Texture classification through combination of sequential colour texture classifiers

Francesco Bianconi[1], Antonio Fernández[2], Elena González[2] and Fernando Ribas[3]

[1] Università degli Studi di Perugia, Dipartimento Ingegneria Industriale
Via G. Duranti 67 - 06125 Perugia, Italia
bianco@unipg.it

[2] Universidade de Vigo, Departamento de Diseño en la Ingeniería
E.T.S.I.I. - Campus Universitario, 36310 Vigo - España
{antfdez,elena}@uvigo.es

[3] Universidade de Vigo, Departamento de Física Aplicada
E.U.I.T.I. - Torrecedeira 86, 36208 Vigo - España
fribas@uvigo.es

Abstract. The sequential approach to colour texture classification relies on colour histogram clustering before extracting texture features from indexed images. The basic idea of such methods is to replace the colour triplet (RGB, HSV, Lab, etc.) associated to a pixel, by a scalar value, which represents an index of a colour palette. In this paper we studied different implementations of such approach. An experimental campaign was carried out over a database of 100 textures. The results show that the choice of a particular colour representation can improve classification performance with respect to grayscale conversion. We also found strong interaction effects between colour representation and feature space. In order to improve accuracy and robustness of classification, we have tested three well known expert fusion schemes: weighted vote, and a posteriori probability fusion (sum and product rules). The results demonstrate that combining different sequential approaches through classifier fusion is an effective strategy for colour texture classification.

1 Introduction

Texture analysis is recognized as a key point in the development of artificial vision systems. Within texture analysis, classification is a major research topic, due to the numerous applications in areas like medical imaging, remote sensing, quality control and others. Texture classification techniques are very attractive for industrial applications, especially in those situations where it is important to group products in lots according to the criterion of "same visual appearance". In many industrial areas there is a growing interest in systems capable of performing such kind of tasks automatically.

Texture classification involves two major processes: feature extraction and label assignment. The whole formed by these two building blocks is usually referred to as an *expert*. It is commonly accepted that substantial gain in classification performance can be obtained by combining the results of individual experts [1, 2]. In this work we adopted different combination schemes for sequential colour texture classification. The most innovative contributions of this paper are: on the one hand, the use of colour indexing methods that have not been implemented yet in colour texture classification by sequential approaches, and, on the other hand, the combination of sequential colour texture classifiers by classifier fusion.

The remainder of the paper is organized as follows: section 2 describes the colour indexing approach to texture classification. Feature spaces and classifiers used in this work are described in section 3. Combination of experts is detailed in section 4. The experimental activity is described in 5 and its results are presented and discussed in section 6. Final conclusions are reported in section 7.

2 Colour representation

Several attempts have been made to incorporate colour and texture features during the last years. Up to now, there has been no general consensus about the best way to combine these two properties. It is widely accepted that taking into account colour in texture classification can provide additional information [3]. However some authors argue that colour and texture have to be regarded as separate phenomena [4]. According to Palm [5], the approaches to combine colour and texture can be grouped in parallel, sequential and integrative approaches. In the *parallel* approach, textural features extracted from the luminance plane are considered together with pure chrominance features. *Sequential* methods involve colour histogram clustering before extracting texture features from indexed images. *Integrative* models characterize a texture through spatial interaction within each color plane and between different colour planes.

In this paper we focus on sequential methods. The basic idea is to replace the colour triplet (RGB, HSV, Lab, etc.) by a scalar value, which represents an index of a colour palette. This is usually referred to as *colour indexing*. The selection of a particular technique for colour histogram clustering should be done carefully, since it strongly influences the ability of the features extracted from

