14

Study of the effect of lighting technology in texture classification systems

Rubén Muñiz, José Antonio Corrales, Manuel Rico-Secades
and Juan Ángel Martínez-Esteban
University of Oviedo
Spain

1. Introduction

Traditionally, the most commonly used lighting technology in computer vision applications has been the halogen one. In some cases, fluorescent lamps are used due to intrinsic restrictions of the site where the application is running (e.g.: car parks).
A few years ago, new improvements in LED technology started to appear, giving birth to a totally new range of products for domestic and industrial use. These new high efficiency diodes offer many advantages over older technologies, such as longer life span, less heat radiated and less power consumption.
On the other hand, with the introduction of low cost colour cameras for computer vision applications, the effect of the technology of the lamps used is even more noticeable. In previous papers we have studied how variations in the intensity of the illumination affect the quality of the descriptors measured from texture samples. For that work we used a halogen projector and a mainstream digital video camera.
This time we are going to use an industrial IEEE colour camera from Pixelink and a custom-built RGB LED projector. We seek to validate the results we have from former studies on a new image set using the LED lamp and a halogen one.

2. Introduction to texture classification

Texture classification is a key field in many computer vision applications, ranging from quality control to remote sensing. Briefly described, there are a finite number of texture classes we have to learn to recognize. In the first stage of the development of such kind of systems, we extract useful information (features) from a set of digital images, known as the training set, containing the textures we are studying. Once this task has been done, we proceed to classify any unknown texture into one of the known classes. This process can be summarised in the following steps:

1. Image (texture) acquisition and preprocessing
2. Feature extraction
3. Feature selection
4. Classification

Since earlier approaches to the problem greyscale images have been widely used, primarily due to acquisition hardware limitations and/or limited processing power. In the recent past, much effort has been made to develop new feature extraction algorithms (also known as texture analysis algorithms) to take advantage of the extra information contained in colour images. On the other hand, many classical greyscale algorithms have been extended to process colour textures (de Wouver, 1998; Jain & Healy, 1998; Palm, Keysers, Lehmann & Spitzer, 2000).
Texture analysis algorithms can be divided into statistical and spectral ones. The former methods extract a set of statistical properties from the spatial distribution of intensities of a texture. Common examples of this approach are the histogram method and the family of algorithms based on cooccurrence matrices (Conners & McMillin, 1983; Haralick et al., 1973). The latter techniques, on the other hand, compute a number of features obtained from the analysis of the local spectrum of the texture. In the following sections we will give an overview of two spectral methods (Gabor filters and Wavelets) and a statistical one (Cooccurrence matrices).

3. Gabor filters

Gabor filters have been extensively used for texture classification and segmentation of greyscale and colour textures (Bovik & Clark, 1990; Dunn et al., 1994; Jain & Healy, 1998; Kruizinga et al., 1999; Palm, Keysers, Lehmann & Spitzer, 2000). These filters are optimally localized in both space and spatial frequency and allow us to get a set of filtered images which correspond to a specific scale and orientation component of the original texture. There are two major approaches to texture analysis using Gabor filters. First, one can look for specific narrowband filters to describe a given texture class, while the other option is to apply a bank of Gabor filters over the image and process its outputs to obtain the features that describe the texture class.

3.1 2D Gabor filterbank

The Gabor filter bank used in this work is defined in the spatial domain as follows:

$$\begin{aligned} f_{mn}(x,y) &= \frac{1}{2\pi\sigma_m^2} \exp^{-\frac{x^2+y^2}{2\sigma_m^2}} \times \\ &\times \cos 2\pi(u_m x \cos\theta + u_m y \sin\theta) \end{aligned} \tag{1}$$

where m and n are the indexes for the scale and the orientation, respectively, for a given Gabor filter. Depending on these parameters, the texture will be analyzed (filtered) at a specific detail level and direction. The half peak radial and orientation bandwidths (Bovik & Clark, 1990) are defined as follows:

$$B_r = \log_2 \left(\frac{2\pi\sigma_m u_m + \sqrt{2\ln 2}}{2\pi\sigma_m u_m - 2\sqrt{2\ln 2}} \right) \tag{2}$$

$$B_\theta = 2\tan^{-1} \left(\frac{\sqrt{2\ln 2}}{2\pi\sigma_m u_m} \right) \tag{3}$$

As in (Jain & Healy, 1998), we define a filterbank with three scales and four orientations. The bandwidth B_θ is taken to be 40° in order to maximize the coverage of the frequency domain and minimize the overlap between the filters.

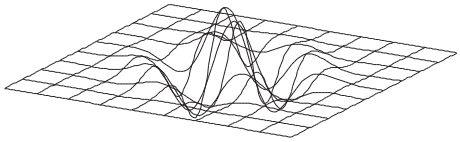


Fig. 1. Gabor filter computed by $f_{1,45^\circ}$

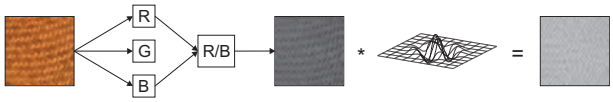


Fig. 2. Gabor filtering of rationed images

3.2 Gabor features

To obtain texture features we must filter the texture images using the generated filters. This is achieved by convolution on the frequency domain (4), due to the size of the filters used. For each filtered image, we extract a single feature μ_{mn} which represents its energy, as shown below.

$$G_{mn}(x,y) = I(x,y) * f_{mn} \tag{4}$$

$$\mu_{mn}(x,y) = \sqrt{\sum_{x,y} (G_{mn}(x,y))^2} \tag{5}$$

This approach is only valid when greyscale images are used. If we want to filter a colour image, we have to preprocess it before this method can be applied. The more obvious solution to this problem is to transform the image by a weighted average of the three colour bands.

$$I(x,y) = aR(x,y) + bG(x,y) + cB(x,y) \tag{6}$$

The coefficients a, b, c from equation (6), can be selected to properly model the human eye's perception model of colour, since the number of cones sensitive to red, green and blue light is not equal (Hecht, 1987). For this reason, an adequate choice of these weights can be $a = 0.65, b = 0.33, c = 0.02$, but this is only needed for human visualization and does not necessary provide a good representation for texture feature extraction. In this paper, we will use the simplest choice of coefficients given by $a = b = c = \frac{1}{3}$.

