Sampling CIELAB color space with perceptual metrics

Philippe Colantoni[1], **Jean-Baptiste Thomas**[2] **and Alain Trémeau**[3]

[1]Université Jean Monnet,
Centre Interdisciplinaire d'Etudes et de Recherches sur l'Expression Contemporaine,
Saint-Étienne, France.
philippe.colantoni@univ-st-etienne.fr

[2]Laboratoire Electronique, Informatique et Image,
Université de Bourgogne,
Dijon, France.
jean-baptiste.thomas@u-bourgogne.fr

[3]Université Jean Monnet,
Laboratoire Hubert Curien,
Saint-Étienne, France.
alain.tremeau@univ-st-etienne.fr

ABSTRACT

Sampling a perceptually uniform or pseudo-uniform color space is required for applications from image processing to computational imaging. However, one can face two problems while trying to perform a uniform sampling of such space. First, the usual cubic grid is not perceptually uniform in most cases. Second, perceptual metrics are often not Euclidean. We propose to overcome these problems. We apply our solution on $CIELAB$ *color space to test its efficiency. We propose an algorithm to define a tabulated color space with regard to a non-Euclidean color difference formula, i.e.* ΔE^*_{00} *in* $CIELAB$. *The tabulated data are available at* `http://data.couleur.org/deltaE/`. *Later, we propose to combine this tabulated color space with an approximated 3D close packed hexagonal regular sampling of* $CIELAB$. *Evaluations of the transform and of the regular sampling are performed and compared with literature standards.*

Keywords: Colorimetry; Perceptually uniform color space; Color differences; Sampling; 3D close packed hexagonal grid.

2000 Mathematics Subject Classification: 06-Order, lattices, ordered algebraic structures; 68-Computer Science, P and W.

Computing Classification System: Algorithm; Theory

1 Introduction

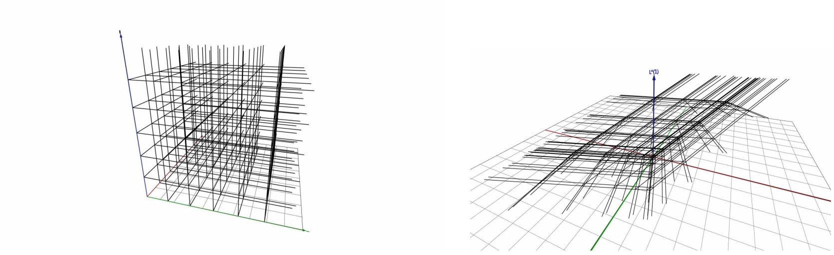
$CIELAB$ color space (CIE, 2004) has been accepted by the CIE (International Commission on Illumination) as a perceptual pseudo-uniform color space such that the Euclidean distance between two specified colors in this space is proportional to the color difference between these colors perceived by a standard observer. Although this color space has been defined only for very well defined and limited colorimetric conditions, it has been, improperly but successfully, used in practice in many applications in color image processing or computational color science. However, the Euclidean metric ΔE^*_{ab} has been shown inappropriate, and alternative solutions have been proposed in the last 20 years, such as ΔE^*_{94}, ΔE^*_{CMC} and ΔE^*_{00} (CIE, 2004; Luo, Cui and Rigg, 2001). These Color Difference Formulas (CDF) are known to be more accurate considering human perception (Kim, Cho and Kim, 2001; Melgosa, Huertas and Berns, 2004; Luo, Minchew, Kenyon and Cui, 2004), but have the major disadvantage to be not Euclidean. This makes them difficult to use in some practical tasks, such as sampling.

Indeed, sampling a color space is a major issue in many applications in terms of hardware complexity and speed, accounting for perception, and resulting image quality (Gentile, Allebach and Walowit, 1990). Historically, a parallelepipedic grid was used for sampling $CIELAB$ space (Hill, Roger and Vorhagen, 1997). Such a grid is defined by a regular lattice that is reproduced over and over in order to fill the space. In some cases, the sampling is performed in the RGB or CMY spaces and then transformed into $CIELAB$, which leads to a large non-uniformity of the final sampling due to the response compression and the chromatic adaptation included in the transform (Mahy, Van Mellaert, Van Eycken and Oosterlinks, 1991; Trémeau, Konik and Lozano, 1996), as shown in Figure 1. Even if the space is directly sampled using a parallelepipedic grid, the sampling is not uniform. For instance, in the case of a cubic sampling of step a, the distance between a sample and its closest neighbors can be either a, $a\sqrt{2}$ or $a\sqrt{3}$, depending on the direction.

Solutions to uniform sampling might be found in 3D close packed hexagonal sampling. Such a sampling has been used already in the field of computational color science and color imaging. It has been used for color specification, such as Munsell re-annotation (Wyszecki, 1954) and OSA color system arrangement (Foss, 1978; MacAdam, 1978). It has also been used specifically for sampling $CIELAB$ color space for color image quantization and analysis (Thomas and Trémeau, 2007; Thomas, Chareyron and Trémeau, 2007; Colantoni, Thomas and Pillay, 2010), for display color characterization (Stauder, Colatoni and Blonde, 2006; Stauder, Thollot, Colantoni and Tremeau, 2007; Colantoni and Thomas, 2009), and for color space investigation (Thomas, Colantoni and Tremeau, 2013).

When it comes to non-Euclidean CDFs based sampling, a great deal of trouble is generated. Philipp Urban *et al.* (Urban, Rosen, Berns and Schleicher, 2007) and Lorenzo Ridolfi *et al.* (Ridolfi, Gattass and Lopes, 2010) proposed two methods to generate a tabulated version of $CIELAB$, which can be used to perform Euclidean operations in this space with respect to perceptual non-Euclidean CDFs. We initiated some preliminary works in this direction also in our image visualization and analysis of art paintings (Colantoni et al., 2010). This article presents a robust solution and complete analysis of such sampling strategy.

Our paper is organized as follows. First, we propose an algorithmic method to sample the $CIELAB$ color space with non Euclidean CDFs. A tabulated space is generated, which enable us to use the Euclidean metric according to the chosen CDF. Then, we perform an approximated uniform sampling of the $CIELAB$ color space, based on 3D close packed hexagonal.



(a) Regular sampling of the $sRGB$ color space.

(b) Transform of a regular sampling from $sRGB$ to $CIELAB$.

Figure 1: **Visualization of a regular sampling of the sRGB color space and its conversion into the $CIELAB$ space. We can notice a large lack of uniformity in such a sampling.**

To evaluate the method, we analyze the statistical distribution of the tabulated space and quantify the accuracy of the transformation. We compare our work with Urban *et al.*, which is the reference work in this domain. Lastly, we analyze and discuss the properties of the 3D close packed hexagonal grid we generate.

2 Tabulated Euclidean space embedding a non-Euclidean metric

This section considers the creation of a tabulated space, uniform with regard to non-Euclidean CDFs. This space is defined such that an Euclidean metric can be used piecewise to approximate any non-Euclidean CDF. Perceptual CDFs such as the ΔE_{94}, ΔE_{CMC}, ΔE_{00} and any color appearance spaces are not Euclidean, referring to the $CIELAB$ color space. Indeed, while using these metrics in relation with the $CIELAB$, it is far more difficult to achieve a uniform sampling. The method that we propose is based on 3D grid piecewise morphing. Next we compare our results with the method proposed by Urban *et al.* (Urban et al., 2007) based also on a tabulated space.

2.1 Concept

Due to the experimental nature of the $CIELAB$ color space and to the visual system features, the Euclidean metric does not permit a very good quantification of color differences in this space. To overcome this, the CIE and other standardization organisms proposed more complex (in term of computation) CDFs: ΔE_{94}, ΔE_{CMC}, ΔE_{00}. These CDFs increased the quality of color sample difference estimation without modifying the space itself. They are typically based on a lightness weighting function, a chroma weighting function, a hue weighting function, an intermediate term between chroma and hue differences and a scaling factor for a^* scale, which mainly affects colors with low chroma. ΔE_{00} includes a hue rotational term to deal with problematic blue regions.

