Evaluation of colour space transformation suitability to optical temperature measurements

A Ziemba and E Fornalik-Wajs
AGH University of Science and Technology, Department of Fundamental Research in Energy Engineering, Al. Mickiewicza 30, 30-059 Krakow, Poland

E-mail: elaf@agh.edu.pl

Abstract. All optical measurement methods base on the image analysis and relation between the measured parameter and some image features. In Digital Particle Image Thermometry (DPIT), such relation represents a function between the temperature and particles’ colour (i.a. Thermochromic Liquid Crystals). For the quantitative data acquisition the “colour” information is necessary, therefore the colour spaces based on hue $H$ are used. Due to the big number of numerical operations needed in the analysis, the choice of colour space transformation is significant due to the accuracy and computational time. In this paper commonly applied RGB to HSI colour spaces’ transformations were compared and evaluation of their suitability to temperature measurement was performed. Time of obtaining the final results was considered as the main criterion. Appropriate calculations were conducted and presented.

1. Introduction

Image became a very important tool of getting an quantitative information about the object. In 2nd half of XX century it was just a source of qualitative information, but development of electronics, computer technologies and programming languages allowed this very useful change. At the moment, image analysis is significant in e.g. medicine, cartography, geology and of course engineering. In the engineering fields of interests image analysis is a basis for measurements with application of optical methods like Particle Image Velocimetry (PIV), Laser Induced Fluorescence (LIF) or Particle Image Thermometry (PIT). In all of these methods, image is analysed pixel by pixel to get relation between measured parameter and some image features, which will help to obtain the measured value. The idea of this paper was born during the image analysis with application of Digital Particle Image Thermometry (DPIT). In this method, for the quantitative data acquisition, the colour information is crucial, in opposite to the luminance or intensity, which are not important. Therefore the colour spaces based on hue $H$ (synonymous: colour, tone, shade, tint) are the most suitable. However, there are some disadvantages, which at first come from number of the available colour spaces and then, from the number of transformations necessary for getting the quantitative information. These disadvantages are mainly connected with proper choice of colour space and then with the time and computer resources needed for the processing. In the literature various transformations for the same colour spaces are reported. That is why authors undertook the task of deriving the linear transformation of red $R$, green $G$ and blue $B$ colour space to the hue $H$, saturation $S$ and intensity $I$ one and also to verify if the commonly used transformations are equivalent.
The main purpose of Digital Particle Image Thermometry is to get an information about object temperature in the basis of special particles’ colour (for instance Thermochromic Liquid Crystals TLC). DPIT with utilization of TLC were applied in many research areas such as the phase change [1], crystal growth [2], impinging heat transfer [3], convection [4] or medicine [5]. Illuminated with white light TLC particles reflect the wavelength in dependence on their temperature. The colour of TLC’s is changing in accordance to temperature changes and it can vary from red through green to blue [6]. Relation between the reflected wavelengths and temperature can be obtained during the calibration process and it makes possible application of the charge-coupled device (CCD) in getting the temperature field of studied system.

The colour is represented by RGB colour space and the number of colours depends on the number of bits. In DPIT method the image coded in 24-bits system will require a lot of operation to transform one colour system to another. That is why the simple and quick transformation is necessary. To describe the contribution of particular components in the colour, the normalized form of $R,G,B$ variables is used: $r = R/(R+G+B)$, $g = G/(R+G+B)$, $b = B/(R+G+B)$, where the values of $R,G,B$ are coming directly from image generation device (e.g. camera). The normalized form of variables does not include the black colour $K = (0,0,0)$ for which $R+G+B=0$. The values of $r,g,b$ for reflected light are unique and characterize one specific light wavelength $\lambda$. If the light reflected by TLC’s has been monochromatic or close to monochromatic (narrow range of wavelengths changes) the transformation of colour to temperature would be simple. It could take the form of look-up table in which the one wavelength corresponded to particular temperature. In practice, the reflected light is not monochromatic but its spectrum is characterized by one dominant wavelength with other wavelengths close to dominant one, causing its broadening. If the broadening is large and/or in the reflected light are also other wavelengths, the set of $r,g,b$ values will not be unique. However, for majority of experimental cases the set $r,g,b$ values form unique function of temperature, because the particular wavelength dominates strongly [6].

