Noise modeling for smoothing the colour histogram

L.Shafarenko, M.Petrou and J.Kittler
Dept. of Electronic and Electrical Engineering,
University of Surrey,
Guildford GU2 5XH, United Kingdom.
e-mail:{l.shafarenko,m.petrou}@ee.surrey.ac.uk

ABSTRACT

In this paper we present a segmentation algorithm for colour images that uses the watershed algorithm to segment either the 2D or the 3D colour histogram of an image. For compliance with the way humans perceive colour, this segmentation has to take place in a perceptually uniform colour space like the Luv space. To avoid oversegmentation, the watershed algorithm has to be applied to a smoothed out histogram. The noise, however, is inhomogeneous in the Luv space and we present here the noise analysis for this space based on assumptions that are experimentally justified.

1 Introduction

In problems of surface industrial inspection and colour grading the spectral classes recognised on an image by a computer vision system have to correspond to chromatic classes perceived as distinct by the human vision system. For this purpose, the Luv colour space has to be used, in which the Euclidean distance between two points is approximately proportional to the perceptual difference between the two colours represented by these points. The colour histogram has to be smoothed out in such a way that the peaks that are closer to each other than this threshold merge. Below we shall refer to this type of smoothing of the colour histogram as perceptual coarsening.

There is another advantage of the Luv space over other colour spaces: the resolution gain. Indeed, both filtering and clustering applied to a 3D histogram are likely to be computationally expensive. While in the Luv space we can uniformly choose a resolution sufficient to adequately represent human perception of colour, the same task in the RGB requires a much higher resolution which makes the algorithm computationally infeasible.

Using the Luv colour space has, however, a drawback arising from the noise introduced by the image acquisition process. For a wide range of hardware the noise in RGB is homogeneous. However, the transformation from the RGB to the Luv space is highly nonlinear. Nonlinearity makes the noise inhomogeneous in the Luv space.

In this paper we present an algorithm for colour segmentation that imitates human perception. To this end, an adaptive filter is used that effectively removes noise from a 3D colour histogram in the Luv colour space, with subsequent perceptual coarsening. A colour clustering method based on the morphological watershed transform is then applied to the colour histogram. The segmentation results are presented for a variety of colour images.

2 Noise in a colour space

We assume that the noise is homogeneous in the RGB space and its distribution is Gaussian with the three colour coordinates being statistically independent. Both these assumptions have been experimentally verified.

The transformation from RGB to Luv space is given by [1]:

$$L = \begin{cases} 116(Y/Y_0)^{1/3} - 16, & \text{if } Y/Y_0 > 0.008856 \\ 903.3Y/Y_0, & \text{if } Y/Y_0 \leq 0.008856 \end{cases}$$
$$u = 13L(\frac{4X}{X+15Y+3Z} - u_0\prime) \qquad (1)$$
$$v = 13L(\frac{9Y}{X+15Y+3Z} - v_0\prime),$$

where

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}, \qquad (2)$$

$$u_0\prime = \frac{4.0x_0}{1.0X_0 + 15.0Y_0 + 3.0Z_0}$$
$$v_0\prime = \frac{9.0y}{1.0X_0 + 15.0Y_0 + 3.0Z_0},$$

and (X_0, Y_0, Z_0) is the reference white. Here L and Y components are linked to the luminosity, while u, v and X, Z are chromatic components.

Since the transformation from RGB to Luv is nonlinear, we can no longer assume the noise to be Gaussian and homogeneous. Nevertheless, it could be shown, that the probability density function of the noise may be assumed locally to be Gaussian with standard deviation that depends on the position itself, if the following condition holds:

$$\frac{L''_{RR}}{L'_R}\sigma_R \sim \frac{\sigma_R}{R} \ll 1,$$

where σ_R is the standard deviation of the distribution in R, and L'_R denotes $\frac{\partial L}{\partial R}$. Similar conditions should hold for G and B.Therefore a Gaussian noise distribution in RGB with a standard deviation σ implies a Gaussian noise distribution in Luv with the following covariance matrix:

$$\sigma^2\begin{pmatrix} (L'_R)^2+(L'_G)^2+(L'_B)^2 & L'_R u'_R+L'_G u'_G+L'_B u'_B & L'_R v'_R+L'_G v'_G+L'_B v'_B \\ L'_R u'_R+L'_G u'_G+L'_B u'_B & (u'_R)^2+(u'_G)^2+(u'_B)^2 & u'_R v'_R+u'_G u'_G+u'_B v'_B \\ L'_R v'_R+L'_G v'_G+L'_B v'_B & u'_R v'_R+u'_G u'_G+u'_B v'_B & (v'_R)^2+(v'_G)^2+(v'_B)^2 \end{pmatrix}$$

This covariance matrix is different for different parts of the colour space.

