FIRE in ImageCLEF 2007: Support Vector Machines and Logistic Models to Fuse Image Descriptors for Photo Retrieval

Tobias Gass, Tobias Weyand, Thomas Deselaers and Hermann Ney

Human Language Technology and Pattern Recognition Group,
RWTH Aachen University, Aachen, Germany
<last name>@i6.informatik.rwth-aachen.de

Abstract. Submissions to the photographic retrieval task of the ImageCLEF 2007 evaluation and improvements of our methods that were tested and evaluated after the official benchmark. We use our image retrieval system FIRE to combine a set of different image descriptors. The most important step in combining descriptors is to find a suitable weighting. Here, we evaluate empirically tuned linear combinations, a trained logistic regression model, and support vector machines to fuse the different descriptors. Additionally, clustered SIFT histograms are evaluated for the given task and show very good results – both, alone and in combination with other features. A clear improvement over our evaluation performance is shown consistently over different combination schemes and feature sets.

Key words: content-based image retrieval, feature combination, SIFT features

1 Introduction

ImageCLEF[1] is an evaluation event for textual (mono- and multilingual) and content-based retrieval of images. Evaluation campaigns are an important factor to foster progress in research and therefore we participated for the third time using our content-based image retrieval system FIRE.

Although the machine learning community has produced a large amount of strong classification techniques, the image retrieval community has so far only employed k-nearest neighbor approaches or techniques derived from (textual) information retrieval. Thus, only few systems build on top of state-of-the-art machine learning and image representation techniques. In this paper we evaluate histograms of SIFT features, a common image descriptor for object recognition, which is so far seldomly used for general photographic image retrieval, and compare different strategies to fuse image descriptors. The considered strategies are maximum entropy-based feature combination as presented last year [1] and support vector machines.

[1] http://www.imageclef.org

Support vector machines have so far been used in image retrieval by [2] for relevance feedback and as a feature combination strategy. A similar approach is presented in [3].

2 ImageCLEF 2007 Photographic Retrieval Task

The database used in the photographic retrieval task [4] was the IAPR TC-12 photographic collection [5] consisting of 20,000 natural still images annotated in three languages. 60 queries were given, each consisting of a short textual description and three sample images. The queries posed in 2007 are very similar to the 2006 queries. This allows the 2006 queries to be used in combination with the relevance judgements to train the log-linear models and SVMs for feature combination.

3 Features

In this section, we present the image descriptors we used in our experiments.

We briefly outline the features we used for our runs in the ImageCLEF 2007 evaluation. Additionally, we present *Clustered SIFT Histograms*, a variant of clustered histograms of local features, which will result in clear improvements over our official evaluation results.

Colour Histograms. Colour histograms are among the most basic approaches and widely used in image retrieval [6]. The colour space is partitioned and for each partition the pixels with a colour within its range are counted, resulting in a representation of the relative frequencies of the occurring colours. In accordance to [7], we use Jeffrey divergence to compare histogram descriptors.

Global Texture Descriptor. In [8] a texture feature is described consisting of several parts: *Fractal dimension*, *Coarseness*, *Entropy*, *spatial Gray-level difference statistics*, and the *circular Moran autocorrelation function*. From these, we obtain 43 dimensional vectors which have been successfully used in preceding ImageCLEF evaluations.

Invariant Feature Histograms. A feature is called invariant with respect to certain transformations if it does not change when these transformations are applied to the image. The transformations considered here are translation, rotation, and scaling. In this work, invariant feature histograms as presented in [9] are used.

Tamura Features. In [10] the authors propose six texture features corresponding to human visual perception: *coarseness*, *contrast*, *directionality*, *line-likeness*, *regularity*, and *roughness*. In our experiments we use coarseness, contrast, and directionality to create a histogram describing the texture [8].

GIFT Colour Descriptors. In [11] the authors propose global and local colour features using a quantised HSV colour space. The global features are colour histograms over the complete image, the local features are extracted from a uniform grid and describe the dominating colours in each grid cell.

Clustered Histograms of Local Features. Local image features offer some advantages over the mainly global image descriptors presented so far, because they allow for matching images w.r.t. the local concepts they have in common. To be able to cope with the amount of data that is to be handled when local image descriptors are extracted in great numbers, we represent the local descriptors in a histogram over a dictionary of visual words as proposed in [12]. As opposed to [12], we do not simply use image patches alone but apply the same technique to SIFT features [13], which have been shown to outperform image patches in many applications [14].

