An inverse display color characterization model based on an optimized geometrical structure

Jean-Baptiste Thomas[a,b], Philippe Colantoni[c], Jon Y. Hardeberg[b],
Irène Foucherot[a] and Pierre Gouton[a]

[a]Laboratoire Electronique, Informatique et Image, Université de Bourgogne, Dijon, France;
[b]The Norwegian Color Research Laboratory, Gjøvik University College, Gjøvik, Norway;
[c]Centre de Recherche et de Restauration des Musées de France, Palais du Louvre, Paris, France

ABSTRACT

We have defined an inverse model for colorimetric characterization of additive displays. It is based on an optimized three-dimensional tetrahedral structure. In order to minimize the number of measurements, the structure is defined using a forward characterization model. Defining a regular grid in the device-dependent destination color space leads to heterogeneous interpolation errors in the device-independent source color space. The parameters of the function used to define the grid are optimized using a globalized Nelder-Mead simplex downhill algorithm. Several cost functions are tested on several devices. We have performed experiments with a forward model which assumes variation in chromaticities (PLVC), based on one-dimensional interpolations for each primary ramp along X, Y and Z ($3 \times 3 \times 1 - D$). Results on 4 devices (2 LCD and a DLP projection devices, one LCD monitor) are shown and discussed.

Keywords: Display device, color characterization, inverse model, tetrahedral structure, optimization

1. INTRODUCTION

This work defines a color characterization algorithm for additive color displays. Color characterization aims to model the relationship between digital values input to the device (RGB) and the displayed color (in a device-independent color space based on CIE colorimetry, such as CIEXYZ or CIELAB). We focus here on the inverse model which aims to estimate the digital value that we need to input to the device in order to display the desired color sensation. To limit the number of measurements, we have fully based our inverse transform on the forward model. Using this model, we can define a tetrahedral structure, which defines our transform.

The Piecewise Linear interpolation assuming Variation in Chromaticities (PLVC) display characterization model has been defined by Post and Calhoun,[1] and studied in[1–3] for CRT technology. It has been shown that good results can be achieved. But since the model inversion is not straightforward and since CRT primaries are not so much varying (at least after black level correction), the benefits of using the PLVC model is not so important with CRT technology.

For LCD technology, it is different, since the chromaticities are varying with the input voltage.[4] A previous study has confirmed that the result is significantly more accurate for this technology, compared with the more common Piecewise Linear assuming Constant Chromaticities (PLCC) or Gain-Offset-Gamma-Offset (GOGO) models.[5]

The PLVC model is not directly invertible analytically.[1] However, it is possible to build a geometrical inverse model based on a tetrahedral structure generated using the forward model. This article proposes and discusses an algorithm to build such a structure. In order to yield this structure, one can define a regular 3-D grid in the destination color space (RGB). This grid defines cubic voxels. Each one can be split in 5 tetrahedra. This tetrahedral shape is preserved within the transform to either CIEXYZ or CIELAB. We can then generalize the model to the entire space, using tetrahedral interpolation.[6]

The straightforward way to build such a grid requires a large number of measurements. If the grid is defined in the destination space, one measurement for each point is needed. If it is defined in the source space, an

Send correspondence to J-B Thomas, e-mail: jib.thomas@gmail.com

Color Imaging XIII: Processing, Hardcopy, and Applications, edited by Reiner Eschbach,
Gabriel G. Marcu, Shoji Tominaga, Proc. of SPIE-IS&T Electronic Imaging,
SPIE Vol. 6807, 68070A, © 2008 SPIE-IS&T · 0277-786X/08/$18

optimization process with several measurements is needed to converge to each point of the grid. In our approach, we aim to limit this number. We therefore propose to use the forward model to define the 3-D structure.

This structure is based on cubic voxels in the destination space (RGB). As the forward transform is neither uniform nor linear, defining a regular structure in the source space leads to heterogeneous interpolation errors, i.e. the interpolation accuracy is not constant in different areas of color space. In CIELAB, for instance, the size of the tetraheda is not the same anymore, we thus have less information for the same perceptual difference. Our goal is to transform this regular cubic structure into a rectangular one. We propose to approximate a regular grid in the source space (CIELAB) using only a few number of parameters. Note that in the following, we denote destination space the device-dependent RGB color space, and the source space the device-independent color-space used (either CIEXYZ or CIELAB).

In the following, we describe the forward model we used, we explain our proposal for building the optimal grid, and finish with considerations about the cost functions optimized to assess that the grid is the best one. Results for 4 display devices are presented. We believe that our algorithm could be beneficial for any numerical forward model which is difficult to invert. Further work is required to find exactly when such an approach is useful, which cost functions would give the best result, and which functions have to be used to build the grid in the destination space as well as how many points for each channel are needed.

2. FORWARD MODEL

Although in principle any forward model could be used, in our approach we propose to use a variant of the PLVC model proposed by Post and Calhoun,[1] including a black correction and a non-linear interpolation. Further studies with this model has been performed on CRT,[2,3] and recently on LCD[5] technologies. In this section, we describe and discuss the principles of this model, and present some experimental results using it.