We propose here to define a piecewise function that enables us to create a tabulated structure based on a non-Euclidean CDF of the $CIELAB$ color space. Thanks to this transform, it is possible to use a piecewise Euclidean metric to approximate a non-Euclidean CDF.

In the literature, only Urban *et al.* and Ridolfi *et al.* proposed a similar approach. One algorithm

(Urban et al., 2007) is based on a local optimization of a square grid via the use of its dual in the a^*b^* plane. The nodes are initially placed at regular distance ΔE^*_{ab}. Next, these grids are optimized by modifying the nodes alternatively toward a pseudo-uniformity considering a given non-Euclidean metric. They set up a set of conditions in order to avoid the grid to collapse, which is more likely to happen with ΔE_{00} considering ambiguity on hue values and discontinuities, such as studied by Sharma *et al.* (Sharma, Wu and Dalal, 2005). This approach is very robust and fast to converge considering the constraints on the global shape of the grid. However, this robustness forbid it to be very accurate locally. The other algorithm (Ridolfi et al., 2010) uses multidimensional scaling techniques to achieve such tabulated space with ΔE_{00}. Their method gives good freedom to the distribution of the data, and includes the problematic Gaussian curvature of the a^*b^* plane. While these approaches consider a global grid of a given number of data adjusted to the non-Euclidean metric, our approach considers a local method, focusing on the volumetric aspect information. In a nutshell, we place every point as good as it can be, one after another, rather than refining a grid. Our main objective is to limit the error in most part of the space at the expense of the error in some very rare area.

2.2 Design of the method

We denote ΔE^{smp} the sampling distance used to generate the tabulated sampling based on a generic distance, $\Delta E_{xx} \in \{\Delta E_{76}, \Delta E_{94}, \Delta E_{CMC}, \Delta E_{00}\}$. We define also the distance ΔE_{xx} between 2 colors C_1 and C_2 as being the average of $\Delta E_{xx}(C_1, C_2)$ and $\Delta E_{xx}(C_2, C_1)$. By doing this, we take into account the lack of symmetry of some perceptual CDFs such as ΔE_{CMC}.
The natural separation of lightness and chroma attributes in the space and in the CDF's definitions allows us to split the tabulated space according to one 2D-LUT corresponding to (a^*, b^*) plane and a 1D-LUT corresponding to the L^* axis. The sampling transform linked to the tabulated space that we propose is defined by the combination of a 1D linear interpolation with a 2D bilinear interpolation.
The two-dimensional grid corresponding to the 2D-LUT of the (a^*, b^*) chromatic plane is the result of a diffusion process that starts from a central point of coordinate $a^* = 0$ and $b^* = 0$. This process is based on pre-computed values over the axis a^* and b^*. These axes serve as anchor to preserve the grid from collapsing.

Stage 1: discretization of the axis according to a given sampling distance ΔE^{smp}.
Let us note that the functions $\delta E_{xx}(a^*)$, $\delta E_{xx}(b^*)$ and $\delta E_{xx}(L^*)$ (distances computed from the L^*, a^* or b^* axis) are monotonous and are independent of the CDFs used. The estimation of the next sample on the plane at a distance ΔE^{smp} is performed using a simple dichotomous search. That enables us to obtain tabulated samples $A_+[I]$ and $A_-[I]$ along the positive and negative directions of the a^* axis; tabulated samples $B_+[I]$ and $B_-[I]$ along the positive and negative directions of the b^* axis, and tabulated samples $L[I]$ along the lightness axis L^* for a given ΔE^{smp}. The synopsis of the algorithm is given in Appendix, see Algorithm 2.

Stage 2: creation of the two-dimensional grid on (a^*, b^*) plane with a diffusion process.
The support for this 2D grid is a matrix of (a^*, b^*) coordinates, its size (N, M) must be sufficient to cover $CIELAB$ destination space, so that it includes the whole gamut considered.

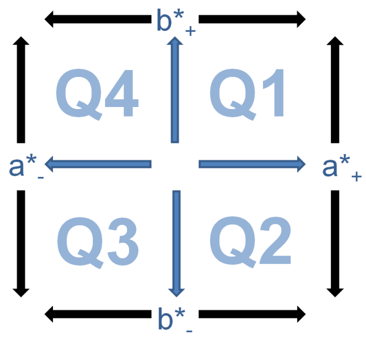


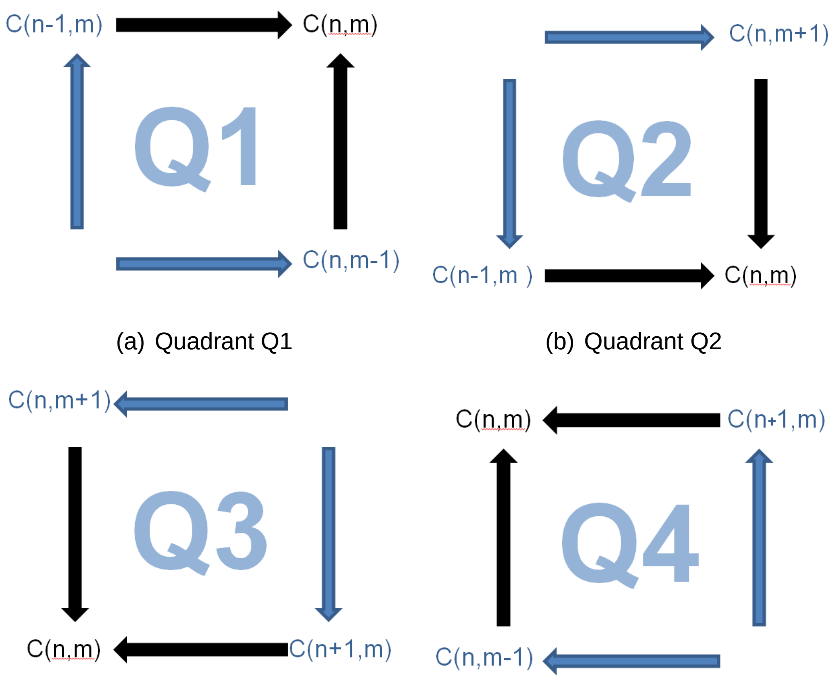
Figure 2: **Quadrants definition in $CIELAB$.**

For each data of the 2D grid, we compute its distance to the center of coordinates $(0, 0)$. This value is used to sort the points by ascending order of distances, in order to determine the order of processing. This enables our diffusion process. The initial transform of the grid is to include the values computed on the axis a^* and b^* in stage 1. The rest of the data is processed successively, according to the stack order, considering the following constraints depending on which quadrant it belongs to (Figure 2).

- $Q1$ $(a^*_+ b^*_+)$: the point $C(n, m)$ has to be at a given distance ΔE^{smp} of $C(n-1, m)$ and $C(n, m-1)$ (Figure 3(a)).
- $Q2$ $(a^*_+ b^*_-)$: the point $C(n, m)$ has to be at a given distance ΔE^{smp} of $C(n-1, m)$ and $C(n, m+1)$ (Figure 3(b)).
- $Q3$ $(a^*_- b^*_-)$: the point $C(n, m)$ has to be at a given distance ΔE^{smp} of $C(n+1, m)$ and $C(n, m+1)$ (Figure 3(c)).
- $Q4$ $(a^*_- b^*_+)$: the point $C(n, m)$ has to be at a given distance ΔE^{smp} of $C(n+1, m)$ and $C(n, m-1)$ (Figure 3(d)).

