

# A Multiprimary Display: Optimized Control Values For Displaying Tristimulus Values

*Friedhelm König<sup>1c</sup>, Kenro Ohsawa<sup>2a,d</sup>, Masahiro Yamaguchi<sup>a,d</sup>, Nagaaki Ohya<sup>a,d</sup>, and Bernhard Hill<sup>b</sup>*

*<sup>a</sup>Akasaka Natural Vision Research Center, Tokyo, Japan;*

*<sup>b</sup>Aachen University of Technology, Aachen, Germany;*

*<sup>c</sup>AGFA Corporation, Munich, Germany;*

*<sup>d</sup>Tokyo Institute of Technology, Tokyo, Japan*

## Abstract

Recently, multiprimary displays have been proposed by several authors. Contrary to conventional displays, such displays use more than three primaries to display a color. There are two main advantages to such a display: First, the additional degrees of freedom can be used to account for observer metamerism, and secondly, the color gamut of a multiprimary display is significantly larger than the device gamut of a comparable conventional display. The discounting of observer metamerism which has been discussed in (1) requires multispectral images. Unfortunately, multispectral images are not always available. Therefore, the focus of this paper is on finding optimized device control values for the display of tristimulus images. Due to the nature of the display, a triplet of tristimulus values can be displayed by a multitude of different control values, resulting in metameric colors with the desired control values. Each of these control values could be chosen to display the desired color. There is, however, a complication because human observers usually deviate from the standard observer and choosing control values randomly can cause visible artifacts. In this paper, we propose a method that guarantees that small changes of the tristimulus values correspond to small changes of the displayed color for arbitrary observers. The method first identifies the subset of feasible control values (output between 0% and 100%) that lead to the desired tristimulus values. In a second step, the center point of this subset is chosen as the desired color vector.

## Introduction

### The Multiprimary Display

Today, most display devices use three primary colors to display a given color. For example, a CRT uses a red, a green and a blue phosphor to mix the colors that are displayed on a screen. This approach has a serious drawback, because not all colors can be displayed using one given set of primary colors. For example, it is impossible to display a red color on a CRT, that is more saturated than the red primary. And the situation gets worse, if a very saturated yellow has to be displayed. The reason for this is the limited gamut, that can be achieved using only three primaries. A few years back, the printing industry had to tackle the same problems. Not all colors could be printed using cyan, magenta and yellow ink. The problem was solved by using additional inks, e.g. black ink for very dark portions of the image and special inks for colors that are needed in a given project, but cannot be achieved using the standard inks.

A similar approach can be applied to a display device. Additional primaries can be used to expand the gamut of the display. In the following, such a device will be called a "multiprimary" display. Both at the Akasaka Natural Vision Research Center and at Aachen University of Technology multiprimary displays have been realized by using two commercial LCD projectors that project their images on the same screen. Geometrical distortion due to the slightly different positions of the two projectors are corrected by software. Each projector is modified by inserting additional interference filters. Due to these filters, six spectrally different primaries are created. Fig. 1 shows the primaries of the Akasaka multiprimary display.

Because the display uses more than three primaries, it is often possible to display the same target color using

<sup>1</sup> Contact: [friedhelm.koenig.fk@germany.agfa.com](mailto:friedhelm.koenig.fk@germany.agfa.com); LG-EV, AGFA Fototechnik Werk München, Tegernseer Landstr. 161, 81539 Munich, Germany

<sup>2</sup> [ohsawa@akasaka.tao.go.jp](mailto:ohsawa@akasaka.tao.go.jp); Akasaka Natural Vision research Center, 1-8-6 Akasaka, Akasaka HKN Bldg. 3F, Minato-ku, Tokyo 107-0052 Japan

different metameric colors. In an earlier paper, we used this property for the discounting of observer metamerism. However, this approach requires a multispectral image as input and, therefore, cannot be used with a conventional tristimulus image.

