generative multi-modal statistical models (Barnard et al., 2003a) and multiple instance learning (Andrews and Hofmann, 2004; Andrews et al., 2002a; Andrews et al., 2002b; Chen and Wang, 2004; Maron, 1998; Maron and Lozano-Perez, 1998;

Maron and Ratan, 1998; Tao and Scott, 2004; Tao et al., 2004a; Tao et al., 2004b; Zhang and Goldman, 2001).

4.1 Generative multi-modal models

We tested two generative multi-model statistical models, specifically the dependent and correspondence models with linear topology (no document level clustering) that we developed in earlier work (Barnard et al., 2003a). Here we assume that images and associated text are generated by choosing one or more concepts (latent factors), l, from a prior distribution, $P(l)$, and then by generating regions and associated words conditionally independent given the latent factors. Thus, the joint probability of an image region and a word can be expressed as

$$P(w,r) = \sum_l P(w \mid l) P(r \mid l) P(l) \tag{6}$$

where w denotes a word, r denotes a region, l indexes latent factors, $P(w \mid l)$ is a probability table over the words, and for the blob model, $P(r \mid l)$, we use a Gaussian distribution over features, with the somewhat naive assumption of diagonal covariance matrices being required to keep the number of parameters reasonable. For the experiments below we set the number of factors to be 2,000 which is roughly comparable, relative to the number of training images, to the 500 factors used previously.

The dependent and correspondence models differ in how generating words and regions jointly relates to generating words for the image as a whole. This is a critical issue because words for the image as a whole are all that is available during training. In the case of the dependent model we assume that the regions for an image provide a posterior over the latent factors which is then sampled for the words for the image. More specifically, the probability of observing a set of image words and regions, $W \cup R$, is given by:

$$P(W \cup R) = \prod_{r \in R} \left(\sum_l P(r \mid l) P(l) \right)^{\frac{\max(|R|)}{|R|}} \prod_{w \in W} \left(\sum_l P(w \mid l) P(l \mid R) \right)^{\frac{\max(|W|)}{|W|}} \tag{7}$$

where the exponents $\frac{\max(|R|)}{|R|}$ and $\frac{\max(|W|)}{|W|}$ adjust for the number of words and regions in the image, with the maximums taken over all training images. $P(l \mid R)$ is computed by

$$p(l \mid R) \propto \sum_{r \in R} P(l \mid r) = \sum_{r \in B} P(r \mid l) P(l) / P(r) \tag{8}$$

We train the model with the expectation maximization (EM) algorithm (Dempster et al., 1977), with the hidden factors responsible for each word and region being represented by missing values.

For the correspondence model we assume that regions and words are strictly emitted as pairs. Differing numbers of words and regions is addressed by assuming duplication of words or regions where required. The analogous expression to (7) is:

$$P(W \cup R) = \prod_{(r,w)\in W\cup R} \left(\sum_{l} P(r \mid l) P(w \mid l) P(l) \right)^{\frac{\max(|W\cup R|)}{|W\cup R|}} \tag{9}$$

Training this model is more difficult than in the dependent model because specifying that each word must be paired with a region means that computing the expectations for the missing values requires marginalizing over all pairings. Since this is impractical, we use graph matching (Jonker and Volgenant, 1987) to choose a maximally likely pairing inside the expectation step of the EM fitting.

4.2 Multiple instance learning

Multiple instance learning (MIL) has been applied to image annotation and categorization in some of the above mentioned papers, but we are not aware of results on region labeling. The connection to region labeling is quite explicit because in this approach an image is labeled with a word, *w*, if the image contains a region with word, *w*. Hence we consider multiple instance learning to be a region labeling approach, despite that fact that apparently it has not been used as such previously.

In the multiple instance learning paradigm, each sample (image) is considered to be a collection (bag) of instances (regions). Each instance is either a positive example or a negative example, and each bag is labeled as either a positive bag (contains at least one positive example), or a negative bag (contains all negative examples). To use multiple instance learning in the image annotation framework, each word is considered in turn as a category label, and a classifier is built to determine if a region instance is in that category. Thus a classifier needs to be built for every item in the vocabulary. This is very expensive with vocabularies with hundreds of items as is the case in our experiments. Experiments reported on so far have been on a substantively lesser scale, and it is not clear to what extent this approach is appropriate for large scale image annotation—hence our interest in including it in our experiments.

Because multiple instance learning treats each word independently, it ignores the relative frequency of the words in the vocabulary. This is potentially an advantage with the "semantic range" evaluation approach, but it is a significant handicap when the frequency of being correct is being scored. Further, while multiple instance learning methods are generally developed in the context of binary classification, both methods that we implemented can output a soft scoring. Soft scoring seems more sensible with our scoring methodology because it provides an opportunity for the algorithm to break ties. Thus for each multiple instance learning method, we consider four sub-variants: hard and soft weighting, either as is, or multiplied by the empirical distribution of the training vocabulary, serving as a prior. We denote these sub-variants by the suffixes, "HARD", "SOFT", "HE", and "SE".

We implemented the expectation maximization-diverse density (EM-DD) algorithm (Zhang and Goldman, 2001), and the mi-SVM algorithm (Andrews et al., 2002b). The EM-DD method simplifies and speeds up the original diverse density algorithm (DD) (Maron and Lozano-Perez, 1998). Assuming uniform priors, the diverse density of a point is the probability of the bag labels using that point as reference for modified distances to the instances in a bag which are combined to give estimated bag labels. The DD method estimates the optimal such point using gradient descent from multiple starting points which are typically the positive instances. The EM-DD algorithm iteratively attempts to achieve the same thing, but at each iteration uses the current guess of the optimal to produce estimates for the instances most responsible for the bag label, and uses that point to get more immediate estimate of the bag label. Interestingly, the reported results suggests that EM-DD is also more accurate than DD, likely due to avoiding local minima better.

The mi-SVM algorithm begins by labeling all instances in positive bags as positive instances, and builds a support vector machine (SVM) classifier to separate them from the instances in the negative bags which are assumed to be reliable negative instances. Instances in the positive bags that are classified as negative are re-labeled as such, subject to the constraint that there must be one positive instance left, and a new SVM classifier is then built.

