Evaluation of Texture Features for Content-Based Image Retrieval

Peter Howarth and Stefan Rüger

Department of Computing, Imperial College London,
South Kensington Campus, London SW7 2AZ
{peter.howarth, s.rueger}@imperial.ac.uk

Abstract. We have carried out a detailed evaluation of the use of texture features in a query-by-example approach to image retrieval. We used 3 radically different texture feature types motivated by i) statistical, ii) psychological and iii) signal processing points of view. The features were evaluated and tested on retrieval tasks from the Corel and TRECVID2003 image collections. For the latter we also looked at the effects of *combining* texture features with a colour feature.

1 Introduction

Texture is a key component of human visual perception. Like colour, this makes it an essential feature to consider when querying image databases. Everyone can recognise texture but, it is more difficult to define. Unlike colour, texture occurs over a region rather than at a point. It is normally defined purely by grey levels and as such is orthogonal to colour. Texture has qualities such as periodicity and scale; it can be described in terms of direction, coarseness, contrast and so on [1]. It is this that makes texture a particularly interesting facet of images and results in a plethora of ways of extracting texture features. To enable us to explore a wide range of these methods we chose three very different approaches to computing texture features: The first takes a statistical approach in the form of co-occurrence matrices, next the psychological view of Tamura's features and finally signal processing with Gabor wavelets.

Our study is the first to focus an evaluation of texture features on the whole image, and to tailor features for optimum retrieval performance in this context. The majority of original papers devising or evaluating texture features used classification or segmentation tasks to measure performance [2,3,4,5]. Both of these tasks are significantly different to the problems faced in image retrieval where one looks at generic queries for an entire picture. Real pictures are made up of a patchwork of differing textures rather than the uniform texture images often used in studies, such as the ones taken from Brodatz's photo book [6]. To that effect we suggest encoding texture in terms of joint histograms of low dimensional texture characteristics over the image in the same way 3D colour histograms are computed, we have called this a Tamura image. Throughout our work we have considered how best to cope with varying image sizes, scales, formats and orientations.

P. Enser et al. (Eds.): CIVR 2004, LNCS 3115, pp. 326–334, 2004.

In the next section we look at the features we have chosen and how they are computed. Sect. 3 then describes the image libraries and similarity measures we used for evaluation. Sect. 4 presents our initial results on a training set and suggests modifications and parameters that we found gave the best retrieval performance. A larger performance comparison is carried out on the TRECVID2003 data set. Finally, Sect. 5 concludes the paper and outlines further work.

2 Texture Features

2.1 Co-occurrence

Statistical features of grey levels were one of the earliest methods used to classify textures. Haralick [7] suggested the use of grey level co-occurrence matrices (GLCM) to extract second order statistics from an image. GLCMs have been used very successfully for texture classification in evaluations [2].

Table 1. Features calculated from the normalised co-occurrence matrix $P(i,j)$

Feature	Formula
Energy	$\sum_i \sum_j P^2(i,j)$
Entropy	$\sum_i \sum_j P(i,j) log P(i,j)$
Contrast	$\sum_i \sum_j (i-j)^2 P(i,j)$
Homogeneity	$\sum_i \sum_j \frac{P(i,j)}{1+\lvert i-j \rvert}$

Haralick defined the GLCM as a matrix of frequencies at which two pixels, separated by a certain vector, occur in the image. The distribution in the matrix will depend on the angular and distance relationship between pixels. Varying the vector used allows the capturing of different texture characteristics. Once the GLCM has been created, various features can be computed from it. These have been classified into four groups: visual texture characteristics, statistics, information theory and information measures of correlation [7,3]. We chose the four most commonly used features, listed in Table 1, for our evaluation.

2.2 Tamura

Tamura et al took the approach of devising texture features that correspond to human visual perception [1]. They defined six textural features (coarseness, contrast, directionality, line-likeness, regularity and roughness) and compared them with psychological measurements for human subjects. The first three attained very successful results and are used in our evaluation, both separately and as joint values.

