Depth Map Color Constancy

Marc Ebner and Johannes Hansen
Ernst-Moritz-Arndt-Universität Greifswald
Institut für Mathematik und Informatik
Walther-Rathenau-Straße 47, 17487 Greifswald, Germany
Tel: (+49)3834/86-4646, Fax: (+49)3834/86-4640
marc.ebner@uni-greifswald.de

Abstract

A human observer is able to determine the color of objects independent of the light illuminating these objects. This ability is known as color constancy. In the first stages of visual information processing, data is analyzed with respect to wavelength composition, orientation, motion and depth. With this contribution, we investigate whether depth information can help in estimating the color of the objects. We assume that local space average color is computed in V4 through resistively coupled neurons in order to estimate the color of the illuminant. We show how this computational model can be extended to incorporate depth information.

I. Introduction

Color is a product of the brain. Consider a scene with several objects. Depending on the light which is illuminating the scene, the irradiance measured by a sensor will vary. However, a human observer will perceive approximately constant object colors irrespective of the illuminant used (McCann, 2000; McCann et al, 1976; Zeki, 1993). A color constancy algorithm tries to mimic this ability and compute a color constant descriptor which is independent of the illuminant (Ebner, 2007a).

Several different algorithms have been proposed to compute a color constant descriptor (Brainard and Freeman, 1997; Finlayson and Hordley, 2001; Finlayson et al, 2001; Forsyth, 1990; Funt et al, 1991; Geusebroek et al, 2001). It has also been investigated whether color constancy can be learned (Funt et al, 1996; Hurlbert and Poggio, 1988). Some researchers have also tried to combine the output of several color constancy algorithms (Cardei and Funt, 1999). Lu et al (2009) have used a classifier to classify images into rough 3D geometry models. The most suitable color constancy algorithm (determined off-line on a larger data set) is used for each geometrical structure to estimate the illuminant. A weighted average of these estimates is taken as the illuminant estimate for the entire image.

Color constancy algorithms take an image as input (as measured by a sensor) and produce an output image. Some algorithms only compute a color constant descriptor while other algorithms try to estimate reflectance. A perfect algorithm would compute spectral reflectance of the objects shown in the image. However, this is not possible because usually we have a locally varying illuminant and only three measurements per pixel. Performance of color constancy algorithms has been analyzed in several different studies (Barnard et al, 2002a,b; Ebner, 2007a; Funt et al, 1998).

The problem of computing a color constant descriptor can be greatly simplified by assuming that a uniform illuminant is illuminating the scene. Given a three band input image, only three values need to be determined. Buchsbaum (1980) has put forward the gray world assumption. A requirement for the method to work is that there are a sufficiently large number of differently colored objects in the image. According to the gray world assumption, the world is gray on average. Simply put, it says that the average of each channel is a constant. This result can be used to estimate the illuminant. Van der Weijer et al. (2007) have put forward the gray edge hypothesis according to which the average edge difference in a scene is achromatic.

For natural scenes, however, the illuminant is usually not uniform. Several different color constancy algorithms have been proposed for scenes with varying illumination (Barnard et al, 1997). In this case, the illuminant needs to be estimated locally for each image pixel. The original Retinex algorithm of Land and McCann (1971) estimates the illuminant locally. Several different variants of this method have been proposed (Blake, 1985; Frankle and McCann, 1983; Funt et al, 2004; Horn, 1974; Moore et al, 1991). Land (1986) also provides a formulation of the Retinex theory where the difference between the logarithm of the lightness for a small area (or single pixel) and the logarithm of the lightness for an extended area is computed. In order to compute the average over an extended area, the receptors were assumed to be distributed with a density which varies with $\frac{1}{r^2}$ as the distance r from the current pixel increases. Gijsenij et al (2012) suggested patch based illumination estimation followed by clustering of the estimates and finally back-projecting the computed clusters to obtain a local estimate of the illuminant.

Ebner (2009) has shown that the gray world assumption can also be applied locally. The method is based on the computation of local space average color which is used as an estimate of the illuminant. In its original form, the method assumes a grid of processing elements (one for each image pixel). Each processing element computes local space average color by exchanging data with neighboring elements. A variant of this method may be used inside the human brain to arrive at a color constant descriptor (Ebner, 2012). According to this method, resistively coupled neurons in visual area V4 form a resistive grid, which is used to estimate local space average

color.

Another method tries to determine the illumination gradient (Ebner, 2007b). Using this method, each processing element averages the data in a direction perpendicular to the illumination gradient. This method works if the illumination gradient is smooth as shown in Figure 1. The method does not work well at a door which separates two rooms with different illuminants, because the iso-illumination lines are not quite correct in the corner of the door frame.

With this contribution, we investigate whether depth information is helpful in estimating the illuminant locally. Depth information is readily available in the human visual system. The idea is that given a large depth discontinuity which separates two objects in the image, it is quite likely that a different illuminant is illuminating the two objects. Hence, we should keep these two areas separate in estimating the illuminant. For machine vision, there are numerous different methods which can be used to obtain depth information (Horn, 1986; Jain et al, 1995; Szeliski, 2010). For our experiments, we have used a Microsoft Kinect (Microsoft Corporation, 2011) to obtain a depth map for the input images.

This article is structured as follows. In the next section, we briefly summarize the color constancy algorithm based on local space average color and how this algorithm can be mapped to the human visual system. In Section II we describe the human visual system and how this color constancy algorithm can be mapped to it. In Section III we show how this algorithm can be extended using depth information. Section IV describes how we used the Microsoft Kinect to obtain RGB images with an associated depth map. Section V describes the experiments which illustrate how the algorithm works. Section VI concludes this contribution.

