Noname manuscript No.
(will be inserted by the editor)

On the effect of scene motion on color constancy

Marc Ebner

the date of receipt and acceptance should be inserted later

Abstract A series of experiments with human subjects have shown that color constancy improves when an object moves. It has been hypothesized that this effect is due to some kind of influence of high-level motion processing. We have built a computational model for color perception which replicates the results qualitatively which have been obtained with human subjects. We show that input from high-level motion processing is not required. In our model, the dependence is an effect of eye movement in combination with neural processing. Depending on the type of stimulus used, the eye either tracks the object or the background. When the object moves but is tracked by the observer, the background appears to move when considering the stimulus with respect to eye coordinates. Hence, the retinal input is different for the two conditions leading to a difference in color constancy performance.

Keywords Color Constancy, Color Perception, Computational Modelling, Object Motion

1 Motivation

Recently, Werner (2007) has shown, that for certain stimuli, color constancy improves, when an observed object moves. Werner compared color constancy under a series of different conditions (a) static scene: object in-front of background (b) motion parallax: object remains stationary while background moves (c) global motion: object and background move together as if they were connected (d) object motion: an object moves across a static background. In addition to these four stimuli, Werner also used a 5th stimulus: (e) ambiguous motion: where an object moves across a background consisting of horizontal stripes. Stimulus (e) is referred to as ambiguous motion because it cannot be discerned whether the background moves horizontally or not. Hence, this motion stimulus is ambiguous. Since the results obtained for the ambiguous motion experiments were not consistent across subjects, we do not consider this type of stimulus here.

Werner has shown that color constancy improves significantly for stimulus (d) compared to (a). Werner has used a black and white checkerboard pattern as background and a gray patch as a test patch. This artificial scene was virtually illuminated using either red or green light. Werner concluded from the experiments that high-level motion influences color perception. We now provide an alternative explanation for the observed results. We show how the results (obtained by Werner with human subjects) can be replicated in simulation using a computational model developed by Ebner (2007a,b). His model has already been shown to be in line with experimental results obtained by Helson (1938). We were able to reproduce the experimental results obtained by Werner (2007) using Ebner's model of color perception in conjunction with eye movements. We show that the results can be explained through purely bottom up processing. No interaction with motion areas (neither low- nor high-level areas) is required. Anatomical studies have shown neural connections between motion processing areas in the dorsal pathway and color processing areas in the ventral pathways (Felleman and Essen, 1991). In light of our results it seems

M. Ebner
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
E-mail: marc.ebner@uni-greifswald.de
http://www.math-inf.uni-greifswald.de/mathe/index.php/mitarbeiter/608-prof-marc-ebner

more likely that such connections are used to combine motion and color such that both can be used for tracking and/or figure/ground separation.

2 Color Image Formation

Before we can address the experimental results using computational processing, we first need a model of color image formation. Processing of visual information of course starts with light entering the eye where it is measured by the receptors of the retina. In order to perceive anything, at least one light source is required. Light from one or more light sources reaches an object patch. Some of the light is absorbed by the patch, the remainder is reflected. Some of the reflected light enters the eye where it is measured. Let $I(\mathbf{x}_r)$ be the energy measured by a retinal receptor at position $\mathbf{x}_r$ on the retina. The light entering point $\mathbf{x}_r$ on the retina originated from some object patch located at position $\mathbf{x}_o$. Let $L(\mathbf{x}_o, \lambda)$ be the irradiance falling onto object patch at position $\mathbf{x}_o$ at wavelength λ. Let $R(\mathbf{x}_o, \lambda)$ be the reflectance at object point $\mathbf{x}_o$ at wavelength λ. Let $S(\lambda)$ be the sensitivity of the retinal sensor. If we assume that the object patch reflects the incident light diffusely, i.e. uniformly, then we obtain the following expression for the measured energy $I(\mathbf{I}_r)$.

$$I(\mathbf{x}_r) \propto \int S(\lambda) R(\mathbf{x}_o, \lambda) L(\mathbf{x}_o, \lambda) d\lambda \tag{1}$$

The measured energy is also proportional to a geometry factor which takes into account the foreshortening of an object patch. However, this factor can be subsumed into a combined shading/reflectance factor. Thus, there is no need to consider this geometry factor here.

We now drop the coordinates $\mathbf{x}_r$ and $\mathbf{x}_o$ and switch to a two-dimensional coordinate system (x, y). We can do this because there is a one-to-one correspondence between points on the retina and points on the objects viewed. Thus, we obtain

$$I(x, y) \propto \int S(\lambda) R(x, y, \lambda) L(x, y, \lambda) d\lambda. \tag{2}$$

Three types of cones can be distinguished which respond to light in the red, green, and blue parts of the spectrum (Brown and Wald, 1964; Marks et al, 1964). Let $S_i(\lambda)$ be the sensitivity of cone type i with $i \in \{r, g, b\}$. We now assume that the sensors are very narrow band, i.e. that they have the shape of delta functions. Using $S_i(\lambda) = \delta(\lambda - \lambda_i)$, we obtain

$$I_i(x, y) \propto \int \delta(\lambda - \lambda_i) R(x, y, \lambda) L(x, y, \lambda) d\lambda. \tag{3}$$

This gives us

$$I_i(x, y) \propto R(x, y, \lambda_i) L(x, y, \lambda_i). \tag{4}$$

It is of course clear that the retinal receptors do not respond only to a a single wavelength. However, the above relationship will help us to develop a mathematical theory of how the visual system processes information in order to arrive at an approximate color constant descriptor. In the following, we will refer to $\mathbf{I}(x, y) = [I_r(x, y), I_g(x, y), I_b(x, y)]$ as the measured color, i.e. the data measured by the system. Using this data, a color constant or rather approximately color constant descriptor is computed. When performing computations, we assume that arithmetic operators operate on the individual components of this vector, i.e. we will write $a \cdot b$ as a short hand notation for

$$[a_r b_r, a_g b_g, a_b b_b]. \tag{5}$$

3 The gray world assumption

Numerous computational approaches to color constancy have been developed. The most well known approaches are probably the white-patch Retinex algorithm (Funt et al, 1998) and the gray-world-assumption (Buchsbaum, 1980). Both algorithms assume that the illumination of the scene is uniform. The white-patch Retinex algorithm is a simplified version of Land's Retinex algorithm (Land, 1964, 1974, 1986). This algorithm simply takes the maximum response of each color channel and divides the measured color by this maximum response. The maximum response is assumed to originate from a white patch. Therefore, this response is proportional to the illuminant, i.e. if $R_i(x, y) = 1$ then $I_i(x, y) \propto L_i(x, y)$.

