

Study of Camera Spectral Reflectance Reconstruction Performance using CPU and GPU Artificial Neural Network Modelling

Mihael LAZAR, Dejana JAVORŠEK, Aleš HLADNIK

Abstract: Reconstruction of reflectance spectra from camera RGB values is possible, if characteristics of the illumination source, optics and sensors are known. If not, additional information about these has to be somehow acquired. If alongside with pictures taken, RGB values of some colour patches with known reflectance spectra are obtained under the same illumination conditions, the reflectance reconstruction models can be created based on artificial neural networks (ANN). In Matlab, multilayer feedforward networks can be trained using different algorithms. In our study we hypothesized that the scaled conjugate gradient back propagation (BP) algorithm when executed on Graphics Processing Unit, is very fast, but in terms of convergence and performance, it does not match Levenberg-Marquardt algorithm (LM), which, on the other hand, executes only on CPU and is therefore much more time-consuming. We also presumed that there exists a correlation between the two algorithms and is manifested through a dependency of MSE to the number of hidden layer neurons, and therefore the faster BP algorithm could be used to narrow the search span with the LM algorithm to find the best ANN for reflectance reconstruction. The conducted experiment confirmed speed superiority of the BP algorithm but also confirmed better convergence and accuracy of reflectance reconstruction with the LM algorithm. The correlation of reflectance recovery results with ANNs modelled by both training algorithms was confirmed, and a strong correlation was found between the 3rd order polynomial approximation of the LM and BP algorithm's test performances for both mean and best performance.

Keywords: artificial neural networks; camera spectral reflectance reconstruction; error backpropagation; graphics processing unit; Levenberg-Marquardt algorithm

1 INTRODUCTION

Colour information of an object surface, or of a coloured patch sample, can be described in many device-dependent or device-independent ways. An ideal and unambiguous description should represent colour independently of the capturing device characteristics. A widely used approach involves the specification of colour with three components in CIE 1931 XYZ colour space, where the illuminant spectral power distribution must be known. These tristimulus values are illuminant dependent, though, thus described as being perceived under a specific light source. By defining the colour in a spectral space via surface reflectance, light source dependency is removed. During the last decades, many research efforts have been put into studying the methods for conversion of camera readings into spectral data. Approaches vary and are based on either trichromatic or multispectral imaging systems and numerous methods, from strictly mathematical, of finding a model that conforms to training data and all the constraints, to some other optimisation methods among which artificial neural networks (ANNs) have gained a lot of popularity in recent years [1-8].

Much research has been done in order to find appropriate mathematical methods for estimating the spectral reflectance. Mansouri, Sliwa and Hardeberg proposed an algorithm that makes Principal components analysis (PCA) adaptive in the framework of reflectance recovery from colour camera tri-stimuli values [9]. Bianco developed a local optimisation-based method to recover reflectance spectra from tristimulus values using adaptive estimation with metameric shape correction [10]. Zhang and Xu assumed that three PCs are not sufficient to accurately estimate the spectral reflectance. Thus the spectral space was first divided into 11 subgroups, and the PCs were calculated for individual subgroups. Then the number of PCs was further extended from three to nine through the residual spectral error of the reflectance in each subgroup [11]. Koh, Moroney and Gottwals upon testing their irregular sampling void filling algorithm in the

context of mobile colour sensing application, used in-scene camera captured colour chart with known colourimetric data to perform a dynamic tetrahedral tessellation in RGB space. The selected region of interest of RGB values was afterwards compared with the tessellation, the enclosing tetrahedron was found, and tetrahedral interpolation was performed to transform RGB values into a device-independent CIELAB space [12].

Similarly, several studies were conducted focusing on applying ANNs for reflectance reconstruction. Usui, Nakauchi and Nakano built a wine-glass-type five-layered network with the bottleneck three-neuron hidden middle layer to extract three-dimensional colour attributes, where inputs and outputs of the network were the same spectral data of Munsell colour chips [13] as in our study. Mansouri, Marzani and Gouton proposed cascade of ANNs in the form of auto-associative and hetero-associative memory to build robust reflectance reconstruction systems where inputs were readings from a multispectral camera [5]. Cheung, Westland, Connah and Ripamonti compared the usefulness of two camera characterisation approaches. Both polynomial transforms, and ANNs exhibited similar performance of mapping camera outputs to CIE tristimulus values, but the authors concluded that ANNs can be difficult and time-consuming to train [14].

After the turn of the millennium, the rapid development of computer graphics cards offered a platform for performing massively parallel processing on a graphical processing unit (GPU). ANNs are typical parallel computing applications and as such the power of new GPUs can be efficiently harvested for a remarkable speedup of ANNs and their training algorithms [15-19]. Different types of ANNs are known in terms of architecture and learning algorithms, and many new have been proposed, especially in the field of deep learning applications [19-23]. ANNs are usually programmed and run on conventional computers with installed appropriate simulators, but also on supercomputers, on custom neural circuits which are application-specific integrated circuits

(ASIC) and on some other forms of computer configurations, like neural accelerators and neuromorphic computers [24]. Training of a larger ANN can be time-consuming, and many efforts have been made to reduce the training time.

ANNs can serve many different purposes. Common examples include classification and function approximation. A typical ANN has one input layer, one output layer and one or more hidden layers in-between. Each layer consists of several basic components-neurons. Usually the number of inputs is larger than that of the outputs, as in the case of classification of pictures, where inputs can be image pixel values or image features and outputs are names of individual classes. Similarly, when we are dealing with the functions, inputs can be reflectances at equidistant wave lengths, and outputs are CIE XYZ values that unambiguously describe a specific color.

Alternatively, if the task of an ANN is the reconstruction of spectral values from 3D colour space, the topology of an ANN is reversed, having a small number of inputs and many outputs, each representing one spectral component at a specific wavelength. When input colours are precisely defined in a device independent colour space, as is the case with CIE XYZ colour space or, by knowing the light source characteristics - with CIE Lab colour space -the ANN that computes spectral components from these inputs can be trained efficiently and unambiguously. However, if we only have RGB values retrieved from camera readings, the spectral power distribution of light source is not known, and the characteristics of the camera optics and sensors are not available, an alternative approach has to be adopted. In this case, to reconstruct reflectances, additional information about lighting conditions and the camera has to be somehow implicitly taken into account. One possibility, which we used in our experiment, is the implementation of an array of colour patches with known reflectances that is photographed in the same conditions as the object itself. Known reflectances along with obtained RGB values from the array of patches become new learning set applied to train ANN for these specific conditions, and if training is successful, the reflectance of an unknown object can be reconstructed using newly trained ANN.

