A geometrical approach for inverting display color-characterization models

Jean-Baptiste Thomas
Philippe Colantoni
Jon Y. Hardeberg
Irène Foucherot
Pierre Gouton

Abstract — Some display color-characterization models are not easily inverted. This work proposes ways to build geometrical inverse models given any forward color-characterization model. The main contribution is to propose and analyze several methods to optimize the 3-D geometrical structure of an inverse color-characterization model directly based on the forward model. Both the amount of data and their distribution in color space is especially focused on. Several optimization criteria, related either to an evaluation data set or to the geometrical structure itself, are considered. A practical case with several display devices, combining the different methods proposed in the article, are considered and analyzed.

Keywords — *Display device, color characterization, inverse model, geometrical structure, 3-D look-up table, interpolation, data distribution, optimization.*

DOI # 10.1889/JSID16.10.1021

1 Introduction

The goal of color characterization of display devices is to define the relationship between the device-dependent color space, typically RGB, and a device-independent color space, typically CIEXYZ or CIELAB, based on a standard CIE colorimetric observer describing the perceived color.[1] This relationship has two directions. The forward transformation aims to predict the displayed color for any set of digital values input to the device, *i.e.*, a triplet (d_r, d_g, d_b). The inverse transformation provides the set of digital values to input to the device in order to display a desired color.

Among the models or methods used to achieve color characterization, we can distinguish two categories. The first one contains models that are analytically invertible,[2–8] such as the PLCC (piecewise linear assuming chromaticity constancy), the black corrected PLCC*, the GOG (gain-offset–gamma) or GOGO (gain-offset–gamma-offset) models. The second category contains the models or methods which are not analytically invertible. Models of this second category require other methods to be inverted. We can list some typical problems and methods used to invert these models:

- A condition has to be verified, such as in the masking model.[9]
- A new matrix might have to be defined by regression in numerical models.[10,7,8,11]
- A full optimization process has to be set up for each color, such as in S-curve model II,[12,13] in the modified masking model,[9] or in the PLVC (piecewise linear assuming variation in chromaticity) model.[3,6]
- The optimization process can appear only for one step of the inversion process, as in the PLVC[3] or in the S-curve I[12,13] models.
- Empirical methods based on 3-D LUT (lookup table) can be inverted directly,[14] using the same geometrical structure. In order to have a better accuracy, however, it is common to build another geometrical structure to yield the inverse model. For instance, it is possible to build a draft model to define a new set of color patches to be measured.[15]

The computational complexity required to invert these models makes them seldom used in practice, except the full 3-D LUT whose major drawback is that it requires a lot of measurements. However, these models do have the possibility to take into account more precisely the device color-reproduction features, such as interaction between channels or chromaticity inconstancy of the primaries. Thus, they are often more accurate than the models of the first category.

Given any of these models, we propose here to build a geometrical inverse model based on an optimized tetrahedral structure generated from the forward model, such that no more measurements than the ones needed to set up the forward model are required.

The main contribution of this work is to propose and discuss some algorithms allowing the establishment of the optimized geometrical structure of such an inverse model. This work follows and completes a preliminary study which has been presented earlier.[16] Two main grid features are studied, as well as the number of points used and their distribution.

In the following, we first introduce our general framework, modeling the problem and introducing notations. We then propose several methods to optimize both the number and the distribution of data used to build this inverse model.

J-B. Thomas is with the Norwegian Color Research Laboratory, Gjøvik University College, Gjøvik, Norway, and the Université de Bourgogne, Le2i, 9 av. Alain Savary, Djion, Bourgogne 21078, Dijon, France; telephone +47-611-40426, fax –35240, e-mail: jib.Thomas@gmail.com.

J. Y. Hardeberg is with the Norwegian Color Research Laboratory, Gjøvik University College, Gjøvik, Norway.

P. Colantoni is with the Centre de Recherche et de Restauration des musées de France, Paris, France.

I. Foucherot and P. Gouton are with the Université de Bourgogne, Dijon, France.

Finally, we present results on the inversion of the PLVC forward model applied to several display devices.

2 Background work

Building an inverse color-charaterization model based on a 3-D LUT is not new. Such a model is defined by the number and the distribution of the color patches used in the LUT, and on the interpolation method used to generalize the model to the entire space. In this section, we review some basic tools and methods. We distinguish between works on displays and more general works which have been performed in this way either in a general purpose or especially for printers. We present in detail the tetrahedral structure we used as a basis of our work. Our method to obtain the inverse model is based on the use of a forward model and on the refinement of the grid in the destination space rather than in the source space. We have chosen the PLVC model as forward transform, which is described in the second part of this section.

2.1 Building an inverse model based on a 3-D LUT

3-D LUT inverse models in displays are often based on the measurement of a defined number of color patches, *i.e.,* we know the transformation between CIELAB and RGB in a small number of color-space locations. Then, this transformation is generalized to the entire color space by interpolation. Previous studies assess that these methods achieve good results for display devices,[14,15] depending on the combination of the interpolation method used,[17,24–26,22] the number of patches measured, and on their distribution[15] (some of the interpolation methods cited above cannot be used with a non-regular distribution). However, to be precise enough, a lot of measurements are typically required, *e.g.,* a $10 \times 10 \times 10$ grid of patches measured in Bastani's paper.[14] Note that such a model is technology independent since no assumption is made about the device, only that the display will always have the same response as at the measurement location, *i.e.,* that it is spacially uniform. Such a model needs high storage capacity and computational power to handle the 3-D data. The computational power is usually not a problem since graphic processor units (GPU) can perform this type of task easily today. The high number of measurements needed is a greater challenge.

The problem of measurement is even more restrictive for printers and many works have been carried out in using a 3-D LUT for the color characterization of these devices. Moreover, since printer devices are highly non-linear, their colorimetric models are complexe, and it has been customary in the last decade to use a 3-D complex LUT for the forward model, created by using an analytical forward model; both to reduce the amount of measurement and to perform the color-space transform in a reasonable time. The first work we know about creating a LUT based on the forward model is a patent from Stokes.[27] In this work, the LUT is built to replace the analytical model in the forward direction. It is based on a regular grid designed in the printer CMY color space, and the same LUT is used in the inverse direction, simply in switching the domain and co-domain. Note that in displays, the forward model is usually computationally simple and that we need only to use a 3-D LUT for the inverse model. The uniform mapping of the CMY space leads to a non-uniform mapping in CIELAB space for the inverse direction, and it is common now to re-sample this space to create a new LUT. To do that, a new grid is usually designed in CIELAB and is inverted after gamut mapping of the point located outside the gamut of the printer. Several algorithms can be used to re-distribute the data[28–30] and to fill the grid.[31–33]

Returning to displays, let us call source space the independent color-space CIELAB (or either CIEXYZ), the domain from where we want to move and destination space, the RGB color space, the co-domain, where we want to move to. If we want to build a grid, we then have two classical approaches to distribute the patches in the source space, using the forward model. One can use directly a regular distribution in RGB and transform it to CIELAB using the foward model; this approach is the same as used by Stokes for printers,[27] and leads to a non-uniform mapping of the CIELAB space which leads to inaccuracy for the inverse direction. An other approach can be to distribute the patches regularly in CIELAB, following a given pattern, for instance following Stauders[15] for an hexagonal structure or any of the methods used in printers.[28–30] Then, an optimization process using the forward model can be performed for each point to find the corresponding RGB value.