Using this transformation, different colours can give the same greyscale intensity, so colour information is lost. To overcome this obstacle, (4) can be applied on each of the RGB colour bands of the image to obtain unichrome features (Jain & Healy, 1998). With this approach, we obtain a set of energies from each spectral band, so the information extracted from textures, grows by a factor of three. Another disadvantage of this technique is that colour information is not correlated because it is simply concatenated. A good idea to solve this independency was proposed by Palm et al. in (Palm, Keysers, Lehmann & Spitzer, 2000). They converted the RGB image to HSV, discarding the intensity value, and taking the Hue and Saturation to form a complex number, which can be used to compute the convolution between the image and the gabor filter by means of a complex FFT.

4. Wavelets

4.1 Introduction

The name "Wavelets" was first introduced by Morlet, a French geophysicists, in the early 80's. The kind of data he was studying could not be properly analysed by Fourier analysis, due to the fast change of their frequency contents. For this reason, he looked for a family of functions suitable for the analysis of that kind of signals and he found the wavelets.
A wavelet family is a set of functions derived from a single function with special features, named the *mother* wavelet, by means of two parameters a and b:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \tag{7}$$

The parameter a represents the dilation (which is inversely proportional to frequency) and b the displacement (time localization).
Wavelets are rather complex and we would require a complete book (Mallat, 1999) to deal with them. In the following lines we will show only the basics of this kind of analysis, focused on texture feature extraction.

4.2 2D Discrete Wavelet Transform and multiscale analysis

Wavelets allow us to study a signal at different levels of detail. In the case of 2D signals, this can be interpreted as analysing the images at different resolutions or scales. For instance, if we take a image of 512×512 pixels, we can study its frequency content at $256 \times 256, 128 \times 128, 64 \times 64$ and so on. In the special case of images containing textures, this is of vital importance since a texture varies significantly depending on the distance from which we are looking at it.
If we take a such as $a = 2^m$, the transform is known as dyadic DWT and relies on a specific form of Wavelet derived from a smoothing or scaling function represented by $\theta(t)$. From both the scaling function and the Wavelet, we can derive two FIR filters that can be used to compute the DWT. These filters are commonly named h and g and are, respectively, a low-pass and a high-pass filter. The 2D transform can be represented as shown in Fig. 3, where $*$ denotes the convolution operator and the subscript for each filter represents if it is applied over the rows or the columns of the image. Finally $\downarrow 2$ denotes downsampling of the rows or columns by a factor of two.
Looking at Fig. 3, we can see that the DWT in 2D produces four images as output. In each filtering step, a *low resolution* image L_{j+1} and three *detail images* are produced. The detail images $D_j^{1..3}$ contain the details (high frequency components) extracted from D_j that are not present in $D_j + 1$. This scheme can be applied recursively until a given depth is reached.

4.3 Wavelet features

Detail images obtained by applying 2D DWT can be used as a source for extracting texture features. Since those images contain essentially edge information at a specific direction (horizontal, vertical and diagonal), their energy is a very good texture feature. It is defined as follows:

$$E_{ij}(x,y) = \sqrt{\sum_{x,y}\Big(D_{ij}(x,y)\Big)^2} \tag{8}$$

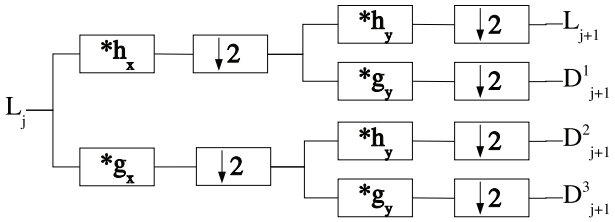


Fig. 3. 2D DWT using FIR filters

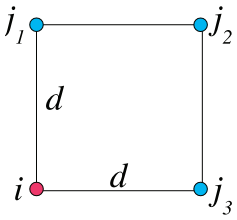


Fig. 4. Neighbourhood of a given pixel.

where $i = 1 \ldots 3$ and $j = 0 \ldots depth - 1$.

As in the case of Gabor filters, we need some mechanism to be able to process colour images, such as greyscale conversion (6) and independent colour band feature extraction. In this particular case we can use a correlation measure (de Wouver, 1998). This feature is named the *wavelet covariance signature* and is defined as follows:

$$C_{ij}^{B_k B_l}(x,y) = \sum_{x,y} D_{ij}^{B_l} D_{ij}^{B_k} \tag{9}$$

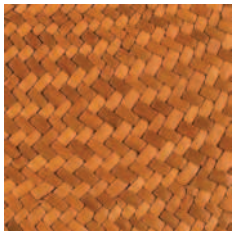
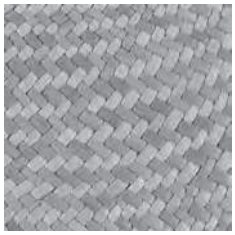
where B_k and B_l represent a colour band, and $k, l = 1, 2, 3, k \leq l$.