The experimental calibration allows finding of such unique relation between $r,g,b$ values and temperature. Direct way for finding this relation is to draw temperature versus the colour components $r,g,b$ as $T=f(r,g,b)$. However this kind of function gives various results and the transformation is not unique. Therefore there is common sensation that substitution of three variables with one is more accurate and efficient. This one variable is hue $H$, which is very difficult to define. It is such a property of pure colour, which helps to recognize it from the other pure colours. This property is identified by the name of colour and connected with the light wavelength. On the proposed by Newton and applied by Munsell wheel of colours, hue is an angle of pure colour.

2. Colour transformation from RGB to HSI colour space

The colour spaces, in which hue $H$ is one of the components are numerous. They were presented in detail in [7]. The colour spaces transformations presented in [6,8-11] are non-linear (require mathematical operations with application of trigonometrical functions, raising to a power or extraction of roots). During the image processing there is a need of multiple determination of HSI values, what can significantly increase time of necessary calculations. Moreover the non-linear transformation has a plane of singularity $2R-G-B=0$, what is an additional difficulty. In the scientific literature regarding digital image processing the examples of linear colour spaces transformation of RGB to HSI can be found, but they are presented without derivation of formulas [12-14]. The linear transformations are obtained on the basis of assumption that in the range, where one of $R,G,B$ components is maximal and another one is minimal, the value of hue can be approximated with linear function of third component equal to central value. The hue $H$ and saturation $S$ components of RGB colour space coming from non-linear transformation reported by [15] have their origin in the linear transformation. Many researchers are using the linear transformation formula in the Saturation calculations [16].
Smith in [12] presented an example of linear transformation with entire derivation of formulas and their geometrical interpretation. Instead of intensity \( I \) he was using the maximal brightness called value \( V \). The \( R,G,B \) values were the input parameters, while \( H,S,V \) values – output ones. All numbers were normalized to the range \((0,1)\). The concept of Smith transformation was converted to an algorithm by Authors and compared with other from the equivalence and computational time points of view. The other example of linear transformation of \( RGB \) colour space to \( HSV \) one was described by Foley [13], who presented it in the form of program written in C language, however the derivation was omitted. In his case the variable \( V \) was slightly different than intensity \( I \). The algorithm was transcribed by Authors from Foley program [13] to prepare the form suitable for comparison with the other algorithms. The third example of discussed algorithm is connected with Java platform. In accordance with documentation of programming Java platform [14] the class \texttt{java.awt.Color} includes a procedure \texttt{RGBtoHSB}, which allows transformation of \( RGB \) colour space to \( HSB \) one, in which the brightness \( B \) value, similar to the definition of value \( V \) is determined in the same way. Based on the class source code, Authors wrote down the algorithm of colour space transformation. As an input parameters the \( R,G,B \) values were given in a form of integer number from the range \([0,255]\), while as an output parameters the values of \( H, S, B \) in a form of floating point numbers, normalized to the range \([0,1]\).

In the algorithm based on Foley program [13], the colour angle hue \( H \) belonged to the range \([0°, 360°]\), while the java procedure \texttt{RGBtoHSB} [14] and algorithm coming from Smith program [12] were returning the values from range \([0,1]\). When saturation \( S \) was equal to zero \( S=0 \), in Smith program, hue \( H \) value was undefined, in \texttt{RGBtoHSB} procedure the hue value was equal to zero and in Foley program the hue reached the value outside the range of \([0°, 360°]\). Because of this the concept of new faster and simpler linear transformation was born.

3. Derivation of proposed linear transformation

Figures 1 and 2 presents colour hexagon divided into six equilateral triangles, marked from 0 to 5. Two neighbouring triangles (being next to each other) having the same side and laying on the projection of \( RGB \) system axis (segments of solid lines) form the rhombus. Inside of them there are the projections of colours characterized by maximal value of one of primaries. The values of \( R,G,B \) primary components in particular triangles of hexagon from maximal (\( \text{max} \)) trough central (\( \text{cen} \)) to minimal (\( \text{min} \)) are listed in Table 1.