Because of the large scale of our data, we restricted the number of positive and negative bags to 200, randomly chosen from the entire data set. When there were fewer than 200 positive examples (rare

words), we included random duplicates to bring the number up to 200. For the EM-DD algorithm we set the optimal threshold for classification by holding out 10% of the training data.

We implemented the mi-SVM algorithm with a linear kernel. Experiments on the same data as in the original mi-SVM paper suggested that increasing performance with non-linear kernels is difficult, and that it would be very expensive to find good kernels with cross-validation on data of the scale below. The mi-SVM algorithm iterates until there is no change in imputed labels. We did not exceed 100 iterations even if changes were still being detected. If the set of labels kept switching in a loop within the maximum number of iterations, we chose the model which gave the best accuracy on a held out data set. For our implementation we took advantage of the freely available libsvm software (version 2.8) (Chang and Lin). We used the facility for scaling the data before generating the models.

4.3 Experimental protocol

For our experiments we prepared four data sets. We began with 39,600 Corel™ images. For each of these images we have a small number of keywords (typically between three and five). We segmented the images with a modified version of normalized cuts (Gabbur, 2003; Shi and Malik., 2000). We extracted features similar to those used in (Barnard) representing color, texture, size, position, shape and color context (Barnard et al., 2003b). More specifically:

- 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 represented using the average and standard deviation of (r=R/(R+G+B), g=G/(R+G+B), S=(R+G+B)) over the region. We use this color space instead of RGB to reduce correlation among the three bands.
- Texture is represented using the average and variance of 16 filter responses. We use 4 difference of Gaussian filters with different sigmas, and two sets of 12 oriented filters, aligned in 30 degree increments, each one at a different scale. See (Shi and Malik., 2000) for additional details and references on this approach to texture.
- 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.
- Color context is represented by four colors each one representing the color of adjacent regions, restricted to four 90 degree wedges (Barnard et al., 2003b).

We excluded a few images due to problems with pre-processing. We withheld the 1,014 labeled images from training. We then constructed an easier subset (“restricted”) by removing images from the

CorelTM CD's that did not have any representative images in the 1,014 images[2]. We constructed vocabularies by limiting the keywords to those that occur 20 times in each of the data sets (50 for a second pair of data sets). Images that were left devoid of words were excluded. These 4 training sets are summarized in Table 1.

4.4 Performance measures

We used three evaluation measures, name image annotation performance and the two approaches to region labeling performance ("semantic range" and "frequency correct"). For annotation performance we use the key word prediction measure from previous work. Specifically, if there are M keywords for the image, we allow each algorithm to predict M words. We then simply record the percent correct.

To evaluate region labeling, we applied the method developed above (§3) to compute a score matrix for each image that specified the value of predicting each vocabulary word for each region provided by the segmentation algorithm. Since the algorithm vocabulary came from the Corel™ keywords as described in the previous sub-section, it was distinctly different than the sense-disambiguated ground-truth vocabulary. Thus both the localization mapping and vocabulary mapping processes were needed for this evaluation.

The region labeling algorithms produce weight vectors of sum one for each machine generated segment which express a preference for each word (e.g. a posterior probability distribution). From this we simply note the word with maximal weight, and use the score in the score matrix for the word / segment combination. A reasonable alternative would be to use the dot product of the weight vector with the score vector.

4.5 Results

The results for three measures on four data sets are provided in Tables 2 through 5. These results confirm our expectation that the MIL methods require modification when the task rewards performance on common entities. With our proposed modifications (soft classification, multiplication by empirical distribution), we observed much improved performance on image annotation and region labeling where frequency correct was counted. The results also confirm that if the task instead requires recognition independent of how common entities are (semantic range measure) the modification does not make sense.

The dependent translation model gives the best results for annotation and "frequency correct" region labeling measure, followed closely by EMDD-SE. We expect that if we provide the MIL methods the empirical distribution of the words over the regions (instead of images) in the training data, then they would do even better, but this information is not readily available. These experiments suggest that it may be worthwhile to approximate it. It makes sense that the translation model does well by this measure because it is the only approach which simultaneously learns region frequency statistics and how to recognize them based on features. Interestingly, the dependent translation method always performed better than the correspondence method. This is in contrast with earlier work that suggested that the correspondence model should be better at labeling regions.

The MIL methods performed well on the semantic range task, with EMDD-SOFT doing the best if the results from the four tests are averaged, followed by SVM-SOFT and the dependent translation model which are close. Interestingly, the SVM-SOFT is clearly the best on the R50 and R20 data sets, and relatively worse on the U50 and U20 data sets which have much more noise. Further experiments are required to establish if this is simply a tuning issue.

One high level question that we wanted to address was the extent to which the annotation score is a good proxy measure for region labeling performance. The results show that there is modest correlation in the case of frequency correct, but annotation performance is not at all indicative of semantic range performance as it is driven by common words.

4.6 Human validation of region labeling scores

In a final study we considered how indicative the measured performance is of human evaluations of the same task. Since manually quantifying region labeling on an absolute scale is difficult, we had participants simply choose between labelings of the same image as computed by different algorithms. We studied three forms of evaluation. In the first two, participants familiar with our system, attempted to make intuitive judgments of "semantic range" and "frequency correct". Evaluating "semantic range" using image pairs is difficult, but we attempted to judge when a labeling was better based on labeling accuracy of rare words in contrast to common words. In the third experiment, two participants ignorant of

our work were asked to simply judge which labeling was better, by whatever process they thought made sense. We expected that these participants would tend towards frequency correct which is what we found.

The results are presented in Table 6. We found complete agreement between the preferred algorithm for the particular task as determined by the automated evaluation methods and the human validation.

5 Discussion

Our experiments have gone significantly beyond previous work in two ways. First the data set size and the vocabularies were substantively larger than what has gone before. Second, the scale of evaluation, carried out on a region level for over 1,000 images, is also significantly larger than earlier efforts.

In our experiments we have found the MIL methods to be a very interesting alternative to the translation methodology, and that they have potential for excellent performance. However, we wish to emphasize that currently that performance comes at a *very large cost*. The MIL models are trained one word at a time at non-negligible cost, and our vocabularies have hundreds of words. In fact, we expended an order of magnitude more computational resources on the MIL methods compared with the translation methods, which took only a few hours to learn a model for the complete vocabulary. Thus we are investigating on how to adopt the ideas from MIL that work into a more scalable learning strategy.

