A COMPARATIVE STUDY OF EVIDENCE COMBINATION STRATEGIES

Alexei Yavlinsky, Marcus J. Pickering, Daniel Heesch and Stefan Rüger

Department of Computing, South Kensington Campus,
Imperial College London SW7 2AZ, UK
{alexei.yavlinsky, m.pickering, daniel.heesch, s.rueger}@imperial.ac.uk

ABSTRACT

This paper reports on experimental results obtained from a performance comparison of feature combinations strategies in content based image retrieval. The use of Support Vector Machines is compared to CombMIN, CombMAX, CombSUM and BordaFuse combination strategies, all of which are evaluated on a carefully compiled set of Corel images and the TRECVID 2003 search task collection.

1. INTRODUCTION

The question of how to combine features optimally is a topic of great interest in the information retrieval community. There has been considerable research into ways of combining text retrieval systems to improve the overall precision [1, 2] and often the problem amounts to finding the appropriate set of weights for the retrieval systems.

In content based image retrieval, many researchers have focused on relevance feedback as the prime technique for weighting the importance of image features, and it is now widely employed in image retrieval systems [3, 4]. However, the success of this process depends on the quality of results that are initially presented to the user in response to their query. There have to be at least a few relevant images returned for it to be worthwhile, and thus it is important to maximise the expected performance of the system. Here we investigate the usefulness of Support Vector Machine meta-classification, CombMIN, CombMAX, CombSUM and BordaFuse evidence combination strategies to address this problem. We evaluate these strategies on both the Corel collection and key frames from the TRECVID 2003 video collection. The paper is structured as follows: Section 2 describes the features and the image collections used for the evaluation. Sections 3 and 4 discuss, respectively, the motivation behind our chosen approach and the results obtained on the two image collections.

2. EXPERIMENTAL SETUP

2.1. Features

For both collections, we use 9 low-level colour and texture features. For the TRECVID collection we also use speech recognition transcripts. These features are described in more detail in our earlier work [5].

HSV Global Colour Histograms. We use the quantised HSV colour space [6] which has proven to work better than other 3D colour models for image retrieval in our experiments [7, 8].

This work was partially funded by the EPSRC, UK

HSV Focus Colour Histograms. This feature is essentially a truncated version of the aforementioned feature. Only pixels from the central 25% of the image are taken into account for feature computation.

Colour Structure Descriptor. This feature is based on the recently introduced HMMD (Hue, Min, Max, Diff) colour space, which is specified in the MPEG-7 standard [9] and is useful for capturing the local colour distribution.

Marginal RGB colour moments. For this desciptor, we formed individual histograms for each of the three colour channels and computed the mean and second, third and fourth central moment of each marginal colour distribution.

Thumbnail feature. This feature is obtained by scaling down the original image to a small thumbnail of a fixed size and then storing the grey value of each of the pixels into a feature vector. It is suited to identify near-identical copies of images, e.g., keyframes of repeated shots such as adverts.

Convolution filters. For this feature we use Tieu and Viola's method [10], which relies on a large number of highly selective features. Each feature attains high values only for a small proportion of the image collection and thus it is possible to find a small set of features that discriminate well between the example set and the rest of the collection.

Variance Feature. The texture variance feature is a histogram of grey value standard deviations within a a sliding window, determined for each window position. The histogram is computed for each of 9 non-overlapping image tiles and the bin frequencies concatenated into a single feature vector.

Smoothness Feature. This feature obtains a texture smoothness value for each of the 8×8 image tiles and is derived from the texture variance $\sigma^2(z)$.

Uniformity Feature. Uniformity is another statistical texture feature, for which a single uniformity value is recorded for each 8×8 image tiles.

Bag of words. Using the textual annotation obtained from the speech recognition transcripts, we compute a bag-of-words feature consisting for each image of the set of accompanying stemmed words (Porter's algorithm) and their weights. These weights were determined using the standard tf-idf formula and are normalised so that they sum to one.

2.2. Distance and Similarity Measures

In order to compare two images in the database we use the distances of their corresponding features. For these we use the L_1-norm throughout, except in the bag-of-stemmed-words feature which uses the L_1 norm raised to the power of 3. We denote with $\mathrm{dist}_f(p, q)$ the distance between images p and q under feature f.