II. A Computational Model for Color Vision Based on Local Space Average Color

Let us first see how a color constant descriptor may be computed by the brain. Ebner (2004) introduced a parallel algorithm for color constancy based on local space average color. This algorithm can also be mapped to what is known about the human visual system (Ebner, 2007c). Local space average color is assumed to be computed within visual area V4 of the brain using resistively coupled neurons. These neurons form a resistive grid. Once local space average color has been computed, a color constant descriptor can be computed by subtracting local space average color from the color measured by the retinal receptors. This algorithm and its mapping to the visual system is illustrated in Figure 2.

Visual information processing of course starts with the retinal receptors (Tovée, 1996). A review of cortical mechanisms of color vision is given by Gegenfurtner (2003). The human retina contains three types of receptors for color vision which mainly respond to light in the red, green, and blue parts of the spectrum (Dartnall et al, 1983; Dowling, 1987). So called color opponent cells perform a rotation of the coordinate system. After processing, the major axes are no longer red, green and blue. They are bright-dark, yellow-blue and red-green. The visual information is analyzed within the primary visual cortex, area V1. Inside V1, we find cells whose optimal stimulus is light of different wavelengths or oriented lines (Livingstone and Hubel, 1984). Local space average color is presumably computed in visual area V4 through resistively coupled neurons which form a resistive grid.

For the explanation of how a color constant descriptor is computed, we omit the rotation of the coordinate system. Let us assume that these receptors are very narrow-band. This assumption makes the theoretical derivation of the computation of a color constant descriptor easier. Of course in reality, the sensitivity of the retinal sensors is not narrow band and human color constancy is known not to be perfect (McCann et al, 1976).

Let $\mathbf{c}(x,y) = [c_r(x,y), c_b(x,y), c_g(x,y)]$ be the light falling onto the retina for three wavelengths in the red, green and blue part of the spectrum at position (x,y) of the retina. The output of the retinal sensor $\mathbf{c}'(x,y)$ is assumed to depend logarithmically on the incoming light (Herault, 1996; Land and McCann, 1971). Therefore, we have $c'_i(x,y) = log c_i(x,y)$ with $i \in \{r,g,b\}$. The light is assumed to be proportional to the reflectance $R_i(x,y)$ and the irradiance $L_i(x,y)$ at the corresponding object point. Hence, we have $c_i(x,y) = fR_i(x,y)L_i(x,y)$ for some f. Assuming that local space average color $\mathbf{a}$ is an estimate of the illuminant (see Ebner (2009)), i.e. $a_i(x,y) = \log L_i(x,y) + k$, where k is a constant, we obtain a color constant descriptor by subtracting local space average color from the output of the retinal sensor.

$$
\begin{aligned}
c'_i(x,y) - a_i(x,y) &= \log c_i(x,y) - a_i(x,y) && (1)\\
&= \log c_i(x,y) - \log L_i(x,y) - k && (2)\\
&= \log \frac{fR_i(x,y)L_i(x,y)}{L_i(x,y)} - k && (3)\\
&= \log R_i(x,y) + k' && (4)
\end{aligned}
$$

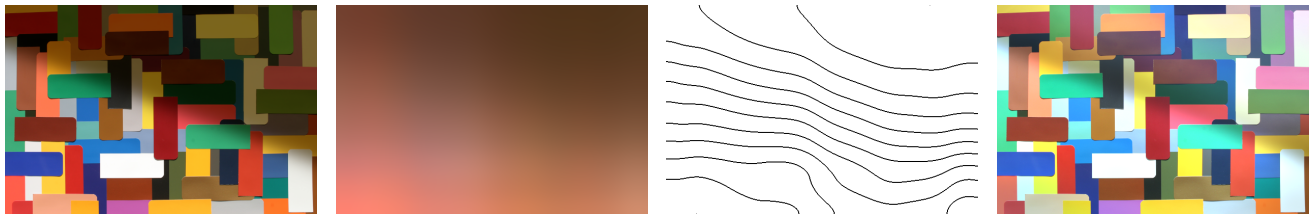
with $k' = \log f - k$.

Local space average color is computed through resistively coupled neurons. Let $a_i(x,y)$ be the estimated value of local space average color for channel i of a neuron whose receptive field corresponds to position (x,y) of the retina, let $N(x,y)$ be the neurons to which the current neuron is resistively coupled and let p be a small value larger than zero. The following two update equations are executed iteratively to model the resistive grid.

$$
\begin{aligned}
a'_i(x,y) &= \frac{1}{|N(x,y)|} \sum_{(x',y') \in N(x,y)} a_i(x',y') && (5)\\
a_i(x,y) &= c_i(x,y)p + a'_i(x,y)(1-p) && (6)
\end{aligned}
$$

The parameter p determines the area of support over which local space average color is computed and depends on the strength of the resistive coupling between neurons as detailed in (Ebner, 2009). Over time, $a_i(x,y)$ converges to local space average color. The time needed for convergence can be reduced if this process is simulated on a sequential computer by using successive over relaxation.

If we apply the algorithm to pre-stored images which use the sRGB color space, then we first have to apply a gamma correction to transform the RGB values to a linear color space.



(a) (b) (c) (d)

Fig. 1: (a) Image with smooth illumination gradient. (b) Illuminant estimate. (c) Estimated iso-illumination lines. (d) Output image. Algorithm described by Ebner (2007a).