The details of our results are naturally a function of implementation choices and our data. What is key is that our measurement infrastructure provides necessary feedback to improve each algorithm. Importantly, the results show that the performance on the three tasks is not simply linked. How an algorithm performs over the suite of three tasks gives substantive insight into what it does well and why, and this can be used to take a more guided approach to improving performance. For example, it seems clear that a blend of the ideas from the MIL and translation frameworks could increase performance over what we have measured so far.

The fact that annotation performance is not trivially a good proxy for recognition validates our assumption that careful region labeling is necessary to characterize performance. Since this is a labor intensive task, we wished to do so in a way that provides maximal benefit to the vision community. In particular, we have provided an infrastructure for using the labeled data to automatically evaluate a diverse range of recognition results. Our initial experiments have already provided some insight into

region labeling methods, suggesting that this kind of data is extremely useful. We look forward to hearing about other interesting applications of our infrastructure.

Acknowledgments

We acknowledge substantive help in this project. David Martin, Charless Fowlkes, and Jitendra Malik supplied the human level segmentations. The same group, together with Doron Tal, also supplied a version of the normalized cuts software, which was modified by Prasad Gabbur for the machine segmentations used in the experimental data set. Nikhil Shirahatti labeled a number of images. Finally, we are grateful to Lockheed Martin who funded the initial data gathering work and who graciously has let the data go into the public domain.

Appendix A: Labeling rules specific to humans

a) If the skin color is obvious in the segment classify human as white, black or Asian.
b) If the origin or function is obvious by the clothing label as the clothing suggest (e.g. eskimo, south american, arab, soldier, skier, cowboy, etc.)
c) If a piece of clothing is selected in addition to human label the part as the clothing (e.g. dress) or the clothing part (e.g. sleeve).
d) If the human is wearing a special gear in the segment, label it (e.g. headdress, gloves, weapon, etc.)
e) If, in the segment the hair color is obvious, label it ("blond hair", "black hair", "brown hair", "red hair", "white hair")
f) Connect body part with human label (e.g. boy, boy_face) but not clothes (e.g. boy, shirt).
g) While labeling parts of the human face, it is not required to mention skin color for parts such as eye, nose, mouth etc. (e.g. boy, boy_eye).

Appendix B: Online data and infrastructure

The data and associated infrastructure is available online (kobus.ca//research/data/IJCV).For each of the 1014 images we have made available the ground truth segmentations (courtesy of the UC Berkeley segmentation group) and a text file with our labels. We also have made available a Java program for labeling which can be used to improve our labelings and label additional images. To label a new image, one will first need to create a ground truth segmentation. For this we recommend the segmentation tool

from the UC Berkeley computer vision group (http.cs.berkeley.edu//projects/vision/grouping/segbench) which writes segmentations in the format that our program reads.

We have also made available a program which builds a scoring matrix from machine segmentations and an arbitrary vocabulary. The scoring matrices have one row per image region, and one column for each vocabulary word. The matrix entries are the scores deserved for predicting that word for that region. The program supports scoring both for semantic range and frequency correct. We also provide an generic scoring matrix that encodes a general score for matching of each vocabulary word with each ground truth word in the context of a particular image, independent of segmentation. Finally, we provide the feature vectors for the four data sets R20, R50, U20, and U50.

Notes

1. Available from the Sowerby Research Center, British Aerospace, FPC 267, PO Box 5, Filton, Bristol BS12 7NE, England.

2. The Corel™ data is organized into CD's, each with 100 images that tend to be semantically related.

References

Agarwal, S., Awan, A. and Roth, D., The UIUC Image Database for Car Detection. Available from http://l2r.cs.uiuc.edu/~cogcomp/Data/Car/.

Agarwal, S., Awan, A. and Roth, D., 2004. Learning to detect objects in images via a sparse, part-based representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(11): 1475--1490.

Agirre, E. and Rigau, G., 1995. A proposal for word sense disambiguation using conceptual distance, 1st International Conference on Recent Advances in Natural Language Processing, Velingrad.

Andrews, S. and Hofmann, T., 2004. Multiple Instance Learning via Disjunctive Programming Boosting, Advances in Neural Information Processing Systems (NIPS 16).

Andrews, S., Hofmann, T. and Tsochantaridis, I., 2002a. Multiple Instance Learning With Generalized Support Vector Machines, AAAI.

Andrews, S., Tsochantaridis, I. and Hofmann, T., 2002b. Support Vector Machines for Multiple-Instance Learning, Advances in Neural Information Processing Systems, 15, Vancouver, BC.

Barnard, K., Data for Computer Vision and Computational Colour Vision. Available from kobus.ca/research/data.

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

Barnard, K. et al., 2003a. Matching Words and Pictures. Journal of Machine Learning Research, 3: 1107-1135.

Barnard, K., Duygulu, P., Raghavendra, K.G., Gabbur, P. and Forsyth, D., 2003b. The effects of segmentation and feature choice in a translation model of object recognition, IEEE Conference on Computer Vision and Pattern Recognition, Madison, WI, pp. II:675-682.

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

Berg, A.C., Berg, T.L. and Malik, J., 2005. Shape Matching and Object Recognition using Low Distortion Correspondence, CVPR.

Carbonetto, P., Freitas, N.d. and Barnard, K., 2004. A Statistical Model for General Contextual Object Recognition, European Conference on Computer Vision.

Chang, C.-C. and Lin, C.-J., LIBSVM -- A Library for Support Vector Machines. Available from http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

Chen, Y. and Wang, J.Z., 2004. Image Categorization by Learning and Reasoning with Regions. Journal of Machine Learning Research, 5: 913-939.

Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological), 39(1): 1-38.

Fei-Fei, L., Fergus, R. and Perona, P., 2004. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Workshop on Generative-Model Based Vision, Washington, DC.

Fellbaum, C., Miller, P.G.A., Tengi, R. and Wakefield, P., WordNet - a Lexical Database for English. Available from http://www.cogsci.princeton.edu/~wn.