The position of each new point is estimated from the previous data and from the axis rigid data as follows, according to its location in the tabulated data:

- in Q1 (Figure 3(a)),
  $C(n, m)_a = C(n-1, m-1)_a + (A_+[n] - A_+[n-1])$
  $C(n, m)_b = C(n-1, m-1)_b + (B_+[m] - B_+[m-1])$
- in Q2 (Figure 3(b)),
  $C(n, m)_a = C(n-1, m+1)_a + (A_+[n] - A_+[n-1])$
  $C(n, m)_b = C(n-1, m+1)_b + (B_-[m] - B_-[m+1])$
- in Q3 (Figure 3(c)),
  $C(n, m)_a = C(n+1, m+1)_a + (A_-[n] - A_-[n+1])$
  $C(n, m)_b = C(n+1, m+1)_b + (B_-[m] - B_-[m+1])$
- in Q4 (Figure 3(d)),
  $C(n, m)_a = C(n+1, m-1)_a + (A_-[n] - A_-[n+1])$
  $C(n, m)_b = C(n+1, m-1)_b + (B_+[m] - B_+[m-1])$.



(a) Quadrant Q1 (b) Quadrant Q2

(c) Quadrant Q3 (d) Quadrant Q4

Figure 3: **Data coordinates by quadrants.**

Once this position is evaluated, we optimize the (a^*, b^*) coordinates of $C(n, m)$. For the $Q1$ quadrant, we optimize the distances between $C(n, m)$ and $C(n-1, m)$ and between $C(n, m)$ and $C(n, m-1)$ in order to get as close as possible to ΔE^{smp}. Due to the nature of the metrics and due to the Gaussian curvature of the plane, the result of this optimization might be an approximation. This optimization moves repeatedly the point $C(n, m)$ as shown in Algorithm 1.

Algorithm 1 Optimization of the position of $C(n, m)$.

$iter \leftarrow 0$
repeat
  $l1 \leftarrow \Delta E_{xx}(C(n, m), C(n, m-1))$
  $l2 \leftarrow \Delta E_{xx}(C(n, m), C(n-1, m))$
  $C(n, m)_a \leftarrow C(n, m)_a - (C(n, m)_a - C(n, m-1)_a) \times \frac{(l1-\Delta E^{smp})}{1000} - (C(n, m)_a - C(n-1, m)_a) \times \frac{(l2-\Delta E^{smp})}{1000}$
  $C(n, m)_b \leftarrow C(n, m)_b - (C(n, m)_b - C(n, m-1)_b) \times \frac{(l1-\Delta E^{smp})}{1000} - (C(n, m)_b - C(n-1, m)_b) \times \frac{(l2-\Delta E^{smp})}{1000}$
  $iter \leftarrow iter + 1$
until $(iter < MAXITER)$ or $((l1 \neq \Delta E^{smp})$ and $(l2 \neq \Delta E^{smp}))$

The motion of the iterative move is proportional to the remaining distance from an optimal $C(n, m)$. We stop this iterative process when $C(n, m)$ is at a perfect position, *i.e.* $\Delta E_{xx}(C(n, m), C(n, m-1)) = \Delta E_{xx}(C(n, m), C(n-1, m)) = \Delta E_{smp}$, or when we reach a maximum iteration (we used $MAXITER = 20000$).

We proceed in a similar way when $C(n, m)$ is in $Q2$, $Q3$ and $Q4$. When the stack is empty, the

Table 1: **Comparison of the data provided by Urban with our data; ΔE_{94}, ΔE_{CMC}, ΔE_{00} CDFs are used to generate tabulated spaces based on a ΔE^{smp} of 1. Results show that all the created tabulated spaces are rather well uniform.**

$\Delta E_{xx} = 1$ 2D/3D Urb/Us			Number of Nodes	Average segment	Average error	Std Dev	95 perc.	Maximum segment	Minimum segment
ΔE_{94}	2D	Urban	14,019	0.877	0.123	0.055	0.208	0.993	0.764
	2D	Our	11,226	**1.001**	0.001	0.000	0.001	**1.001**	**0.998**
	3D	Urban	1,299,755	0.919	0.081	0.074	0.200	1.000	0.764
	3D	Our	1,025,327	**1.000**	0.000	0.000	0.001	**1.001**	**0.998**
ΔE_{CMC}	2D	Urban	14,029	1.122	0.122	0.157	0.401	2.279	0.235
	2D	Our	19,418	**1.001**	0.001	0.001	0.002	**1.018**	**0.997**
	3D	Urban	1,224,412	1.094	0.094	0.145	0.380	2.279	0.235
	3D	Our	1,712,716	**1.000**	0.000	0.001	0.001	**1.018**	**0.997**
ΔE_{00}	2D	Urban	13,274	0.931	0.069	0.125	0.281	1.906	0.510
	2D	Our	13,126	**1.003**	0.003	0.010	0.020	**1.069**	**0.971**
	3D	Urban	965,939	0.960	0.040	0.101	0.221	1.906	0.510
	3D	Our	935,341	**1.001**	0.001	0.007	0.002	**1.069**	**0.971**

chromatic plane of the new space is defined from these tabulated data. The synopsis of the algorithm is given in Appendix in algorithms 4 and 3. The lightness tabulation is straightforward.

2.3 Analysis

To evaluate the accuracy of our sampling scheme and to compare it with the sampling scheme proposed by Urban *et al.* we analyze first-order statistics computed over the grid, for different ΔE^{smp} and different ΔE_{xx}. The comparison is done from data provided on their website (Urban et al., 2007). Note that the difference of number of samples computed with our tabulated space is due to the method used, but also and mainly to the fact that their data does not cover the entire spectrum locus, as can be seen on Figure 4 and 5. This is the case because Urban *et al.* used the boundaries of the encoding $CIELAB$ (this is reasonable since color difference metrics are designed to close to achromatic colors; however, in practice it might be useful to cover the entire locus). Instead, we limit our data set to the spectrum locus computed for a D65 illumination. The spectrum locus boundaries are approximated based on the combination of uni-modal spectral primaries shaped as rectangular functions. In the following, the terms 2D and 3D correspond, respectively, to the chromatic (a^*, b^*) plane and to the entire 3D space, including the lightness.

Table 1 shows that, with a ΔE^{smp} of 1, our method generates a very accurate tabulated space with a mean distance value of 1 and a rather small standard deviation for all tested CDFs. The highest standard deviation is equal to 0.010 (resp. 0.157) with our 2D method (resp. Urban's method) with the ΔE_{00} CDF (resp. the ΔE_{CMC} CDF), whereas the highest standard deviation is equal to 0.007 (resp. 0.145) with our 3D method (resp. Urban's method) with the ΔE_{00} CDF (resp. the ΔE_{CMC} CDF). In the worst case, the maximal error is of (1.069-1) with our method (with the ΔE_{CMC} CDF), meanwhile it is of (2.279-1) with the Urban's method (with also the ΔE_{CMC} CDF). These results are simply based on the grid's data, with no transformation needed.

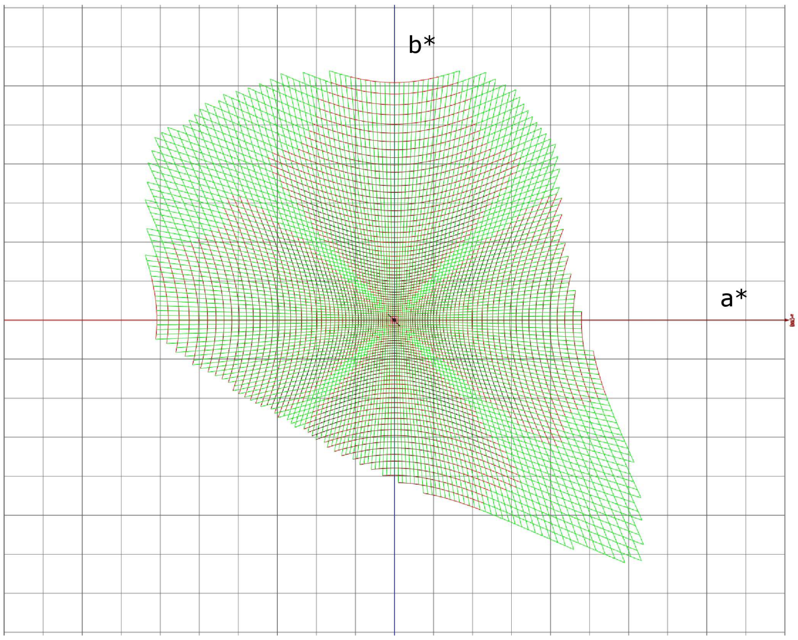


Figure 4: Visualization of Urban's data set in the (a^*, b^*) plane of the $CIELAB$ color space. We set $\Delta E_{94} = 1$ to generate the tabulated space. The data are constrained to the projected spectrum locus boundaries. At each node (sample) corresponds eight segments. Each segment relies two adjacent tabulated samples.