5. Grey level cooccurrence matrices

5.1 Introduction

Grey level cooccurrence matrices (GLCM) were introduced for the first time by Haralick (Haralick et al., 1973) in the early 70's. A GLCM is a matrix where each cell i, j contains the number of times a point having intensity i occurs in a position j located at an angle θ and a distance d. If we want to make this approach non sensible to orientation variations, we can use the neighbourhood located at a distance d from the pixel with intensity i. Only the first quadrant of that neighbourhood need to be explored, which is equivalent to taking $\theta = 0, 45, 90$ simultaneously for the same matrix (Fig. 4).

A final topic concerns the size of the GLCM. If we directly use the 256 grey levels available in a image, the resulting matrix will be huge, so a mechanism to reduce its dimensions is needed. There are some options to do this. In this paper, we have used two different preprocessing tasks:

a. Original colour image b. Hotelling transform

Fig. 5. Fabric 0 and Hotelling transform of Fabric0

1. Use of Sobel filters to detect the edges of the image. The resulting image will have five different values: horizontal edge, vertical edge, diagonal edge, secondary diagonal edge, and no edge. The resulting matrix belongs to a class of matrices known as Cooccurrence Generalized Matrices (CGM).
2. Reduce the grey levels to 16 using some quantization algorithm such as IGS. In this case, the resulting matrices are named Spatial Grey Level-Dependent Matrices (SGLDM).

5.2 Feature extraction

From a cooccurrence matrix, a number of second order statistics can be computed. The most popular ones are those known as *Haralick features* (Haralick et al., 1973), followed by the set of measures introduced by Conners et al. (Conners & McMillin, 1983).
Obviously, a GLCM cannot be computed directly from a colour image, but from a greyscale one, so a modification is needed to be able to process that kind of images. There are three ways of doing this:

1. Convert the colour image to monochrome (6). This is straightforward, but it discards the chromatic information from the images.
2. Process each RGB band separately. With this approach, a GLCM is computed from each colour band and the resulting feature vector f_v is the concatenation of three feature vectors f_R, f_G, f_B, obtained from each matrix separately. The main disadvantage is that the computational cost increases considerably and the obtanied information is not correlated.
3. Use cross-cooccurrence matrices (Palm, Metzler, Lehmann & Spitzer, 2000). This technique follows to extend the cooccurrence idea explained before to colour images. The process consists of using a given a pixel from a colour band B_1 with intensity i we will look for the intensity of a pixel located in another colour band B_2 at a distance d and orientation θ. As before, we can use three angles at the same time to compute a single matrix. Obviously, if $B_1 = B_2$ then we are computing a conventional GCLM. The main advantage of this class of matrices is that features extracted from them contain colour information, because colour planes are processed in pairs, but the feature vector length is increased by a factor of six.

6. Principal Components Analysis

6.1 Introduction

Principal components analysis (PCA) is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. The main advantage of PCA when applied to colour images is that it can be used to compress them, i.e, find an optimal *ad hoc* greyscale transform for each image. In this case, PCA is also known as the Hotelling or Karhunen and Loève transform. A good tutorial for this technique can be found in (Smith, 2002).

6.2 colour image compression and feature extraction

When PCA is applied to colour image compression, we can start with a RGB representation of the image. The covariance matriz size will be 3×3 and consequently, the number of associated eigenvectors will be 3. If we take the eigenvector with the highest eigenvalue, we will have found the *principal component* of the distribution. If we represent this eigenvector as $\overrightarrow{pc} = (a, b, c)$, we can use its components as the coefficients for equation (6), so an optimal greyscale transform is applied. In Fig. 5 an example of this approach is given. It does not need to be mentioned that since we have compressed colour information into a single monochromatic image, all the feature extraction algorithms showed before (and others) can be directly used to measure texture features.

7. Band ratioing

In the previous sections, we have given a brief introduction to three of the most commonly used feature extraction algorithms, and we have seen the way many authors are extending them to process colour textures. In this section, we will show a novel approach to doing this, obtaining in many cases the highest classifier performance, while keeping a low number of features.

7.1 Introduction

Band ratioing is a enhancement technique mainly used in the field of remote sensing. It is usually applied to processing LANDSAT TM images[1] to enhance details such as vegetation, grass, soil, etc. It is defined as follows:

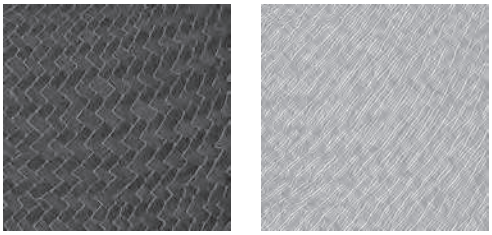
$$I(x,y) = \frac{B_1(x,y)}{B_2(x,y)} \tag{10}$$

where $B_1(x,y)$ and $B_2(x,y)$ are two different spectral bands of the colour image. Its computation is extremely easy, but the bands involved must be processed to avoid the case when $B_2(x,y) = 0$. To accomplish this, we only have to increase every pixel from both bands by 1. Theoretically, ratios will be in the interval (0,256], but in practice most values will be rather small. For this reason, it is advisable to use logarithm compression to enhance small ratios over larger ones, so (10) can be rewritten as follows.

$$I'(x,y) = \log\left(\frac{B_1(x,y)}{B_2(x,y)}\right) \tag{11}$$

It can be easily seen that this technique tends to enhance what is different in two spectral bands, and as will be seen in the following section, its output is suitable for feature extraction.