The creation of these histograms is a three step procedure:

1. local features are extracted from all training images,
2. the local features of all training images are jointly clustered using the EM algorithm for Gaussian mixtures to form 256-8000 clusters,
3. all information about each local feature is discarded except its closest cluster center. Then, for each image a histogram over the cluster identifiers of the respective patches is created, thus effectively coding which words from the code-book occur in the image.

These histograms are created using SIFT features and image patches. We reduce the dimensionality of the local features to 40 using PCA transformation to reduce the amount of data.

The histograms over SIFT features were not used in the official evaluation. In section 5, we show that they can lead to a performance improvement.

Sparse Patch Histograms. A computationally more efficient method for generating patch histograms was proposed in [15]. First, all patches are transformed into a lower-dimensional space using PCA. Then, a histogram grid is estimated by calculating the mean and variance of each axis of this space. Now, a patch histogram is created for each image by inserting all patches from the image into this grid. This technique allows to skip the computationally expensive step of creating a visual vocabulary by spanning the complete feature space, but does not perform as good as clustered histograms since the created histograms are sparse and the bins do not represent visual words.

4 Image Retrieval Method

Given a set of positive example images $\mathcal{Q}^+$ and a (possibly empty) set of negative example images $\mathcal{Q}^-$ a score $S(\mathcal{Q}^+, \mathcal{Q}^-, X)$ is calculated for each image X from the database:

$$S(\mathcal{Q}^+, \mathcal{Q}^-, X) = \sum_{Q \in \mathcal{Q}^+} S(Q, X) + \sum_{Q \in \mathcal{Q}^-} (1 - S(Q, X)). \tag{1}$$

where $S(Q, X)$ is the score of database image X with respect to query Q and is calculated as $S(Q, X) = e^{-\gamma W(Q,X)}$ with $\gamma = 1.0$. $W(Q, X)$ is a weighted sum of distances calculated as

$$D(Q, X) := \sum_{i=1}^{I} w_i \cdot d_i(Q_i, X_i). \qquad (2)$$

Here, Q_i and X_i are the ith feature of the query image Q and the database image X, respectively. d_i is the corresponding distance measure and w_i is a weighting coefficient. For each d_i, $\sum_{X \in \mathcal{B}} d_i(Q_i, X_i) = 1$ is enforced by re-normalisation.

Given a query $(\mathcal{Q}^+, \mathcal{Q}^-)$, the images are ranked according to descending score and the K images X with highest scores $S(\mathcal{Q}^+, \mathcal{Q}^-, X)$ are returned by the retrieval engine.

The selection of the weights w_i is a critical step, for which two methods have been deployed so far. It is possible to heuristically choose feature weights based on the performance of the individual features. This usually leads to superior results than an unweighted baseline. The second approach uses the maximum entropy framework to train a logistic model, which can then be used to calculate a score for a query/database image pair. Additionally, we present a novel approach to weight features using support vector machines.

4.1 Scoring of Distances Using Classifiers

We consider the problem of image retrieval to be a classification problem. Given the query image, the images from the database have to be classified to be either relevant (denoted by +1) or irrelevant (denoted by −1)

Because we want to classify the relation between images into the two categories “relevant” or “irrelevant” on the basis of the distances between their features, we choose the following way to derive the training data: For each query image Q_n of the 2006 task the distance vector $D(Q_m, X_n) = (d_1(Q_{m1}, X_{n1}), \ldots, d_I(Q_{mI}, X_{nI}))$ are calculated. This leads to N distance vectors for each of the images Q_m, where N denotes the number of images in the database. These distance vectors are then labelled according to the relevances: those $D(Q_m, X_n)$ where X_n is relevant with respect to Q_m are labelled +1 (relevant) and the remaining ones are labelled with the class label −1 (irrelevant).

This leads to a training set, so that any classifier can be trained. This classifier should be able not only to classify distance vectors of unseen images to an image from the database, but additionally return some score or confidence with which the database images can be ranked. In the following, we present two suitable approaches to compute these scores from the distance vectors.

4.2 Logistic Regression Feature Weights for Image Retrieval

To obtain suitable feature weights, a logistic model is promising, because it is suited to combine features of different types.