Fergus, R. and Perona, P., The Caltech Database. Available from http://www.vision.caltech.edu/html-files/archive.html.

Fergus, R., Perona, P. and Zisserman, A., 2003. Object Class Recognition by Unsupervised Scale-Invariant Learning, IEEE Conference on Computer Vision and Pattern Recognition, Madison, WI.

Gabbur, P., 2003. Quantitative evaluation of feature sets, segmentation algorithms, and color constancy algorithms using word prediction. Masters Thesis, University of Arizona, Tucson, AZ.

Gale, W., Church, K. and Yarowsky, D., 1992. One Sense Per Discourse, DARPA Workshop on Speech and Natural Language, New York, pp. 233-237.

Jeon, J., Lavrenko, V. and Manmatha, R., 2003. Automatic Image Annotation and Retrieval using Cross-Media Relevance Models, SIGIR.

Jonker, R. and Volgenant, A., 1987. A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems. Computing, 38: 325-340.

Karov, Y. and Edelman, S., 1998. Similarity-based word sense disambiguation. Computational Linguistics, 24(1): 41-59.

Lavrenko, V., Manmatha, R. and Jeon, J., 2003. A Model for Learning the Semantics of Pictures, NIPS.

Leibe, B. and Schiele, B., The TU Darmstadt Database. Available from http://www.vision.ethz.ch/leibe/data/.

Maron, O., 1998. Learning from Ambiguity. Ph.D. Thesis, Massachusetts Institute of Technology.

Maron, O. and Lozano-Perez, T., 1998. A framework for multiple-instance learning, Neural Information Processing Systems. MIT Press.

Maron, O. and Ratan, A.L., 1998. Multiple-Instance Learning for Natural Scene Classification, The Fifteenth International Conference on Machine Learning.

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

Mihalcea, R. and Moldovan, D., 1998. Word sense disambiguation based on semantic density, COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems, Montreal.

Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K.J., 1990. Introduction to WordNet: an on-line lexical database. International Journal of Lexicography, 3(4): 235 - 244.

Opelt, A. and Pinz, A., TU Graz-02 Database. Available from http://www.emt.tugraz.at/~pinz/data/GRAZ_02/.

Shi, J. and Malik., J., 2000. Normalized Cuts and Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(9): 888-905.

Tao, Q. and Scott, S., 2004. A Faster Algorithm for Generalized Multiple-instance Learning, Seventeenth Annual FLAIRS Conference, Miami Beach, Florida, pp. 550-555.

Tao, Q., Scott, S., Vinodchandran, N.V., Osugi, T.T. and Mueller, B., 2004a. An Extended Kernel for Generalized Multiple-Instance Learning, IEEE International Conference on Tools with Artificial Intelligence.

Tao, Q., Scott, S.D. and Vinodchandran, N.V., 2004b. SVM-Based Generalized Multiple-Instance Learning via Approximate Box Counting, International Conference on Machine Learning, Banff, Alberta, Canada, pp. 779-806.

Torralba, A., Murphy, K.P. and Freeman, W.T., The MIT-CSAIL Database of Objects and Scenes. Available from http://web.mit.edu/torralba/www/database.html.

Torralba, A., Murphy, K.P. and Freeman, W.T., 2004. Sharing features: efficient boosting procedures for multiclass object detection, IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, pp. II:762--769.

Traupman, J. and Wilensky, R., 2003. Experiments in Improving Unsupervised Word Sense Disambiguation, Computer Science Division, University of California Berkeley.

Vivarelli, F. and Williams, C.K.I., 1997. Using Bayesian neural networks to classify segmented images, IEE International Conference on Artificial Neural Networks.

Weber, M., Welling, M. and Perona, P., 2000. Unsupervised Learning of Models for Recognition. In: D. Vernon (Editor), 6th European Conference on Computer Vision, pp. 18-32.

Yarowsky, D., 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods, 33rd Conference on Applied Natural Language Processing. ACL, Cambridge.

Zhang, Q. and Goldman, S.A., 2001. EM-DD: An improved multiple-instance learning technique, Neural Information Processing Systems.

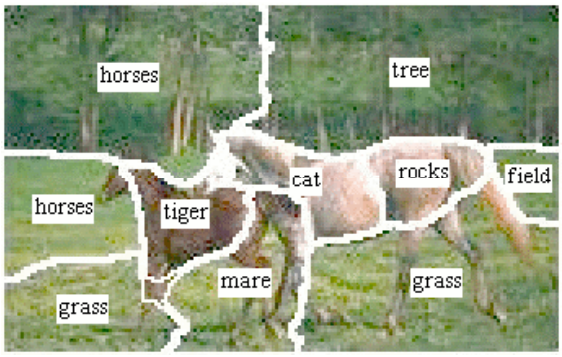

Figure 1. An example of region labeling that gives relatively good annotation results despite several obvious incorrect correspondences. Both "horses" and "mares" are good words for the image, but neither are correctly placed. In the training data used for this experiments, horses are and grass of the above color and texture co-occur often, and are only rarely a part, making it difficult to learn the difference. Higher throughput localization performance measurement will help characterize our systems.

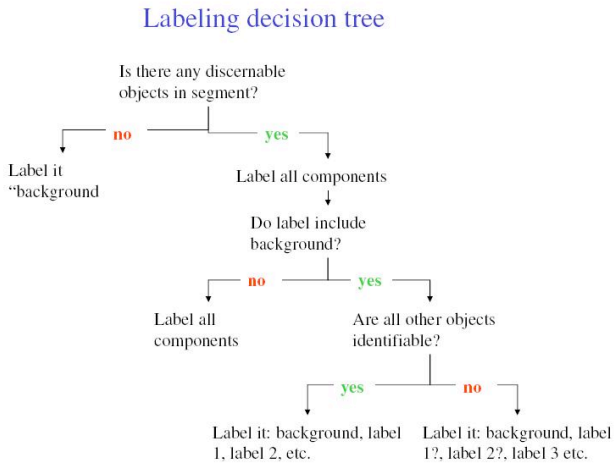

Figure 2. A decision tree that specifies how to handle segments that may include multiple objects and/or background that does not have any identifiable objects in it.