2 REFLECTANCE RECONSTRUCTION METHOD

It has been proven that multilayer feedforward networks are, under very general conditions on the hidden unit activation function, universal approximators provided that sufficiently many hidden units are available [25], and as demonstrated by Cybenko [26] that any continuous function can be uniformly approximated by a continuous neural network having a single internal, hidden layer and with an arbitrary continuous sigmoidal nonlinearity. Taking this into account, we modelled the ANN with an input layer receiving inputs from RGB input vector, one hidden layer with a varying number of neurons, and an output layer, spawning an output vector of reflectance values at equidistant 10 nm steps of wavelength from 380 nm to 730 nm. The network map is depicted in Fig. 1, where $x_{k,j}$ are k -th inputs (RGB), $b_j^{(L)}$ and $w_{j,i}^{(L)}$ denote biases and connection weights and $y_{k,j}$ are values of the

output layer neurons-spectral components of k -th reconstructed reflectance; j is neuron index in (L) -th layer and i in previous layer.

Having a learning set of (RGB, reflectance) pairs from the photographed set of patches, the supervised training of the network is done by adjusting connection weights and biases via minimising the network cost function, which in our case is the spectral mean-squared error (Eq. (1)).

$$C = \frac{1}{N_p} \sum_{k=1}^{N_p} \left[\frac{1}{N_\lambda} \sum_{j=1}^{N_\lambda} (r_{k,j} - y_{k,j})^2 \right] \quad (1)$$

The cost function is calculated as an average sum of mean-squared errors (MSE), where N_p is the number of patches, N_λ number of reflectance wavelengths, k is patch index and j wavelength index, $r_{k,j}$ is a j -th spectral component of k -th patch measured reflectance and $y_{k,j}$ j -th spectral component of k -th patch reconstructed reflectance.

For the purpose of function approximation, Matlab supports backpropagation (BP) network training algorithm based on standard gradient descent which can be run on a GPU and is thus very fast but also supports Levenberg-Marquardt algorithm (LM) which is specifically designed to minimise sum-of-square error functions [27]. As stated in the hypotheses below, we assume that LM learning of ANN leads to a better convergence of the cost function and hopefully to a smaller output error and an improved reflectance spectra reconstruction.

LM training updates ANN's parameters by combining standard gradient descent and Newton method (Eq. (2)):

$$w_{j+1} = w_j - (J^T J + \lambda I)^{-1} J^T e \quad (2)$$

Where:

$$(J)_{k,i} = \frac{\partial e_k}{\partial w_i} \quad (3)$$

w_{j+1} is new, w_j is the previous vector of network parameters (weights and biases), e is network output error vector, and J is a Jacobian matrix (Eq. (3)). If the parameter λ is large, the method acts like the standard gradient descent, while if λ is small, LM algorithm performs like the Newton method. Even though Hessian matrix in the algorithm is approximated by Jacobian matrix product ($H \approx J^T J$), the LM algorithm requires a lot of computer memory and is not adapted to run on GPU.

If there exists a correlation between the standard BP training algorithm adapted to run on GPU, and LM, the former can be used to search for a narrower span of the number of neurons in the hidden layer. The pursuit of a better performing ANN model would then continue within this span via a much slower but more efficient LM algorithm. If the correlation exists, it should manifest itself through the MSE function depending on the number of HLN.

Our hypotheses in this study are as follows:

1. Standard BP training algorithm converges slowly, but when running on highly parallel processing GPU, it is computationally fast.

2. LM training algorithm runs on CPU, becomes computationally time-consuming with an increasing number of neurons, but converges much better in comparison to BP, and gives better reflectance reconstruction results.

3. A correlation between BP and LM learning algorithms is manifested through a dependency of MSE to the number of HLN.

4. Running BP training algorithm on GPU can lead to a successful search of a narrower span of the number of HLN, and training ANNs within this HLN span with LM algorithm results in a better performing reflectance reconstruction model.

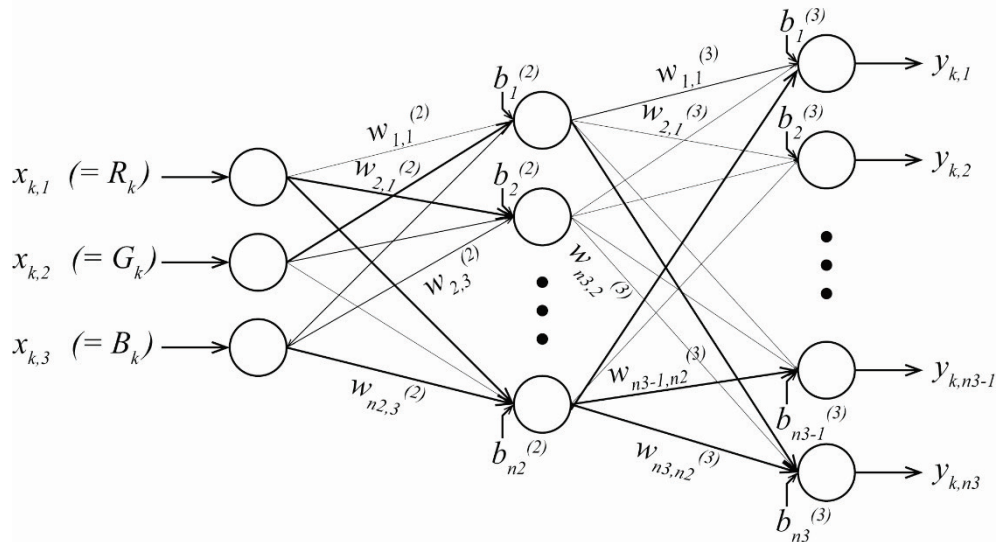


Figure 1 Architecture of an ANN with a single hidden layer and a variable number of hidden neurons, for the reconstruction of reflectances from RGB values

3 MATERIALS AND METHODS

Our research focuses on recovering reflectance spectra from trichromatic camera values by supervised learning of a multilayer perceptron, realised with Matlab Neural Network Toolbox on a general purpose personal computer. Inputs of the ANN model are RGB values of camera readings and outputs are vectors of reflectance spectral components - readouts of the spectrophotometer.

