Evaluation Strategies for Image Understanding and Retrieval

Keiji Yanai
Dept. of Computer Science,
The University of
Electro-Communications
1-5-1 Chofugaoka, Chofu-shi,
Tokyo, 182-8585 JAPAN
yanai@cs.uec.ac.jp

Nikhil V. Shirahatti
Prasad Gabbur
Electrical and Computer
Engineering Department,
University of Arizona
Tucson, AZ, 85721 USA
shirahatti@gmail.com
pgsangam@ece.arizona.edu

Kobus Barnard
Computer Science
Department,
University of Arizona
Tucson, AZ, 85721 USA
kobus@cs.arizona.edu

ABSTRACT

We address evaluation of image understanding and retrieval large scale image data in the context of three evaluation projects. The first project is a comprehensive strategy for evaluating image retrieval algorithms and provides an open reference data set for doing so. The second project develops word prediction as a semantically relevant evaluation strategy, and applies it to the evaluation of of image processing methods for semantic image analysis. The third project evaluates words for suitability of their visual properties for use in an image annotation framework.

Categories and Subject Descriptors

H5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems

General Terms

Algorithms, Design, Human Factors, Experimentation

Keywords

Evaluation, Image Retrieval, Image Recognition

1. INTRODUCTION

Automated methods for browsing and searching large image data sets promise instant access to vast amounts of visual information. However, to provide such tools we need to develop much more effective methods for inferring semantics from visual data. To move forward we need to more precisely understand what it is that we are trying to achieve, which naturally links to understanding how to measure it. This thinking has lead us to three research projects related to the evaluation of image understanding and retrieval. The linking theme is that we need to measure performance on a true semantic level, reflecting the users' needs. Strategies for evaluation include grounding retrieval in human evaluation (first project), and using word prediction as a measure for semantic understanding that can be applied on a large scale (second project). In our third project we consider the notion that evaluation must be done in the scope of semantics which can be linked to visual features, and we provide a method for mining large datasets for words that have mutual information with visual features. This helps large scale evaluation and algorithm development because we exploiting larger data sets becomes more tractable once the non-visual words have been pruned.

In §2 we present a comprehensive strategy for evaluating image retrieval algorithms and provides an open reference data set for it [43]. Because automated image retrieval is only meaningful in its service to people, performance characterization must be grounded in human evaluation. Thus we have collected a large data set of human evaluations of retrieval results, both for query by image example and query by text. The data is independent of any particular image retrieval algorithm and can be used to evaluate and compare many such algorithms without further data collection. We provide the data and calibration software to the public (`http://kobus.ca/research/data/`). We develop and validate methods for generating sensible evaluation data, calibrating for disparate evaluators, mapping image retrieval system scores to the human evaluation results, and comparing retrieval systems. The experimental results show how annotation and retrieval performance are linked as well as comparison of the several existing image retrieval algorithms.

In §3 we use word prediction performance to evaluate low and mid level vision processes such as segmentation and feature extraction [6]. We do this in the framework of an image understanding approach where the task is defined as predicting words for either image as a whole (auto-annotation), or specific image regions (recognition) [7, 17, 5]. In particular, we propose using word prediction performance as a task oriented evaluation measure for lower level algorithms. This approach makes possible large scale experiments linked to inferring semantics In this paper we provide representative

MIR'05, November 10–11, 2005, Singapore.

results for three different segmentation algorithms and several feature sets.

In §4 we present a method for the evaluation of visual properties of words [49]. Given our interest in large scale scale linking of image data to semantic entities represented by words from a very large vocabulary, a natural consideration is the extent to which a word relates to visual properties at all. We propose "image region entropy" which is a measure of "visualness" of concepts, that is, what extent concepts have visual characteristics. If a word has limited mutual information with image features that we can measure, then it is not a good candidate for image annotation. Our method performs probabilistic region selection for images which are labeled as concept "X" or "non-X", and computes an entropy measure which represents "visualness" of concepts associated with words. In our experiments, we collected about forty thousand images from the World-Wide Web using the Google Image Search for 150 concepts. We examined which concepts are suitable for annotation of image contents.

2. EVALUATION OF IMAGE RETRIEVAL ALGORITHMS

The problem of automatically retrieving desired images from a large, often unstructured data set has attracted much attention in the research community [22, 44, 20, 41, 29, 10, 48, 15, 44, 31, 36]. The task is difficult and tightly connected to computer vision because users are interested in the semantics of the retrieved images [19, 18, 3, 32, 21].

These studies confirm that current image retrieval methods are well off the required mark. We argue that moving forward will require quantifying real performance, and that the image retrieval community will be well served by an appropriate evaluation process and reference data set. Thus we have made our data and software available on-line (`http://kobus.ca/research/data/`).

Automated image retrieval is only meaningful in its service to human users, and thus performance must be grounded in direct human evaluations. Our approach is to evaluate query-result pairs for both query by image example and query by text. By focusing only on the input and output, such data is applicable to any image retrieval method.

Often evaluation of image retrieval is focused on results obtained with a specific instance of a specific algorithm. With this approach, changes to the algorithm require additional human evaluation, which is expensive. More automatic methods typically involve having sets of images tagged with high level concepts (e.g., *sky*, *grass*), and retrieval is evaluated based on those labels [45, 46, 47], making performance evaluation similar to that in text retrieval [39]. The Benchatholon project proposes providing much more detailed and publicly available keywords of images using a controlled vocabulary [23, 37, 1]. A problem with both these approaches is that they are only indirectly connected to the task that they are trying to measure. For example, there is an implicit assumption that a person seeking an image like one labeled *grass* will be content with all the images labeled *grass* and none of the ones not labeled *grass*. While we do not reject this hypothesis outright, image retrieval evaluations need to be grounded on tasks closer to what end-users do, hence this work. Our results can be used to calibrate these less expensive measures.