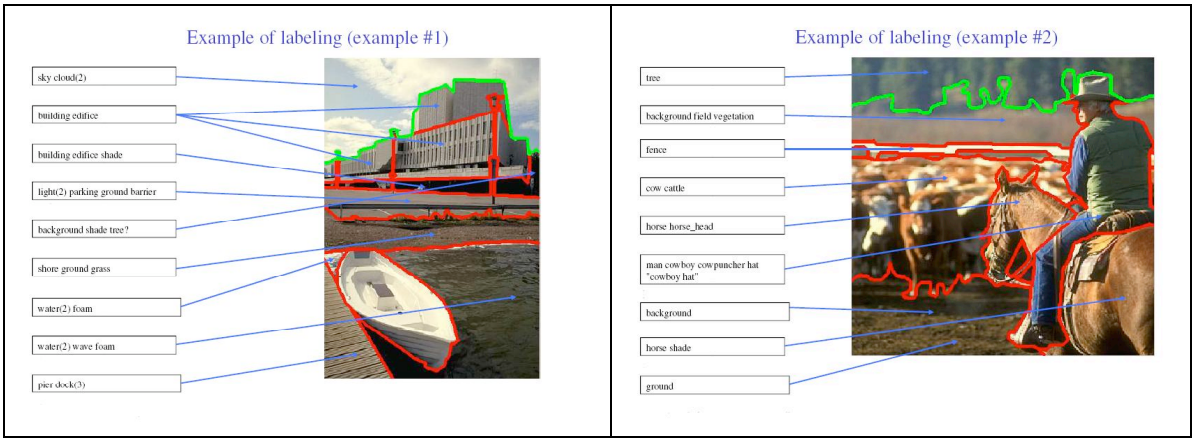

Figure 3. Examples of image labeling that are consistent with the labeling rules described in the text.

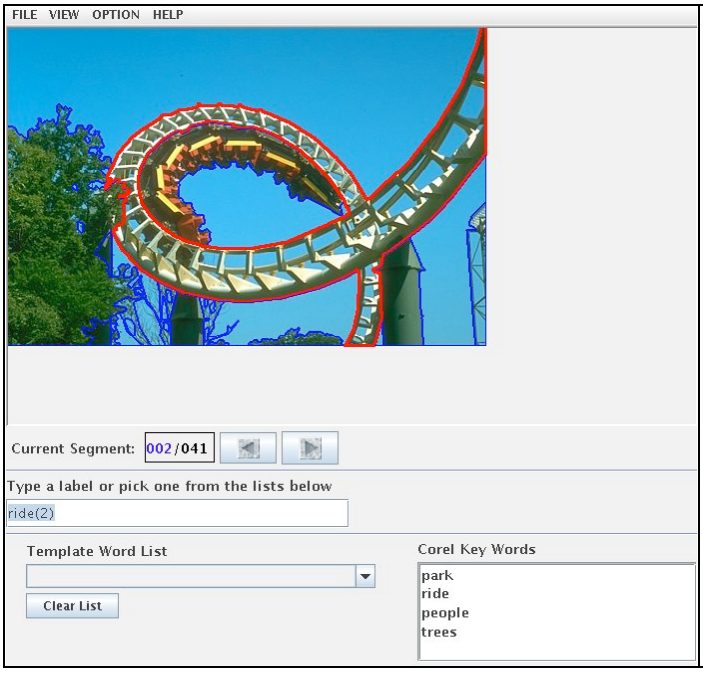

Figure 4. A screen shot of the labeling tool. We have made the tool available so that others can improve and extend the data set. Features that make the labeling process more efficient include a list of words from a template for labeling similar genres of images, a list of any image key words, and the ability to reload an earlier labeling result and continue previous work, check existing labelings, or improve the labeling.

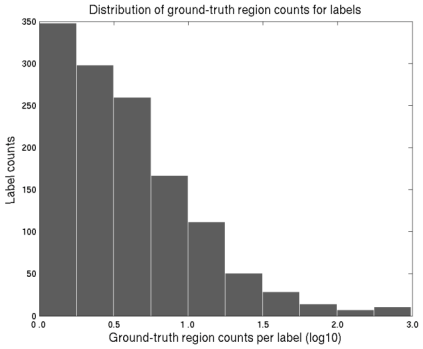

Figure 5. The distribution of the number of examples for the 1297 labels in the ground-truth. As one would expect, there are many labels with sparse representation, and only a moderate number with excellent representation. Nonetheless, the data set is large enough that there are many labels with enough examples to be useful as training data.

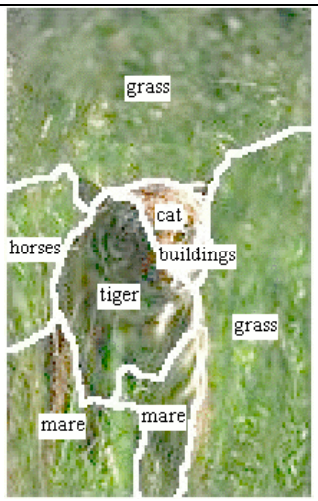

Figure 6. An example of an image labeling effort that must be mapped into the ground truth localization and semantic spaces. The segmentation used by the algorithm breaks up semantic regions leading to the questions such as how the scores for the two regions labeled "grass" should be weighted. Intuitively the region labeled "cat" should receive some score, but not as much as "tiger". A methodology for dealing with these issues is developed in the text.