As a source of colourimetric data for the experiment, we selected The Munsell Book of Color Matte Collection, which includes 44 sheets with 1301 patches with different chroma, value and hue values, including many natural colours of soil, skin, foliage and neutral tones. 40 sheets are divided into 2,5 steps Munsell hue circle (2,5; 5; 7,5; 10 for R, YR, Y, GY, G, BG, B, PB, P and RP), and four sheets contain neutral colours - neutral and lightly hued greys. Each patch was measured by i1Pro 2 spectrophotometer (X-Rite) at five points (near corners and at the patch centre). Reflectance vectors with 107 components for wavelengths from 376,66 to 730 nm with the step size of 3,33 nm were obtained with a maximum standard deviation for five measurements of less than 0,4%. The average values for the five patch measurements were computed, and reflectances were downsampled to 36 components from 380 nm to 730 nm.

The sheets were photographed in the photo studio, under the constant lighting conditions with Nikon D600 camera and constant manual settings (Fig. 2). Lightsource's spectral power distribution was measured with a spectrophotometer at the position of the sheet with colour samples, and the correlated colour temperature (CCT) of 3019°K was obtained. To ensure constant conditions and to process all the RAW images equally,

images were normalised using Adobe Lightroom software to the CCT, tint was balanced to 0, and chromatic aberration, even though unnoticeable, was corrected for the profile "Nikon AF-S NIKKOR 50 mm f/1,4G". Median RGB values for each patch were acquired at the inner 50% square surface of the patch.



Figure 2 Photo studio setting

The complete set consisting of (RGB, reflectance) pairs for all 1301 Munsell Matte patches, with three values for each RGB and 36 values for each reflectance was assembled, to feed into an ANN supervised learning algorithm. Before training, independent samples were separated from the complete set to calculate additional measures of reflectance reconstruction performance. The remaining set formed the learning set, which was randomly split into training, validation and testing sets in a 70:15:15 ratio at the beginning of each ANN model training.

Due to the nature of ANN training algorithms, which tend to find local, instead of global minima of the cost function, the search of optimal ANN parameters was repeated 41 times for each model, with sizes of the hidden layer changing by the step of 1 from 3 to 48 neurons. In our previous experiments with 21 learning repetitions it was observed that by increasing the number of HLN, the

probability of finding better ANN parameters does not improve noticeably. In the presented study, however, we wanted to find the best ANN models with even greater certainty, so the number of repetitions was doubled, i.e. to 42. The odd number (41) was chosen due to the calculations of some additional statistics not mentioned in the article. The calculations were performed for five different learning set sizes (Tab. 1) and both learning algorithms (BP and LM).

Table 1 Size of learning sets

of the complete set / %	RGB-reflectance learning set pairs	independent samples
90	1171	130
50	650	651
30	390	911
20	260	1041
15	195	1106

4 RESULTS

To reiterate, for each of the 46 ANN models with 3 to 48 HLN, and for each of the five learning sets, the learning with BP and LM training algorithm was repeated for 41 times. The data of the average and the best results were collected, and the results were visualized and compared. In total, more than 200 hours of 4-core parallel processing CPU time and almost 12 hours of GPU time were spent. Parallel CPU computing is supported by Matlab Parallel Computing Toolbox. ANN modelling on all CPU cores was activated by useParallel training option.

The average time for one ANN model learning with both compared algorithms, and with the largest learning set, is visualised using logarithmic y-axis in Fig. 3 along with LM/BP learning time ratio. The speed of convergence expressed as the "number of epochs" how many times the output error had to back-propagate through the network to adjust its independent parameters in order to minimize the cost function-is visualised in Fig. 4 along with the BP/LM number of epochs ratio.

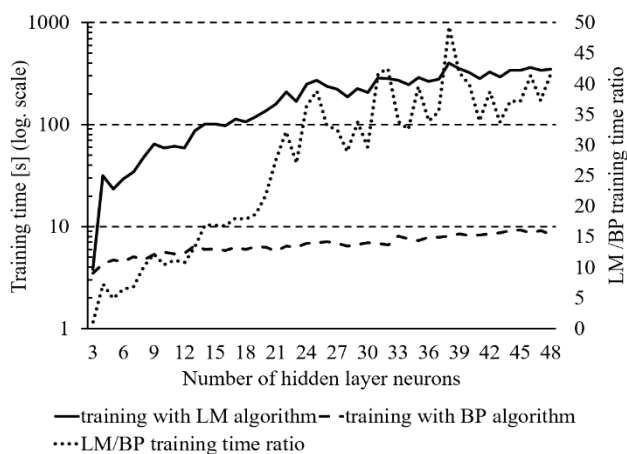


Figure 3 BP and LM algorithm average learning times for the largest learning set, with ratio curve

After the training of every single ANN model with the particular number of HLN is finished, performance is calculated on the test set of samples. After 41 training repetitions of ANN with the same HLN size, the average test performance is calculated and the best result at the minimum of test performance is found. Average and

minimum test performance-MSE values-according to the number of HLN is shown in Fig. 5 and Fig. 6 along with the BP/LM performance ratio and the performance polynomial trend lines of the 3rd order.

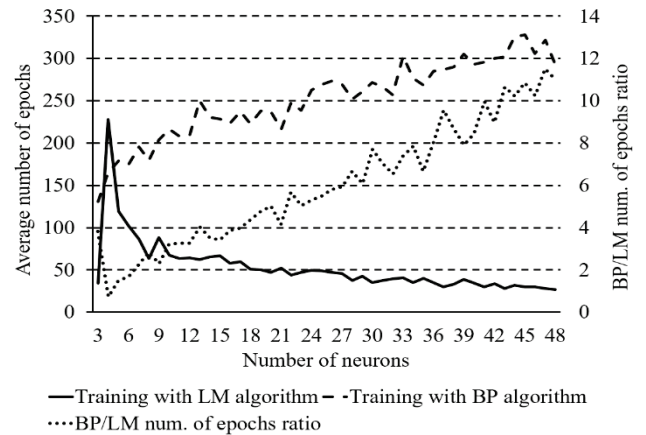


Figure 4 BP and LM algorithm average number of epochs to train the ANN model, for the largest learning set, with ratio curve