Knowing the tristimulus values X, Y, and Z for each primary as a function of the digital input value, assuming additivity, the resulting color's tristimulus values can be expressed as the sum of tristimulus values for each component (primary) at the given input level. Note that in order not to add several times the black level, we remove it from all measurements used to define the model. Then, it's added to the result, to return to a correct standard observer color space.[2,3] The model is summarized in Equation (1) for N primaries, and illustrated in Equation (2) for a three primaries RGB device, following the formula given by Jimenez del Barco et al.[3]

For an N primaries device, let's say that the digital input to the N primaries are $d_i(m_i)$, 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:

$$\begin{array}{l} X(..., d_i(m_i), ...) = \sum_{i=0, j=m_i}^{i=N-1} [X(d_i(j)) - X(d_k)] + X(d_k) \\ Y(..., d_i(m_i), ...) = \sum_{i=0, j=m_i}^{i=N-1} [Y(d_i(j)) - Y(d_k)] + Y(d_k) \\ Z(..., d_i(m_i), ...) = \sum_{i=0, j=m_i}^{i=N-1} [Z(d_i(j)) - Z(d_k)] + Z(d_k), \end{array} \quad (1)$$

with d_k the N-tuplet defined by $d_k = (0, ..., 0)$.

Let's illustrate this for 3 primaries RGB device, with each channel 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, $XYZ(d_r(i), d_g(j), d_b(l))$ can be expressed by:

$$\begin{array}{l} X(d_r(i), d_g(j), d_b(l)) = [X(d_r(i)) - X(d_k)] + [X(d_g(j)) - X(d_k)] + [X(d_b(l)) - X(d_k)] + X(d_k) \\ Y(d_r(i), d_g(j), d_b(l)) = [Y(d_r(i)) - Y(d_k)] + [Y(d_g(j)) - Y(d_k)] + [Y(d_b(l)) - Y(d_k)] + Y(d_k) \\ Z(d_r(i), d_g(j), d_b(l)) = [Z(d_r(i)) - Z(d_k)] + [Z(d_g(j)) - Z(d_k)] + [Z(d_b(l)) - Z(d_k)] + Z(d_k) \end{array} \quad (2)$$

All our tested devices were RGB primary devices, thus the relationship between digital RGB values and RGB device's primaries is as direct as possible. They are listed in Table 1.

We used 18 measurements by channel, including a measurement of the black level each time. The $A(d_k)$, $A \in \{X, Y, Z\}$ has been obtained, averaging 3 measurements of the black level. The $[A(d_i(j)) - A(d_k)]$, are

Table 1. References of the tested devices.

Device reference	Related in the text as :
Panasonic PT-AX100E	ProjectorLCD1
3M-X50	ProjectorLCD2
ProjectionDesign Action One	ProjectorDLP
DELL monitor	MonitorLCD

obtained by one dimensional interpolation with the measurement of a ramp along each primary. Any 1-D interpolation method could be used; we have used Akima spline interpolation.[7]

Previous studies of the PLVC model have shown good results, especially on dark and mid-luminance colors. When the colors reach higher luminances, the additivity assumption is less true, and the accuracy decreases (depending on the device properties). More precisely, Post and Calhoun[1,2] stated that chromaticity error are lower for PLVC than for PLCC in low luminances. This is due to the setting of primary's chromaticities at maximum intensity in PLCC. Both models show inaccuracy for high luminances color due to inter-channel dependence. Jimenez del Barco et al[3] found that for CRT technology, the high level of brightness in the settings leads to an amount of light for a (0,0,0) input. This light should not be added three times, and a correction has to be done 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 inaccuracies in measurements, and in high luminances, due to channel dependences. In LCD technology this amount of light comes from the failure of the LCD panel to stop all the light. For DLP, it is due to the combination of the imperfection of the black body, which is supposed to absorb unwanted light, and the lens system.

The results of the forward model for tested devices are presented in Table 2, we are also giving the values obtained with a black corrected PLCC model.

Table 2. Results for the PLVC forward model, in comparison with the black corrected PLCC model. For LCD technology devices, significant improvements are obtained.

ΔE^*_{ab} for a set of 100 random patches		PLCC	PLVC
ProjectorLCD1	Mean	3.93	1.32
	Max	8.28	3.16
ProjectorLCD2	Mean	1.78	0.56
	Max	2.96	1.66
ProjectorDLP	Mean	0.99	0.91
	Max	2.17	2.85
MonitorLCD	Mean	4.88	2.20
	Max	9.36	5.26

The inversion of this model is more troublesome. According to Post and Calhoun,[1] it could be performed by defining all subspaces defined by the matrices of each combination of measured data. It is also possible to perform an optimization process for each color, 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 also proposed to define a full look up table (LUT) considering all colors, but they stated themselves that this is inefficient. We will thus define a 3-D LUT, and perform 3-D linear interpolations. In the following, we propose a method to build an optimal grid for this purpose.

3. INVERSE MODEL

In this section, we describe our inverse model. We have set up an optimization process which allows us to yield a 3-D grid with respect to one choosen feature. This grid is fully based on the forward model, and once it has been set up, it permits to find any digital values to input to the device in order to display any wanted color (within device gamut limitations). The different aspects of our algorithm are described in the following. First we explain the framework we used to define such a grid. We discuss then the characteristics taken into account to perform the optimization.