The main idea of the method and the notation we used are the following: One can define a regular 3-D grid in the destination color space (RGB). This grid defines cubic voxels. Each one can be split into five tetrahedra. This tetrahedral shape is preserved within the transform to the source space (either CIEXYZ or CIELAB). Thus, the model can be generalize to the entire space, using tetrahedral interpolation.[17] The generalized way to define such a grid is to take directly a linear distribution of points on each digital R, G, and B axis as seeds and to fill up the rest of the destination space (Fig. 1). A tetrahedral structure is then built with these points. The built structure is used to retrieve any RGB value needed to display a specific color inside the device's gamut. The more points are used to build the grid, the more the tetrahedra will be small and the interpolation accurate.

In this work, we characterize such a grid by $N_{\mathrm{rgb}} = N_{\mathrm{r}} + N_{\mathrm{g}} + N_{\mathrm{b}}$, where N_{r} (resp., N_{b}, N_{g}) is the number of steps along channel R (resp., G, B). Each vertex is defined by $V_{i,j,k} = (R_i, G_j, B_k)$, where $R_i = d_i$, $G_j = d_j$, $B_k = d_k$, and $d_i, d_j, d_k \in [0,1]$ are the possible normalized digital values, for a linear distribution. $i \in [0, N_r - 1]$, $j \in [0, N_g - 1]$, and $k \in [0, N_b - 1]$ are the indexes (integers) of the seeds of the grid along each primary.

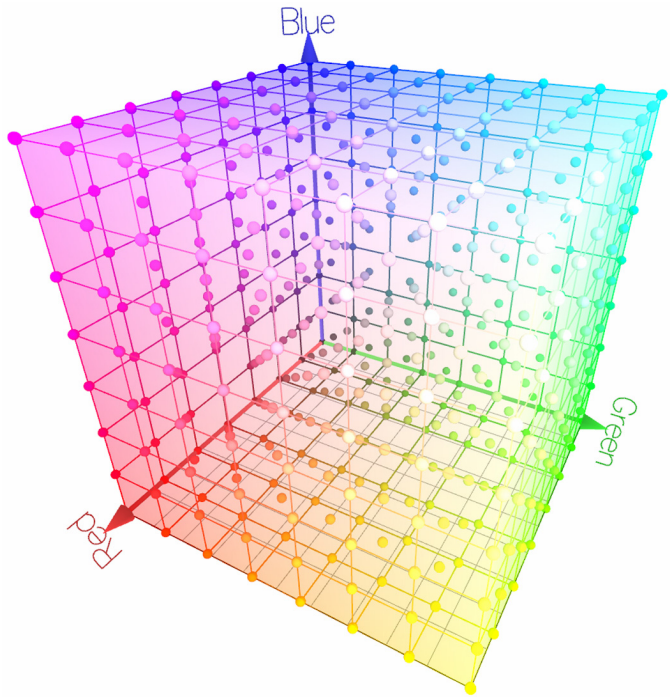


FIGURE 1 — Design of a regular grid N_{rgb} = 24 in RGB.

Once this grid has been built, we define the tetrahedral structure for the interpolation following Kasson *et al.*[17] Then we use the forward model to transform the structure into CIELAB colorspace (see Fig. 2). We have built an inverse model.

According to the non-linearity of the CIELAB transform, the size of the tetrahedra is not anymore the same as it was in RGB. In the following section, we propose to modify this framework to make this grid more homogeneous in the source color space where we perform the interpolation, this should lead to a better accuracy, following Ref. 29.

2.2 The PLVC model

The forward model we used in this work is the piecewise linear assuming variation in chromaticity (PLVC). This model has been introduced first by Farley and Gutmann[19] in 1980. Post and Calhoun[3] have performed a study of this model on CRT technology. Del Barco *et al.*[6] have proposed a modification of this model including a black correction. Other studies has been performed on different display technologies.[20,21,23] This model does not consider the channel inter-dependence, but models the chromaticity shift of the primaries. In this section we present this model and the features which characterize it.

Knowing the tristimulus values X, Y, and Z for each primary as a function of the digital input, assuming additivity, the resulting color tristimulus values can be expressed as the sum of tristimulus values for each component (*i.e.*, primary) at the given input level. The black level is removed from all measurements which define the model in order not

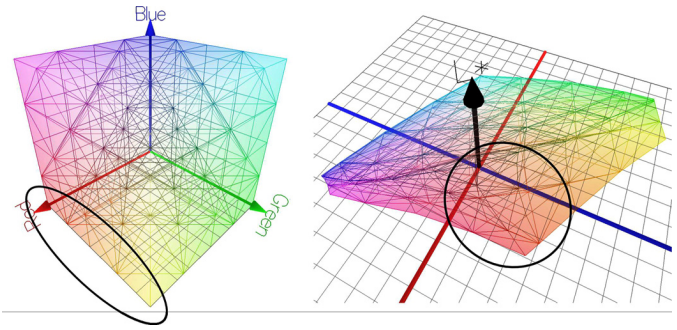


FIGURE 2 — Tetrahedral structure based on a regular grid, N_{rgb} = 15 in RGB (left) and its transform in CIELAB (right).

to include it several times and is added back at the end to return to a correct standard observer color space.[6,20] The model is summarized and generalized in Eq. (1) for N primaries and illustrated in Eq. (2) for a three primaries RGB device, following an equivalent formulation as the one given by Jimenez del Barco *et al.*[6]

For an N primaries device, we consider the digital input to the i-th primary, $d_i(m_i)$, with i an integer $\in [0, N]$, and m_i an integer limited by the resolution of the device (*i.e.*, $m_i \in [0,255]$ for a channel coded on 8 bits). Then, a color XYZ(...,$d_i(m_i)$,...) can be expressed by

$$\begin{aligned} X(\ldots,d_i(m_i),\ldots) &= \sum_{i=0,j=m_i}^{i=N-1}[X(d_i(j))-X_k]+X_k,\\ Y(\ldots,d_i(m_i),\ldots) &= \sum_{i=0,j=m_i}^{i=N-1}[Y(d_i(j))-Y_k]+Y_k,\\ Z(\ldots,d_i(m_i),\ldots) &= \sum_{i=0,j=m_i}^{i=N-1}[Z(d_i(j))-Z_k]+Z_k, \end{aligned} \tag{1}$$

with X_k, Y_k, Z_k the N-tuplet defined by a (0,...,0) input.

