Illumination-invariant color image correction

Benedicte Bascle, Olivier Bernier and Vincent Lemaire

France Télécom R&D Lannion, France
benedicte.bascle@francetelecom.com

Abstract. This paper presents a new statistical approach for learning automatic color image correction. The goal is to parameterize color independently of illumination and to correct color for changes of illumination. This is useful in many image processing applications, such as color image segmentation or background subtraction. The motivation for using a learning approach is to deal with changes of lighting typical of indoor environments such as home and office. The method is based on learning color invariants using a modified multi-layer perceptron (MLP). The MLP is odd-layered and the central bottleneck layer includes two neurons that estimates the color invariants and one input neuron proportional to the luminance desired in output of the MLP(luminance being strongly correlated with illumination). The advantage of the modified MLP over a classical MLP is better performance and the estimation of invariants to illumination. Results compare the approach with other color correction approaches from the literature.

1 Introduction

The apparent color of objects in images depends on the color of the light source(s) illuminating the scene. That is why changes in illumination cause apparent color changes in images. Because of this color constancy problem, image processing algorithms using color, such as color image segmentation or object recognition algorithms, tend to lack robustness to illumination changes. Such changes occur frequently in images due to shadows, switching lights on or off, and the variation of sunlight during the day. To deal with this, a color correction scheme that can compensate for illumination changes is needed.

Section 2 presents the state of the art for color correction. Section 3 details our approach, based on learning color correction using a modified MLP. The motivation for this is discussed, and the learning method is described. The approach is compared to using a classical MLP for learning color correction. Section 4 shows experimental results and comparisons.

2 Illumination correction - state of the art

Color in images is usually represented by a triband signal, for instance Red-Green-Blue (RGB). As discussed in the introduction, this signal is sensitive to changes in illumination. However, image processing techniques need to be robust to such changes. Therefore color needs to be parameterized independently of illumination. This can be done by parameterizing color with one or two parameters or by correcting the triband signal. A number of color parametrization and color correction schemes have been described in the literature.

An example of mono-band parametrization of color is hue (from hue-saturation-value, a.k.a. HSV) [GW01]. Examples of bi-band color parameterization are chrominances uv (from the YUV color space) [GW01] and the *ab* values from the CIE Lab color space [GW01]. These three color representations (H, uv or ab) are analytical and thus do not require learning. They are fast pixel-wise methods. They have a limited robustness to illumination changes.

An approach for estimating color invariants from images consists in calculating ratios of RGB components at a given pixel (R/B) or between neighboring pixels (such as $(R_{x_1}G_{x_2})/(G_{x_1}R_{x_2})$) [GS99]. This method is also pixel-wise and thus fast. These invariants are also very robust to illumination changes. However, a lot of information about the original color signal is lost, and reconstructing the original signal from these invariants is difficult.

A more sophisticated method has been proposed by [FDL04]. It estimates a mono-band invariant and is based on a physical model of image formation. It works globally on the image. In $(log(R/B), log(G/B))$ color space, an axis invariant to illuminant color is determined by entropy minimisation. The projection of colors onto a line perpendicular to the invariant axis gives corrected colors. The approach does not require learning and applies to any type of illuminant, but is relatively slow. It also requires that the image contains relatively few different colors and also includes many changes of illumination for each color.

Yet another approach explicitly estimates the color of the illuminant [FCB97]. A neural network estimated the chromaticity of the illuminant from the histogram of chromaticity of the image. The method works globally from the whole image and supposes there is only one illuminant for the entire image.

3 A statistical approach to measure color invariants

3.1 A modified multi-layer perceptron: motivation

The motivation of this work is twofold: (1) to parameterize color compactly and independently of illumination by two invariants (2) to do it in real-time. Firstly, two parameters are needed to parameterize color with enough degrees of freedom to reconstruct a triband signal, given a luminance (or a gray level signal). Secondly, real-time processing (or more exactly video rate processing, e.g. processing 25 or 30 images per second) is also necessary for some applications. This means that methods such as [FCB97] and [FDL04] are out, since they work on the whole image. To obtain real time performance, pixel-wise processing is necessary. Hue-Saturation and uv (from YUV) and ab (from the CIE Lab color space) are three 2-parameter pixel-wise representations of color from the literature that can be calculated in real-time. However they lack robustness