3.1 Design of the 3-D grid

A number of forward color characterization models are not invertible directly analytically. Obviously the full empirical models belong to this category. But some numerical models which could make a really good predictions with a few number of measurements are not used in practice because of the difficulties faced to invert them. Nowadays, 3-D interpolations based on LUT can be performed faster and faster as long as the LUT has been built. We propose to use these good forward models to build a 3-D LUT in order to perform the inversion.

3.1.1 General framework

The simplest way to define such a grid is to take directly the measurement points as seeds, and to fill up the rest of the destination space. A tetrahedral structure can be built using these points. We can then use the forward model to transform these points to CIELAB, the source space. This structure can be used to retrieve any RGB value needed to display a specific color inside the device's gamut. To define a more custom grid one can take points equally distributed along each digital R, G and B axis (Figure 1). The more points you take, the more your tetrahedrons will be small and the interpolation accurate.

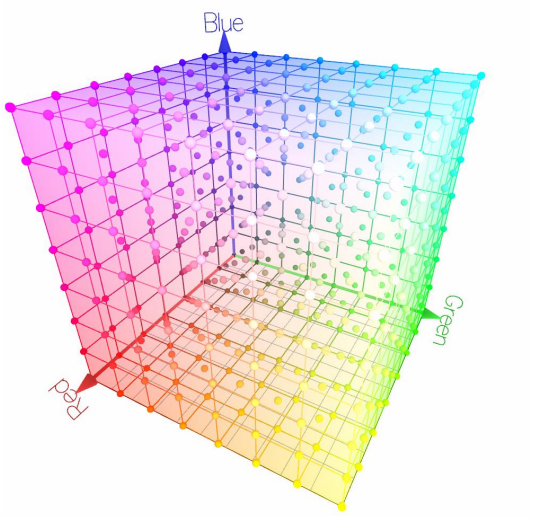

Figure 1. Design of a regular grid with $N_{rgb} = 8$ in RGB.

Such a grid is characterized by the number of patches along each channel N_r, N_b, N_g, such that $N_r = N_g = N_b$, and each vertex is defined by $V_{i,j,k} = (R_i, G_i, B_i)$, $R_i, G_j, B_k \in [0, 1]$, considering $R_i = d_i, G_j = d_j, B_k = d_k$ with $d_i, d_j, d_k \in [0, 1]$ normalized digital values possible. And $i \in [0, N_r - 1]$, $j \in [0, N_g - 1]$, $k \in [0, N_b - 1]$, integers which are the indexes of the seeds of the grid along each primary.

Once this grid has been built, we can define the tetrahedral structure.[6] Then we use the forward model to transform the structure into CIELAB colorspace (see Figure 2). We have thus built an inverse model.

Now we can make a remark, according to the non-linearity of the CIELAB transform, the size of the tetrahedron is not anymore the same as it was in RGB. Thus we propose to modify this framework to make this grid more homogeneous in the source colorspace where we perform the interpolation.

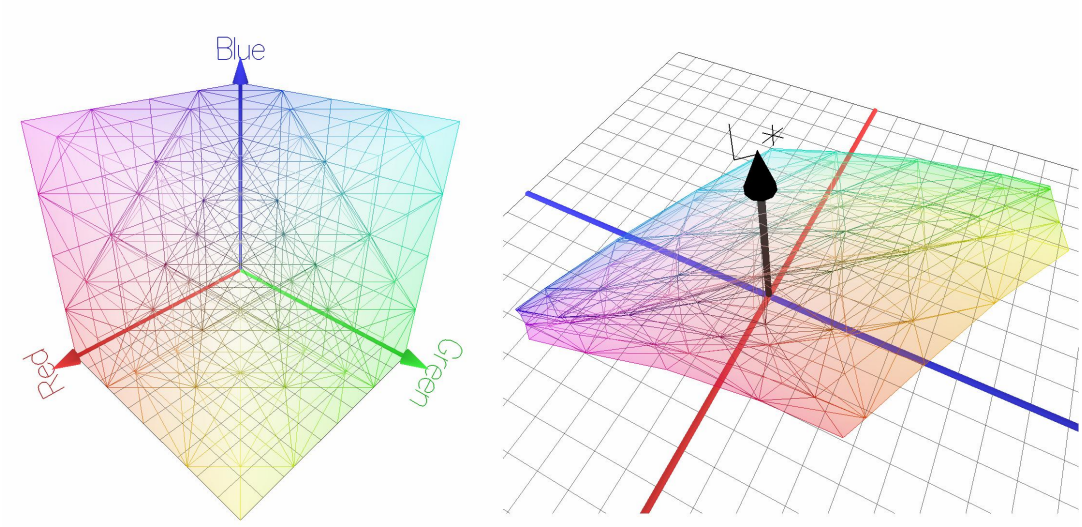

Figure 2. Tetrahedral structure based on a regular grid, $N_{rgb} = 5$ in RGB (left) and its transform into CIELAB (right).

In order to build a more uniform grid in the source colorspace, we propose to modify the previous framework as follows:

- The number of patches along each primary have not necessarily to be equal, then $N_r \neq N_g \neq N_b$.
- The patches have not to be regularly spaced along a primary in the destination space, Then $R_i = f(d_i), G_j = f(d_j), B_k = f(d_k)$. Considering f a monotonically increasing function from $\mathcal{R}$ to $\mathcal{R}$:

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

  f could be any simple function, either a gamma with one constant parameter, or a S-shape curve function with one to four constants parameters.