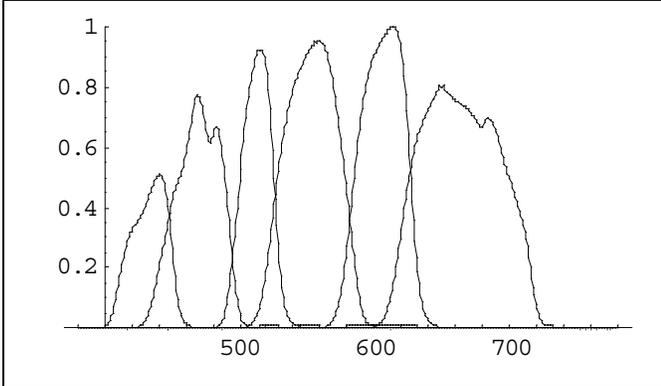


Figure 1. The six primaries of the multiprimary display

### The Influence of Observer Metamerism

At first glance, the use of a tristimulus image appears to be much simpler, because it should be sufficient to use any of a number of possible metameric colors. Starting from this idea, a method using matrix switching was implemented. This method showed good results for natural images. However, using synthetic images, sometimes strong artifacts could be examined. This was most pronounced examining an image that showed an  $a^*, b^*$  plane of the CIE  $L^*a^*b^*$  color space. In this image, it was possible to see, at which  $a^*, b^*$ -positions matrix switching took place.

The reason for these artifacts is observer metamerism. Due to matrix switching approach, it is possible, that two very similar colors correspond to vastly different device control values, and therefore to very different spectra. These spectra, however, are almost metameric and for the standard observer the difference between these two colors appears to be very small. For a human observer, who deviates from the standard observer, however, these changes can be very pronounced.

This is an rather unusual experience, usually the standard observer is a quite good description for an actual human observer. The reason for the unusual large influence of observer metamerism can be found in the nature of the device primaries. These primaries posses very steep flanks. Therefore even a small difference in the cone sensitivities of a human observer can cause large color differences.

The aim of this paper is to propose a control method for a multiprimary display that avoids artifacts, that are caused by observer metamerism.

### Mathematical Model of the Display

Any optimization involving a multiprimary display is - of course- dependent on an accurate display model describing the relationship between the displays input signals and the color displayed for these input values. The

input signals will be described using an input vector  $C$  and the resulting color by using its spectral power distribution  $s(\lambda)$ . The use of a spectrum instead of simple XYZ values is necessary, because the influence of observer metamerism is important in the context of this paper.

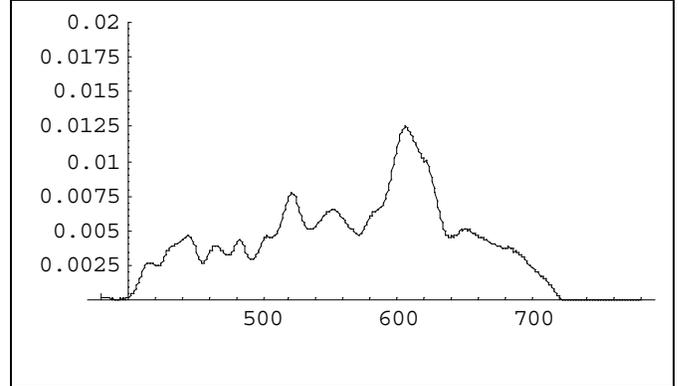


Figure 2. The bias of the multiprimary display

There are several possible definitions for the control vector  $C$ . The most obvious choice would be to use the two projectors control signals  $C_i$  ( $C_i \in \{0, 1, 2, \dots, 255\}$ ) directly. This approach, however, is not optimal due to the influence of nonlinearities (gamma curve) and we choose a slightly different definition:

$$C = (c_1, c_2, c_3, c_4, c_5, c_6)^T,$$

where the control signals  $c_i$  ( $0 \leq c_i \leq 1$ ) are a linearized version of the original control values  $C_i$ .

The model itself consists of 6 primaries  $p_i(\lambda)$  and a bias  $b(\lambda)$ , where the bias  $b(\lambda)$  is used to describe the rather high black level of the display. The bias can be seen in Fig. 2. Using these variables, the display output is calculated by:

$$s(\lambda) = b(\lambda) + \sum_{i=1}^6 c_i \cdot p_i(\lambda) \quad (1)$$

In tristimulus XYZ space, this is equivalent to:

$$S_{XYZ} = B_{XYZ} + \sum_{i=1}^6 c_i \cdot P_{XYZ,i} \quad (2)$$

### Optimal Control Values

In this paper, an optimized method of calculating the control values for a multiprimary display is proposed. Therefore, it is necessary to define the properties of optimal control values. Control values will be considered optimal, if the following three conditions are fulfilled.

- 1) The device control values need to be feasible. The algorithm used to calculate the device control values for an input XYZ vector must ensure, that each control signal has a value between 0 and 1.