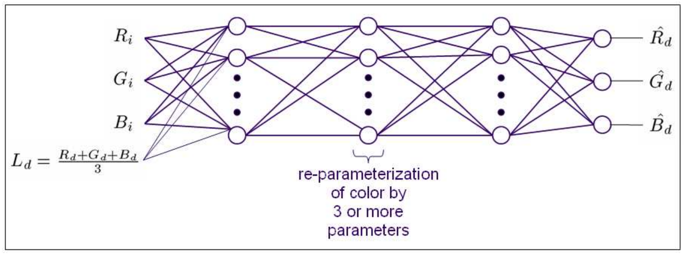

Fig. 1. A classical MLP with 4 inputs can be used to perform color correction. (R_i, G_i, B_i) is the input color. (R_d, G_d, B_d) is the desired output color, corresponding to the same color seen under a different illumination. L is the luminance of the expected output and is a direct function of the illumination. This fourth input neuron prevents the mapping to be learnt by the MLP from including one-to-many correspondences and thus makes it solvable. If the MLP contains a bottleneck layer with 3 neurons, then these perform a re-parameterization of RGB space. However the three color parameters estimated by the 3 neurons have no reason to be invariant to illumination.

to illuminations changes. Mathematical and/or physical models could be used to find a more robust parameterization [GS99]. They are very general, but lose information so that the original color signal is difficult to reconstruct from them. However, in practice, a limited range of illumination sources, and thus a limited range of illumination changes, are available in indoor environments. It is therefore interesting to use learning methods to find a color parameterization invariant to the "usual" illumination changes. While more restricted in their application, such parameters should also be more robust. Another interest of learning about typical illuminants in indoor environments is that it provides global a priori information about the illuminants, so the approach is not completely local (considering the fact that Land's Mondrian experiments showed that illuminant correction cannot be performed purely locally [LM71]). In practice, the lighting customarily found in home and offices comes from fluorescent lights, incandescent light bulbs and natural sunlight from windows. They tend towards the whitish and yellowish areas of the spectrum (very few bluish or reddish lights). These are the sort of illuminants that our approach will deal with.

Our learning method of choice has been neural networks and more specifically multi-layer perceptrons (or MLPs), for their ease of use and adaptability. The first architecture that comes to mind to estimate a re-parametrization of color robust to illumination changes is a odd-layered MLP with three input neurons, three output neurons, and three neurons in its bottleneck layer (plus a bias neuron of course). The 3 neurons of the bottleneck layer would reparameterize color. Or, if color reparameterization was not desired, and only color correction was aimed for, a generic MLP with 3 input neurons and 3 output neurons could be used, and the number of layers and neurons per hidden layer could be optimised. The measured (R,G,B) values corresponding to the same color viewed under two different illuminations can be given as input and ouput of the MLP to train it. However, several illumination changes are possible, and this means that the same entry could correspond to several different outputs. This is impossible for a MLP. Therefore a classical MLP with 4 inputs needs to be used. To reflect the fact that the same input color can correspond to different output colors depending on illumination, a fourth input, the luminance desired in output, is added to the MLP. The architecture of such a MLP is shown in fig. 1 with a bottleneck layer to reparameterize color with 3 parameters. However, in such an architecture, the influence of color and illumination would be mixed in the 3 parameters. The coding of color independently of illumination is not garanteed.

To force the MLP to code color independently of illumination, the architecture of the traditional MLP is modified and a new architecture is proposed to force the network to separate color and luminance. The modified architecture is illustrated by fig. 2. The new MLP includes a compression layer with two neurons (λ, μ). During training, it learns from the inputs (R_i, G_i, B_i) and the desired outputs (R_d, G_d, B_d) to compress color into two parameters (λ, μ). However this is not a trivial compression network. The difference is that there is a fourth input, a context input, which is directly dependent on illumination, and which has its input point in the middle layer of the network (where (λ, μ) are calculated). This context input does not depend on the input (R_i, G_i, B_i) or the actual output $(\hat{R_d}, \hat{G_d}, \hat{B_d})$ of the network, but on the desired luminance $L_d = \frac{R_d+G_d+B_d}{3}$ of the output of the network. With such an input, the network learns to reconstruct the desired output color using directly L_d as an input. Thus it learns to ignore the luminance of the input (R_i, G_i, B_i) and learns to estimate two variables (λ, μ) that are invariant to illumination, and related only to color.

The approach does not require any camera calibration or knowledge about the image.