In most cases, we improve on the accuracy of the tabulated space defined by Urban's data. Table 1 gives only first indications, since data do not overlap the same area in the $CIELAB$ space. Moreover, we can note that for ΔE_{CMC}, Urban's method does not provide very good results, this might be due to the lack of symmetry of this CDF. Additional local results are provided in Figures 4 and 5, next in Figures 6 and 7. The color of the segments created is set to black, green or red according to the corresponding error. **Black is set for relative errors of 0 to 5%, green for 5 to 10% and red when it is over 10%**. Figures 5 and 7 show that our grid is very accurate in this case, oppositely to the Urban's method, as we can see in Figures 4 and 6. Let us also notice that Figures 6 and 7 are very interesting as they emphasize the fundamental difference between these two methods. We can clearly see in Figures 7 and 6 that errors with our grid are concentrated in a very small area, whereas the other method provides a more homogeneous error. This is to be compared with the results obtained by Ridolfi *et al.*, because their method seems to perform very well especially in this area. The lowest error is obtained with our 3D method with the ΔE_{94} CDF, next with the ΔE_{CMC} CDF (errors are very closed to those obtained with the ΔE_{94} CDF) and lastly with the ΔE_{00} CDF. We observe the same tendency with Urban's data.

From now, we study the influence of the discretization step on the accuracy of the method. We can notice in Table 2 that the increase of the method's accuracy is inversely proportional to the decrease of the discretization step. Once again, in Table 2, we can notice that there are more errors with the ΔE_{00} CDF than with the ΔE_{94}. The difference of accuracy between the ΔE_{00} CDF and the ΔE_{94} tends to decrease when the discretization step tends to decrease also, as these CDFs tend to be more and more linear with the increasing of color difference.

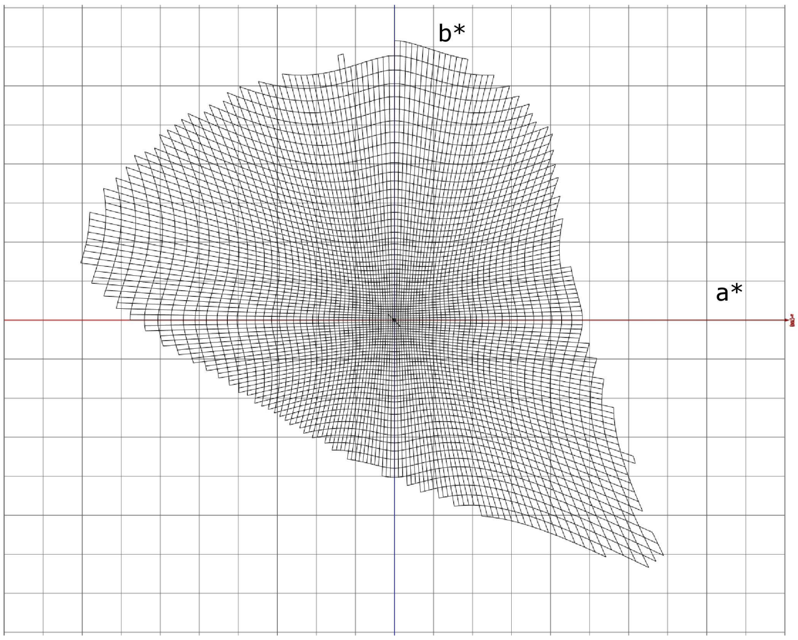


Figure 5: Visualization of our data set in the (a^*, b^*) plane of the $CIELAB$ color space. We set $\Delta E_{94} = 1$ to generate the tabulated space. The data are constrained to the projected spectrum locus boundaries. At each node (sample) corresponds eight segments. Each segment relies two adjacent tabulated samples.

In order to evaluate the accuracy of the three tabulated spaces, three different experiments have been performed. These evaluations are all based on 10 millions of pairs of samples randomly selected, with pairs of samples spaced at a given distance. The selection is performed as follow: First, 10 millions of color patches are selected randomly within the spectrum locus. Second, for each color patch, a second color patch is selected, based on two random angles in spherical coordinates, with a radius of *distance patches* ΔE^P. Third, these 2 points are transformed through our tabulated space, and the resulting distance between them in the new space is the average of the distance computed from one to the other and *vice versa*.

- Evaluation A: Couples of data are generated in $CIELAB\Delta E_{xx}$, sampled with ΔE^{smp} of 0.25, 0.5 and 1. Both points are transformed to $CIELAB$, then the ΔE_{xx} between them is evaluated by the average of the distance in both directions, ΔE^M. The error is expressed as $E = \Delta E^P - \Delta E^M$.

- Evaluation B: Data are generated in $CIELAB$, with $\Delta E^P = 1$. These data are transformed to $CIELAB\Delta E_{xx}$, then the Euclidean distance between them is computed (this is equivalent to approximate ΔE_{xx}) and compared with the ΔE_{xx} computed directly in $CIELAB$.

- Evaluation C: Data are generated in $CIELAB$, then transformed to $CIELAB\Delta E_{xx}$, and transformed back to $CIELAB$. Evaluation is performed on how they come back alike with ΔE^*_{ab}.

Tables 3 to 8 show the results of these evaluations for three different CDFs and different tabulated spaces. Depending on the accuracy wanted, different parameters would be preferred.

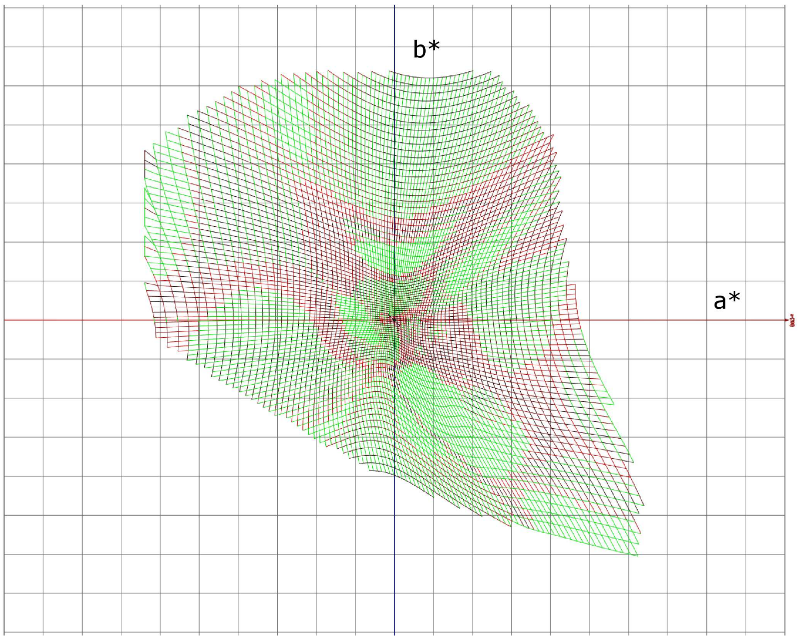


Figure 6: Visualization of Urban's data set in the (a^*, b^*) plane of the $CIELAB$ color space. We set $\Delta E_{00} = 1$ to generate the tabulated space. The data are constrained to the projected spectrum locus boundaries. At each node (sample) corresponds eight segments. Each segment relies two adjacent tabulated samples.