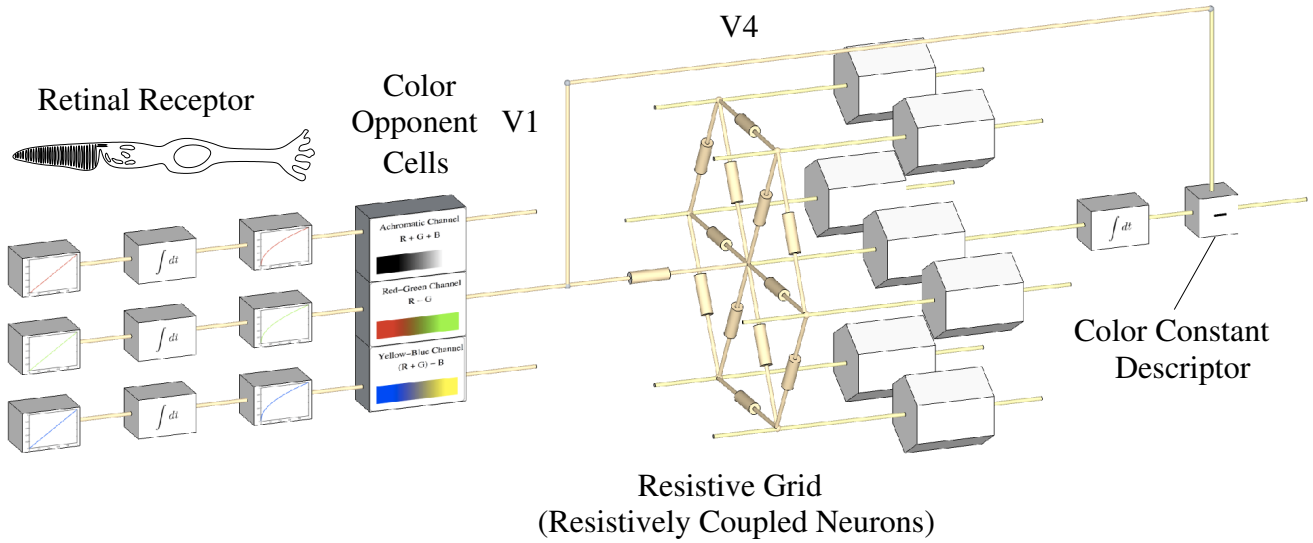


Fig. 2: Processing of visual information starts with the retinal receptors. The signal from the retinal receptors is sent to area V1. Color opponent cells perform a rotation of the coordinate system. Resistively coupled neurons in V4 are assumed to compute local space average color. A color constant descriptor is obtained by subtracting local space average color from the signal available from V1.

The parameter p depends on the size of the input image. We can make it independent of the dimensions of the input image by specifying the percentage $sigma$ of the image over which local space average color should be computed by setting

$$p = \frac{1}{\sigma^2 s^2 + 1} \tag{7}$$

where $s = \max\{w, h\}$ and w and h are the width and the height of the input image.

III. Improving Color Constancy Performance Using Depth Information

Depth information is readily available inside the human visual system. Processing of depth information can start with so called ocular dominance columns located in V1 (Livingstone and Hubel, 1984). With this contribution, we explore how depth information may aid in the computation of a color constant descriptor. Computing a color constant descriptor is especially difficult if there are several different illuminants in the scene. For natural scenes, we usually have multiple illuminants.

Natural light may be falling through a window while artificial light may already have been turned on inside the building. Consider a typical office scene with a desk lamp on top of a desk. The objects on top of the desk are directly illuminated by the desk lamp. However, the objects underneath the desk are illuminated by indirect light which may have been reflected multiple times by surrounding objects. Hence, two different illuminants illuminate the floor and the objects on top of the desk.

When we look at this desk, we see that a depth discontinuity separates the top of the desk from the floor. However, when we take an image of this scene with a standard digital camera we have no depth information. The algorithm described above would average across this depth discontinuity. However, it would make more sense to keep the area on top of the table separate from the area below the table. Given a stereo camera which also provides a dense depth map in addition to a color RGB image, we can make use of this depth information.

Another example is shown in Figure 3(a). The room, where the camera is located, is illuminated by sunlight falling through a window. The next room is illuminated by artificial light. The

door frame separates the two illuminants. By modifying the above algorithm, it is possible to keep the two areas separate. All we need to do is to detect large depth discontinuities and refrain from averaging across these depth discontinuities. Of course it may occur that the same illuminant is present on both sides of a depth discontinuity. However, treating the two areas as separate areas is assumed to do no harm.

Figure 3(b) shows how our modified algorithm estimates the illuminant. The edge separating the two areas was marked manually. We don't average across this edge. We have marked only the left side of the door and the top of the door frame. On the right hand side of the door we have a smooth illumination gradient. Hence, this border is not marked.

To take depth discontinuities into account, we assume that in addition to the linear RGB image $c_i(x, y)$, we also have a dense depth map $d(x, y)$ which specifies the distance of each object point to the camera. The depth map as well as the individual channels of the RGB image are assumed to have the range $[0, 1]$. All we now have to do is to use a modified neighborhood $N_e(x, y)$ instead of the neighborhood $N(x, y)$ which is used in the algorithm above. Let ϵ be the depth threshold. We do not want to average across pixels having a larger depth difference than ϵ. Hence, the neighborhood $N_e(x, y)$ is defined as follows.

$$N_e(x, y) = \{(x', y') \in N(x, y) \text{ with } |d(x, y) - d(x', y')| \leq \epsilon\} \tag{8}$$

In other words, neighboring pixel whose depth differs by more than ϵ are excluded from the averaging. Only neighboring color values with a depth difference smaller than ϵ are included in the average.

An estimate of reflectance obtained with this method is shown in Figure 3(c). This estimate was computed by dividing the measured color $c_i(x, y)$ shown in Figure 3(a) by twice the local space average color shown in Figure 3(b). It is clear that this method is able to obtain a very good estimate of the corresponding reflectance as both rooms now seem to be illuminated by the same, standard, illuminant.

Figure 4 shows how our method compares to three other methods: the standard gray world assumption, computation of local space average color without depth information, and anisotropic averaging along iso-illumination lines for two sample images. Both images were taken with a Canon EOS 5D Mark III. Naturally, the gray world assumption performs poorly in estimating the illuminant correctly. The proposed method performs similarly to the computation of local space average color in areas where no depth continuities are present. The difference is largest in the vicinity of the depth discontinuities.