2.1 Developing a reference data set

We set up human retrieval evaluation experiments to gather grounded data for two tasks: query by image and query by text paradigms. For the query by image paradigm we present the user with one query image and four result images. The selection of the result images is discussed in detail in the subsequent paragraph. The participant was asked to score each of the four result images on a scale of 1 to 5, with 1 being a poor match and 5 being a good match. We provided an additional choice of *undecided* (ignored) so that participants could move onto the next example without spending too much time on ones they find hard to evaluate. Participants were given very little in the way of guidelines for making their selection. For the second interface, we presented the participant with a text query and a corresponding result image. They rated the match by selecting a score from 1-9 or *undecided*.

Avoiding too many negative matches. The main difficulty in setting up such an experiment is choosing query-result pairs. If they were randomly generated then nearly all the matches would be judged *poor match*. Ideally, we would like roughly a uniform distribution of the responses of the evaluations (excluding *undecided* where fewer is always better).

The main idea is to use existing image retrieval systems to help bias the selection process to get more uniform responses. Doing so may put us at risk for introducing unwanted biases in the test set due to some poorly characterized property of the retrieval system. While we do not expect significant problems, we guard against this by using four very different image retrieval processes. Each one is used for the selection of one of the four result images, randomly permuted for each query.

The second issue is how to use the retrieval systems to improve the sampling. Initially, we know very little about the relationship between retrieval results and human evaluation results. However, trial and error revealed that choosing images with probability proportional to the negative fifth power of the rank gave a serviceable starting point. This can be improved once some data has been collected, as our approach revolves around estimating the mapping from computer scores to human scores. In §2.4.2 we present results which suggests that this iterative process is helpful.

It is critical to understand that the query-result pairs are evaluated completely independently of the retrieval systems used to help select the images. Ideally, the only effect of the selection process is that the responses are more uniformly distributed. Using four different systems allows us to address the whether the process introduces significant bias into the measurement of retrieval systems (§2.4.3).

Evaluation experimental protocol. We asked many people to evaluate query-result pairs. This achieves two goals. First, we are interested in the range of results due to human subjectivity. Second, we wanted to collect as much data as possible. We collected data for two experiments two paradigms: query by image and query by text. In this paper we focus on query by image.

Due to practical considerations, roughly half of the data was produced by a single person. In total, 20,000 query-result pairs were evaluated for query by image example and 5,000 pairs were evaluated for query by text example. The evaluation was performed by 32 participants, out of which 3 participants evaluated both the paradigms. The data domain of this work is 16,000 images from the Corel data set.

Calibrating for participant variance. We used the data from the common sets to reduce the biases due to the different participants. To do so, we mapped the results of each participant in a given experiment by a single linear transformation so that their mean and variance of their results on the common set was the global mean and variance on this set. The effect of this is studied in §2.4.1.

2.2 Image retrieval systems

Keyword retrieval. The Corel image keywords can be used as a pseudo query by example method. Here, we score the match of two images by:

$$score = \frac{\mid W_Q \cap W_R \mid}{min(\mid W_Q \mid, \mid W_R \mid)} \tag{1}$$

where W_Q is the set of words associated with the query, W_R is and the set of words associated with the retrieved image, and $\mid X \mid$ is the number of elements in a set X. We denote this retrieval method as “Keywords”.

Region based multi-modal mixture models. Recent work proposes modeling image data as being generated by hidden factors which are responsible for jointly generating image region features and associated text [7, 4, 17, 5]. Here we model the joint probability of a particular blob, b, and a word w, as

$$P(w,b) = \sum_l P(w|l)P(b|l)P(l) \tag{2}$$

where l indexes over the concepts, $P(l)$ is the concept prior, $P(w|l)$ is a frequency table, and $P(b|l)$ is a Gaussian distribution (diagonal covariance) over features.

To train such models, we represent the node responsible for each image word and region by missing values, and use Expectation-Maximization to iteratively estimate the model parameters and the expectations for the missing values. However, in our case, where the correspondence between words and image regions is not known, there are additional choices and complexities. In particular, we must make choices how word likelihood from regions becomes word likelihood for the image which is what can be observed [5].

To implement image retrieval, we compute the probability that the model parameters for a database document can generate the observed regions of the query document.

The model can be trained with both image region features and words (labeled “RWMM”), or using regions only (“ROMM”). For image retrieval, we only use the image feature part of the model. Thus, if words are used at all, it is only during training. We further consider two retrieval scenarios. The first assumes complete access to all data, and thus we are able to match images in our training set. While in most situations using the training set model is not interesting, in the retrieval context it can make sense. In this case we affix the suffix “ALL” to the method label.

The second scenario uses the model as a template for matching new images. Neither the query image, nor the result image is part of the training set. Here we affix the suffix “TEST” to the method label. In particular, the method RWMM-TEST seems like an interesting retrieval paradigm. The words help ensure that the model encodes some relationship between image features and semantics, but the model is applicable to matching images without keywords and that have not been seen by the training system — of course, regions with the appropriate semantics must be in the training data.

The variant used for image selection in the query-by-example experiment, “ROMM-CALIB” is an older version of the system which was trained without words on subsets of the entire image data set. The results were then concatenated. Image selection for the query-by-text case used the analogous method, but text was included (“RWMM-CALIB”).