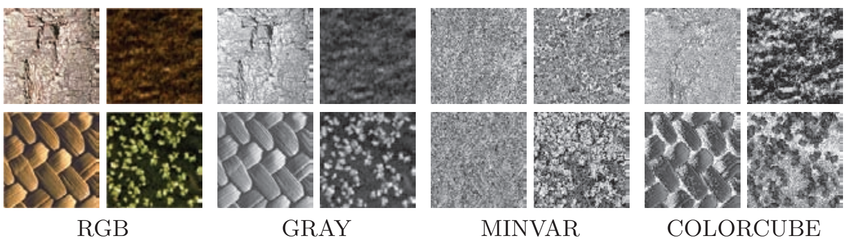

Fig. 1. From left to right: original RGB image; grayscale conversion; minimum variance quantization, colorcube colourmap mapping.

the indexed images to describe colour texture, no matter the feature space considered. Although there is a vast amount of work on the integration of texture and colour in a unique model, few implementations of the sequential scheme have been reported. Song [6] proposed an approach to defect detection in colour textures based on k-means clustering and perceptual merging. More recently, Arvis [7] applied uniform quantization of the 3D colour histogram to texture classification. Uniform quantization involves dividing the color cube into a number of equal-sized boxes. The effects of representing the original images through different colour spaces as well as the effects of varying the number of colour indices have been studied in [8].

Herein we adopted the sequential approach to colour texture analysis, also referred to as chromato-spatial approach. In addition to the classical grayscale conversion, we propose *minimum variance quantization* and *colorcube colourmap mapping* as colour indexing techniques. Different colour representations are likely to produce diverse descriptions of textural data, and thus it makes sense to integrate them through classifier fusion. One can easily realize from figure 1, that the transformed images look significantly different from the original RGB images. Nevertheless, textural data are not lost: they are rather stored in a different way, as it comes out from the results shown in section 6. Based on such idea, we integrated colour indexing methods together with grayscale conversion through different classifier fusion architectures.

In minimum variance quantization the RGB color cube is recursively subdivided into smaller volumes of different sizes (not necessarily cubes). The size of each cluster depends on the distribution of colours in the image [9]. In contrast, colourmap mapping uses a predefined colourmap. Each pixel of the indexed image is then assigned the index of the cluster that contains the colour of the pixel. Applying minimum variance quantization to each image separately does not seem a promising approach, since the meaning of the resulting indices would change from one image to another. Instead, we compute the minimum variance colour map by quantizing the colour distribution of the whole image database (fig. 2). On the other hand, we have chosen Matlab's *colorcube* mapping [10] since it contains as many regularly spaced colours in the RGB space as possible, and thus it can work well in the majority of the situations.

3 Classification framework

3.1 Feature extraction

The original RGB images have been converted to single-channel images as described in the previous section. Texture features have been extracted from single-channel images using *Coordinated Clusters Representation* (*CCR*), *Local Binary Patterns* (*LBP*) and *Gabor filters*.

CCR and LBP features represent texture through the histogram of 3x3 binary patterns [11, 12]. The only difference between LBP and CCR texture models is that LBP employs a local binarization threshold while CCR uses a global one.

In this work we used as binarization threshold the gray level (or colour index) which splits the entropy of the histogram of a single-channel image into two equal parts. This technique is based on the isentropic quantization approach, which has been successfully applied in the knowledge extraction stage of the construction of fuzzy sets [13]. The dimension of the CCR and LBP feature space is 512 and 256, respectively.

Gabor features consist of the mean and standard deviation of the output of a filter bank applied to the input image. Based on the result of previous work [14, 15], we adopted here a filter bank with 4 frequencies and 6 orientations. The dimension of the associated feature space is 48.

3.2 Label assignment

Label assignment (usually referred to as *classification*), is about assigning a class label to an unknown texture. Many different approaches have been proposed in literature. For a comprehensive review readers are referred to references [16–18]. Herein we adopted the well known nearest neighbour approach, which assigns a pattern the class label of the nearest labeled pattern in the feature space.

4 Combination of experts

Combination of multiple experts has recently emerged as a major topic in pattern analysis and machine intelligence. Though numerous approaches have been proposed and tested, they can be well classified in two main families: *fusion of label outputs* and *fusion of continuous-value outputs* [1, 2].