To retrieve images with respect to a given feature we use a variant of the distance-weighted k-nearest neighbour (k-NN) approach [11]. Positive examples are supplied by the user, and a number of negative examples are randomly selected from the database. The distances, for feature f, from the test image T_i to each of the k nearest positive or negative examples (where 'nearest' is defined by the above mentioned L_1 distance) are determined, and the similarity between the query and the image for the given feature is calculated as

$$\mathrm{Sim}_f(Q, T_i) = \frac{\sum_{q \in Q} \left(\mathrm{dist}_f(T_i, q) + \varepsilon\right)^{-1}}{\sum_{n \in N} \left(\mathrm{dist}_f(T_i, n) + \varepsilon\right)^{-1} + \varepsilon}$$

where Q and N are the k-nearest positive and negative examples, ($|Q| + |N| = k$) and ε is a small positive number to avoid division by zero.

The combination strategies we describe in this paper use the similarity scores computed for each image to establish the overall ranking.

2.3. The Corel Collection

We use a subset of the Corel collection (described in our earlier work [7]) and use category queries to determine how well the system returns images that are visually similar to the query. The collection is created from the Corel Gallery 380,000 package and 63 image categories are selected that are reasonably coherent visually (i.e. semantic categories such as "Lifestyles", or "Paris" are excluded). Each category contains at least 90 images and there are no categories that duplicate each other. The collection is randomly partitioned into 25% training subset and 75% test subset, and remains the same throughout the experiment to ensure that the query images are not retrieved by the system. A retrieved image is judged relevant if it comes from the same category as the source images contained in the query. For each category, 10 n-image queries are randomly generated from the training set, for each n between 1 and 6, and for each query all images are retrieved from the test subset ranked by their similarity.

For this evaluation the experimental setup is modified, by further dividing the two partitions into two equal disjoint image sets, thus creating two separate image collections, each having the same proportion of training and test images. For each collection the 3780 queries are generated as above. One of these collections is used for gathering ground truth data for the Support Vector Machine and the other is used to evaluate its performance; we refer to these as the *training partition* and the *test partition*, respectively.

2.4. TRECVID 2003 Search Task Collection

The other image collection we use for our evaluation is the set of 32,318 keyframes from this year's TRECVID video collection [5], which consists of news video footage. The search task specifies 25 topics, each exemplified by a few query images for which the relevant video keyframes need to be retrieved. We use the published relevance judgements for these topics to evaluate the performance of our combination strategies. This collection is split in the same manner as the Corel collection, but the same queries are kept for both the training and the evaluation.

3. FEATURE COMBINATION

Our approach is to train a linear Support Vector Machine (SVM) to act as a metaclassifier for the scores associated with each feature given by the k-NN algorithm, and use it to rank unseen images by returning a single relevance score for each.

SVMs have been used for the task of combining classifiers in the earlier work by Lin et al. [12], where good empirical performance has been reported. Our contribution is in the application of this technique to content based image retrieval, and a thorough evaluation of its performance on two well-defined image collections.

3.1. Support Vector Machines

SVMs [13] are learning machines that are capable of performing binary classification. Given a set of l training points belonging to two separate classes

$$D = \{(x_1, y_1), \ldots, (x_l, y_l)\}, x \in \mathbb{R}^n, y \in \{-1, 1\},$$

the objective of the SVM is to separate them with a hyperplane function $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$ such that

$$\min_i |\langle \mathbf{w}, \mathbf{x_i} \rangle + b| = 1$$

subject to

$$y_i[\langle \mathbf{w}, \mathbf{x_i} \rangle + b] \geq 1 \text{ for all } i = 1, \ldots l.$$

This ensures that the hyperplane separates the two classes correctly with the maximum margin possible. The solution to this problem is found by minimisng the function $\frac{1}{2}\|\mathbf{w}\|^2$ under the above constraint, which can be solved using quadratic programming.

SVMs are known to have superior generalisation properties compared with many other binary classifiers and fast implementations are widely available. One such implementation, SVM^{light} [14], is used for our experiments.

3.2. Combination Strategy

To train the SVM, scores and relevance information are obtained by running retrieval tests on the training partion of the image collection. We define a score vector of an image as the vector containing n similarity scores for that image, each pertaining to one of the n features. Given a query, m score vectors that are associated with retrieved images from the target image category and m score vectors that are associated with any other category, are taken at random, and are labelled as positive and negative training examples, respectively. We select a query at random for every one of C target image categories, thus sampling $C \times m$ positive examples and an equal number of negative examples. For the TRECVID experiments, we treat each topic as a category and sample m or fewer score vectors for each, depending on how many positive images there are in the training partition for that topic.