We make the hypothesis that if we can find the good parameters for the function f, and the good number of patches for each of the primaries, the structure defined by the 3-D grid will become more homogeneous, and will allow us to have a better inversion. In order to find these parameters we will use an optimization process.

3.1.2 Framework of our experiment

As far as we know, such an approach has never been studied, and for a first work, we have fixed some conditions, in order to limit the number of parameters to find. We therefore defined the framework of our experiment as:

- We used the condition $N_{rgb} = N_r = N_g = N_b$ the number of patches by primary, and we will perform more investigation on the value of N_{rgb}.
- We used only a gamma function to redistribute the vertices of the grid $R_i = d_i^{\alpha}, G_j = d_j^{\beta}, B_k = d_k^{\gamma}$. As our digital values are normalized, the gamma function follows the requirements defined in Equation 3.

At this point our grid will be defined by N_{rgb} and by the optimization of α, β, γ values. Figure 3 illustrates this construction for $N_{rgb} = 5$, and $\alpha = 1.23, \beta = 1.27, \gamma = 1$.

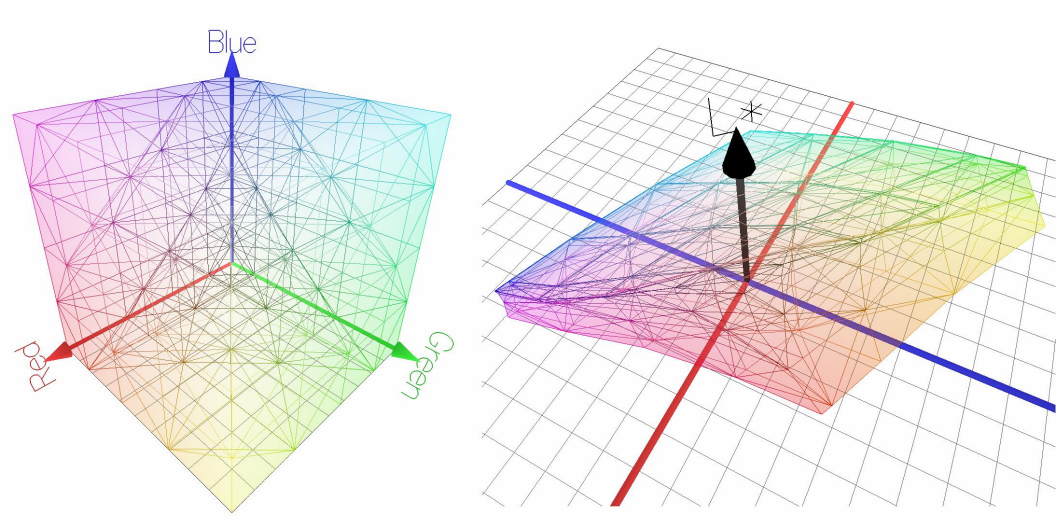

Figure 3. Tetrahedral structure based on a regular grid, $N_{rgb} = 5$ and $\alpha = 1.23, \beta = 1.27, \gamma = 1$ obtained for the criterion "mean" in RGB (left) and its transform into CIELAB (right).

3.2 Proposed optimization criteria

We have three values to optimize. The optimization process enables us to re-distribute the patches in the destination space, so we adapt the tetrahedral structure in the source space. In this section we propose 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 a training set of patches.

3.2.1 Criterion defined by the geometrical properties of the structure

The first type of cost functions are the functions which are directly characterizing the geometry of the grid. The cost function we have tested is the variance of the length of the tetrahedron's edges in the source colorspace. 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, then minimize the maximum error of the model. We call this criterion "edges" in the following.

3.2.2 Criterion linked with a training data set

The second type of cost functions affects the structure indirectly. We used a training data set, and tried to minimize either the maximum error or the average error of the model for this data set. The training data where 100 measured patches equiprobably distributed in the destination colorspace. We call the two criteria "mean" and "max" in the following. A remark has to be done about the choice of the training set. As the data are distributed in the destination space, they are not at all well distributed in the source space. That means that these indicators are not trying to homogeneise the grid in the source space. But minimizing the feature for a training data set can optimize the grid for a special type of images, or for a sequence of a movie for instance for one given device.

As 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.[8] This seems to fit well with our problem, and we can use it easily to N variables to optimize for further studies. In minimizing the aforementioned functions, we can retrieve our best parameters α, β, γ which will permit to define our definitive structure. For instance, in Figure 3, the cost function "mean" has been used, and the results obtained are a mean error of 0.041337 and a maximum error of 0.188310 in % in RGB, for a data set of 100 patches, and for the α, β, γ cited before for this grid, while the original regular grid gives a mean error of 0.045085 and a maximum error of 0.207900.

3.3 Interpolation using the tetrahedral structure, and gamut mapping considerations

Now that we have the position of the points of the grid, we can use it to perform the transform to the destination space. We are using a bounding box centered on the color we want to display to determine the tetrahedron it belongs to. Then a linear tetrahedral interpolation is performed.[6] In the case of a point out of the grid, or out of gamut, we have performed a clipping.