Coarseness has a direct relationship to scale and repetition rates and was seen by Tamura et al as the most fundamental texture feature. An image will

contain textures at several scales; coarseness aims to identify the largest size at which a texture exists, even where a smaller micro texture exists. Computationally one first takes averages at every point over neighbourhoods the linear size of which are powers of 2. The average over the neighbourhood of size $2^k \times 2^k$ at the point (x, y) is

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} f(i,j)/2^{2k} \quad .$$

Then at each point one takes differences between pairs of averages corresponding to non-overlapping neighbourhoods on opposite sides of the point in both horizontal and vertical orientations. In the horizontal case this is

$$E_{k,h}(x,y) = |A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)| \quad .$$

At each point, one then picks the best size which gives the highest output value, where k maximizes E in either direction. The coarseness measure is then the average of $S_{\text{opt}}(x,y) = 2^{k_{\text{opt}}}$ over the picture.

Contrast aims to capture the dynamic range of grey levels in an image, together with the polarisation of the distribution of black and white. The first is measured using the standard deviation of grey levels and the second the kurtosis α_4. The contrast measure is therefore defined as

$$F_{con} = \sigma/(\alpha_4)^n \quad \text{where} \quad \alpha_4 = \mu_4/\sigma^4 \quad ,$$

μ_4 is the fourth moment about the mean and σ^2 is the variance. Experimentally, Tamura found $n = 1/4$ to give the closest agreement to human measurements. This is the value we used in our experiments.

Directionality is a global property over a region. The feature described does not aim to differentiate between different orientations or patterns, but measures the total degree of directionality. Two simple masks are used to detect edges in the image. At each pixel the angle and magnitude are calculated. A histogram, H_d, of edge probabilities is then built up by counting all points with magnitude greater than a threshold and quantising by the edge angle. The histogram will reflect the degree of directionality. To extract a measure from H_d the sharpness of the peaks are computed from their second moments.

Tamura Image is a notion where we calculate a value for the three features at each pixel and treat these as a spatial joint coarseness-contrast-directionality (CND) distribution, in the same way as images can be viewed as spatial joint RGB distributions. We extract colour histogram style features from the Tamura CND image, both marginal and 3D histograms. The regional nature of texture meant that the values at each pixel were computed over a window. A similar 3D histogram feature is used by MARS [8].

2.3 Gabor

One of the most popular signal processing based approaches for texture feature extraction has been the use of Gabor filters. These enable filtering in the frequency and spatial domain. It has been proposed that Gabor filters can be used to model the responses of the human visual system. Turner [9] first implemented this by using a bank of Gabor filters to analyse texture. A bank of filters at different scales and orientations allows multichannel filtering of an image to extract frequency and orientation information. This can then be used to decompose the image into texture features.

Our implementation is based on that of Manjunath et al [10,11]. The feature is computed by filtering the image with a bank of orientation and scale sensitive filters and computing the mean and standard deviation of the output in the frequency domain.

Filtering an image $I(x, y)$ with Gabor filters g_{mn} designed according to [10] results in its Gabor wavelet transform:

$$W_{mn}(x, y) = \int I(x, y) g^*_{mn}(x - x_1, y - y_1) dx_1 dy_1$$

The mean and standard deviation of the magnitude $|W_{mn}|$ are used to for the feature vector. The outputs of filters at different scales will be over differing ranges. For this reason each element of the feature vector is normalised using the standard deviation of that element across the entire database.

3 Experimental Set Up

We followed a two-stage approach: Initial evaluation and modifications to the features were tested using a carefully selected subset of the Corel image library and the vector space similarity measure. We then ran larger tests on the TRECVID2003 data using the k-nearest neighbour measure (k-nn). We have a baseline for evaluation from previous work with the TREC dataset for which k-nn has consistently proved the best retrieval method.

Image Collections. We selected 6,192 images from the Corel collection to give 63 categories that were visually similar internally, but different from each other [12]. A set of 630 single-image category queries was executed to test performance across all categories. Relevance judgments on the retrieved images were based on the categorisation. The results shown in Section 4 are the mean average precision (m.a.p.).

A second larger image collection was used to give a more realistic performance comparison. This comprised of 32,318 key-frames from TRECVID2003 collection [13]. The search task specified for TRECVID2003 consisted of 25 topics, for each topic a few example images were given as a query. The published relevance judgments for these topics were used to evaluate the retrieval performance for different features and combinations of features.

