Colour-texture analysis of paintings using ICA filter banks

Nanne van Noord NANNE@TILBURGUNIVERSITY.EDU
Tilburg University, Warandelaan 2, 5037 AB Tilburg, The Netherlands

Eric Postma E.O.POSTMA@TILBURGUNIVERSITY.EDU
Tilburg University, Warandelaan 2, 5037 AB Tilburg, The Netherlands

Ella Hendriks HENDRIKS@VANGOGHMUSEUM.NL
Van Gogh Museum, Postbus 75366, 1070 AJ Amsterdam, The Netherlands

Keywords: colour-texture analysis, independent component analysis, image segmentation, colour

Abstract

In human and computer vision, the analysis of visual texture is of pivotal relevance for object recognition and scene understanding. The analysis of texture defined by both intensity and colour variations contributes to the automatic classification in satellite images, medical images, or food processing. Biologically motivated studies have proposed Independent Component Analysis (ICA) as a model for the analysis of colour texture in human vision. To aim of this paper is to determine the effectiveness of ICA for colour texture analysis in computer vision. The effectiveness of ICA on the segmentation of paintings is assessed in unsupervised and supervised settings. The results are encouraging and suggest ICA to be a suitable basis for colour texture analysis.

1. Introduction

Both in biological and artificial vision systems, visual texture provides an important cue for the recognition of objects and scenes (Tuceryan & Jain, 1993). Visual texture refers to the visual appearance of, for instance, textile or surfaces. Texture analysis enables the measurement and quantification of visual texture. Two common applications of texture analysis are texture classification and texture segmentation. In texture classification, images or image patches are classified into useful classes. For instance, the texture of pieces of fruit may be classified according to the type of fruit. In texture segmentation images are subdivided into similarly-textured regions. For instance, texture segmentation can be used to automatically subdivide satellite images into different regions. Although texture is traditionally analysed by considering the spatial variation in pixel intensity (grey scale values) (Tuceryan & Jain, 1993), the application examples given underline the fact that most natural textures are defined in terms of both intensity and colour (see, e.g., (Cheng et al., 2001)). Such textures are called colour textures (Drimbarean & Whelan, 2001). In recent years, colour texture analysis became an active research field. The main challenge in this field is to find an effective way to integrate intensity and colour cues in colour texture analysis (Fernandez & Vanrell, 2012; Wang et al., 2011).

The current study is part of a research project that aims to support painting conservators in their assessments of the ageing of paints. Paintings consist of different types of pigment applied to a canvas. A high-resolution digital reproduction of a painting can be considered to form a mixture of colour texture components, corresponding to the various pigments, brush strokes, and the canvas. From a signal- or image-processing perspective, the problem of recovering the components is a so-called blind source separation problem (Kleinsteuber & Shen, 2012). A widely used algorithm for blind source separation is Independent Component Analysis (ICA) (Hyvärinen & Oja, 2000). ICA is concerned with finding an unmixing matrix by which a mixed signal can be decomposed into its source signals.

Wachtler, Lee, & Sejnowski (2001) showed that ICA is a biologically plausible method for the analysis of nat-

Appearing in *Proceedings of BENELEARN 2013*.

ural scenes and that ICA, when applied to colour images, yields integrated representations of intensity and colour information. Such ICA representations may be invoked as filter banks for colour texture analysis (Jenssen & Eltoft, 2003; Chen et al., 2006). While filter banks for image (or texture) segmentation commonly consist of pre-defined Gabor or Wavelet filters (Ni et al., 2013; Fan, 2012; Jain & Farrokhnia, 1991; Wang et al., 2011). ICA-based filter banks have been shown to outperform Gabor filters (Chen et al., 2006).

To aim of this paper is to determine the effectiveness of ICA for colour texture analysis in both an unsupervised and a supervised setting. Unsupervised ICA-based colour texture segmentation has been studied before (Cheng et al., 2003), but the authors report only qualitative results. To the best of our knowledge, we are the first to study ICA-based colour texture analysis in a supervised and quantitative setting.

The outline of the remainder of the papers is as follows; in Section 2 the ICA method is described. Then, in Section 3 the experimental setup is described, detailing the dataset, experiments and evaluation procedure. The experimental results are presented in Section 4 and the discussion of the results and conclusions are given in Section 5.