One might argue that our points are defined in the RGB device-dependent colorspace, thus no colors will be out of gamut. It is of course true that the colors all belong to the gamut by definition, however it is not obvious that they belong to the volume defined by the grid. Indeed, due for instance to the interaction between channels and the forward model's failure to take that into account (see Section 2), some colors could be out of the grid; thus a gamut clipping id needed.

4. RESULTS

In this section, we evaluate the performance of the model defined with regard to the different criteria proposed. We correlate the error of the inverse model with the error of the forward model. We have performed experiments on the 4 devices listed in Table 1. The result for the forward model are listed in Table 2. The better prediction of this model for LCD technology is another justification for this work.

4.1 Criterion used, grid size

The results given by our algorithm are presented in Figures 4, 5, 6, and 7. On these graphs the reader can see the convergence to a maximum accuracy with increasing N_{rgb} (the number of patches). This is not surprising at all as we increase the quantity of information.

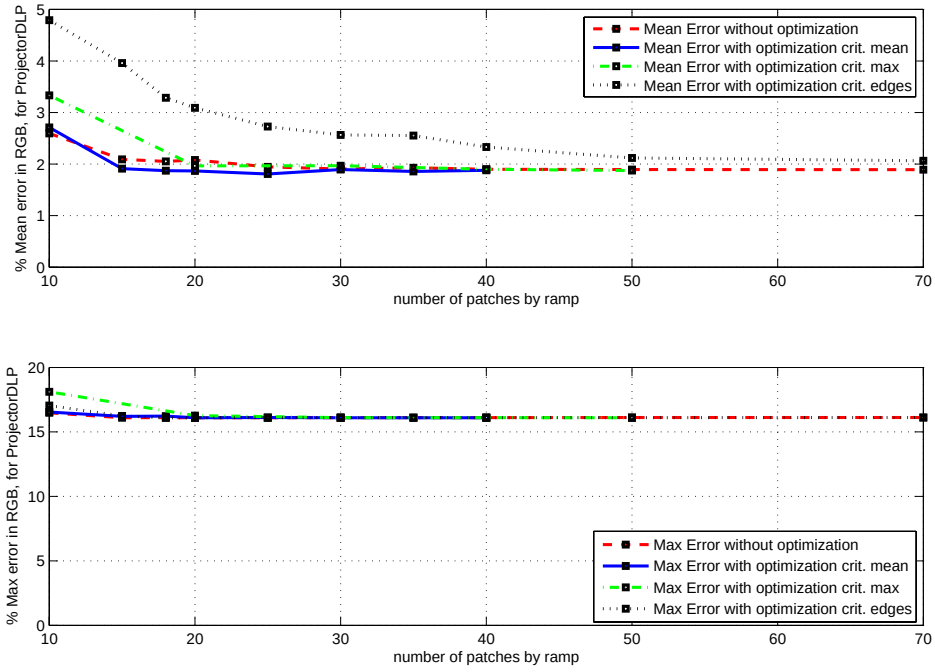

Figure 4. Algorithm performance for ProjectorDLP, depending on the way to optimize the grid and on the number of patches used.

What is more interesting is that each device tested has a different behavior according to the criteria proposed. For ProjectorLCD1, the "edges" criterion provides exactly what we were expected, i.e. reducing the maximum

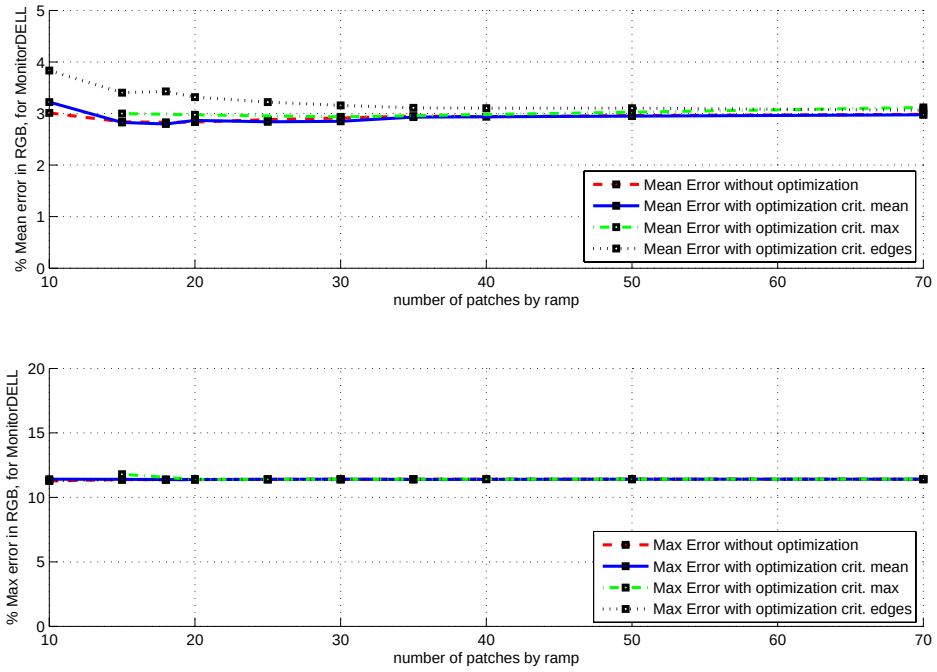

Figure 5. Algorithm performance for MonitorLCD, depending on the way to optimize the grid and on the number of patches used.

