

# Colour Constancy using Grey Edge Framework and Image Component analysis

**Martin Šavc and Božidar Potočnik**

Faculty of Electrical Engineering and Computer Science, University of Maribor  
2000 Maribor – Slovenia

[e-mail: martin.savc@um.si]

\*Corresponding author: Martin Šavc

*Received April 18, 2014; revised July 17, 2014; revised September 8, 2014; accepted October 11, 2014;  
published December 31, 2014*

---

## **Abstract**

This article presents a reformulation of the Grey Edge framework for colour constancy. Colour constancy is the ability of a visual system to perceive objects' colours independently of their scenes' illuminants. Colour constancy algorithms try to estimate the colour of an illuminant from image values. This estimation can later be used to correct the image as though it were taken under a white illuminant. The modification presented allows the framework to incorporate image-specific filters instead of the commonly used edge detectors. A colour constancy algorithm is proposed using PCA and FastICA linear component analyses methods for the construction of such filters. The results show that the proposed method improves the accuracies of the Grey Edge framework algorithms whilst on the other hand, achieving comparable accuracies with the state-of-the-art methods, but improving their time efficiencies.

---

**Keywords:** colour constancy, grey edge framework, image components, linear analysis, PCA, FastICA

## 1. Introduction

Colour constancy is the ability to perceive the colours of observed objects invariant with respect to the colours of the scenes' illuminants [1]. Such an ability would be very useful for computer vision and image analyses (e.g., for colour-based object detection and tracking, and surface colour analysis). Colour constancy algorithms use image colour values for estimating the colour of the illuminant. Due to the difficulty of estimating the illuminant for each image pixel (e.g. [2]) most algorithms assume that the scene contains uniform illuminant [1].

Based on our research experiences, three basic types of estimation mechanism can be identified. The first type exploits the physical properties of light and surface interactions. Most of such algorithms exploit the dichromatic model of image formation (e.g. [3], [4]). They estimate the illuminant by identifying specular surfaces' reflectances. The second type exploits the statistical regularities of image features. These features include raw pixel values or spatially filtered values, observed image colour gamuts or high-level semantic features. This type also includes various biologically-inspired algorithms [2]. The third type combines or selects from individual algorithms based on image classification, e.g. [5], [6]. Such algorithms only estimate the illuminant indirectly. An overview and comparison of recent algorithms can be found in [1]. Our work focused on statistically-based algorithms, that do not depend on certain physical phenomena and many of them are also computationally efficient.

An important advance in statistically-based algorithms has been the Grey Edge framework described in [7]. It has introduced the usages of spatial image derivatives for colour constancy. This has then inspired several further improvements, e.g. [8], [9], [10], [11]. A likely explanation for their success is the de-correlating effects of the derivatives. Thus, large uniform surfaces have less effect on the estimation, which is a well-known problem of simple pixel-based algorithms.

Existing methods rely on predefined linear filters for spatially de-correlating image data. This article proposes the use of de-correlating linear filters adjusted to individual image data. Firstly, image samples are extracted and pre-processed. These samples are then analysed in order to find spatial filters based on image content. The filters are used together with the Grey Edge framework. This framework was selected for its simplicity and efficiency.

This paper is organised as follows. In **Section 2**, the Grey Edge framework introduced by [7] is reformulated to accommodate image-specific filters. The construction of filters using image component analysis is presented in **Section 3**. **Section 4** describes the experiment used to validate the performance of the proposed method, followed by comparisons of the obtained results with the results of other algorithms. **Section 5** discusses and compares the results.

## 2. Reformulation of Grey Edge framework

In this section the Grey Edge (GE) framework has been reformulated to accommodate arbitrary linear filters. The GE framework [7] estimates the illuminant  $\mathbf{e} = [e_R e_G e_B]^T$  as:

$$k e_c^{j,m,\sigma} = \left( \int \left| \frac{\partial^j I_c^\sigma(\mathbf{x})}{\partial \mathbf{x}^j} \right|_F^m d\mathbf{x} \right)^{1/m}, \quad (1)$$

where  $I_c(\mathbf{x})$  is the image value at spatial coordinate  $\mathbf{x}$  for colour channel  $c$ . The superscript  $\sigma$  denotes a smoothing operation and  $j$  the order of the applied spatial derivative. The  $|\cdot|_F$  denotes the Frobenius norm. The Minkowski norm with parameter  $m$  over all image points calculates the illuminant estimate  $e_c^{j,m,\sigma}$  multiplied by an arbitrary factor  $k$ . The final estimate of illuminant colour  $\hat{\mathbf{e}}^{j,m,\sigma}$  is the normalised value of  $e^{j,m,\sigma}$  [7].