Thus we see that the depth map provides important information on locations in the image where local space average color should be computed uniformly in all directions. If a depth discontinuity is present, we better average only in a direction perpendicular to the depth discontinuity. Even though lines along which the illuminant is approximately constant can be estimated from the input image Ebner (2007b), a depth map makes it much easier to find directions along which the average should be taken.

IV. Aligning the Depth Map with the Color Image

The algorithm, as described above, is able to work with any dense depth map. For our experiments, we have used a Kinect sensor. The advantage of this sensor is its low cost. Unfortunately, the depth map provided by the Kinect is not perfectly aligned with the color image. Hence it has to be aligned by determining the extrinsic and intrinsic parameters of the two Kinect cameras.

The Kinect has originally been developed for the Xbox 360 video game console (Microsoft Corporation, 2011). Kofler (2011) is giving a detailed description of the Kinect. However, due to its low cost and excellent depth map, it has become popular in the scientific community. The Kinect can be used for motion tracking. Due to its built in microphone array, it can also be used for sound position tracking. The setup of the Kinect sensor is shown in Figure 5. The horizontal bar, which rests on a motorized tilt unit, contains: a RGB camera, a depth sensor, and a multi-array microphone. Figure 5(b) shows a sample image which has been taken with the Kinect's color camera. The depth map for this image is shown in Figure 5(c).

The Kinect sensor has been used in many different research projects. Newcombe et al (2011), performed dense surface mapping and tracking. Gabel et al (2012) have used it for full body gait analysis. An analysis of this sensor for computer vision applications is given by (Andersen et al, 2012).

The data of the Kinect sensor can be accessed through the libfreenect package[1]. This package provides also sample code which is available online. The color images have 8 bit per channel while the depth images have 11 bit per pixel. The depth map has to be aligned with the color image because the intrinsics and extrinsics of the two sensors differ. The color and the depth sensor are a small distance apart from each other and they do not necessarily point into the same direction. Also, the depth sensor only covers a smaller area compared to the color sensor.

We have aligned the depth map with the color image using stereo calibration (Kofler, 2011). The OpenKinect package only provides depth data with a nonlinear correspondence to distance (Andersen et al, 2012). We have used a formula given by Burrus[2] to transform the output of the Kinect depth map d_{raw} to a distance d in meters.

$$d = \frac{1.0}{-0.0030711016 \cdot d_{\text{raw}} + 3.3309495161} \tag{9}$$

Another problem that we face using the Kinect sensor is that even though depth information is estimated for most pixels, for some pixels a depth value could not be estimated due to occlusions. The Kinect sensor projects a laser grid which is viewed by an infrared camera. Since the projector is located to the left of the camera, certain areas may not be visible by the camera. This always happens to areas on the left hand side of a depth discontinuity. Hence, we iteratively fill in depth data from the left hand side. In the description of this filling in process, we will call pixels for which the Kinect sensor was able to estimate a depth value "a valid depth value" while all

[1] http://openkinect.org

[2] http://burrus.name/index.php/Research/KinectCalibration

(a) (b) (c)

Fig. 3: (a) Image with two illuminants and a depth discontinuity separating the two illuminants. The depth discontinuity has been manually highlighted in red. (b) Illuminant estimate. (c) Output image.

other pixels are called "invalid depth values". Before filling in, we remove isolated valid depth values surrounded by invalid depth values. These values are assumed to be outliers. Next, we iterate n_f times over the image. For each row of the depth map, we iterate over all pixels from left to right. If a pixel contains an invalid depth value, then we estimate the depth by interpolating depth values from the top, upper left, left, lower left and the bottom neighbors of this pixel. A weight of 1 is used for the values from the top, left, and from the bottom. A weight of $1/\sqrt{2}$ is used for the diagonal pixels. After this filling in process, we have a dense depth map which we need for our algorithm.

V. Experiments and Results

For our experiments, we have used an absolute depth threshold $\epsilon = 0.1$ for the entire image. We have used $\sigma = 0.25$ i.e. the area over which local space average color is computed is roughly 50% of the image. We have obtained images of size 640×480 from the Kinect. The color and the depth image have to be aligned as described above. After alignment, depth is undefined for some pixels at the border of the image. Therefore, we crop the images to size 572×433 to remove these border areas.

One of the main difficulties in testing the algorithm using the Kinect is to find suitable scenes to shoot. The Kinect offers a relatively small field of view, which is additionally constrained by the range of the depth sensor. It only provides usable depth information in the range from 0.8 to 3.5 meters (Andersen et al, 2012). At a distance of 2m, the depth resolution is only 1cm. Hence, it is not possible to shoot scenes which extend over a significant area. Also, the use of the infrared projection also limits the use in direct sunlight. The color image of the Kinect is very noisy if little light is available. We usually take the average of multiple images in order to reduce noise.

Figure 6 shows the results for 4 sample scenes. Images 1, 2 and 4 show photographs where a door separates two rooms which have been illuminated by different illuminants. Image 3 shows a single room where the background is illuminated by a blueish light whereas the front is illuminated by a yellowish light source. For all four images, depth discontinuities are very helpful in separating the areas illuminated by different illuminants from each other.

Comparison results for three other algorithms: the gray world assumption (GW), computation of local space average color (LSA), and the computation of anisotropic local space average color along iso-illumination lines (Iso-LSA) are shown in Figure 7, Figure 8, and Figure 9 respectively. Among these three algorithms, the computation of anisotropic local space average color along iso-illumination lines seems to perform best. However, the iso-illumination lines are always smooth. Therefore, the estimated illuminant is not quite accurate in the corner of the door frames. The proposed method which uses the depth discontinuities better separates the two illuminants. This is especially clear when comparing the proposed method with the computation of local space average color.

