Color Textons for Texture Recognition

Gertjan J. Burghouts and Jan-Mark Geusebroek

Abstract

Texton models have proven to be very discriminative for the recognition of grayvalue images taken from rough textures. To further improve the discriminative power of the distinctive texton models of Varma and Zisserman (VZ model) (IJCV, vol. 62(1), pp. 61-81, 2005), we propose two schemes to exploit color information. First, we incorporate color information directly at the texton level, and apply color invariants to deal with straightforward illumination effects as local intensity, shading and shadow. But, the learning of representatives of the spatial structure *and* colors of textures may be hampered by the wide variety of apparent structure-color combinations. Therefore, our second contribution is an alternative approach, where we weight grayvalue-based textons with color information in a post-processing step, leaving the original VZ algorithm intact.

We demonstrate that the color-weighted textons outperform the VZ textons as well as the color invariant textons. The color-weighted textons are specifically more discriminative than grayvalue-based textons when the size of the example image set is reduced. When using 2 example images only, recognition performance is 85.6%, which is an improvement over grayvalue-based textons of 10%. Hence, incorporating color in textons facilitates the learning of textons.

1   Introduction

The appearance of rough 3D textures is heavily influenced by the imaging conditions under which the texture is viewed [3, 7]. The texture appearance deviates as a consequence of a change of the recording setting. Among others, the imaging conditions have an influence on the texture shading, self-shadowing and interreflections [3], contrast and highlights. Texture recognition [7, 8, 1] and categorization [6] algorithms have been proposed to learn or model the appearance variation in order to deal with varying imaging conditions. In this paper, we consider the challenge of recognizing textures from few examples [8], for which discriminative models are needed to distinguish between textures, but also invariance of models is required to generalize over texture appearances. Note that texture recognition differs from texture categorization [6], where also generalization is needed over various textures belonging to one category.

Texture recognition methods based on texture primitives, i.e. textons, have successfully learned the appearance variation from grayscale images [8]. Although color is a discriminative property of texture, the color texture appearance model of [7] was tested on the Curet dataset [3] and has been outperformed by the grayvalue-based texton model [8]. This can be explained partly from the use of color image features that are not specific

for texture, e.g. raw color values [7], and partly from using color features that are not robust to the photometric effects that dominate the appearance variation of textures. In this paper, we aim at describing robustly both spatial structure and color of textures to improve the discriminative power for learning textures from few examples. Due to their high discriminative power, we extend the texton models of Varma and Zisserman (VZ) [8] to incorporate robustly color texture information.

Textons are typical representatives of filter bank responses. The MR8-filterbank [8], on which VZ is based, is designed such that it describes accurately the spatial structure of texture appearances [8]. A straightforward extension of the grayvalue-based MR8 filterbank would be to apply it to each channel of a multivalued image to describe the spatial structure of color images. However, the true color variations and the appearance deviations due to e.g. shading, interreflections and highlights are manifold. Hence, incorporating color information directly in the filter bank requires many examples to learn the color textons well. Moreover, color textons that are learned directly from the data may not be representative for all appearance deviations in the dataset, with the consequence that the representation of each texture will become less compact. Color invariants (e.g. [4]) have provided means to capture only object-specific color information which simplifies the learning and representation of appearances. However, this leaves one with a suitable choice of color invariant features. This is a nontrivial problem as most color invariants aim to disregard intensity information [4], which is a very discriminative property of textures [7, 8]. A change of the local intensity level is a common effect when textures are viewed under changing settings of the illumination and camera viewpoint [7]. Our first contribution is to propose color texture invariants that are largely insensitive to the local intensity level, while maintaining local contrast variations.

The learning of representatives of the spatial structure *and* colors of textures may be hampered by the wide variety of apparent structure-color combinations. An alternative approach to incorporate color directly, would be to incorporate color information in a post-processing step, leaving VZ intact. We propose a color-based weighting scheme for the coloring of grayvalue-based textons. The weighting scheme is based on the characteristics of color invariant edges, based on non-linear combinations of Gaussian derivative filters [5]. The Gaussian filter provides robustness to image noise. The quality of the color images may be poor, hence uncertainties are introduced in the extraction of color edge information. We characterize locally the color edges by their magnitude and direction, where we propagate magnitude and direction uncertainties to obtain a robust color edge model. We exploit this model to provide an efficient color-weighting scheme to extend VZ to incorporate color information.