In practice, image smoothing and derivation are combined through convolution of the image with orthogonal kernels of spatially-derived Gaussian functions. A colour image  $\mathbf{I}(x, y)$  is first split into three colour channel images  $I_R(x, y)$ ,  $I_G(x, y)$ , and  $I_B(x, y)$ , which are then convolved separately with a kernel  $G_f$  as:

$$I_{c,f}(x, y) = I_c * G_f(x, y), \quad (2)$$

where  $*$  denotes convolution. It should be stressed that the spatial vector  $\mathbf{x}$  was expanded into its coordinates  $(x, y)$ . Each derivative is calculated using its corresponding kernel.

By using the above-mentioned derivative estimation, **Eq. (1)** for illuminant estimation can be rewritten as a convolution framework (CFW) for illumination estimation as:

$$e_c^{p,m} = \left[ \sum_{x,y} \left( \sum_f^F |I_{c,f}(x, y)|^p \right)^{m/p} \right]^m, \quad (3)$$

where  $|\cdot|$  denotes the absolute value. The convoluted data is combined using a vector  $p$ -norm (parameter  $p$ ) over the  $F$  different kernels at each position. When  $p = 2$ , the results equal the Frobenius norm used in **Eq. (1)**. A diagram of this framework is shown in **Fig. 1**. The input image on the right is convolved with the selected kernels  $G_f(x, y)$ . A  $p$ -norm combined image is shown on the right. The  $p$  and  $m$  norms are denoted as  $\|\cdot\|_p$  and  $\|\cdot\|_m$ , respectively.

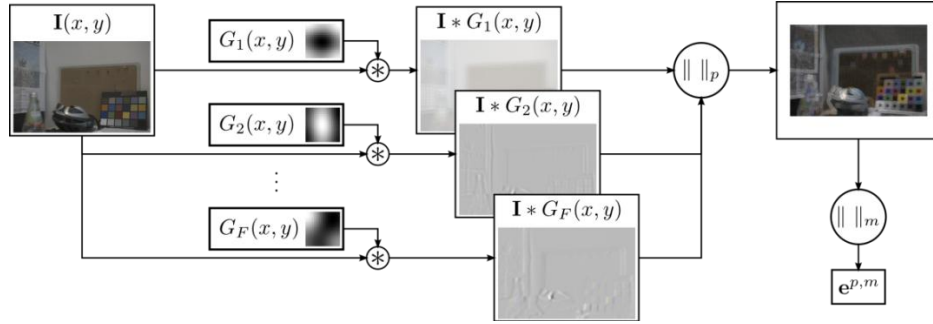
The  $F = 2$  first order derivative kernels are used in order to calculate the first-order Grey Edge using the CFW. The  $F = 4$  second order derivative kernels are used in order to calculate the second-order Grey Edge. Only an optional smoothing kernel should be used in order to calculate Grey World Assumption.

One benefit of this modified framework based on convolution is that the general Gaussian derivative kernels can be replaced with image-specific kernels. These kernels can be designed to find spatially decorrelated or independent values within a specific image.

### 3. Proposed method

This section describes the method for constructing image-specific kernels, which is the main contribution of this paper. These kernels are applied afterwards within a reformulated GE framework (see **Section 2**) for an estimated calculation of the illuminant.

A diagram of the proposed method is shown in **Fig. 2**. Firstly, the samples are extracted from the image and preprocessed. These samples are then analysed using either PCA or FastICA methods [12] resulting in linear image components and estimators. The estimators are transformed into filter kernels to be used within CFW (see the left side in **Fig. 2**). The results can also be approximated using the extracted image samples (the right side in **Fig. 2**).



**Fig. 1.** Diagram of the reformulated convolution-based GE framework using filter kernels.

Large images are first scaled down to a manageable size. Sampling of large images would result in either too big or too many samples. Recent work [10] has shown that image scaling can also speed up illumination estimation without degrading its accuracy.

A sampling window of fixed size  $W_\theta$  and with fixed sampling step  $W_\delta$  is selected to traverse the image. At each position 3 samples  $J_{c,n}(x, y)$  are extracted, where  $(x, y)$  denote the position inside the window. The index  $n$  is the sequential number of each sample. Samples are discarded that contain saturated or otherwise unwanted elements. Let the total number of valid sampling positions be denoted as  $M$  then the total number of samples is  $N = 3M$ .

Each extracted sample is preprocessed by subtracting the mean and normalised by the standard deviation calculated across all samples. The preprocessed sample  $\bar{J}_{c,n}(x, y)$  is calculated as:

$$\bar{J}_{c,n}(x, y) = \frac{J_{c,n}(x, y) - \mu_c(x, y)}{\sigma_c(x, y) + \epsilon}, \quad (4)$$

where  $\mu_c(x, y)$  and  $\sigma_c(x, y)$  are the estimated mean and standard deviation at position  $(x, y)$  inside the window, respectively, whilst  $\epsilon$  (e.g.  $10^{-3}$ ) is a small positive constant.