Definition of colour intensity generally applied [12-14] has the following form

\[
I = \max = \max(R,G,B).
\]

Saturation \( S \) is a measure of colour purity and measure of a distance from intensity \( I \) axis [6]. On the intensity \( I \) axis colour saturation \( S \) is equal to zero (\( S=0 \)). On those of the colour cube edges, where one of the colour components takes the value equal to zero and at least one another component takes the value equal to 1, the saturation \( S \) takes maximal value equal to 1 (or 255, in dependence on accepted limits of \( RGB \) components variation). Hanbury in [17] introduced commonly accepted definition of saturation as \( S=\text{max}–\text{min} \) and demonstrated that its distribution is regular and symmetrical.

Mathematical definition of saturation \( S \) for colour cube giving the values within the range \([0,1]\) can be derived with assumption that for \( \text{min}=\text{max}=0 \), saturation \( S \) is equal to zero \( S=0 \), because \( R,G,B \) components of grey colours on the intensity axis \( I \) are equal to each other. For \( \text{min}=0 \) and \( \text{max}>0 \) it can be assumed that relative saturation is equal to one, because the difference between components is maximal. Black colour described by \( K=(0,0,0) \), for which intensity \( I \) is equal to zero \( I=0 \) (\( \text{min}=\text{max}=0 \)) is excluded from the transformations - saturation \( S \) and hue \( H \) values are undetermined. Assuming, that saturation \( S \) is linear function of minimal value \( S = a \cdot \text{min} + b \) the coefficients \( a \) and \( b \) can be calculated from following equations system:
Equation 2(c) describing saturation $S$ confirms the relation presented by Smith in [12] and as distinct from formula $S = \frac{\text{max} - \text{min}}{\text{max}}$, it can be treated as relative saturation. It is a function of intensity $I = \text{max}$, what means that variables $H, S, I$ are dependent in the linear transformation. For intensity equal to zero $I = 0$, equation 2(c) would be singular, if the black colour $K = (0,0,0)$ has not been excluded. For the grey colours located on the intensity axis $I$, saturation is equal to zero $S = 0$ and hue $H$ is undetermined. For these cases all algorithms are giving conventional values. On the edges of arbitrary cube of size smaller or equal to the typical RGB colour cube, saturation value is invariable and does not depend on the formula $S = \frac{\text{max} - \text{min}}{\text{max}}$ or $S = 1 - \frac{\text{min}}{\text{max}}$. 

\begin{align*}
0 &= a \cdot \text{max} + b \\
1 &= a \cdot 0 + b \\
a &= -1/\text{max} \\
b &= 1
\end{align*}

(2a) 

Saturation $S$ then can be calculated from the equation

\[ S = -\left(\text{min}\right)/\left(\text{max}\right) + 1 = 1 - \left(\text{min}\right)/\left(\text{max}\right). \]

(2c) 

**Table 1.** Colour primary component values in dependence on triangle number

| Colour of rhombus | No. of triangle | Component |
|-------------------|----------------|-----------|
| red               | 5              | $\text{max} \quad \text{min} \quad \text{cen}$ |
|                   | 0              | $\text{max} \quad \text{cen} \quad \text{min}$ |
| green             | 1              | $\text{cen} \quad \text{max} \quad \text{min}$ |
|                   | 2              | $\text{min} \quad \text{max} \quad \text{cen}$ |
| blue              | 3              | $\text{cen} \quad \text{min} \quad \text{max}$ |
|                   | 4              | $\text{cen} \quad \text{min} \quad \text{max}$ |

Figure 1. The colour example in triangle No. 5

Figure 2. The colour example in triangle No. 0
Following derivations of saturation $S$ and hue $H$ formulas in red rhombus are based on the geometrical relations presented in figures 1 and 2. On these figures, the projections of particular colour vector coordinates segments on the chrominance plane $R+G+B=0$ are shown. The descriptions of segments inform about their real length. However the derivation uses only the proportions of projected segments with length being a product of real segments length and value coming from geometrical relations.