error. For this device, a 20^3 sized grid is efficient for an application needing less than 10% maximum error, with a mean error of less than 3% for the data set used. However for ProjectorLCD2, the "edges" criterion gives little increase of accuracy on the data set used. It still converges, but gives poor results compared to a regular grid in the destination space. The "mean" and "max" criterions give interesting results for this device. However, a 18^3 sized grid with the "max" criterion gives approximately the same result as a 35^3 sized grid without optimization. The accuracy improvement is of approximately 1% for the maximum error. The two other devices follow the same behavior. The edge criterion is still not the best criterion for this given data set. The convergence to a maximum error follows approximately the same rules for each method, we believe that it is due to the gamut mapping process on one color of the data set. Note that a zoom on the curves permits to see some differences, but it is around 0.01%, and not significant. The "mean" and the "max" criterions give a small improvement. The similarity of the behavior of these devices is somewhat strange as they are not of the same technology, and their response to the forward model are not the same. Especially the maximum error in prediction is better for MonitorLCD, but for the inverse model it is the inverse. We have no explanation for this.

We explain the unexpected results of the "edges" criterion for some devices with the choice of the constraints. Indeed, it seems that either by the function used (gamma), and the common number of patches along each channel are well designed to reach a good enough homogeneity in the source space. Further work have have to be done in this direction.

We present in Table 3 some results for a fixed number of patches $N_{rgb} = 20$ with the four methods and the PLCC. It appears that the PLCC is less accurate for the two LCD projectors. For the DLP, it is approximately the same. This is explained by the primary constancy in DLP technology. for the monitor, we reduced the mean error, but the maximum error remains the same. For this number of patches and our data set, it appears that the mean error is approximately the same with or without grid optimization, except for the edges criterion which gives an higher error. For the two LCD projectors, we reduced the max error with the "max" and "mean" criteria for both devices and with the "edges" criterion, we achieved good results for ProjectorLCD1.

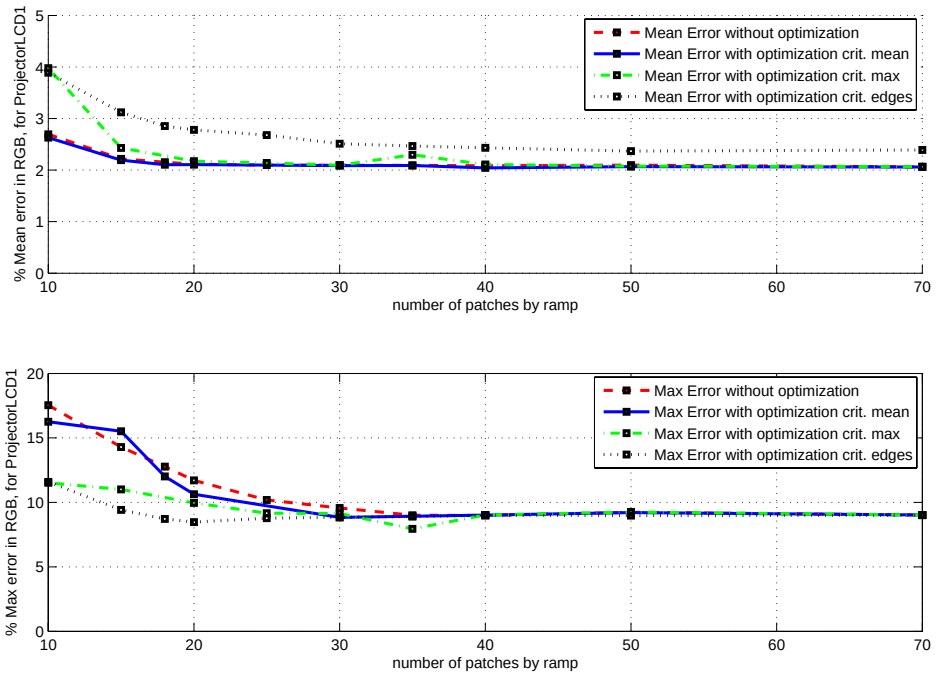

Figure 6. Algorithm performance for ProjectorLCD1, depending on the way to optimize the grid and on the number of patches used. The "edges" criterion reduces the maximum error for the same number of patches.

Table 3. Results for the inverse model. PLCC, PLVC with a 20^3 grid in RGB, a regular one, an optimized one for the edges criterion, an optimized one for the mean criterion, an optimized one for the max criterion. We see that for ProjectorLCD1, we reduced the maximum error with all criteria. In general, the best results are obtained with the "mean" criterion. Note that the maximum error of MonitorLCD remains constant whatever the criterion used. Furthermore, in all cases, the results are better than the black corrected PLCC.