We consider the recognition of textures from few examples, for which challenging datasets, containing a wide variety of appearances of textures, have been recorded. The Curet dataset [3] contains color images of textures under varying illumination and viewing direction. Recognition rates of 77% have been reported when textures are learned from two images only [8]. In this paper, we improve VZ's discriminative power to increase recognition performance when learning Curet textures from few images.

The paper is organized as follows. In Section 2 we shortly overview the original texton algorithm by Varma and Zisserman [8]. To incorporate color, we consider two alternative modifications of the VZ algorithm, as introduced above. In Section 3, we repeat the experiments of [8] to investigate the discriminative power of (a) the grayvalue-based textons, (b) the grayvalue-based textons plus weighting, and (c) color invariant textons.

2 Combining Textons and Color Texture Information

2.1 VZ

Before we propose two alternative modifications of the original grayvalue-based texture recognition algorithm of Varma and Zisserman (VZ) [8], we briefly overview it.

The VZ algorithm normalizes all grayvalue images to zero mean and unit variance. The MR8-filterbank is convolved with a train set of grayvalue images. The filters are *L2*-normalized and their outputs are rescaled according to Weber's law. See [8] for details.

From the filterbank-responses, textons are learned by performing *k*-means clustering (Euclidean distance), yielding a texton dictionary. The texton dictionary is found to be universal: a different learn set achieves similar results.

Next, each image is represented as a texton model. To that end, each image is filtered with the MR8 filter bank and at each pixel the texton that is closest in feature space is identified. The texton model of an image is a histogram, where each bin represents a texton and its value indicates the number of occurrences of the texton as it occurs in the image.

2.2 VZ-color

As a first attempt to extend the VZ algorithm to use color information, we incorporate color directly at the filterbank level [9]. Here, we extend the original MR8-filterbank [8] to filter color channels directly, where we manipulate the color channels to obtain invariance to intensity changes. First, each image is transformed to opponent color space. Using opponent color space, we benefit from the advantage that the color channels are decorrelated. As a consequence, the intensity channel is separated from the color chromaticity values. The transformation from RGB-values to the Gaussian opponent color model is given by [5]:

$$
\begin{bmatrix} \hat{E}(x,y) \\ \hat{E}_\lambda(x,y) \\ \hat{E}_{\lambda\lambda}(x,y) \end{bmatrix} = \begin{pmatrix} 0.06 & 0.63 & 0.27 \\ 0.30 & 0.04 & -0.35 \\ 0.34 & -0.60 & 0.17 \end{pmatrix} \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix}, \tag{1}
$$

where $\hat{E}$, $\hat{E}_\lambda$ and $\hat{E}_{\lambda\lambda}$ denote the intensity, blue-yellow and green-red channel.

The VZ-algorithm has shown to deal with intensity information in a very robust manner, by normalizing the grayvalue image first to zero mean and unit variance, thereby obtaining a large degree of invariance to changes of the viewing or illumination settings. We normalize the intensity channel in the same way.

We propose a physics-based normalization of the color values, such that the color values are invariant to local intensity changes, we term this scheme VZ-color. Here, the color values are rescaled by the intensity variation, but not normalized to zero mean to avoid further loss of chromaticity information. We start with the model: for direct and even illumination, the observed energy E in the image may be modelled by:

$$
E(x,\lambda) = i(x)e(\lambda)R(x,\lambda), \tag{2}
$$

where $i(x)$ denotes the intensity which varies over location x, effectively modelling local intensity including shadow and shading. Further, $e(\lambda)$ denotes the illumination spectrum, and $R(x,\lambda)$ denotes object reflectance depending on location x and spectral distribution which is parameterized by λ. Depending on which parts of the wavelength spectrum

are measured, $E(x,\lambda)$ represents the reflected intensity, and $E_\lambda(x,\lambda)$ compares the left and right part of the spectrum, hence may be considered the energy in the "yellow-blue" channel. Likewise, $E_{\lambda\lambda}(x,\lambda)$ may be considered the energy in the "red-green" channel. The actual opponent color measurements of $E(x,\lambda)$, $E_\lambda(x,\lambda)$ and $E_{\lambda\lambda}(x,\lambda)$ are obtained from RGB-values by Equation 1.