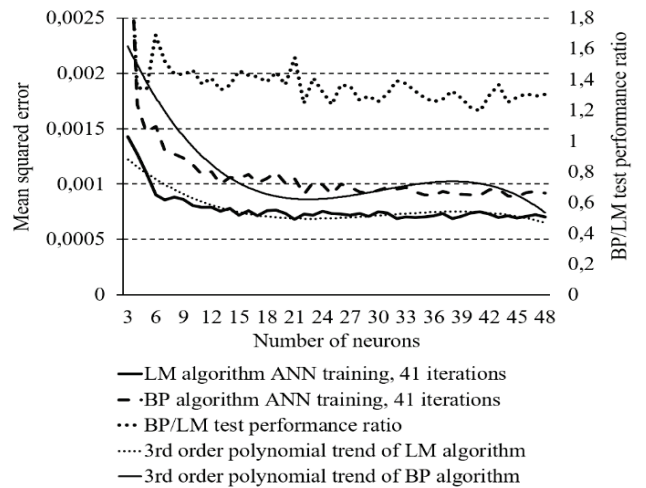


Figure 5 BP and LM algorithm Mean Test Set Performance of 41 repetitions of ANN training for each number of HLN

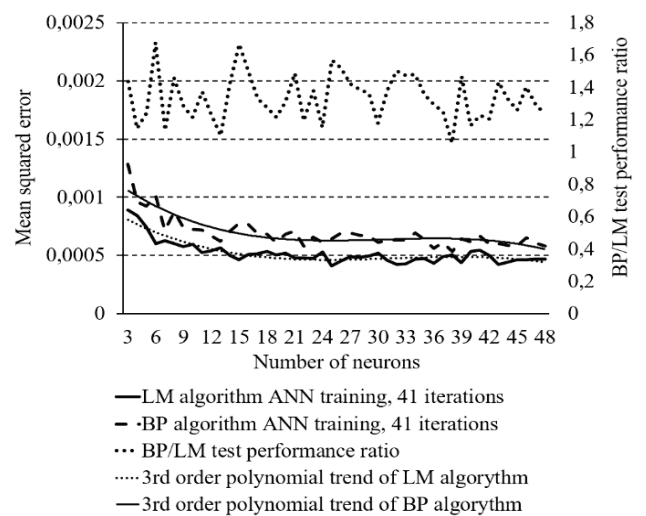


Figure 6 BP and LM algorithm Best (Minimum) Test Set Performance of 41 repetitions of ANN training for each number of HLN

Correlation between LM and BP algorithm performances was calculated for each learning set along

with the p-value at the significance level 0,05. Correlations were obtained for mean performance, min performance and the 3rd order polynomial approximation of mean and min performance (Tab. 2). All correlations were found to be statistically significant ($p < 0,05$).

Minimum of LM and BP algorithm average performance and best performance, and of their 3rd order polynomial approximation was found for each learning set, along with the number of HLN at these points of best ANN models for reflectance reconstruction (Tab. 3). Differences between the number of HLN at these points were calculated between "LM algorithm average performance" and "LM algorithm average performance 3rd order polynomial approximation", "BP algorithm average performance" and "BP algorithm average performance 3rd order polynomial approximation". Also, the same differences were calculated for the best performances. Mean values and standard deviations of obtained differences were calculated (Tab. 4).

Based on the observed results, we proposed and tested a new procedure to search for the best ANN model. Reflectance spectra of independent samples, which had been excluded from each of the five learning sets-see Tab. 1 were reconstructed with the best performance ANN models found with the LM algorithm using the proposed procedure, and with the BP algorithm. $L \times a \times b \times$ values were calculated from the measured and reconstructed spectra for the illuminants A, D50, D65 and F2. Colour differences were calculated and classified. As ΔE_{00} colour difference formula was used, classification into seven quality groups, as proposed by Yang, Ming and Yu [28], was performed. Average relative differences between LM and BP algorithm performance, normalized to BP performance were calculated through all learning sets, to assess the efficiency of the LM algorithm. Classification of colour differences and relative differences between LM and BP algorithms are presented in Tab. 5.

Table 2 Correlation between LM and BP algorithm average and best (min.) performance of 41 repetitions of ANN training for each number of HLN

Learning set size	Mean Test Performance (TP) Correlation Coefficient (CC)	3 rd order Polynomial Approximation (PA) of Mean TP CC	Min. TP CC	3 rd order PA of Min. TP CC
1171	0,8934	0,9998	0,8248	0,9945
650	0,9222	0,9896	0,7193	0,9143
390	0,9038	0,9396	0,7390	0,9450
260	0,6836	0,7012	0,6126	0,8346
195	0,4156	0,4090	0,1584	0,4609

Table 3 Minimum values of mean and best (min.) LM and BP performance functions, with HLN positions of the best ANN models

Learning set size	1		2		3		4		5		6		7		8	
	Min. of "mean LM (of 41 models)"	HLN No.1	Local min. of "3 rd o. PA of meanLM"	HLN No.2	Min. of "mean BP (of 41 models)"	HLN No.3	Local min. of "3 rd o. PA of meanBP"	HLN No.4	Min. of "min LM (of 41 models)"	HLN No.5	Local min. of "3 rd o. PA of min LM"	HLN No.6	Min. of "min BP (of 41 models)"	HLN No.7	Local min. of "3 rd o. PA of min BP"	HLN No.8
1171	0,000682	21	0,000687	23	0,000887	45	0,000862	22	0,000410	25	0,000463	25	0,000534	38	0,000628	25
650	0,000760	24	0,000778	22	0,000992	46	0,000975	23	0,000471	21	0,000480	22	0,000555	38	0,000670	26
390	0,000736	27	0,000725	21	0,000976	27	0,000885	22	0,000425	21	0,000471	21	0,000527	36	0,000606	24
260	0,000853	24	0,000867	19	0,001160	36	0,001075	22	0,000436	20	0,000523	21	0,000553	20	0,000655	23
195	0,000953	23	0,001014	18	0,001430	23	0,001343	22	0,000467	23	0,000548	18	0,000503	18	0,000684	26

Abbreviations: HLN No.= Number of neurons in the hidden layer; 3rd o. PA = Polynomial approximation of the 3rd order

Table 4 Differences between the number of HLN at positions of the best ANN models, along with mean and standard deviation of differences

Learning set size	diff(HLN No.1, HLN No.2)	diff(HLN No.1, HLN No.3)	diff(HLN No.1, HLN No.4)	diff(HLN No.5, HLN No.6)	diff(HLN No.5, HLN No.7)	diff(HLN No.5, HLN No.8)
1171	-2	-24	-1	0	-13	0
650	2	-22	1	-1	-17	-5
390	6	0	5	0	-15	-3
260	5	-12	2	-1	0	-3
195	5	0	1	5	5	-3
Mean of diff.	3,20	-11,60	1,60	0,60	-8,00	-2,80
Standard dev. of diff.	2,93	10,31	1,96	2,24	8,81	1,60

