UDC 535.6
ORIGINAL SCIENTIFIC PAPER
RECEIVED: 21-01-2012
ACCEPTED: 31-05-2012

ACTA GRAPHICA 201

Theoretical Prediction of Colours and Illuminants in sRGB Colour Space

Authors

Dejana Javoršek*, Andrej Iskra, Helena Gabrijelčič Tomc, Andrej Javoršek

Faculty of Natural Sciences and Engineering,
University of Ljubljana
Ljubljana, Slovenia
E-mail: dejana.javorsek@ntf.uni-lj.si

Abstract:

Photometric parameters and the evaluation of colour phenomena based on real human perception of light and colour were introduced into colour science in the first half of the 20th century. However, in the last two decades some known software programs for 3D computer graphics attempt to simulate realistic light source and optical effects with the help of photometric parameters. The reason for somewhat late arrival of such software applications can be found in the prevalent use of RGB colour space and the complexity of achieving realistic light source and colour effects. The aim of the research was to write and implement a simple gamut compression algorithm that will make colour transformation from input XYZ data to appropriate RGB data of sRGB colour space within large number of colours (1872) and different illuminants (16), and also to define appropriate light source intensity in 3D modelling software to achieve optimal matching of simulated and real photographic lights with predefined RGB values. Due to discrepancies between real light sources and the interpretation of light source colours in 3D, RGB colours of sRGB had to be adjusted with calibration procedure developed. The results of the research offer the possibility for theoretical prediction of virtual colours and light source intensity for different colours of light sources.

Keywords:

RGB Colour Space, Simulation of Colour and Light Source, 3D Graphic Software, Colour Conversion

1. Introduction

Three minimal conditions are indispensable for colour phenomena: a light source, an object and an observer. The light with defined spectral power distribution, emitted from light source, enables the illumination of an object and its surroundings. Optical, physical and chemical properties of an object and its material enable the selective reflectance/transmittance of light of defined wavelength interval. The observer and its visual system receive the reflected/transmitted light and transform it in colour information (*Valberg 2005, Tilley 2000, Evans, 1993*).

For screens, colour is an excitation of three phosphors (R-red, G-green and B-blue), and for the printing process colour is a reflectance of four inks (C-cyan, M-magenta, Y-yellow and K-black) on the paper surface. Mathematical representation of our colour perception and a tool to visualize, create and specify the colour are defined by colour space.

In a computer generated 3D environment, where a vectorial and perceptually non-linear RGB colour space is usually used, three conditions build up a virtual scene, e.g. a virtual object, a camera and a light source. The analogy with the natural system is obvious, especially in the case of an object and light source, while a 3D camera plays a role of an extension of visual system of humans in a virtual space. With the settings of virtual camera, its optical elements and mechanisms, the observing conditions are changed, so the observed image can be easily manipulated. In real cameras the image is stored on photosensitive elements such as CCD (charge-coupled device), on the other hand a 3D camera virtually acquires the image with predefined settings of camera. A virtual scene is finally rendered and displayed on the monitor (*Kerlow 2009*). This suggests that the simulation of colour in CG (computer graphics) is specific and differs from the analogue phenomenon in the real world. It is particularly difficult to achieve a realistic simulation of an object, light reflection and light source. The simulation of light source is possible with a predefined standardized colour of light source and a numerical definition of colour intensity, light filters, distribution, shadow effects, etc. Regardless of the development of colourimetry and its application in various research fields and branches, a photometrical definition of light source has been somehow neglected, due to a preferential use of non-perceptual colour models such as RGB, HSV (Hue, Saturation, Value) and HSL (Hue, Saturation, Lightness). The most commonly used colour space in different CG applications and other devices is sRGB.

Some settings of light source in virtual space are defined in the same way as in the real world. On the other hand, light phenomena in the virtual world do not happen spontaneously (shadows, volumetric lights, decay of illumination), so the settings of light source need special attention and a perfect knowledge of light behaviour in the nature (*Erzetič et al 2010*). Many researchers have investigated illuminants effect (*Stokes et al 2004, Debevec 2002, Greenberg et al 1997*). 3D modelling and rendering software applications use different solutions that result in more or less successful simulations for light-to-object interaction. Illumination models avoid complex calculations and are only an attempt on making the renderings of CG objects believable. Due to the complexity of light phenomena and colour transformations, the CG simulations are in general only approximations of the real appearance, which can nevertheless result in amazing visual realism. In most cases the best result is obtained by means of endless tricks and experiments with the material, shading algorithms, light source and rendering settings. Realistic renderings are achieved with the help of photometrical parameters: light source colour and light source intensity (*Birn 2000*, *Greenberg et al 1997*).