Buchsbaum (1980) has proposed the gray world assumption. According to Buchsbaum, the world is gray on average. And indeed, if we assume that a large number of different colors are contained in the image and we also assume that all colors are equally likely, then the spatial average over all image pixels is proportional to the illuminant. Let n be the number of image pixels, then we obtain (Ebner, 2004)

$$\mathbf{L} \propto \frac{1}{n} \sum_{x,y} \mathbf{I}(x, y) \tag{6}$$

as an estimate of the illuminant $\mathbf{L}$. Ebner (2009) has shown that the gray world assumption can also be applied locally. We only need to convolve the input image using an extensive smoothing kernel $G(x', y')$. This allows us to estimate the illuminant $\mathbf{L}(x, y)$ locally for

each image or retinal point.

$$\mathbf{L}_e(x,y) \propto \int\int G(x',y')\mathbf{I}(x',y')dx'dy' \tag{7}$$

Once we have a local estimate of the illuminant, we can compute a color constant descriptor $\mathbf{o}_{\text{cc}}$ which is independent of the illuminant by dividing the measured color $\mathbf{I}(x,y)$ by the estimate of the illuminant $\mathbf{L}_e(x,y)$. We obtain

$$\mathbf{o}_{\text{cc}}(x,y) = \frac{\mathbf{I}}{\mathbf{L}_e(x,y)} \tag{8}$$

With $\mathbf{L}_e(x,y) \propto \mathbf{L}(x,y)$ and our derivation from above, we obtain

$$\mathbf{o}_{\text{cc}}(x,y) = \frac{\mathbf{R}(x,y)\mathbf{L}(x,y)}{\mathbf{L}(x,y)} = \mathbf{R}(x,y) \tag{9}$$

which is independent of the illuminant. Note that here, $\mathbf{R}$ is a product of shading and reflectance.

4 A computational model of color perception

We now describe Ebner's computational model for color perception (Ebner, 2007a,b). Processing of course starts with the receptors of the retina. The relationship between the output of the retinal cones and the measured light could be either logarithmic (as proposed by Faugeras (1979)) or follow a cube root or a square root (see Hunt (1957)). All of these functions can be made quite similar to each other on a given range with a proper choice of parameters. Suppose that the response follows a logarithmic function. Thus, the output of the retinal receptors transforms the measured product of reflectance times illumination into a sum of logarithms:

$$\mathbf{o}_{\text{retina}}(x,y) = \log \mathbf{I}(x,y) \tag{10}$$

$$= \log \mathbf{R}(x,y)\mathbf{L}(x,y) \tag{11}$$

$$= \log \mathbf{R}(x,y) + \log \mathbf{L}(x,y) \tag{12}$$

The information then reaches V1 through connections which go through the lateral geniculate nucleus and eventually reach V4. V4 is assumed to be the location where a color constant descriptor is computed (Zeki, 1993; Zeki and Bartels, 1999; Zeki and Marini, 1998). The function of the color opponent cells can be viewed as performing a rotation of the coordinate system. The retinal coordinate system has the three axes red, green, and blue. After the coordinate system is rotated, processing continues in a rotated coordinate system where the axes are dark-bright, red-green, and yellow-blue. For our mathematical treatment, we omit this rotation as it does not matter whether the coordinate system is rotated or not.

Ebner (2007a,b) assumes that a grid of resistively coupled neurons in V4 computes local space average color. Adjacent neurons are presumably coupled through gap junctions which are known to behave as resistors (Herault, 1996). These neurons form a resistive grid, which performs extensive smoothing of the input. Each neuron of this resistive grid is used to compute local space average color. Let $\mathbf{a}(x,y)$ be local space average color estimated by a neuron which processes data from position (x,y) on the retina. Each neuron is resistively coupled to a number of neighboring neurons. Let $N(x,y)$ be the set of neurons which are resistively coupled to neuron (x,y), i.e.

$$N(x,y) = \{(x',y') | (x',y') \text{ is neighbor of neuron } (x,y)\}. \tag{13}$$

Each neuron computes local space average color by averaging local space average color which has already been estimated by neighboring elements. A small amount of the measured color $\mathbf{o}_{\text{retina}}$ is then added to this intermediate result. Note that here, $\mathbf{o}_{\text{retina}}$ is the output of the retinal sensors using the red, green, blue coordinate system but in the brain $\mathbf{o}_{\text{retina}}$ corresponds to the rotated coordinate system. As described above, this rotation can be omitted in the mathematical treatment. In summary, the function computed by the resistive grid of neurons can be described by the following two update equations

$$\mathbf{a}'(x,y) := \frac{1}{|N(x,y)|} \sum_{(x',y')\in N(x,y)} \mathbf{a}(x',y') \tag{14}$$

$$\mathbf{a}(x,y) := \mathbf{o}_{\text{retina}}(x,y)\cdot p + \mathbf{a}'(x,y)\cdot(1-p) \tag{15}$$

where p is a small value larger than zero. These update equations are executed indefinitely by the resistive grid in the brain. Mathematically, this process converges to local space average color $\mathbf{a}(x,y)$ which is a local estimate of the illuminant. Quite a large number of iterations are required (ca. 50000) before convergence (Ebner, 2007a). However, over-relaxation (Bronstein et al, 2001) can be used to speed up this process (only approximately 8000 iterations are required). The parameter p defines the extent over which local average color will be computed. If p is very small, local space average color will be computed over a relatively large area. If p is relatively large, local space average color will be computed over a small area.

The algorithm does not have to be run until convergence in order to use its output. An estimate of local space average color is available at any point in time. The human visual system is assumed to use this estimate to compute a color constant descriptor. It is precisely for this behavior that motion has an influence on

the perceived color of an object. This is purely bottom up behavior without any input from high-level motion areas.

Local space average color $\mathbf{a}$ is used to compute a color constant descriptor by subtracting local space average color from the measured color. Thus, the color constant descriptor $\mathbf{o}_{\text{cc}}$ describing the output color will be given as

$$\mathbf{o}_{\text{cc}}(x,y) = \mathbf{o}_{\text{retina}}(x,y) - \mathbf{a}(x,y). \tag{16}$$

In order to understand that this is a color constant descriptor, we will assume that $\mathbf{a}(x,y)$ is equivalent to global space average color, and that the scene is uniformly illuminated. Let n be the number of pixels in the image, then we obtain (Ebner, 2007a)

$$a_i(a,x) = \frac{1}{n}\sum_{x,y} o_{i,\text{retina}}(x,y) \tag{17}$$

$$= \frac{1}{n}\sum_{x,y} (\log R_i(x,y) + \log L_i) \tag{18}$$

$$= \log \mathbf{L} + \frac{1}{n}\sum_{x,y} \log R_i(x,y). \tag{19}$$

for a single color channel i. Let us now assume that the scene contains a large number of different colored patches. We do not know which colors actually occur in the scene. Therefore, we will assume a uniform distribution of all colors, i.e. reflectances. Thus, we obtain

$$a_i(x,y) = \log \mathbf{L} + \frac{1}{n}\sum_{j=1}^{n} \log\left(\frac{j}{n}\right) \tag{20}$$

$$= \log \mathbf{L} + \frac{1}{n}\log\left(\frac{n!}{n^n}\right) \tag{21}$$

$$= \log \mathbf{L} + \log\left(\frac{n!^{\frac{1}{n}}}{n}\right). \tag{22}$$

This allows us to estimate the last term by a constant using Stirling's formula

$$\lim_{n\to\infty} \frac{(n!)^{\frac{1}{n}}}{n} = \frac{1}{e}, \tag{23}$$

provided that we have a sufficiently large number of pixels. We now obtain

$$a_i(x,y) = \log L_i - 1. \tag{24}$$

Therefore, local space average color is an estimate of the logarithm of the local illuminant except for a constant offset. The color constant descriptor is thus given as