The saturation $S$ absolute value of exemplary colour $(R,G,B)$ in triangle 5 (marked in figure 3 as $S_o=R–G$) is invariable for the colours, for which the blue colour component $B$ changes from $B=G$ to $B=R$. After projection of this colour on plane $G=0$, the colour without grey $(R-G,0,B-G)$ can be obtained. Its absolute saturation is also equal to $S_o$. After successive projection on the plane $R=R$, the colour of maximal saturation in sub-cube of size $R (R,0,R·(B-G))/(R-G))$ is obtained. Its absolute saturation $S_{am}$ is equal to $R$. Absolute saturation of exemplary colour two successive projections can be applied in determination of relative saturation in triangle 5, $S=S_o/S_{am}=(R-G)/R$. In accordance with Table 1, the equation $S=(max−min)/min$ can be presented and it is in agreement with equation 2(c).

To determine hue value $H$ in triangle 5 the segments of $(R–G)$ and $(B–G)$ lengths (shown in figure 3) are utilized. The segments quotient $(R–G)/(B–G)$ is equal to:

- 0, when $B=G$ (hue $H=360°$ for colours belonging to triangle of following vertices: red $R=(1,0,0)$, white $W=(1,1,1)$ and black $K=(0,0,0))$,
- 1, when $B=R$ (hue $H=300°$ for colours belonging to triangle of following vertices: magenta $M=(1,0,1)$, white $W=(1,1,1)$ and black $K=(0,0,0))$,
- 0.5, when $B=0.5·(B+G)$ (hue $H=330°$ for colours belonging to triangle of following vertices: $0.5·(R+M)=(1,0,0.5)$, white $W=(1,1,1)$ and black $K=(0,0,0))$.

It follows from this relation that in triangle 5 hue value can be presented as linear function of quotient $–(B–G)/(R–G)$. When the description listen in Table 1 is used and after assurance of positive hue values (trough addition of round angle 360°) the following equation can be presented:

$$H = 60° · \left( -\frac{cen-min}{max-min} + 6 \right). \quad (3a)$$

The hue values in other triangles could be derived in similar way and can be presented as following equations:

$$H = 60° · \left( +\frac{cen-min}{max-min} + 0 \right). \quad (3b)$$
$$H = 60° · \left( -\frac{cen-min}{max-min} + 2 \right), \quad (3c)$$
$$H = 60° · \left( +\frac{cen-min}{max-min} + 2 \right), \quad (3d)$$
$$H = 60° · \left( -\frac{cen-min}{max-min} + 4 \right), \quad (3e)$$
$$H = 60° · \left( +\frac{cen-min}{max-min} + 4 \right). \quad (3f)$$

Saturation in each triangle can be calculated with application of equation 2(c). All equations 3(a)–3(f) form set of equations, from which hue $H$ can be calculated in whole range. This set of equations is called set_3 and such abbreviation will be applied in the following sections.

The equivalence of all mentioned in this paper transformations was proved and two examples are presented in the form of Tables 2-3. Table 2 contains a proof of $H$ formulas equivalence in Smith algorithm and the RGBtoHSB method, while Table 3 lists a proof of $H$ formulas equivalence in Smith algorithm and proposed equations set_3.

4. Calculation time estimation

In paper [18] the analysis of time needed for presented transformations of RGB to HIS colour system was evaluated. Only one example was consistent with described algorithms and corresponded to the Foley algorithm. Authors concluded that for the object recognition purposes the transformation of RGB to HLS is better than to HSI due to application of the luminance $L$ definition instead of intensity $I$ one. Luminance in their opinion is more suitable for calculations and is giving satisfactory time of computing. As it was mentioned in Introduction, for DPIT method luminance $L$ cannot be applied.