A change of the region's intensity level is a common effect when textures are viewed under changing settings of the illumination and camera viewpoint. We consider manipulations of $E(x,\lambda)$, $E_\lambda(x,\lambda)$ and $E_{\lambda\lambda}(x,\lambda)$ to obtain some invariance to such appearance changes. With the physical model from Equation 2, the measured intensity $\hat{E}$ can be approximated by:

$$\hat{E}(x,\lambda) \approx i(x)e(\lambda)R(x,\lambda). \tag{3}$$

For the spectral derivatives, we obtain the approximations:

$$\hat{E}_\lambda(x,\lambda) \quad \approx \quad \frac{d}{d\lambda}i(x)e(\lambda)R(x,\lambda) = i(x)\frac{d}{d\lambda}e(\lambda)R(x,\lambda), \tag{4}$$

$$\hat{E}_{\lambda\lambda}(x,\lambda) \quad \approx \quad i(x)\frac{d}{d\lambda\lambda}e(\lambda)R(x,\lambda). \tag{5}$$

We obtain the color measurements $\hat{E}_\lambda(x)$ and $\hat{E}_{\lambda\lambda}(x)$ directly from RGB-values according to Equation 1. The global variation in these color measurements due to variations of illumination intensity, shadow and shading is approximated by $i(x)$. The intensity measurement $\hat{E}(x)$, also directly obtained from Equation 1, is a direct indication of the intensity fluctuation. Therefore, the standard deviation of $\hat{E}$ over all pixels is used to normalize globally each of the color measurements $\hat{E}_\lambda(x)$ and $\hat{E}_{\lambda\lambda}(x)$, thus dividing by $\sigma(\hat{E})$, to obtain better estimates of the actual color variation. We do a global normalization here, and not per pixel, as local intensity variation in the color channels is considered important texture information. Finally, the MR8-filterbank is applied to these 3 color invariant signals.

2.3 `VZ-dipoles`

As an alternative to incorporating color at the level of the filterbank, the `VZ` algorithm is extended by a post-processing step where textons are weighted according to the color edge at the location of a particular texton. Color edges are measured by color gradients, of which the magnitudes and directions are used to characterize texture edges (subsection 2.3.1). The directions of color gradients are taken relative to the direction of the intensity gradient. Weights are computed to determine to which degree the color gradient direction corresponds to the intensity direction. The weights are combined to obtain an indication of the color transition at the location of a particular texton (subsection 2.3.2), with which the texton is weighted when adding it to the texton histogram (subsection 2.3.3). This process is outlined in Figure 1.

2.3.1 Color Invariant Gradients

To exploit color information in a robust fashion, we base ourselves on noise-robust Gaussian image measurements. From these measurements, we extract color invariant gradients that are robust to changes of the intensity level [5], to achieve the same level of invariance as in the previous subsection. We overview shortly the derivation of color invariant

d₊ φ weights for dλ₊ and dλλ₊
get smaller for larger φ
(and weights for dλ₋ and dλλ₋ will get larger)