$$o_{i,\text{cc}}(x,y) = o_{i,\text{retina}}(x,y) - a_i(x,y) \tag{25}$$

$$= \log L_i(x,y) + \log R_i(x,y) \tag{26}$$

$$- \log L_i(x,y) + 1 \tag{27}$$

$$= \log R_i(x,y) + 1. \tag{28}$$

5 Stimuli and Results

We tested a variant of this computational model on the same set of experiments which were conducted by Werner (2007). Figure 1 illustrates the input stimuli observed by human subjects for experiments (a) through (d). Experiment (a) uses a static scene. The test patch is shown in front of an achromatic checkerboard pattern (background). The remaining experiments use continuous motion of either the test patch, the background or both. Only a single snapshot is shown for experiment (a) the static scene. Figure 1(b) shows the motion parallax sequence. The background moves behind the test patch. The test patch remains stationary. Figure 1(c) shows global motion. Test patch and background move together in the same direction. Figure 1(d) shows the object motion sequence. The test patch moves across a static background. When the test patch leaves the screen on the left, it reenters the screen on the right. Subjects have to adjust the illuminant until the test patch appears achromatic to them. They have to do this for a standard D65 illuminant (see e.g. Ebner (2007a) for a description of this illuminant) and also for two test illuminants (red and green). The extent of the necessary adjustment is used as a measure of color constancy.

Local motion seems to be important for tracking moving objects whereas global motion processing seems to be important for gaze stabilization (Lindner et al, 2001; Ilg, 1997). The movement of the eyeball for the four stimuli is shown in Figure 2. The measurements were obtained using a standard video camera (Sony HDR-TG3E, video resolution 960×540 with 25 frames per second). Head movements were not prohibited during the experiment. Head movements were later removed using image stabilization. The video sequence was transformed into a stream of images. The image sequence was then stabilized using patch matching with sub-pixel accuracy such that the eye appears stationary in the image sequence. We then extracted the color of the pupil and computed the distance of every pixel to the color of the pupil. The distance image pixels were transformed to the range [0,1] using linear scaling such that 1 corresponds to the eye color. Distances below 0.9 were clipped and the result again transformed to the range $[0,1]$ using linear scaling. Next, a morphological closing operation with a structure element of size 5×5 was applied, followed by a morphological opening using a structure element of size 10×10. Finally, Gaussian smoothing was applied with a standard deviation of 60. The center of mass was taken as the position of the pupil. The horizontal eye position was smoothed using a median filter with a window size of 7. Finally, the

Table 1 Computation of eye movements based on stimulus.

Stimulus	Eye movement
a)	Eye fixates center of the patch
b)	Eye first looks at the center of the patch but then tracks the background. Whenever the background has moved by a distance equal to twice the patch size, then the eye performs a saccade moving the focal point of the eye by a distance equal to twice the patch size in a direction opposite to the movement of the background.
c)	Eye fixates center of the patch
d)	Eye fixates center of the patch

eye orientation was computed based on the position, size of the pupil and the distance to the camera.

For stimulus (a) static stimulus, the eye shows only small movements. For stimulus (b) motion parallax, the eye shows more extensive movements. For stimuli (c) and (d) the eye performs a smooth pursuit tracking motion.

We assume that the eyes track important structures in the image. Since the subjects are queried on the color of the test patch, it is assumed that the eyes position the test patch directly in the center of the field of view. Table 1 explains how the movement of the eye is computed. Figure 3 shows the same stimuli as described above but viewed from the point of view of the retinal receptors. The eyes essentially track the test patch maintaining it exactly in the fovea of the retina in (a) and (c-d). In Figure 3(a), the entire scene is stationary. Thus, this experiment is equivalent to Figure 1(a). In Figure 3(b), the subject is assumed to track the background with periodic saccades to maintain the test patch inside the fovea. As a result of this tracking, black stripes occur on the border of the image. For the case of global motion shown in Figure 3(c), the object is tracked, maintaining it exactly in the center of the fovea. Because object and background move together, the perceived image will be almost identical to experiment (a) except for the black stripes on the left or right side of the retina. The stripes occur whenever the object leaves the scene at the left and re-enters at the right. For experiment (d), as shown in Figure 3(d), the background appears to move behind the object because the object is tracked by the observer.

The experimental results obtained by Werner are shown in Figure 4(a). Three human subjects were exposed to the stimuli shown in Figure 1. One subject was informed while the other two were naive subjects. Werner uses a constancy index (CI) to describe the performance of the subjects. In Werner's experiments, the subjects have to adjust the illuminant of the input stimuli until the test patch is perceived as achromatic. This adjusted color of the stimulus is obtained once for the illuminant D65 and once for another illuminant (red or green). The Euclidean distance in Luv color space between these two adjusted colors is divided by the Euclidean distance between the colors of the two illuminants D65 and the non-standard illuminant. This is the color constancy index CI.

6 Methods

It is possible to replicate the results obtained by Werner using the following computational algorithm. Figure 4(b) shows the results from our computational model. Note that the error measure that we compute for the computational model is computed from internal data of the model. Hence, for our model color constancy is best when the error is smallest. Werner's color constancy performance is best the larger the CI measure is. Figure 2 shows the parameters used. The same parameters were used by Werner (2007). The algorithm was terminated after 8470 iterations using $p = 0.001705$. The stimuli had image size 242×176 pixels. The parameter p was set such that local space average color is computed using a kernel radius which extends across 10% of the image.

The computational algorithm computes color constant descriptors $\mathbf{o}_{\mathrm{cc}}(x, y)$ similar to the theoretical derivation as described above. Processing starts with retinal input $\mathbf{o}_{\mathrm{retina}}(x, y)$ and a processing matrix with one element per image pixel. The stimuli falling onto the retina is exactly as shown in Figure 3. Color stimuli were converted from Luv color space to sRGB as shown in Figure 2. These will be used as reflectances $\mathbf{R}$ and illuminant $\mathbf{L}$. The three channels RGB correspond to the retinal receptors which absorb light in the red, green and blue parts of the spectrum. The receptors measure the reflectances $\mathbf{R}$ scaled by the illuminant $\mathbf{L}$. Thus, we obtain for the light $\mathbf{c}(x, y)$ entering the retina

$$\mathbf{c}(x, y) \propto \mathbf{R}(x, y)\mathbf{L}(x, y) \tag{29}$$

The eye adapts to the amount of available light, hence we scale all channels by the maximum value m with $m = \max_{i,x,y}\{c_i(x, y)\}$.

$$\mathbf{I}(x, y) = \frac{\mathbf{c}(x, y)}{m} \tag{30}$$

We will now drop the index (x, y) when it is clear from context.