The preprocessed samples are then analysed using PCA or FastICA analysis [12]. Both methods estimate a linear model for the extracted samples as:

$$\bar{s}_{c,f}(n) = \sum_{x,y} \bar{J}_{c,n}(x, y) G_f(W_\theta - x, W_\theta - y). \quad (5)$$

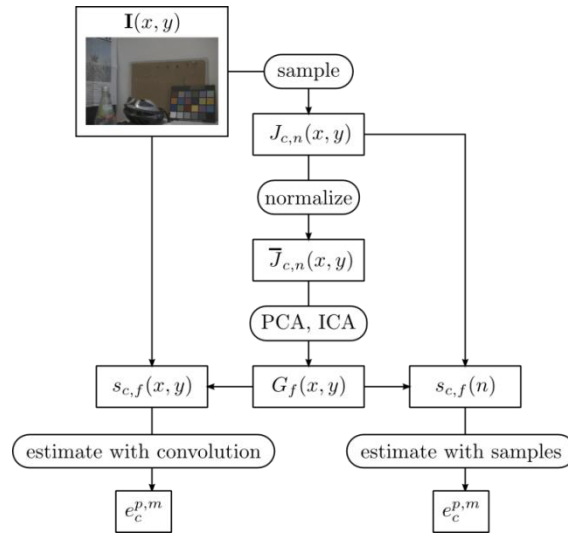
Values  $\bar{s}_{c,f}(n)$  are descriptors of  $\bar{J}_{c,n}(x, y)$  and  $G_f(x, y)$  are estimators. The estimators in Eq. (5) are traversed backwards so that they can be used as convolution kernels in Eq. (2).

When  $\bar{s}_{c,f}(n)$  and  $G_f(x, y)$  are estimated using PCA, the values  $\bar{s}_{c,f}(n)$  are spatially non-correlated for different  $n$ . On the other hand, when  $\bar{s}_{c,f}(n)$  and  $G_f(x, y)$  are estimated using FastICA, then descriptors  $\bar{s}_{c,f}(n)$  are also spatially independent [12].

The estimators are the first  $F$  principal components for the PCA method. The parameter  $F$  is the number of independent components to be estimated for the FastICA method.

Similar results can be achieved by using only the descriptors  $s_{c,f}(n)$  of image samples  $J_{c,n}(x, y)$ . These are calculated as:

$$s_{c,f}(n) = \sum_{x,y} J_{c,n}(x, y) G_f(W_\theta - x, W_\theta - y). \quad (6)$$



**Fig. 2.** Diagram of the proposed method with image sampling, sample analysis, and estimation.

The illuminant is then estimated by summing over all samples  $N$  instead of all image positions in **Eq. (3)**. If the sampling window step  $W_\delta$  equals 1, then this equation equals **Eq. (2)**. The difference increases as the window step gets larger.

## 4. Experiment and results

The error between the estimated illuminant  $\mathbf{e}_{est}$  and the ground truth illuminant  $\mathbf{e}_{gt}$  was calculated using a well-known angular error defined as **[13]**:

$$E_{ang} = \arccos \left( \frac{\mathbf{e}_{est}}{\|\mathbf{e}_{est}\|_2} \cdot \frac{\mathbf{e}_{gt}}{\|\mathbf{e}_{gt}\|_2} \right). \quad (7)$$

The angular error is measured in degrees. Reporting results follows an established practice from the literature (e.g. **[13]**). The results are summarised using the mean, median, and trimean errors. The obtained results were compared either with the published results from **[14]** or with the results of available implementations, where the results were unavailable.

### 4.1 Colour checker dataset

The proposed method was tested on a dataset of natural images taken by P. Gehler et al. **[15]**, preprocessed and made available by L. Shi et al. **[16]**. This will now be referred to as the GehlerShi dataset. This dataset contains 568 images. Each image contains the Macbeth colour checker card used to extract ground-truth illuminant values. The area containing the card was masked out during the experiments along with any pixels having values greater than 85% of the maximal possible pixel value. This threshold was determined by observing the number of pixels with the exact maximal image value. Three splits defined by the original author **[15]** were used for cross-validation. The dataset also contains labels for indoor and outdoor image classifications (see **[15]**). These labels enable detailed analyses of results with respect to indoor or outdoor scene classifications.

A slightly different preprocessing of this dataset was done by S. E. Lynch et al. [17]. This will be referred to as the GehlerLynch dataset. The images were converted from cameras to a linear sRGB colour space. Only the 482 images were retained in this dataset. Detailed information can be found in [17]. The same splits as defined for the original dataset were applied.

## 4.2 Parameter tuning

The proposed method has 6 parameters, namely image scaling factor  $\varphi$ , sampling window size  $W_\sigma$ , sampling window step  $W_\delta$ , number of kernels  $F$ , and two norm parameters  $p$  and  $m$ . In order to tune the parameters, an exhaustive search over selected values was conducted, using cross-validation training sets. The selected values are gathered in Table 1. The values were selected based on experience from our previous experiments.