As features f_i for the logistic model we choose the distances between the i-th feature of the query image Q and the database image X_n:

$$f_i(Q, X_n) := d_i(Q_i, X_n i).$$

To allow for modelling prior probabilities, we include a constant feature $f_{i=0}(Q, X_n) = 1$. Then, the scores $S(Q, X_n)$ from Eq. (1) are replaced by the posterior probability for class +1 and the ranking and combination of several query images is done as before:

$$\begin{aligned} S(Q, X_n) &:= p(+1|Q, X_n) \qquad (3) \\ &= \frac{\exp\left[\sum_i \lambda_{+1i} f_i(Q, X_n)\right]}{\sum\limits_{k \in \{+1,-1\}} \exp\left[\sum_i \lambda_{ki} f_i(Q, X_n)\right]} \\ &= \frac{1}{1 + \exp\left(\sum_i \lambda_i f_i(Q, X_n)\right)} \text{ with } \lambda_i = \lambda_{-1i} - \lambda_{+1i} \end{aligned}$$

Alternatively, Eq. (3) can easily be transformed to be of the form of Eq. (1) and the w_i can be expressed as a function of λ_{+1i} and λ_{-1i}.

We train the λ of the logistic model from Eq. (3) using the GIS algorithm.

4.3 Support Vector Machine Scoring

Since dividing given distances into "relevant" and "irrelevant" is a two-class problem, it is quite natural to employ a support vector machine (SVM) [16].

SVMs are, contrary to logistic models described above, not a probabilistic method providing class-posterior probabilities to base the classification decision upon, but directly predict the label of the observation. An SVM commonly discriminates between two classes: −1 and +1, using the decision rule to classify an unseen observation X:

$$X \mapsto \hat{c}(X) = \operatorname{sgn}\left\{ \sum_{v_i \in \mathcal{S}} \alpha_i K(X, v_i) + \beta \right\} \qquad (4)$$

where K is a kernel function, $\mathcal{S}$ is the set of support vectors v_i, and the α_i are the corresponding weights, β is a bias term.

Considering the distance vector D, as described above as feature vectors, it is possible to rank images using the distances to the separating hyperplane. That is, given the distance vector $D(Q_m, X_n)$, by computing the distance

$$d(D(Q_m, X_n)) = \sum_{v_i \in \mathcal{S}} \alpha_i K(D(Q_m, X_n), v_i) + \beta \qquad (5)$$

, it is possible to compute a score $S(q, X) = \exp(d)$, which can then be used to replace the score of Eq. 1.

Since the number of "relevant" distance vectors given the photographic retrieval task is small compared to the number of "irrelevant" ones we randomly select a subset of "irrelevant" distance vectors to have the same number of distance vectors for both classes. Informal experiments have shown that using far more vectors from one class than from the other decreases the performance.

Table 1. Overview of our submitted results, competing submissions, and improvements presented in this work. "emp" denotes empirically determined weights, "ME" denotes maximum-entropy(logistic) scoring, and "SVM" denotes support vector scoring

submission	with text.	MAP	comment
average-NT	no	0.07	with query expansion
RWTH-FIRE-NT-emp	no	0.08	
RWTH-FIRE-ME-NT-20000	no	0.11	
best-NT	no	0.19	with query expansion
average(monolingual English)	yes	0.14	with query expansion
RWTH-FIRE-emp	yes	0.20	
RWTH-FIRE-ME-500	yes	0.20	
best(monolingual English)	yes	0.32	with query expansion
SVM-rbf-NT-withsift	yes	0.13	this work
FIRE-emp-withsift	yes	0.20	this work
SVM-linear	yes	0.21	this work
SVM-rbf	yes	0.25	this work

5 Experimental Results

In total, we submitted nine runs to the photographic retrieval task, five using textual and visual information jointly and four runs using only visual information. As can be seen in Table 1, textual information (monolingual English) greatly helps to achieve a better retrieval result, which was to be expected. In the visual-only runs, logistic regression also clearly helps to improve the results. It should be noted that 90.6% of the submissions to the photographic retrieval task used query expansion, which we did not use at all.

In the following, we show how to further improve our results.

5.1 SIFT Features

In Table 2(a) an overview of the performance of clustered SIFT histograms using different numbers of clusters is given. It can be seen that the performance increase correlates strongly with the number of clusters. However, using more than 1024 clusters did not lead to any more improvement. It can also be seen that the SIFT Features perform significantly better than global colour histograms, which usually perform quite well on this type of images. This is a strong indicator that the SIFT features capture important local information. Table 3 shows the results of adding SIFT histograms to our system, where they led to slight improvements using manually tuned weights but did not help significantly when other strong features were present, or a strong classifier was used.

5.2 SVM Scoring

The SVM Scoring approach presented in section 4.3 helped increase the MAP of our submissions by up to 25% relative. Even using linear kernels for the

Table 2. (a) Performance using clustered SIFT histograms with different numbers of clusters. (b) Different feature combination strategies compared.