The exposure of the retinal receptors is modeled as

$$\mathbf{o}_{\mathrm{exp}} = (1 - p_e)\mathbf{o}_{\mathrm{exp}} + p_e\mathbf{I} \tag{31}$$

with $p_e = 0.8$. The retinal receptors appear to either have a logarithmic response function as proposed by

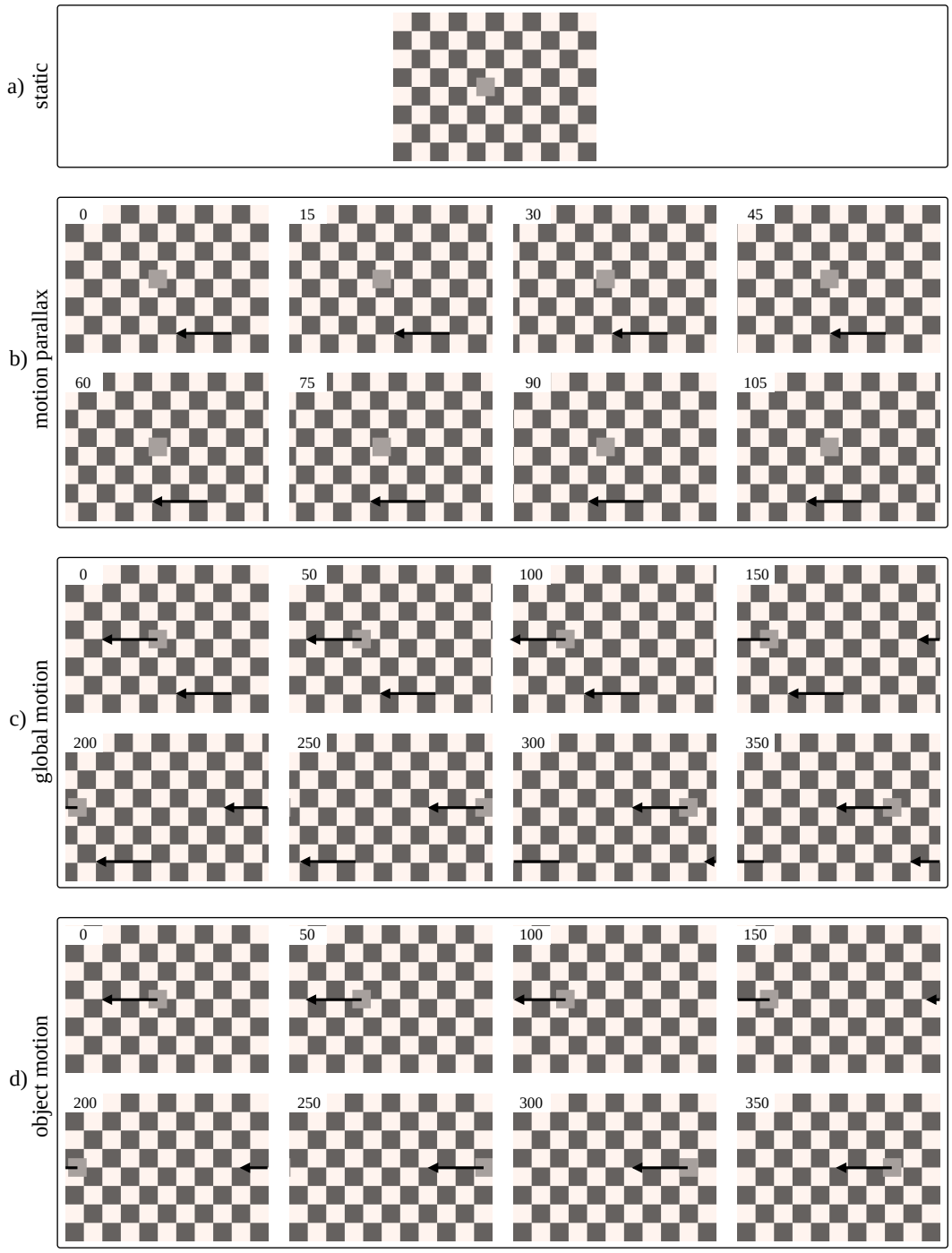


Fig. 1 Visual stimulus for experiments (a) - (d) as presented on the screen.

Faugeras (1979) or the response function follows a cube root or a square root (see Hunt (1957)). We use a cube root response function because the cube root relationship is also used for the CIE $L^*u^*v^*$ color space (International Commission on Illumination, 1996). As detailed in Ebner et al (2007), both the cube root law and a logarithmic response can be made similar to each other with a proper choice of parameters. Thus we have,

$$\mathbf{o}_{\text{retina}} = \mathbf{o}_{\text{exp}}^{1/3}. \tag{32}$$

Presumably inside V4, local space average color is computed. Let $N(x, y)$ be the neighborhood as described above, then local space average color $\mathbf{a}(x, y)$ is com-

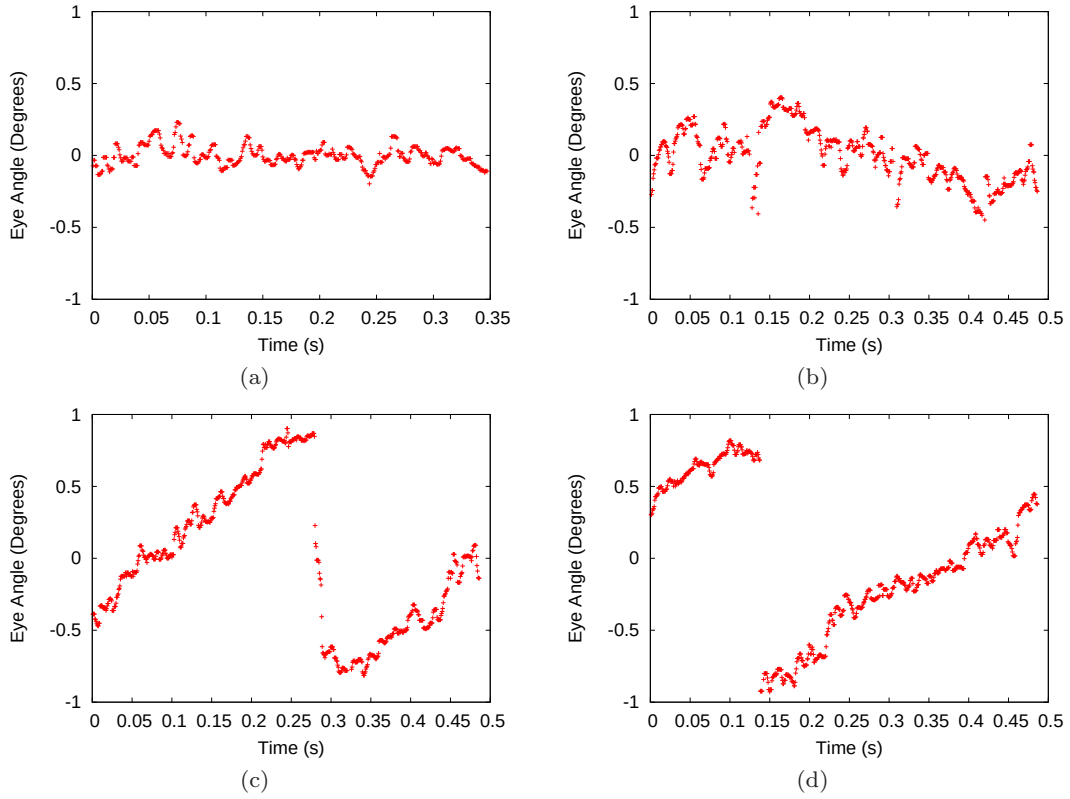


Fig. 2 Movement of the eyeball for stimuli (a-d). (a): the eyeball moves only slightly (b): the motion of the eyeball is larger (c): smooth pursuit motion from right to left (d): smooth pursuit motion from right to left

Table 2 Parameters for computational experiments.