**Table 1.** Parameters, values used in the exhaustive search and the best values found for each dataset, each method and each split (displayed from left to right).

| Parameter              | A set of values                         | GehlerShi |     |     | GehlerLynch |     |     |     |    |    |    |          |          |
|------------------------|---|-----------|-----|-----|-------------|-----|-----|-----|----|----|----|----------|----------|
|                        |   | PCA       | ICA | PCA | ICA         | PCA | ICA |     |    |    |    |          |          |
| $\varphi$              | [2, 4, 8]                               | 2         | 2   | 2   | 2           | 2   | 2   | 8   | 2  | 2  | 4  | 2        | 2        |
| $(W_\theta, W_\delta)$ | [16,14, 12,10]                          | 10        | 10  | 10  | 10          | 12  | 10  | 12  | 10 | 14 | 10 | 14       | 14       |
| $F$                    | [24, 16, 12, 8, 6, 4, 2]                | 16        | 12  | 12  | 24          | 16  | 8   | 24  | 24 | 24 | 24 | 16       | 16       |
| $p$                    | [0.5, 1, 2, 3, 4,5, 6, 7, 8, $\infty$ ] | 0.5       | 0.5 | 0.5 | 2           | 2   | 1   | 0.5 | 1  | 1  | 3  | $\infty$ | $\infty$ |
| $m$                    | [0.5, 1, 2, 3, 4,5, 6, 7, 8, $\infty$ ] | 2         | 2   | 2   | 2           | 2   | 2   | 2   | 3  | 3  | 2  | 3        | 3        |

The number of samples extracted from an image is mostly controlled by window-step  $W_\delta$ . As long as the number of samples is fairly large ( $3N \gg W_{\sigma^2}$ ) the exact selection of  $W_\delta$  has little effect on the results of the estimation. For this exhaustive search, the window-step  $W_\delta$  was arbitrarily chosen as half of window size  $W_\theta$ .

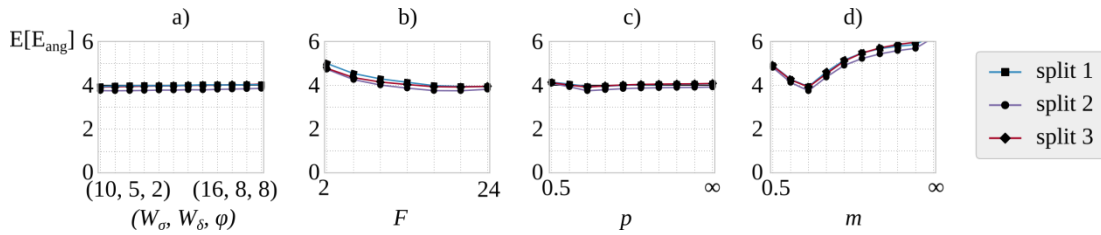
A scaled image with  $1/\varphi$ -times the width and  $1/\varphi$ -times the height of the original image is constructed by using a scaling parameter  $\varphi$ . Scaling the image by a factor of  $1/\varphi$  has a similar effect as changing the size of the sampling window by a factor  $\varphi$ . Using a smaller image produces similar results but requires less memory space and processing time.

The illuminant was estimated with image samples by using Eq. (6). For each training split the parameters with the minimal mean angular error were chosen after the exhaustive search.

The mean angular error as a function of an observed parameter is studied in order to obtain some insight into the importance of specific parameters. The unobserved parameters are set to their optimal values. Charts for the FastICA method on GehlerShi dataset are shown in Fig. 3. The sampling window size  $W_\theta$  and step  $W_\delta$  have little effect (see chart (a)). Increasing the number of components  $F$  slowly improves the mean angular error (see chart (c)) but levels off soon. In addition, the parameter  $p$  does not have a noticeable effect on the mean angular error (see chart (b)). The parameter  $m$ , however, has a noticeable local minimum at 2.0 for all three splits (see chart (d)). Other charts were similar and, are thus, left out.

## 4.3 Results

The experimental results are summarised in the following tables. The proposed method is labelled as  $CFW_{PCA}$  or  $CFW_{ICA}$  for PCA and FastICA methods' analyses, respectively. The illuminant was estimated using Eq. (6). Labels  $GGW$ ,  $GE1$  and  $GE2$  denote the methods'



**Fig. 3.** Effect of selected parameter on the mean angular error using the FastICA method on the GehlerShi dataset. Unobserved parameters are set to optimal values.

General Grey World, and the first and second-order Grey Edges, respectively. These methods are described by the original GE framework. The parameters for these methods were chosen using cross validation, as reported by [7]. The results are also compared with the SpatioSpectral statistics (*SpSpStats*) method proposed by [9] and the derivative methods *SpSpStatsEdge*, *SpSpStatsChr* proposed by [10]. The last two methods are Photometric edge weighting (*PWEdge*) by [8] and the Zeta Image method (*ZetaImage*) by [4].

