The Multimodal Signature Method : An Efficiency Study*

D. Koubaroulis[1,2] J. Matas[1,2] J. Kittler[1]

[1]Centre for Vision Speech and Signal Processing
University of Surrey, Guildford, GU2 7XH , UK
d.koubaroulis@ee.surrey.ac.uk

[2]CMP, CTU Prague
Karlovo nám. 13, 121 35 CZ
matas@cmp.felk.cvut.cz

Abstract

The multimodal neighbourhood signature (MNS) method has presented acceptable results both for the colour-based image retrieval and the object recognition task. Local colour content is concisely represented by invariant features computed from neighbourhoods with multimodal colour density function.

In this paper, efficiency related issues regarding the MNS algorithm are investigated. Its performance, speed, sensitivity and storage requirements are tested on a standard colour object recognition experiment. Very good recognition rate (99.9%) was achieved in real time (0.28 seconds per match). MNS signatures are concise and thus significant data reduction is achieved. The complexity of signature computation and matching are analysed and efficient implementations are proposed.

1 Introduction

The Multimodal Neighbourhood Signature (MNS) approach [5] addresses the colour indexing task by computing colour features from local image neighbourhoods with multimodal colour probability density function. A robust mode estimator, the mean shift algorithm [2], is used to detect the modes of the density function. From the mode colours a number of local invariant features are computed, depending on the adopted model of colour change. Under different assumptions, the resulting multimodal neighbourhood signatures consist of colour ratios, chromaticities, raw colour values or combinations of the above. The advantages of the proposed algorithm and further details can be found in [5].

In our previous work, the multimodal neighbourhood signature algorithm has presented good results for a number of image retrieval experiments [5, 4]. In this application area, colour histograms methods are a de-facto standard. In image retrieval, speed is a very important performance characteristic and any method aspiring to challenge the dominance of the histogram-based approaches must have comparable run-time. Especially for web-based applications, where a comparatively large number of image signatures need to be computed on-line, efficiency is highly desired. In addition, retrieval from large image databases or video sequences, as well as object recognition in real time, require very fast signature matching and low storage requirements.

In the work reported in this paper we focus on efficiency related issues of the MNS method. The computation speed for both signature creation and matching was evaluated both theoretically and experimentally. Signature size is also an important factor that influences matching and needs to be considered in the context of web-based retrieval systems. Robustness of the algorithm in terms of its internal parameters is also significant for applications working on images of scenes with diverse colour content.

A brief outline of the computation of a MNS signature is given in section 2 and the matching technique is discussed in section 3. Section 4 presents details of the experimental setup and the results obtained are presented in section 5. The results are discussed in section 6 and section 7 concludes the paper.

2 Computing the MNS signature

The image plane is covered by small compact neighbourhoods of rectangular shape (chosen for convenient image processing since the actual neighbourhood shape

*The research was supported EC fifth framework project IST-13082 ASSAVID. The second author was supported by the Czech Ministry of Education under the grant VS96049. The first author acknowledges support from the Digital VCE, the EPSRC and the I. Latsis Foundation.

is not critical for our application). For every image neighbourhood defined by a randomised grid, the modes of the colour density function are located in the RGB space with the mean shift algorithm (for details see [5]). Modes with relatively small coverage are ignored as they usually represent noisy information. The neighbourhoods are then categorised according to their modality as unimodal, bimodal, trimodal etc. For the computation of the colour signature only multimodal neighbourhoods are considered. For every pair of mode colours m_i and m_j in each neighbourhood, a vector $v = (m_i, m_j)$ is constructed in a joint 6-dimensional domain denoted RGB^2.