3.2 Training the modified multi-layer perceptron

As shown in fig. 2, the modified MLP includes 5 layers (this could be generalized to an odd number of layers). The input and output layers have 3 neurons each (plus an additional bias), for RGB inputs and outputs. The middle layer includes 3 neurons (two real and one virtual, excluding bias): their outputs are called λ, μ and L. The second and fourth layers have arbitrary numbers of neurons (typically between 3 and 10 in our experiments). The links between neurons are associated to weights. Neurons have sigmoid activation functions. The network includes biases and moments [Bis96].

A database of images showing the same objects under different illuminations is used to train the modified MLP. The illuminations are typical of indoor environments such as home and office: fluorescent lights, incandescent light bulbs and natural sunlight coming from windows.

A classic MLP training scheme based on backpropagation is applied, with two additional changes due to the structure of the modified MLP. As commonly done with MLPs, a pixel is randomly sampled at each iteration from the training set. Its RGB values before and after an illumination change (from real images)

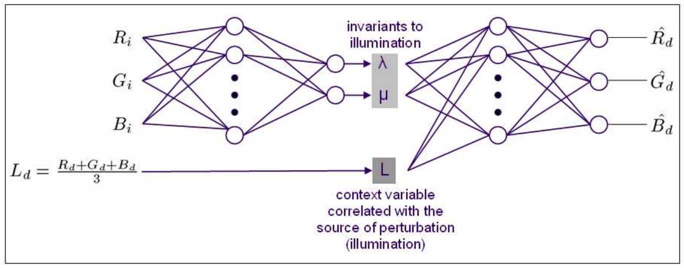

Fig. 2. A modified MLP for color correction and color invariant learning. (R_i, G_i, B_i) is the input color. (R_d, G_d, B_d) is the desired output color, corresponding to the same color seen under a different illumination. $L_d = \frac{R_d+G_d+B_d}{3}$ is the luminance of the desired output and is a direct function of the illumination. λ and μ are the color parameters invariant to illumination that the MLP is trained to estimate. $(\hat{R}_d, \hat{G}_d, \hat{B}_d)$ are the actual outputs of the network. Bias neurons are omitted from this figure.

are used as input (R_i, G_i, B_i) and desired output (R_d, G_d, B_d) to the network. Propagation and back-propagation are then performed, with two modifications (as mentioned above). First, during propagation, the output L of the third neuron of the third layer is forced to the value of the luminance corresponding to the desired output color, e.g. $L_d = (R_d + G_d + B_d)/3$. The idea is that the network is trained to do the best possible reconstruction of the RGB output (R_d, G_d, B_d) from the intermediate variables λ, μ and the imposed luminance L_d. Since L_d is a direct function of the illumination, the estimated λ and μ should be related to characteristics of color that are invariant to illumination. The second modification to training the MLP (compared to classic propagation and back-propagation) is that, during back-propagation, the error on the output L of the third neuron is not back propagated.

3.3 Use of the modified multi-layer perceptron

The trained modified MLP can be used to correct color images. Each image pixel is propagated through the trained network to find the invariants λ and μ. An arbitrary luminance L is imposed on the pixel by forcing the output of the third neuron of the third layer to L. The output of the trained network then gives the corrected color. If a constant luminance L is used for all pixels in the image, an image corrected for shadows and for variations of illumination across the image and between images is obtained. The color correction can be tabulated for fast implementation.

The approach could be easily extended to a greater number of inputs and outputs than 3 or different inputs/outputs than RGB. For instance, YUV or HSV, or redundant characteristics such as RGBYUVLab could be used as inputs and outputs.

4 Image correction results

4.1 Experimental conditions and database

Fig. 3. Examples of images before and after an illumination change from the training database. This database includes examples of illumination changes typical of office and home environments.

The network was trained using 546000 pixels. These were randomly sampled from 91 training images (6000 pixels per image), taken by 2 webcams (Philips ToUCam Pro Camera and Logitech QuickCam Zoom). The training images are of different indoor scenes (and partial views of the outdoors through windows) under varying illuminations, from home and office environments. An example is shown in fig. 3. The variations of illuminations are caused by indoor lighting such as typically found in homes and offices (fluorescent lights and incandescent light bulbs) and natural sunlight (coming from windows). Testing was performed on other images taken by the 2 webcams used for training and by a third webcam, not used for training, a Logitech QuickCam for Notebooks Pro.