Firstly, the selected algorithms were compared in terms of used processing time (see Table 2). The processing times of using Eq. (2) based on image filtering and using Eq. (6) based on image samples are both reported. The algorithms were implemented in Matlab using the original authors' implementations where available. The parameters of the proposed methods were  $\varphi = 2$ ,  $W_\theta = 16$ ,  $W_\delta = 8$ ,  $F = 16$ ,  $p = 2$ ,  $m = 2$ , whilst *GGW*, *GE1*, and *GE2* methods used  $\sigma = 3$ ,  $m = 2$ . The parameters for *PWEdge* were  $\sigma = 1$  and  $\kappa = 6$ . The measurements were done on a computer system with an Intel Core i5 650 processor having a 3.2 GHz system clock and 8GB RAM. It can be noticed that *CFW<sub>P</sub>CA* using Eq. (6) is approximately 5 times slower than the fastest but one of the simpler methods (*ZetaImage*), and about 5 times faster than the sophisticated but slowest method (*SpSpStats*).

**Table 2.** Mean run times with standard deviations for compared algorithms implemented in Matlab. Times were measured by processing all images from the GehlerLynch dataset scaled with  $\varphi = 2$ .

| Algorithm                          | Mean run time (s) | Algorithm                          | Mean run time (s) |
|------------------------------------|-------------------|------------------------------------|-------------------|
| <i>ZetaImage</i>                   | $0.26 \pm 0.01$   | <i>CFW<sub>ICA</sub></i> (Eq. (6)) | $2.98 \pm 0.73$   |
| <i>GGW</i>                         | $0.32 \pm 0.02$   | <i>SpSpStatsEdge</i>               | $3.88 \pm 1.11$   |
| <i>PWEdge</i>                      | $0.37 \pm 0.13$   | <i>SpSpStatsChr</i>                | $3.75 \pm 0.62$   |
| <i>GE1</i>                         | $0.43 \pm 0.02$   | <i>SpSpStats</i>                   | $5.99 \pm 0.47$   |
| <i>GE2</i>                         | $0.57 \pm 0.03$   | <i>CFW<sub>P</sub>CA</i> (Eq. (2)) | $7.05 \pm 0.26$   |
| <i>CFW<sub>P</sub>CA</i> (Eq. (6)) | $1.23 \pm 0.09$   | <i>CFW<sub>ICA</sub></i> (Eq. (2)) | $8.28 \pm 0.60$   |

In the sequel, the selected algorithms were compared with respect to the angular error of the estimated illuminants. Table 3 shows the obtained results on the GehlerShi dataset. The results from the *ZetaImage* implementation differed slightly from the results reported in [4]. The results of *SpSpStatsEdge*, *SpSpStatsChr* were omitted, as they were very similar to *SpSpStats* method. The results obtained on the GehlerLynch dataset are collated in Table 4. The parameters for the methods were determined using predefined cross-validation sets.

The label *Mean Ill* denotes the error of the illuminant, estimated using the mean ground truth illuminant calculated over all images within a dataset. The *do nothing* baseline error uses a neutral RGB illuminant estimate, i.e. equal values for all colour channels. The white point of

the camera colour space of images in the GehlerShi dataset does not correspond to the neutral value in RGB space. This is reflected in the much larger errors of the *do nothing*. The tables also report the results for indoor and outdoor image subsets.

**Table 3.** Mean, median and trimean angular errors for the selected algorithms on the GehlerShi dataset. The best result in each category is marked.

| Algorithm                | all images (568) |            |            | indoor images (246) |            |            | outdoor images (322) |            |            |
|--------------------------|------------------|------------|------------|---------------------|------------|------------|----------------------|------------|------------|
|                          | mean             | median     | trimean    | mean                | median     | trimean    | mean                 | median     | trimean    |
| <i>do nothing</i>        | 13.7             | 13.6       | 13.5       | 13.9                | 13.4       | 13.5       | 13.5                 | 13.6       | 13.6       |
| <i>Mean Ill</i>          | 6.8              | 5.6        | 5.7        | 7.0                 | 6.4        | 6.4        | <b>2.3</b>           | <b>1.9</b> | <b>1.9</b> |
| <i>GGW</i>               | 4.7              | 3.5        | 3.8        | 5.0                 | 3.9        | 4.3        | 4.3                  | 3.2        | 3.5        |
| <i>GE1</i>               | 5.3              | 4.5        | 4.7        | 5.7                 | 5.3        | 5.4        | 5.0                  | 4.2        | 4.3        |
| <i>GE2</i>               | 5.1              | 4.4        | 4.6        | 5.1                 | 4.6        | 4.7        | 5.2                  | 4.2        | 4.5        |
| <i>PWEdge</i>            | 5.4              | 4.6        | 4.8        | 6.3                 | 6.2        | 6.1        | 4.7                  | 3.8        | 4.0        |
| <i>Zetalimage</i>        | 4.0              | 3.2        | 3.4        | <b>3.8</b>          | 3.3        | 3.4        | 4.2                  | 3.2        | 3.6        |
| <i>SpSpStats</i>         | <b>3.7</b>       | 3.0        | <b>3.1</b> | 4.2                 | 3.6        | 3.7        | <b>3.3</b>           | <b>2.5</b> | <b>2.7</b> |
| <i>CFW<sub>PCA</sub></i> | <b>3.7</b>       | <b>2.7</b> | <b>3.1</b> | <b>3.8</b>          | <b>3.0</b> | <b>3.3</b> | 3.6                  | <b>2.5</b> | 2.8        |
| <i>CFW<sub>ICA</sub></i> | 4.0              | 3.0        | 3.3        | 4.0                 | 3.1        | 3.5        | 3.9                  | 3.0        | 3.3        |