- 2) For the standard observer, the tristimulus values XYZ of the resulting color are the input XYZ values, that were used to calculate the control values. This is only possible, if the input XYZ values are within the gamut of the display. For an input vector that is not within the display gamut, gamut mapping is required. A method to determine whether an input vector is within the display gamut will be discussed in the next section. Gamut mapping, however, is not within the scope of this paper.
- 3) A gradual change of the input tristimulus values results in a gradual change of the perceived color for arbitrary observers. This can only be achieved, if the spectral change also is gradual. Therefore, a gradual change of the color perceived by an arbitrary observer can only be achieved by a gradual change of the device control values.

### Calculating the Device Gamut

The second condition for optimal control values requires a method to check whether a control vector is within the gamut of the display. While the gamut of the multiprimary display is not obvious in XYZ space, it is quite easy to describe in device control space. Here, every control vector  $\mathbf{C}$  where each component  $c_i$  is a value between 0 and 1 is within the gamut of the display. Therefore, the display gamut is a hypercube in a six-dimensional space. By calculating the projection of this hypercube into the three-dimensional tristimulus space (XYZ) the device gamut in XYZ can be determined.

This transformation is strictly linear. Therefore, the convexity of the hypercube is preserved and the gamut in XYZ also is convex. This property allows an easy answer to the question, whether a given XYZ vector is within the device gamut by the following method.

First, for each corner of the control hypercube ( $\mathbf{C}$ , where all  $c_i \in \{0,1\}$ ) the corresponding XYZ values are calculated resulting in a set of tristimulus vectors  $\mathbf{C}_{XYZ,i}$ . The XYZ device gamut can now be determined by calculating the convex hull of these points. Please note, that same corners of the hypercube are projected into the interior of the tristimulus gamut.

On the other hand, all of the faces of the tristimulus gamut correspond to faces of the hypercube and all edges of the gamut correspond to edges of the hypercube.

Because each edge of the hypercube is described by a single control value  $c_p$ , the corresponding edge is parallel to the respective primary vector  $\mathbf{P}_{XYZ,i}$ . Furthermore, each face of the gamut is defined by two primaries, and can be described by an offset  $o_{ij}$  and the face vectors  $\mathbf{F}_{XYZ,i,j}$ , which are calculated by:

$$\mathbf{F}_{XYZ,i,j} = \mathbf{P}_{XYZ,i} \times \mathbf{P}_{XYZ,j} \quad (3)$$

where  $i < j$  and  $i,j \in \{1, 2, \dots, 6\}$ . The next step is to calculate a set of offsets  $o_{i,j,k}$  for each face vector  $\mathbf{F}_{XYZ,i,j}$ , by calculating the scalar product of the face vector with each corner vector  $\mathbf{C}_{XYZ,k}$ :

$$o_{i,j,k} = \mathbf{F}_{XYZ,i,j} \cdot \mathbf{C}_{XYZ,k} \quad (4)$$

Next, for each face vector  $\mathbf{F}_{XYZ,i,j}$  the maximum  $M_{i,j}$  and minimum  $m_{i,j}$  offset is determined:

$$m_{i,j} = \text{MIN}(o_{i,j,k}) \quad (5)$$

$$M_{i,j} = \text{MAX}(o_{i,j,k})$$

Using this information, an effective method to check whether an input tristimulus vector  $\mathbf{I}_{XYZ}$  is within the gamut of the display can be implemented. If and only if the input vector is within the gamut of the display the following inequations will be fulfilled:

$$m_{i,j} \leq \mathbf{F}_{XYZ,i,j} \cdot \mathbf{I}_{XYZ} \leq M_{i,j} \quad (6)$$

### Reformulating the Problem

Previously, the control signals of the multiprimary display were described by six-dimensional vectors, where each component corresponds to one control signal. This is not the only possible basis system for the description of the control vector. In fact, an alternative basis system allows to significantly simplify the problem of calculating optimized control signals. The first three components of the new basis system are used to describe the color information, while the remaining three components describe the metameric information (in relation to the standard observer). In the following, the construction of the new basis system will be described.