d₋

direction
spaces
relative to
d₊ and **d₋**

dλ₊ = 0.85
dλ₋ = 0.05 dλλ₊ = 0.65
dλλ₋ = 0.10

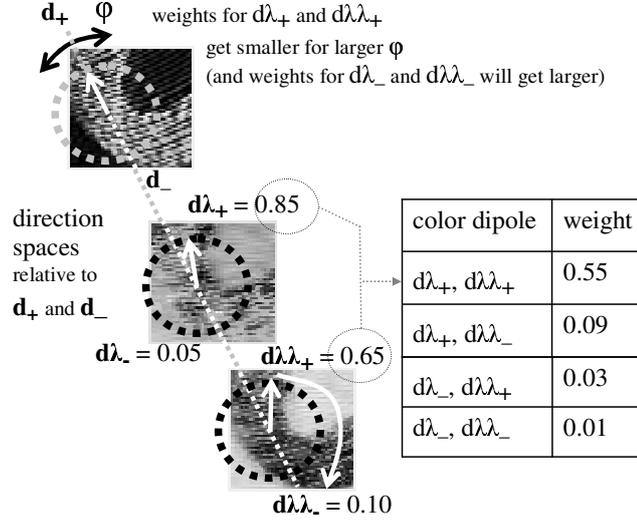| color dipole	weight
dλ₊, dλλ₊	0.55
dλ₊, dλλ₋	0.09
dλ₋, dλλ₊	0.03
dλ₋, dλλ₋	0.01

Figure 1: The color dipole framework. The three images denote the color representation of a color texture. For each color channel, the gradient is computed, depicted by the arrows. The direction of the opponent color gradients (d_λ and $d_{\lambda\lambda}$) are taken relative to the direction of the intensity gradient (d). For both the same ($+$) and opposite ($-$) direction, weights are determined. The smaller the weight gets for one direction, the larger it gets for the opposite direction. To obtain a combined weight for each combination of directions, the weights are multiplied.

gradients. First, we consider the transformation of RGB-values to opponent color space, yielding opponent color values E, E_λ and $E_{\lambda\lambda}$ to represent the intensity, blue-yellow and green-red channel, respectively, likewise the previous subsection (Equation 1) . From the opponent color values, spatial derivatives in the x-direction are computed by convolution with Gaussian derivative filters $G_x^\sigma(x,y)$ with scale σ:

$$\hat{E}_x^\sigma(x,y) = E(x,y) * G_x^\sigma(x,y), \tag{6}$$
$$\hat{E}_{\lambda x}^\sigma(x,y) = E_\lambda(x,y) * G_x^\sigma(x,y), \tag{7}$$
$$\hat{E}_{\lambda\lambda x}^\sigma(x,y) = E_{\lambda\lambda}(x,y) * G_x^\sigma(x,y), \tag{8}$$

where ($*$) denotes convolution. The spatial derivatives of opponent color values, $\hat{E}_x^\sigma$, $\hat{E}_{\lambda x}^\sigma$ and $\hat{E}_{\lambda\lambda x}^\sigma$, are transformed respectively into color invariants $\mathcal{W}_x^\sigma$, $\mathcal{W}_{\lambda x}^\sigma$ and $\mathcal{W}_{\lambda\lambda x}^\sigma$ providing robustness to changes of the intensity level by normalizing by the local intensity $\hat{E}^\sigma$:

$$\mathcal{W}_x^\sigma(x,y) = \frac{\hat{E}_x^\sigma(x,y)}{\hat{E}^\sigma(x,y)}, \; \mathcal{W}_{\lambda x}^\sigma(x,y) = \frac{\hat{E}_{\lambda x}^\sigma(x,y)}{\hat{E}^\sigma(x,y)}, \; \mathcal{W}_{\lambda\lambda x}^\sigma(x,y) = \frac{\hat{E}_{\lambda\lambda x}^\sigma(x,y)}{\hat{E}^\sigma(x,y)}. \tag{9}$$

This normalization may become unstable for low pixel values, but with the local smoothing some robustness to noise is obtained.