Color Patch	Color ($L^*u^*v^*$)	RGB
red	$[40.40, 0.23, 0.46]$	$[0.1767, 0.0958, 0.1231]$
green	$[40.40, 0.17, 0.47]$	$[0.0627, 0.1303, 0.1168]$
blue	$[50.60, 0.20, 0.44]$	$[0.1970, 0.1791, 0.2677]$
yellow	$[50.60, 0.19, 0.50]$	$[0.1660, 0.2038, 0.1134]$
achromatic test patch	$[50.60, 0.20, 0.47]$	$[0.1959, 0.1878, 0.1843]$
bright achromatic patch	$[60.60, 0.20, 0.47]$	$[0.2981, 0.2857, 0.2804]$
dark achromatic patch	$[40.40, 0.20, 0.47]$	$[0.1190, 0.1141, 0.1119]$
red illuminant	$[40.40, 0.23, 0.46]$	$[0.1767, 0.0958, 0.1231]$
green illuminant	$[40.40, 0.17, 0.47]$	$[0.0627, 0.1303, 0.1168]$

puted using

$$\mathbf{a}'(x,y) := \frac{1}{|N(x,y)|} \sum_{(x',y')\in N(x,y)} \mathbf{a}(x',y') \tag{33}$$

$$\mathbf{a}(x,y) := \mathbf{o}_{\text{retina}}(x,y) \cdot p_a + \mathbf{a}'(x,y) \cdot (1-p_a). \tag{34}$$

We have used $p_a = 0.001705$ as described above. If no retinal input exists, i.e. if $\mathbf{o}_{\text{retina}} = [0,0,0]$, then only averaging occurs using:

$$\mathbf{a}(x,y) := \frac{1}{|N(x,y)|} \sum_{(x',y')\in N(x,y)} \mathbf{a}(x',y'). \tag{35}$$

A color constant descriptor is computed by essentially subtracting local space average color from the measured color. Since this process involves additional neural processing, we have included a temporal averaging of local space average color. Let $\tilde{a}(x,y)$ be the current temporal average, then the temporal average is updated using

$$\tilde{\mathbf{a}}(x,y) := p_t \mathbf{a}(x,y) + (1-p_t)\tilde{\mathbf{a}}(x,y) \tag{36}$$

with $p_t = 0.1$.

The color constant descriptor is then computed using

$$\mathbf{o}_{\text{cc}}(x,y) := \mathbf{o}_{\text{retina}} - \tilde{\mathbf{a}}(x+d_x, y+d_y) \tag{37}$$

with $d_x = 6$ and $d_y = 6$, i.e. temporal average of local space average color is slightly offset compared to the measured color. We describe below why this offset

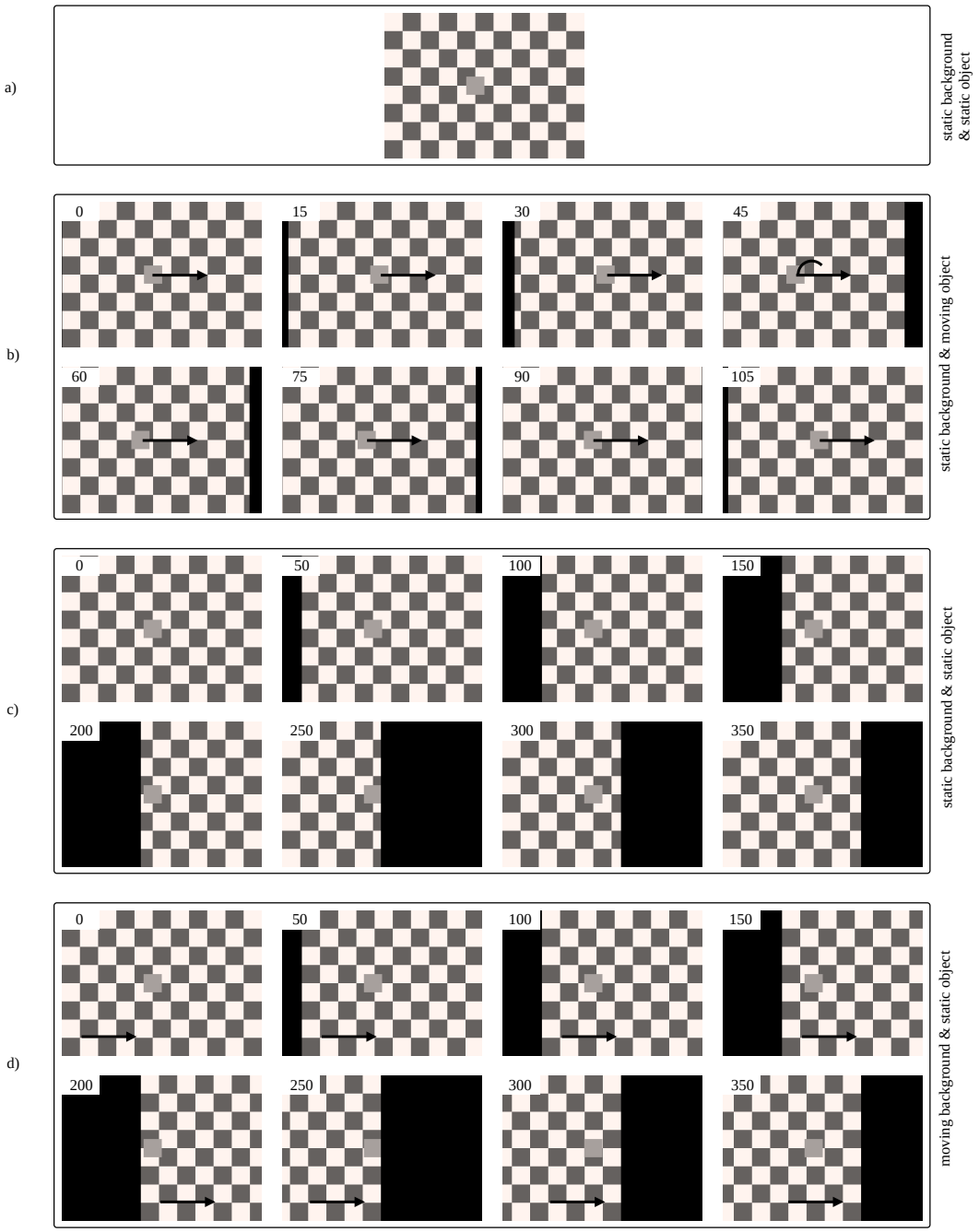


Fig. 3 Visual stimulus for experiments (a) - (d) as measured by the retinal receptors.