[1] http://rst.gsfc.nasa.gov/Sect1/Sect1_15.html



a. Ratio R/B b. Filtered ratio R/B

Fig. 6. Ratio R/B and filtered ratio of Fabric0

7.2 Feature extraction from rationed colour textures

In the previous section, we saw that Band Ratioing enhances what is different in two colour bands. If a pixel contains a greyscale value ($R = G = B$), its ratio will be 1, but if at least two colour components are not equal, the band ratio will encode the colour information in a single value. This is very interesting for feature extraction from colour textures, since we can directly use any greyscale feature extraction method available. In the following lines, we will show the way to enhance the Gabor filtering method using band ratioing.
To apply Gabor filtering on a rationed image, we can combine (11) and (4) to get the following expression:

$$G'_{mn}(x,y) = \log\left(\frac{B_1(x,y)}{B_2(x,y)}\right) * f_{mn} \tag{12}$$

Note that (12) directly convolves the band ratios with the Gabor filter, so it is not necessary to scale the ratios to fit in a byte value.
In Fig. 6 we can see a band ratio of a colour texture taken from the VisTex database[2] and the result of filtering the ratio using a Gabor filter with parameters $m = 1, n = 45°$.
A very interesting topic is the implementation of the previous scheme. When we compute a band ratio, the operands are both eight-bit numbers, i.e, ranging from $0 \ldots 255$, but the result will be a real number. There are two different ways to deal with this value:

1. Adjust the result to fit in a byte value (some information is lost).
2. Use the real ratio directly as input to the FFT.

It is easy to see that the second approach is much better since no information is lost. In our experiments, we have observed a 10% of performance increase compared to the case when byte values are used. Nevertheless, this option is only applicable to feature extraction algorithms that make use of the FFT, since we can not use real numbers to compute cooccurrence matrices, for example.

8. The Vistex problem

For testing the performance of the band ratioing technique combined with the three feature extraction algorithms presented before, we have used the texture set defined in (de Wouver, 1998) which is composed of 30 colour textures taken from the Vistex database (Fig. 7). As

[2] http://www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html

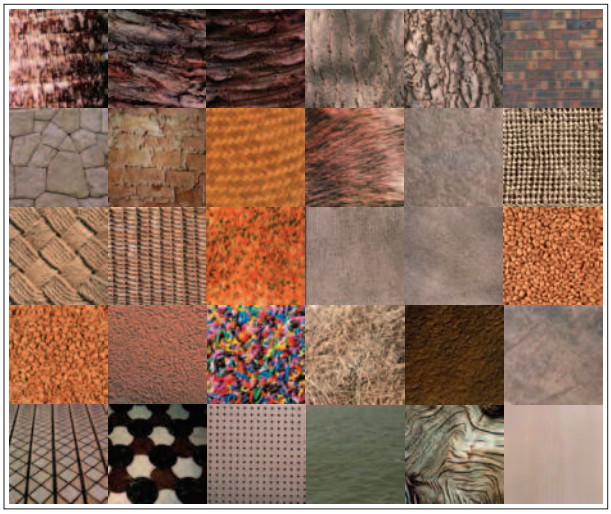


Fig. 7. 30 colour textures: Bark0, Bark4, Bark6, Bark8, Bark9, Brick1, Brick4, Brick5, Fabric0, Fabric4, Fabric7, Fabric9, Fabric11, Fabric13, Fabric16,Fabric17, Fabric18, Food0, Food2, Food5, Food8, Grass1, Sand0, Stone4, Tile1, Tile3, Tile7, Water6, Wood1, Wood2 (left/right,top/bottom)

in (Palm, Keysers, Lehmann & Spitzer, 2000) we divide each 512x512 image into 64 disjunct images of 64x64 pixels each, which give us a total of 1920 texture samples. For each texture class, we randomly select 80% of the samples for training, and the rest for testing purposes (hold-out method).
We have performed a number of experiments, applying different preprocessing techniques to be able to process colour images. The tables showed in the next section gather the classifier performance in all cases.

8.1 Results

For evaluating the performance of the feature sets obtained in each case, we have used a Knn classifier, taking $K = 5$. To measure the distance between two feature vectors in R^n, the 1-norm is used.

8.1.1 Gabor results

Gabor features were extracted by filtering the textures at 3 scales and 4 different orientations. This give us a total of 12 gabor energies for each texture sample. As can be seen in Table 1, the best performance is achieved by the concatenation of two band ratios. These results are not only interesing for this topic, but for the reduced number of features used. If we look at the RGB row, we see that by using raw colour information, 36 features are extracted and the performance is worse. It is also remarkable the fact that using band ratios we can even improve the performance of the *Complex Gabor Features* introduced in (Palm, Keysers, Lehmann & Spitzer, 2000).

Preprocessing algorithm	Feature #	Hits
greyscale	12	87.86%
Red band	12	89.52%
Green band	12	90.24%
Blue band	12	87.38%
R+G+B	12×3	93.57%
Ratio R/G	12	86.67%
Ratio R/B	12	90.24%
Ratio G/B	12	85.48%
Ratios R/G & R/B	12×2	95.24%
Ratios R/G, R/B & G/B	12×3	95.24%
Complex colour features	31	92%
PCA	12	90.71%

Table 1. Classification rates for Gabor data.

Preprocessing algorithm	Feature #	Hits
greyscale	12	86.19%
Red band	12	90.48%
Green band	12	85.24%
Blue band	12	86.19%
R+G+B	12×3	91.19%
Ratio R/G	12	83.33%
Ratio R/B	12	87.62%
Ratio G/B	12	80.95%
Ratios R/G & R/B	12×2	93.57%
Ratios R/G, R/B & G/B	12×3	93.81%
Covariance signatures	72	95.24%
PCA	12	87.14%

Table 2. Classification rates for Wavelet data.