Nonetheless, a ΔE^{smp} of 1 provides correct results. This suggest that a finer sampling is not necessarily the best, since it includes more relative errors in the distribution of data. However, this is to be balanced with the accuracy of the transform as we will show in the following.

3 Uniform $CIELAB$ sampling with non Euclidean CDFs

In this section we use the tabulated uniform version of $CIELAB$ color space based on a given CDF and a 3D close packed hexagonal sampling scheme. This permits studying the accuracy of the transform itself.

3.1 Results

Table 9 provides statistics on the uniform sampling obtained with a distance of 1 and a tabulated space based on different ΔE^{smp}. Results show that the use of a small ΔE^{smp} reduces the average error but increases the maximum error and *vice versa*. This is directly due to the tabulation accuracy, as shown in Table 2. Figure 8 illustrates the sampling for a distance of 4. Although this distance does not mean much in term of colorimetry, it provides a better visibility of the error location. We can observe more errors in the bluish area.
Table 10 provides statistics for the uniform sampling of a cylinder of radius 50 centered and aligned with the lightness axis. This makes us able to compare the efficiency of both methods fairly since we cannot cover the spectrum locus with Urban's data. In having a radius of 50, we may be a bit favored as our grid is more accurate in close to achromatic area, but as a

Table 2: **Comparison for different value of ΔE^{smp}: 0.25, 0.5 and 1.0 The comparison is performed for 2D and 3D data with the ΔE_{94}, ΔE_{CMC}, ΔE_{00} CDFs.**

ΔE_{xx} 2D/3D & 0.25/0.5/1			Number of Nodes	Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
ΔE_{94}	0.25	2D	179,892	0.2519	0.2549	0.2440	0.7646	0.4662	1.4461
		3D	65,818,399	**0.2512**	0.2549	0.2440	**0.4848**	0.4820	1.3143
	0.5	2D	45,118	0.5010	0.5023	0.4968	0.1946	0.1179	0.3656
		3D	8,258,326	**0.5006**	0.5023	0.4968	**0.1236**	0.1223	0.3327
	1	2D	11,226	1.0005	1.0013	0.9984	0.0503	0.0290	0.0923
		3D	1,025,327	**1.0003**	1.0013	0.9984	**0.0317**	0.0309	0.0846
ΔE_{CMC}	0.25	2D	305,935	0.2528	0.2756	0.2464	1.1201	1.1940	3.1337
		3D	108,079,761	**0.2517**	0.2756	0.2464	0.6700	1.0976	2.5127
	0.5	2D	77,670	0.5014	0.5232	0.4946	0.2705	0.3673	0.8133
		3D	13,713,475	**0.5008**	0.5232	0.4946	0.1638	0.3433	0.6287
	1	2D	19,418	1.0006	1.0182	0.9973	0.0563	0.0927	0.1511
		3D	1,712,716	**1.0003**	1.0182	0.9973	0.0331	0.0866	0.1266
ΔE_{00}	0.25	2D	207,211	0.2533	0.3018	0.2329	1.3320	2.7743	6.2991
		3D	59,143,450	**0.2518**	0.3018	0.2329	**0.7030**	1.9108	2.5965
	0.5	2D	52,564	0.5029	0.5584	0.4779	0.5780	1.7105	3.4226
		3D	7,496,169	**0.5015**	0.5584	0.4779	**0.2952**	1.1635	0.7664
	1	2D	13,126	1.0028	1.0691	0.9707	0.2819	1.0307	1.9457
		3D	935,341	**1.0014**	1.0691	0.9707	**0.1384**	0.6947	0.1829

Table 3: **Accuracy of our method with the ΔE_{94} CDF for different values of ΔE^{smp} (Evaluation A). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{94}			Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
	ΔE^{smp}	Distance Patches						
Eval A	0.25	1	1.003	1.220	0.685	0.325	5.999	12.255
		2	2.007	2.327	1.657	0.353	5.960	12.206
		4	4.019	4.651	3.326	0.470	5.909	12.161
	0.5	1	1.000	1.153	0.823	0.017	5.789	11.911
		2	2.000	2.305	1.651	0.015	5.782	11.903
		4	4.005	4.611	3.314	0.134	5.747	11.848
	1	1	0.999	1.154	0.820	0.114	5.770	11.894
		2	1.998	2.306	1.642	0.081	5.763	11.882
		4	4.002	4.606	3.309	0.039	5.728	11.820

Table 4: **Accuracy of our method with the ΔE_{94} CDF for different values of ΔE^{smp} (Evaluations B and C). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{94}			Average error (absolute)	Maximum error (absolute)	Minimum error (absolute)	std Dev (absolute)	95 perc. (absolute)
	ΔE^{smp}	Distance Patches					
Eval B	0.25	1	0.016	0.370	0	0.017	0.049
		2	0.031	0.446	0	0.032	0.098
		4	0.062	0.497	0	0.062	0.193
	0.5	1	0.015	0.529	0	0.019	0.048
		2	0.029	0.577	0	0.032	0.095
		4	0.058	0.557	0	0.061	0.188
	1	1	0.017	1.128	0	0.039	0.049
		2	0.031	1.108	0	0.046	0.098
		4	0.060	1.274	0	0.068	0.193
Eval C	0.25	1	0.002	1.886	0	0.026	0.002
		2	0.002	1.930	0	0.026	0.002
		4	0.002	1.932	0	0.026	0.002
	0.5	1	0.006	3.815	0	0.083	0.007
		2	0.006	3.815	0	0.083	0.007
		4	0.006	3.851	0	0.084	0.007
	1	1	0.023	7.321	0	0.232	0.015
		2	0.023	7.321	0	0.231	0.015
		4	0.023	7.318	0	0.231	0.015

Table 5: **Accuracy of our method with the ΔE_{CMC} CDF for different values of ΔE^{smp} (Evaluation A). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{CMC}			Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
	ΔE^{smp}	Distance Patches						
Eval A	0.25	1	1.003	1.528	0.320	0.296	8.572	18.322
		2	2.006	3.061	0.634	0.275	8.516	18.183
		4	4.012	6.065	1.286	0.295	8.463	18.074
	0.5	1	0.998	1.463	0.257	0.215	8.767	18.738
		2	1.997	2.935	0.536	0.219	8.771	18.779
		4	3.992	5.848	1.086	0.195	8.779	18.864
	1	1	0.996	1.436	0.227	0.386	8.952	19.044
		2	1.992	2.874	0.446	0.389	8.959	19.077
		4	3.985	5.767	0.928	0.367	8.975	19.163

Table 6: **Accuracy of our method with the ΔE_{CMC} CDF for different values of ΔE^{smp} (Evaluations B and C). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{CMC}			Average error (absolute)	Maximum error (absolute)	Minimum error (absolute)	std Dev (absolute)	95 perc. (absolute)
	ΔE^{smp}	Distance Patches					
Eval B	0.25	1	0.031	1.947	0	0.068	0.103
		2	0.062	3.757	0	0.132	0.204
		4	0.124	6.175	0	0.250	0.403
	0.5	1	0.035	2.585	0	0.084	0.109
		2	0.069	4.768	0	0.165	0.214
		4	0.136	7.365	0	0.314	0.427
	1	1	0.039	3.188	0	0.102	0.118
		2	0.075	5.521	0	0.191	0.230
		4	0.145	8.106	0	0.356	0.448
Eval C	0.25	1	0.001	0.938	0	0.015	0.001
		2	0.001	0.938	0	0.015	0.001
		4	0.001	0.938	0	0.015	0.001
	0.5	1	0.004	1.891	0	0.053	0.004
		2	0.004	1.891	0	0.053	0.004
		4	0.004	1.891	0	0.053	0.004
	1	1	0.016	3.773	0	0.150	0.012
		2	0.016	3.762	0	0.150	0.012
		4	0.016	3.773	0	0.150	0.012