In the first scenario each expert e_k returns, for each point $\mathbf{x}$ in the feature space, a class label j:

$$e_k(\mathbf{x}) = j; \qquad \begin{cases} k = 1, ..., K \\ j \in \{1, ..., n\} \end{cases} \tag{1}$$

where K is the number of experts and n is the number of classes.

In the second scenario each expert produces, for each point $\mathbf{x}$, a vector of *a posteriori* probabilities for that point to pertain to one of the possible classes:

$$e_k(\mathbf{x}) = [P_k(\omega_1|\mathbf{x}), ..., P_k(\omega_n|\mathbf{x})] \,. \tag{2}$$

Fusion of label outputs is usually based on some voting scheme: majority vote or weighted majority vote. In the first approach it is assumed that all the experts are of identical accuracy. In this case each expert gives the same contribution to the final decision. Weighted voting, instead, tries to give the more competent experts more power in taking the final decision. Weights are usually based on some *a priori* knowledge of experts accuracy.

Three different strategies to combine multiple experts have been considered here: weighted vote, and fusion of *a posteriori* probabilities based on sum and product rule.

4.1 Weighted vote

For weighted vote to be applied, we need a way to estimate the reliability of each single expert. The accuracy of each expert can be evaluated through its confusion matrix [19, 20]. The r_{ij}^k element of the confusion matrix represents the number of samples of class ω_i that have been classified of class ω_j by the expert e_k. In a perfect expert all the elements outside the principal diagonal of the matrix should be zero. Given the confusion matrix R_k of an expert e_k, an event $e_k(\mathbf{x}) = j$ can be described in terms of the conditional probabilities that the propositions $\mathbf{x} \in \omega_i$ are true when the event $e_k(\mathbf{x}) = j$ occurs:

$$P(\omega_i | e_k(\mathbf{x}) = j) = \frac{r_{ij}^k}{\sum_{i=1}^{n} r_{ij}^k} \tag{3}$$

In practice each event $e_k(\mathbf{x}) = j$ gives a different support (or weighted vote) to each hypothesis $\mathbf{x} \in \omega_i$, $i = \{1, ..., n\}$. The total support $S(\omega_i)$ of a proposition $\mathbf{x} \in \omega_i$ given a set of events $e_k(\mathbf{x}) = j$, $j = \{1, ..., n\}$ and $k = \{1, ..., K\}$, is simply computed as the sum of the support of each classifier:

$$S(\omega_i) = \sum_{k=1}^{K} P(\omega_i | e_k(\mathbf{x}) = j) \tag{4}$$

The vector $\mathbf{x}$ is then assigned the label with the highest support.

The confusion matrix needs to be computed before classifying. Here we estimate the confusion matrix of each classifier through cross-validation using the points of the training set.

4.2 Fusion of *a posteriori* probabilities

When different experts provide a posteriori class probabilities , such values can be combined in different ways to provide a label output. Despite various approaches have been proposed to this purpose, the simple *sum* and *product* rules have been recognized as reliable and robust [2, 20]. A pattern $\mathbf{x}$ is assigned the label j which maximizes the sum (product) of the a posteriori probabilities provided by each expert (eq. 5 and 6).

$$j = \underset{i \in \{1,...,n\}}{argmax} \left(\sum_{k=1}^{K} P_k(\omega_i | \mathbf{x}) \right) \tag{5}$$

$$j = \underset{i \in \{1,...,n\}}{argmax} \left(\prod_{k=1}^{K} P_k(\omega_i | \mathbf{x}) \right) \tag{6}$$

In order to quantify a posteriori probabilities (sometimes referred to as *memberships*), it seems natural to adopt a distance-based normalized similarity measure: the less the distance between a test point and the nearest labeled neighbour, the highest the probability for that point to belong to the same class of the closest labeled point. We adopted here the following membership:

Fig. 2. Experimental dataset.

$$P(\omega_i|\mathbf{x}) = \frac{\dfrac{1}{1+d(\mathbf{x},\overline{\mathbf{x}}_i)}}{\displaystyle\sum_{j=1}^{n}\frac{1}{1+d(\mathbf{x},\overline{\mathbf{x}}_i)}} \tag{7}$$

where d is a generic distance function, and $\overline{\mathbf{x}}_i$ is the pattern of class ω_i closest to $\mathbf{x}$ in the feature space. Equivalent formulations have been proposed by other authors [21, 22]. In this work we adopted the L_1 (Manhattan) distance.