2. Independent Component Analysis for colour texture analysis

Our method for performing Independent Component Analysis to colour texture analysis consists of two steps: (1) finding the independent components and treat them as filters, and (2) convolving the ICA filters with an image. In this work the fixed-point algorithm FastICA by Hyvärinen and Oja (2000) is used for all ICA operations.

ICA is a variant of factor analysis that directly finds the independent components of any non-Gaussian distribution (Hyvärinen & Oja, 1997).

2.1. Finding the independent components

The underlying assumption of ICA is that a collection of observed signals or vectors $\mathbf{x}$ consist of statistically independent components (Hyvärinen & Oja, 1997). These components are denoted by $\mathbf{s}$, and can be found by means of a linear transformation of the observations $\mathbf{x}$ with a weight matrix $\mathbf{W}$;

$$\mathbf{s} = \mathbf{W}\mathbf{x}. \tag{1}$$

Although the weight matrix $\mathbf{W}$ and the independent components $\mathbf{s}$ are unknown, the mixing model can be rewritten as

$$\mathbf{x} = \mathbf{A}\mathbf{s}, \tag{2}$$

where $\mathbf{A}$ represents the mixing matrix. It is possible to estimate both the mixing matrix $\mathbf{A}$ and the components $\mathbf{s}$ from the observed signals $\mathbf{x}$ using an unsupervised learning procedure. $\mathbf{A}$ is obtained such that its (pseudo)inverse $\mathbf{W}$ multiplied by $\mathbf{x}_i$ is an estimate of $\mathbf{s}_i$.

In this work ICA is applied to obtain a filter bank g_i, consisting of size $D \times D \times 3$ filters. A weight matrix $\mathbf{W}$ is learned for each image I_i by applying ICA to a collection of randomly sampled patches of size $D \times D \times 3$. Each collection consists of a fixed amount of $50,000$ image patches, sampled from its respective image I_i.

The resulting filter bank is constructed by reshaping each row of weight matrix $\mathbf{W}$ into a $D \times D \times 3$ filter.

2.2. Convolving the ICA filters with an image

The obtained filter banks are applied by convolving each image I_i with its respective filter bank g_i as follows;

$$\mathbf{G}_i(x, y, z) = \mathbf{I}_i(x, y, z) \otimes \mathbf{g}_i. \tag{3}$$

The outcome of the convolution can be summed across all dimensions in order to calculate the energy distribution as follows;

$$f_i = \sum_{y=1}^{N} \sum_{x=1}^{M} \sum_{z=1}^{P} \mathbf{G}_i(x, y, z). \tag{4}$$

The energy value describes how strong the interaction of a filter is with the image at a given location. The energy distribution for two similar areas is expected to be more similar than the energy distribution of two dissimilar areas. The resulting feature matrix f_i consists of a $MN \times |g_i|$ matrix, where each row describes the energy distribution across colour channels for a given pixel.

The size of the filter bank depends on how many independent components are chosen. For our experiments we varied the number of independent components depending on the colour space and for comparability to other results.

3. Experimental setup

The experimental evaluation of our ICA-based colour texture analysis method is performed by segmenting a painting into paints and primed-canvas regions. Both

Figure 1. Daubigny's Garden by Van Gogh (1890).

an unsupervised and a supervised setting will be examined.

3.1. Dataset

The dataset is extracted from a digital reproduction of a painting by Van Gogh: *Daubigny's Garden*, a 50.7×50.7 cm painting that was created in 1890. Figure 1 shows a reproduction of *Daubigny's Garden*. The key characteristic of this painting is that the primed-canvas is visible in some areas of the painting. Ground truth on these primed-canvas locations is available in the form of an images-sized mask consisting of a binary label for each pixel of the 8176×6132 image of Daubigny's Garden. The dataset consists of a random selection of $50,000$ $D \times D$ patches ($D = 8$).

The images in I_i consist of two typical areas of size 1004×750, examples shown in Figure 2, cropped from the high-resolution image of Daubigny's Garden.

3.2. Unsupervised setting

We employ two unsupervised methods: k-means clustering (Cheng et al., 2003) and graph-based clustering (Felzenszwalb & Huttenlocher, 2004), see also (Peng et al., 2013) .