Peak signal to noise ratio (PSNR) and goodness of fit coefficient (GoFC) for reconstructed reflectances of independent samples were calculated (Eq. (4) and Eq. (5)) and classified into three classes based on PSNR value span and into four classes based on GoFC value span [29, 30]. Relative differences between LM and BP algorithm were calculated and are presented in Tab. 6.

$$PSNR(r, y) = 20 \log_{10} \left(1 / \left(\sqrt{\frac{1}{n} \sum_{j=1}^n (r_j - y_j)^2} \right) \right) \quad (4)$$

$$GoFC = \sum_{j=1}^n r_j y_j / \left(\sqrt{\sum_{j=1}^n r_j^2} \sqrt{\sum_{j=1}^n y_j^2} \right) \quad (5)$$

where r is the measured, and y is the reconstructed n -dimensional reflectance vector.

Table 5 Percentage of ΔE colour difference between measured and reconstructed reflectances of independent samples reconstructed with LM and BP algorithm and classified into seven quality categories

Percentage of samples in learning set (the rest are independent samples)	Illuminant	Hardly / %		Slight / %		Noticeable / %		Appreciable / %		Much / %		Very much / %		Strongly / %	
		$0 \leq dE$		$0,5 \leq dE$		$1,5 \leq dE$		$3 \leq dE$		$6 \leq dE$		$12 \leq dE$		$24 \leq dE$	
		$< 0,5$		$< 1,5$		< 3		< 6		< 12		< 24			
		LM	BP	LM	BP	LM	BP	LM	BP	LM	BP	LM	BP	LM	BP
90% (130 independent samples)	A	10,0	6,9	65,4	43,1	20,8	36,9	3,1	10,8	0,8	2,3	0,0	0,0	0,0	0,0
	D50	6,9	9,2	53,8	35,4	33,8	44,6	4,6	7,7	0,8	3,1	0,0	0,0	0,0	0,0
	D65	6,2	8,5	48,5	32,3	38,5	48,5	6,2	7,7	0,8	3,1	0,0	0,0	0,0	0,0
	F2	7,7	7,7	55,4	44,6	31,5	33,1	5,4	11,5	0,0	3,1	0,0	0,0	0,0	0,0
50% (651 independent samples)	A	10,0	5,1	65,1	44,4	21,0	32,7	3,1	14,6	0,8	3,1	0,0	0,2	0,0	0,0
	D50	7,1	4,5	56,4	37,3	30,0	36,7	5,4	17,2	1,2	3,5	0,0	0,8	0,0	0,0
	D65	6,9	3,7	52,4	35,3	31,2	36,4	8,4	18,9	1,1	4,6	0,0	1,1	0,0	0,0
	F2	6,8	4,5	59,8	41,3	27,8	33,8	4,9	17,7	0,8	2,5	0,0	0,3	0,0	0,0
30% (911 independent samples)	A	9,1	6,5	62,2	52,1	24,7	30,0	3,4	7,4	0,5	4,1	0,0	0,0	0,0	0,0
	D50	6,9	4,8	53,9	44,6	31,9	37,0	6,0	11,6	1,2	2,0	0,0	0,0	0,0	0,0
	D65	6,4	4,5	51,4	40,8	33,7	41,1	7,7	11,4	0,9	2,2	0,0	0,0	0,0	0,0
	F2	7,8	5,4	52,7	46,0	32,6	36,4	6,3	10,1	0,7	2,1	0,0	0,0	0,0	0,0
20% (1041 independent samples)	A	8,2	5,0	56,0	50,2	25,4	36,4	8,5	7,0	2,0	1,2	0,0	0,2	0,0	0,0
	D50	7,1	3,1	48,5	40,8	31,8	42,5	9,5	11,7	3,1	1,6	0,0	0,3	0,0	0,0
	D65	6,7	2,3	45,8	38,9	33,4	41,9	10,6	13,9	3,4	2,7	0,1	0,3	0,0	0,0
	F2	6,6	4,7	51,0	44,9	30,7	38,0	9,7	11,1	1,9	1,0	0,0	0,2	0,0	0,1
15% (1106 independent samples)	A	5,1	6,3	53,3	43,0	34,1	37,9	6,6	11,3	1,0	1,4	0,0	0,0	0,0	0,0
	D50	3,9	4,4	46,7	40,1	40,2	40,7	8,4	12,8	0,8	2,0	0,0	0,0	0,0	0,0
	D65	3,3	3,2	44,0	39,2	41,5	42,5	9,9	12,2	1,1	2,9	0,1	0,0	0,0	0,0
	F2	5,2	3,5	51,4	40,3	35,6	42,4	7,1	12,2	0,8	1,5	0,0	0,0	0,0	0,0
Average difference (LM-BP)/BP / %	A	45,1		30,6		-27,5		-45,0		-36,9		-100,0			
	D50	39,2		31,9		-16,5		-42,0		-30,0		-100,0			
	D65	59,8		30,8		-15,1		-30,1		-49,8		-83,3			
	F2	36,7		24,9		-13,6		-43,7		-36,8		-100,0		-100,0	