The color invariant features are computed at multiple scales to obtain scale invariance. We compute each scale-normalized invariant at 3 scales ($\sigma \in \{1,2,4\}$ pixels)

and select the scale of the invariant that maximizes the response. Next, the color invariant gradients are computed. The gradient magnitude is determined from: $\hat{\mathscr{W}}_{\lambda^i w} = \sqrt{\hat{\mathscr{W}}_{\lambda^i x}(x,y)^2 + \hat{\mathscr{W}}_{\lambda^i y}(x,y)^2}$, whereas its direction is determined from: $\arctan(\frac{\hat{\mathscr{W}}_{\lambda^i y}(x,y)}{\hat{\mathscr{W}}_{\lambda^i x}(x,y)})$. We obtain per pixel the color and scale invariant gradients $\hat{\mathscr{W}}_w$, $\hat{\mathscr{W}}_{\lambda w}$ and $\hat{\mathscr{W}}_{\lambda\lambda w}$. After application of the color invariants to the image set that is used for training (see Experiments), we learn their standard deviation. We normalize each invariant by its standard deviation, which effectively boosts color information.

2.3.2 Color Dipoles

An edge in a color image may be characterized by measuring for each color channel the energy gradient, as outlined in [10]. In order to exploit the a-priori structure in texture images, we investigate the correlation between intensity edges and color edges. Therefore, we determine at each pixel the orientation of intensity and color gradients, and measure the correlation between the orientations of intensity and color gradients over all pixels in the Curet dataset [3]. To measure the correlation between orientations at edge locations only, we determine the weighted correlation, where weights are provided by the total gradient magnitude at a particular pixel, measured by $\sqrt{\mathscr{W}_w(x,y)^2 + \mathscr{W}_{\lambda w}(x,y)^2 + \mathscr{W}_{\lambda\lambda w}(x,y)^2}$. The orientations of intensity and color gradients are strongly correlated: $r(\mathscr{W}_w, \mathscr{W}_{\lambda w}) = 0.77$, $r(\mathscr{W}_w, \mathscr{W}_{\lambda\lambda w}) = 0.81$ and $r(\mathscr{W}_{\lambda w}, \mathscr{W}_{\lambda\lambda w}) = 0.82$.

We have observed that edges are largely characterized by color gradient magnitudes, and whether these gradients are directed in the same or opposite direction as the intensity gradient. The characterization of a color edge by a dichotomic framework is termed a color dipole. An example of a color dipole is displayed in Figure 1. The figure also displays poor image quality, indicating that robust modelling of color information is required.

We start with the alignment of the color dipole framework to the direction of the intensity gradient $\mathscr{W}_w$. The direction of $\mathscr{W}_{\lambda w}$ is compared to the direction of $\mathscr{W}_w$. Two Gaussian kernels in direction-space measure the certainty that the direction is the same or opposite to the intensity gradient direction, see Figure 2. The choice of the size of the kernels has no significant effect on texture recognition results (data not shown). The
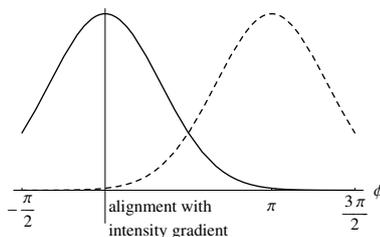

Figure 2: Two Gaussian kernels in direction-space measure the certainty that the direction is the same or opposite to the intensity gradient direction.

kernels in direction-space yield 2 direction weights, one for the same direction as the intensity gradient and one for the opposite direction. The more the direction and opposite direction differ with respect to the direction and opposite direction of the intensity gradi-

ent, the lower the weight. Also, the smaller the weight gets for one direction, the larger it gets for the opposite direction. Analogously, two direction weights are determined for the gradient $\mathscr{W}_{\lambda\lambda w}$.

In total, we obtain for each feature two direction weights per pixel, which for two features yields $2 \times 2 = 4$ combinations. For each of the four combinations, we obtain a single weight by multiplying the two corresponding feature direction probabilities, see Figure 1. For each of the four dipole possibilities, we have obtained a single weight representing the probability that the edge under investigation is characterized by it.

To ensure that the feature directions are stable, we weight the dipole framework per pixel by the total edge strength $\sqrt{\mathscr{W}_w(x,y)^2 + \mathscr{W}_{\lambda w}(x,y)^2 + \mathscr{W}_{\lambda\lambda w}(x,y)^2}$ at that pixel. We normalize the sum of all weights over the image to unity. The dipole framework provides robustly the probability for each of the four color dipoles per pixel.