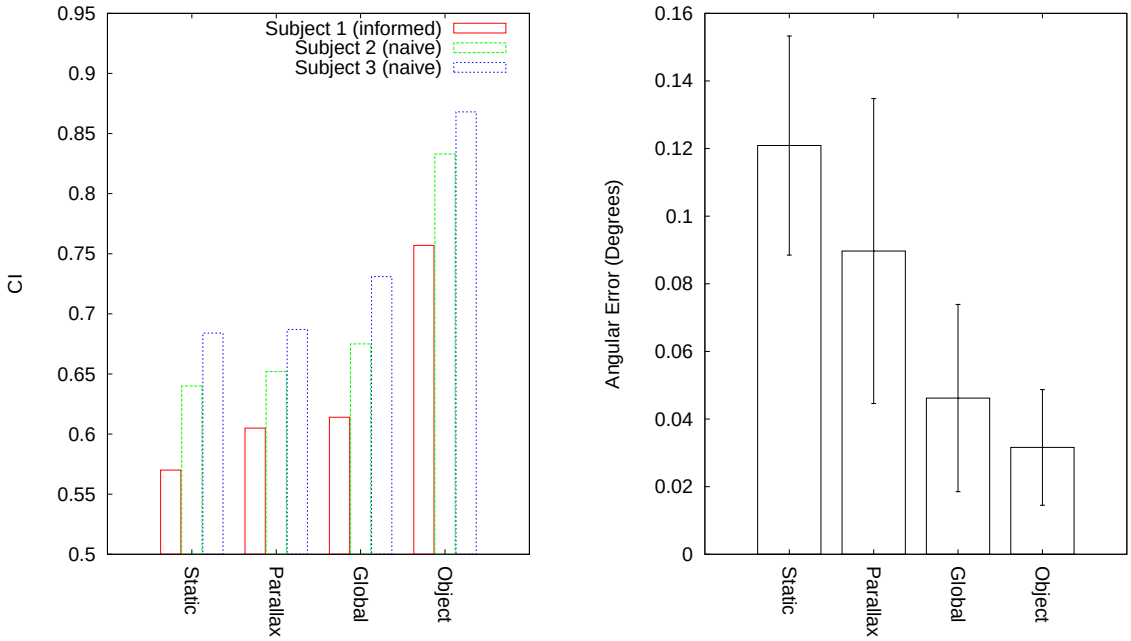
is necessary. Alternatively, we could also envision that the computed temporal average or local space average color is shuffled to a certain extent, i.e. the correspondence between the measured color and local space average color is not perfect. This is quite likely given that the brain is a developmental system. Indeed, a local disordering of image data may have certain advantages (Koenderink and van Doorn, 1999, 2000).

To evaluate the computational model, we compute reflectance estimates. As in the gray world assumption, we assume that reflectances are uniformly distributed in the image. Therefore, we add a constant offset k with

$$k = \left(\sum_{i=0}^{n} \frac{i}{n} \right)^{1/3} \tag{38}$$

and $n = 10000$ to the color constant descriptors and then apply the inverse of the cube root function to obtain reflectance estimates $\tilde{\mathbf{R}}$.

$$\tilde{\mathbf{R}} = \left(|\mathbf{o}_{\mathrm{cc}}(x, y) + [k, k, k]| \right)^{3.0} \tag{39}$$



(a) (b)

Fig. 4 (a) Color constancy results for an informed observer and two naive observers (data from Werner (2007)). (b) Angular error obtained using the computational model. Vertical bars denote the standard deviation. Note that the angular error is minimal for perfect color constancy, whereas Werner's color constancy measure is maximal for perfect color constancy. Thus, the computational model qualitatively models Werner's results.

We compute the average reflectance estimates over all pixel values of the test patch. Let $\tilde{\mathbf{R}}_d$ be the estimated reflectance under the canonical illuminant D65 and let $\tilde{\mathbf{R}}_e$ be the estimate under a non-standard illuminant. We compare these two estimates by computing the angular error e between the two.

$$e = cos^{-1} \frac{\tilde{\mathbf{R}}_e \tilde{\mathbf{R}}_d}{|\tilde{\mathbf{R}}_e||\tilde{\mathbf{R}}_a|} \tag{40}$$

This color constancy measure is zero for a perfect match between the two estimates. Note that Werner's color constancy index amounts to perfect color constancy for a maximum of 1, while our color constancy measure amounts to perfect color constancy for a minimum of 0. Figure 4(b) shows the results obtained from the computational model.

The computational model is completely deterministic. However, when a moving stimulus is used, then the output depends on the position of the test patch and on the position of the background. Therefore, we have averaged the output of our model over all possible initial conditions (with a granularity of 100 runs per stimulus) for experiments (b-d). Conditions where the stimuli is at the far right or left of the scene were excluded. This resulted in 82 different conditions for the moving stimulus. Figure 4(b) shows the angular error e between the estimated reflectance under a standard D65 illuminant and a non-standard illuminant. The standard deviation is also shown. A smaller angular error corresponds to better color constancy. Color constancy is significantly better (at 99% confidence level) during experiment (d) when compared to color constancy during experiment (a).

Werner also conducted an experiment with a color checkerboard pattern using stimuli (a) and (d). This color checkerboard pattern is shown in Figure 5(a). Figure 5(b) shows the Werner's color constancy index obtained in these experiments. Figure 5(c) shows the angular error obtained with our computational model. Again, the performance is significantly better in experiment (d) compared to experiment (a).

In an additional experiment, Werner looked at the dependence of the color constancy measure in relation to the speed of the stimulus with which it moved across the screen. Werner varied the stimulus from $2.4\frac{^\circ}{s}$ to $14.4\frac{^\circ}{s}$. The horizontal size of the entire scene was 19.8°. Hence, the left to right motion of the stimulus was completed in 8.25s (slowest speed) or 1.375s (fastest speed). The results from this experiment are shown in Figure 6(a). Again, our computational model also shows a dependence on the speed of the target patch as shown in Figure 6(b).