Besides light colour and intensity, basic settings of 3D light source are also minimal and maximal distances of light projection and shadow parameters, while advanced effects include map projections and some surface modifiers.

The main goal of the research was to write and implement a colour gamut mapping function that will transform colours from input XYZ data to appropriate RGB data of sRGB colour space within large number of colour samples (1872) and different illuminants (16), as well as to analyze light source settings, especially intensity, in a 3D virtual scene created with 3D modelling Blender software (*Blender website*) in order to achieve optimal simulations of real colour of light sources. sRGB colour space is quite limited by representing realistic colours, so colour conversion from CIEXYZ to RGB values should also include some colour compressions. Several gamut mapping methods used to transform colours from source space to the destination space by basic clipping of colours and complex transformation of colours have already been studied (*Kang 2003, Sharma 2003, Morovic 2001, Morovic 2000, Ebner et al 1997, Kimmel 2005*). In our research a simple gamut mapping function based on colourimetric rendering intent was implemented.

2. Experimental

In our research we used 1872 colours (printed with textile inkjet printer Mimaki Textile Jet Tx2-1600 with eight colours) and 16 illuminants, such as reflector illuminant with colour temperature 2760 K and 15 different filters for reflector light source.

2.1 Background: From spectra to sRGB

Tristimulus values CIEXYZ of illuminants (Table 1) and colours were obtained from spectral data measured with Spectrophotometer EyeOne (X-Rite).

Since the open source program for 3D modelling Blender 2.5 uses sRGB colour space, conversions from tristimulus XYZ values to RGB of sRGB colour space for illuminant and colour were used. If the reference white of the RGB model matches that of the CIE colour system, no chromatic adaptation is needed, so the choice of adaptation method has no effect (*Linbloom*). In our case reference white of both colour spaces was D65. In the research all calculations were performed using an open source program Octave 3.0 (*Octave website*).

Table 1: Different illuminants with their tristimulus values

Illuminant	X	Y	Z
reflector	110.71	100.00	32.17
dark green	36.13	100.00	24.62
light green	64.81	100.00	16.58
dark red	226.03	100.00	0.91
light red	207.00	100.00	2.21
dark blue	105.57	100.00	544.67
blue	68.23	100.00	268.14
yellow	123.60	100.00	2.95
violet	136.51	100.00	62.63
pink	166.04	100.00	33.08
orange	149.54	100.00	0.94
grey	109.49	100.00	31.49
dark grey	108.30	100.00	31.08
light blue	102.49	100.00	50.41
dark violet	247.62	100.00	123.96
green	48.44	100.00	71.01

2.2 Simple gamut compression algorithm

After colour conversion of 1872 colours from XYZ to RGB values, a great number of corresponding RGB values were still out of colour gamut of sRGB colour space. It was therefore necessary to implement some gamut mapping methods that will move colour of illuminants into sRGB colour space. For that purpose a simple gamut compression algorithm "RGB_adapt" was written and implemented, based on colourimetric ICC (International Colour Consortium) rendering intent (Table 2). In this algorithm, the function XYZtoRGB.m was used for the calculation of XYZ to RGB. If RGB values were out of sRGB gamut (255<RGB<0), chromaticity coordinates x, y as well as the distance between white point and colour point in CIE 1931 x, y diagram were calculated. Afterwards the distance between white point and colour point was decreased for value of 0.01 until colours were fitted into sRGB colour space. In Table 2 the distance is represented as purity. When calculated RGB values were in a range of 0-255, the loop terminated.

Table 2: "RGB_adapt" algorithm

```
algorithm [RGB_adapt]=gammapxy(XYZ)
%XYZn are XYZ values of standard illuminant D65
%mxn is chromatic coordinate x of CIE chromaticity diagram for D65
%myn is chromatic coordinate y of CIE chromaticity diagram for D65
%mx is chromatic coordinate x of CIE chromaticity diag. for particular colour
%my is chromatic coordinate y of CIE chromaticity diag. for particular colour
XYZn=[95.05 100 108.88];
mxn=XYZn(1)/(XYZn(1)+XYZn(2)+XYZn(3));
myn=XYZn(2)/(XYZn(1)+XYZn(2)+XYZn(3));
%XYZtoRGB.m is function for calculation of RGB from XYZ
RGB=XYZtoRGB(XYZ/100);
while true
       switch=1;
       for i=1:3
              if (RGB(i) > 255)
                     switch=0;
              endif
              if (RGB(i) < 0)
                     switch=0;
              endif
       endfor
       if (switch == 0)
              mx=XYZ(1)/(XYZ(1)+XYZ(2)+XYZ(3));
              my=XYZ(2)/(XYZ(1)+XYZ(2)+XYZ(3));
%purity is saturation
%angle is hue angle
              purity =sqrt((mx-mxn)^2+(my-myn)^2);
              angle=atan2((my-myn),(mx-mxn));
              if myn>my
                     angle=angle+2*pi;
              end
              purity = purity -0.01;
              mx=mxn+cos(angle)* purity;
              my=myn+sin(angle)* purity;
              XYZ(1)=mx*XYZ(2)/my;
              XYZ(2)=XYZ(2);
              XYZ(3)=(1-mx-my)*XYZ(2)/my;
% XYZtoRGB.m is function for calculation of XYZ from RGB
              RGB=XYZtoRGB(XYZ/100);
       else
              RGB_adapt=[RGB(1) RGB(2) RGB(3)];
              break;
       endif
end
```