% of error in RGB for a set of 100 random patches		PLCC	PLVC Reg	PLVC Edges	PLVC Mean	PLVC Max
ProjectorLCD1	Mean	0.0509	0.021126	0.027803	0.021052	0.021376
	Max	0.1663	0.117070	0.084756	0.106295	0.091626
ProjectorLCD2	Mean	0.0449	0.007793	0.015687	0.006280	0.008197
	Max	0.0988	0.027602	0.071880	0.019190	0.022855
ProjectorDLP	Mean	0.0183	0.020796	0.030904	0.018660	0.019647
	Max	0.1651	0.160979	0.160948	0.160952	0.162654
MonitorLCD	Mean	0.0572	0.028348	0.033178	0.028670	0.029752
	Max	0.1187	0.113846	0.114133	0.113835	0.113888

The larger errors are localized around the border of the RGB cube. This is not surprising because it is at these places that there is more failure in the additivity properties of the display (see Section 2 and references[1–3]). This is illustrated with Figure 8 which shows spheres centered on the color which sizes are proportional to the error of the inverse model for the ProjectorLCD1, for each plane, RG, RB and GB. The same effect can be seen on the other displays tested.

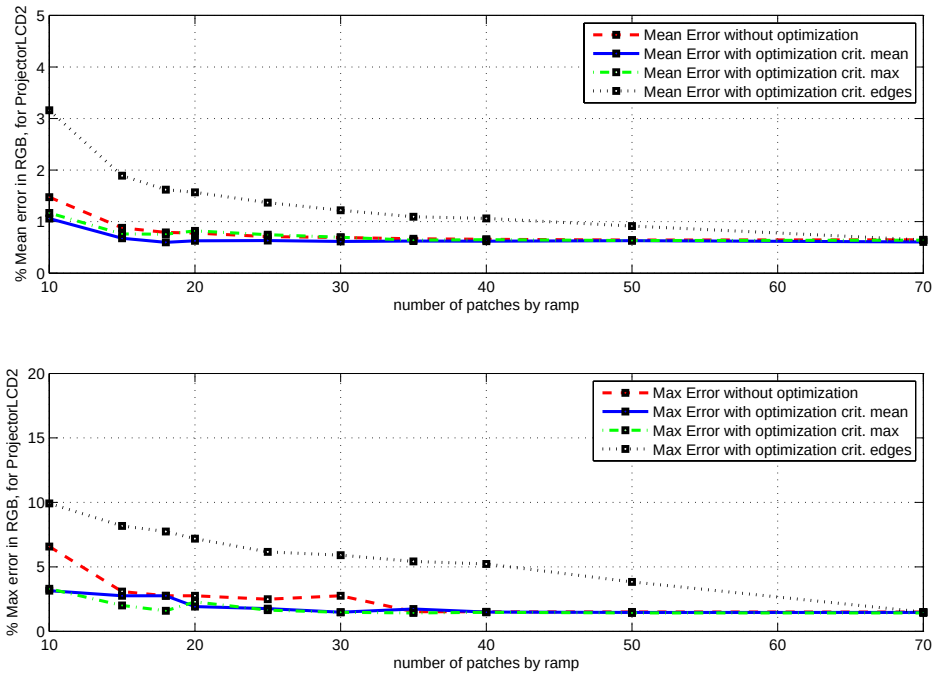

Figure 7. Algorithm performance for ProjectorLCD2, depending on the way to optimize the grid and on the number of patches used. Criterions based on a training data set give good results.

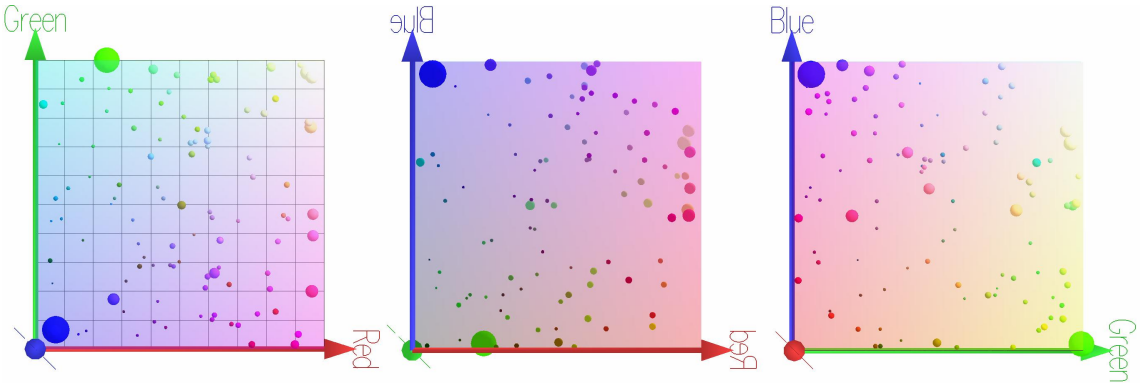

Figure 8. Visualization of the error in RG, RB and GB planes for the inverse model based on the regular grid. Errors are located in highly saturated colors, and in high luminances.

In Table 4 we present the optimized constant parameters of the gamma function for each primary. The parameters obtained are totally different according to the criterion used.

- For “edges”, we obtained parameters below 1 which compensate for the distorsions caused by the transform from RGB to CIELAB. That leads to more vertices in the low intensities.
- However for the criterions linked with the result on a data set, “mean” and “max”, the algorithm tends to refine the grid in the high luminance and in saturated areas with parameters equal to or above 1. It correspond to:

– where the main errors appear, i.e. in the highly saturated area. This is due to the forward model errors, and refining the grid in this area seems to be useless. In order to avoid this case, a clipping of the training patches inside the "forward model gamut" should be done before the optimization process.
– where the data set colors are located, i.e. in the high luminances in CIELAB. This is due to the fact that the data set is equiprobably distributed in RGB.