We illustrate this for a three-primary RGB device, with each channel coded on 8 bits. The digital input are $d_r(i)$, $d_g(j)$, $d_b(l)$, with i, j, l integers $\in [0,255]$. In this case, a $XYZ[d_r(i), d_g(j), d_b(l)]$ can be expressed by

$$\begin{aligned} X(d_r(i),d_g(j),d_b(l)) &= [X(d_r(i))-X_k]+[X(d_g(j))-X_k]\\ &\quad +[X(d_b(l))-X_k]+X_k,\\ Y(d_r(i),d_g(j),d_b(l)) &= [Y(d_r(i))-Y_k]+[Y(d_g(j))-Y_k]\\ &\quad +[Y(d_b(l))-Y_k]+Y_k,\\ Z(d_r(i),d_g(j),d_b(l)) &= [Z(d_r(i))-Z_k]+[Z(d_g(j))-Z_k]\\ &\quad +[Z(d_b(l))-Z_k]+Z_k, \end{aligned} \tag{2}$$

Previous studies of this model have shown good results, especially on dark and mid-luminance colors. When the colors reach higher luminances, the additivity assumption is less true for CRT technology. Then the accuracy decreases (depending on the device properties). More precisely, Post and Calhoun[3,20] stated that chromaticity error is lower for the PLVC than for the PLCC in low luminances. This is due to the setting of primary chromaticities at maximum intensity in the PLCC. Both models show inaccuracy for

high luminance colors due to channel inter-dependence. Jimenez del Barco *et al.*[6] found that for CRT technology, the higher level of brightness in the settings leads to a non-negligible amount of light for a (0,0,0) input. This light should not be added three times, and they proposed a correction for that. They found that the PLVC model was more accurate in medium-to-high luminance colors. Inaccuracy is more important in low luminances, due to inaccuracy of measurements, and in high luminances, due to channel dependencies. These results are generalized for other technologies in Ref. 23.

The inversion of this model is more troublesome than for matrix-based models such as the PLCC. For a three-primary display, according to Post and Calhoun,[3] it can be performed, defining all subspaces defined by the matrices of each combination of measured data (the intercepts have to be subtracted, and once all the contributions are known, they have to be added). It is also possible to perform an optimization process for each color[6] or to define a grid in RGB which will allow us to perform the inversion using 3-D interpolation. Note that Post and Calhoun have proposed to define a full LUT considering all colors. They stated themselves that it is inefficient. With the method proposed in this paper, we are able to define an optimized reduced grid in RGB which will allow us to inverse this model.

3 Method proposed

Beginning with a regular grid in the destination space, our goal is to transform this regular cubic structure into a rectangular one, in order to make it more uniform or better in the source space. We propose to approximate an homogeneous grid in the source space using only a few number of parameters. Note that this approach ressembles, but differs from the existing methods in the way that we do not modify the grid in the source space but directly and firstly in the destination space. This leads to some differences: first, no gamut mapping is required to map the points which rise up outside the gamut during the creation of a grid in the source space. Secondly, the grid is taken in its globality and we will not reach a perfect uniformity in the source space, only a rough approximation of it, depending on the function we use. Thirdly, the order between data is preserved, which leads to an easier indexation of the tetrahedra.

In order to build a more uniform grid in the source color space, we propose to assume that

- The number of steps has not necessarily to be the same along each primary in the destination space, then $N_r \neq N_g \neq N_b$.
- The steps have not to be regularly spaced along a primary in the destination space, but they can be defined such as $R_i = f_r(d_i)$, $G_j = f_g(d_j)$, $B_k = f_b(d_k)$. Considering $f \in \{f_r, f_g, f_b\}$, a monotonically increasing function from R to R,

$$f: \begin{cases} [0,1] \rightarrow [0,1] \\ d \rightarrow f(d) \end{cases}. \tag{3}$$

f could be any simple function, either a power function with one constant parameter or an S-shaped curve function with one to three constant parameters. Let us say that this function has M parameters, then the number of parameters used to set-up the distribution of the grid points is $P = 3 \times M$.

We make the hypothesis that if we can find the good function f, and the good number of steps, different for each primaries, the structure defined by the 3-D grid will become more homogeneous in the source space and will allow us to have a better inversion map. By extension, we assess that if we can build a "better" grid in the source space, the inversion will become better. The purpose of the next subsection is to define what a better grid is.

3.1 Estimation of the grid quality

A better grid can be defined in various ways. The straightforward way to assume that a grid is better is to assume that it is more regular in the source space, then some geometrical indicators should be used as cost functions to estimate the function f. It could also be a structure well designed for a given application, then an evaluation data set could be required. For a given number of steps along each channel, we have P values to optimize. The optimization process enables us to re-distribute the steps in the destination space, so we adapt the tetrahedral structure in the source space. We propose below three criteria as cost functions. One is directly linked with the geometrical structure of the grid. Others are linked to the result of the inverse model for an evaluation set of patches.

3.1.1 Criterion defined by the geometrical properties of the structure

The first type of cost functions we consider contains the functions directly characterizing the geometry of the grid. The one we have used is the variance of the length of the tetrahedron's edges of the geometrical structure in the source space. If we minimize this function, the tetrahedron of the grid will have more or less the same shape. Using this criterion, we want to reduce the heterogeneity of the interpolation error throughout the space, and so to minimize the error of the model. We call this criterion "Edges" in the following.