5 Experimental activity

Combined classifiers have been set up using the different colour conversion approaches described in section 2 and the CCR, LBP and Gabor feature spaces. The performance of each single expert and of their combinations has been evaluated over a database of 100 texture classes (fig. 2). Each texture image has been divided into 16 sub-images, resulting in 1600 texture samples. To assess expert performance, we considered the percentage of correctly classified textures. Classification error has been evaluated by split-half validation with stratified sampling

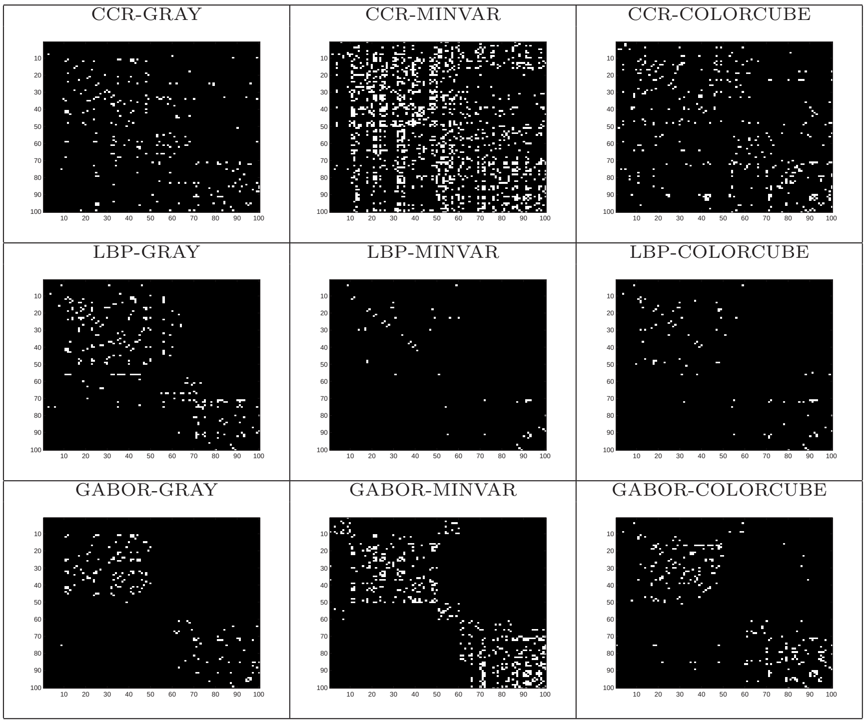

Fig. 3. Simplified representation of the confusion matrices $R_k(i,j)$ of the various classifiers. For visualization purposes the main diagonal of each confusion matrix has been set to 0 (white). Each black point indicates that the k-th classifier makes at least one mistake in classifying a patterns of class i as a pattern of class j.

[23]. The error is averaged over 100 random partitions of data into training and validation set in order to make the estimation stable.

6 Results and discussion

The results (table 1) of the experimental activity are suggestive of interesting considerations. First, it appears that the choice of a particular colour representation has significant effects on texture classification. It is worth noticing that switching from grayscale conversion to minimum variance quantization improves performance in the LBP feature space (87,37 % → 97,27 %), but it drastically reduces it in the CCR feature space (87,82% → 53,07 %). Second it results that combining multiple experts provides substantial gain in classification performance. The percentage of correct classification shows significant increase either by adopting different feature spaces -as one could expect- or, more interestingly, by using different colour representations and the same feature space. The best