The SVM is trained on these examples to derive the hyperplane that separates the positive and the negative examples with least error. Given the set of retrieved images for a new query, their relevance is defined as the distance of their score vectors from the hyperplane and they are ranked accordingly. The distance of a vector from the hyperplane is just a linear weighted sum of the vector's components; in this context, the hyperplane represents the

Feature/Combination Method	M.A.P.
HSV Global Histogram	0.2464
Convolution	0.2195
Variance	0.1613
HSV Focus Histogram	0.1586
Marginal RGB Moments	0.1163
Uniformity	0.1097
Smoothness	0.1013
SVM	**0.3931**
CombSUM	0.3650
BordaFuse	0.3521
CombMIN	0.2341
CombMAX	0.2279

Table 1. Feature and combination method performance on the test partition of the Corel collection

set of weights for the linear sum of scores that maximise the expected mean average precision. This linear sum projects the score points onto a new basis normal to the hyperplane; the exact position of the hyperplane does not matter, as we are only interested in the relative distances.

To test the SVM on the Corel collection, we use the 3780 queries generated from the test partition, where a separate SVM is trained for each query size n. For the TRECVID collection all topics have 3 example images, and 25-fold leave-one-out cross-validation is used to make sure that the mean average precision is not biased towards any of the 25 queries.

3.3. Combination Baselines

To judge the effectiveness of our combination approach we compare it against BordaFuse [15] and the popular CombMIN, CombMAX and CombSUM algorithms [2]. The combined similarity score for the last 3 is given, respectively, by the minimum, maximum, and the sum of the similarity scores of the retrieved image. The scores are normalised such that for a given query they have a zero mean and a unit variance for each feature. BordaFuse imitates the process of voting, where decisions are made by combining the opinions of a number of independent "experts"; the images are treated as candidates and features as voters and for each image the combined similarity score is the sum of rank values from ranked image lists pertaining to each feature.

4. RESULTS

4.1. Corel

Table 1 shows the mean average precision (MAP) of the combination methods and the individual features used in experiments on the Corel test partition. CombSum and BordaFuse provide a substantial improvement over the best feature (48% and 32% respectively), but the SVM outperforms both and gives a 60% improvement. A paired one-sided t-test reveals that the mean average precision values obtained through CombSum and of the SVM are significantly different at a confidence level of $\alpha \leq 0.01$. The training set for the SVM consists of 4 positive and 4 negative examples from each query belonging to one of the 63 categories. However, we found that increasing the set further does not result in a significant rise of mean average precision.

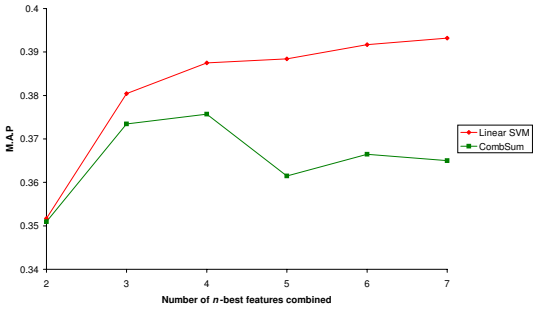

Fig. 1. Mean average precision of combining n-best features

	HSV, RGB and Convolution	HSV, Smoothness and Uniformity
Raster Scan	0.3563	0.2744
SVM	0.3536	0.2758
CombSUM	0.3143	0.2379

Table 2. SVM mean average precision vs. raster scanning the weight space

A noteworthy property of the SVM for the Corel collection is shown in Figure 1, which is that the addition of poor quality features does not degrade the combined performance, unlike CombSUM.

We use two additional benchmarks to evaluate the performance of the SVM: one is the mean average precision obtained by using optimal, *fixed*, weights, the other is the mean average precision obtained by using the optimal weights for each *individual* query. In each case we use a brute force raster scan of the weight space by quantising it into a multidimensional grid, which is then searched to find the particular set of weights for CombSUM that gives the maximum average precision. The search space can be drastically reduced by constraining the weights such that $\sum w_i = 1$ and $w_i \in [0, 1]$.

We apply the first benchmark to the combination of two sets of three different features, as shown in Table 2; both times the SVM performs as well as the weights established by raster scanning. Applying the second benchmark test to all the features yields an impressive mean average precision of 0.5022, confirming observations made earlier in [16] that feature weights are in fact highly query-specific, and that fixed weights cannot harness the full recognition capability of the features.

4.2. TRECVID

Table 3 shows the mean average precision of the combination methods and the individual features used in experiments on the TRECVID test partition. The results clearly indicate that the bag-of-words feature is on average much more robust than any of the visual features, yet even this feature yields a relatively low performance compared with results of other features in the Corel experiments. The baseline methods do not improve performance over the best single feature, which implies that, for many topics, visual features carry very little or no information. The training set for the SVM consists of 15 negative and 15 or fewer positive examples for each topic. Intuitively, one would expect the SVM to learn that the bag-of-words feature is far better at discriminating relevant images

Feature/Combination Method	M.A.P.
Bag of Words	0.1238
Thumbnail	0.0245
Colour Structure Descriptor	0.0186
HSV Global Histogram	0.0176
Convolution	0.0172
Variance	0.0142
CombSUM	0.1204
SVM	0.1170
CombMAX	0.0900
CombMIN	0.0864
BordaFuse	0.0603

Table 3. Performance of individual features and five different combination methods on the test partition of the TRECVID collection

but its mean average precision lies below that of CombSUM. One possible explanation for this is that the TRECVID score data contains more noise than that of Corel, owing to the mostly semantic nature of the topics in the TRECVID collection.