Similarity Measures. Distances between feature vectors were calculated using the Manhattan metric. The resultant distances were then median normalised to give even weighting when combined. The plain vector space model was used for retrieval on the Corel data set as these involved only simple 1-image queries.

For querying the TREC data a version of the distance weighted k-nn approach was used [14], with $k = 40$. Positive examples (P) are supplied as the query and negative examples (N) randomly selected from the collection. To rank an image i in the collection we identify those images in P and N that are amongst the k-nearest neighbours of i. Using these neighbours we determine the dissimilarity:

$$D(i) = \frac{\sum_{n \in N} d^{-1}(i,n)}{\sum_{p \in P} d^{-1}(i,p)}$$

4 Evaluation and Results

For each feature we evaluated performance in the configuration described in Sect. 2. Ideas to improve performance were devised and evaluated. The general themes considered were how best to represent an entire image, how to accommodate differing sizes and scale of images and how to cope with the regional qualities of textures. These evaluations were run on the Corel data. Paired t-tests were carried out to check whether results were statistically significant at $\alpha = 0.05$.

The best performing features from the initial evaluation were then tested on the TRECVID2003 data set. Tests were run with each texture feature combined with a high performing colour feature.

4.1 Co-occurrence

The two main variables when creating a GLCM are the number of quantisation levels and the vector. We decided to use four vector angles: 0, 45, 90, 135 and four distances. This could be used to calculate up to sixteen GLCMs. However, as the statistics are not invariant under rotation we also tried summing the four angles at each distance into a single matrix. GLCMs can be made symmetrical by including the reverse vector; symmetric and asymmetric matrices were tested. The number of quantisation levels dictate the size of matrix and density of the matrix. This may become a problem with small images or tiles. The effect of varying quantisation between 4 and 64 levels was tried. Features were calculated for whole and tiled images.

Preliminary results showed that distances between 1 and 4 pixels gave the best performance. There was no significant difference between symmetrical and asymmetric matrices. Tiling of the image gave a large increase in retrieval which flattened out by 9×9 tiles. The results in Table 2 are for 7×7 tiles. Similarly increasing quantisation improves performance. The concatenated features (cat) gave better results at all points than the rotationally invariant summed matrices (sum). The best feature was homogeneity with a m.a.p. of 12.2%.

Table 2. Co-occurrence features — mean average precision retrieval

	Quantisation				
Feature	4	8	16	32	64
Energy: cat	7.63%	8.09%	9.30%	9.85%	9.54%
Energy: sum	7.04%	7.79%	8.85%	9.19%	8.96%
Entropy: cat	8.12%	9.22%	10.41%	11.09%	11.36%
Entropy: sum	7.54%	8.76%	9.79 %	10.37%	10.70%
Contrast: cat	8.46%	8.51%	8.35%	8.29%	8.28%
Contrast: sum	7.83%	7.85%	7.65%	7.59%	7.57%
Homogeneity: cat	9.17%	10.18%	11.16%	11.83%	12.19%
Homogeneity: sum	8.50%	9.52%	10.39%	10.93%	11.26%

4.2 Tamura

When calculating standard Tamura features for whole or tiled images the main variable is the k value for coarseness. This effect of varying this, and the number of tiles, can be seen in Table 3. The dashes in the table are where the image size resulting from tiling meant that the k value was too large to be used because of the border needed.

With the histogram features the main variable to evaluate was the window size. Coarseness can be calculated at a pixel level. However, both the directionality and contrast features operate over a region. A large window would smear the feature and lose resolution; conversely a small window may invalidate the statistical features, particularly if the directionality histogram is too sparsely populated. To evaluate this the features were run over several window sizes, creating a histogram for each feature.

A little surprisingly initial results showed that increasing the k value for coarseness reduced the performance — the optimum value was 2. This may be due to the large borders necessary for higher values of k. However, it is more likely caused by the nature of textures in images and the way the algorithm averages the 2^k values. There are unlikely to be textures with a coarseness of 64 or 32 pixels in a normal image. The algorithm may still detect noise at this dimension, biasing the average value of the feature. A change to the algorithm was made so that it took the values of k rather than 2^k — effectively introducing a logarithmic scaling of the coarseness and giving less influence to the larger scales. This gave a significant increase in performance for the histogram, from 6.1% to 10.1%, but no improvement when applied to the standard feature.