We apply k-means clustering to the energy distribution f_i obtained by convolving an image with the filter bank. The value of k determines the extent of the segmented regions (large for low values of k and small for high values of k). Segmenting an image in a few large regions will often result in a high recall, due to many of the ground layer pixels being encapsulated inside the same region. However, large regions are also much more coarse, and will thus contain many unwanted pixels, resulting in a low precision. The benefit of having a few large regions is that it is trivial to manually select the best regions. In our experiments, k was set to 5, 10, and 100.

The graph-based clustering method has three adjustable parameters; σ, t and min. The first parameter σ is used to smooth the image using a Gaussian filter before segmenting it. The second parameter t which is used to determine the scale of observation, a higher value of t will result in larger components. The third parameter min is used during post-processing to enforce a minimum region size, preventing too many small regions. All experiments were performed with the recommended values of 0.8 for σ and 20 for min. The value of t is varied between the recommended value of 500 and the, in preliminary experiments determined to be appropriate, value of 350.

3.3. Supervised setting

Supervised algorithms make use of training data, or labeled data in order to learn a model that describes the data and allows for predictions to be made about unseen samples. A supervised classifier operates by learning a mapping, for each instance, of the features to the provided label. For the experiments conducted the instances are represented by single pixels. For each instance the ground-truth is noted by a binary label, indicating whether or not it is part of the ground layer. The features consist of the energy distribution f_i.

Ideally a classifier can be trained on a subset of the dataset and can be used to accurately segment an entire image. Several supervised classifiers are evaluated using a standard k-fold cross validation approach which divides the data in a training and test set.

3.3.1. Classifiers

Three classifiers were used in order to tackle the problem at hand, namely the MATLAB implementations of the k-nearest neighbour (k-NN), naive Bayes, and RUSBoost classifiers.

k-NN assigns the unseen samples the (majority) label of the k nearest neighbouring samples in the training set (Hastie et al., 2003). For all experiments described in this paper the value of k was kept at 1.

Naive Bayes is a probabilistic classifier that assigns labels to unseen sample by using Bayes rule (Hastie et al., 2003).

RUSBoost is an ensemble learner that employs a weak learner, a tree, and is particularly well-suited for dealing with class imbalance (Seiffert et al., 2010). RUSBoost combines the AdaBoost boosting procedure with random under-sampling in order to overcome class imbalance, results are obtained using 10,

(a) (b)

Figure 2. Image regions I_a (a) and I_b (b) extracted from Daubigny's Garden

50 and 100 learners.

3.3.2. Evaluation

The evaluation of our colour texture analysis method is split into an evaluation procedure for the unsupervised setting and one for the supervised setting.

Unsupervised setting. The unsupervised setting may yield more than two different regions. For instance, k-means classifier results in a segmentation into k different regions, although our task requires only a binary segmentation into canvas and non-canvas regions. To deal with this issue, we include only a subset of the k regions, namely those that contribute to the overall performance. A region contributes to the performance when the number of true positives for a region (r_{tp}) is greater than the number of false positives for that region (r_{fp}) multiplied by ϵ. As such the solution to finding the best combination of all regions r_i, r_{best} becomes

$$r_{best} = \{r|0 < (r_{tp} - r_{fp} * \epsilon)\}. \quad (5)$$

The value of ϵ was optimised in preliminary experiments yielding a value of 0.23.

Supervised setting. Two commonly used measures for evaluating the performance of classification systems in a supervised setting are precision and recall, which are also used to evaluate segmentation performance (Martin et al., 2004). Precision is a measure of the correctness of the classification. Fewer mistakes lead to a higher precision. Recall is a measure of completeness. The precision of a generated segmentation is calculated by dividing the number of pixels correctly classified as ground layer (true positives) by the total number of pixels classified as ground layer (true positive + false positive), $precision = \frac{tp}{tp+fp}$. Recall is calculated by dividing the true positives by the true positives and the number of ground layer pixels that were not classified as such (false negatives), $recall = \frac{tp}{tp+fn}$.

As a joint measure of precision and recall, we use the F measure, which is the (harmonic) mean of precision and recall: $F = 2 \cdot (precision \cdot recall)/(precision + recall)$.

4. Results

We start the presentation of results by showing an example of the filter bank obtained by applying ICA to our dataset. Figure 3 is obtained by reshaping each row of matrix $\mathbf{W}$ into a n-dimensional filter (where n denotes the number of colour channels). Evidently, the filters capture spatial and colour information from the dataset.