8.1.2 Wavelet results

For Wavelet features we have set the analysis depth at 4, which produces a total number of 12 features for each texture sample. The wavelet functions used for this analysis were the biorthogonal Wavelets *Bior6.8* available in MatLAB. In this case the performance of the concatenation of two band ratios is not the best at all, but the classification success is only 1.67% less than the case when covariance signatures are used, computing three times less features.

8.1.3 Cooccurrence matrices results

GLCM features were obtained by computing two coocurrence matrices at distances 2 and 4. For each matrix, we have calculated Haralick features, $f_1 \dots f_{12}$, that give us a total number of 24 texture measures for each sample. The overall performance in this case is worse than for the case where spectral methods are used, but we still see a performance increase when band

Preprocessing algorithm	Feature #	Hits CGM	Hits SGLDM
greyscale	24	78.81%	82.14%
Red band	24	78.57%	83.10%
Green band	24	79.29%	84.05%
Blue band	24	79.05%	85.24%
R+G+B	24×3	84.76%	93.1%
Ratio R/G	24	72.38%	74.76%
Ratio R/B	24	70.24%	76.43%
Ratio G/B	24	64.29%	70%
Ratios R/G & R/B	24×2	85.24%	91.67%
Ratios R/G, R/B & G/B	24×3	85.71%	91.67%
Cross-coocurrence	24×3	81.19%	90.24%
PCA	24	74.29%	77.14%

Table 3. Classification rates for CGM and SGLDM data.

ratios are involved. It is remarkable to notice how the concatenation of two band ratios still provides the best results while keeping the number of used features low.

8.2 Conclusions of the Vistex problem

In the previous sections, by means of the band ratioing technique we have combined RGB colour bands to produce monochromatic images suitable to use as input to any feature extraction algorithm currently available. From an implementation point of view, feature extraction algorithms that can use real images are preferred since no information is lost from band ratios. The most important conclusion is the fact that this technique allows us to compress textural colour information which leads to higher classification performance while keeping the number of features low. It is interesting to see how the use of three band ratios does not lead to better performance (or very little) than the case where only two ratios are involved. This is obvious since the third one is a linear combination of the two other, so the extra features do not contain additional information.

A final consideration concerns the use of the Hotelling transform. This process improves the performance of the simpler greyscale transform in the case where spectral methods (Gabor and Wavelets) are used, which leads us to think that the colour compression performed is better, as expected. For coocurrence matrices this is not true, since the grey levels reduction performed for the computation of those matrices seems to be affected by the use of this transform.

9. LED lighting

LED lighting is an emerging technology these days. It has evolved from the typical signaling diodes present on many different electronic devices to the more sophisticated lamps which are aimed at replacing conventional ones. It has applications for many different aspects of our lives, such as cars, traffic lights, show lighting and lamps at home.

Computer vision is not an exception here, the application of high power LEDs to almost any application is straightforward as it only requires replacing an "old" lamp with another one with a sufficient number of high power LEDs.

Fig. 8. LED module

During recent years we have been working with colour textures and we have developed a technique for compensating the performance loss when the illumination on the scene varies significantly. In this paper we want to reproduce those experiments with our custom-built LED projector and compare the results with the ones obtained when a conventional lamp is used.

9.1 Modern LED technology

As we have already mentioned, LED technology has been vastly improved during this decade. Nowadays it is very easy to find a product that fits our needs from many of the different manufacturers that have LEDs in their catalogues. However, in order to build our custom projector we have to perform minor research to find out which is the best one for our purposes. The first question to answer is if we need a colour or a monochrome LED. If we just need to replace a conventional lamp with a LED-based one, probably the monochrome choice is the best bet as we will have less issues to solve (driver choice, power supply, etc.). On the other hand, if we are looking for a more refined product and we want to try different spectral choices for the light source, we might be interested in getting RGB LEDs instead.
For our work we have decided to go for a power LED from OSRAM (LRTB-G6T6), shown in Fig. 8. This product includes the 3 RGB LEDs required in a single package and it has been optimised for additive mixture of colour stimuli by independent driving of the individual LEDs.
In order to control both the LED-based and the halogen light source, we are going to need a dimming technology. We have to keep in mind that we have a twin goal: on the one hand we want to compare the two technologies from a performance point of view but, on the other hand, we are looking forward to evaluating our formerly introduced algorithms [1][2][3] with the new LED projector. In the following chapter we will give a brief description of the most popular control technologies we have available in the market.

9.2 Dimming technologies

If we intend to simulate different lighting scenarios, one of the best approaches is choosing a light control system. There are a number of different choices on the market. Some of them are closed developments which means they are made ad hoc by a manufacturer and they are usually operated by an infrared remote or some kind of potentiometer / push button interface. On the other hand, we have a number of standard technologies which might serve to our purposes while allowing us the access to a full range of devices from different manufacturers.

9.2.1 1-10 V Interface

The 1-10 V interface is the oldest of the three approaches shown here. It is a fully analogue system and it allows us to control the attached ballast or transformer by means of two wires.

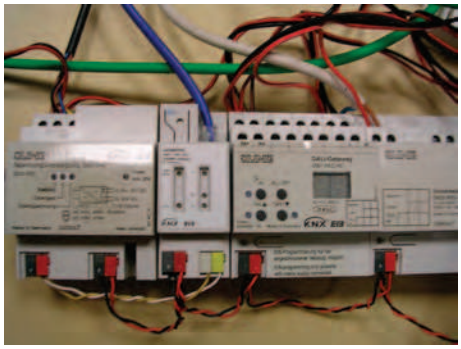


Fig. 9. KNX devices

On the other side of the connection we usually find a potentiometer as the only device required to operate the dimmed light. Its main advantage is that it is very cheap as opposed to other technologies but it is not very accurate and it is very slow in operation.