2.3 Simulation of different illuminants in sRGB colour space

The colours of illuminants represented in CIE 1931 x, y diagram are displayed in Fig. 1 (triangle represents sRGB colour space). This figure shows that duller and more saturated illuminant colours, such as dark green, dark red, light red, dark blue, blue, yellow, orange, dark violet and green, are located out of sRGB colour space.

sRGB colour space represented in Lab diagram, shows the most extended area of colour gamut when CIELAB lightness L* is approximately 51. Since lightness L* is dependent on CIEXYZ lightness Y as exponential function, calculated Y was 20. So, all input XYZ values of illuminants were normalized to 20.

As we have already mentioned, in comparison with CIEXYZ colour space, sRGB is very small colour space. For this reason, we decided to decrease the purity of illuminants colour until appropriate RGB values in a range from 0 to 255 were obtained. For that purpose simple gamut compression algorithm "RGB_adapt" was used.

2.4 Defining the light source intensity

The following procedure of defining light source intensity in the Blender software included the setting up of a virtual scene in 3D space (cf. Fig. 2). The scene was composed of object surface (simple 2D plane), four light sources (type Sun simulating parallel light rays, angle of light sources 45°), and a camera (lenses 35 mm). No specific rendering settings were used. With this scene, the composition of uniform lighting of the plain surface was achieved. The surface of the object reflects in a lambertian way the same illuminating light with modified purity. The simulations of the modified reflector illuminant and 15 filters were obtained by means of changing the RGB values of virtual light source and light source intensity from values 0 to 1 and more. Light effects achieved on a virtual plain object were saved as image in tiff file format (8 bit, with no compression, 150 dpi) and analysed in Photoshop (Adobe) by means of a colour picker tool. In order to define the light source intensity in 3D virtual scene, a method of linear interpolation was used.

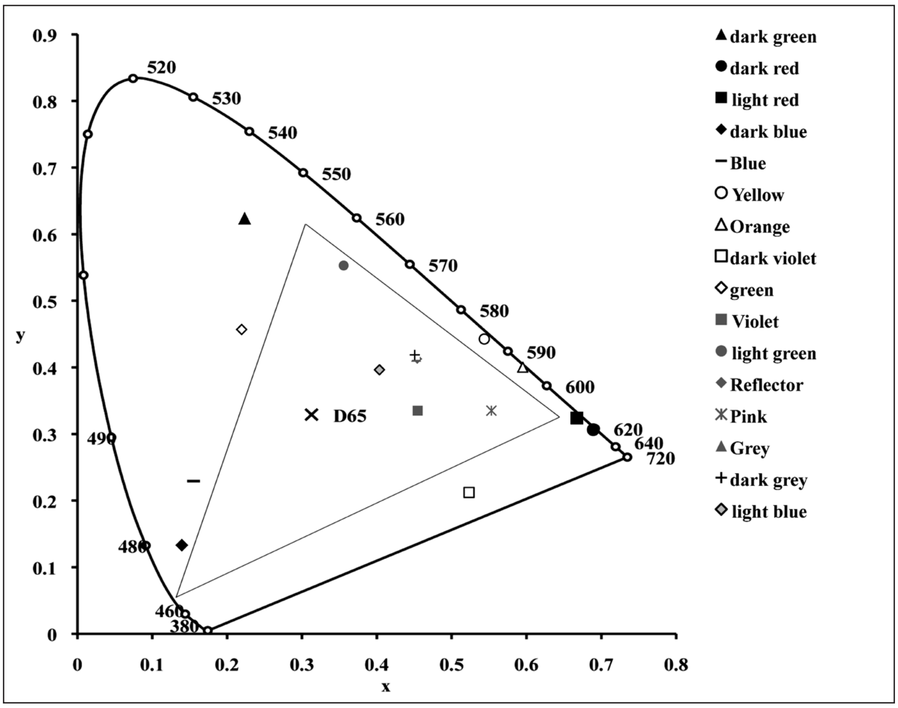

Figure 1: Illuminants colours in CIE 1931 x, y diagram, triangle - sRGB colour space