VI. Conclusion

For natural scenes, depth discontinuities are an important cue to separate areas with different illumination. We have extended the parallel algorithm based on local space average color to take depth discontinuities into account. The algorithm is based on a grid of processing elements modeling a resistive grid. The resistive grid is modified based on the distance of the corresponding object point to the observer. Only processing elements with a similar distance are resistively coupled. This allows us to estimate the illuminant separately for areas separated by a depth discontinuity. The proposed method can be used for robot vision systems for improved color perception. Whether the human visual system actually uses depth information in computing a color constant descriptor remains to be shown.

References

Andersen MR, Jensen T, Lisouski P, Mortensen AK, Hansen MK, Gregersen T, Ahrendt P (2012) Kinect depth sensor evaluation for computer vision applications. Tech. Rep. ECE-TR-6, Aarhus University, Denmark

Barnard K, Finlayson G, Funt B (1997) Color constancy for scenes with varying illumination. Computer Vision and Image Understanding 65(2):311–321

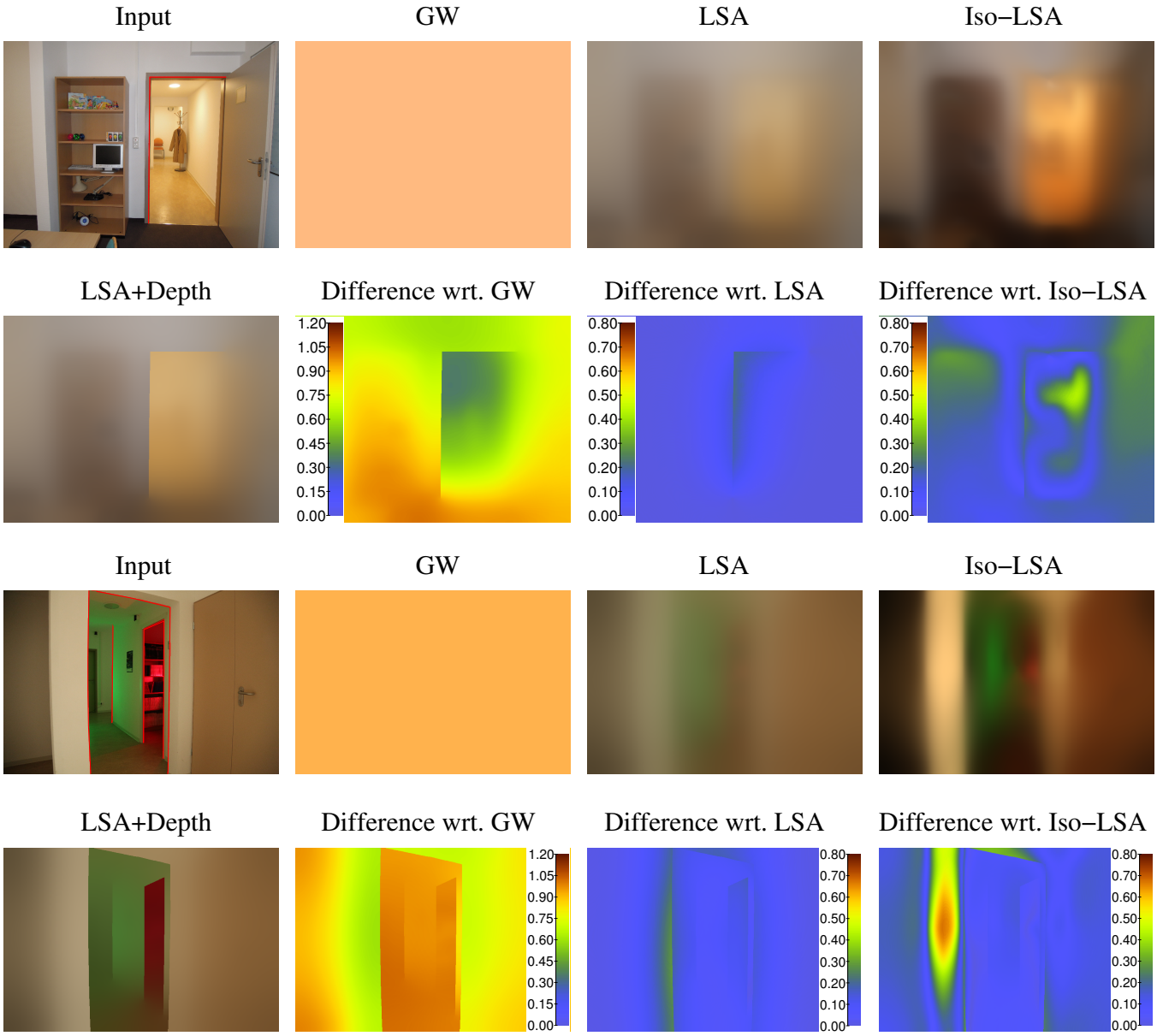


Fig. 4: Comparison of the proposed method for two sample images to three other color constancy algorithms: gray world assumption (GW), computation of local space average color (LSA), computation of anisotropic local space average color along iso-illumination lines (Iso-LSA). Depth discontinuities have been manually highlighted in red. The difference between the proposed method and the respective other color constancy algorithm in estimating the illuminant is also shown. Blue corresponds to almost no difference whereas red corresponds to large differences. Note the different scales for the three difference images.