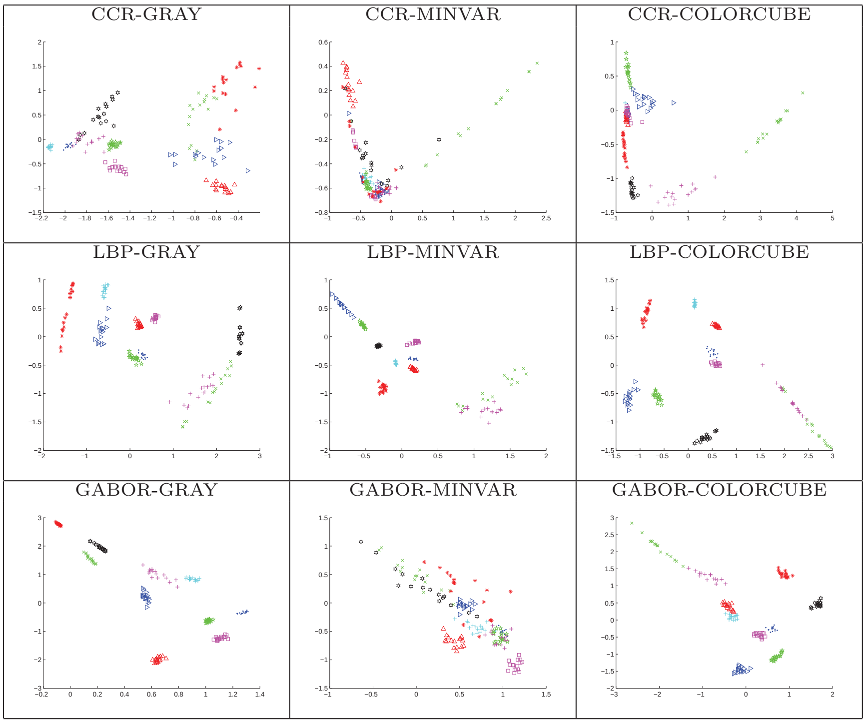

Fig. 4. Representation of the first ten texture classes of the data set of fig. 2 in the first-two principal components space.

performance is achieved when all the nine possible combinations are employed. The performance of the best combined expert approaches 100 %. Another interesting result is that classifier fusion appears a robust approach: even if we include a classifier that provides poor results (i.e. CCR+MINVAR), the global perfomance is usually better than that of the best clasifier. Only in two cases we have a very slight reduction: GRAY+MINVAR+COLORCUBE (LBP), from 97.27% to 97.23% (probability fusion, sum rule), and from 97.27% to 97.24% (probability fusion, product rule). The results obtained with the three different fusion architectures are essentially the same. Therefore the above conclusions are valid all the fusion schemes considered in this paper.

7 Conclusions

Fusion of classifiers is supposed to work well when there is a reasonable difference among the classifiers, or, in other words, when the classifiers do not make the same mistakes. It is well known that LBP, CCR and Gabor features produce different representations of textures, as we can appreciate in figures 3 and 4.

Table 1. Performance of single experts and different combinations of experts (expressed as percentage of correct classification). The numerical results corresponding to different fusion schemes are shown in different fonts. Normal font: weighted vote; *italics*: a posteriori probability fusion (sum rule); **boldface**: a posteriori probability fusion (product rule).

	CCR	LBP	GABOR	CCR+ LBP+ GABOR
GRAY	87,82	87,37	88,86	96,40 *95,82* **95,88**
MINVAR	53,07	97,27	78,84	97,88 *98,15* **97,94**
COLORCUBE	87,50	96,14	89,48	97,45 *97,57* **97,44**
GRAY+ MINVAR+ COLORCUBE	96,91 *97,04* **96,92**	97,49 *97,23* **97,24**	97,03 *97,83* **97,73**	99,35 *99,14* **99,18**

In this study we have demonstrated that diverse descriptions of textural data can also be obtained through different colour representations. This fact can be exploited to improve the overall classification performance by combining multiple experts that result from different feature spaces and colour representations.