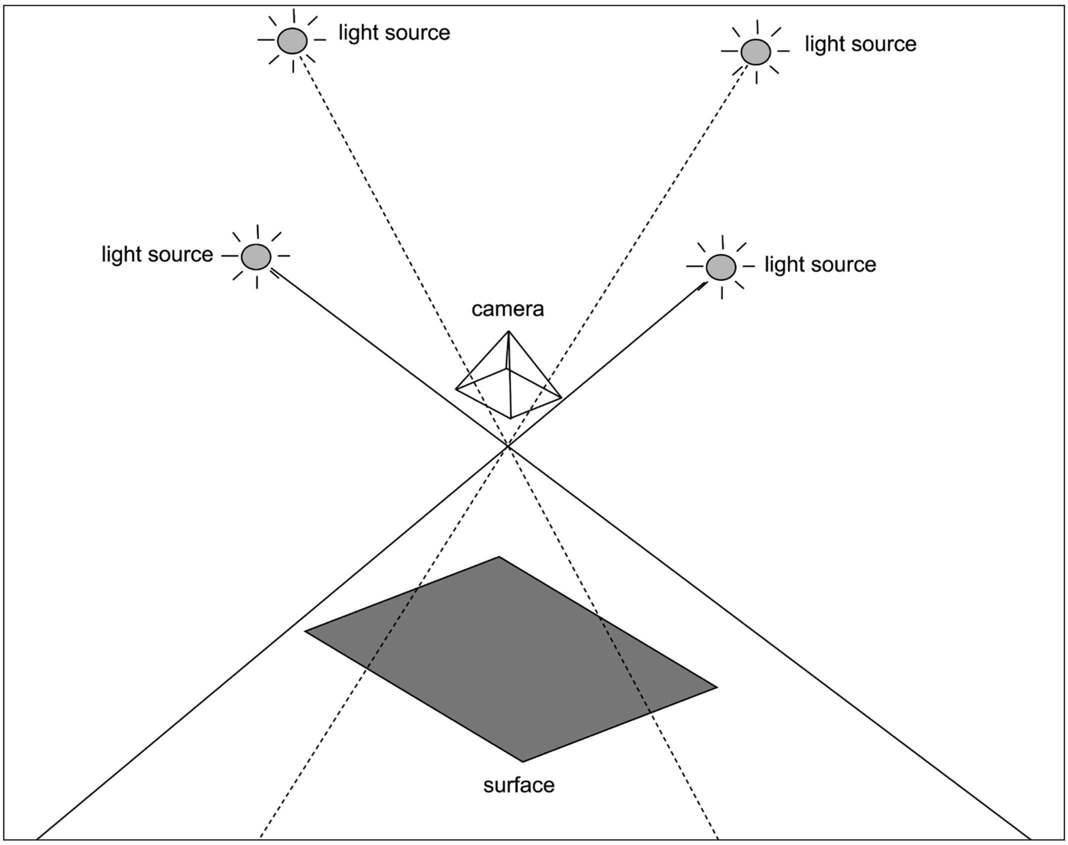

Figure 2: Virtual scene in the Blender software

3. Results and discussion

one of the main goals of our research was the simulation of colours and 16 different illuminants in the 3D Blender program. Since the scene in 3D Blender program is, besides other parameters, composed of light source and object surface, illuminants and colours could be analysed and discussed separately.

The results of direct colour conversion from CIEXYZ values to the corresponding RGB values of sRGB colour space shows that only 534 equal colours from 1872 colours under some illuminants (reflector and reflector with filters such as violet, pink, grey, dark grey, light blue) could be simulated without any colour compression into smaller sRGB colour space of Blender software (Fig. 3).