Performance of the directionality feature was poor. A detailed look at the operation of the algorithm showed that this was largely due to the sparse population of the histogram and subsequent difficulty in calculating valid variance of its peaks. Several options for improvement were tried including calculating global variance of the histogram and using entropy. The latter gave a substantial improvement, from 6.6% to 9.7%, for the standard feature but negligible effect on the histogram.

Table 3. Tamura features — mean average precision retrieval

	Standard features					Histogram features			
	Tiling					Window size			
Feature	1x1	3x3	5x5	7x7	9x9	2	4	8	16
Contrast	3.24%	6.08%	7.20%	8.07%	8.03%	5.96%	6.71%	7.01%	6.92%
Directionality: peak finding	2.91%	4.16%	5.02%	5.79%	6.64%	5.39%	5.59%	5.57%	4.93%
Directionality: entropy	2.74%	5.35%	7.45%	8.93%	9.73%	4.89%	4.37%	5.24%	5.43%
Coarseness-2: 2^k	4.42%	8.33%	9.48%	9.87%	9.91%	6.90%	5.99%	6.09%	6.01%
Coarseness-3: 2^k	3.54%	7.57%	8.79%	9.19%	9.02%	6.52%	5.85%	5.96%	5.83%
Coarseness-4: 2^k	3.49%	7.16%	7.68%	6.98%	—	6.12%	5.71%	5.64%	5.40%
Coarseness-5: 2^k	3.25%	5.74%	—	—	—	—	—	—	—
Coarseness-6: 2^k	2.92%	—	—	—	—	—	—	—	—
Coarseness-2: k	4.43%	7.96%	9.32%	9.57%	9.59%	6.44%	9.98%	9.83%	8.22%
Coarseness-3: k	3.91%	7.50%	8.92%	9.10%	8.94%	5.68%	10.08%	9.24%	7.93%
Coarseness-4: k	3.41%	6.95%	7.74%	7.15%	—	8.81%	9.33%	8.12%	7.67%

Finally the combined marginal and 3D histograms were evaluated using a window size of 8, k of 3 and entropy directionality. In addition a combined feature vector of the 3 standard features was evaluated. The m.a.p. results were: marginal histogram 12.0%, 3D histogram 13.7% and standard 14.3%. All gave a significant improvement over the single features.

4.3 Gabor

Sect. 2.3 describes the generation of this feature. However, there still remain questions over how to apply it to a heterogeneous set of images. The problems of scale, varying size and so on apply. The evaluation in [10] was applied to fixed tiles extracted from the Brodatz album. In [11] the feature was used successfully with aerial photographs split into a large number of fixed size tiles and then querying to find individual tiles. We decided to evaluate the feature in two configurations across a range of scale and orientation values. The first scaled the filter dictionary to the size of the image. This should scale the response so that the same image of different size gives a similar value. The second approach was to use a fixed size filter and apply this to a sliding window over the image.

Initial results showed that scaling the filter size gave much superior results to the sliding window approach. Tiling increased performance in a similar manner to the other features. The results shown in Table 4 are for 7×7 tiling. The best performance is obtained from just 2 scales and 4 orientations. This was unexpected as most literature recommends 4 scales and 6 orientations. Looking at the filtered images indicated that, as for Tamura, this may be due to noise at coarser scales.

4.4 Evaluation Using TRECVID2003 Video Data

A range of the best performing features were run on the TRECVID2003 data and evaluated using the published relevance judgments. The queries were run singly

Table 4. Gabor wavelets — mean average precision retrieval

Scale	Orientation		
	3	4	6
2	13.1%	14.0%	13.9%
3	11.0%	11.4%	11.3%
4	10.8%	11.4%	11.2%

and then combined with a colour histogram feature, HSV [12]. The results are shown in Table 5. For comparison some features used for previous evaluations [12] gave m.a.ps of: HSV 1.9%, convolution 2.2% and variance 1.7%; random retrieval would give 0.26%.