3.1.2 Criterion linked with an evaluation data set

The cost functions of the second type affect the structure indirectly. We used an evaluation data set and tried to minimize either the maximum error or the average error of the model for this data set. The evaluation data were 100 meas-

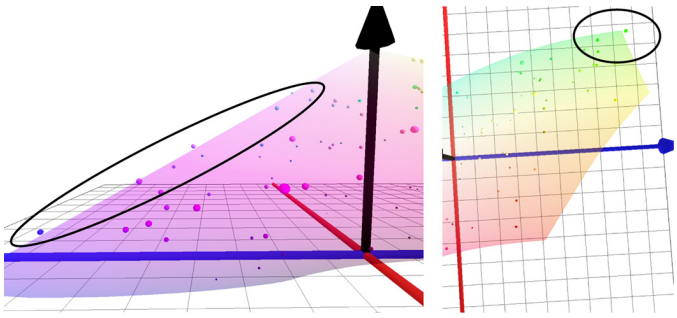


FIGURE 3 — The gamut generated with the PLVC forward model does not correspond exactly with the device gamut. Then some colors inside the true device gamut have to be mapped inside the geometrical structure to be transformed into RGB values. On these figures, one can see that some measured colors defined by some RGB values does not belong to the gamut defined by the forward model.

ured patches equiprobably distributed in the destination space. We call the two criteria "Mean" and "Max" in the following.

A remark has to be done first about the choice of the evaluation data set. As the data are distributed in the destination space, they are not well distributed in the source space. That means that these indicators are not trying to homogenize the grid in the source space. If the goal is to homogenize the grid in the source space, then the data set has to be equiprobably distributed in this space. However, minimizing the feature for an evaluation data set optimizes the grid for instance for a special type of images or for a sequence of a movie for one given device.

A second important thing to notice here is that even if the color of the training (or evaluation) data set belongs to the gamut of the device, since they are defined in RGB, it is not obvious that it belongs to the gamut defined by the forward model (see Fig. 3). Indeed, due to the forward model's failure, some colors could be out of the grid; thus, a gamut mapping is required. For instance, if the forward model does not take into account interaction between channels, some colors could have more (or less) luminance than the forward model can predict and then be outside of the grid.

In our previous study[16] we noticed that the optimization algorithm was trying to refine the grid around these colors where, for some displays, a critical maximum error was coming from the mapping of a color from the evaluation data set. This effect appears especially for the "Max" criteria.

Since the model and the features seem not to be directly related with a derivable function, we have chosen a numerical optimization method. We used the globalized Nelder–Mead simplex downhill algorithm.[18] This fits well with our problem, and we can use it easily with P variables to optimize. Let us notice that *criteria* = $G(p)$ is the function to minimize, with p a vector of dimension P.

To give an exemple, let us consider N_{rgb} = 15, N_r = N_g = N_b = 5, in minimizing the aforementioned functions, we can retrieve the best parameters α, β, γ of a power function, respectively, for each channel R, G, B, which will allow to define our definitive structure (p = [α, β, γ]). In Fig. 4, the cost function "mean" has been used, and the results obtained

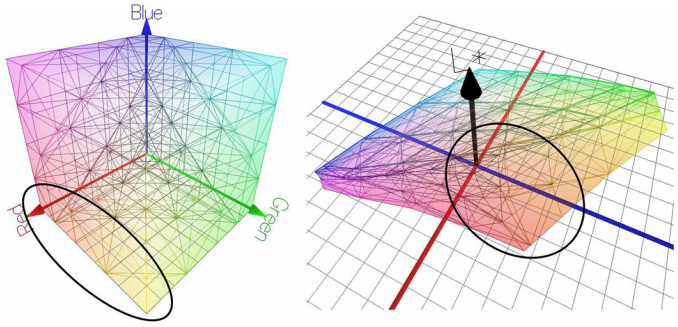


FIGURE 4 — Tetrahedral structure based on a regular grid, N_{rgb} = 15 and α = 1.23, β = 1.27, γ = 1 obtained for the criterion "mean" in RGB (left) and its transform in CIELAB (right).

are a mean error of 4.13% and a maximum error of 18.83% in RGB, for a testing data set of 100 patches, while the original regular grid with the same number of patches gives a mean error of 4.50% and a maximum error of 20.79%.

3.2 Distribution of the grid seeds

In this section, we first discuss the functions we used to modify the distribution of the grid seeds along the R, G, and B axis, with an equal number of data for each axis. We then discuss the number of points that should be used along each axis, considering that these numbers could be different. We separated these steps in two different algorithms to see the influence of each of these. Both are then combined in a practical case in Sec. 4.

3.2.1 Static changing of the grid morphology

To redistribute the grid seeds along each channel, we used either a simple power function or an S-shaped curve. For both cases, we have fixed the condition N_r = N_g = N_b, in order to limit the number of parameters and for modeling purpose. Let us investigate on the criterion used to find the function parameters and on the value of N_{rgb}.

- Power function: the grid seeds are expressed as R_i = ${d_i}^{\alpha}$, G_j = ${d_j}^{\beta}$, B_k = ${d_k}^{\gamma}$. As our digital values are normalized, the power function follows the requirements defined in Eq. (3). There are three parameters to optimize.
- S-shaped curve function: the grid seeds are expressed as

$$\text{if } d_i < 0.5,\ R_i = \frac{(2d_i)^{\alpha}}{2},\ \text{else } R_i = 1 - \frac{\left[2(1-d_i)\right]^{\alpha}}{2},$$

$$\text{if } d_j < 0.5,\ G_j = \frac{(2d_j)^{\beta}}{2},\ \text{else } G_j = 1 - \frac{\left[2(1-d_j)\right]^{\beta}}{2},$$

$$\text{if } d_k < 0.5,\ B_k = \frac{(2d_k)^{\gamma}}{2},\ \text{else } B_k = 1 - \frac{\left[2(1-d_k)\right]^{\gamma}}{2},$$

such that the curve stays inside [0,1]. There are as well three parameters to optimize.

Note that we have chosen a simple expression for the sigmoid, in order to have only one parameter for each channel to optimize. This reduces the freedom of the function, but makes the method easier to explain and understand.

Let's illustrate the method. Using a power function, the grid is defined by N_{rgb} and by the optimization of α, β, γ values. Figure 4 illustrates this construction for $N_{rgb} = 15$, *i.e.*, $N_r = N_r = N_r = 5$. Using the criterion "Mean," we found $\alpha = 1.23$, $\beta = 1.27$, $\gamma = 1$.

3.2.2 Spreading of the grid seeds

An equal number of steps along each R, G, and B channel is not sufficient *a priori* to reach the homogeneity in CIELAB, even if they are not distributed linearly. Thus, we propose a second way to build the geometrical structure which consists in distributing the steps along each channel in the best way whitout the condition $N_r = N_g = N_b$. We used a brute force approach for this task. Given a number of points, we tried the possibilities sequentially such as $N_{rgb} = N_r + N_g + N_b$ and we selected the best combination, considering the chosen cost function. To study the effect independently of the previous proposition, we distributed in a first time the steps regularly along each channel.