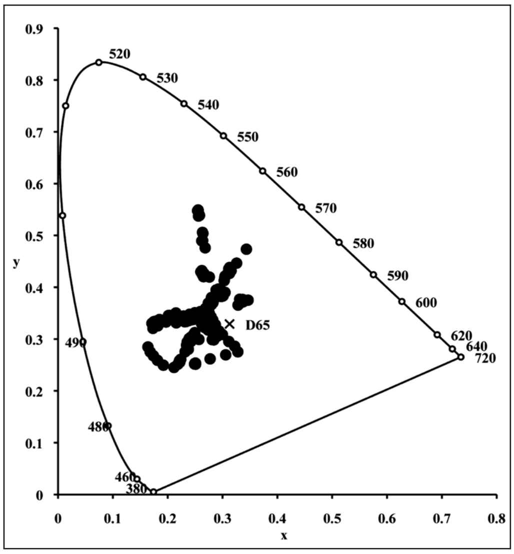

Figure 3: Colours (534) represented in CIE 1931 x, y diagram

In addition, the results of colour and illuminant conversion from CIEXYZ to sRGB using function "RGB_adapt" were presented.

3.1 Results of colour conversion from CIEXYZ to sRGB using "RGB_adapt" function

For the calculation of appropriate (adapted) RGB values of sRGB colour space, our simple gamut mapping function "RGB_adapt" was used. Colour differences ΔE^*_{ab} between the input and adapted data were calculated afterwards. Results show that 1670 colours were simulated with minimal and seven colours with small colour differences, while in the case of 158 colours, large colour differences were obtained (Table 3). In the case of colours that were set out of the gamut of sRGB colour spaces, large colour differences were obtained. Purity of those samples was decreased so as to achieve appropriate RGB values in a range of 0 to 255. Figures 4a-4b show colour shift from the boundary of sRGB colour space (input data) to area of less purity (adapted data).

Table 3: Number of colour patches divided into one of the four groups of ΔE^*_{ab}