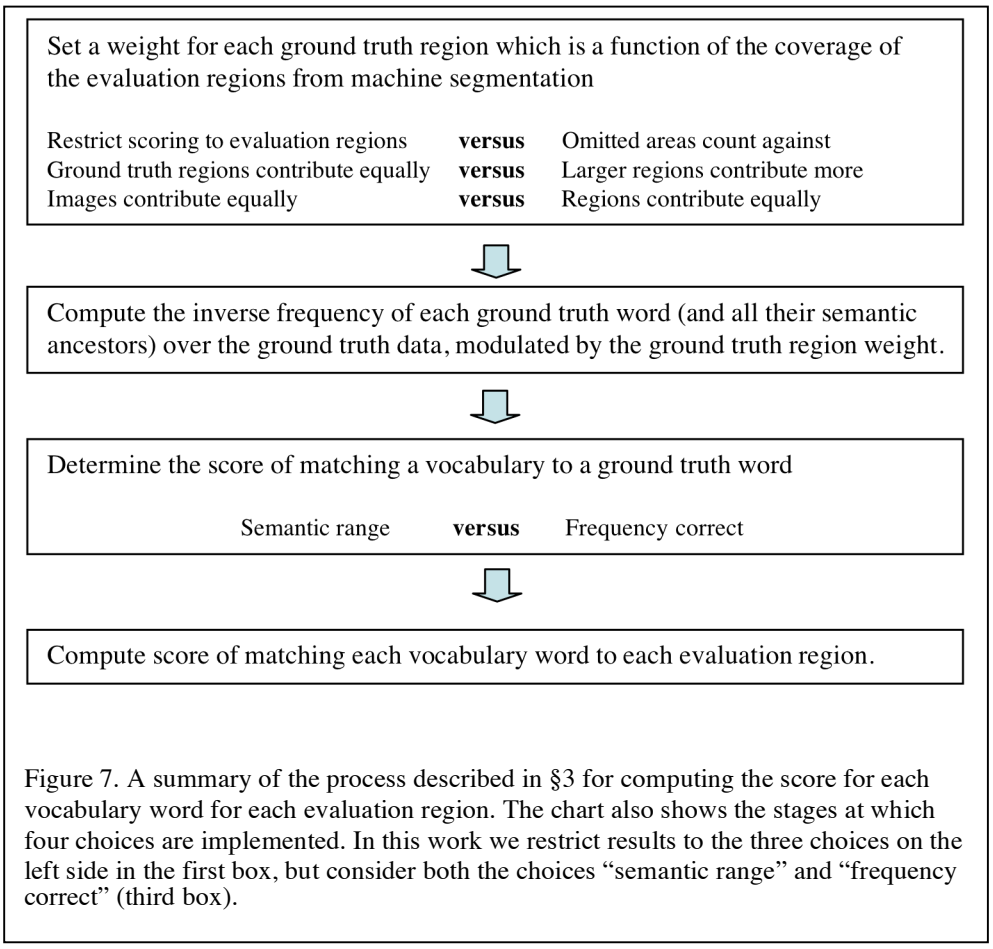

Figure 7. A summary of the process described in §3 for computing the score for each vocabulary word for each evaluation region. The chart also shows the stages at which four choices are implemented. In this work we restrict results to the three choices on the left side in the first box, but consider both the choices "semantic range" and "frequency correct" (third box).

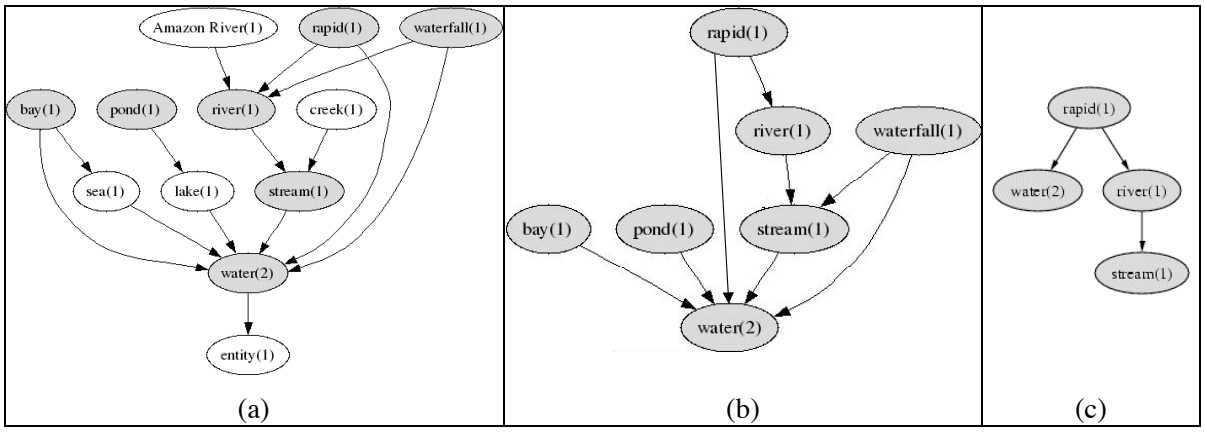

Figure 8. The use of WordNet to establish semantic scoring for related words. The WordNet hierarchy can be viewed as a directed graph in which words are nodes and edges represent different semantic relationships between words such as hyponym, holonym, has-member and has-instance. Figure a) shows a subgraph of the hierarchy for nouns. The shaded words are in the ground truth. The number in the parenthesis is the sense of the word set by WordNet. Figure b) shows the directed acyclic graph (DAG) constructed over the ground truth words using the approach described in the paper. Note that a word (e.g. "rapid") can have more than one path to some of its ancestors (e.g. "water") in the DAG. Thus for each word we construct a breadth first search (BFS) tree which encodes the shortest path to related words. Figure c) shows the BFS tree populated from the DAG for the "rapid".

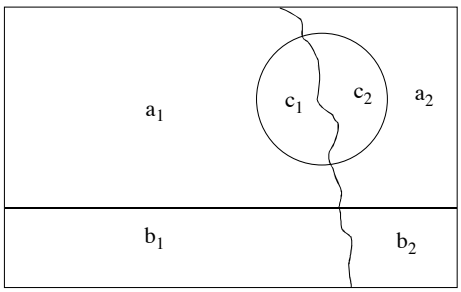

Figure 9. Illustration of the method for weighting the scores for labels from machine segmentations against the ground truth label for a ground truth region. The squiggly line represents an edge from a machine segmentation which divides each of three ground truth regions (a,b, c), with labels (A,B,C) respectively, into two parts. We assume that (A, B, C) is the entire vocabulary, that $s_w(x,y)$ is one if $x=y$ and zero otherwise, that the ground truth region weights, $w(G)$, are all one, and that $|c_1| = |c_2|$. Ignoring the localization factor, both regions score the same for C, but region one scores more for A and B. The localization factor increases the score of C in region two, because C covers relatively more of it. Because region one has more of the image's semantic space, measured by the effective number of ground truth regions covered, region one has more effect on the overall score.