In practice, using 8 neurons in the second and fourth layers of the MLP gives good performance. A gain of 1.0 was used, with a momentum factor of 0.01 and a learning rate of 0.001. Pixels that were too dark (luminance ≤ 20) or too bright / saturated (luminance ≥ 250) were not used for training.

4.2 Comparison with a "classical" multi-layer perceptron

	for a classical MLP	for the modified MLP
mean error (in pixel values, the pixel values going from 0 to 255)	10.47	5.54
relative mean error	4.11%	2.17 %

Table 1. Mean error between reconstructed and target images for a "classical" MLP and the modified MLP presented in this article. The mean error was calculated using 748 320x240 test images (not in the training set). The error is averaged over the three color components (R,G,B).

Table 1 shows that the modified MLP (fig. 2) performs better in reconstructing target images than a classic MLP (fig. 1). The reconstruction is done given the expected luminances L_d of the pixels of the desired target image.

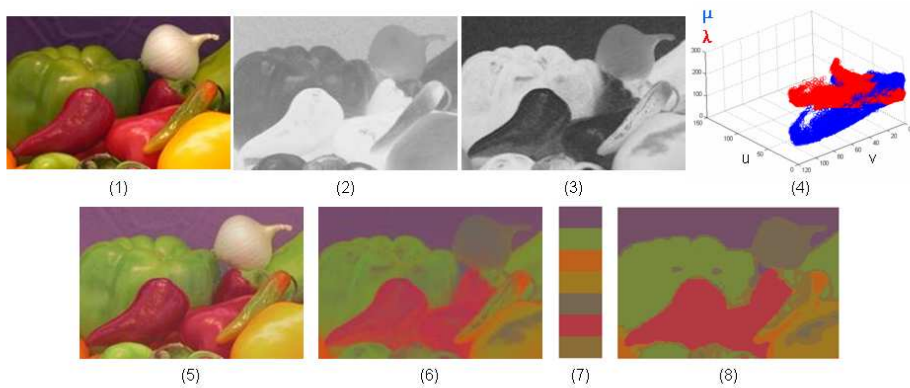

Fig. 4. Example of color correction learnt by the modified MLP. (1) is the original image (unknown illumination). (2) and (3) show the 2 invariants λ and μ estimated by the MLP from the image. (4) is the locus of the invariants in the uv chrominance space of image pixel values. (5) is the corrected image reconstructed by the modified MLP with the pixel luminance inputs set to values proportional to the pixel luminances in the original image (plus a constant). (6) is the corrected image reconstructed by the modified MLP with the pixel luminance inputs set to a constant value for all pixels. (7) shows the 7 color peaks found by mean shift [CRM00] in the corrected image shown in (6). (8) shows the resulting image segmentation.

4.3 Invariant estimation by the modified MLP

Figure 4 shows the two invariants (λ, μ) learnt by the modified MLP and calculated on an image (see part (1) of fig 4) of unknown illumination. The two invariants are seen in parts (2) and (3) of the figure. It can be seen that objects of similar color to the human eye have similar values of λ and μ. In addition, part (4) of fig. 4 shows the locus of the invariant values (λ, μ) in the image as a fonction of the chrominance values (u, v) (from YUV color space) of the image pixels. This plot demonstrates that the locii of the two invariants are not identical, and thus we have two invariants and not only one.

Part (6) of the figure shows the corrected image estimated by the modified MLP from the two invariants (λ, μ) and a constant luminance input over the image. Much of the influence of shading and variations of illumination across the image is removed, apart from specularities (white saturated areas) which are mapped to gray by the network. Indeed areas of similar color to the human eye in the original image (despite shading and illumination) have much more homogeneous color in the corrected image. This can be further seen by performing mean-shift based color segmentation [CRM00] on the corrected image. Seven areas of uniform color are readily identified and segmented (see part (7) and (8) of fig. 4) from the corrected image. They correspond roughly to what is expected by a human observer. This example illustrates that our modified MLP successfully learns a parameterization of color by two parameters that are invariant to illumination.