For an image of dimensions $M \times N$ and a grid spacing $m \times n$, the number of neighbourhoods considered over the image is MN/mn. For each neighbourhood a mean shift search for the closest mode is started from a subset s of the neighbourhood pixels using all the pixels of the neighbourhoods as data points. The complexity of mode seeking per pixel is $\mathcal{O}(mni)$ where i is the average number of iterations required for convergence. Therefore, the total computational cost for processing all neighbourhoods is $\mathcal{O}(mni) \times \mathcal{O}(k_s) \times (MN/mn) = \mathcal{O}(MNk_s i)$, where $k_s = |s|$. Taking into account that the average number of iterations is typically 3-5 and k_s is usually a small number, the processing cost is approximately proportional to the size of the image. Computation speed can be increased by requiring a minimum distance of a kernel width between two adjacent modes of the density function. Then, unimodal neighbourhoods can be identified simply by the fact that all colour values in the neighbourhood fall inside the kernel at convergence for some starting point/

In order to compute a concise image descriptor, the colour pairs are clustered in the RGB^2 space and a representative vector for each cluster is stored. The proposed colour signature consists of the modes of the colour-pair distribution. For the clustering, the mean shift algorithm is applied once more to locate the local maxima. The arithmetic complexity of the clustering is $\mathcal{O}(k_m p i)$ where k_m is the number of multimodal neighbourhoods in the image, p is the average number of colour pairs per multimodal neighbourhood and i is the average number of iterations for a mean shift search to converge. Clearly, since both the number of multimodal neighbourhoods per image and their modality is relatively small, the processing cost is not significantly affecting the total signature computation time.

Finally, the computed signature consists of a number of RGB^2 vectors depending on the complexity of the colour structure in the scene. In general, the resulting structure is very concise and flexible. From the MNS signature a number of invariants has been proposed to enable recognition under changing geometrical and illumination conditions [5].

3 Matching MNS Signatures

Let Q, D be a pair of signatures of a query model image (or region) and a database image respectively. We assume that the model signature contains information only about the object of interest. This assumption is realistic since in both recognition and retrieval applications special care is taken to compute a model signature either by using a uniform background for the model images (in recognition) or by manual query delineation (in retrieval). Therefore our task is to interpret each -ideally all- model features as a distorted instance of a unique feature of the database signature.

As detailed in our earlier work, each signature consists of a set of features $Q = \{f_Q^i : i = 1..m\}$ and $N = \{f_N^j : j = 1..n\}$ where $m, n \in \mathbb{N}$. For every pair f_Q^i, f_N^j the distance $d(f_Q^i, f_N^j) \equiv d_{ij}$ in the feature space is computed and assigned to it. Note that for our application the distance is symmetric ($d_{ij} = d_{ji}$).

MNS matching is an assignement problem, i.e. a problem of uniquely associating each quary feature to a test feature. We define a match association function $u(i) : Q \to 0 \bigcup D$, mapping each model feature i to the database (test) feature it matched or to 0 if it did not match. Similarly, a test association function $v(j) : D \to 0 \bigcup Q$, $j \in J$, maps a database feature to a query feature or to 0 in case of no match. A threshold T_h is used to define the maximum allowed distance between two matched features. x

Viewing Q, D as a bipartite graph and the distances d_{ij} as edge weights the problem is identical to finding the minimum flow for a weighted bipartite graph (with the introduction of dummy vertices to transform the problem of matching in to that of a network flow and to account for the asymmetry in the sets to be matched). This problem can be solved in $\mathcal{O}(V^3)$ time where V is the number of vertices in the graph [9].

In current implementation, we require that the matching is a model-oriented stable matching problem [1]. The main idea is that a match is established between a model and the *closest* (in terms of the specified distance function) database feature to it that is not closer to any other model feature and within the maximum allowed distance.

Algorithm 1: **MNS Matching**

1. Set $u(i) = 0$ and $v(j) = 0$ $\forall i, j$.

2. From each signature s compute the invariant features f_i^M, f_j^T according to the colour change model dictated by the application.