Once we have estimated the probability density function of the noise, we must use it to improve the colour histogram of the image. We can view each histogram value as a measurement which carries with it a certain degree of uncertainty. That is, each measurement we are having, could have arisen with varied degrees of probability from a whole range of possible true measurements. As we cannot possibly know from which of these measurements exactly it arose, all we can do is to replace this measurement (which can be thought of as a delta function in the absence of any noise) by the finite width Gaussian. The width of this Gaussian is measurement dependent. Our histogram then looks like a whole lot of shifted and overlapping Gaussians and to cope with the uncertainty they manifest, we must integrate them out. In other words, we must convolve the histogram with this variable width Gaussian.

In addition to this, another step is required before clustering could take place. Namely, the colour histogram needs to be "coarsened" to correspond to human perception, in other words we do not want to distinguish between clusters that the human eye is not able to recognise as different, even though they might be separated by the algorithm. One way to ensure this is to choose the resolution of the Luv colour space to be low enough. This effectively results in the colour histogram being smoothed by a rectangular window whose width corresponds to the threshold of the human perception.

3 Clustering

After the colour histogram has been filtered to remove the noise and coarsened to represent the human perception of colour, we are in a position to perform clustering. We are aiming at automatic clustering, where all information should be extracted from the image itself. For this purpose we have used a well-known morphological algorithm, watershed transform For details the reader is referred to extensive literature on the subject ([2], for example).

Having defined the clustering algorithm, we are now in a position to concentrate on the segmentation we are trying to achieve. In this paper we are addressing two different segmentation procedures, the first being chromatically-based, and the second taking into consideration both chromaticity and intensity of the image. In both of them we use the watershed algorithm to segment the *colour histogram* of the image. We shall discuss now both approaches in more detail.

3.1 Chromaticity-based segmentation

The need for this type of segmentation arises when we are presented with a problem of segmenting the image according to colour information alone, ignoring the intensity. A good example of such a task would be segmentation of the image, where the creases of the fabric and the shadows due to illumination changes across the scene should not prevent us from segmenting out the region in question as having uniform colour. Having adopted this approach we should not expect to be able to distinguish between points in colour space that differ only in intensity.

It might appear that for chromaticity-based segmentation we should consider only a two-dimensional colour histogram, summing up votes for all intensities occurring at each point of the chromaticity plane (which is the uv plane in the case of Luv colour space). This is not the case, however, due to the fact that noise "mixes up" colour coordinates, that is each point in colour space contributes into several points on the chromaticity plane, rather than solely into its projection onto the chromaticity plane. The noise filtering we propose is essentially three-dimensional, and summation over intensity should be done only after such filtering has been performed. The situation is different for the operation of perceptual coarsening which is commutative with the operation of summation over intensity and could therefore be performed in chromaticity plane rather than in the three-dimensional colour space.

Thus the algorithm for chromaticity-based segmentation is as follows:

- Calculate the colour histogram of the image.
- Filter it for noise reduction.
- Project it onto the chromaticity plane by summing over the intensity coordinate.
- Perform perceptual coarsening.
- Perform clustering using the watershed algorithm in 2D.

Segmentation results using this algorithm are shown in Fig.2.

3.2 Segmentation based on both intensity and chromaticity

A very different type of segmentation occurs when we are facing the problem of segmenting the image "as seen", that is when it is necessary to recognise as different those colours that differ in their luminance value. In this case we have the ability to distinguish between black and white, but we lose the tolerance to shades and creases. In this case we have to do clustering on the three-dimensional colour histogram. The algorithm is similar to the previous one, with the exception that we do not perform the projection onto the chromaticity plane. Segmentation results using this algorithm with several different values of parameters are shown in Fig.1. To give the reader an idea of the algorithm performance, colour values for each pixel were replaced by those of the cluster the pixel belongs to. The colour values of the cluster were calculated as its mean L, u, and v . Note that this was done to visualise the result rather than for any further use of the image. If such use is intended (for compression purpose, for example) the ways to represent the cluster should be researched, which is beyond the scope of this paper.