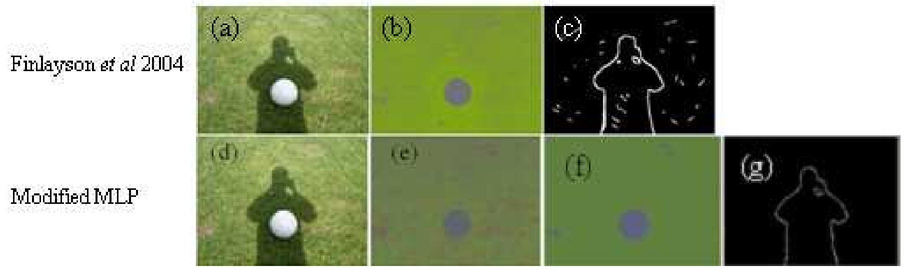

Fig. 5. Comparison of the pixel-wise color correction by the modified MLP presented in this paper and the whole-image color correction method of Finlayson *et al* [FDL04]. Application to shadow detection. Example I. (a) and (d) original image. (b) invariant image obtained using the method of [FDL04]. (c) shadow edges estimated from (b). (e) corrected image estimated using the modified MLP. (f) and (g) results of mean shift color segmentation [CRM00] from (e). (g) shadow edges estimated from (f).

4.4 Performance of a LUT implementation of the trained modified MLP

Color correction by the modified MLP can be tabulated, making it one the fastest possible color correction approaches. Execution time for image correction, based on a Look-Up Table implementation of the modified MLP, is 3.75 ms for an entire 320x240 image, on a Pentium4 3GHz. Such a fast LUT implementation is possible because the approach is pixel-wise.

An HSV correction scheme could be as fast (since it could also be implemented using LUTs), but it would be less performant, as illustrated on a example by fig. 6. A color correction scheme based on [FDL04] would be of equal performance, as illustrated on examples by fig. 5. It could deal with more changes of illumination, since our approach is limited to the type of frequently found indoor lighting the modified MLP was trained for. However, working globally on the image, it could not be implemented as a LUT, and would thus be significantly slower.

4.5 Comparison with other color correction methods from the literature

Figures 6 and 5 compare our color correction approach with an HSV-based correction (HSV being hue-saturation-value) and the color correction scheme of [FDL04] on several examples and for different applications.

Figure 6 compares our approach to HSV-based color correction and applies it to color-based background subtraction. The two first images of the first and second columns of the figure show that the color correction scheme presented in this paper is indeed robust to changes in illumination, since there is much less difference between the images after correction than before. Figure 6 also shows