Table 7: **Accuracy of our method with the ΔE_{00} CDF for different values of ΔE^{smp} (Evaluation A). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{00}			Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
	ΔE^{smp}	Distance Patches						
Eval A	0.25	1	1.002	1.673	0.054	0.211	9.811	21.449
		2	2.003	3.338	0.109	0.138	9.726	21.166
		4	3.999	6.650	0.225	0.034	9.580	20.786
	0.5	1	0.998	1.577	0.035	0.198	9.998	21.962
		2	1.995	3.138	0.070	0.266	9.951	21.815
		4	3.983	6.265	0.142	0.432	9.849	21.515
	1	1	0.996	1.515	0.027	0.362	10.123	22.310
		2	1.992	3.020	0.054	0.425	10.081	22.176
		4	3.977	6.029	0.111	0.587	9.996	21.912

Table 8: **Accuracy of our method with the ΔE_{00} CDF for different values of ΔE^{smp} (Evaluations B and C). ΔE^{smp} values tested are equal to 0.25, 0.5 and 1.0. The distance values (1, 2 and 4) between pairs of color patches had been chosen in order to have reasonable color differences. The evaluation was performed from 10,000,000 pairs randomly selected.**

ΔE_{00}			Average	Maximum	Minimum	std Dev	95perc.
	ΔE^{smp}	Distance Patches	error (absolute)	error (absolute)	error (absolute)	(absolute)	(absolute)
Eval B	0.25	1	0.026	3.960	0	0.083	0.089
		2	0.051	7.585	0	0.163	0.174
		4	0.101	14.320	0	0.312	0.342
	0.5	1	0.029	7.153	0	0.122	0.094
		2	0.057	13.435	0	0.240	0.185
		4	0.113	23.921	0	0.457	0.365
	1	1	0.034	10.453	0	0.153	0.100
		2	0.064	19.207	0	0.293	0.197
		4	0.122	28.701	0	0.551	0.379
Eval C	0.25	1	0.002	1.754	0	0.024	0.004
		2	0.002	1.733	0	0.024	0.004
		4	0.002	1.733	0	0.024	0.004
	0.5	1	0.008	3.593	0	0.077	0.012
		2	0.008	3.593	0	0.077	0.012
		4	0.008	3.604	0	0.077	0.012
	1	1	0.025	7.074	0	0.210	0.026
		2	0.025	7.069	0	0.210	0.026
		4	0.025	7.074	0	0.210	0.026

Table 9: **Evaluation of the uniform sampling of the spectrum locus based on a $\Delta E_{xx} = 4$ and a $\Delta E_{xx} = 1$.**

$\Delta E_{xx} = 1$		Number of	Average	Maximum	Minimum	Average	Std Dev	95 perc.
	ΔE^{smp}	Nodes	segment	segment	segment	error in %	in %	in %
ΔE_{94}	0.25	4695117	1.001	1.250	0.653	0.098	6.114	13.171
	0.5	4734831	0.998	1.134	0.850	0.200	5.897	12.874
	1	4747514	0.997	1.135	0.845	0.284	5.882	12.827
ΔE_{CMC}	0.25	7245575	1.009	1.476	0.394	0.862	8.721	16.926
	0.5	7399510	1.004	1.413	0.452	0.380	8.988	18.026
	1	7455161	1.002	1.388	0.413	0.239	9.200	18.373
ΔE_{00}	0.25	4245726	1.000	1.624	0.322	0.025	10.288	21.609
	0.5	4327014	0.996	1.521	0.329	0.374	10.606	21.750
	1	4358975	0.995	1.467	0.337	0.503	10.711	22.210

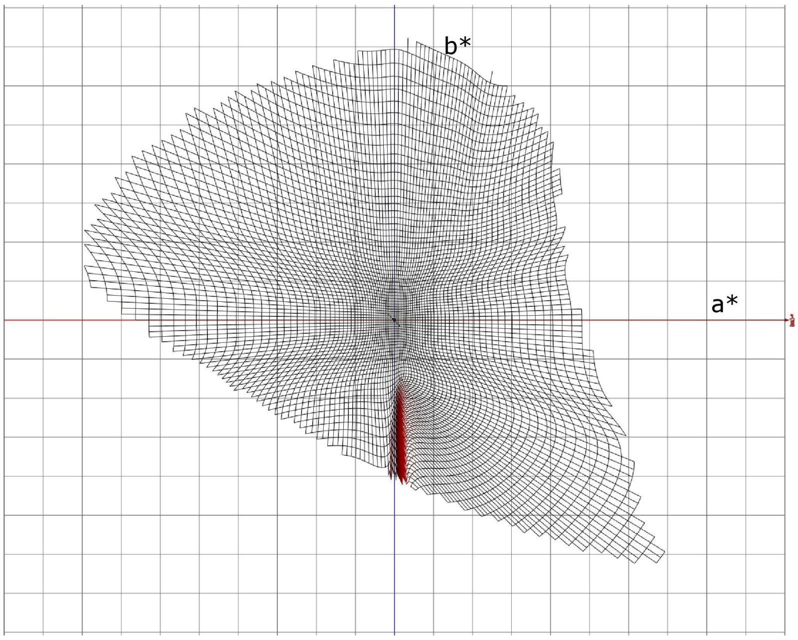


Figure 7: Visualization of our data set in the (a^*, b^*) plane of the $CIELAB$ color space. We set $\Delta E_{00} = 1$ to generate the tabulated space. The data are constrained to the projected spectrum locus boundaries. At each node (sample) corresponds eight segments. Each segment relies two adjacent tabulated samples.

counterpart, color differences are more accurate in this area, so it is a meaningful evaluation. Our method outperforms Urban's method while looking at the average of the segment's size. However, the maximum and minimum length of the segments are better preserved by Urban's grid, which outperforms our algorithm in the extreme cases. This was predictable and is due to the fundamental differences between the two methods. Urban's is more global and limits large errors where the color difference formula lacks continuity and introduces much distortions. Our method is more local and provides more accurate results in most cases but when the data are *closing* the grid. This is where all the error accumulated during the optimization is expressed, such as in Figure 7.

3.2 Analysis on the uniform sampling of $CIELAB$

Judd and Wyszecki(Judd and Wyszecki, 1975) talked about 10,000,000 discernible colors included into the theoretical limits of the colorimetric visual system. Pointer and Attridge(Pointer and Attridge, 1998) considered some restriction in the possible natural spectra (MacAdam limits) and talked of about 2,279,381 colors. The natural scene analysis proposed by Linhares *et al.* (Linhares, Pinto and Nascimento, 2008) talked of about 2,275,698 colors. Although the first ones used a parallelepipedic grid and the last ones used an analysis based on ΔE^*_{00} color difference, the number of discernible colors mentioned in these two works is quite similar.
If we look at our results using ΔE^*_{ab}, we can find a number of discernible colors of 12,163,500 using a distance of 1 units, which is close to the number given by Judd and Wyszecki. On another hand, the JND of 1 seems to be not very well fitted by the ΔE^*_{ab} formulas. If we

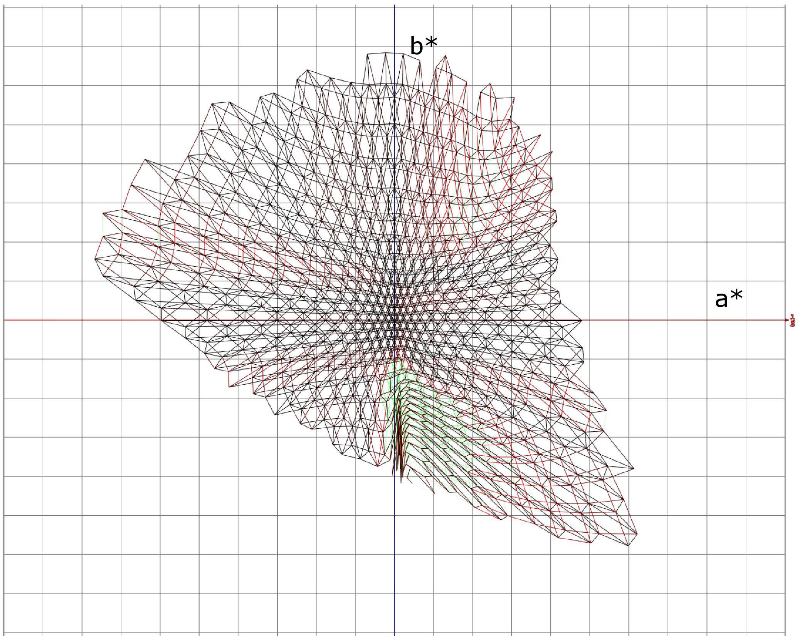


Figure 8: **Uniform sampling of the** $CIELAB\ color\ space\ with\ the\ \Delta E_{00}$ **CDF.**