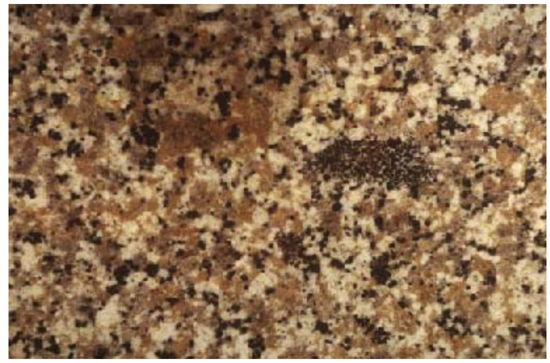
4 Conclusions

A new algorithm is proposed for segmentation of colour images, which takes into account the noise that is inevitably present during the image acquisition. It is shown that such noise affects human perception of the image due to the nonlinear nature of the human perception. This leads to a situation when even a low absolute value of the noise is noticeable to the human eye in certain areas of the colour space. The *Luv* colour space was used for perceptual coarsening of the colour histogram, as well as for the resolution gain it can offer compared to the *RGB* space. The clustering method was based on the morphological watershed transform performed on the 3D colour histogram. The resulting algorithm is highly suitable for automatic colour segmentation. Indeed, there are only two parameters involved: the width of the noise distribution and the size of the window for perceptual coarsening. The former describes the hardware setup, while the latter reflects the desired degree of coarseness in the segmentation. When a combination of these two parameters is found that results in a good segmentation, the algorithm performs well for a wide range of images acquired using the same hardware.

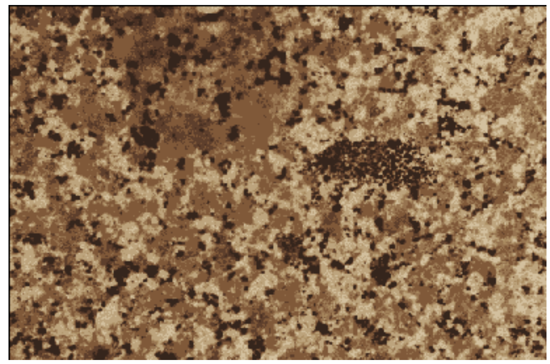
References

[1] G.Wyszecki, W.S.Stiles, *Color Concepts and Methods, Quantitative data and Formulae*, John Wiley & Sons, 1982.

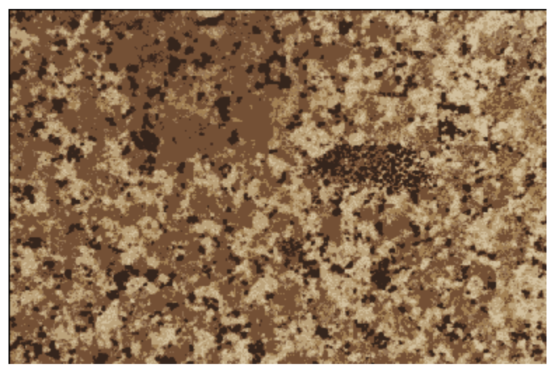
[2] S.Beucher and F.Meyer, The Morphological Approach to Segmentation: The Watershed Transformation, *Mathematical Morphology in Image Processing*, New York, Marcel Dekker, 1993, pp.443-481.



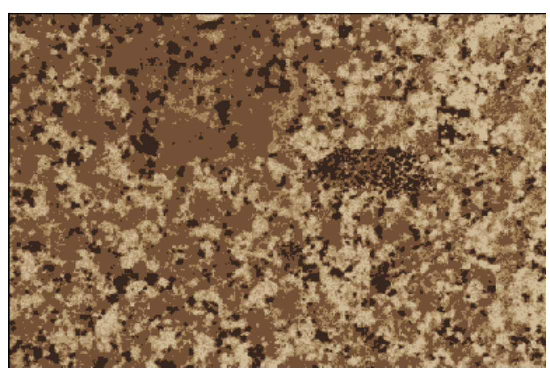
(a)



(b)



(c)



(d)

Figure 1: 3D segmentation of a granite image. (a)original image; (b)segmentation with ρ=1.5 (9 clusters); (c)segmentation with ρ=2. (7 clusters) (d)segmentation with ρ=4. (6 clusters). σ=3.6 for all cases.