4.1. Unsupervised results

The results for the unsupervised experiments are presented in Tables 1 and 2, for Figures 2(a) and 2(b), respectively. The features that served as input for the unsupervised experiments were obtained using a filter bank of 64 filters, learned from RGB patches. For the initial experiments we have chosen an under-complete basis of 64 rather than 192 independent components in order to facilitate a fair comparison with the results obtained on grey scale images.

For the k-means algorithm a larger value of k results in a higher F-score and precision. In the case of I_a the recall does not change much when changing the value of k. While for I_b smaller values of k give a better result. A possible explanation for this can be found in the difference between the image regions. The perfor-

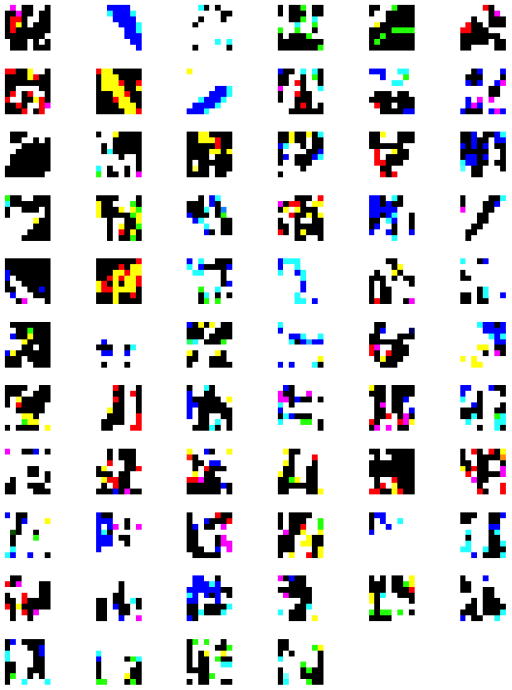


Figure 3. Example of ICA filter bank obtained for $D = 8$.

Table 1. Results of unsupervised algorithms on image I_a

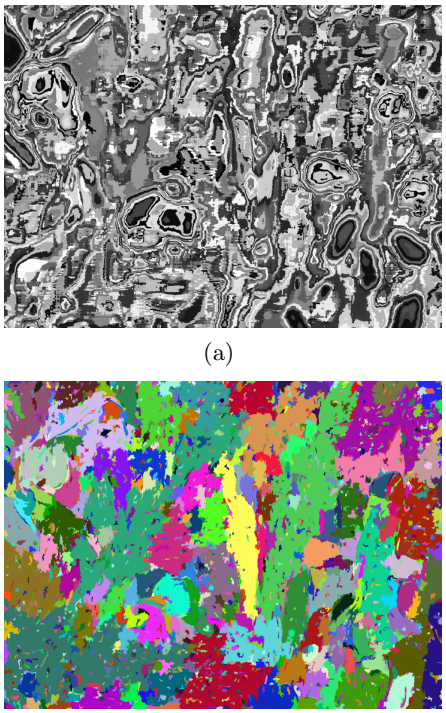
Method	Setting	F-score	precision	recall
k-means	$k = 5$	.309	.230	.469
	$k = 10$	.315	.236	.474
	$k = 100$	.329	.254	.466
graph-based	$t = 350$	.376	.301	.486
	$t = 500$	.349	.303	.413

mance of the graph-based algorithm improves with the lower t value. While the performance with the recommended value ($t = 500$) is on par with k-means (with $k = 100$), the lower value ($t = 350$) results in a better segmentation.

In Figure 4(a) the grey-scale result of the segmentation using k-means with $k = 100$ is shown, which at first glance appears to be more complex than the colour result of the graph-based algorithm shown in Figure 4(b). However, every region in the output of the graph-based algorithm has a randomly-assigned label, which means that identical sections of the image in different locations will be labeled differently. This is not the case for the k-means algorithm that labels identical sections in a consistent way.

Table 2. Results of unsupervised algorithms on image I_b

Method	Setting	F-score	precision	recall
k-means	$k = 5$	.390	.275	.677
	$k = 10$	.393	.280	.660
	$k = 100$	.423	.330	.600
graph-based	$t = 350$	.439	.308	.761
	$t = 500$	.398	.284	.664



(a)

(b)

Figure 4. Segmentation result of Figure 2(a), (a) is the outcome of k-means with $k = 100$, (b) is the outcome of the graph-based algorithm. Each section is indicated by a unique colour.