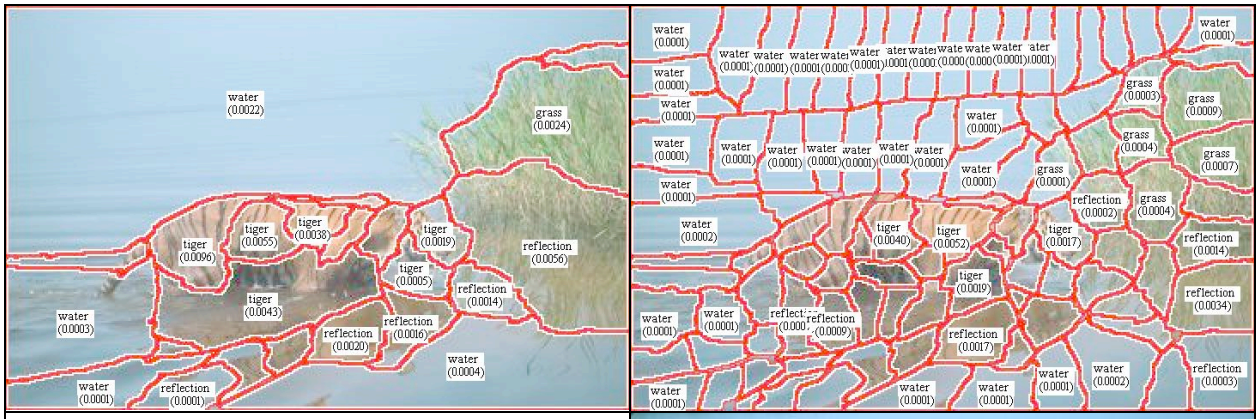

Figure 10. Illustration of mapping of the ground truth data onto experiment specific data. The regions are labeled with the word that would receive maximal score (in parenthesis) under the semantic range scoring method. The machine produced segments (top) for two different segmentation methods are scored as a blend of the underlying ground truth segments (right). The blend is not simple because the vocabulary words need to be connected to one or more ground truth words which may be scored according to how frequent they are. The scale of the scores is small because the combined score of any entity (for example water_1) over the 1002 ground truth images is set to sum to one.

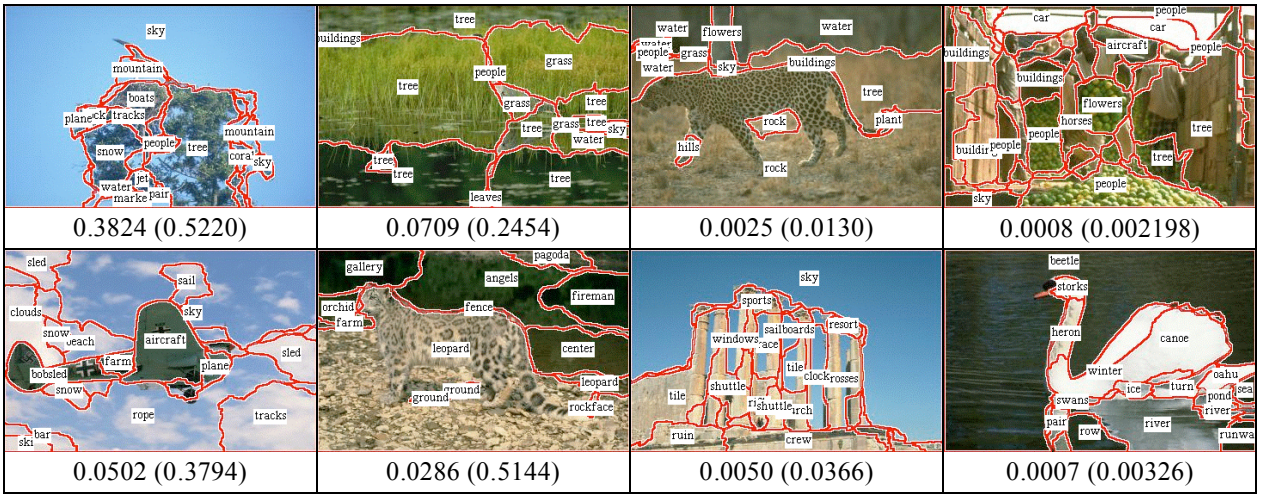

Figure 11. Illustration of region labeling with associated scores for the R20 data set. Raw scores decrease left to right. The top row shows example from the dependent translation model using frequency correct scoring. The bottom row illustrates EMDD-SOFT with semantic range scoring. The scores are highly dependent on the quality of the segmentation. The scores relative to the best possible score given the segmentations are provided in parentheses. Naturally, the tend to follow the raw scores, but the leopard image (bottom row, second from left) is an exception. The segmentation mistake substantively lowers the raw score, but the score relative to the particular segmentation is good.

Data set	Minimum number of times each vocabulary word is used	Vocabulary size	Number of images
R20	20	509	26,078
R50	50	272	25,985
U20	20	656	37,337
U50	50	383	37,257

Table 1. A summary of the training data sets constructed for the experiments. In all cases the test set was drawn from the 1014 human evaluated images. (We used all with R50 and U20; two were omitted with R20 and U20). In all cases, algorithms expressed semantic prediction with respect to the training vocabularies that were then evaluated using a mapping into ground truth vocabulary.

Algorithm	Annotation (keyword prediction)	Region labeling semantic range performance (relative to the score of 0.0707 for mapped human labels)	Frequency that region semantics are correctly identified (relative to the score of 0.291 for mapped human labels)
Test data empirical distribution	0.207	0.0085	0.124
Dependent translation model	0.292	0.0255	0.163
Correspondence translation model	0.251	0.0212	0.116 (–)
mi-SVM (hard)	0.021 (–)	0.0099	0.007 (–)
mi-SVM (soft)	0.040 (–)	0.0282	0.027 (–)
mi-SVM (hard, empirical as prior)	0.115 (–)	0.0113	0.094 (–)
mi-SVM (soft, empirical as prior)	0.256	0.0127	0.120 (–)
EMDD (hard)	0.019 (–)	0.0113	0.031 (–)
EMDD (soft)	0.084 (–)	0.0279	0.032 (–)
EMDD (hard, empirical as prior)	0.137 (–)	0.0155	0.057 (–)
EMDD (soft, empirical as prior)	0.230	0.0113	0.094 (–)

Table 2. The results for the R50 data set. The maximum in each column is identified by a heavy border. All numbers are an average over the 1012 human labeled images held out from training. The results for the translation models are further the average over 5 training runs with different random initializations. The annotation is how well algorithms predict the 3-5 keywords for each image. Because some of the Corel™ words occur repeatedly, it is difficult to do better than the empirical distribution on tasks which reward frequency correct (first and third tasks). Algorithms that perform worse than that baseline are marked with a (–).