To compare various RGB to HSI colour spaces transformations described in previous sections the special tool was created. This tool was prepared in java language and enabled speed testing of
conducted transformations for following methods: empty – it didn’t perform any calculations, that is why it represented theoretically the fastest possible reference transformation; modJava – it was a source code of Color.RGBtoHSB method [19], in which the input and output parameters were changed to adopt them to other methods; set_3 – represented proposed equation system (3); Smith – regarded to the Smith algorithm; stdFoley – indicated the standard Foley algorithm; optFoley – referred to the optimized Foley algorithm of 40% better performance [20]. Applied tool, six times calculated H, S, I values for each transformation in 256³ points of RGB standard colour cube, which corresponded to 49 images of 1920×1080 pixels resolution.

Table 2. The proof of H formulas equivalence in Smith algorithm and the RGBtoHSB method

| No. of triangle | Working variables of Smith algorithm: | Hue of color according to Smith algorithm | Working variables of RGBtoHSB method: | Hue of color according to RGBtoHSB method |
|----------------|--------------------------------------|------------------------------------------|-------------------------------------|------------------------------------------|
|                | V = max(R,G,B)                       |                                          | red = max-R                           |                                          |
|                | X = min(R,G,B)                       |                                          | green = max-G                         |                                          |
|                | r = V−R                               |                                          | blue = max-B                          |                                          |
|                | g = V−G                               |                                          |                                      |                                          |
|                | b = V−R                               |                                          |                                      |                                          |
| 5              | r = 0                                 |                                          | red = 0                              |                                          |
|                | g = 1                                 |                                          | = b−1 = b−1 + 6                       |                                          |
|                | b = b                                 |                                          | H = 5 + b                            |                                          |
| 0              | r = 0                                 |                                          | red = 0                              |                                          |
|                | g = g                                 |                                          | = 1 − g                              |                                          |
|                | b = 1                                 |                                          | H = 1 − g                            |                                          |
| 1              | r = r                                 |                                          | red = r                              |                                          |
|                | g = 0                                 |                                          | = 2 + r − 1                          |                                          |
|                | b = 1                                 |                                          | H = 1 + r                            |                                          |
| 2              | r = 1                                 |                                          | red = 1                              |                                          |
|                | g = 0                                 |                                          | = 2 + 1 − b                          |                                          |
|                | b = b                                 |                                          | H = 3 − b                            |                                          |
| 3              | r = r                                 |                                          | red = 1                              |                                          |
|                | g = g                                 |                                          | = 4 + g − 1                          |                                          |
|                | b = 0                                 |                                          | H = 3 + g                            |                                          |
| 4              | r = 1                                 |                                          | red = 1                              |                                          |
|                | g = 1                                 |                                          | = 4 + 1 − r                          |                                          |
|                | b = 0                                 |                                          | H = 5 − r                            |                                          |

Time estimation was done in the basis of System.nanoTime() [19] method. To averaged the influence of VM java (Virtual Machine) operational way on the calculation time, at first the same speed test was performed but without time estimation. In such case VM java was getting the profiling information about all investigated methods [21]. In this step the standard options of VM were used together with – server and –client options in respective cases. VM sever option is more complex and effective while
running long-term programs, which at the beginning are slower but with time the adaptive optimization technology of native code alternates it. \textit{VM client} option is optimized to limit start-up time and memory footprint of computer programs. Such profile is suitable for interactive programs or short time running [22,23]. The time results were averaged (out of three runs) for both profiles of \textit{VM}.

Table 3. The proof of $H$ formulas equivalence in Smith algorithm and proposed equations set no. 3