4.2. Supervised results

The results for the supervised experiments are presented in Table 3 and 4, for Figures 2(a) and 2(b), respectively. As with the results presented in Section 4.1 the results shown in Table 3 and Table 4 are the result of convolutions with filters obtained from the 64 independent components learned from RGB image patches.

The performance of the first classifier, RUSBoost, is about the same for both images. Increasing the number of learners, the main RUSBoost parameter, results in a lower recall, indicating that upscaling the number of of learners will eventually result in lower performance. Similarly to RUSBoost the height of the F-

Table 3. Results of supervised classifiers on image I_a

Classifier	F-score	precision	recall
RUSBoost(10)	.297	.187	.723
RUSBoost(50)	.302	.193	.702
RUSBOOST(100)	.305	.195	.701
Naive Bayes	.281	.174	.730
1-NN	**.556**	.555	.557

Table 4. Results of supervised classifiers on image I_b

Classifier	F-score	precision	recall
RUSBoost(10)	.408	.288	.703
RUSBoost(50)	.419	.308	.652
RUSBoost(100)	.419	.315	.628
Naive Bayes	.378	.251	.766
1-NN	**.532**	.533	.531

score for Naive Bayes stems largely from the high recall, with the precision being very low. Indicating neither classifier is able to accurately distinguish between the different classes and thus assigns the ground-layer label to too large regions.

The precision and recall of the k-NN classifier are very similar, resulting in the highest F-score amongst all classifiers considered. While the recall is lower compared to the other classifiers the higher precision indicates that the k-NN classifier is capable of distinguishing between the majority of the ground-layer pixels and the rest of the painting.

5. Discussion

While the presented results are encouraging we feel that two points are not adequately dealt with, which we address by presenting a selection of preliminary results of our ongoing experiments. The first point concerns the influence of the type of image on the performance. In our current investigations we employ image regions, such as I_c, shown in Figure 5. This image has a stronger distinction between the primed-canvas and paint layers than images I_a and I_b.

The second point is the impact of type of filter bank, colour spaces, and number of features, on segmentation performance. In our studies we focus on new filter banks, such as Log-Gabor filters (Field, 1987), additional colour spaces (e.g., grey-scale and CIELAB), and dimensionality reduction (using PCA).

To provide an impression of how these variations affect the segmentation performance, Table 5 shows the results obtained with the k-NN classifier. The results

Figure 5. Image region I_c extracted from Daubigny's Garden

Table 5. F-scores of the preliminary supervised experiments. Pure refers to results without any PCA preprocessing, PCA(10) are results obtained on the 10 remaining features after PCA.

Filters	Colour space	I_i	Pure	PCA(10)
ICA	Grey	a	.478	.436
		b	.499	.458
		c	.635	.612
	RGB	a	.556	.538
		b	.532	.562
		c	.630	.592
	LAB	a	.528	.519
		b	.565	.510
		c	.646	.638
Log-Gabor	Grey	a	.576	.432
		b	.611	.465
		c	.651	.595
	LAB	a	.572	.595
		b	.617	.465
		c	.652	.596

on image I_c are much better than those on the other two image regions. We suspect that this is due to the greater distinction between the layers present in this image, caused by the increased thickness of the paints applied in this area, completely masking the underlying canvas. The Log-Gabor filters generally perform better than he ICA filters to a degree that depends on the settings. Dimensionality reduction is detrimental for the Log-Gabor filters, but not for ICA.

These preliminary results hint at directions for future study.

6. Conclusion

Our experiments show that ICA in a supervised setting yields better results than in an unsupervised set-

ting. This finding is not surprising, because using the ground truth for modeling textural differences yields a stronger model. More importantly, the k-NN performance obtained with ICA is reasonable and results obtained when analysing colour texture outperform those obtained on grey-scale texture.

While Log-Gabor filters show better results in the setting without PCA, with PCA the ICA filters outperform the Log-Gabor filters. Which is notable as PCA drastically reduces the amount of time required by the classifier. As we extend the method to the entire painting, the size of the dataset increases by a factor of 40, which would put the required time well over what would be feasible for any practical application. Furthermore, there is no difference between the grey scale and colour Log-Gabor settings, which implies that the Log-Gabor method does not efficiently incorporate the additional chromatic information.