Barnard K, Cardei V, Funt B (2002a) A comparison of computational color constancy algorithms – part I: Methodology and experiments with synthesized data. IEEE Transactions on Image Processing 11(9):972–984

Barnard K, Martin L, Coath A, Funt B (2002b) A comparison of computational color constancy algorithms – part II: Experiments with image data. IEEE Transactions on Image Processing 11(9):985–996

Blake A (1985) Boundary conditions for lightness computation in Mondrian world. Computer Vision, Graphics, and Image Processing 32:314–327

Brainard DH, Freeman WT (1997) Bayesian color constancy. Journal of the Optical Society of America A 14(7):1393–1411

Buchsbaum G (1980) A spatial processor model for object colour perception. Journal of the Franklin Institute 310(1):337–350

Cardei VC, Funt B (1999) Committee-based color constancy. In: Proceedings of the IS&T/SID Seventh Color Imaging Conference: Color Science, Systems and Applications, Scottsdale, Arizona, pp 311–313

Dartnall HJA, Bowmaker JK, Mollon JD (1983) Human visual

(a) (b) (c)

Fig. 5: (a) Kinect sensor. (b) RGB image (c) Depth map.

Fig. 6: Results for 4 sample images taken with the Kinect. From top to bottom, we have input image, depth map, detected edges, illuminant estimate and the computed output image.

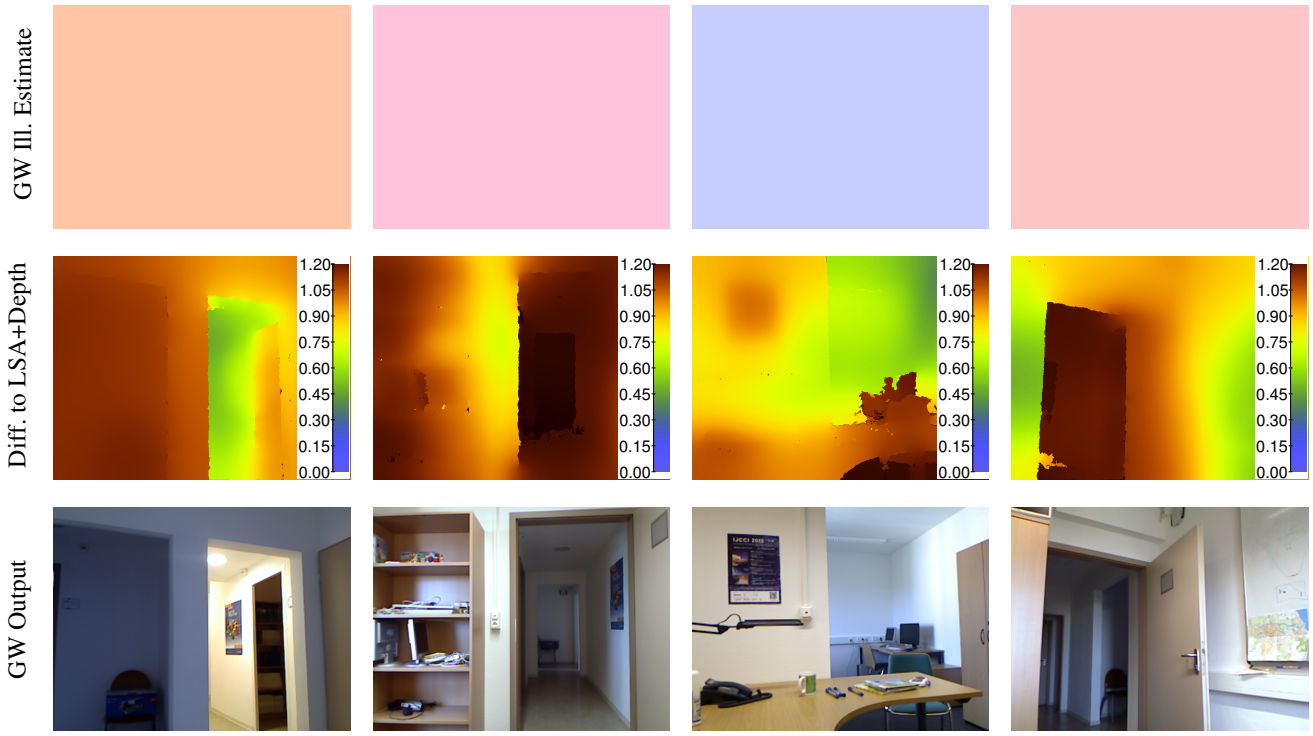


Fig. 7: Results for the gray world assumption (GW) for comparison with the proposed method. Top to bottom: estimate of the illuminant, difference when compared to the illuminant estimate obtained with the proposed method, and output image computed with the gray world assumption.