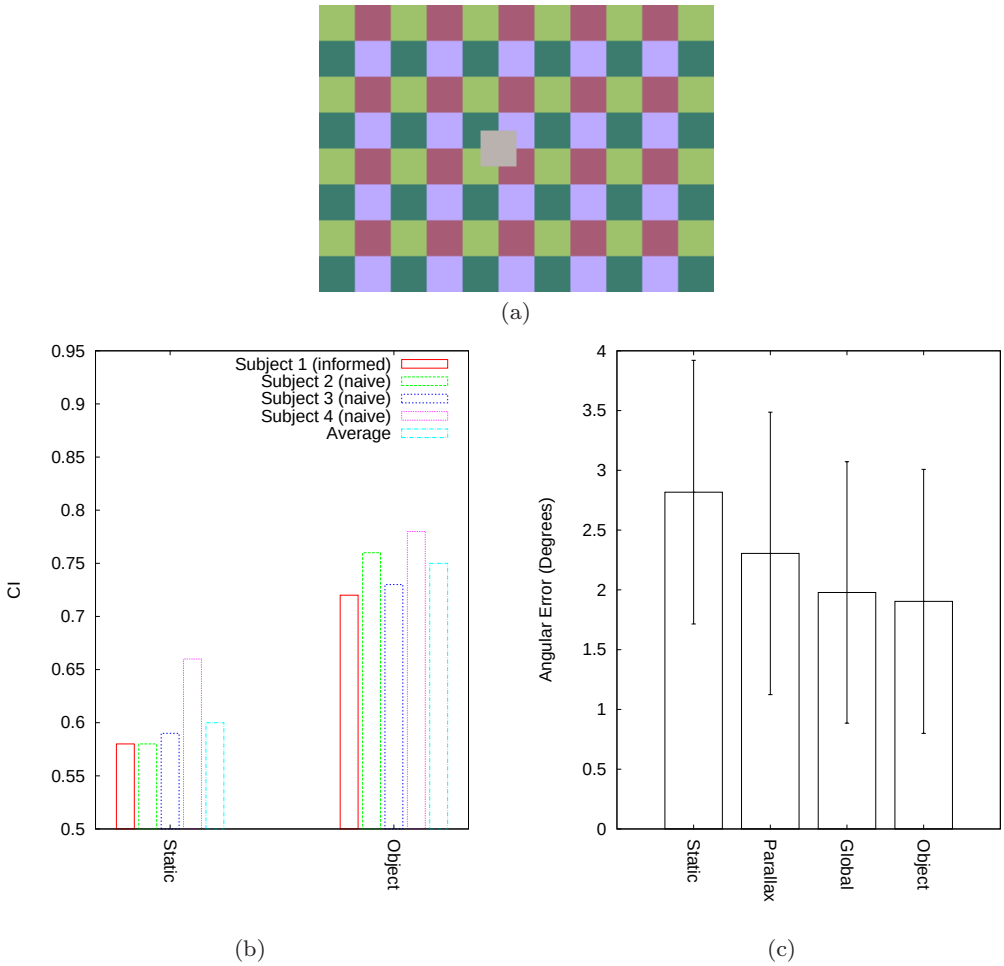


Fig. 5 (a) Color checkerboard pattern. (b) Color constancy index using the color checkerboard pattern and human subjects (data from Werner (2007)). (c) Angular error for stimuli (a-d) using the color checkerboard pattern and the computational model.

7 Analysis

Note that our results are obtained by pure bottom up computation. No motion detectors are required. The results are obtained as an effect of the algorithm used. Let us now analyze the results shown in Figure 4 and Figure 5 in more detail. Figure 7 shows local space average color computed by our algorithm for the four different stimuli (a) static scene, (b) motion parallax, (c) global motion and (d) object motion using the color checkerboard pattern. In the model, local space average color is computed in a cube root color space. In order to visualize local space average color, we have applied the inverse transform as described above. Each image shown in Figure 7 shows the local estimate of the illuminant. The color of the illuminant at pixels located inside the center of a patch are slightly biased towards the product of the reflectance of the patch times the color of the illuminant. Note that all input pixels are a product of the reflectance of the object patch and the illuminant. The illuminant scales all pixels. Thus, we can divide by the color of the illuminant and illustrate the effect in a better way. Figure 8 shows the color bias for experiments (a-d). In experiment (a), the viewer perceives a static scene. Hence, the center pixels of each patch are biased towards the color of the object patch. The colors in between are intermediate values. Because of this non-uniform bias, each color constant descriptor which is computed for the test patch will have a slightly different value. That's why we took the average over the color constant descriptors of the test patch to compute the perceived color of the test patch. Given the computational algorithm as described above, it is of course clear, that color constancy would be perfect for a gray checkerboard background and a gray test patch. That's why an offset between the estimated illuminant and the input is assumed when computing the color constant descriptor.

The motion parallax stimulus (b) is very similar to the static stimulus (a) when considering the stimulus

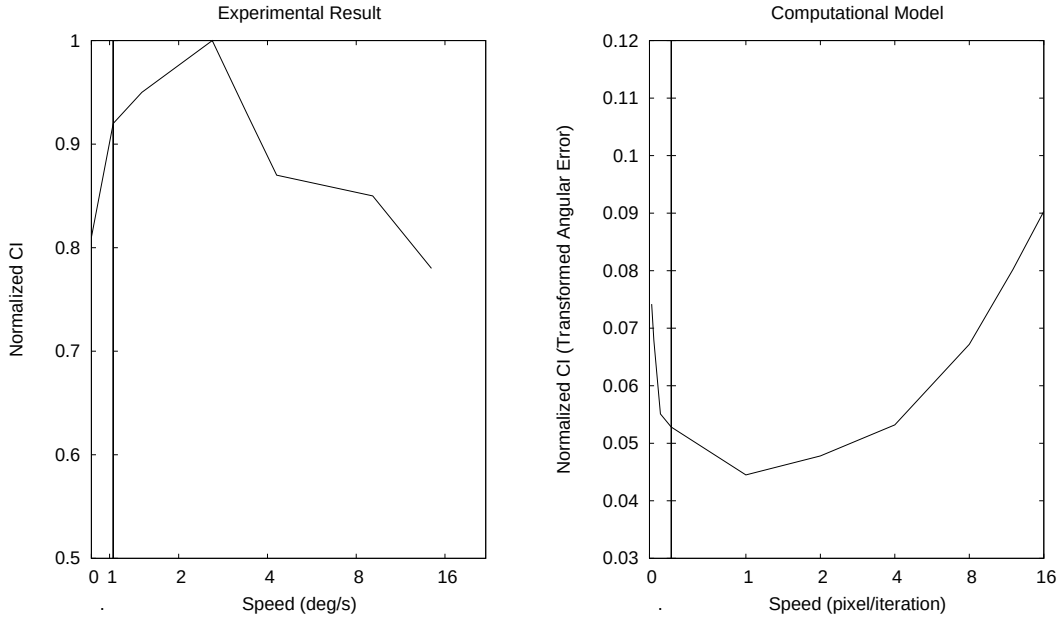


Fig. 6 (a) Color constancy as a function of speed (experimental data from Werner (2007)). (b) Results from the computational model. Note that the angular error should be minimized whereas the color constancy index should be maximized for perfect color constancy. Hence, the results again match qualitatively.

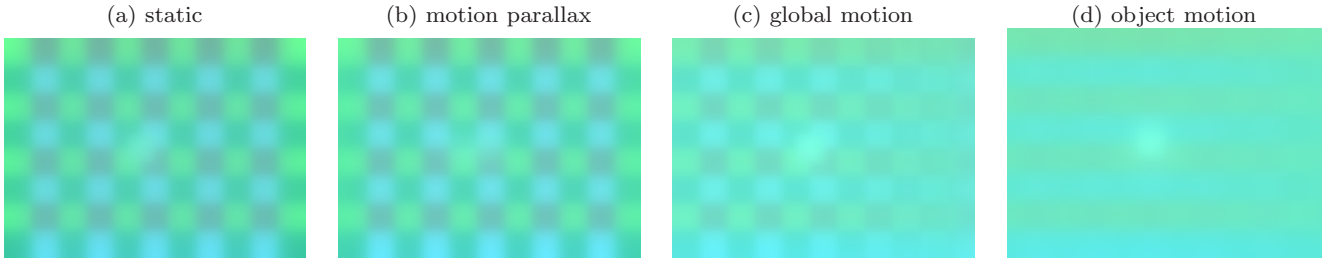


Fig. 7 Local space average color computed by the computational model.