ΔE^*_{ab}	Number of colours
0-1	1670
1-3	7
3-6	37
6-22	158

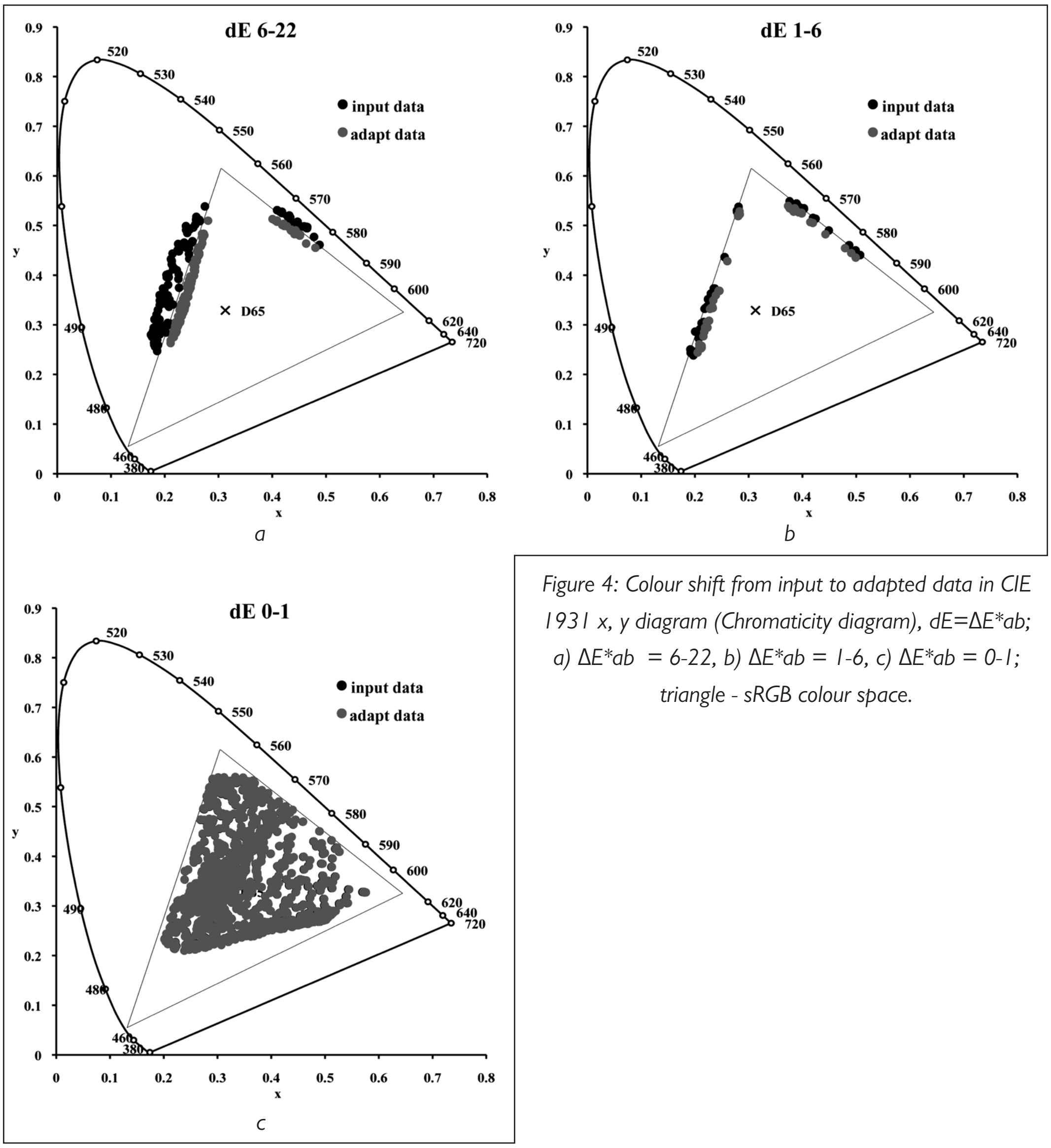

*Figure 4: Colour shift from input to adapted data in CIE 1931 x, y diagram (Chromaticity diagram), dE=ΔE*ab; a) ΔE*ab = 6-22, b) ΔE*ab = 1-6, c) ΔE*ab = 0-1; triangle - sRGB colour space.*

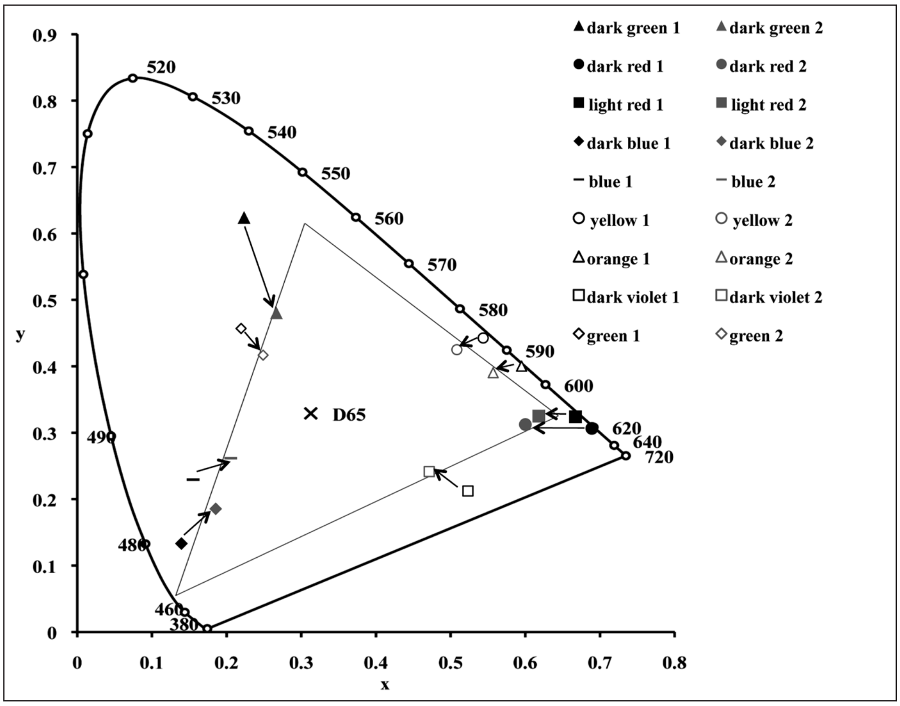

*Figure 5: Colour shift of illuminant from input to adapted data in a case of large ΔE*ab; CIE 1931 x, y diagram, triangle - sRGB colour space*