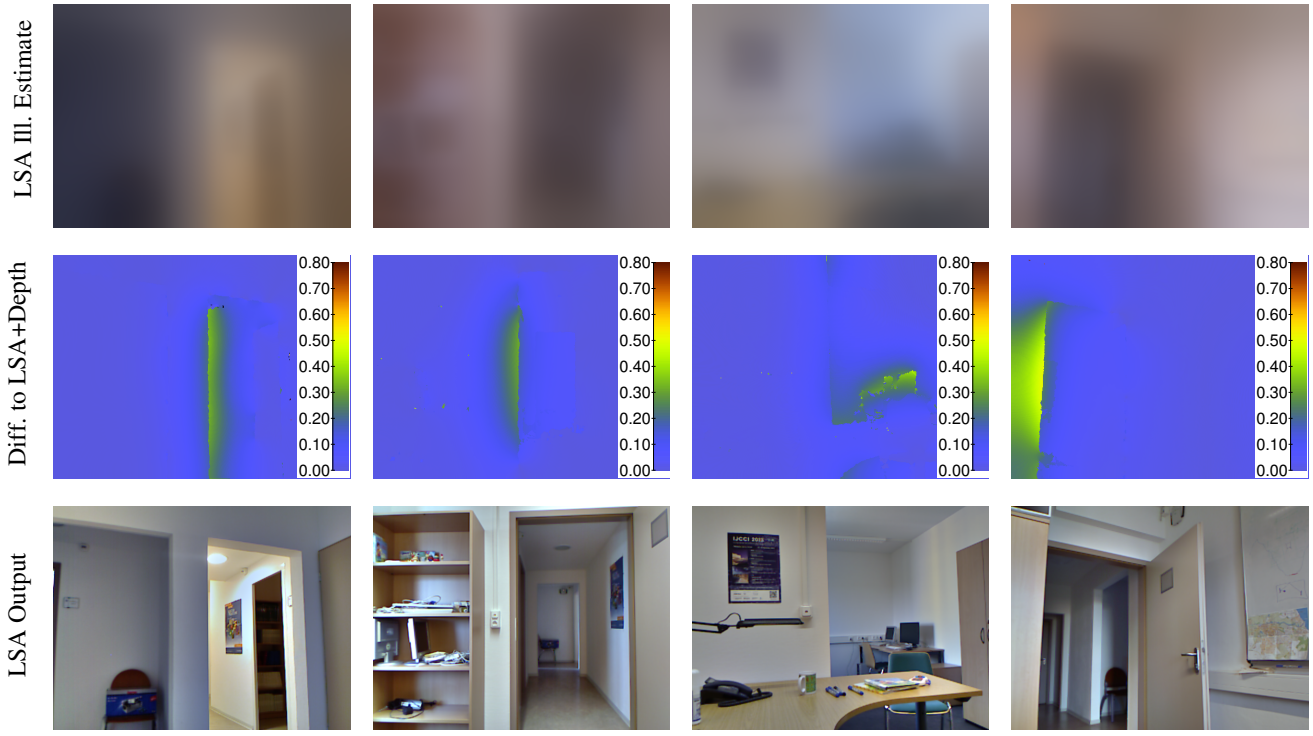


Fig. 8: Results for local space average color (LSA) for comparison with the proposed method. Top to bottom: estimate of the illuminant, difference when compared to the illuminant estimate obtained with the proposed method, and output image computed with local space average color.

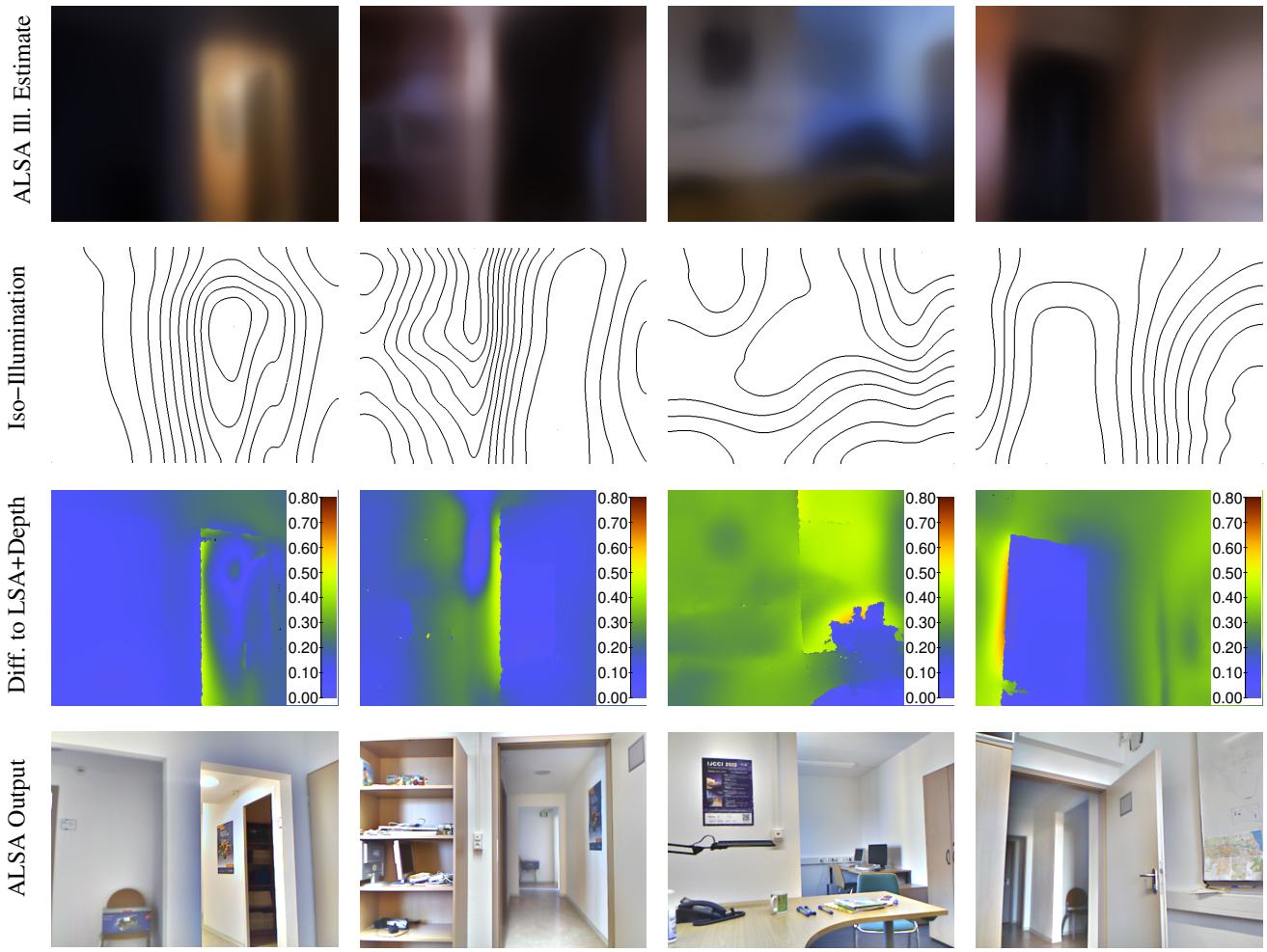


Fig. 9: Results for the computation of anisotropic local space average color along iso-illumination lines (Iso-LSA) for comparison with the proposed method. Top to bottom: estimate of the illuminant, iso-illumination lines, difference when compared to the illuminant estimate obtained with the proposed method, and output image computed with anisotropic local space average color.