4 Results

We have defined two ways to build a geometrical structure considering a given criterion. One is based on the static changing of the morphology of the grid in the destination space. The other focuses on the number of steps along each axis in the destination space. In this section, we present the results obtained for each method developped to study the influence of both distributions. We study these methods in function of the total number of steps used to build the grid. We then present a practical case where the two methods are combined to build the best geometrical structure considering the distribution of the points in the destination space for a given number of points and for a given criterion. In order to analyze the results, we have chosen one forward model, the PLVC which assumes independence between channels, but takes into account the chromaticity shift of primaries. This model has been presented in Sec. 2. Results and analysis considering this model in the forward direction and for the displays we tested can be found in Ref. 23. For the forward model, the A_k, $A \in \{X, Y, Z\}$, were obtained by accurate measurement of the black level. In this work, the $[A(d_i(j)) - A_k]$, were obtained using Akima spline interpolation[22] with the measurement of a ramp along each primary. Note that any 1-D interpolation can be used.

Practically, we used a bounding box centered on the color we wanted to display to determine the tetrahedron it belongs to. Then a linear tetrahedral interpolation is performed.[17] In the case of a point out of the grid, *i.e.*, out of gamut, we performed a simple clipping toward the 50% luminance. Since our data set is defined in RGB, the gamut clipping happens only when the forward model fails.

We show and discuss results obtained for the inversion of the PLVC model. We tested the algorithms on three displays, a tri-LCD projector 3M-X50 (PLCD), a CRT monitor Philips 107s (MCRT), and a LCD monitor DELL 1905FP (MLCD). The measurements of the color patches were taken using the spectroradiometer CS-1000 from Minolta.

4.1 Static changing of the grid morphology

This part discusses results with the first method proposed in Sec. 3.2. We consider an equal number of steps along each primary, and we distribute them using a function which parameters were found using an optimization process. In Fig. 5, one can see the evolution of the accuracy of the model in function of the number of steps used. All combinations between the functions and the criterion used are presented. The reference is the regular grid.

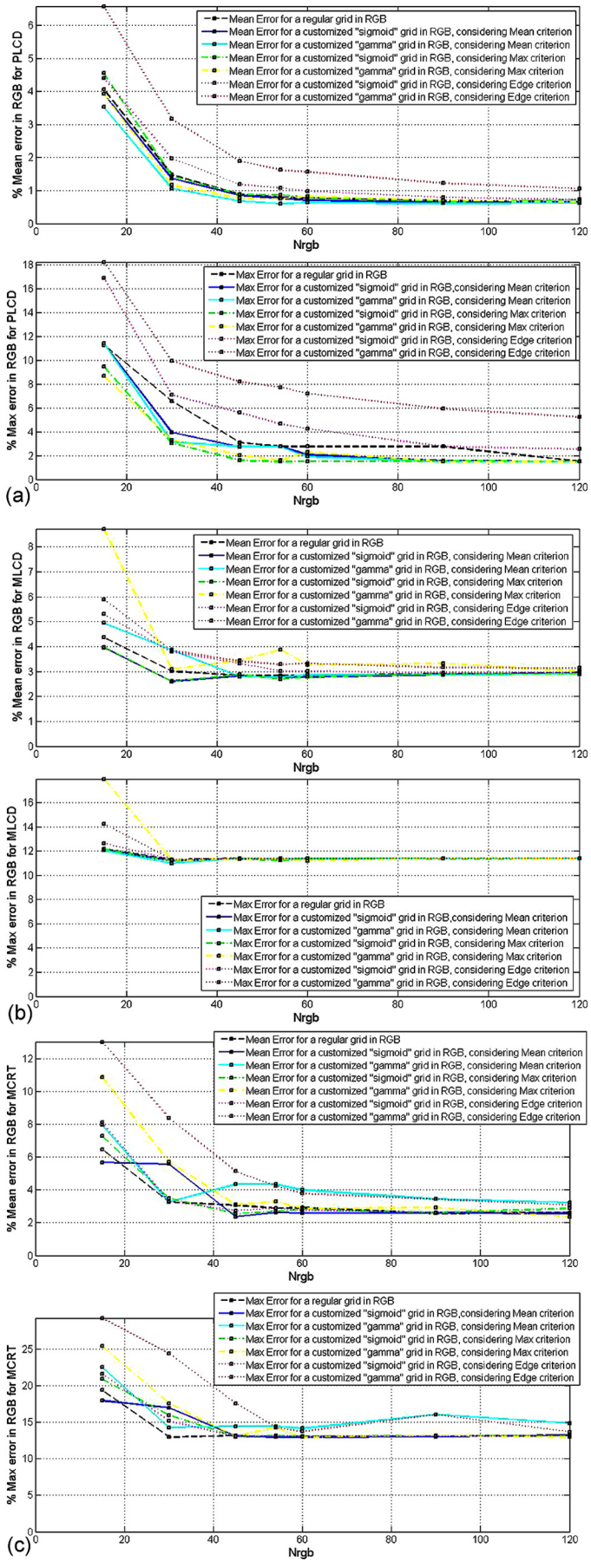
In general, it is hard to say that one criterion combined with one type of function performs much better than the other possibilities.

The criterion "Edges" does not seem to give really good results neither in average nor in maximum error except for the CRT display when it is combined with the S-shaped curve. It is possible that this is the result of the testing data set which was equiprobably spreaded in RGB.

For PLCD [Fig. 5(a)], criteria based on a training data set gives equal or better results than the regular grid, generally following the cost function used. However, one can see that the combination of the criterion "Max" and a power function gives a good compromise in both the average and maximum error for this display. This combination performs better than a regular grid. The difference is major for a reduced grid; for instance, with $N_{rgb} = 30$, the regular grid gives an average error of about 1.5%, the "Edges" criterion an error of 2% to over 3% depending on the function used, while all the other methods with the "Mean" and the "Max" criteria give an average error of 1–1.5%. If we look at the maximum error for the same N_{rgb}, the regular grid gives an error over 6.5%, the refinement of the grid using the training data set gives an error of 3–4%. That means that we reduced the maximum error by a factor of two for this case. The criterion "Edges" gives a maximum error higher than the error of the regular grid.