2.3.3 Color-weighted Textons

The color-weighting scheme only extends the VZ-algorithm in the way in which the occurrences of grayvalue-based textons contribute to the histogram bins of the texton model. Rather than accumulating a unity weight for each occurrence of a particular texton, we add weights according to the dipole measured at the location of interest. Since we have four weights, each of the original histogram bins of VZ are split into four, such that each of the four weights per texton can be added to the four bins that correspond to the particular texton. Like VZ, the histograms are normalized to unity, and compared using the χ^2-statistic.

In recapitulation, the VZ-dipole algorithm affects only the cardinality of the texton model. Hence, VZ-dipole is a low-cost strategy to obtain colored textons, while avoiding the introduction of essentially different textons for the learning and representation of textons in the image. The color invariant textons, VZ-color-norm, affects the cardinality of the filterbank. In addition, the learning of textons from the color-based VZ-color-norm filterbank requires a learn set that is both representative of the texture shape primitives in the dataset as well as their colors. Table 2 summarizes the proposed modification of and extension to the original grayvalue-based texture recognition VZ algorithm [8].

Table 1: Characteristics of the VZ algorithm and proposed modifications of VZ.

	Texton learn set representative of	Size of filterbank	Size of representation
VZ	texture shape primitives	8	# textons
VZ-color	texture shape primitives *and* colors	24	# textons
VZ-dipoles	texture shape primitives	8	$4 \times$ # textons

3 Texture Recognition Experiment

In this section, we demonstrate the discriminative power of colored textons for texture recognition. We follow the experimental setup of Varma and Zisserman [8] to classify the 61 textures of the Curet dataset [3]. Textons are learned from the same 20 textures as used in [8] and [2]. For each texture, 13 random images are convolved with the MR8-filterbank [8], from which all responses are collected and 10 cluster means are learned to obtain 10 textons. Hence, using 20 textures to learn textons from, 200 textons are learned; this is the texton dictionary. For each of the 61 textures in the Curet dataset, 92 images have been selected by [8] to obtain a total of 5612 images. Each image is represented by a histogram of grayvalue-based texton frequencies [8].

For the recognition of textures, we also follow [8]. To classify textures, each texture is represented by 46 models obtained from alternating images in the total of 92 images per texture. These 46 models are the learning set; the remaining 46 images are test images.

3.1 Baseline Performance

We consider the recognition of only 20 textures as used in [8] and [2], based on all 46 models. The VZ algorithm, termed VZ, based on grayvalue-textons achieves a recognition performance of 97.8%. With the physics-based normalization of opponent color values, exploited in VZ-color, the results are better: 98.4%. With the dipole-weighted textons, VZ-dipoles, the highest performance is achieved: 98.7% of the 20 textures is classified correctly.

Due to their improvement in recognition performance over grayvalue-based textons, we consider VZ-color and VZ-dipoles in comparison to VZ for the recognition of all 61 textures from the Curet dataset. As a baseline, with VZ, the accuracy of classifying all 61 textures is: 96.4%.

With VZ-color, a recognition accuracy of 97.1% is achieved. This is a good result, but we want to know the effect of the choice of the image set to learn textons from. To that end, we have selected randomly alternative sets of images to learn the textons from. We have conducted 10 trials, for each trial random images are taken from the textures used in [8] and [2]. For VZ-color, the texture recognition results depend significantly on the texton learn set: recognition accuracy varies from 92.7% to 97.1%, while for the grayvalue-based textons the results vary mildly from 96.0% to 96.4%. We conclude that the learning of discriminative color textons is more sensitive to the choice of the learn set.

Because for grayvalue textons the choice of the learn set is of much less importance, the results obtained with VZ-dipoles are stable under the choice of the learn set: recognition accuracy varies from 96.1% to 96.5%. It should be noted that 200 textons are used, identical to the textons used in VZ, but with 4 weights attached to each texton. Increasing the number of textons to 800 increases only very marginally the performance of VZ [8].