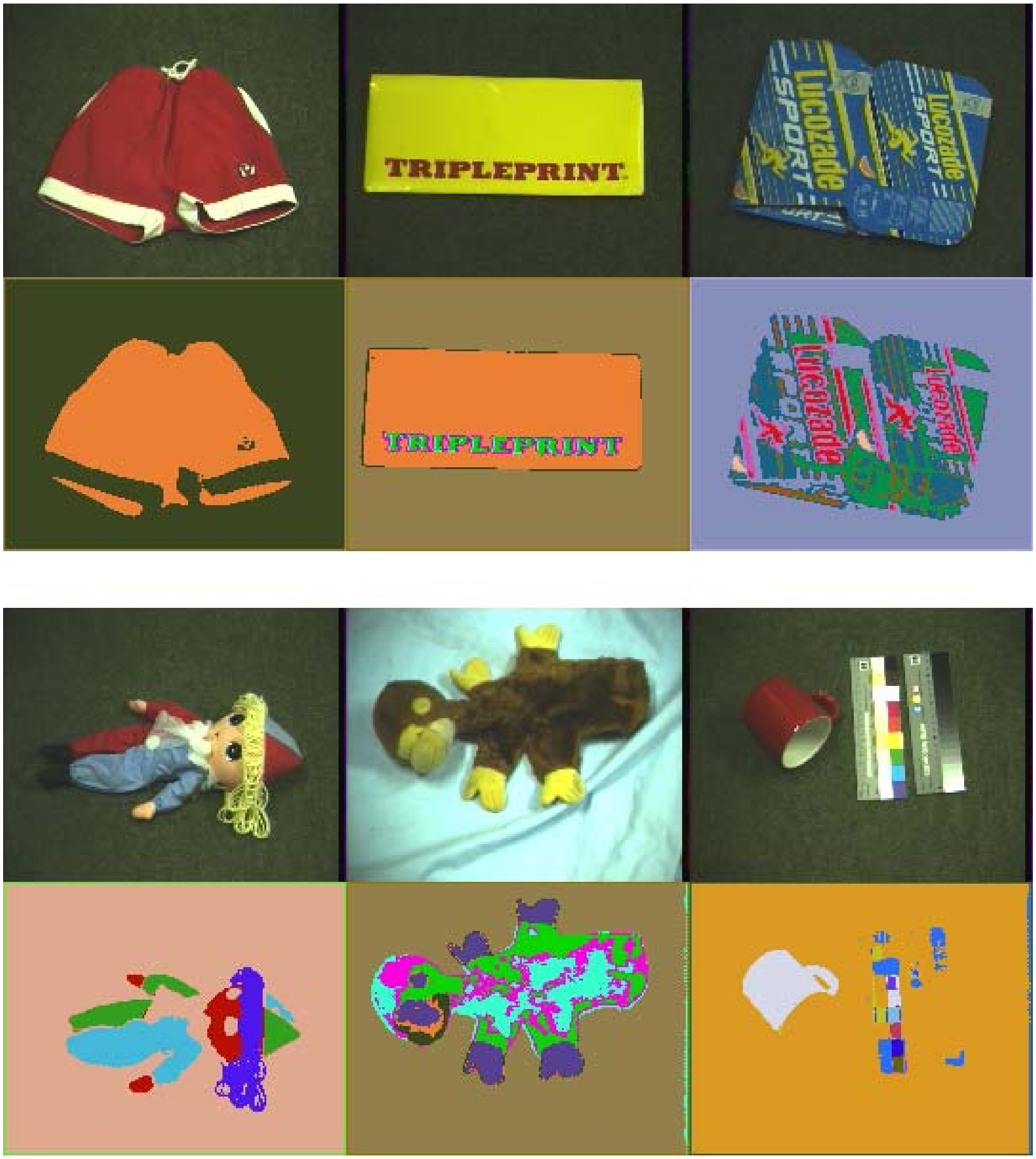


Figure 2: Segmentation on a variety of images using a shromaticity-based approach