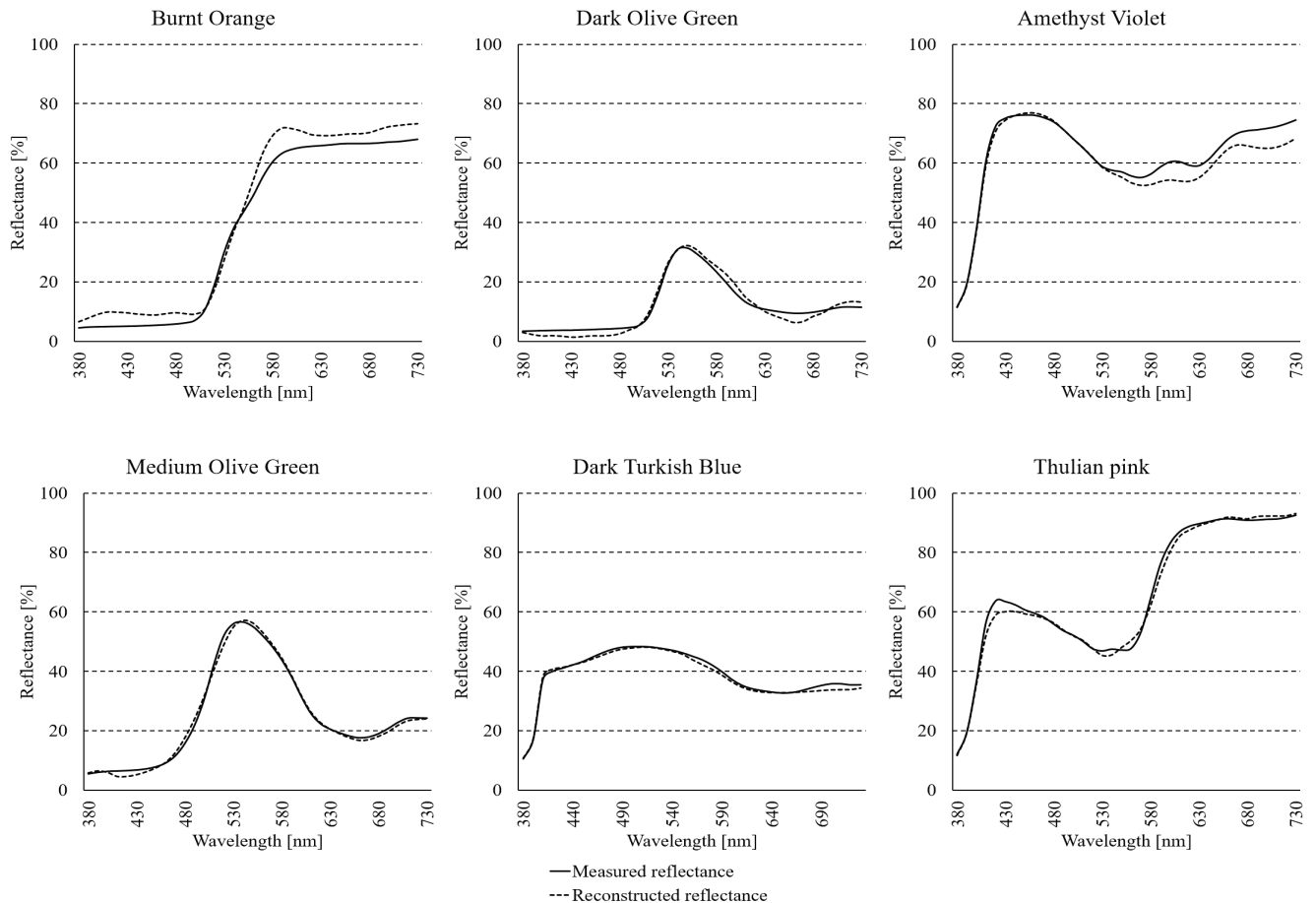


Figure 7 Examples of some bad and good reflectance reconstruction

Table 6 Percentage of reconstructed reflectances of independent samples, classified into three classes

Percentage of samples in learning set	PSNR						GoFC							
	Poor / %		Accurate / %		Good / %		Poor / %		Accurate / %		Good / %		Excellent / %	
	PSNR < 34 dB		34 dB ≤ PSNR < 40 dB		PSNR ≥ 40 dB		GoFC < 0,995		0,995 ≤ GoFC < 0,999		0,999 ≤ GoFC < 0,9999		GoFC > 0,9999	
	LM	BP	LM	BP	LM	BP	LM	BP	LM	BP	LM	BP	LM	BP
90%	39,2	47,7	36,2	34,6	24,6	17,7	13,1	17,7	35,4	48,5	47,7	33,1	3,8	0,8
50%	34,7	44,9	40,7	42,1	24,6	13,1	16,0	25,3	34,7	35,2	45,6	36,6	3,7	2,9
30%	34,4	40,9	41,9	41,1	23,7	18,0	15,1	21,0	37,3	36,4	43,7	40,8	3,8	1,8
20%	40,3	45,9	41,9	42,1	17,8	12,0	18,7	24,2	36,0	39,5	43,4	35,6	1,8	0,7
15%	43,4	45,7	40,8	40,3	15,8	14,0	20,9	19,2	38,2	46,6	38,9	33,4	2,0	0,9
Average difference (LM-BP)/BP / %	-14,7		0,8		44,0		-20,9		-10,5		22,9		167,3	

Examples of some bad and good reflectance reconstructions are shown in Fig. 7. Burnt Orange, Dark Olive Green and Amethyst Violet have the "appreciable" perceptible colour difference, and either "poor" MSE or "poor" GoFC, Medium Olive Green, Dark Turkish Blue and Thulian Pink have "hardly" or "slight" perceptible colour difference, good GoFC and either "accurate" or "good" MSE. Shown reconstructions are made by ANN trained upon learning set containing 30% of patches. Reconstructed samples belong to the separate independent test set.

5 DISCUSSION

Comparison of the convergence of BP and LM training algorithms (Fig. 4) indicates a linear growth of BP/LM number of epochs ratio with respect to the number of HLN. As soon as the number of neurons in the hidden layer is higher than five, the LM algorithm converges faster than BP. At ten HLN it is 3-times, at 20 HLN 5-times and at 30 HLN almost 8-times better. For example, modelling an ANN with 20 neurons with LM algorithm takes 50 epochs but with BP almost 250 epochs. This clearly shows the superiority of the LM algorithm in terms of convergence efficiency. Why then use the BP algorithm at all?

When looking from a different point of view, i.e. when focusing on the calculation speed (Fig. 3), it can be stated that when comparing the speed of the LM algorithm running in parallel on quad-core CPU versus the BP training algorithm running on GPU, the latter is found to be faster right from the start. LM/BP average learning time ratio shows that at five HLN, BP algorithm is five-, at ten about ten, at twenty about twenty times faster and so on. Modelling an ANN with 20 neurons with LM algorithm takes 150 seconds, while with BP algorithm only 7 seconds. Such a time saving evidently favours BP training algorithm.

Finally, we can examine the accuracy of reflectance reconstruction (Fig. 5 and Fig. 6), i.e. the performance, calculated as the mean squared error between the reconstructed and the measured reflectances. When comparing the average performance of the ANN models trained with LM and BP algorithms, the ratio shows about 30% better LM algorithm performance. When comparing these with the best performing ANN models, we found the best performance to be by approximately 30% better than the average performance, at the same BP/LM performance ratio.