3. Compute all pairwise distances $d_{ij} = d(f_i^M, f_j^T)$ between the query and test features.

4. Set $u(i) = j, v(j) = i$ if $d_{ij} < d_{kl}$ and $d_{ij} < T_h \quad \forall\, k, l$ with $u(k) = 0$ and $v(l) = 0$.

5. Compute signature dissimilarity as
$D(s^M, s^T) = \sum_{(\forall\, i : u(i) \neq 0)} d_{ij} + \sum_{(\forall\, i : u(i) = 0)} T_h$

The complexity of finding a stable matching is linearly dependent on the number of all pairs input to the algorithm [9]. The algorithm we use for matching MNS signatures is a modified version of an implementation based on a sorted list (proposed in [8]). The modifications were necessary to account for unmatched features and the assymetry in the size of the sets to be matched. A further reduction in the time required to compute the dissimilarity score is achieved by ignoring all pairs with distance greater than the maximum distance threshold value.

4 Experimental Setup

To compare MNS performance with results reported in the literature, we performed a well known colour object recognition experiment using a publicly available dataset collected by M. Swain on which several algorithms have been previously tested (e.g. [10, 3, 7]). The model image set consisted of 66 household objects imaged on black background under the same light (for a full colour image of the database see [10]) . The test set consisted of 32 images, a subset of model objects rotated, displaced or deformed (e.g. clothes).

Performance evaluation was identical to Swain's [10]. Funt and Finlayson [3] introduced ratio histogram matching for recognition. However, for that experiment, 11 model and 8 test images were removed from the database due to saturated pixels whereas we used all images. The same experiment was repeated by Park et al. [7] using a colour adjacency graph representation of image colour structure.

The MNS matching algorithm was implemented in C++ and tested on a SUN Ultra Enterprise 450 with quad 400MHz UltraSPARC-II CPUs. Computation of each signature took on average 0.1 seconds. Image size was 90×128 pixels for both the model and test image sets. No image preprocessing, sub-sampling or smoothing was applied before signature computation. All internal parameters (mean shift kernel width, neighbourhood size etc) were set to default values, that is, they were not tuned for Swain's database. Image retrieval results with the same settings were presented in [4]. The average signature size was 150 bytes [6].

Method	Rank				Av. Match
	1	2	3	>3	Percentile
MNS (Default)	27	2	2	1	0.995
CC Colour Indexing	22	2	0	0	0.998
Colour Indexing	29	3	0	0	0.999
MNS (Swain)	29	3	0	0	0.999
Hybrid graph	32	0	0	0	1.000

Table 1. Comparative colour object recognition results for Swain's database

5 Results

Reported results using 6D RGB^2 feature matching and default parameter setting are shown in Table 1. Each test object signature was matched against all 66 model signatures. For a single test object the matching process took on average 0.28 sec, i.e. 4 msec per match. Our method compares favourably to other 3 algorithms with reported results for the same task even when applied with its default parameters. In particular, when an appropriate mean shift kernel width for Swain's database was selected, a recognition performance of 99.9% was achieved.

Among the objects that were not classified as rank 1, are mostly objects with very similar red-white colour boundaries which are very common in Swain's database. Perfect recognition performance was achieved only by the hybrid graph which makes use of localised feature matching. An extension to the proposed MNS algorithm to localise matched colour features by effectively using spatial information between and within image neighbourhoods, is under development.

6 Efficiency considerations

The sensitivity of the MNS algorithm was tested experimentally and the results showed that recognition rate was not significantly affected by the selection of the L-metric distance. The number of test objects that were ranked up to rank 6 and above for a number of different metrics are presented in Table 2.

	Rank							Av. Match
Metric	1	2	3	4	5	6	>6	Percentile
L_1	27	2	2	0	1	0	0	0.995
L_2	27	2	2	0	0	1	0	0.994
L_3	27	2	2	0	0	0	1	0.993
L_∞	27	1	3	0	0	0	1	0.993

Table 2. Recognition for different L-metrics

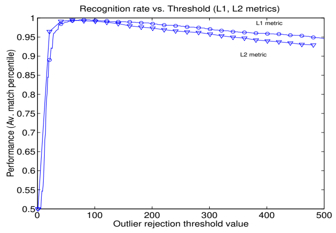

Figure 1. Recognition for different thresholds

The marginally better result for the L_1 metric was most probably due to the more robust behaviour of the function to outliers. Note that the time to compute the L_1 and L_∞ distance scores is minimum compared to other L-metrics requiring calculation of a p-th order root at each run. Ideally, the distance function should be *learned* from the colour content properties over a set of training images.