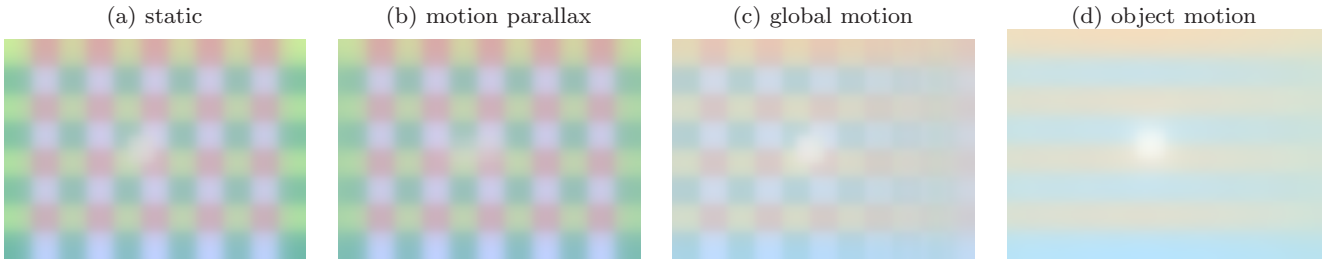


Fig. 8 Bias with respect to the illuminant for experiments (a-d).

from an eye-centered coordinate system. Hence, local space average color is quite similar to experiment (a). The only difference is that the test patch appears to move across the image close to the fovea. This enables the observer to perceive the stimulus in different positions relative to the background. Because of this, color constancy will improve during experiment (b) when compared to experiment (a). For global motion (c), the patch is fixated. This results in black stripes at the border where no input is obtained from the screen. Hence, there is less bias from the background. Finally during object motion (d), the background appears to move behind the test patch. Since local space average color is computed iteratively, each background pixel either has the color of an illuminated yellow patch, followed by an illuminated red patch or it has the color of an il-

luminated green patch followed by an illuminated blue patch. The iterative averaging leads to an overall average of the two neighboring patches, i.e. the bias is reduced as is shown in Figure 8(d). Therefore, color constancy performance is best, when the object moves across the background.

Given what we have just illustrated, it is also clear that color constancy will depend on the speed of the target patch across a checkerboard pattern. If the target patch remains stationary, then we have a static scene. As the target starts to move slowly across the background, color constancy will improve because a better estimate of the illuminant can be obtained. The bias (as described above) is reduced. Color constancy improves as the speed of the stimulus is increased. Why does color constancy eventually decrease? Well, the decrease is due to the processing speed of the brain/computational model. In order to understand how this works, consider a camera taking a video of a turning wheel. Initially, the wheel is at rest and the spokes are stationary. As the wheel starts to turn, we see how the wheel turns in on direction. Then the speed of the wheel is increased. Eventually, it will turn so fast that the processing speed (frame rate) cannot take up with the motion of the wheel. The spokes will become blurry. It will appear that the wheel slows down and eventually comes to rest. As the speed is increased even further, it will appear that the wheel turns into the opposite direction even though this is not really the case. If the speed of the target patch is increased to a certain point, then the visual image that is used to compute local space average color will be equivalent to a static image. For high speeds, the object will be moving so fast, that the object will blend with the background.

8 Conclusion

We have shown that Ebner's model of color perception is able to reproduce results from perceptual experiments. Previously it had been thought that these results would require an influence from high-level motion processes on color processing in the brain. However, the results show that this is not necessary. The computational model, which we have presented here, computes a color constant descriptor purely bottom up. No motion detection occurs at any point in the color constancy algorithm. It is only required for eye movements. Because of the eye movements, the output is dependent on whether the object, the background or both moves. It all depends on the signal reaching the retinal receptors. Thus, for the model presented here, color constancy is dependent on motion but high-level motion or any kinds of input from motion processing to color constancy processing is not required.

Acknowledgements

Marc Ebner thanks Annette Werner form the Dept. of Experimental Ophthalmology, Center for Ophthalmology from the Eberhard Karls Universität Tübingen for comments on this work. Marc Ebner acknowledges funding through the German Research Foundation (DFG) under grant no. EB 241/7-1 "Bio-inspired Machine Intelligence".

References

Bronstein IN, Semendjajew KA, Musiol G, Mühling H (2001) Taschenbuch der Mathematik, 5th edn. Verlag Harri Deutsch, Thun und Frankfurt/Main

Brown PK, Wald G (1964) Visual pigments in single rods and cones of the human retina. Science 144:45–52

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

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) 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, Tischler G, Albert J (2007) Integrating color constancy into JPEG2000. IEEE Transactions on Image Processing 16(11):2697–2706

Faugeras OD (1979) Digital color image processing within the framework of a human visual model. IEEE Transactions on Acoustics, Speech, and Signal Processing ASSP-27(4):380–393

Felleman DJ, Essen DCV (1991) Distributed hierarchical processing in the primate cerebral cortex. Cerebral Cortex 1:1–47

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
Helson H (1938) Fundamental problems in color vision. I. the principle governing changes in hue, saturation, and lightness of non-selective samples in chromatic illumination. Journal of Experimental Psychology 23(5):439–476
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
Hunt RWG (1957) Light energy and brightness sensation. Nature 179:1026–1027
Ilg UJ (1997) Slow eye movements. Progress in Neurobiology 53:293–329
International Commission on Illumination (1996) Colorimetry, 2nd edition, corrected reprint. Tech. Rep. 15.2, International Commission on Illumination
Koenderink JJ, van Doorn AJ (1999) The structure of locally orderless images. International Journal of Computer Vision 31(2/3):159–168
Koenderink JJ, van Doorn AJ (2000) Blur and disorder. Journal of Visual Communication and Image Representation 11:237–244
Land EH (1964) The retinex. American Scientist 52:247–264
Land EH (1974) The retinex theory of colour vision. Proc Royal Inst Great Britain 47:23–58
Land EH (1986) Recent advances in retinex theory. Vision Res 26(1):7–21
Lindner A, Schwarz U, Ilg UJ (2001) Cancellation of self-induced retinal image motion during smooth pursuit eye movements. Vision Research 41:1685–1694
Marks WB, Dobelle WH, MacNichol, Jr EF (1964) Visual pigments of single primate cones. Science 143:1181–1183
Werner A (2007) Color constancy improves, when an object moves: High-level motion influences color perception. Journal of Vision 7(14):1–14
Zeki S (1993) A Vision of the Brain. Blackwell Science, Oxford
Zeki S, Bartels A (1999) The clinical and functional measurement of cortical (in)activity in the visual brain, with special reference to the two subdivisions (V4 and V4α) of the human colour centre. Proc R Soc Lond B 354:1371–1382
Zeki S, Marini L (1998) Three cortical stages of colour processing in the human brain. Brain 121:1669–1685

Marc Ebner Dr. Marc Ebner received the M.S. degree in computer science from New York University, NY, in 1994, the degree Dipl.-Inform. from the Universität Stuttgart, Germany, in 1996 and the degree Dr. rer. nat. from the Eberhard Karls Universität Tübingen, Germany, in 1999. He received the venia legendi in 2006 from the Julius Maximilians Universität Würzburg, Germany. Dr. Ebner has received a Heisenberg Scholarship from the German Research Foundation (DFG). In May 2008, he moved to the Universität Tübingen. Since October 2011, he is full professor of computer science at the Ernst-Moritz-Arndt-Universität Greifswald. He is author of over 50 peer-reviewed publications and is a frequent speaker at international conferences; particularly on areas such as machine intelligence, computer vision, biologically inspired systems and genetic programming. Dr. Ebner is author of the first textbook on color constancy, i.e. algorithms for automatic white balance which try to mimic human color perception. He serves or has served as a reviewer for many different technical journals such as Evolutionary Intelligence, ACM Transactions on Applied Perception, Color Research and Application, IEEE Transactions on Pattern Analysis and Machine Intelligence, Journal of the Optical Society of America A or Neurocomputing and is a member of the program committee for major conferences.