GIFT [2] is an open framework for content-based image retrieval. In its standard implementation, it is a pixel based CBIR system based on both local and global color and texture histograms. We use the standard system as one of the four systems used for improving the uniformity of the human evaluation results. In the retrieval method comparison (§2.4.2) we evaluate the effect of limiting the GIFT to use only color (“GIFT-color”), and only texture (“GIFT-texture”).

SIMPLIcity [48] is a region-based CBIR system which combines semantic classification methods, a wavelet based approach for feature extraction, and an integrated region matching based on image segmentation.

2.3 Mapping retrieval algorithm scores to human evaluation scores

The ground-truth data is composed of human scores corresponding to pairs of query-result images from the evaluation data set. We want to use this data to provide a mapping which takes the image retrieval scores into human evaluation scores for each system. Such mappings will put all systems onto the same scale, namely human evaluation scores. They also render retrieval scores as absolute scores which is useful for negotiating with users regarding the quality of the images to be returned (e.g. “good match” versus top 10).

We tried three kinds of mapping methods as follows:

(a) Monotonic mapping minimizing squared error In this method we map the computer scores to the human evaluation scores such that the average sum of the Euclidean distance between the mapped scores and the human scores is minimized, subject to mapped scores being monotonic. If $\mathbf{X}$ is a vector of computer scores arranged in ascending order and $\mathbf{Y}$ be a vector of corresponding human scores. If the mapped scores are represented by $\tilde{\mathbf{Y}}$, then the *objective* function to be minimized is:

$$E = \sum_{i=1}^{N} (\tilde{y}_i - y_i)^2 \tag{3}$$

subject to the constraint that $\tilde{\mathbf{Y}}$ is monotonic.

The preceding problem was solved using the MATLAB$^{©}$ routine *quadprog*. Since the number of constraints is large, we adopt bootstrapping [13] to average over the samples and find the estimate of Y that minimizes Eq. 3 subject to the constraint that $\tilde{\mathbf{Y}}$ is monotonic.

(b) Monotonic mapping maximizing correlation

Since we propose to use the correlation between the human scores and the computer scores as a measure of performance, it seems logical to obtain a mapping function that maximizes the correlation. Hence, the second fitting method performs the mapping such that the correlation coefficient between the mapped scores and human scores is maximized, subject to the mapped scores being monotonic. The task is to maximize:

$$C = \sum_{i=1}^{N} \frac{(y_i - \mu)(\tilde{y}_i - \tilde{\mu})}{\sigma \tilde{\sigma}} \tag{4}$$

where μ and $\tilde{\mu}$ are the mean for the original and mapped data respectively and similarly σ and $\tilde{\sigma}$ are the variances.

We would expect the correspondence obtained in this method to be higher than that obtained with the previous method and Table 2 confirms this for a majority of the data. The reader is forewarned that the method employed to carry out the optimization is guaranteed to give only a local minima.

Non-linear programming tools available with $MATLAB$ solve Eq.4. Specifically a routine *fmincon* is used which is based on Newton's method for large scale nonlinear minimization [13],[12]. We again use bootstrapping to get a generalization on the error and also obtain a vector of mapped scores that corresponds to the human scores.

(c) Monotonic Bayesian curve fitting

Since *fmincon* does not guarantee a global maxima/minima and we may be overfitting with the previous approach, we adopt a sampling method [24, 26] as the third method, which employs Markov Chain Monte-Carlo (MCMC) simulation to obtain the parameters of a model that maximize the posterior. Monotonicity is constrained during the sampling. This approach runs fine on our entire data set, and often gives us the best mapping function (§2.4.3).

This is a generalized monotonic curve fitting approach that is based on the Bayesian analysis of the isotonic regression model. Isotonic regression schemes [38], [40] fit monotonically increasing step functions to data. This model uses the concept of *change-points* to fit *cubic ogives.*

A function f(x), $x \in [a, b] \subseteq \Re$ is said to be an ogive in the interval [a,b] if it is monotone increasing and there is a point of inflection x^* such that f(x) is convex up to x^* and concave thereafter. The model is assumed to be continuous piecewise and differentiable between the *knots* (change-points). These assumptions lead to the characteristics of the model that is piecewise linear between the knots. Starting from first principles [38] the cubic ogive function is derived to be:

$$f(x) = \delta + \gamma(x - t_0) + \beta(x - t_0)^2 + \frac{1}{6}\sum_{i=1}^{k+1} \beta_i (x - t_{i-1})^3 \tag{5}$$

where the t_0 is the inflection point and δ, γ, β are model parameters.

The method is briefly outlined. The data is assumed to be normally (Gaussian) generated around *change points or knots* whose position and number are random. The dimensionality of the model is related to the number of change points accommodated in the model. Hence, this forms the space of varying multi-dimensional mixture models (because the space is now a mixture of varying multi-dimensional parameter vectors). Around each knot the authors adopt a prior to generating the data. If (y_i, x_i), $i = 1, ..., N$, denote N data pairs of corresponding human scores and computer scores respectively, such that the x_i are ordered in an ascending order, then if the ordered set of M change points is denoted by $\overrightarrow{t} = t_1, t_2, ..., t_{M-1}$, this forms M disjoint sets. The conjugate priors are assumed on the y_i's. The data generative model assumes identically independent distributions from each of the disjoint sets B, hence the probability of generating data within a set i is:

$$y_i = N(y_i | \mu_j, \Psi) \tag{6}$$

where μ_j is the mean-level in the j^{th} set and Ψ is the global variance term. The likelihood of data being generated by the model parameters in a set j is given by:

$$P(Y_j | M, t, \Psi, \mu_j) = \Pi_{i=1}^{n_j} f(y_i | \mu_j, \Psi) \tag{7}$$

The likelihood of the complete data Y given the model is just the product of the likelihoods within sets. Hence the complete likelihood is:

$$P(Y_j | M, \overrightarrow{t}, \Psi, \mu) = \Pi_{j=1}^{M} \Pi_{i=1}^{n_j} f(y_{ij} | \mu_j, \Psi) \tag{8}$$

Combining the likelihood and the priors the posterior is established. Since its computation requires the integration over varying model space which is not an easy task a simpler solution of MCMC approach is suggested. The MCMC sampler draws samples from the unconstrained model space and retains only those samples for which the monotonic constraint holds. The working of the MCMC simulation is a variant of the Metropolis-Hastings algorithm [25, 35] and is explained briefly below:

1. The chain is started from the simplest model with just one change point with a global mean level and variance drawn from the prior.