pigments: microspectrophotometric results from the eyes of seven persons. Proc R Soc Lond B 220:115–130

Dowling JE (1987) The retina: an approachable part of the brain. The Belknap Press of Harvard University Press, Cambridge, Massachusetts

Ebner M (2004) A parallel algorithm for color constancy. Journal of Parallel and Distributed Computing 64(1):79–88

Ebner M (2007a) Color Constancy. John Wiley & Sons, England

Ebner M (2007b) Estimating the color of the illuminant using anisotropic diffusion. In: Kropatsch WG, Kampel M, Hanbury A (eds) Proceedings of the 12th International Conference on Computer Analysis of Images and Patterns, 27-29 August, 2007, Vienna, Austria, Springer-Verlag, Berlin, pp 441–449

Ebner M (2007c) How does the brain arrive at a color constant descriptor? In: Mele F, Ramella G, Santillo S, Ventriglia F (eds) Proceedings of the 2nd International Symposium on Brain, Vision and Artificial Intelligence, 10-12 October, 2007, Naples, Italy, Springer, Berlin, pp 84–93

Ebner M (2009) Color constancy based on local space average color. Machine Vision and Applications Journal 20(5):283–301

Ebner M (2012) A computational model for color perception. Bio-Algorithms and Med-Systems 8(4):387–415

Finlayson GD, Hordley SD (2001) Color constancy at a pixel. Journal of the Optical Society of America A 18(2):253–264

Finlayson GD, Hordley S, Pubel PM (2001) Color by correlation: A simple, unifying framework for color constancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(11):1209–1221

Forsyth DA (1990) A novel algorithm for color constancy. International Journal of Computer Vision 5(1):5–36

Frankle JA, McCann JJ (1983) Method and apparatus for lightness imaging. United States Patent No 4,384,336

Funt B, Cardei V, Barnard K (1996) Learning color constancy. In: Proceedings of the IS&T/SID Fourth Color Imaging Conference, Scottsdale, pp 58–60

Funt B, Barnard K, Martin L (1998) Is machine colour constancy good enough? In: Burkhardt H, Neumann B (eds)

Fifth European Conference on Computer Vision (ECCV '98), Freiburg, Germany, Springer-Verlag, Berlin, pp 445–459

Funt B, Ciurea F, McCann J (2004) Retinex in MATLAB. Journal of Electronic Imaging 13(1):48–57

Funt BV, Drew MS, Ho J (1991) Color constancy from mutual reflection. International Journal of Computer Vision 6(1):5–24

Gabel M, Gilad-Bachrach R, Renshaw E, Schuster A (2012) Full body gait analysis with Kinect. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Diego, CA, IEEE, pp 1964–1967

Gegenfurtner KR (2003) Cortical mechanisms of colour vision. Nature Reviews Neuroscience 4:563–572

Geusebroek JM, van den Boomgaard R, Smeulders AWM, Geerts H (2001) Color invariance. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(12):1338–1350

Gijsenij A, Lu R, Gevers T (2012) Color constancy for multiple light sources. IEEE Transactions on Image Processing 21(2):697–707

Herault J (1996) A model of colour processing in the retina of vertebrates: From photoreceptors to colour opposition and colour constancy phenomena. Neurocomputing 12:113–129

Horn BKP (1974) Determining lightness from an image. Computer Graphics and Image Processing 3:277–299

Horn BKP (1986) Robot Vision. The MIT Press, Cambridge, Massachusetts

Hurlbert AC, Poggio TA (1988) Synthesizing a color algorithm from examples. Science 239:482–483

Jain R, Kasturi R, Schunck BG (1995) Machine Vision. McGraw-Hill, Inc., New York

Kofler M (2011) Inbetriebahme und untersuchung des Kinect sensors. Master's thesis, FH Oberösterreich, Österreich

Land EH (1986) An alternative technique for the computation of the designator in the retinex theory of color vision. Proc Natl Acad Sci USA 83:3078–3080

Land EH, McCann JJ (1971) Lightness and retinex theory. Journal of the Optical Society of America 61(1):1–11

Livingstone MS, Hubel DH (1984) Anatomy and physiology of a color system in the primate visual cortex. The Journal of Neuroscience 4(1):309–356

Lu R, Gijsenij A, Gevers T, Nedović V, Xu D, Geusebroek JM (2009) Color constancy using 3d scene geometry. In: Proceedings of the 12th IEEE International Conference on Computer Vision, Kyoto, Japan, pp 1749–1756

McCann JJ (2000) Simultaneous contrast and color constancy: Signatures of human image processing. In: Davis S (ed) Color Perception: Philosophical, Psychological, Artistic, and Computational Perspectives. Volume 9. Vancouver Studies in Cognitive Science, Oxford University Press, pp 87–101

McCann JJ, McKee SP, Taylor TH (1976) Quantitative studies in retinex theory. Vision Res 16:445–458

Microsoft Corporation (2011) Programming with the Kinect for Windows SDK

Moore A, Allman J, Goodman RM (1991) A real-time neural system for color constancy. IEEE Transactions on Neural Networks 2(2):237–247

Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, Kohli P, Shotton J, Hodges S, Fitzgibbon A (2011) Kinectfusion: Real-time dense surface mapping and tracking. In: Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality, IEEE, pp 127–136

Szeliski R (2010) Computer Vision. Springer, Berlin

Tovée MJ (1996) An introduction to the visual system. Cambridge University Press, Cambridge

van de Weijer J, Gevers T, Gijsenij A (2007) Edge-based color constancy. IEEE Transactions on Image Processing 16(9):2207–2214

Zeki S (1993) A Vision of the Brain. Blackwell Science, Oxford