Table 10: **Comparison of Urban's method and our method for the uniform sampling of the intersection between the spectrum locus and a cylinder of radius 50, based on** $\Delta E_{xx} = 4,\ 2\ and\ 1$**.**

$\Delta E_{xx} = 4$	method	Number of Nodes	Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
ΔE_{94}	Urb	48724	3.704	4.030	3.111	7.404	5.348	18.692
	Us	37760	4.001	4.531	3.413	0.033	5.686	12.867
ΔE_{CMC}	Urb	42713	4.416	6.866	3.478	10.397	11.398	34.496
	Us	62967	3.998	5.450	1.734	0.046	9.887	23.982
ΔE_{00}	Urb	36176	3.876	6.166	2.796	3.093	5.779	14.159
	Us	34426	3.971	5.733	1.521	0.724	8.586	13.716
$\Delta E_{xx} = 2$	method	Number of Nodes	Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
ΔE_{94}	Urb	408947	1.850	2.024	1.540	7.506	5.454	18.862
	Us	321128	1.998	2.266	1.697	0.107	5.807	12.977
ΔE_{CMC}	Urb	346678	2.205	3.412	1.723	10.243	11.467	34.184
	Us	507136	1.999	2.751	0.845	0.042	9.921	21.925
ΔE_{00}	Urb	302682	1.941	3.541	1.364	2.934	6.312	14.487
	Us	292026	1.988	2.877	0.762	0.621	8.893	14.493
$\Delta E_{xx} = 1$	method	Number of Nodes	Average segment	Maximum segment	Minimum segment	Average error in %	Std Dev in %	95 perc. in %
ΔE_{94}	Urb	3344851	0.925	1.019	0.760	7.541	5.496	18.952
	Us	2634667	0.998	1.135	0.845	0.1509	5.853	13.087
ΔE_{CMC}	Urb	2808670	1.103	1.723	0.855	10.256	11.516	34.376
	Us	4094264	1.000	1.388	0.413	0.0186	9.927	22.171
ΔE_{00}	Urb	2487821	0.971	1.864	0.675	2.861	6.650	14.642
	Us	2389836	0.994	1.448	1.448	0.579	9.062	14.779

consider, as Pointer and Linhares, a given number of about 2,300,000 discernible colors then the JND is around 1.78 ΔE^*_{ab} units in average. This value would be considered as an average JND in $CIELAB$ when the sampling is done with the Euclidean distance. In all cases, a variation can be tolerated due to the approximation done on the gamut boundaries and on the pseudo-perceptual uniformity of $CIELAB$ space over ΔE^*_{ab}.
With the method described in this work, we can improve the discussion. If we consider the tabulated data, *i.e.* a parallelepipedic grid, we may find for a distance of 1, 1,000,000 samples for ΔE^*_{94}, 1,700,000 samples for ΔE^*_{CMC} and 900,000 for ΔE^*_{00}. This is very different from the results obtained with an hexagonal uniform sampling. With a uniform sampling, we obtained 2.6 millions samples for ΔE^*_{94}, while with Urban data, we ended up with 3.35 millions. For ΔE^*_{CMC}, we obtained 4 millions, while Urban's data provided 2.8 millions. Finally for ΔE^*_{00}, we are closer for both data set with 2.4 and 2.5 millions. This is very interesting as the number is closed to what has been found by Pointer and Linhares. This is also surprising since with our data, ΔE^*_{94} seems to provide a similar number. However, the difference obtained with the use of one or other data set is that large that further investigation seems to be required in order to agree on the use of such tabulated space.

4 Conclusion

We provided a framework to generate an accurate tabulated sampling of the $CIELAB$ color space based on non-Euclidean color CDFs. We have compared our method with the state of the art and shown that our method performs well compared with existing methods. It exists other methods based on finite elements that could improve the accuracy of our method but their computational cost is more expensive than what we proposed. Although it must be stated that these data needs to be computed only once.
Our tabulated data are available on the author's website as well as supplementary material. These results may be used freely if related to reference to this work.
Possible applications include the design of color targets for device calibration and / or the design of accurate vision tests (more accurate than the Farnsworth-Munsell 100 Hue Color Vision Test). Especially when color space does not have the ability of constant hue, which could limit the use of non uniformly distributed data.

5 Appendix

Considering one color on an axis (defined by a direction), the next color on this axis according this direction will be computed by the procedure BUILDALLAXIS define in 2.
We use the same process for B_+ and B_- with the directions $(0, 0, 1)$ and $(0, 0, -1)$

Considering that we want to create a size $size_a \times size_b$ (defined as $N \times M$ in **Stage 2**) tabulated grid of La^*b^* colors, the center of this grid will be at the (n_a, n_b) position in the corresponding matrix (where $n_a = size_a/2$ and $n_b = size_b/2$) and will correspond to the $a^* = 0$ and $b^* = 0$ coordinate.

Algorithm 2 Axis Sampling Generation

```
procedure FindNextColorOnAxis(in Color_start, direction, ΔE^smp out result)
    factor ← 0.1
    start ← Color_start
    repeat
        end ← direction × factor × ΔE^smp + start
        distance ← (ΔE_xx(start, end) + ΔE_xx(end, start))/2
        if distance ≤ ΔE^smp then
            factor ← factor + 0.25
        end if
    until distance ≤ ΔE^smp
    iter ← 0
    done ← false
    repeat
        med ← (Color_start + end)/2
        distance ← (ΔE_xx(Color_start, med) + ΔE_xx(med, Color_start))/2
        if |distance − ΔE^smp| < eps or iter > MAXITER then
            done ← true
        else
            iter ← iter + 1
            if distance > ΔE^smp then
                end ← med
            else
                start ← med
            end if
        end if
    until !done
    result ← med
end procedure
procedure BuildAllAxis
    L[0] ← (0, 0, 0)
    pos ← 0
    while L[pos] < (100 + ΔE^smp) do
        FindNextColorOnAxis(L[pos], (1, 0, 0), ΔE^smp, L[pos + 1])
        pos ← pos + 1
    end while
    A_+[0] ← (0, 0, 0), A_−[0] ← (0, 0, 0)
    pos ← 0
    while A_+[pos] < A_max do
        FindNextColorOnAxis(A_+[pos], (0, 1, 0), ΔE^smp, A_+[pos + 1])
        pos ← pos + 1
    end while
    pos ← 0
    while A_−[pos] < −A_max do
        FindNextColorOnAxis(A_−[pos], (0, −1, 0), ΔE^smp, A_−[pos + 1])
        pos ← pos + 1
    end while
end procedure
```

Notes: The grid values corresponding the a^* and b^* axis (A and B arrays in this algorithm) are computed during BUILDALLAXIS and the initialization process of $grid$ is done using these values.

Algorithm 3 Grid Optimize

```
procedure OPTIMIZE(in src1, src2, ΔE^smp in/out dstOri)
    dst ← dstOri
    iter ← 0
    done ← false
    repeat
        distance1 ← (ΔE_xx(src1, dst) + ΔE_xx(dst, src1))/2
        distance2 ← (ΔE_xx(src2, dst) + ΔE_xx(dst, src2))/2
        if |distance1 − ΔE^smp| < eps and |distance2 − ΔE^smp| < eps then
            done ← true
        else
            factor1 ← (distance1 − ΔE^smp)/1000
            factor2 ← (distance2 − ΔE^smp)/1000
            dst ← dst − (factor1 × (dst − src1) + factor2 × (dst − src2))
            iter ← iter + 1
        end if
    until iter < MAXITER and !done
end procedure
```

References

CIE 2004. *015:2004, Colorimetry, 3rd edition*, Commission Internationale de l'Eclairage.