Although we have obtained a reasonable performance on a colour texture analysis task using an ICA filter bank, there are two points of concern. First, it is unclear to what extent the complexity of the task influences the performance. A comparable experiment on a painting with a less fine-grained distinction between layers would help in answering this question. Second, the binary labeling of the training data might not have been detailed enough for the classifiers to learn a clear distinction. Further experiments are required to determine whether a richer labeling of the training data would improve performance.

Notwithstanding these concerns, our results are encouraging and suggest ICA to be a suitable basis for colour texture analysis. Future research will focus on exploring ways to improve classification performance by experimenting with conditional random fields, other colour spaces and an extended dataset. We expect that incorporating an interactive user feedback model will improve performance as well as be a practical solution to obtain labeled training data with relatively little effort.

Acknowledgments

The research reported in this paper is performed as part of the REVIGO project, funded by NWO in the context of the Science4Arts research program. The authors thank the anonymous reviewers for their helpful comments.

References

Chen, X.-w., Zeng, X., & van Alphen, D. (2006). Multi-class feature selection for texture classification. *Pattern Recognition Letters*, *27*, 1685–1691.

Cheng, H. D., Jiang, X. H., Sun, Y., & Wang, J. L. (2001). Color image segmentation: Advances and prospects. *Pattern Recognition*, *34*, 2259–2281.

Cheng, J., Chen, Y.-W., Lu, H., & Zeng, X.-Y. (2003). Color- and texture-based image segmentation using local feature analysis approach. *Proc. SPIE 5286, Third International Symposium on Multispectral Image Processing and Pattern Recognition*, 600–604.

Drimbarean, A., & Whelan, P. F. (2001). Experiments in colour texture analysis. *Pattern Recognition Letters*, *22*, 1161–1167.

Fan, J. (2012). Feature learning based multi-scale wavelet analysis for textural image segmentation. In D. Jin and S. Lin (Eds.), *Advances in computer science and information engineering*, vol. 168 of *Advances in Intelligent and Soft Computing*, 461–466. Springer Berlin Heidelberg.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, *59*, 167–181.

Fernandez, S. A., & Vanrell, M. (2012). Texton theory revisited: a bag-of-words approach to combine textons. *Pattern Recognition*, *45*, 4312–4325.

Field, D. J. (1987). Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, *4*, 2379–2394.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2003). *The Elements of Statistical Learning*. Springer. Corrected edition.

Hyvärinen, A., & Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computing Surveys*, *9*, 1483–1492.

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, *13*, 411–430.

Jain, A. K., & Farrokhnia, F. (1991). Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, *24*, 1167–1186.

Jenssen, R., & Eltoft, T. (2003). Independent component analysis for texture segmentation. *Pattern Recognition*, *36*, 2301–2315.

Kleinsteuber, M., & Shen, H. (2012). Blind source separation with compressively sensed linear mixtures. *IEEE Signal Processing Letters*, *19*, 107–110.

Martin, D. R., Fowlkes, C. C., & Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*, 530–549.

Ni, W., Gao, X., & Yan, W. (2013). Sar image segmentation based on gabor filter bank and active contours. In J. Yang, F. Fang and C. Sun (Eds.), *Intelligent science and intelligent data engineering*, vol. 7751 of *Lecture Notes in Computer Science*, 531–538. Springer Berlin Heidelberg.

Peng, B., Zhang, L., & Zhang, D. (2013). A survey of graph theoretical approaches to image segmentation. *Pattern Recognition*, *46*, 1020–1038.

Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., & Napolitano, A. (2010). Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, *40*, 185–197.

Tuceryan, M., & Jain, A. K. (1993). texture analysis. In C. H. Chen, L. F. Pau and P. S. P. Wang (Eds.), *Handbook of pattern recognition & computer vision*, chapter Texture analysis, 235–276. River Edge, NJ, USA: World Scientific Publishing Co., Inc.

Wachtler, T., won Lee, T., & Sejnowski, T. J. (2001). The chromatic structure of natural scenes. *The Journal of the Optical Society of America A*, *18*, 65–77.

Wang, X.-Y., Wang, T., & Bu, J. (2011). Color image segmentation using pixel wise support vector machine classification. *Pattern Recognition*, *44*, 777–787.