2. Changes are then adopted in the model, which may be one of these adding a new change point, or deleting an existing change point or by altering a change point in the model. These changes are accepted with probability Q:

   $$Q = min(1, \frac{p(M'|Y)S(M|M')}{p(M|Y)S(M'|M)}) \tag{9}$$

   where M represents all the model parameters in the current model and M' denotes the model with changes and S is the proposal distribution which is set to be a Gaussian. As the model is changed, the μ's and Ψ's change accordingly in the next iteration of the MCMC.

3. If $u \sim U(0, 1) < Q$ then $M(t+1) = M'$, else $M(t+1) = M$.

4. The constraint $\mu_1 \leq \mu_2 \leq .... \leq \mu_{M-1}$ is applied to the samples and only those samples, which obey the constraint, are retained.

5. For any point x in $\mathbf{X}$ the distribution y is an average of the distribution of y for each of the models given x and the model parameters.

The model we have used is from the biostatistics [25] literature. This model fits cubic curves between the random points. This information is encoded in the model parameters M. A more detailed treatment to this subject is given in [25, 35].

2.4 Experiments

2.4.1 Variance across evaluators

Table 1 shows the average variance of the results for the common test set for each of paradigms with and without the normalization described in (§2.1). The results show that there is variability in the participants that is worth calibrating for. Thus we apply the transformation computed on the common set to adjust all the results from that participant.

Table 1: The effect of adjusting on human evaluation scores to reduce differences among participants. The table shows the average standard deviation for standardized scores (global mean 0 and variance 1) for the three experiments before and after adjustment using the method described in the text (§2.1). This adjustment significantly reduces the variance.

Interface	Query by	Query by text	
	1-5	Binary	1-9
Number of participants	24	6	5
Average variance with standardized scores	1.38	0.19	2.88
Average variance with person dependent adjustment	1.15	0.036	0.937

2.4.2 Updating evaluation pair choice based on measured mapping functions

As described in §2.1, once we have a reasonable amount of evaluation data, we can use the retrieval system specific mapping functions (§2.3) to further improve the generation of query-result pairs for subsequent data collection. Recall that our goal is to have a roughly uniform response over our evaluation responses. A simple measure of this for 5 categories is $\frac{1}{5}\sum_{i=1}^{5}\|f(i) - 0.20)\|$, where $f(i)$ is the fraction of responses for category i. We computed this measure for the responses from the sampling based on the initial proposal (negative fifth power of rank), and the responses from subsequent data based on the mapping functions computed from the first part. The results in Table 2 show that the second data set induced more uniform responses.

Table 2: Deviation from uniformity of human evaluation results for the four calibration retrieval systems: (1) GIFT; (2) SIMPLIcity; (3) ROMM-CALIB; and (4) Keywords.

	1	2	3	4
initial data	0.20	0.19	0.14	0.08
mapped data	0.15	0.14	0.12	0.05

2.4.3 Mapping CBIR system scores to human evaluation results

Table 3 provides the correlations between the mapped score and the adjusted human score for all three fitting methods. In order to investigate sources of bias, we computed results for each of the four calibration system evaluated using only the images selected by each of the four. We found no significant consistent trend that using the same algorithm for selection and testing is an advantage to that algorithm. For example, if we used the maximum of each of the three fitting methods, and allow each algorithm to be paired with its own selection results, then the rank order does not change compared to using the mean, or the value from all data.

In general, we find that the Bayesian fitting method gave consistently good results. The constrained correlation maximization method also gave serviceable results. In contrast, least squares fitting did not work very well, which is perhaps not surprising given that we settled for correlation as our main measure of interest.

2.4.4 Comparing image retrieval algorithms

To compare image retrieval algorithms we first find a good mapping of the scores of that algorithm on the evaluation set to the adjusted human scores as described above. We then compute the correlation of the mapped scores to the human scores. The results are in Table 4.

Table 4: Grounded comparison of content based retrieval methods. We report the correlation of mapped computer scores with human scores. Each method uses its own, most favorable, monotonic mapping.

	Correlation of the calibrated human to the mapped system scores
ROMM-ALL	0.24
ROMM-TEST	0.17
RWMM-ALL	0.35
RWMM-TEST	0.23
GIFT	0.17
GIFT-color	0.15
GIFT-texture	0.07
SIMPLIcity	0.19
Keywords	0.51

Estimated precision recall curves. We consider the correlation results to be the best single indicator of performance under our methodology. However, we can use our results to estimate other performance characterizations such as Typically one plots the average values of precision versus recall over a threshold modulating the number of images returned. We emphasize that the form of our data is *different* than the form suggested by the formulas, and thus producing estimated PR curves requires care. We have a large number of query-result pairs which, by design, are a non-uniform sampling of the space of such pairs. Since we have many such pairs we can weight our averages to correct for the sampling. To compute the curves we essentially treat the top M CBIR responses as a single query for which we can compute the three quantities in the above two formulas. However, in order to estimate the ratios in the case of uniform sampling, which, in turn, estimates the ratios if we had all the data, we weight the computation of the quantities in (3) and (4) by the reciprocal of the sampling function. The estimated PR curves are in Figure 2.4.4.