Result of a reduction in the learning set size (Tab. 1) is that the BP algorithm running on GPU still remains much faster in terms of an ANN modelling time, LM algorithm still converges faster while the ratio of BP/LM performance remains the same. The first two hypotheses have therefore been confirmed.

When monitoring the shape of the mean- and the best performance curves for the test sets, the fluctuation of values is evident and results from the randomness of initial values of the ANN free parameters (weights and biases) so that every time a different local minimum of the cost function is found. These fluctuations remain even after a larger number of ANN modelling repetitions. Linear correlations between LM and BP algorithm average- and best performances were found to be high and statistically significant, with small p-values, and thus the third hypothesis was also confirmed. In an attempt to effectively smooth the performance curves, different methods were tested. Averaging of neighbouring values and the polynomial approximations were promising approaches, and a high correlation was found between the 3rd order polynomial approximations of LM and BP training algorithm mean- and best performance curves. Test performance correlation coefficients for mean and best test LM and BP values are, understandably, lower than those for smoother 3rd order polynomial approximations. It was also found that the fluctuation of the best curves of LM and BP algorithms becomes more pronounced at a certain point (at 20% of samples or less), and the correlation coefficient rapidly decreases, although high correlation coefficient values for the polynomial approximation of the third order still indicate a strong correlation.

Test performance mean- and best (i.e. minimum) curves as a function of the number of HLN at some point reach their minimum values, and by increasing the number of neurons, performance does not get any better. This minimum value is the point we are searching for. This point, we propose, indicates the optimal number of HLN to get the best reflectance reconstruction results. To narrow the search area for finding this point at the test performance curve for LM algorithm, differences were calculated between the number of HLN at the positions of the best ANN models. We compared the number of HLN at the minimum of the LM algorithm average performance curve with its 3rd order polynomial approximation, with the BP algorithm average performance curve and with the 3rd order polynomial approximation of the latter. In the same way, differences were calculated between the number of HLN at the minimum of the LM algorithm best performance

curve, its 3rd order polynomial approximation, the BP algorithm best performance curve and the 3rd order polynomial approximation of the latter. Mean, and standard deviation were calculated for the obtained differences for all five learning set sizes. It can be seen that the mean and standard deviation for differences between LM algorithm performance and 3rd order approximation of BP algorithm performance are far smaller than between LM algorithm performance and BP algorithm performance, for both average and best performances.

Based on these findings, we propose the following procedure to search for the best ANN model at any other specific size of the learning set:

1. Run the ANN with the BP algorithm on GPU with varying the HLN number in a broader span and repeat the calculations many times at each HLN number, to increase the possibility of finding a satisfactory ANN model.
2. Calculate the 3rd order polynomial approximation of the BP algorithm (3rd o. PA BP) performance curve in dependence of HLN number, and find the HLN number at 3rd o. PA BP minimum value (THLN3OPABPmin) that precedes the curve inflection point.
3. Run the ANN with the LM algorithm on CPU varying the HLN number in a narrower span of ± 5 HLN (about 3 times the standard deviation) from THLN3OPABPmin, and repeat the calculations many times at each HLN number, again to increase the possibility of finding satisfactory ANN model.

After executing this procedure, we found a good LM algorithm ANN model in less than 30% of time necessary for scanning through the complete span from the smallest to the largest number of HLN (where the finding of the best ANN model is expected) with LM algorithm alone.

Results of reflectance reconstruction with the best performance ANN model found with the LM algorithm using the proposed procedure, and with the best model found with BP algorithm alone, are presented with differences between the measured and the reconstructed colours in Tab. 5, taking into account different illuminants. It can be seen that LM algorithm puts, compared to BP algorithm, about 40% more reconstructed colours into the best category, where the colour difference is hardly perceptible. About 30% more samples fall into the second best category, with a slight perceptible colour difference, while in the third category, where the colour difference is noticeable, LM puts about 15% less reconstructed colours than BP. Samples classified into lower categories with LM algorithm reach in most cases less than 10% of all. In Tab. 6 results are put into three categories based on the logarithm of mean squared error (PSNR) and into four based on the goodness of fit. LM algorithm relative to BP places more than 40% reconstructed reflectances into the highest category (Good), and almost 15% less into the low one (Poor), while the middle one (Accurate) is almost equally populated. In terms of GoFC, LM algorithm puts (relatively to BP) about 25% more reconstructions into a category where GoFC is higher than 0,999 and about than 20% less into the lowest category with GoFC under 0,995.

6 CONCLUSIONS AND FUTURE WORK

Executing the proposed procedure, effective ANN models with an optimal number of neurons in the hidden

layer can be found in less than 30% of the time necessary to find such models using only the LM algorithm. If the search step is wider, when instead of testing performance for every number of HLN, search advances in a step of 2 or 3, time could be further reduced.

Observation of results, especially comparison of measured and reconstructed reflectance values (Fig. 7), reveals some discrepancies, where the reconstructed reflectance values fall into the category of "appreciable" perceptible colour difference, but these only form less than 10% of all reconstructed reflectances and by increasing the learning set, this figure falls below 5%.

We assume that by increasing the number of hidden layers, the results could get even better and, presumably, by capturing RGB values with two or more different cameras, performance could increase. We will test both assumptions in our following experiments.