In this evaluation the texture features performed extremely well in comparison with previous benchmarks. Gabor gave the best results, 3.9% or 15 times better than random retrieval. Of the Tamura features the best performing was the combined standard features. The top 3 performing texture features combined and giving a m.a.p of 4.22%.

Combining with the HSV feature improved average retrieval performance in all cases, but at an individual query level the benefits were both positive and negative. It is interesting that using simple combination of features gives varying degrees of improvement; being able to choose the optimum combination based on the query would be beneficial.

Table 5. TREC evaluation — mean average precision retrieval

Feature	Single	Combined with HSV
gabor-2-4	3.93%	4.31%
co-occurence homogeneity	2.85%	3.03%
tamura standard all	2.57%	3.43%
tamura CND	1.65%	2.72%
tamura coarseness-2	0.97%	2.49%

5 Conclusions

We selected 3 different texture features, implemented and evaluated them. Both the evaluation and implementation focussed on query-by-example image retrieval rather than the usual classification task.

This led to some novel modifications to the Tamura features. We found that looking for large scale coarseness degraded performance, so we limited the range and used a logarithmic scale. An improvement in directionality performance over small window sizes was achieved by using an entropy measure rather than taking

the second moments of the peaks. We also encoded the features in terms of joint histograms, the overall performance of these was similar to the standard features.

To improve the retrieval with Gabor we scaled the filter size to that of the image, rather than using a fixed size filter. Rather unintuitively we found that fewer scales gave higher retrieval rates. Our tests of co-occurrence matrices showed a solid performance — as expected!

Our evaluation with TRECVID2003 data showed that the top 3 texture features performed better than previously used colour features. Combination with a colour feature boosted retrieval performance in all cases. Overall we have demonstrated that we have produced robust texture features for image retrieval.

We would like to carry out further evaluations on larger data sets, particularly investigating the interaction of different feature combinations. Finally, texture features have an advantage over colour features in that performance should be the same for monochrome images. It would be interesting to perform an evaluation on a library of black and white pictures.

Acknowledgement. This work was partially supported by the EPSRC, UK.

References

1. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. IEEE Trans on Systems, Man and Cybernetics **8** (1978) 460–472
2. Ohanian, P., Dubes, R.: Performance evaluation for four classes of textural features. Pattern Recognition **25** (1992) 819–833
3. Gotlieb, C.C., Kreyszig, H.E.: Texture descriptors based on co-occurrence matrices. Computer Vision, Graphics and Image Processing **51** (1990) 70–86
4. Jain, A.K., Farrokhnia, F.: Unsupervised texture segmentation using gabor filters. Pattern Recognition **23** (1991) 1167–1186
5. Randen, T., Husøy, J.H.: Filtering for texture classification: A comparative study. IEEE Trans on Pattern Analysis and Machine Intelligence **21** (1999) 291–310
6. Brodatz, P.: Textures: A Photographic Album for Artists & Designers. Dover (1966)
7. Haralick, R.: Statistical and structural approaches to texture. Proceedings of the IEEE **67** (1979) 786–804
8. Ortega, M., Rui, Y., Chakrabarti, K., Mehrotra, S., Huang, T.S.: Supporting similarity queries in MARS. In: ACM Multimedia. (1997) 403–413
9. Turner, M.: Texture discrimination by Gabor functions. Biological Cybernetics **55** (1986) 71–82
10. Manjunath, B., Ma, W.: Texture features for browsing and retrieval of image data. IEEE Trans on Pattern Analysis and Machine Intelligence **18** (1996) 837–842
11. Manjunath, B., Wu, P., Newsam, S., Shin, H.: A texture descriptor for browsing and similarity retrieval. Journal of Signal Processing: Image Communication **16** (2000) 33–43
12. Pickering, M., Rüger, S.: Evaluation of key-frame based retrieval techniques for video. Computer Vision and Image Understanding **92** (2003) 217–235
13. Alan Smeaton, W.K., Over, P.: TRECVID 2003 — An introduction. In: TRECVID 2003 Workshop. (2003) 1–10
14. Mitchell, T.M.: Machine Learning. McGraw Hill (1997)