The second column is the semantic range performance, relative to that of the score that would be achieved using the appropriate mapped manual labeling (0.0707) which is the best score possible given our machine segmentations and training vocabulary. The fact that the numbers are far less than unity reflects the fact that general object recognition is an unsolved problem. The third column is the frequency that a semantic entity is correct, again relative to that of the appropriate mapped manual labeling (0.291).

We shade the results for variants of mi-SVM and EMDD which we assumed would be less appropriate for the task being measured. In particular, we expected soft classification to do better than hard classification, and we expected that using the empirical distribution would be necessary for good results in the case of annotation and frequency correct, and that it would be a liability when applied to semantic range performance. In all four experiments, the variants that we assumed made most sense gave the best result.

Algorithm	Annotation (keyword prediction)	Region labeling semantic range performance (relative to the score of 0.105 for mapped human labels)	Frequency region semantics are correctly identified (relative to the score of 0.312 for mapped human labels)
Test data empirical distribution	0.210	0.0057	0.116
Dependent translation model	0.277	0.0181	0.155
Correspondence translation model	0.241	0.0152	0.116
mi-SVM (hard)	0.019 (–)	0.0038 (-)	0.006 (–)
mi-SVM (soft)	0.041 (–)	0.0257	0.038 (–)
mi-SVM (hard, empirical as prior)	0.105 (–)	0.0076	0.099 (–)
mi-SVM (soft, empirical as prior)	0.241	0.0124	0.131 (–)
EMDD (hard)	0.016 (–)	0.0086	0.013 (–)
EMDD (soft)	0.058 (–)	0.0228	0.034 (–)
EMDD (hard, empirical as prior)	0.155 (-)	0.0124	0.114 (–)
EMDD (soft, empirical as prior)	0.251	0.0114	0.156

Table 3. The results for the R20 data set. This data set is similar to R50, but the vocabulary is larger (509 versus 272), since every word only needs to occur only 20 times in the training data instead of 50. See the caption for Table 2 for details.

Algorithm	Annotation (keyword prediction)	Region labeling semantic range performance (relative to the score of 0.0819 for mapped human labels)	Frequency region semantics are correctly identified (relative to the score of 0.297 for mapped human labels)
Test data empirical distribution	0.222	0.0073	0.123
Dependent translation model	0.280	0.0207	0.168
Correspondence translation model	0.238	0.0147	0.102 (–)
mi-SVM (hard)	0.014 (–)	0.0037 (–)	0.009 (–)
mi-SVM (soft)	0.028 (–)	0.0147	0.022 (–)
mi-SVM (hard, empirical as prior)	0.089 (–)	0.0098	0.089 (–)
mi-SVM (soft, empirical as prior)	0.237	0.0122	0.132 (–)
EMDD (hard)	0.016 (–)	0.0098	0.021 (–)
EMDD (soft)	0.031 (–)	0.0256	0.069 (–)
EMDD (hard, empirical as prior)	0.142	0.0158	0.128
EMDD (soft, empirical as prior)	0.223	0.0147	0.160

Table 4. The results for the U50 data set. This data set is similar to R50, but with 40% more training images, most of which are completely unlike the test images. See the caption for Table 2 for further details.

Algorithm	Annotation (keyword prediction)	Region labeling semantic range performance (relative to the score of 0.120 for mapped human labels)	Frequency region semantics are correctly identified (relative to the score of 0.317 for mapped human labels)
Test data empirical distribution	0.222	0.0050	0.114
Dependent translation model	0.260	0.0134	0.145
Correspondence translation model	0.230	0.0100	0.109 (–)
mi-SVM (hard)	0.015 (–)	0.0033 (–)	0.004 (–)
mi-SVM (soft)	0.036 (–)	0.0117	0.016 (–)
mi-SVM (hard, empirical as prior)	0.104 (–)	0.0067	0.088 (–)
mi-SVM (soft, empirical as prior)	0.234	0.0092	0.137
EMDD (hard)	0.014 (–)	0.0050	0.008 (–)
EMDD (soft)	0.048 (–)	0.0175	0.049 (–)
EMDD (hard, empirical as prior)	0.164 (–)	0.0117	0.115
EMDD (soft, empirical as prior)	0.223	0.0100	0.151

Table 5. The results for the U20 data set. This data set is similar to R20, but with 40% more training images, most of which are completely unlike the test images. See the caption for Table 2 for further details.

Algorithms compared	Range comparison	Frequency comparison	Generic comparison
EMDD (soft) / COR-trans	29% / 16% (0.018 / 0.010)	12% / 33% (0.05 / 0.11)	27% / 45%
EMDD (soft with prior) / COR-trans	Poor performance in both (hard to compare)	35% / 12% (0.15 / 0.11)	39% / 17%
EMD (soft) / DEP-trans	28% / 14% (0.018 / 0.013)	15% / 39% (0.05 / 0.15)	Data not collected

Table 6. Results of human comparisons on selected algorithm pairs for the U20 data set. For each pair, we show the percentage of region labelings out of 500 examples that were preferred for each of the two algorithms, with the remaining labelings being relatively equal. Range comparison is not very natural in the context of single image pair comparisons, but the raters were aware that some words like "sky" and "water" are both common and easy to get right, and they focused their attention on more interesting labels. The generic comparison was undertaken by two raters who did not know any details of the study, and who were not given any instructions on how to compare labelings. Discussions after that fact suggested that they mostly rated on the basis of frequency correct, which is consistent with their scores. The comparable automated results (rounded) from Table 5 are provided in parenthesis. It is clear that there is good agreement between our evaluation methodology and human judgment on pair-wise comparisons.