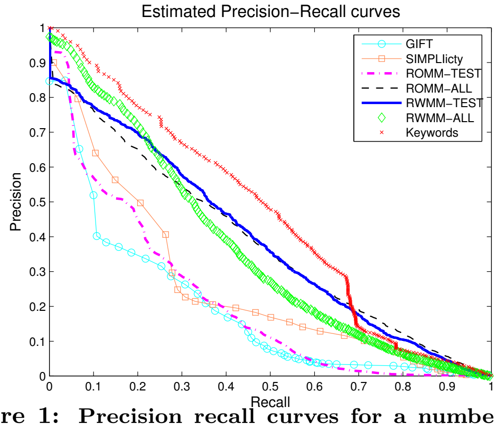

Figure 1: Precision recall curves for a number image retrieval methods. A relevant retrieved image corresponds to an adjusted human evaluation score greater than 3. Because the evaluation set is biased towards good matches, we have to estimate the PR curves by reversing the bias in rank. See text for details.

Our results show, not surprisingly, that keyword retrieval outshines the image based methods. This simply reflects the

Table 3: The correlation between the mapped scores and the human evaluation scores. The tabulated values are the correlation measures for each of the four calibration systems, as computed based on the samples provided from each of the four systems, the average of those results, and based on all the data combined. The systems are: 1) GIFT; 2) SIMPLIcity; 3) ROMM-CALIB; and 4) Keywords.

Fitting methods	*Average correlation between human scores and mapped GIFT scores on data selected by different systems*					
	1	*2*	*3*	*4*	*Mean*	*All*
a	0.18	0.10	0.13	0.10	0.13(0.04)	0.10
b	0.13	0.16	0.26	0.23	**0.20**(0.03)	**0.17**
c	0.13	0.18	0.22	0.21	0.19(0.04)	0.10

Fitting methods	*Average correlation between human scores and mapped SIMPLIcity scores on data selected by different systems*					
	1	*2*	*3*	*4*	*Mean*	*All*
a	0.13	0.20	0.14	0.20	0.17(0.04)	0.18
b	0.19	0.23	0.24	0.31	**0.24**(0.05)	0.18
c	0.17	0.25	0.23	0.25	0.23(0.04)	**0.19**

Fitting methods	*Average correlation between human scores and mapped ROMM scores on data selected by different systems*					
	1	*2*	*3*	*4*	*Mean*	*All*
a	0.17	0.18	0.18	0.20	0.18(0.01)	0.21
b	0.22	0.26	0.29	0.37	0.29(0.06)	0.23
c	0.31	0.33	0.43	0.34	**0.35**(0.05)	**0.24**

Fitting methods	*Average correlation between human scores and mapped Keywords scores on data selected by different systems*					
	1	*2*	*3*	*4*	*Mean*	*All*
a	0.17	0.28	0.51	0.41	0.34(0.14)	0.27
b	0.25	0.32	0.61	0.57	0.44(0.17)	0.38
c	0.53	0.58	0.62	0.56	**0.57**(0.04)	**0.51**

fact that semantics play a dominant role in what users consider a match, and that we are not very good at determining image semantics from features. The results also corroborate the notion that annotation oriented evaluation can serve as a proxy for grounded evaluation. However, the results also suggest that the scope of such a proxy is limited. Since the keyword results were far from perfect, a significant portion of what our participants expressed through their choices is not captured, and thus not measurable, using the keyword proxy.

Using words in training helps capture some relation between semantics and features, and the methods RWMM-ALL and RWMM-TEST do relatively well as a result. Without words, but still encoding the entire training set, the performance drops but is still respectable. We see this method (ROMM-ALL) to be a alternative method to SIMPLIcity in that it reports a match over several image regions. However, while ROMM-ALL models the statistics of the data set, SIMPLIcity computes the matches on the fly. We found that ROMM-ALL performs a bit better than SIMPLIcity. When forced to model images in general, but not in the training set, the the mixture model approach becomes a simple feature match method, and the performance results reflect this (worse than SIMPLIcity, same as GIFT).

2.4.5 Effect of half of the ground truth developed by one person

In an experiment not reported elsewhere, we examined our data for possible bias due to the fact that one person (codenamed "Master") was responsible for 50% of the ground-truth data. We thus segregated the evaluations into that of the master, and that of the other 31 participants. Table 5 shows the correlation scores along with the standard error for the retrieval systems discussed in §2.2 on data obtained from one person, the rest and the combined data. The correlation scores on data from three sources are within the standard errors of each other suggesting that there does not appear to be a bias in the ground truth data.

Table 5: Correlation scores of image retrieval systems on data obtained from the evaluations of Master , Others and the Combined data.

Systems	*Master*	*Others*	*Combined*
ROMM-ALL	0.21 (0.03)	0.19 (0.04)	0.24 (0.01)
ROMM-TEST	0.20 (0.03)	0.19 (0.02)	0.17 (0.02)
RWMM-ALL	0.24 (0.03)	0.25 (0.03)	0.35 (0.03)
RWMM-TEST	0.23 (0.04)	0.22(0.03)	0.23 (0.03)
GIFT	0.17 (0.02)	0.20 (0.03)	0.17 (0.01)
SIMPLICITY	0.16 (0.03)	0.21 (0.02)	0.19 (0.01)
KEYWORDS	0.57 (0.04)	0.50 (0.05)	0.51 (0.02)

3. WORD PREDICTION PERFORMANCE FOR LARGE SCALE EVALUATION

Our second approach for measuring how well a system captures semantics is based on a processes which translate from images (visual representation) to words (semantics). In particular, we present work using this approach to evaluate potential of low and mid level computer vision algorithms to support inferring semantics from visual data. Translation performance can be measured on a large scale, by comparing a proxy measure for the proposed translation (predicted words) with the observed, associated text. Any of a number of recently developed approaches for image annotation could be used in this paradigm (e.g. [7, 17, 5, 27, 30, 28, 9]). For the results reported here, we use the region based multi-modal mixture model described above.