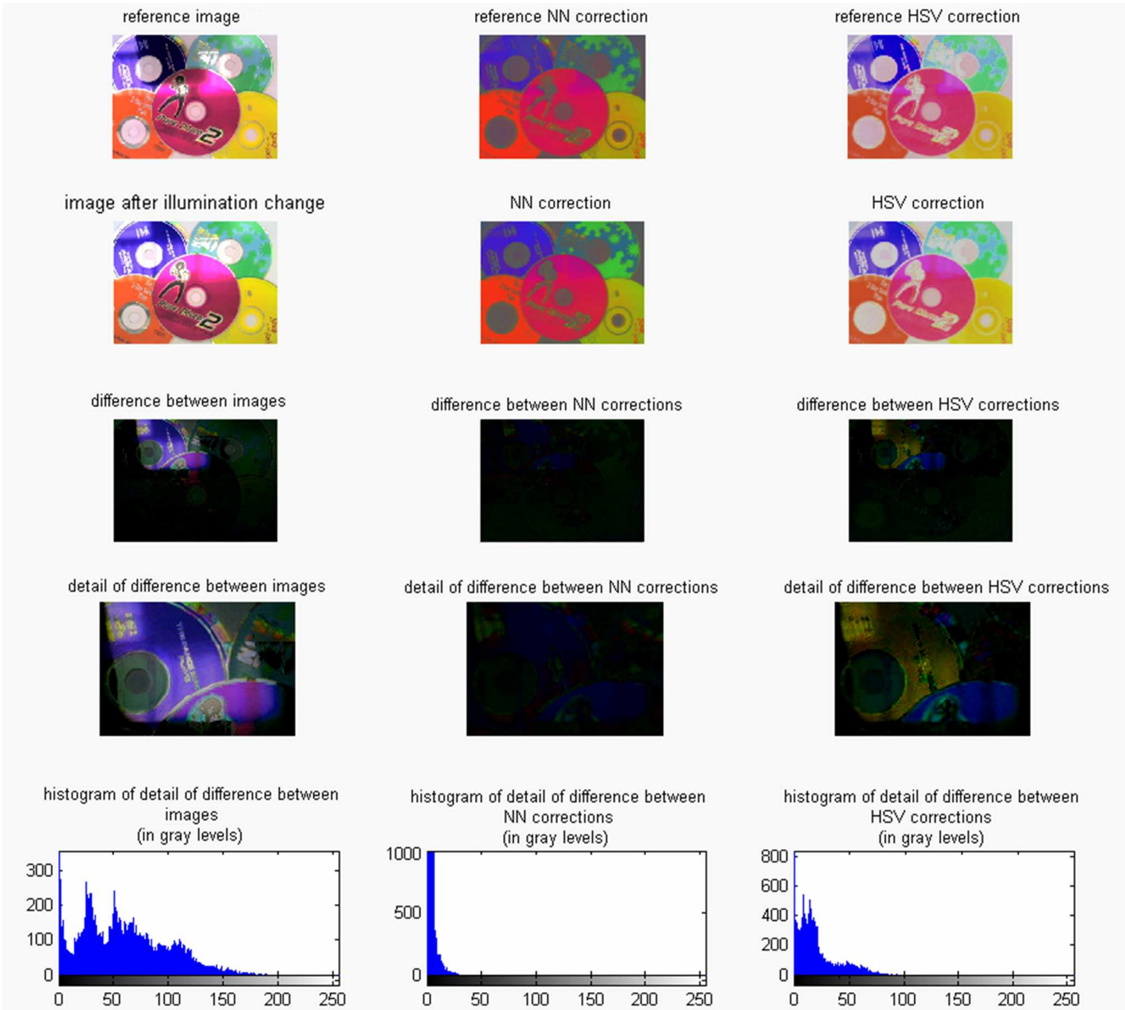

Fig. 6. Comparison of the pixel-wise color correction by the modified MLP presented in this paper and pixel-wise HSV-based color correction, HSV being the well known hue-saturation-value color space.

that the correction performed in this paper compares favorably with an HSV-based color correction (which consists in taking an RGB color to hue-saturation-value space, setting its value/luminance to a constant, then going back to RGB space to get the corrected color).

Figure 5 illustrates that our correction is of similar quality to that of Finlayson et al [FDL04] (briefly described in the introduction of this paper). The application of color correction is the detection of shadow contours (which can be used for shadow removal, as shown in [FDL04]). Even though it might be less robust to large light changes or unusual light changes (such as turning on a blue or red light), our method is faster, being pixel-wise.

5 Conclusion

This paper presents a new neural network-based approach to estimating image color independently of illumination. A modified multi-layer perceptron is trained to estimate two color invariants and an illumination- corrected color for each input color. It is trained for typical indoor home and office lighting (fluorescents and light bulbs) and outdoor natural light, using two webcams. Experiments with light changes and another webcam show that the training seems to have good generalization properties. The approach could be generalized to other applications where one or several invariants of a signal (here color) to a perturbation (here illumination) need to be found. If a database of signals before and after perturbation, and measurements directly correlated to the perturbation are available, then a modified MLP architecture of the type presented here can be used to learn the invariants.

References

[Bis96] C.M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1996.

[CRM00] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proc. of IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR 2000)*, 2000.

[FCB97] B. Funt, V. Cardei, and K. Barnard. Neural network colour constancy and specularly reflecting surfaces. In *Proc. of AIC Color 97, Kyoto, Japan*, 1997.

[FDL04] G.D. Finlayson, M.S. Drew, and C. Lu. Intrinsic images by entropy minimization. In *Proc. 8th European Conf. on Computer Vision (ECCV'04), Prague, pp 582-595*, 2004.

[GG01] I.Y.-H. Gu and V. Gui. Colour image segmentation using adaptive mean shift filters. In *Proc. of . Int. Conference on Image Processing (ICIP'01)*, 2001.

[GS99] T. Gevers and A.W.M. Smeulders. Color based object recognition. *Pattern Recognition*, 1999.

[GW01] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Addison-Wesley Longman Publishing Co., Inc., 2001.

[LK04] Q. Luo and T.M. Khoshgoftaar. Efficient image segmentation by mean shift clustering and MDL-guided region merging. In *Proc. of 16th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI'04)*, 2004.

[LM71] E.H. Land and J.J. McCann. Lightness and retinex theory. *J. Opt. Soc. Am.*, 1971.

[TFA05] M.F. Tappen, W.T. Freeman, and E.H. Adelson. Recovering intrinsic images from a single image. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005.