Each pair of algorithms was further tested using the Wilcoxon signed rank test as suggested in [10]. All tests were made at 0.05 significance level. The following conclusions were drawn.

On the entire GehlerShi dataset the illuminant estimations obtained by *CFW<sub>ICA</sub>* were statistically significantly better than those of *GGW*, *GE1*, *GE2* and *PWEdge*, whilst any differences between estimations obtained by *Zetalimage* and our algorithm were statistically insignificant. The estimations obtained by *CFW<sub>PCA</sub>* were statistically superior than those of *GGW*, *GE1*, *GE2*, *PWEdge*, *Zetalimage* and *CFW<sub>ICA</sub>*. The results of *SpSpStats* were statistically significantly better than those of all other algorithms.

**Table 4.** Mean, median and trimean angular errors for the selected algorithms on the GehlerLynch dataset. The best result in each category is marked.

| Algorithm                | all images (482) |            |            | indoor images (175) |            |            | outdoor images (307) |            |            |
|--------------------------|------------------|------------|------------|---------------------|------------|------------|----------------------|------------|------------|
|                          | mean             | median     | trimean    | mean                | median     | trimean    | mean                 | median     | trimean    |
| <i>do nothing</i>        | 9.7              | 5.1        | 6.5        | 19.4                | 21.6       | 20.6       | 4.1                  | 3.7        | 3.7        |
| <i>Mean Ill</i>          | 7.8              | 6.2        | 6.4        | 8.7                 | 9.4        | 8.9        | <b>2.1</b>           | <b>1.7</b> | <b>1.7</b> |
| <i>GGW</i>               | 4.1              | 2.5        | 3.0        | 4.6                 | 3.2        | 3.5        | 3.8                  | <b>2.1</b> | 2.7        |
| <i>GE1</i>               | 3.9              | 2.6        | 3.0        | 4.2                 | 3.0        | 3.2        | 3.8                  | 2.5        | 2.9        |
| <i>GE2</i>               | 4.1              | 2.7        | 3.1        | 4.6                 | 3.0        | 3.3        | 3.9                  | 2.5        | 2.9        |
| <i>PWEdge</i>            | 3.9              | 2.4        | 2.8        | 4.7                 | 2.8        | 3.2        | 3.4                  | 2.2        | 2.6        |
| <i>Zetalimage</i>        | 4.3              | 2.9        | 3.3        | 3.7                 | 2.8        | <b>3.0</b> | 4.6                  | 3.0        | 3.6        |
| <i>SpSpStats</i>         | <b>3.3</b>       | 2.5        | <b>2.7</b> | <b>3.5</b>          | 3.0        | <b>3.0</b> | <b>3.1</b>           | 2.2        | <b>2.5</b> |
| <i>CFW<sub>PCA</sub></i> | 4.0              | 2.6        | 3.0        | 4.2                 | 2.7        | 3.1        | 3.9                  | 2.4        | 2.9        |
| <i>CFW<sub>ICA</sub></i> | 3.8              | <b>2.3</b> | 2.9        | 3.9                 | <b>2.5</b> | <b>3.0</b> | 3.7                  | 2.2        | 2.8        |

On the GehlerLynch dataset the accuracy of *CFW<sub>PCA</sub>* was insignificantly different than *GGW*, *GE1*, and *GE2*, whereas *CFW<sub>ICA</sub>* was statistically significantly better than *GGW*, *GE1*, *GE2*,



*ZetaImage* and  $CFW_{PCA}$ . Estimations obtained by *PWEdge* were statistically equal to  $CFW_{ICA}$  results. The illuminant estimations of *SpSpStats* were statistically significantly better than estimations of other compared algorithms. The *SpSpStats*,  $CFW_{PCA}$  and  $CFW_{ICA}$  illuminant estimations for the indoor images for both datasets did not differ statistically.

As noted in [13], the just noticeable perceptual difference between the two algorithms is 6% difference between their angular errors. Using this criterion to compare algorithms, the  $CFW_{PCA}$  was noticeably better than *SpSpStat* for 39.1% and there were no noticeable differences for 10.6% of the images in the GehlerShi dataset. The  $CFW_{ICA}$  was better than *SpSpStat* or there were no noticeable differences for 46.1% of the images within the GehlerLynch dataset.