| No. of triangle | Working variables of Smith algorithm: $r = \frac{V \cdot r}{V \cdot X}$, $g = \frac{V \cdot g}{V \cdot X}$, $b = \frac{V \cdot b}{V \cdot X}$ | Hue of color according to Smith algorithm $a = \frac{cen - min}{max - min}$ | Working quotient of equations $3: a = \frac{cen - min}{max - min}$ | Hue of color according to equations $3$ rescaled to interval $[0, 1]$ |
|-----------------|-------------------------------------------------------------|------------------------------------------|------------------------------------------|------------------------------------------|
| 5               | $r = 0$, $g = 1$, $b = b$                                   | $cen = B$                                | $H = -a + 6 = -\frac{B - \text{min}}{\text{max} - \text{min}} + 6 = \frac{-B + \text{max} - \text{min} - \text{max}}{\text{max} - \text{min}} + 6$ | $max = V$ |
|                 |                                                             | $\downarrow$                             | $= -1 + \frac{\text{max} - B}{\text{max} - \text{min}} + 6 = 5 + \frac{V - B}{V - X}$ | $min = X$ |
| 0               | $r = r$, $g = g$, $b = 1$                                   | $cen = G$                                | $H = a + 0 = G - \frac{\text{max} - \text{min}}{\text{max} - \text{min}} + 0 = 1 - \frac{\text{max} - G}{\text{max} - \text{min}} = 1 - \frac{V - G}{V - X}$ | $H = 1 - g$ |
| 1               | $r = r$, $g = 0$, $b = 1$                                   | $cen = R$                                | $H = -a + 2 = -\frac{R - \text{min}}{\text{max} - \text{min}} + 2 = \frac{-R + \text{max} - \text{min} - \text{max}}{\text{max} - \text{min}} + 2$ | $H = 1 + r$ |
| 2               | $r = 1$, $g = 0$, $b = b$                                   | $cen = B$                                | $H = a + 2 = B - \frac{\text{max} - \text{min}}{\text{max} - \text{min}} + 2 = \frac{-B + \text{max} - \text{min} - \text{max}}{\text{max} - \text{min}} + 2$ | $H = 3 - b$ |
| 3               | $r = 1$, $g = g$, $b = 0$                                   | $cen = G$                                | $H = a + 4 = \frac{G - \text{min}}{\text{max} - \text{min}} + 4 = \frac{G + \text{max} - \text{min} - \text{max}}{\text{max} - \text{min}} + 4$ | $H = 3 + g$ |
| 4               | $r = r$, $g = 1$, $b = 0$                                   | $cen = R$                                | $H = a + 4 = \frac{R - \text{min}}{\text{max} - \text{min}} + 4 = \frac{R + \text{max} - \text{min} - \text{max}}{\text{max} - \text{min}} + 4$ | $H = 5 - r$ |
In figure 3 the calculation time for all analyzed algorithms are presented in relation to the time of empty method. In figure 4 the calculation time reduced of the empty method is presented and referred to the modJava time. It was a time estimation of pure algorithms, which is not exact due to the application of VM and its optimization procedures. All calculations were conducted with double precision and the results are presented as percentage. Additional test was also done and it showed that all algorithms (except empty method) are giving similar results within the accuracy of $10^{-12}$. Calculations were conducted on the computer equipped with Windows 7 32 bit system, with JDK environment version 8, update 74, disconnected from the internet.

5. Conclusions
The light reflected from TLC particles is closely connected with their temperature. This feature is utilized in the PIT method. Image analysis requires transformation of RGB colour space to HSI one. Presented in the paper colour space transformations were verified from optical measurement point of view. Their suitability was evaluated. All analyzed methods are equivalent analytically and numerically. VM sever run the calculations about 1.8 time faster than VM client, which was calculating the fastest modJava and optFoley algorithms, while the slowest was the Smith method. The average time results were obtained for set_3 and stdFoley methods. VM sever better optimized the codes of set_3 and optFoley algorithms than the codes of modJava and Smith methods. Optimization of stdFoley method was the average one. Method of optFoley was faster than stdFoley of about 33.5% for VM client and 28.5% for VM sever (figure 4), what confirmed the results obtained in [20].

Because the transformations themselves were the part of test tool, their influence and numerical effectiveness on the calculation time was limited. Similar effect will be obtained in any case of Digital Image Processing. Even though, the time differences are not very big, the implementation of various transformation algorithms in one software should be considered. In dependence on the language, programming methods and needed operations number, significant increase of performance can be obtained.

Acknowledgement
The present work was supported by the Polish Ministry of Science (Grant AGH No. 11.11.210.312).