3.2 Reducing the Learn Set

To test the recognition accuracy when fewer models are incorporated in the learn set, we start to decrease the number of learn models, likewise [8]. The learn set is reduced by discarding models that contribute least to the recognition performance. Models were discarded in each iteration step based on a greedy reduced nearest-neighbor algorithm.

We emphasize that, by reducing the number of models, first the noisy models are discarded, improving the texture recognition performance. Here we consider the performance of the algorithms VZ and VZ-dipoles, which have demonstrated most stable under the choice of the texton learn set (see above). Experiments over all 61 textures, where models are removed from the learn set by means of the reduced nearest neighbor rule, the best recognition accuracy obtained with the color textons of VZ-dipoles is 98.3%. Thus, the best results obtained with VZ-dipoles are somewhat lower than achieved by Broadhurst: 99.2% [1]. Broadhurst modelled filterbank responses directly, i.e. without the abstraction step of modelling textons, by a 26-dimensional Gaussian, which was subsequently used in a Bayes recognition framework. It is interesting that the compact texton models achieve a performance that is almost similar to the performance of the elegant models proposed by Broadhurst.

It is interesting how the recognition accuracy decreases when using only few learning models. When 2 models are used, the texture recognition performance increases from 77.1% (VZ) to 85.6% (VZ-dipoles) when including color information. We conclude that exploiting color information facilitates the learning of texture appearances.

The results can be summarized as follows. VZ-dipoles outperforms consistently both the original VZ textons as well as the color textons obtained from color invariant filterbank responses, see Table 3. Interestingly, the results obtained with the color-weighted textons (VZ-dipoles) are most stable over: (a) image sets to learn textons from, and, more importantly, (b) image sets to learn textures from.

Table 2: Performance of the VZ algorithm and proposed modifications of VZ.

	Textures:	20	61		61
	Textons:	20	20		20
	Models:	46	46		2
Algorithm			best	worse	
VZ		97.8%	96.4%	96.0%	77.1%
VZ-color		98.4%	97.1%	92.7%	—
VZ-dipoles		98.7%	96.5%	96.1%	85.6%

4 Conclusion

In this paper, we have proposed methods to incorporate robustly color information in VZ textons [8] to model the appearance of textures. The textons are learned from filterbank responses. First, we have incorporated color directly at the level of the filterbank. We have shown that the learning of discriminative color textons that are representative of both the textures' shape primitives and colors is not trivial and the recognition accuracy is very dependent on the set of images to learn the color textons from.

As an alternative to incorporate color directly at the filterbank, we have proposed a color weighting scheme to weight grayvalue-based textons by the color edges that generate the texture. This framework captures robustly essential color texture information, is efficient to compute, and provides a simple extension to the original texton model.

In the experiments, we have modelled color texture images from the Curet dataset by the traditional textons and the color-weighted textons. With color-weighted textons, the texture recognition performance is increased significantly, up to ten percent when only two texton models per texture are used. Incorporating color in a robust manner by means of the proposed dipole model adds discriminative power for texture recognition, which facilitates the learning of color textures.

References

[1] R. E. Broadhurst. Statistical estimation of histogram variation for texture classification. In *Proceedings of Texture 2005*, pages 25–30, 2005.

[2] O. G. Cula and K. J. Dana. 3d texture recognition using bidirectional feature histograms. *International Journal of Computer Vision*, 59(1):33–60, 2004.

[3] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999.

[4] B. V. Funt and G. D. Finlayson. Color constant color indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):522–529, 1995.

[5] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts. Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1338–1350, 2001.

[6] E. Hayman, B. Caputo, M. Fritz, and J. O. Eklundh. On the significance of real-world conditions for material classification. In *Proceedings of the European Conference Computer Vision*, number 3, pages 253–266. Springer Verlag, 2004.

[7] P. Suen and G. Healey. The analysis and recognition of real-world textures in three dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(5):491–503, 2000.

[8] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1-2):61–81, 2005.

[9] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *Proceedings of the International Conference Computer Vision*, pages 1800–1807. IEEE Computer Society, 2005.

[10] S. Di Zenzo. Note: a note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing*, 33(1):116–125, 1986.