For MLCD [Fig. 5(b)], we notice the fast convergence of the maximum error for all methods (the regular grid included). This is possibly due to the gamut mapping due to the failure of the forward model. For this device and $N_{rgb} = 30$, we notice that the sigmoid used gives better results than the regular grid (from around 3–2.6%) whatever the "Max" or "Mean" criterion we used.

MCRT [Fig. 5(c)] shows that, on this device, the regular grid gives better results than the others and that "Edges" criterion seems to perform well on it, combined with a S-



curve. However, if we look at the results for N_{rgb} = 45, the sigmoid combined with the criteria based on the training data set gives better results, reducing the error from 3% to 2.4 and 2.5%.

When the number of points increases, all methods seem to converge. All are going towards the same accuracy. Note that the "Edges" criterion combined with a power-shaped curve converge more slowly than the other for PLCD.

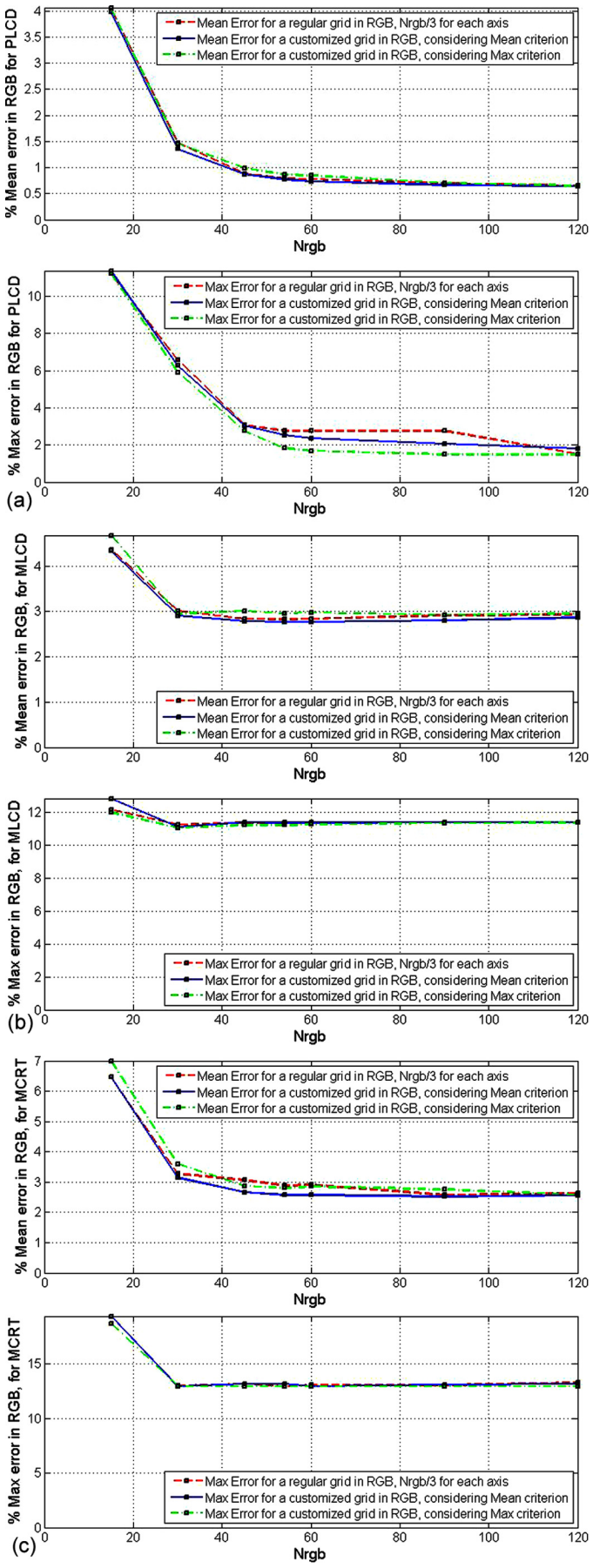
Looking at these data, we can not say in general: "this criterion and this function is the best combination for any display." In particular, for a given amount of steps, a difference appears, but nothing that can be generalized.

The algoritm can fall into local minima for the "Mean" and more often for the "Max" criteria, depending on the seeds given for the optimization. However, the "Edges" criterion is very stable despite a not so good general result. The fall into local minima explains the shape of the curves which are not, in many cases, decreasing with increasing N_{rgb}.

4.2 Spreading of the grid seeds

This subsection discusses results obtained with the second method proposed in 3.2. Considering the total number of steps, we distributed them along each channel such that the cost function is minimized. We do not discard this criterion, however, because it can be valuable for other displays or other evaluation data sets. We do not put aside that this criterion can be valuable for other displays or other evaluation data set. Looking at Fig. 6, it appears that the "Mean" criterion performs slightly better than the "Max," since the convergence of the maximum error is fast. However, for PLCD [Fig. 6(a)], the average error remains quite similar for the three methods used, and, a better compromise provided by the "Mean" criterion, the "Max" criterion permits a slight reduction in the maximum error (from twice to once if we look at a grid of N_{rgb} = 60, comparing it with the regular grid). For MLCD and MCRT [Figs. 6(b) and 6(c)], the maximum error converges fast, and the "Max" criterion seems to be weak. Looking at the results for the "Mean" criterion, one can say that it follows the regular grid for MLCD. However, it decreases the average error for a mid-sized grid of N_{rgb} = 45 with MCRT from 3.2 to 2.7%.

FIGURE 5 — Results obtained with the geometrical model in function of the total number of steps used. The three cost-function Mean, Max, and Edges and the two functions proposed were used on the three displays: (a) PLCD, (b) MLCD, and (c) MCRT. The mean error in percentage in RGB is shown on the upper part of each figure, the maximum error is shown on the bottom. One can see that the sigmoid function with the "Mean" criterion performs generally better than the others for both the maximum and the mean error.



4.3 A practical case

In this section, we propose to evaluate a practical case, combining both approaches previously defined. We fixed a number of steps, $N_{rgb} = 90$, which appears to us to be a good compromise between the size of the structure and the optimization facilities, *i.e.*, smaller, it is a loss of accuracy for nothing, bigger, there will be no interest in optimizing it, then a regular grid should be enough compared with the cost of the optimization. We retrieve the best function to define each axis of the destination space, considering the different criteria. We have chosen to select the best one with regard to the mean error estimation, but it is also possible to consider that the maximum error is more important, depending on the application. We keep both the power and the sigmoid functions. We then look for the best spreading of steps along channels, still with regard to the mean error of the model.