6. References
[1] Kowalewski T A, Cybulski A and Rebow M 1994 Particle Image Velocimetry and Thermometry in Freezing Water 8th International Symposium on Flow Visualization (Sorrento, Italy, September 1-4)
[2] Fornalik E, Leiner W, Szmyd S J, Kowalewski T A and Ozoe H 2005 Visualization of the Flow Structure and Temperature Field in the Region of Mixed Convection in Mechanics of 21st Century (Springer Verlag)

[3] Fornalik E, Yamamoto Y, Chen W, Nakabe K and Suzuki K 1999 Visualization of Heat Transfer Enhancement Regions Modified by the Interaction of Inclined Impinging Jets into Crossflow, MG&V 8 597–609

[4] Bednarz T, Fornalik E, Tagawa T, Ozoe H and Szmyd J S 2006 Convection of Paramagnetic Fluid in a Cube Heated and Cooled from Side Walls and Placed Below a Superconducting Magnet: Comparison Between Experiment and Numerical Computations Thermal Sci. Eng. 14 107–114

[5] Stasiek J, Jewartowski M and Kowalewski T A 2014 The Use of Liquid Crystal Thermography in Selected Technical and Medical Applications J. Crystallization Process Technol. 4 46–59

[6] Park H G, Dabiri D and Gharib M 2001 Digital Particle Image Velocimetry/Thermometry and Application to the Wake of Heated Circular Cylinder Exp. Fluids 30 327–338

[7] Ziembka A and Fornalik-Wajs E 2014 Transformation of Colour Space Dedicated to an Experimental Analysis Fulfilling the Applicability Criteria J. Phys.: Conf. Ser. 530 012047, http://dx.doi.org/10.1088/1742-6596/530/1/012047

[8] Dabiri D and Gharib M 1991 Digital Particle Image Thermometry: The Method and Implementation, Exp. Fluids 11 77–86

[9] Bunting F 1998 The ColorShop Color Primer (Light Source Computer Images, Inc. An X-Rite Company)

[10] Gonzalez R C and Woods R E 2002 Digital Image Processing, 2nd ed. (Upper Saddle River, New Jersey: Prentice Hall)

[11] Hay J L and Hollingsworth T H 1996 A Comparison of Trichromic Systems for Use in the Calibration of Polymer-Dispersed Thermochromic Liquid Crystals Exp. Therm. Fluid Sci. 12 1–12

[12] Smith A R 1978 Color Gamut Transform Pairs SIGGRAPH ’78 Proc. of the 5th annual Conf. on Computer graphics and interactive techniques (New York) vol 12 (New York: ACM) pp 12–19

[13] Foley J D, van Dam A, Feiner S K and Hughes J F 1996 Computer Graphics: Principles and Practice in C, 2nd ed. (Addison–Wesley)

[14] http://www.oracle.com/technetwork/java/javase/downloads/index.html

[15] Kowalewski T A 2007 Thermochromic Liquid Crystals, in Handbook of Experimental Fluid Mechanics ed C Tropea, A Yarin and J F Foss (Berlin, Heidelberg: Springer–Verlag) chapter B7.1 487–500

[16] Russ J C 2002 The Image Processing Handbook, 4th ed. (Boca Raton, London, New York, Washington D.C.: CRC Press LLC)

[17] Hanbury A 2008 Constructing Cylindrical Coordinate Colour Spaces Pattern Recognit. Lett. 29 494–500.

[18] Palus H and D Bereska 1995 The Comparison Between Transformations from RGB Colour Space to IHS Colour Space, Used for Object Recognition, 5th International Conference on Image Processing and Its Applications (Edinburgh, UK, 4-6 July)

[19] http://docs.oracle.com/javase/8/docs/api/index.html

[20] http://lolengine.net/blog/2013/01/13/fast-rgb-to-hsv

[21] https://docs.oracle.com/javase/8/docs/technotes/guides/vm/performance-enhancements-7.html

[22] https://docs.oracle.com/cd/E22289_01/html/821-1274/configuring-the-default-jvm-and-java-arguments.html

[23] http://www.oracle.com/technetwork/java/whitepaper-135217.html