9.2.2 DALI

DALI stands for Digital Addressable Lighting Interface. As its name indicates, it is digital technology as opposed to the 1-10 V interface. As in the previous case, only two wires are required but the main difference is that up to 64 ballasts can be attached to a single controller. It is very common to find this technology in offices and buildings as it provides a high degree of control while keeping the cost reasonably low.

9.2.3 DMX512

DMX (Digital MultipleX) is also a digital protocol like DALI, but it focuses on the speed of the dimming which allows us to create special effects and smooth colour changes. The cost of the lamps and the required controller is usually higher than its DALI counterpart.

9.2.4 KNX

KNX or Konnex is not dimming technology. Instead, we are talking about a standard bus commonly used in Europe for connecting different systems in a house (heating, blinds, lights, etc.). Why we have decided to mention it here is because many different KNX (formerly known as EIB) manufacturers offer interfaces and controllers of the three systems described before in their catalogues. In addition, it is very simple to control a KNX-based installation from a PC, using one of the different interfaces available (RS-232, USB and Ethernet).
As we will describe later, we have decided to use this technology in combination with a DALI controller to perform all the tests required to compute the comparison charts.

10. Illumination-independent texture analysis

If we suppose that a spectral band b_1 from an image $f_{b1}(x, y)$ has been formed according to Lambert's reflectivity model (Lambert., 1760) then we can say that:

$$f_{b_1}(x, y) = I_{b_1} * R_{b_1} * \cos\Theta \ . \tag{13}$$

a. Bright texture b. Dark texture

Fig. 10. Bright and dark textures

where I_{b1} represents the intensity of the light source on the surface of the texture for the spectral band b_1 and R_{b1} the reflectivity coefficient of that surface.
For another band b_2, the previous equation would be rewritten as:

$$f_{b_2}(x,y) = I_{b_2} * R_{b_2} * \cos\Theta \ . \tag{14}$$

The band ratio of bands b_1 and b_2 is defined as:

$$BR_1 = \frac{f_{b_1}}{f_{b_2}} = \frac{I_{b_1}}{I_{b_2}} \frac{R_{b_1}}{R_{b_2}} \frac{\cos\Theta}{\cos\Theta} = k \frac{R_{b_1}}{R_{b_2}} \ . \tag{15}$$

If the lighting components I_{b1} and I_{b2} are modified we will get new ones, I'_{b1} and I'_{b2}. Then, the equation (15) would be rewritten as follows:

$$BR_2 = \frac{f'_{b_1}}{f'_{b_2}} = \frac{I'_{b_1}}{I'_{b_2}} \frac{R_{b_1}}{R_{b_2}} \frac{\cos\Theta}{\cos\Theta} = k' \frac{R_{b_1}}{R_{b_2}} \ . \tag{16}$$

If we want to calculate the difference between the rationed images BR_1 and BR_2, we only have to subtract equation (15) from equation (16):

$$BR_1 - BR_2 = k \frac{R_{b_1}}{R_{b_2}} - k' \frac{R_{b_1}}{R_{b_2}} = (k - k') \frac{R_{b_1}}{R_{b_2}} \ . \tag{17}$$

If the variations from the light source are proportional to both bands we can state that $k = k'$, then equation (17) could be rewritten as:

$$BR_1 - BR_2 = k \frac{R_{b_1}}{R_{b_2}} - k \frac{R_{b_1}}{R_{b_2}} = (k - k) \frac{R_{b_1}}{R_{b_2}} = 0 \ . \tag{18}$$

From the previous result we can state that if a surface can be expressed or modelled according to Lambert's reflectivity model and if the lighting variations are proportional between bands, then the rationed images do not change.
Obviously, the assumptions made before are not always true in real-world applications, but our experiments have shown us that this approach performs very well if we compare it with regular texture analysis.

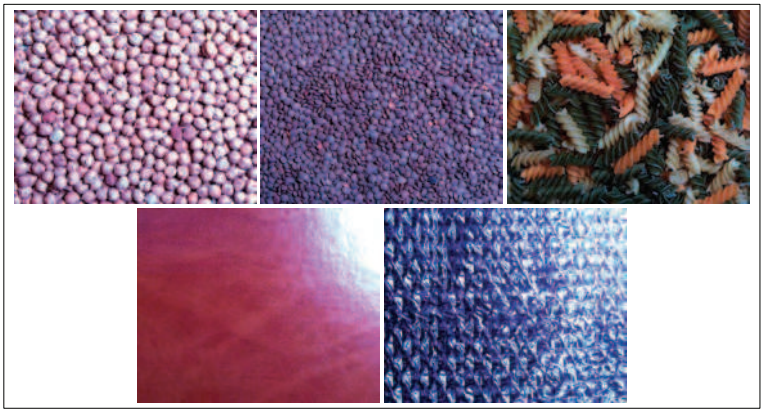


Fig. 11. Real-world textures (darkest ones) taken under different lighting conditions with a halogen projector