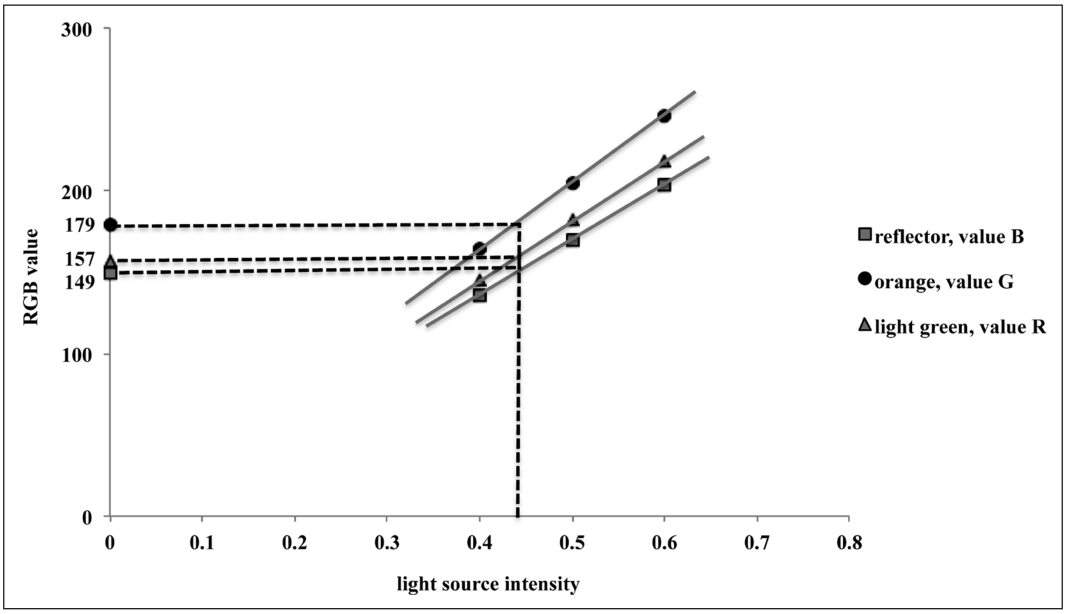

Figure 6: Linear correlation between light source intensity and RGB values of renderings for reflector, orange and light green illuminant

3.2 Results of illuminant simulation using "RGB_adapt" function

Some illuminants simulations, such as a simulation of reflector, light green, violet, pink, grey, dark grey and light blue, produced small ΔE^*_{ab} between the input and adapted data of illuminants (Table 4). In the case of other, more saturated illuminants, purity was decreased to the point where illuminants fitted into sRGB colour space.

Fig. 5 represents a colour shift of illuminants from input to adapted data in a case when large colour differences were calculated.

Table 4: Colour differences between input and adapted data of illuminants

Illuminant	ΔE^*_{ab}
reflector	0.30
dark green	39.34
light green	0.39
dark red	48.37
light red	31.11
dark blue	29.89
blue	19.36
yellow	25.51
violet	0.29
pink	0.27
orange	28.39
grey	0.30
dark grey	0.30
light blue	0.30
dark violet	30.35
green	18.25

Table 5 represented RGB values of illuminant colours of sRGB colour spaces obtained when our function "RGB_adapt" was used and Y was normalized to 20.

Table 5: RGB values of illuminant colours of sRGB colour spaces when Y was normalized to 20

Illuminant	R	G	B
reflector	165	112	55
dark green	9	141	78
light green	87	136	5
dark red	245	10	55
light red	244	18	35
dark blue	21	125	215
blue	9	133	168
yellow	179	105	16
violet	189	95	91
pink	222	66	61
orange	204	88	11
grey	163	112	54
dark grey	162	113	53
light blue	150	117	78
dark violet	236	15	132
green	9	140	102

3.3 Results of light source intensity

Setting the light source intensity in Blender software was very delicate, since changing its numerical value on the second decimal could result in visually perceived colour differences on rendered object. The changes in the light source intensity lower than 0.4 and higher than 0.6 resulted in non-perceivable and numerically irrelevant colour differences. Table 6 represents RGB results of light source intensity with values 0.4, 0.5 and 0.6.

In Figure 6 an example of linear correlation between light source intensity with the values ranging from 0.4 to 0.6 and RGB values of renderings is presented.

Table 6: RGB values of different illuminants defined from rendered image

Illuminant	Intensity								
	0.4			0.5			0.6		
	R	G	B	R	G	B	R	G	B
reflector	152	103	51	191	129	64	229	155	77
dark green	4	131	68	5	164	85	7	197	102
light green	79	127	4	99	158	5	119	190	7
dark red	223	30	0	255	38	0	255	45	0
light red	214	47	2	255	58	2	255	70	3
dark blue	4	115	202	5	144	252	7	172	255
blue	11	122	152	14	152	191	17	183	229
yellow	159	98	4	199	123	5	239	148	7
violet	174	87	84	217	108	105	255	130	126
pink	204	61	56	255	76	70	255	91	84
orange	176	89	2	220	111	2	255	134	3
grey	150	103	49	188	129	61	225	155	74
light blue	138	108	70	173	135	88	208	162	105
dark violet	218	4	122	255	5	152	255	7	183
green	4	129	94	5	161	117	7	194	141

Table 7: RGB values of illuminants using the light source intensity value 0.44

Illuminant	R	G	B
reflector	168	113	56
dark green	5	144	75
light green	87	139	5
dark red	245	33	0
light red	235	51	2
dark blue	5	126	222
blue	12	134	168
yellow	175	108	5
violet	191	95	93
pink	225	67	62
orange	194	98	2
grey	165	113	54
light blue	152	119	77

Regardless of colour of light source, the appropriate light source intensity was 0.44 (Table 7) regarding to input RGB values (Table 5). In a case of red, blue and dark violet illuminants, additional intermediate values of light source intensity should be calculated with linear interpolation in order to achieve exact results. However, as results demonstrated optimal values of intensity of 0.4 and 0.5, exactly 0.44, the latter was used in our research also in a case of less defined illuminant colours.