(a)

number of clusters	map
colour histogram	0.022
256	0.027
512	0.039
1024	0.046
2048	0.042

(b)

scoring	train(2006)	test(2007)
emp	0.1625	0.1969
ME	0.1479	0.1974
SVM-linear	0.1581	0.2080
SVM-rbf	0.2091	0.2460

Table 3. Performance increase using SIFT features in combination with other visual features and in combination with other visual features and text.

	features used	unweighted	emp	ME	SVM-rbf
no text	baseline	0.0840	0.1017	0.1122	0.1282
	+sift	0.0870	0.1100	0.1110	0.1302
with text	baseline	0.1260	0.1946	0.1970	0.246
	+sift	0.1290	0.2015	0.1970	0.241

SVM, the performance increases compared to the logistic approach. The best results were achieved using an RBF-kernel with parameters estimated on the queries of the 2006 photographic retrieval task, which were used as development data. An overview of the results using the different scoring approaches is given in Table 2(b) which compares the unweighted baseline to hand-tuned linear weights, logistic scoring and SVM-scoring.

6 Conclusion

In this work, we described our approach to the photographic retrieval task of the ImageCLEF2007 evaluation. It can be seen that, for the given task, textual information is crucial to obtain good retrieval accuracy. Among the visual features, the clustered SIFT histograms perform better than other widely used features if used on their own, nonetheless in combination with other features they only lead to minor improvements.

Additionally, we presented an SVM-based approach for ranking retrieval results. This method outperforms our logistic regression scoring method by up to 25%, relatively.

For the future, the impact of user interaction is an important research topic. In particular it might be interesting to investigate discriminative machine learning techniques to learn good feature weights from user interaction.

Acknowledgement

This work was partially funded by the DFG (Deutsche Forschungsgemeinschaft) under contract NE-572/6.

References

1. Deselaers, T., Weyand, T., Ney, H.: Image retrieval and annotation using maximum entropy. In: Evaluation of Multilingual and Multi-modal Information Retrieval, CLEF 2006. Volume 4730 of LNCS., Alicante, Spain (2007) 725–734
2. Setia, L., Ick, J., Burkhardt, H.: Svm-based relevance feedback in image retrieval using invariant feature histograms. In: IAPR Workshop on Machine Vision Applications, Tsukuba Science City, Japan (2005)
3. Yavlinski, A., Pickering, M.J., Heesch, D., Rüger, S.: A comparative study of evidence combination strategies. In: ICASSP 2004. Volume 3., Montreal, Canada (2004) 1040–1043
4. Grubinger, M., Clough, P., Hanbury, A., Müller, H.: Overview of the ImageCLEF 2007 photographic retrieval task. In: Proceedings of the CLEF 2007 Workshop. LNCS, Budapest, Hungary (2008)
5. Grubinger, M., Clough, P., Müller, H., Deselaers, T.: The iapr benchmark: A new evaluation resource for visual information systems. In: LREC 06 OntoImage 2006: Language Resources for Content-Based Image Retrieval, Genoa, Italy (2006)
6. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 1349–1380
7. Puzicha, J., Rubner, Y., Tomasi, C., Buhmann, J.: Empirical evaluation of dissimilarity measures for color and texture. In: ICCV 1999. Volume 2., Corfu, Greece (1999) 1165–1173
8. Deselaers, T.: Features for image retrieval. Master's thesis, Human Language Technology and Pattern Recognition Group, RWTH Aachen University, Aachen, Germany (2003)
9. Siggelkow, S.: Feature Histograms for Content-Based Image Retrieval. PhD thesis, University of Freiburg, Institute for Computer Science, Freiburg, Germany (2002)
10. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. IEEE Transaction on Systems, Man, and Cybernetics **8** (1978) 460–472
11. Squire, D.M., Müller, W., Müller, H., Raki, J.: Content-based query of image databases, inspirations from text retrieval: Inverted files, frequency-based weights and relevance feedback. In: SCIA, Kangerlussuaq, Greenland (1999) 143–149
12. Deselaers, T., Keysers, D., Ney, H.: Discriminative training for object recognition using image patches. In: CVPR 05. Volume 2., San Diego, CA (2005) 157–162
13. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110
14. Dorkó, G., Schmid, C.: Object class recognition using discriminative local features. Rapport de recherche RR-5497, INRIA - Rhone-Alpes (2005)
15. Deselaers, T., Hegerath, A., Keysers, D., Ney, H.: Sparse patch-histograms for object classification in cluttered images. In: DAGM, Pattern Recognition, 27th DAGM Symposium. Volume 4174 of LNCS., Berlin, Germany (2006) 202–211
16. Schölkopf, B.: Support Vector Learning. Oldenbourg Verlag, Munich, Germany (1997)