It is widely agreed that segmentation measures should be task oriented (see [8] for related work). We argue that word prediction is an excellent task because it is associated with higher level image semantics and recognition. An interesting orthogonal approach is to link segmentation performance to those provided by human subjects [34, 33]

3.1 Predicting words from images

Given an image region, its features imply a probability of being generated from each node according to the multimodal mixture model. For the results that follow we use 500 mixture components. 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.

3.2 Experimental Protocol

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. For the feature evaluation experiments images were segmented using Normalized Cuts [42].

Performance measures. To quantify word prediction 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.

3.3 Semantic based segmentation evaluation

We evaluate six variants from three classes of segmentation methods: the expectation-maximization segmenter used for Blobworld [11], Normalized Cuts [42], and the mean shift algorithm [14]. 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, gives three options (over segmentation, under segmentation, and quantization).

A possible confound in our process is the number of segments used for word prediction and thus in Figure 3.3 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).

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.

Table 6: 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 best single cue, followed by texture.

Feature set	Word prediction performance		
	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	0.079	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

3.4 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. 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.

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 by 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 6 we give word prediction performance for a number of combinations of features. Not surprisingly given the

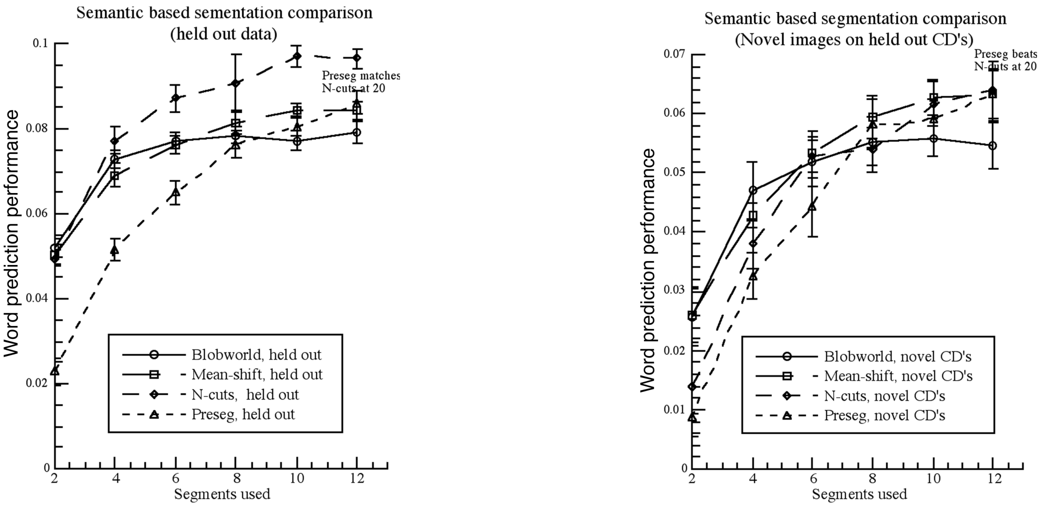

Figure 2: Segmentation methods compared using word prediction performance, evaluated on held out date (left), and novel data (right). All values plotted are positive, which means that performance always exceeds that using the empirical distribution.

nature of the Corel data, color is most useful. Interestingly, color space makes a significant difference. 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.

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.

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.

4. EVALUATION OF WORDS FOR IMAGE ANNOTATION

Not all words are appropriate for image annotation, since some words are not related to visual properties of images relative to our features. We summarize a method to measure the "visualness" of concepts using Web images; that is, to what extent concepts have visual characteristics. Knowing which concept has visually discriminative power is important for automatic image annotation, since not all concepts are related to visual contents. Such systems should first consider the concepts which have visual properties.

So far, most of the work related to image annotation or image classification has either ignored the suitability of the vocabulary, or selected concepts and words by hand. The popularity of sunset images in this domain reflects such choices, often made implicitly. We propose that increasing the scale of the endeavor will be substantively helped with automated methods for selecting a vocabulary which has visual correlates.

As an example of how this can be helpful, we are currently studying how to incorporate adjectives into image annotation models mentioned above. Adjectives bound to nouns have great potential to reduce correspondence ambiguity. For example, if a training image is labeled as "red ball", and "red" is known, but "ball" is not, the "red" item in the image will be weighted more heavily as a theory on what the "ball" is. However, although there are many adjectives, not all of adjectives are appropriate to use for image annotation task. Some adjectives have only a little or no relations to visual properties presented in images. For example, adjectives related to color such as "blue" and "green" are apparently good for annotation, while "hard" and "soft" are not likely adequate since it seems to be difficult to be distinguished from only visual properties. A measure of "visualness" of concepts can help select adjectives we should use.

Our method performs probabilistic region selection for regions that can be linked with concept "X" from images which are labeled as "X" or "non-X", and then we compute a measure of the entropy of the selected regions based on a Gaussian mixture model for regions. Intuitively, if such an entropy is low, then the concept in question can be linked with region features. Alternatively, if the entropy is more like that of random regions, then the concept has some other meaning which is not captured by our features.