Colantoni, P. and Thomas, J.-B. 2009. A color management process for real time color reconstruction of multispectral images, *in* A.-B. Salberg, J. Y. Hardeberg and R. Jenssen (eds), *Lecture Notes in Computer Science*, Vol. 5575 of *16th Scandinavian Conference, SCIA*, pp. 128–137.

Colantoni, P., Thomas, J.-B. and Pillay, R. 2010. Graph-based 3d visualization of color content in paintings, *Proceedings of the 11th VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage, 22-24 September 2010, Paris*, Vol. 2, pp. 25–30.

Foss, C. E. 1978. Space lattice used to sample the color space of the committee on uniform color scales of the optical society of america, *J. Opt. Soc. Am.* **68**(11): 1616–1619.

Gentile, R. S., Allebach, J. P. and Walowit, E. 1990. Quantization of color images based on uniform color spaces, *Journal of imaging technology* **16**(1): 11–21.

Hill, B., Roger, T. and Vorhagen, F. W. 1997. Comparative analysis of the quantization of color spaces on the basis of the cielab color-difference formula, *ACM Trans. Graph.* **16**: 109–154.

Judd, D. B. and Wyszecki, G. 1975. *Color in business, science, and industry / Deane B. Judd and Gunter Wyszecki*, 3d ed. edn, Wiley, New York :.

Kim, D.-H., Cho, E. K. and Kim, J. P. 2001. Evaluation of cielab-based colour-difference formulae using a new dataset, *Color Research & Application* **26**(5): 369–375.

Linhares, J. a. M. M., Pinto, P. D. and Nascimento, S. M. C. 2008. The number of discernible colors in natural scenes, *J. Opt. Soc. Am. A* **25**(12): 2918–2924.

Algorithm 4 Grid Generation

procedure GRIDGENERATION
  $grid$ is a $size_a \times size_b$ matrix of La^*b^* colors
  $gridDone$ is a $size_a \times size_b$ matrix of booleans initialized at $false$
  $sortedPlane$ is a $size_a \times size_b$ array of 2 dimensions vectors which contain all the coordinates of the $grid$ matrix sorted by the distance to the center (n_a, n_b)
  A is an $size_a$ array of La^*b^* colors based of A_- (values before the index n_a) and A_+ (values after the index n_a)
  B is an $size_b$ array of La^*b^* colors based of B_- (values before the index n_b) and B_+ (values after the index n_b)
  for $i \leftarrow 0$ **to** $i < size_a$ **with** $i \leftarrow i+1$ **do**
    $gridDone(i, n_b) \leftarrow true$
  end for
  for $i \leftarrow 0$ **to** $i < size_b$ **with** $i \leftarrow i+1$ **do**
    $gridDone(n_a, i) \leftarrow true$
  end for
  for $i \leftarrow 0$ **to** $i < size_a$ **with** $i \leftarrow i+1$ **do**
    for $j \leftarrow 0$ **to** $j < size_b$ **with** $j \leftarrow j+1$ **do**
      $grid(i, j) \leftarrow A[i] + B[j]$
    end for
  end for
  for $i \leftarrow 0$ **to** $i < size_a \times size_b$ **with** $i \leftarrow i+1$ **do**
    $pos_i \leftarrow sortedPlane[i]_i$
    $pos_j \leftarrow sortedPlane[i]_j$
    $index \leftarrow pos_i \times size_b + pos_j$
    if !$gridDone[index]$ **then**
      if $pos_i - n_a < 0$ **then**
        $direction_a \leftarrow -1$
      else
        $direction_a \leftarrow 1$
      end if
      if $pos_j - n_b < 0$ **then**
        $direction_b \leftarrow -1$
      else
        $direction_b \leftarrow 1$
      end if
      $grid(pos_i, pos_j) \leftarrow (grid(pos_i - direction_a, pos_j) + grid(pos_i, pos_j - direction_b))/2$
      $grid(pos_i, pos_j)_a \leftarrow grid(pos_i, pos_j)_a + A[pos_i]_a - A[pos_i - direction_a]_a$
      $grid(pos_i, pos_j)_b \leftarrow grid(pos_i, pos_j)_b + B[pos_j]_b - B[pos_j - direction_b]_b$
      OPTIMIZE($grid(pos_i - direction_a, pos_j)$, $grid(pos_i, pos_j - direction_b)$, ΔE^{smp}, $grid(pos_i, pos_j)$)
      $gridDone(pos_i, pos_j) \leftarrow true$
    end if
  end for
end procedure

Luo, M. R., Cui, G. and Rigg, B. 2001. The development of the cie 2000 colour-difference formula: Ciede2000, *Color Research & Application* **26**(5): 340–350.

Luo, M. R., Minchew, C., Kenyon, P. and Cui, G. 2004. Verification of CIEDE2000 using industrial data, *Color and Paints, Interim meeting of the International Color Association*, pp. 97–102.

MacAdam, D. L. 1978. Colorimetric data for samples of osa uniform color scales, *J. Opt. Soc. Am.* **68**(1): 121–130.

Mahy, M., Van Mellaert, B., Van Eycken, L. and Oosterlinks, A. 1991. The influence of uniform color spaces on color image processing : a comparative study of cielab, cieluv, and atd, *Journal of imaging technology* **17**(5): 232–243.

Melgosa, M., Huertas, R. and Berns, R. S. 2004. Relative significance of the terms in the ciede2000 and cie94 color-difference formulas, *J. Opt. Soc. Am. A* **21**(12): 2269–2275.

Pointer, M. R. and Attridge, G. G. 1998. The number of discernible colours, *Color Research & Application* **23**(1): 52–54.

Ridolfi, L., Gattass, M. and Lopes, H. 2010. Investigating euclidean mappings for ciede2000 color difference formula, *Color and Imaging Conference* **2010**(1): 327–333.

Sharma, G., Wu, W. and Dalal, E. N. 2005. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations, *Color Research & Application* **30**(1): 21–30.

Stauder, J. F., Colatoni, P. F. and Blonde, L. F. 2006. Device and method for characterizing a colour device, EP1701555.

Stauder, J., Thollot, J., Colantoni, P. and Tremeau, A. 2007. Device, system and method for characterizing a colour device, European Patent WO/2007/116077, EP1845703.

Thomas, J.-B., Chareyron, G. and Trémeau, A. 2007. Image watermarking based on a color quantization process, *Multimedia Content Access: Algorithms and Systems* **6506**(1): 650603.

Thomas, J. B., Colantoni, P. and Tremeau, A. 2013. On the uniform sampling of cielab color space and the number of discernible colors, *IAPR, The fourth Computational Color Imaging Workshop*, pp. 53–57.

Thomas, J.-B. and Trémeau, A. 2007. A gamut preserving color image quantization, *ICIAPW '07: Proceedings of the 14th International Conference of Image Analysis and Processing - Workshops*, IEEE Computer Society, Washington, DC, USA, pp. 221–226.

Trémeau, A., Konik, H. and Lozano, V. 1996. Limits of using a digital color camera for color image processing, *Proceedings of the IS&T/OSA Optics & Imaging in the Information Age, 20-24 October 1996, Rochester, New York*, pp. 150–155.

Urban, P., Rosen, M. R., Berns, R. S. and Schleicher, D. 2007. Embedding non-euclidean color spaces into euclidean color spaces with minimal isometric disagreement, *J. Opt. Soc. Am. A* **24**(6): 1516–1528.

Wyszecki, G. 1954. A regular rhombohedral lattice sampling of munsell renotation space, *Journal of the optical society of america* **44**(9): 725–734.