*SpSpStat* was indeed statistically the more accurate but the above analysis pointed out that our method demonstrates a comparable success, especially in the case of indoor scenes' images; on the other hand, our method outperforms *SpSpStat* by around five times in terms of the time complexity. Our method can therefore be a suitable replacement for the *SpSpStat* method in cases where the time complexity is the crucial factor.

Examples of the image-specific kernels constructed by PCA and FastICA methods for two images are shown in Fig. 4.

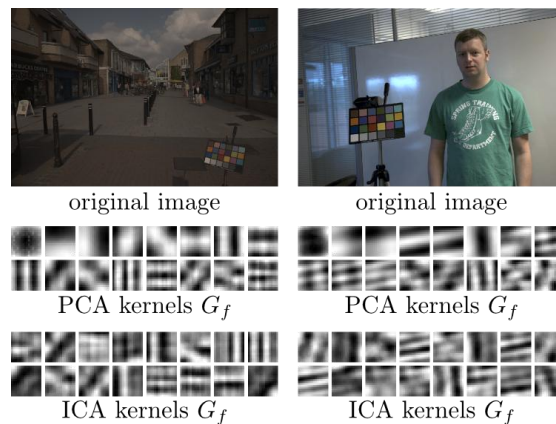
## 5. Discussion and conclusion

A colour constancy algorithm was proposed that extends the GE framework. The generic convolution kernels for GE were replaced in order to improve the spatial decorrelation with image-specific kernels constructed by using either PCA or FastICA analyses methods. The experimental evaluation showed this approach improved illuminant estimation compared to other GE framework methods and is comparable to state-of-the-art methods.

The overall results of the presented method were superior compared to the results of GE framework methods on both datasets, and comparable (in some respects slightly worse) to the results of the *SpSpStats* method. A recently published colour constancy method [11] reported far better results. However, its authors have not made the code of their algorithm publicly available. Despite extensive efforts, it was impossible to replicate the reported results using our own implementation of their algorithm. Therefore, this algorithm was excluded for comparison. Analyses of both our proposed methods (i.e., based on PCA or FastICA) pointed out that neither of the approaches seemed superior (see Table 3 and Table 4, rows  $CFW_{PCA}$  and  $CFW_{ICA}$ ). Interestingly, using PCA analysis produced slightly better results on the GehlerShi dataset, which was contrary to expectations. However, FastICA outperformed PCA on the correctly reprocessed GehlerLynch dataset.

Some interesting observations can be made by looking at the results of indoor and outdoor images separately. Looking at the *Mean Ill* error points out that most of the illuminant variations within the dataset came from indoor images. For the outdoor images the *Mean Ill* error was lower than the error of any colour constancy algorithm used in this study. On the indoor images, the proposed methods produced results similar to the *SpSpStats* method.

Our proposed methods improved the results of the basic GE framework methods and are comparable in accuracies to the results of the *SpSpStats* method. By using an approximation of the original GE framework, the presented methods were also up to 4.8 times faster than the Spatio Spectral Statistics. It can be concluded that the presented methods present a viable compromise between simpler but faster and more sophisticated but slower methods.



**Fig. 4.** Examples of image-specific kernels. Sixteen kernels per original image constructed by PCA analysis are shown in the second row, whilst the kernels obtained by FastICA method are shown in the third row. The images have been preprocessed for display purposes.

It can be seen from the examples of image-specific kernels in **Fig. 4** that the PCA kernels are ranked from lower to higher frequency content. The PCA kernels were also very similar between both images. On the other hand, the FastICA kernels could not be ordered easily. They also contained more visible variations between images.

Based on the obtained results, the proposed methods could also be used in combination with other GE framework and non GE framework methods. Many existing algorithms assume that the illuminant can be separated from the reflectance values based on spatial frequency analysis—an example is the Local space average colour method [18]. Based on this assumption, weighting schemes based on frequency analyses of filters constructed using the FastICA method through the GE framework could be used for improving illuminant estimations.

The proposed methods analyse achromatic image samples. Coloured image samples can be analysed in a similar way. Estimating the illuminant from the results of such analysis would not be trivial. The three channel colour values are reduced to single channel descriptors. Colour information is contained within the kernels. However, it would help exploit regularities between colour channels that are exploited by *SpSpStats* and other sophisticated methods.

In conclusion, an improvement of the GE framework was presented by introducing the usage of image-specific kernels. The kernels were constructed using PCA and FastICA methods for improving spatial decorrelation. The results show an improvement in Grey Edge and General Grey World methods. The results are also comparable with the state-of-the-art colour constancy method Spatial Spectral Statistics.