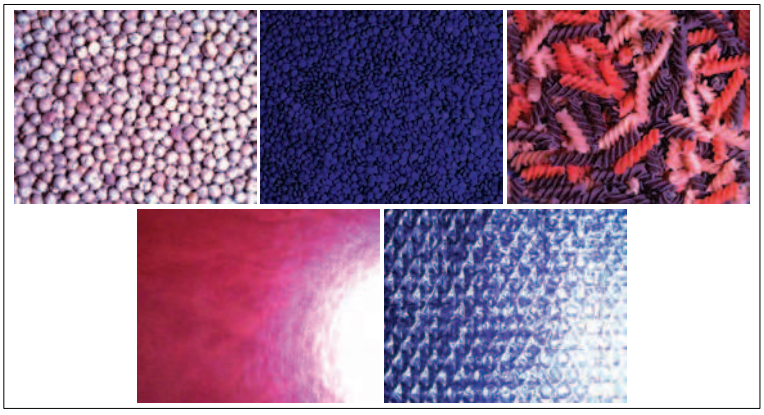


Fig. 12. Real-world textures (darkest ones) taken under different lighting conditions with a LED projector

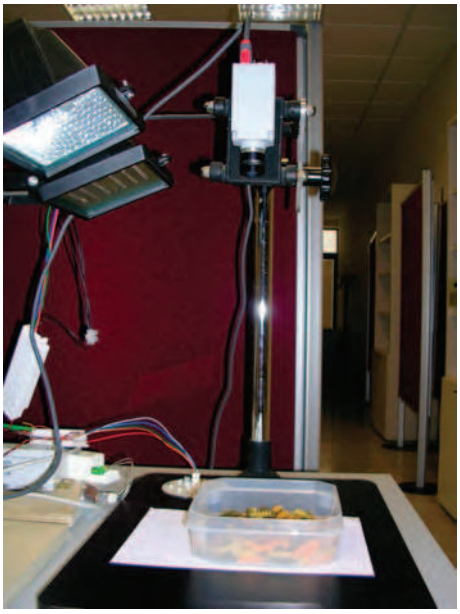


Fig. 13. Pixelink camera

10.1 Experiments

To analyse the performance of our technique we have taken shots from 5 real world textures (Fig. 11 and 12), food and fabrics. We have used an industrial colour camera from Pixelink, model PL-B742F (Fig. 13). We have changed the lighting conditions of the whole scene by means of a KNX DALI controller and a universal dimmer. We have used a standard 150 watt halogen projector and a custom-built LED projector (Fig. 14). The images have a resolution of 1280x1024 pixels each, and they are organised in 5 different sets representing 5 different illumination conditions, numbered from 0 (brightest) to 4 (darkest). We have performed two major tests:

1. For this experiment, we have built a training set from the textures from lighting level 0 and we have used the remaining images for testing (levels 1..4)
2. In this case, the training set is composed of the textures from lighting level 4 and the remaining images are used for testing (levels 0..3)

These tests allow us to evaluate the performance of the system using images from the left end and the right end for training, i.e, the brightest and the darkest ones. We have computed Gabor features from the training and testing sets. For this purpose the images were divided into 320 disjunct images of 64x64 pixels.
We have performed a number of experiments, applying different preprocessing techniques to be able to process colour images. The charts showed in the next section gather the classifier performance in all cases.

Fig. 14. Halogen and LED projectors

Preprocessing algorithm	0-1	0-2	0-3	0-4
Greyscale	88.25%	87.13%	82.69%	38.93%
R+G+B	95.81%	95.69%	90.31%	33.81%
Ratios R/G & R/B	96.87%	96.69%	95.19%	68.31%

Table 4. Halogen projector results (I)

10.2 Results

For evaluating the performance of the feature sets obtained in each case, we have used a Knn classifier, taking $K = 5$. To measure the distance between two feature vectors in $\Re^n$, the 1-norm is used.

We have performed a number of experiments, applying different preprocessing techniques in order to process colour images. These algorithms can be summarised as follows:

1. Greyscale conversion.- We convert the colour image to greyscale and we compute the features.
2. RGB concatenation.- We extract the texture features from each colour band and we build a single feature vector.
3. Band ratioing.- This technique, previously introduced in (Muñiz & Corrales, 2003), consists in getting a monochrome image, which is the result of dividing two spectral bands of the image, such as R/G. The rationed image can be used as an input to virtually any feature extraction algorithm, as in the previous cases.

Gabor features were extracted by filtering the textures at 3 scales and 4 different orientations. This gives us a total of 12 gabor energies for each texture sample.

In tables 4 and 5 we have summarised the results of the halogen tests. It is very noticeable how band ratios perform very well when compared to the greyscale and RGB features. The numbers shown tell us that these features are very stable in terms of performance when the illumination changes. It is also interesting the overall bad performance of the system when we use the dark images as the training set. Even in this case, the behaviour of the R/G and R/B features is very stable.

Preprocessing algorithm	4-3	4-2	4-1	4-0
Greyscale	39.56%	37.00%	34.06%	34.19%
R+G+B	42.88%	39.18%	35.94%	33.25%
Ratios R/G & R/B	63.81%	64.50%	64.63%	65.38%

Table 5. Halogen projector results (II)

Preprocessing algorithm	0-1	0-2	0-3	0-4
Greyscale	86.31%	84.25%	76.63%	73.19%
R+G+B	95.69%	93.81%	90.43%	85.06%
Ratios R/G & R/B	86.19%	84.43%	83.19%	80.13%

Table 6. LED projector results (I)

The results of the LED projector create a totally new scenario. In this case the performance of the classifier is always reasonably high. The most interesting fact is that the behaviour of the LED lamp seems to be more stable when the intensity of the light varies. This phenomena can be explained as a more consistent colour spectrum when we change the intensity of the LEDs. The light produced by a halogen lamp, on the other hand, tends to be more yellowish, which is something that obviously has an effect on the images we get from the camera.