Raster scanning the space of fixed weights for all topics reveals that it is best to stick with the bag-of-words feature alone, whereas scanning the weight space for each individual topic achieves the mean average precision of 0.1904. This shows that for some topics the visual features do play a significant role.

Whilst the SVM does not perform well across all topics on average, it is capable of learning the weights well for some of them individually. One example is the topic for which one had to identify all shots of baseball players pitching: using the same set of features, the SVM was trained with a set of 100 positive examples and 100 negative examples and resulted in a mean average precision of 0.3885, compared to only 0.2967 of the next best method (CombSUM). We have found, however, that a large number of positive examples for a topic must be available for training in order for the SVM to be successful in this task.

5. CONCLUSIONS

We have compared the performance of the SVM meta-classification approach against the standard feature combination strategies. The implications of this study are two-fold. First, results for the Corel image set are not indicative of the performance of evidence combination, or in fact of image retrieval in general, on real world challenges such as the TRECVID search task. This performance discrepancy between Corel and TRECVID is consistent with that reported earlier by Westerveld and de Vries [17]. None of the strategies which proved successful in the Corel experiments were capable of improving over the best single feature in TRECVID. One should therefore take great care in identifying adequate benchmark collections when evaluating these approaches. However, in TRECVID the SVM meta-classification remains a promising approach for retrieval of images for specific topics, and this may turn out to be useful for classification tasks such as detecting anchor person shots in sequences of video keyframes.

6. REFERENCES

[1] B T Bartell, G W Cottrell, and R K Belew, "Automatic combination of multiple ranked retrieval systems," in *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994.

[2] J A Shaw and E A Fox, "Combination of multiple searches," in *Proceedings of the 2nd Text REtrieval Conference (TREC-2)*, 1994.

[3] Y Rui, T S Huang, and S Mehrotra, "Relevance feedback techniques in interactive content-based image retrieval," in *IEEE Transactions on Circuits and Systems for Video Technology*, 1998.

[4] Y Rui and T S Huang, "A novel relevance feedback technique in image retrieval," in *ACM Multimedia (2)*, 1999.

[5] D Heesch, M J Pickering, S Rüger, and A Yavlinsky, "Video retrieval using search and browsing with key frames," in *Proceedings of TRECVID 2003, NIST*, 2003.

[6] D Travis, *Effective Color Display*, Academic Press, San Diego, CA, 1991.

[7] M J Pickering and S Rüger, "Evaluation of key frame based retrieval techniques for video," *Computer Vision and Image Understanding*, vol. 92, no. 2, pp. 217–235, 2003.

[8] M J Pickering, S Rüger, and D Sinclair, "Video retrieval by feature learning in key frames," in *Proceedings of International Conference on Image and Video Retrieval (CIVR)*, July 2002.

[9] B S Manjunath and J-S Ohm, "Color and texture descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 703–715, 2001.

[10] K Tieu and P Viola, "Boosting image retrieval," in *5th International Conference on Spoken Language Processing*, Dec. 2000.

[11] T M Mitchell, *Machine Learning*, McGraw Hill, 1997.

[12] W-H Lin, R Jin, and A Hauptmann, "Meta-classification of multimedia classifiers," in *International Workshop on Knowledge Discovery in Multimedia and Complex Data*, 2002.

[13] V N Vapnik, *The Nature of Statistical Learning Theory*, SpringerVerlag, 1995.

[14] "Svmlight," http://svmlight.joachims.org/.

[15] J A Aslam and M Montague, "Models for metasearch," in *Proceedings of the 24th ACM SIGIR Conference on Research and Developement in Information Retrieval*, 2001.

[16] D Heesch, A Yavlinsky, and S Rüger, "Performance comparison between different similarity models for CBIR with relevance feedback," in *Proceedings of International Conference on Image and Video Retrieval*, 2003.

[17] T Westerveld and A de Vries, "Experimental evaluation of a generative probabilistic image retrieval model on 'easy' data," in *Proceedings of the 26th ACM SIGIR Conference on Research and Development in Information Retrieval, Multimedia Information Retrieval Workshop*, 2003.