To this end, the primary vectors  $\mathbf{P}_{XYZ,i}$  in equation 2 are combined into a 3x6 matrix  $\mathbf{P}$ :

$$\mathbf{S}_{XYZ} = \mathbf{B}_{XYZ} + \mathbf{P} \cdot \mathbf{C} \quad (7)$$

Because different control vectors  $\mathbf{C}$  can correspond to the same XYZ vector, this equation, unfortunately, cannot be inverted. It is, however, possible to use the pseudo inverse  $\mathbf{P}^*$  of the matrix  $\mathbf{P}$  to calculate one possible solution. The columns of this pseudo inverse can be considered to be three basis vectors in a six-dimensional space. These basis vectors span the "color"-subspace of the control vectors. Using well known mathematical methods, it is possible to derive 3 additional six-dimensional basis vectors  $\mathbf{M}_1$ ,  $\mathbf{M}_2$  and  $\mathbf{M}_3$  that span the subspace orthogonal to the "color"-subspace resulting in:

$$\mathbf{C} = \mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}) + m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3, \quad (8)$$

where  $m_1$ ,  $m_2$  and  $m_3$  are arbitrary values.

Using this formulation, it is easy to find a display control vector that leads to a given XYZ vector. The only catch is that this vector needs to be feasible. It was already mentioned that the valid control signals are located within a six-dimensional hyper-cube. This is also true in the new coordinate system, although the cube might be rotated and stretched. In the new basis system, the feasible device control vectors need to fulfill the following inequations:

$$0 \cdot (\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}) + m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i \quad (9)$$

$$1 \cdot (\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}) + m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i$$

where  $(\mathbf{V})_i$  is used to denote the  $i^{\text{th}}$  component of the vector  $\mathbf{I}$  and  $1 \leq i \leq 6$ . These equations can be reformulated as:

$$-(\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}))_i \cdot (m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i \quad (10)$$

$$1 - (\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}))_i \cdot (m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i$$

As long as the input XYZ-vector is within the displays gamut, these inequations describe a three-dimensional volume in  $(m_1, m_2, m_3)$ -space. Some examples are given in Fig. 3. All  $(m_1, m_2, m_3)$ -vectors within these volumes lead to feasible display control vectors.

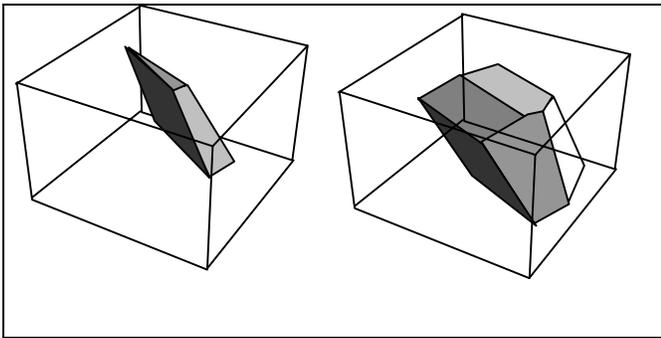


Figure 3. Volumes of  $(m_1, m_2, m_3)$ -vectors, that lead to feasible display control vectors for two sample XYZ vectors.

Now, the question is, which point of this volume represents the optimal solution. A good solution is to choose the center of gravity, because the volume changes gradually with gradual changes of the input XYZ vectors. In the following a method to calculate this center of gravity will be presented.

### Calculating the Center of Gravity

#### Representing Volumes

First, a good way of representing the volumes of possible  $(m_1, m_2, m_3)$ -vectors has to be found. One convenient representation is to describe the volume by using two fields *corner* and *face*.

*corner*: This field contains a list of the  $(m_1, m_2, m_3)$ -coordinates of the corners of the volume.

*face*: This is a field that contains a sorted list for each face of the volume. The list contains the indices of the corners defining the respective face. The indices are sorted to form a single convex polygon (no crossovers).

Using this description, a unity cube in  $(m_1, m_2, m_3)$ -space is represented by:

*corner*

Index	$m_1$	$m_2$	$m_3$	Index	$m_1$	$m_2$	$m_3$
1	0	0	0	5	1	0	0
2	0	0	1	6	1	0	1
3	0	1	0	7	1	1	0
4	0	1	1	8	1	1	1

*face*

Index	Point 1	Point 2	Point 3	Point 4
1	1	2	4	3
2	1	2	6	5
3	1	3	7	5
Index	Point 1	Point 2	Point 3	Point 4
4	2	4	8	6
5	3	4	8	7
6	5	6	8	7

This description specifies, that the 3<sup>rd</sup> face of the cube is the polygon  $(0,0,0)$ - $(0,1,0)$ - $(1,1,0)$ - $(1,0,0)$ - $(0,0,0)$ . Please note, that the sequence of the points is important, and  $(0,0,0)$ - $(1,1,0)$ - $(0,1,0)$ - $(1,0,0)$ - $(0,0,0)$  would not be permitted, as this polygon is not convex.