## 6. References

- [1] A. Gijsenij, T. Gevers, and J. van de Weijer, “Computational Color Constancy: Survey and Experiment,” *IEEE Trans. on Image Processing*, vol. 20, no. 9, pp. 2475–2489, February, 2011. [Article \(CrossRef Link\)](#)
- [2] B. Funt, F. Ciurea, and J. McCann, “Retinex in Matlab,” *J. Electron. Imaging*, vol. 13, pp. 48-57, January, 2004. [Article \(CrossRef Link\)](#)
- [3] L. Shi and B. Funt, “Dichromatic illumination estimation via hough transforms in 3D,” in *Proc. of Conf. on Colour in Graphics, Imaging, and Vision*, pp. 259–262, June, 2008.
- [4] M. Drew, H. Vaezi Joze, and G. Finlayson, “Specularity, the Zeta-image, and Information-Theoretic Illuminant Estimation,” *Computer Vision – ECV2012: Workshops and*

- Demonstrations*, pp. 411–420, October, 2012. [Article \(CrossRef Link\)](#)
- [5] A. Gijsenij and T. Gevers, “Color Constancy Using Natural Image Statistics and Scene Semantics,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 687–698, February, 2011. [Article \(CrossRef Link\)](#)
- [6] S. Bianco, G. Ciocca, C. Cusano, and R. Schettini, “Automatic color constancy algorithm selection and combination,” *Pattern Recognition*, vol. 43, no. 3, pp. 695–705, March, 2010. [Article \(CrossRef Link\)](#)
- [7] J. van de Weijer, T. Gevers, and A. Gijsenij, “Edge-Based Color Constancy,” *IEEE Trans. on Image Processing*, vol. 16, no. 9, pp. 2207–2214, August, 2007. [Article \(CrossRef Link\)](#)
- [8] A. Gijsenij, T. Gevers, and J. Van De Weijer, “Improving color constancy by photometric edge weighting,” *IEEE Trans. on Pattern Analysis and Machine Int.*, vol. 34, no. 5, pp. 918–929, March 2012. [Article \(CrossRef Link\)](#)
- [9] A. Chakrabarti, K. Hirakawa, and T. Zickler, “Color constancy with spatio-spectral statistics,” *IEEE Trans. on Pattern Analysis and Machine Int.*, vol. 34, no. 8, pp. 1509–1519, June 2012. [Article \(CrossRef Link\)](#)
- [10] M. Rezagholizadeh and J. J. Clark, “Edge-Based and Efficient Chromaticity Spatio-spectral Models for Color Constancy,” in *Proc. of International Conf. on Computer and Robot Vision*, pp. 188–195, May, 2013. [Article \(CrossRef Link\)](#)
- [11] S. Lai, X. Tan, Y. Liu, B. Wang, and M. Zhang, “Fast and robust color constancy algorithm based on grey block-differencing hypothesis,” *Optical Review*, vol. 20, no. 4, pp. 341–347, July, 2013. [Article \(CrossRef Link\)](#)
- [12] A. Hyvärinen, J. Hurri, and P. O. Hoyer, *Natural Image Statistics*, vol. 39, Springer, London, 2009. [Article \(CrossRef Link\)](#)
- [13] A. Gijsenij, T. Gevers, and M. P. Lucassen, “Perceptual analysis of distance measures for color constancy algorithms,” *J. of Optical Society of America A*, vol. 26, no. 10, pp. 2243–2256, September, 2009. [Article \(CrossRef Link\)](#)
- [14] Published Results of Colour constancy algorithms accessed from [colorconstancy.com](http://colorconstancy.com) on 1.1.2014.
- [15] P. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp, “Bayesian Color Constancy Revisited,” in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pp. 1–8, June, 2008. [Article \(CrossRef Link\)](#)
- [16] L. Shi and B. Funt, “Re-processed Version of the Gehler Color Constancy Dataset of 568 Images,” accessed from [www.cs.sfu.ca/~colour/data/](http://www.cs.sfu.ca/~colour/data/) on 21.11.2013.
- [17] S. E Lynch, M. S. Drew, and G. D. Finlayson “Colour Constancy from Both Sides of the Shadow Edge,” in *Proc. of Color and Photometry in Computer Vision Workshop at the International Conf. on Computer Vision*, pp. 899–906, December, 2013. [Article \(CrossRef Link\)](#)
- [18] M. Ebner, *Color constancy*, vol. 6. John Wiley & Sons, 2007.



**Martin Šavc** is a teaching assistant and doctoral student at Faculty of Electrical Engineering and Computer Science in University of Maribor, Slovenia. His main research interests are image processing and computer vision. His doctoral research focuses on colour constancy and colour image analysis.



**Božidar Potočnik** (MSc. 1998, DSc. 2000) is an associate professor at Faculty of Electrical Engineering and Computer Science in University of Maribor, Slovenia. His research interests are focused on digital image and image sequence processing, especially on advanced intelligent and self-adaptive segmentation methods. His bibliography counts more than 200 units, from these more than 20 scientific peer-reviewed journal papers.