Acknowledgements

This work was supported by the *Xunta de Galicia* (Spain) under the grant n. PGIDIT04REM303003PR and by the *Fondazione CARIT* (Italy) under the project entitled “Sviluppo di metodologie di simulazione e sperimentali per la valutazione del comportamento di materiali impiegati nell’edilizia e nell’industria meccanica”.

References

1. Kuncheva, L.: Combining Patterns Classifiers. Methods and Algorithms. Wiley-Interscience (2004)
2. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(3) (1998) 226–239
3. Drimbarean, A., Whelan, P.: Experiments in colour texture analysis. Pattern Recognition Letters **22**(10) (2001) 1161–1167
4. Mäenpää, T., Pietikainen, M.: Classification with color and texture: jointly or separately? Pattern Recognition Letters **37**(8) (2004) 1629–1640
5. Palm, C.: Color texture classification by integrative co-occurrence matrices. Pattern Recognition **37**(5) (2004) 965–976
6. Song, K., Kittler, J., Petrou, M.: Defect detection in random colour textures. Image and Vision Computing **14** (1996) 667–683

7. Arvis, V., Debain, C., Berducat, M., Benassi, A.: Generalization of the cooccurrence matrix for colour images: application to colour texture classification. Image Analysis & Stereology **23** (2004) 63–72
8. van den Broek, E., van Rikxoort, E.: Evaluation of color representation for texture analysis. In: Proc. of the 16th Belgian-Dutch Conference on Artificial Intelligence, Groningen (Holland) (2004)
9. Wu, X.: Color quantization by dynamic programming and principal analysis. ACM Transactions on Graphics **11**(4) (1992) 348–372
10. Various Authors: Matlab release 14. Online documentation (2004)
11. Mäenpää, T., Pietikainen, M.: Texture analysis. In: Handbook of Pattern Recognition and Computer Vision. World Scientific Publishing (2005)
12. Sánchez-Yáñez, R., Kurmyshev, E., Cuevas, F.: A framework for texture classification using the coordinated clusters representation. Pattern Recognition Letters **24**(1-3) (2003) 21–31
13. Ribas, F., Fernández, A.: Generación de etiquetas cualitativas para el modelado de variables meteorológicas en el campo de la dinámica simbólica . In: Actas del Congreso Internacional Conjunto XVI ADM - XIX INGEGRAF, Perugia (Italy) (2007)
14. Bianconi, F., Fernández, A.: Granite texture classification with Gabor filters. In: Actas del XVIII Congreso Internacional de Ingeniería Gráfica, Sitges (Spain) (2006)
15. Bianconi, F., Fernández, A.: Evaluation of the effects of Gabor filter parameters on texture classification. Pattern Recognition. *In press. Available online at doi:10.1016/j.patcog.2007.04.023* (2007)
16. Jain, A., Duin, R., Mao, J.: Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(1) (2000) 4–37
17. Duda, R., Hart, P., Stork, D.: Pattern Classification. Second Edition. Wiley-Interscience (2000)
18. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Third Edition. Academic press (2006)
19. Chen, L., Tang, H.: Improved computation of beliefs based on confusion matrix for combining multiple classifiers. Electronics Letters **40**(4) (2004)
20. Xu, L., Krzyzak, A., Suen, C.: Methods of combining multiple classifiers and their applications to handwriting recognition. IEEE Transactions on Systems, Man and Cybernetics **22**(3) (1992) 418–435
21. Medasani, S., Kim, J., Krishnapuram, R.: An overview of membership function generation techniques for pattern recognition. International Journal of Approximate Reasoning **19**(3-4) (1998) 391–417
22. Duin, R., Tax, D.: Classifier conditional posterior probabilities. Lecture Notes in Computer Science **1451** (1998) 611–619
23. Steyerberg, E., Harrell, F., Borsboom, G., Eijkemans, M., Vergouwe, Y., Habbema, J.: Internal validation of predictive models: Efficiency of some procedures for logistic regression analysis. Journal of Clinical Epidemiology **54** (2001) 774–781