Table 4. Values of the parameters after convergence of the algorithm.

Values of estimated parameters for devices and criteria		PLVC Reg	PLVC Edges	PLVC Mean	PLVC Max
ProjectorLCD1	α	1.0	0.4489	0.9335	0.8929
	β	1.0	0.7171	0.8338	0.6478
	γ	1.0	0.9634	1.1006	0.9775
ProjectorLCD2	α	1.0	0.5109	3.1956	2.9994
	β	1.0	0.8370	1.4988	1.1870
	γ	1.0	0.9222	1.8965	2.2336
ProjectorDLP	α	1.0	0.4422	1.5887	1.6403
	β	1.0	0.7502	1.7943	1.3891
	γ	1.0	0.7401	1.3347	1.0516
MonitorLCD	α	1.0	0.5306	0.9475	1.1885
	β	1.0	0.6525	0.9725	1.7018
	γ	1.0	0.8654	0.8643	0.7803

4.2 Discussion about the optimization

The convergence of the optimization algorithm deserves some discussion. We have used only one seed (first simplex) for the optimization. It seems like the optimization converged to local minima on a few occasions (the points which are missing in Figures 4, 5, 6 and 7 which have been removed in order not to spoil the intended content of the curves). We faced this problem especially for the max criterion, which is highly non-monotonic. For instance one could face two cases:

- The result for the data set is an high error, over 90%
- It could happen as well that some differents α, β, γ give the same result, while $\alpha = 1.11045, \beta = 1.34657, \gamma = 1.06911$ give a mean error of 0.0207714 and a maximum of 0.092479. While $\alpha = 0.89297, \beta = 0.647803, \gamma = 0.977582$ give a mean error of 0.0209729 and a maximum error of 0.0902327 for ProjectorLCD1. This is quite similar considering that the parameters are strongly different.

Both of these effects appear as well with the mean criterion, but less strong. They have never happened with the "edges" criterion, which seems then to be more robust than the others. This can be considered as normal as this is a function directly linked with the structure.

The possibility of falling inside a local minimum can be limited for instance by taking several simplexes as seeds for the algorithm.

In the case of normal convergence, the algorithm iterated (depending on N_{rgb}) between 55 and 85 times for ProjectorLCD1 for the "max" criterion, and between 32 and 56 times for the "mean" criterion. For the "edges" criterion, the number of iteration is between 65 and 100. In the case of a local minimum, it stopped after a few number of iterations (below 10). The stopping conditions for the algorithm were that the cost function for the simplex's vertices should not show a difference over $\epsilon = 10^{-5}$. The unit of ϵ is in % RGB error for "mean" and "max" and in variance unit for "edges". We have allowed a maximum number of iterations of 100.

5. CONCLUSION AND PERSPECTIVES

We have defined a method to invert any forward color characterization model. Our model is based on the creation of a 3-D grid in the destination color space. In order to build this grid we have used an optimization process to set the constant parameters of the function used to yield the grid. The cost function has been choosen either relatively to the characteristics of the 3-D grid in the source colorspace, or relatively to the result of the model/grid on a training data set. Good results have been shown for four different display devices, using the PLVC forward model.

Many future works can be envisaged based on these first results. For instance, reducing the size of the 3-D grid without any loss of accuracy in average while reducing the maximum error. This could be achieved giving the possibility to the number of patches to be different along each channel, but also using an S-shaped curve function instead of a gamma like function. That would lead to a higher dimension optimization problem, but the Nelder-Mead algorithm works well for this kind of situations, and the computational time is not really a problem as long as once the grid is built it is done, i.e. we do not have to compute it again. The criteria used to estimate the shape of the 3-D grid can be discussed, and we are still looking for better indicators. Note that although we have used the PLVC forward model, our algorithm can be used with any forward model which is hardly analytically invertible, or which requires too many computations to perform the inversion. Some existing forward models seem to match with this condition.

REFERENCES

1. D. L. Post and C. S. Calhoun, "An evaluation of methods for producing desired colors on CRT monitors," *Color Research & Application* **14**, pp. 172–186, 1989.
2. D. L. Post and C. S. Calhoun, "Further evaluation of methods for producing desired colors on CRT monitors," *Color Research & Application* **25**, pp. 90–104, 2000.
3. L. Jimenez Del Barco, J. A. Daz, J. R. Jimenez, and M. Rubino, "Considerations on the calibration of color displays assuming constant channel chromaticity," *Color Research & Application* **20**, pp. 377–387, 1995.
4. P. Yeh and C. Gu, *Optics of Liquid Crystal Display*, Wiley, New-York, 1999.
5. J.-B. Thomas, J. Y. Hardeberg, I. Foucherot, and P. Gouton, "Additivity based LC display color characterization," *Proc. of Gjøvik Color Imaging Symposium* **4**, pp. 50–55, 2007.
6. L. M. Kasson, S. I. Nin, W. Plouffe, and J. L. Hafner, "Performing color space conversions with three-dimensional linear interpolation," *J. of Electronic Imaging* **4(3)**, pp. 226–250, 1995.
7. H. Akima, "A new method of interpolation and smooth curve fitting based on local procedures," *JACM* **17**, pp. 589–602, 1970.
8. P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, 1982.