4.1 Method to Compute the Image Entropy

To compute the "image region entropy" associated to a certain concept, we begin by gathering images related to the concept. While it is difficult to manually collect large numbers of images related to one concept, we can gather images likely associated to a certain concept using Web image search engines such as Google Image Search and Ditto. Of course, raw results from the Web image search engines, usually include irrelevant images. Moreover, the images usually include backgrounds as well as objects associated with a concept. Therefore, we need to eliminate irrelevant images and pick up only the regions strongly associated with the concept in order to calculate the image entropy correctly. We use only the regions expected to be highly related to the concepts to compute the image entropy. To select regions associated with concepts, we use a probabilistic method.

To find regions related to a certain concept we use an iterative algorithm. Initially, we do not know which region is associated with a concept “X”, since an image with an “X” label just means the image contain “X” regions. In fact, with the images gathered from the Web, even an image with an “X” label sometimes contain no “X” regions at all. So at first we have to find regions which are likely associated with “X”. To find “X” regions, we also need a model for “X” regions. Here we adopt a probabilistic generative model, namely a mixture of Gaussian, fitted using the EM algorithm.

In short, we need to know a model for “X” and which regions are associated with “X” simultaneously. However, each one depends on each other, so we proceed iteratively. Once we know which regions corresponds to “X”, we can compute the entropy of “X” regions relative to a different mixture of Gaussian, this one being a generic one fitted using the regions for a large number of images.

We segmented the images gathered from the Web as “X” images using JSEG [16]. After segmentation, we extract image features from each region whose size is larger than a certain threshold. As image features, we prepare three kinds of features: color, texture and shape features, which include the average RGB value and its variance, the average response to the difference of 4 different combination of 2 Gaussian filters, region size, location, the first moment and the area divided by the square of the outer boundary length.

We then prepared a generic Gaussian mixture model (GMM) for all image regions. This provides a base for computing the entropy. For this model we used about fifty thousand regions randomly picked up from the images gathered from the Web. We fit the GMM using Expectation Maximization (EM). As EM always includes randomness in the initial setting, we prepared k different generic models, and used the average of the entropies over these (here $k = 5$).

We estimate the entropy of the image features of all the regions weighted by $P(X|x_i)$ with respect to the generic model. The average probability of image features of “X” weighted by $P(X|x_i)$ with respect to the j-th component of the l-th generic base represented by the GMM is given by

$$P(X|c_j, l) = \frac{w_{j,l} \sum_{i=1}^{N_X} P(f_{X,i}; \theta_{j,l}) P(X|r_i)}{\sum_{i=1}^{N_X} P(X|r_i)} \qquad (10)$$

where $f_{X,i}$ is the image feature of the i-th region of “X”, $P(f_{X,i}; \theta_{j,l})$ is the generative probability of $f_{X,i}$ from the j-th component, N_X is the number of all the regions which come from “X” images.

The entropy for “X” is given by

$$E(X) = \frac{1}{k} \sum_{l=1}^{k} \sum_{j=1}^{N_{\text{base}}} -P(X|c_j, l) \log_2 P(X|c_j, l) \qquad (11)$$

where N_{base} is the number of the components of the base (here $N = 250$).

4.2 Experiments

As test images associated with concepts, we used the images gathered from the World Wide Web by providing 150 adjectives for Google Image Search. We obtained about 250 Web images for each adjective. Totally we obtained about forty thousand images associated with adjectives.

Table 7 shows the 10 top adjectives and their image entropy. The entropy of “dark” is the lowest, so in this sense “dark” is the most “visual” adjective among the 150 adjectives we used in this experiment.

We show the ranking of adjectives related color in the lower part of Table 7. They are generally ranked in the upper ranking, although images from the Web included many irrelevant images. This shows the effectiveness of the method.

Table 8 shows the 10 bottom adjectives. In case of “religious”, which is ranked in the 145-th, the computed entropy was relatively large, since the image features of the regions included in “religious” images have no prominent tendency.

In fact, this result is not always consistent with our intuition, since region selection did not work well for some adjectives. As future work, to compute more precise “image region entropy”, we will improve the method, especially the probabilistic region selection method.

Table 7: Entropy ranking.

rank	adjective.	entropy
1	dark	0.0118
2	senior	0.0166
3	beautiful	0.0178
4	visual	0.0222
5	rusted	0.0254
6	musical	0.0321
7	purple	0.0412
8	black	0.0443
9	ancient	0.0593
10	cute	0.0607
(color adjectives)		
7	purple	0.0412
8	black	0.0443
36	red	0.9762
39	blue	1.1289
46	yellow	1.2827

Table 8: Entropy ranking.

rank	adjective.	entropy
141	elderly	2.5677
142	angry	2.5942
143	sexy	2.6015
144	open	2.6122
145	religious	2.7242
146	dry	2.8531
147	male	2.8835
148	patriotic	3.0840
149	vintage	3.1296
150	mature	3.2265

5. SUMMARY

Building retrieval systems which effectively provide the user with semantic access to large data sets will require much additional research effort. We posit that research into the effective evaluation *in a semantic context* is a necessary component of this effort. In this paper we have described three contributions to this effort. First, we provide a method for evaluating image retrieval. Second, we argue that automatic image annotation is an excellent way to approach the large scale evaluation of low and mid level algorithms which support inference of image semantics. Finally, we provide a measure for estimating the potential that a word is sufficiently visual to be included for image annotation and implicit indexing based on image characteristics.

6. REFERENCES

[1] The benchathlon network, http://www.benchathlon.net.

[2] The gnu image-finding tool, www.gnu.org/software/gift.