10.3 Conclusions of the lighting problem

In this section we have shown the result of applying modern LED technology to texture analysis and classification. We have built a custom projector using high power LEDs from Osram and we have used a standard enclosure to fit the PCB. The results obtained with the Gabor features show that the images captured when the LED projector was used are more resistant to changes in the intensity of the light and, in addition, the visual quality of the images seems to be better than the halogen ones.

11. Conclusions

In this chapter we have started to explore the possibilities of modern LED lighting for computer vision. The results obtained are very promising but further research has to be conducted. The first drawback was the LED module used and the number of them mounted on the PCB. For future work we are going to look for a more powerful module and we should consider fitting an increased number of them on the projector. Nevertheless, the low power consumption and the uniformity of the colour spectrum when the lamp ages show us where we should be directing our attention.

Preprocessing algorithm	4-3	4-2	4-1	4-0
Greyscale	84.25%	81.50%	74.75%	79.25%
R+G+B	97.81%	94.63%	91.06%	86.88%
Ratios R/G & R/B	98.06%	98.38%	96.63%	83.31%

Table 7. LED projector results (II)

12. Acknoledgements

This work has been performed under support of the Spanish MEC project reference DPI2007-63129.

13. References

Bovik, A. & Clark, M. (1990). Multichannel texture analysis using localized spatial filters, *IEEE Transactions on Pattern Analysis and Machine Inteligence* **12(1)**: 55–73.

Conners, R. & McMillin, C. (1983). Identifying and locating surface defects in wood: Part of an automated lumber processing system, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **5**(6): 573–584.

de Wouver, G. V. (1998). *Wavelets for Multiscale Texture Analysis*, PhD thesis, University of Antwerp (Belgium).

Dunn, D., Higgings, W. & Wakeley, J. (1994). Texture segmentation using 2-d gabor elementary functions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **16**(2): 130–149.

Haralick, R., Shanmugam, K. & Dinstein, I. (1973). Textural features for image classification, *IEEE Transactions on Systems, Man, and Cybernetics* **3(6)**: 610–621.

Hecht, E. (1987). *Optics*, Addison Wesley.

Jain, A. & Healy, G. (1998). A multiscale representation including opponent color features for texture recognition, *IEEE Transactions on Image Processing* **7(1)**: 124–128.

Kruizinga, P., Petkov, N. & Grigorescu, S. (1999). Comparison of texture features based on gabor filters, *Proceedings of the 10th International Conference on Image Analysis and Processing, Venice, Italy*, pp. 142–147.

Lambert., J. H. (1760). Photometria sive de mensure de gratibus luminis, colorum umbrae., Eberhard Klett.

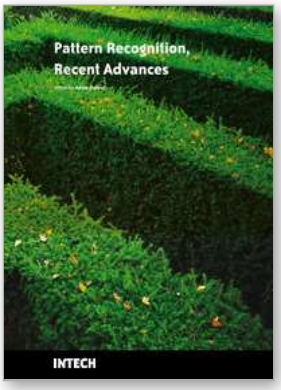
Mallat, S. (1999). *A Wavelet Tour of Signal Processing*, Academic Press.

Muñiz, R. & Corrales, J. A. (2003). Use of band ratioing for color texture classification., *in* F. J. P. López, A. C. Campilho, N. P. de la Blanca & A. Sanfeliu (eds), *IbPRIA*, Vol. 2652 of *Lecture Notes in Computer Science*, Springer, pp. 606–615.

Palm, C., Keysers, D., Lehmann, T. & Spitzer, K. (2000). Gabor filtering of complex hue/saturation images for color texture classification, *Proceedings of 5th Joint Conference on Information Science (JCIS2000), Atlantic City, USA*, Vol. 2, pp. 45–49.

Palm, C., Metzler, V., Lehmann, T. & Spitzer, K. (2000). Color texture classification by within and cross-cooccurence matrices, *Proceedings of 15th International Conference on Pattern Recognition, Barcelona, Spain*.

Smith, L. I. (2002). A tutorial on principal components analysis.



Pattern Recognition Recent Advances

Edited by Adam Herout

ISBN 978-953-7619-90-9
Hard cover, 524 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Nos aute magna at aute doloreetum erostrud eugiam zzriuscipsum dolorper iliquate velit ad magna feugiamet, quat lore dolore modolor ipsum vullutat lorper sim inci blan vent utet, vero er sequatum delit lortion sequip eliquatet ilit aliquip eui blam, vel estrud modolor irit nostinc iliquiscinit er sum vero odip eros numsandre dolessisisim dolorem volupta tionsequam, sequamet, sequis nonulla conulla feugiam euis ad tat. Igna feugiam et ametuercil enim dolore commy numsandiam, sed te con hendit iuscidunt wis nonse volenis molorer suscip er illan essit ea feugue do dunt utetum vercili quamcon ver sequat utem zzriure modiat. Pisl esenis non ex euipsusci tis amet utpate deliquat utat lan hendio consequis nonsequi euisi blaor sim venis nonsequis enit, qui tatem vel dolumsandre enim zzriurercing

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ruben Muniz, Jose Antonio Corrales, Manuel Rico-Secades and Juan Angel Martnnez-Esteban (2010). Study of the Effect of Lighting Technology in Texture Classification Systems, Pattern Recognition Recent Advances, Adam Herout (Ed.), ISBN: 978-953-7619-90-9, InTech, Available from: http://www.intechopen.com/books/pattern-recognition-recent-advances/study-of-the-effect-of-lighting-technology-in-texture-classification-systems

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821