The result gives us the optimized geometrical structure for this given number of points and for the display considered. We compare it with the regular grid which is still our reference. Results of the inverse model estimation are presented in Table 1.

This table shows that simply using the same amount of steps for each channel and a power or a sigmoid function to distribute them could be pretty efficient and could be the best solution for some displays, such as an MLCD where it is the best grid. In this case, we reduce the error by 0.28%, which is relatively a gain of accuracy of almost 10%. Some accuracy can be won in spreading the steps after having applied the function, such as for a PLCD and MCRT. For the first one, we go from 0.69% with a regular grid to 0.66% using a power to 0.59% in spreading the steps, which is a relative improvement of around 15.5%. Not that for this display, before spreading, the best function to step the points is the sigmoid with an error of 0.64%, but after spreading the best result is given by the power function. In other words, the best function should not be chosen at the first step of the algorithm. For MCRT, we go from 2.58 to 2.54 using a power-shaped function to 2.50 in spreading the steps, showing a relative gain of 3%.

The same analysis can be performed for the maximum error. In our data, the smaller maximum error found is when we use the same method, but it is not obvious that this will happen all the time. For instance, looking at a MCRT with a sigmoid function and a spreading, the errors mean and

FIGURE 6 — Results obtained with the geometrical model in function of the number of steps used for the three displays: (a) PLCD, (b) MLCD, and (c) MCRT tested. The two cost functions Mean and Max were used to find the best number of steps along each channel. The mean error in percentage in RGB is shown on the upper part of the figure, the maximum error is shown on the bottom.

TABLE 1 — A practical case, given $N_{rgb} = 90$, considering the criterion "Mean" and the different functions, we find the best grid for minimizing the average error for the three displays studied.

% Error in RGB	Regular grid		Gamma grid		Sigmoid grid		Linear +spreading		Gamma +spreading		Sigmoid +spreading	
$I_{rgb} = 90$	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
PLCD	0.69	2.76	0.66	2.76	0.64	1.53	0.66	2.06	**0.59**	**1.22**	0.62	1.44
MLCD	2.91	11.40	2.87	11.40	**2.63**	**11.14**	2.81	11.41	2.86	11.41	**2.63**	**11.14**
MCRT	2.58	13.05	2.54	13.07	2.59	13.07	2.53	13.07	**2.50**	**13.00**	2.57	13.15

TABLE 2 — Spreading of the grid seeds along each channel for the practical case studied.

Seeds distribution	Linear + spreading			Gamma + spreading			Sigmoid + spreading		
$I_{rgb} = 90$	I_r	I_g	I_b	I_r	I_g	I_b	I_r	I_g	I_b
PLCD	23	34	33	19	47	24	19	43	28
MLCD	49	21	20	40	31	19	30	30	30
MCRT	39	21	30	36	32	22	34	37	19

maximum are 2.57 and 13.15, albeit the results for the linear distribution are 2.58 and 13.05.

To summarize, with the displays we studied, we found a better LUT than the regular one in all the cases with a different gain in accuracy. We do not put aside that the regular grid can be the best one for a given display.

In Table 2, we show the spreading of the steps along each axis. For the PLCD, it seems that the green channel requires a more accurate discretization than the two others, but looking at the other displays, this can not be generalized. It appears that in using a linear distribution or a power-shaped one, the spreading takes quite the same shape, with different strengths. While in using a sigmoid, the spreading is totally different. Note that for the MLCD, the best spreading scheme is the equibalanced one. In general, it seems that no spreading scheme can be recommended *a priori*.

To conclude with the obtained results, no *a priori* function or spreading scheme can be recommended. It has to be device dependent. Furthermore, it seems that it is not technology dependent, since the two LCDs show different behavior (but one is a projector, the other a monitor, so no hasty judgement should be taken). It appears that the best of the grids defined with our methods should be selected with a test, and actually this is what we do to assess their quality.

5 Conclusion

We proposed some methods to build an optimized geometrical structure in order to invert any display color-characterization forward model. We used several criteria linked with the grid itself or with an evaluation data set. We focused on how to distribute the steps on the destination space axis in order to have the better grid in the source space, both in using a power or a sigmoid function and in spreading the steps in a non-uniform way along the axis. We studied a practical case, considering a forward model, a number of steps, and a feature to optimize for three displays. The results obtained with the proposed methods in the practical case are all better than the regular grid, and the relative gain in accuracy is not negligible, from 3% in the worst case to up to 15.5% in the best case.

However, some further work could be done concerning on the criteria used, especially for the ones related to the grid features, and on the training data set. Moreover, since the function used to distribute the steps on the axis could have more parameters, we are thinking about the function of higher order (monotonically increasing) or simply a more-complex S-shaped curve.

We also do believe that this method to build and select the optimized structure to invert the colorimetric model can be applied to any output color device, such as printers.

References

1 H. Widdel and D. L. Post, eds., *Color in Electronic Displays* (Plenum Press, New York, 1992).

2 W. Cowan and N. Rowell, "On the gun independency and phosphor constancy of color video monitor," *Color Research & Application* **11**, S34–S38 (1986).

3 D. L. Post and C. S. Calhoun, "An evaluation of methods for producing desired colors on CRT monitors," *Color Research & Application* **14**, 172–186 (1989).

4 R. S. Berns, R. J. Motta, and M. E. Gorzynski, "CRT colorimetry. Part I: Metrology," *Color Research & Application* **18**(5), 299314 (1993).

5 R. S. Berns, M. E. Gorzynski, and R. J. Motta, "CRT colorimetry. Part II: Metrology," *Color Research & Application* **18**(5), 315–325 (1993).

6 L. Jimenez Del Barco, J. A. D´ýaz, J. R. Jimenez, and M. Rubino, "Considerations on the calibration of color displays assuming constant channel chromaticity," *Color Research & Application* **20**, 377–387 (1995).

7 N. Katoh, T. Deguchi, and R. Berns, "An accurate characterization of CRT monitor (i) verification of past studies and clarifications of gamma," *Opt. Rev.* **8**(5), 305–314 (2001).

8 N. Katoh, T. Deguchi, and R. Berns, "An accurate characterization of CRT monitor (II) proposal for an extension to CIE method and its verification," *Opt. Rev.* **8**(5), 397–408 (2001).

9 N. Tamura, N. Tsumura, and Y. Miyake, "Masking model for accurate colorimetric characterization of LCD," *J. Soc. Info. Display* **11**(2), 333–339 (2003).

10 P. Green and L. MacDonald, eds., *Color Engineering* (Wiley, Chichester, U.K., 2002).

11 S. Wen and R. Wu, "Two-primary cross-talk model for characterizing liquid crystal displays," *Color Research & Application* **31**(2), 102–108 (2006).