[3] L. H. Armitage and P. G. B. Enser. Analysis of user need in image archives. *Journal of Information Science*, 23(4):287–299, 1997.

[4] K. Barnard, P. Duygulu, and D. Forsyth. Clustering art. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages II:434–441, 2001.

[5] K. Barnard, P. Duygulu, N. d. Freitas, D. Forsyth, D. Blei, and M. I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[6] K. Barnard, P. Duygulu, R. Guru, P. Gabbur, and D. Forsyth. The effects of segmentation and feature choice in a translation model of object recognition. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages II:675–682, 2003.

[7] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, pages II:408–415, 2001.

[8] S. Borra and S. Sarkar. A framework for performance characterization of intermediate level grouping modules. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(11):1306–1312, 1997.

[9] P. Carbonetto, N. d. Freitas, and K. Barnard. A statistical model for general contextual object recognition. In *European Conference on Computer Vision*, 2004.

[10] 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*, 24(8):1026–1038, 2002.

[11] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.

[12] T. Coleman and Y. Li. On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds, mathematical programming. 67(2):189–224, 1994.

[13] T. Coleman and Y. Li. 6(4):1040–1058, 1996.

[14] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):603–619, 2002.

[15] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos. The Bayesian image retrieval system, PicHunter: Theory, implementation and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–35, 2000.

[16] Y. Deng and B. S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.

[17] P. Duygulu, K. Barnard, J. d. Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *The Seventh European Conference on Computer Vision*, pages IV:97–112, 2002.

[18] P. G. B. Enser. Query analysis in a visual information retrieval context. *Journal of Document and Text Management*, 1(1):25–39, 1993.

[19] P. G. B. Enser. Progress in documentation pictorial information retrieval. *Journal of Documentation*, 51(2):126–170, 1995.

[20] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The qbic system. *IEEE Computer*, 28(9):22–32, 1995.

[21] D. A. Forsyth. Benchmarks for storage and retrieval in multimedia databases. In *Storage and Retrieval for Media Databases III*, volume 4676. SPIE, 2002.

[22] V. N. Gudivada and V. V. Raghavan. Content-based image retrieval-systems. *IEEE Computer*, 28(9):18–22, 1995.

[23] N. J. Gunther and G. B. Beratta. Benchmark for image retrieval using distributed systems over the internet: Birds-i. In G. B. Beretta and R. Schettini, editors, *Internet Imaging III*, volume 4311, pages 252–267. SPIE, 2001.

[24] N. A. Heard and A. Smith. Bayesian piecewise polynomial modeling of ogive and unimodal curves. Technical report, 2002.

[25] N. A. Heard and A. Smith. Bayesian piecewise polynomial modeling of ogive and unimodal curves. Technical report, 2002.

[26] C. Holmes and N. Heard. Generalized monotonic regression using random change points. *Statistics in Medicine*, 22(4):623–638, 2003.

[27] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *the 26th annual international ACM SIGIR conference*, pages 119–126, 2003.

[28] J. Jeon and R. Manmatha. Using maximum entropy for automatic image annotation. In *In Proc. International Conference on Image and Video Retrieval (CIVR-2004)*, pages 24–32, 2004.

[29] M. La Cascia, S. Sethi, and S. Sclaroff. Combining textual and visual cues for content-based image retrieval on the world wide web. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, 1998.

[30] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *In the Proceedings of NIPS03*, 2003.

[31] W. Ma and B. Manjunath. Netra: A toolbox for navigating large image databases. *Multimedia Systems*, 7:84–198, 1999.

[32] M. Markkula and E. Sormunen. End-user searching challenges indexing practices in the digital newspaper photo archive. *Information retrieval*, 1:259–285, 2000.

[33] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004. in press.

[34] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision*, 2001.

[35] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[36] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254, 1996.

[37] T. Pfund and S. Marchand-Maillet. Dynamic multimedia annotation tool. In G. B. Beretta and R. Schettini, editors, *Internet Imaging III*, volume 4672, pages 206–224. SPIE, 2002.

[38] T. Robertson, F. Wright, and R. Dykstra. *Order-Restricted statistical Inference*. Wiley: New York, 1998.

[39] G. Salton. The state of retrieval system evaluation. *Information Processing and Management*, 28(4):441–450, 1992.

[40] M. Schell and B. Singh. The reduced monotonic regression method. *Journal of the American Statistical Association*, 92(437):128–135, 1997.

[41] S. Sclaroff, L. Taycher, and M. La Cascia. ImageRover: A content-based image browser for the world wide web. In *IEEE Workshop on content-based access of image and video libraries*, 1997.

[42] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[43] V. N. Shirahatti and K. Barnard. Evaluating image retrieval. In *Proc. of IEEE Computer Vision and Pattern Recognition*, pages I:955–961, 2005.

[44] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Matching and Machine Intelligence*, 22(12):1349–1379, 2000.

[45] J. R. Smith. Image retrieval evaluation. In *IEEE Workshop on content-based access of image and video libraries (CBVAILVL)*, 1998.

[46] J. Vogel and B. Schiele. On performance characterization and optimization for image retrieval. In *7th European Conference on Computer Vision*, volume IV, pages 49–63. Springer, 2002.

[47] J. Z. Wang and J. Li. Learning-based linguistic indexing of pictures with 2-D MHMMs. In *ACM Multimedia*, pages 436–445, 2002.

[48] J. Z. Wang, J. Li, and G. Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.

[49] K. Yanai and K. Barnard. Image region entropy: A measure of "visualness" of web images associated with one concept. In *Proc. of ACM International Conference on Multimedia*, 2005.