#### Determining the Volume Representation

The determination of the volume representation of the volume of permitted  $(m_1, m_2, m_3)$ -values requires two steps.

##### *corner data*

First, the *corner* data is determined. The corners of the volume are those places, where at least 3 faces meet. These positions can be calculated by replacing the inequations (1) describing the volume by the following equations.

$$-(\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}))_i = (m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i \quad (11)$$

$$1 - (\mathbf{P}^* \cdot (\mathbf{S}_{XYZ} - \mathbf{B}_{XYZ}))_i = (m_1 \mathbf{M}_1 + m_2 \mathbf{M}_2 + m_3 \mathbf{M}_3)_i$$

These equations describe  $2 \cdot 6 = 12$  planes in  $(m_1, m_2, m_3)$ -space. By choosing three of those planes and calculating the point of interception, a candidate corner is found. There are 160 sensible sets of equations (No set may contain two equations for the same  $i$ , as there is no solution that fulfils the two equations  $a+b=1$  and  $a+b=0$  at the same time).

All 160 solutions are either at the surface of the permitted volume or outside of the volume. Therefore it is necessary to use eq. 10 to check, which of these solutions are feasible. Also, any solutions, that appear more than once (e.g. because 4 or more planes intersect in one point) should be eliminated. The remaining feasible solutions form the *corner* data.

(All calculations are done with computer precision only. Therefore, some solutions might appear to be infeasible that in reality are feasible. To avoid this problem, it is recommended to relax eq. 10 slightly. Similar problems arise during the process of eliminating multiple points.

Again, points that are very close together should be considered to be one point)

### face data

The *face* data is determined next. For each of the 12 equation given in (2) all corners are determined that fulfill this equation (consider numerical precision!). If there are more than 2 points for one equation, these points describe one entry in the *face* data structure.

If an entry of the *face* data contains more than 3 points, it is necessary to determine an order. This can be done by the following algorithm:

- 1) For each entry in *face* do the following
- 2) Copy all but one of the corner points of the entry into a help variable. The point that is not moved is the first subentry of the *face* entry. This point is used as the work point.
- 3) Find another entry in *face*, that contains the work point and shares exactly 1 point with the points contained in help.
- 4) The shared point is the next subentry of the processed entry of the *face* data. It is also the next work point.
- 5) The new work point is removed from the help data.
- 6) As long as the help data contains more than one point, jump to step 3.
- 7) The last point contained in help is the last point of the processed *face* entry.

### Determining the Center of Gravity

After deriving a description of volume of permitted ( $m_1, m_2, m_3$ )-values, its center of gravity needs to be calculated. Unfortunately, there is no trivial solution for this problem. It is, however, quite simple to determine the center of gravity for an tetrahedron. And it is possible to cut the volume into tetrahedrons. Therefore, the determination of the center of gravity requires three steps:

- 1) The volume is divided into tetrahedrons.
- 2) The center of gravity and volume of each tetrahedron is calculated.
- 3) The center of gravity of the original object is calculated as a weighted sum of the centers of gravity of all tetrahedrons.

Because step one is the most complex, we will ignore it for the moment.

The center of gravity  $\mathbf{G}$  of a tetrahedron with the corners  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$  is given by:

$$\mathbf{G} = \frac{1}{4} (\mathbf{A} + \mathbf{B} + \mathbf{C} + \mathbf{D}) \quad (12)$$

The volume of a tetrahedron can be calculated by:

$$V = \frac{1}{6} |(\mathbf{B} - \mathbf{A}) \cdot [(\mathbf{C} - \mathbf{A}) \times (\mathbf{D} - \mathbf{A})]| \quad (13)$$

The center of gravity of a object  $P$  that consists of  $n$  tetrahedron with center of gravity  $\mathbf{G}_i$  and volume  $V_i$  is given by:

$$\mathbf{G}_p = \sum_i \frac{V_i}{\sum_i V_i} \mathbf{G}_i \quad (13)$$

### Dividing the Volume Into Tetrahedrons

The division of a volume into tetrahedrons requires the existence of a three-dimensional volume. This is important to mention, because the previous processing might lead to degenerated volume (all corners are within one plane, within one line or there is only one point.) These cases have to be addressed first.