4. Conclusion

One of the major disadvantages of calculation from CIEXYZ to RGB is its inability to convert colours from bigger colour space into a smaller one. For that purpose, our research implemented a simple gamut compression algorithm "RGB_adapt" and used it for all illuminants and colour conversions. The function decreases purity to the point where all colours fit into sRGB colour space. Results showed that this function is very useful when real colours and illuminants should be simulated in 3D modelling software. Optimal light source intensity regarding to input RGB values could be defined by using linear interpolation. In our research optimal light source intensity in Blender software measured 0.44.

The results of the research offer a possibility of theoretical prediction of virtual colours and light source intensity for different colours of light sources.

References

KERLOW I., The art of 3D computer animation and Effects, 4th ed., John Wiley and Sons, Hoboken New Jersey 2009.

BIRN, J., 2000. Digital lightning and Rendering, New Riders Publishing, USA.

BLENDER WEBSITE. Available from: http://www.blender.org/ [online 2. 2. 2010]

DEBEVEC P., 2002. Tutorial, Image-based lighting, IEEE Computer Graphics and Applications, pp. 26-34. Available from: http://www.debevec.org/CGAIBL2/ibl-tutorial-cga2002.pdf [online 2. 2. 2010]

DIGITAL COLOUR IMAGING HANDBOOK, Edited by Gaurav Sharma, CRC Press 2003.

EBNER, F., FAIRCHILD, M. D., 1997. Gamut mapping from below: Finding minimum perceptual distances for colours outside the gamut volume, Colour Research And Application, **22**, pp. 402-413.

ERZETIČ, B., GABRIJELČIČ, H., 2010. 3D od točke do upodobitve. 2st ed., Pasadena, Ljubljana.

EVANS, D., WILIAMS, C., 1993. Colour & light, Dorling Kindersley, London.

GREEN P., MACDONALD L., 2002. Colour Engineering, John Wiley & Sons, New York.

Greenberg, D. P., Torrance, K. T., Shirley, P., Arvo, J., Ferwerda, J. A., Pattanaik, S., Lafortune, E., Walter, B., Foo, S., Trumbore, B., 1997. A framework for realistic image synthesis, Proceedings ACM Siggraph '97, pp. 477-494. Available from: http://www.cis.rit.edu/jaf/publications/framewrk.pdf [online 2. 2. 2010].

Kang, B. H., Morovic, J., Luo, M. R., Cho, M. S., 2003. Gamut Compression and Extension Algorithms Based on Observer Experimental Data, ETRI Journal, 25(3), pp. 156-170.

Kimmel, R., Shaked, D., Elad, M., Sobel, I., 2005. Space-dependent colour gamut mapping: a variational approach, Image Processing, IEEE Transactions on, **14**(6), pp. 796 – 803.

Lindbloom, B., XYZtoRGB, Available from: http://www.brucelindbloom.com/index.html?Eqn_ChromAdapt.html [online 9. 1. 2010] Morovic, J., Luo, M. R., 2001. The fundamentals of gamut mapping: A survey, Journal of Imaging Science and Technology, 45(3), pp. 283-290.

Morovic, J., Luo, M. R., 2000. Calculating Medium and Image Gamut Boundaries for Gamut Mapping, Colour Research And Application, 25(6), pp. 394-401.

Octave website, Available from: http://www.gnu.org/software/octave/ [online 9. 1. 2010].

Stokes, W., Ferwerda, J. A., Walter, B. J., Greenberg, D. P., 2004. Perceptual Illumination Components: A new approach to efficient, high quality global illumination rendering, Proceedings SIGGRAPH, pp. 742-749. Available from: http://www.cis.rit.edu/jaf/publications/PICS_SIGGRAPH2004.pdf [online 2. 2. 2010]

Tilley, R.J.D., 2000. Colour and optical properties of materials : an exploration of the relationship between light, the optical properties of materials and colour, John Wiley and Sons, Chichester [etc.].

Valberg, A., 2005. Light, vision, colour, John Wiley and Sons, Chichester.