The MNS matching algorithm uses a single threshold value T_h to achieve robustness to outliers in the computation of the dissimilarity value. The method is shown to be insensitive to a wide range of T_h values. Recognition performance deteriorated slowly (Fig. 1) even for extreme values of T_h and converged to a performance limit of 80% for extremely large (practically infinite) thresholds, using the $L_1 - norm$ and the $L_2 - norm$ for feaure distance computation.

Experimentally, it was established that, besides the computational complexity of the matching strategy, the time required for matching two signatures is dominated by the computation of the distances between the feature pairs. An approach that does not require computation of all the pairwise distances to establish signature dissimilarity is being investigated.

Another important parameter of a retrieval system, especially when operating on images on the World Wide Web, is the space required to represent a single image. The huge number of images that will potentially be indexed, dictates the need for concise image descriptions apart from fast signature computation. The MNS method is very competitive representing each image with a small number of 6D vectors depending only on the complexity of the scene. The space needed to store one RGB^2 vector is 6 bytes using fixed point arithmetic to represent the (originally float) mode values. Therefore the size of a MNS signature is $6n$ bytes, where n is the number of RGB^2 vectors extracted from the image. The number of colour pairs depends, besides the data, on parameters of the signature construction stage. For the default MNS settings average signature size for Swain's images was 150 bytes with a maximum of 2.5 Kb for the most complex scenes. This, for example, allows for 1 million MNS signatures to be stored on a hard disk of size approximately 200 Mb.

7 Conclusions

In this paper, we focused on efficiency related issues of the MNS method. The speed and algorithmic complexity of both signature computation and matching was investigated. Multimodal signatures were computed in 0.1 sec on average on a SUN Ultra Enterprise 450 machine. Average signature size was small, afew hundred bytes, which makes the method competitive for applications with fast matching and low storage requirements.

We tested the algorithm's performance on a standard colour object recognition task using a publicly available dataset. Very high recognition rate (average match percentile 99.9%) was achieved in real time (0.28 msec per match) which compares favourably with 3 other reported results for the same task. Recognition rate was fairly insensitive to large changes of the robustness threshold and the distance function in the feature space for a number of common Minkowski metrics (e.g. L_1, L_2, L_∞).

References

[1] G. D. *The Stable Marriage Problem:Structure and Algorithms.* MIT Press, 1989.

[2] K. Fukunaga and L. Hostetler. The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. In *IEEE Transactions in Information Theory*, pages 32–40, 1975.

[3] B. Funt and G. Finlayson. Color Constant Color Indexing . *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):522–529, 1995.

[4] D. Koubaroulis, J. Matas, and J. Kittler. Colour-based Image Retrieval from Video Sequences. In *The Challenge of Image Retrieval, Brighton, UK (to appear)*, 2000.

[5] J. Matas, D. Koubaroulis, and J. Kittler. Colour Image Retrieval and Object Recognition Using the Multimodal Neighbourhood Signature . In *Proceedings of the 6th European Conference in Computer Vision, Dublin, Ireland (to appear)*, 2000.

[6] J. Matas, D. Koubaroulis, and J. Kittler. Performance Evaluation of the Multi-modal Neighbourhood Signature Method for Colour Object Recognition. In *Proceedings of the Czech Pattern Recognition Workshop, Perslac, Czech Republic*, pages 27–34, 2000. (available at http://www.ee.surrey.ac.uk/Personal/-D.Koubaroulis/thesis/pub/matas-cprw00.ps.gz).

[7] K. Park, I.-D. Yun, and S. U. Lee. Color Image Retrieval Using a Hybrid Graph Representation. *Journal of Image and Vision Computing*, 17(7):465–474, 1999.

[8] R. Sara. The Class of Stable Matchings for Computational Stereo. Technical Report CTU-CMP-1999-22, Czech Technical University, 1999.

[9] R. Sedgewick. *Algorithms* . Addison-Wesley, 1988.

[10] M. J. Swain and D. H. Ballard. Color Indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.