The last two cases can be identified by the number of corners involved (two corners, if only a line of permitted values is available or only one corner, if only one choice of ( $m_1, m_2, m_3$ )-values is possible). In both cases, the calculation of the center of gravity is trivial.

An important property of the volume to be divided, is convexity. The algorithm of dividing the volume is based on this property.

First a random corner is chosen. This corner will be called basis point. Due to the convexity, a line that start from the basis point cannot intersect the surface of the volume more than once. Therefore it is possible to divide the volume into tetrahedrons, that all share this point.

The next step is to examine all faces, whether they contain the basis point. If a face does not contain the basis point, we chose a random corner of the face as face basis point (e.g. the first entry in the respective *face* data structure.).

Starting from the face basis point, we chose one corner  $c2$  of the face that is adjacent to the face basis point (e.g. the second entry of the *face* data structure) and the next corner  $c3$  adjacent to this point (here: the third entry of the *face* data structure). The first tetrahedron respective to this face is described by the basis point, the face basis point and these two points  $c1$  and  $c2$ . The second tetrahedron is the tetrahedron consisting of the basis point, the face basis point, point  $c3$  and the point  $c4$  adjacent to  $c3$  (here: the fourth entry of the *face* data structure). This is repeated, as long as the new point is different from the face basis point. To clarify this, the process will be explained using the case of the unity cube mentioned above.

First, point 1 (0,0,0) is chosen as basis point. Examining the *face* data, we find, that faces 4, 5 and 6 have to be processed. Faces 1,2 and 3 need not to be processed, because they contain our basis point.

Corner 2 is chosen as face basis point on face 4. Therefore, the first tetrahedron consist of corners 1 (basis point), 2 (face basis point) 4 (adjacent point  $c1$ ) and 8 (point  $c2$ ). Using the corner data, the tetrahedron becomes (0,0,0),(0,0,1),(0,1,1),(1,1,1).

The second tetrahedron is described by the points 1,2,8 and 6. There is no additional tetrahedron, because the corner adjacent to corner 6 is the face basis point.

In total we get 6 tetrahedrons described in the following table:

No	Point 1	Point 2	Point 3	Point 4
1	1 (0,0,0)	2 (0,0,1)	4 (0,1,1)	8 (1,1,1)
2	1 (0,0,0)	2 (0,0,1)	8 (1,1,1)	6 (1,0,1)
3	1 (0,0,0)	3 (0,1,0)	4 (0,1,1)	8 (1,1,1)
4	1 (0,0,0)	3 (0,1,0)	8 (1,1,1)	7 (1,1,0)
5	1 (0,0,0)	5 (1,0,0)	6 (1,0,1)	8 (1,1,1)
6	1 (0,0,0)	5 (1,0,0)	8 (1,1,1)	7 (1,1,0)

### Conclusion

In this paper, a method for calculating control values for a multiprimary display has been proposed. Contrary to a method proposed in (1), that required multispectral images as input, the new method is applicable to images using tristimulus color information.

Similar methods for images with tristimulus color information have been proposed before. Like the method proposed in this paper, these methods lead to an exact color match for the standard observer. However, the earlier methods frequently caused visible artifacts for non-standard observers. This can be avoided by using the method described in this paper.

The central idea of the new method is to define the volume of control values that lead to the desired tristimulus values for the standard observer. While any point of this volume may be chosen if only the standard observer is considered, the new method chooses the device control vector in such a way, that a gradual change of the input tristimulus vector leads to a gradual change of the device control vector, and, therefore, to a gradual change of the

spectrum used to display the color. Thus, visible artifacts are avoided.

The point chosen from the volume of possible solutions is the center of gravity of this volume. This is done, because a gradual change of the input tristimulus values causes only a gradual change of the volume and therefore of the center of gravity.

### References

1. Friedhelm König, Kenro Ohsawa, Masahiro Yamaguchi, Nagaaki Ohya, and Bernhard Hill, A multiprimary Display: Discounting observer metamerism, AIC Color Conference, (2001)

### Biography

Friedhelm König received his diploma degree in Electrical Engineering from the Aachen University of Technology in 1997. Starting in 1997 he worked as researcher at Aachen University of Technology, dealing with multispectral imaging and similar topics. Between November 2000 and April 2001, he was a visiting scientist at the Akasaka Natural Vision Research Center in Tokyo Japan where he was researching multiprimary displays. In 2001 he joined AGFA corporation, where he is working on improving the image quality of photographic printers.