7 REFERENCES

- [1] Cheung, V., Westland, S., Li, C., Hardeberg, J., & Connah, D. (2005). Characterization of trichromatic color cameras by using a new multispectral imaging technique. *JOSA A*, 22, 1231-1240. <https://doi.org/10.1364/JOSAA.22.001231>
- [2] Hardeberg, J. Y. (2001). Acquisition and reproduction of color images: colorimetric and multispectral approaches. Universal-Publishers.
- [3] Connah, D., Westland, S., & Thomson, M. G. (2001). Recovering spectral information using digital camera systems. *Coloration technology*, 117, 309-312. <https://doi.org/10.1111/j.1478-4408.2001.tb00080.x>
- [4] Shen, H.-L. & Xin, J. H. (2004). Spectral characterization of a color scanner by adaptive estimation. *JOSA A*, 21, 1125-1130. <https://doi.org/10.1364/JOSAA.21.001125>
- [5] Mansouri, A., Marzani, F. S., & Gouton, P. (2005). Neural networks in two cascade algorithms for spectral reflectance reconstruction. Image Processing, 2005. *IEEE International Conference on*, 2, II-718. <https://doi.org/10.1109/ICIP.2005.1530156>
- [6] Dupont, D. (2002). Study of the reconstruction of reflectance curves based on tristimulus values: comparison of methods of optimization. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, 27(2), 88-99. <https://doi.org/10.1002/col.10031>
- [7] Lazar, M., Hladnik, A., Javoršek, D., & Jerman, T. (2013). Prediction of RGB camera values by means of artificial neural networks. *Scientific Papers of the University of Pardubice. Series A, Faculty of Chemical Technology*, 19, 185-194.
- [8] Lazar, M., Učakar, A., Javoršek, D., & Hladnik, A. (2016). Artificial neural networks as a tool for spectral reflectance estimation of RGB color patches. *International Symposium on Graphic Engineering and Design GRID, Proceedings*, 289-296.
- [9] Mansouri, A., Sliwa, T., Hardeberg, J. Y., & Voisin, Y. (2008). An adaptive-pca algorithm for reflectance estimation from color images. *19th International Conference on Pattern Recognition*, 1-4. <https://doi.org/10.1109/ICPR.2008.4761120>
- [10] Bianco, S. (2010). Reflectance spectra recovery from tristimulus values by adaptive estimation with metameric shape correction. *JOSA A*, 27, 1868-1877. <https://doi.org/10.1364/JOSAA.27.001868>
- [11] Zhang, X. & Xu, H. (2008). Reconstructing spectral reflectance by dividing spectral space and extending the

- principal components in principal component analysis. *JOSA A*, 25, 371-378. <https://doi.org/10.1364/JOSAA.25.000371>
- [12] Cheung, V., Westland, S., Li, C., Hardeberg, J., & Connah, D. (2005). Characterization of trichromatic color cameras by using a new multispectral imaging technique. *JOSA A*, 22, 1231-1240. <https://doi.org/10.1364/JOSAA.22.001231>
- [13] Usui, S., Nakauchi, S., & Nakano, M. (1992). Reconstruction of Munsell color space by a five-layer neural network. *JOSA A*, 9, 516-520. <https://doi.org/10.1364/JOSAA.9.000516>
- [14] Cheung, V., Westland, S., Connah, D., & Ripamonti, C. (2004). A comparative study of the characterisation of colour cameras by means of neural networks and polynomial transforms. *Coloration technology*, 120, 19-25. <https://doi.org/10.1111/j.1478-4408.2004.tb00201.x>
- [15] Luo, Z., Liu, H., & Wu, X. (2005). Artificial neural network computation on graphic process unit. *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, 1, 622-626.
- [16] Oh, K.-S. & Jung, K. (2004). GPU implementation of neural networks. *Pattern Recognition*, 37, 1311-1314. <https://doi.org/10.1016/j.patcog.2004.01.013>
- [17] Jang, H., Park, A., & Jung, K. (2008). Neural network implementation using cuda and openmp. *Digital Image Computing: Techniques and Applications*, 155-161. <https://doi.org/10.1109/DICTA.2008.82>
- [18] Scanzio, S., Cumani, S., Gemello, R., Mana, F., & Laface, P. (2010). Parallel implementation of artificial neural network training for speech recognition. *Pattern Recognition Letters*, 31, 1302-1309. <https://doi.org/10.1016/j.patrec.2010.02.003>
- [19] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- [20] Aleksander, I. & Morton, H. (1991). *An introduction to neural computing, 1st edn* Int. Th. Comp. Press, London.
- [21] Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.
- [22] Kohonen, T. (2012). Self-organization and associative memory. *Springer Science & Business Media*, 8.
- [23] Almási, A.-D., Woźniak, S., Cristea, V., Leblebici, Y., & Engbersen, T. (2016). Review of advances in neural networks: Neural design technology stack. *Neurocomputing*, 174, 31-41. <https://doi.org/10.1016/j.neucom.2015.02.092>
- [24] Prieto, A., Prieto, B., Ortigosa, E. M., Ros, E., Pelayo, F., Ortega, J., & Rojas, I. (2016). Neural networks: An overview of early research, current frameworks and new challenges. *Neurocomputing*, 214, 242-268. <https://doi.org/10.1016/j.neucom.2016.06.014>
- [25] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4, 251-257. [https://doi.org/10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T)
- [26] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2, 303-314. <https://doi.org/10.1007/BF02551274>
- [27] Sapna, S., Tamilarasi, A., Kumar, M. P., & Others. (2012). Backpropagation learning algorithm based on Levenberg Marquardt Algorithm. *Comp Sci Inform Technol (CS and IT)*, 2, 393-398. <https://doi.org/10.5121/csit.2012.2438>
- [28] Yang, Y., Ming, J., & Yu, N. (2012). Color image quality assessment based on CIEDE2000. *Advances in Multimedia*, 2012, 11. <https://doi.org/10.1155/2012/273723>
- [29] Hernández-Andrés, J., Romero, J., & Lee, R. L. (2001). Colorimetric and spectroradiometric characteristics of narrow-field-of-view clear skylight in Granada, Spain. *JOSA A*, 18, 412-420. <https://doi.org/10.1364/JOSAA.18.000412>
- [30] Lehtonen, J., Parkkinen, J., Jaaskelainen, T., & Kamshilin, A. (2009). Principal component and sampling analysis of color spectra. *Optical review*, 16, 81-90. <https://doi.org/10.1007/s10043-009-0015-6>

Contact information:

Mihael LAZAR, B.Sc.E.E.,
(Corresponding author)
University of Ljubljana, Slovenia,
Faculty of Natural Sciences and Engineering, Department of Textiles, Graphic Arts and Design,
Snežniška ulica 5, SI-1000 Ljubljana
E-mail: mihael.lazar@ntf.uni-lj.si

Dejana JAVORŠEK, Assoc. Prof. PhD.,
University of Ljubljana, Slovenia,
Faculty of Natural Sciences and Engineering, Department of Textiles, Graphic Arts and Design,
Snežniška ulica 5, SI-1000 Ljubljana
E-mail: dejana.javorsek@ntf.uni-lj.si

Aleš HLADNIK, Assoc. Prof. PhD.,
University of Ljubljana, Slovenia,
Faculty of Natural Sciences and Engineering, Department of Textiles, Graphic Arts and Design,
Snežniška ulica 5, SI-1000 Ljubljana
E-mail: ales.hladnik@ntf.uni-lj.si