12 Y. Kwak and L. MacDonald, "Characterisation of a desktop LCD projector," *Displays* **21**(5), 179–194 (2000).
13 Y. Kwak, C. Li, and L. MacDonald, "Controling color of liquid-crystal displays," *J. Soc. Info. Display* **11**(2), 341–348 (2003).
14 B. Bastani, B. Cressman, and B. Funt, "An evaluation of methods for producing desired colors on CRT monitors," *Color Research & Application* **30**(6), 438–447 (2005).
15 J. Stauder, J. Thollot, P. Colantoni, and A. Tremeau, "Device, system and method for characterizing a colour device," *European Patent WO/2007/116077* (October 2007).
16 J.-B. Thomas, P. Colantoni, J. Y. Hardeberg, I. Foucherot, and P. Gouton, "An inverse display color characterization model based on an optimized geometrical structure," *Color Imaging XIII: Processing, Hardcopy, and Applications, SPIE*, 68070A-1-12 (2008).
17 L. M. Kasson, S. I. Nin, W. Plouffe, and J. L. Hafner, "Performing color space conversions with three-dimensional linear interpolation," *J. Electron. Imaging* **4**(3), 226–250 (1995).
18 P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization* (Academic Press, 1982).
19 W. W. Farley and J. C. Gutmann, "Digital image processing systems and an approach to the display of colors of specified chrominance," *Technical report HFL-80-2/ONR-80, Virginia Polytechnic Institute and State University*, Blacksburg, VA (1980).
20 D. L. Post and C. S. Calhoun, "Further evaluation of methods for producing desired colors on CRT monitors," *Color Research & Application* **25**, 90–104 (2000).
21 J.-B. Thomas, J. Y. Hardeberg, I. Foucherot, and P. Gouton, "Additivity based LC display color characterization," *Proc. Gjøvik Color Imaging Symposium* **4**, 50–55 (2007).
22 H. Akima, "A new method of interpolation and smooth curve fitting based on local procedures," *J. Assoc. Computing Machinery* **17**, 589–602 (1970).
23 J-B. Thomas, J. Y. Hardeberg, I. Foucherot, and P. Gouton, "The PLVC color characterization model revisited," *Color Research & Application* **36** (2008).
24 I. Amidror, "Scattered data interpolation methods for electronic imaging systems: A survey," *J. Electron. Imaging* **11**(2), 157–176 (2002).
25 F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Analysis Machine Intelligence* **11**(6), 567–585 (1989).
26 G. M. Nielson, H. Hagen and H. Muller, "Scientific visualizations, overviews, methodologies, and techniques," *Scientific Visualization*, IEEE Computer Society (1997).
27 M. Stokes, "Method and system for analytic generation of multi-dimensional color lookup tables," *U.S. Patent 5612902* (1997).
28 S. Dianat, L. K. Mestha, and A. Mathew, "A dynamic optimization algorithm for generating inverse printer map with reduced measurements," *IEEE ICASSP* **3** (2006).
29 R. E. Groff, D. E. Koditschek, and P. P. Khargonekar, "Piecewise linear homeomorphisms: The scalar case," *IEEE Intl. Conf. Neural Networks* **3**, 259–264 (2000).
30 J. Z. Chang, "Sequential linear interpolation of multi-dimensional functions," *IEEE Trans. Image Processing* **6**, 1231–1245 (1997).
31 R. Balasubramanian and M. S. Maltz, "Refinement of printer transformation using weighted regression," *Proc. SPIE* **2658**, 334–340 (1996).
32 D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," *Proc. ACM*, 517–524 (1968).
33 D. Viassolo and L. K. Mestha, "A practical algorithm for the inversion of an experimental input-output color map for color correction," *J. Opt. Eng.* **42**(3) (2003).

Jean-Baptiste Thomas received his B.S. degree in applied physics (2004) and his M.S. degree in image, vision, and signal (2006) from the University of Saint-Etienne, France. He is currently pursuing his Ph.D. degree at both the University of Bourgogne, France, and the Gjøvik University College, Norway. His research focuses on colors in projection and tiled projection displays.

Philippe Colantoni is Maitre de conferences (Senior lecturer) at the "Universite Jean Monnet de Saint-Etienne" and Associated Researcher to the C2RMF (Centre de Recherche et de Restoration des Musées de France) since 2006. His research areas include computer vision, color science, multi-spectral imaging, general-purpose computation using graphics hardware, 3-D visualization, and color management.

Jon Y. Hardeberg is a Professor of Color Imaging at Gjøvik University College. He received his Ph.D. from Ecole Nationale Supérieure des Télécommunications in Paris, France, in 1999, with a dissertation on color-image acquisition and reproduction, using both colorimetric and multispectral approaches. He has more than 10 years experience with industrial and academic color-imaging research and development and has co-authored over 100 research papers within the field. His research interests include various topics of color-imaging science and technology, such as device characterization, gamut visualization and mapping, image quality, and multispectral image acquisition and reproduction. He is a member of IS&T, SPIE, and the Norwegian representative to CIE Division 8. He has been with Gjøvik University College since 2001 and is currently head of the Norwegian Color Research Laboratory.

Irene Foucherot received her Ph.D. in computer sciences from the University of Burgundy, France, in 1995. She is an Assistant Professor at the University of Burgundy since 1996. Her precedent research domain was artificial life (genetic algorithms, cellular automata). She joined the Image Processing Group of the LE2I (Laboratoire d'Electronique, Informatique et Image : Laboratory of Electronics, Computer Sciences, and Image) in 2001. Her research interest now involves color-Image processing and especially region-based segmentation of multi-spectral images.

Pierre Gouton obtained his Ph.D. in components, signals, and systems at the University of Montpellier (France) in 1991. From 1988 to 1992, he has worked on passive power components at the Laboratory of Electric Machine of Montpellier. Appointed Assistant Professor in 1993 at the University of Burgundy, France, he joined the Image Processing Group of the LE2I (Laboratoire d'Electronique, Informatique et Image: Laboratory of Electronics, Computer Sciences, and Image). Since then, his main research involves the segmentation of images by linear methods (edge detector) or nonlinear methods (mathematical morphology, classification). He is a member of ISIS (a research group in signal and image processing of the French National Scientific Research Committee) and also a member of the French Color Group. Since December 2000, he is an incumbent of the HDR (Habilitation à Diriger les Recherches : Enabling to Manage Researches).