Fast Computation of Multi–Primary Color Gamuts

Ján Morovič
Hewlett–Packard Company, Barcelona, Spain

Abstract

As the number of primaries used in an imaging system increases, so does the cost of computing the resulting color gamut. While for three or even four primaries it is possible to use exhaustive techniques to generate a relatively small number of samples that provide sufficient gamut boundary information, taking the same approach with devices using a larger number of primaries (e.g. 6+) leads to a combinatorial explosion. To address this issue while providing accurate information about the color gamut of a multi–primary device, the present paper describes the PSS primary space sampling algorithm that reduces gamut computation time from centuries to seconds.

Introduction

Since William Kurtz's first mechanically produced three–color engraving in 1893,[1] the science and technology of color imaging has advanced dramatically on many fronts. Most importantly, form this paper's point of view, the number of primaries (e.g. inks, toners, phosphors, filters, etc.) used in imaging systems including printers, displays and projectors has been increasing steadily from the initial sets of three used in color reproduction.

In printing, ink–sets have increased from the initial cyan, magenta and yellow via the addition of a black to some current printers having as many as 12 inks. Displays and projectors too have seen the addition new primaries to complement the traditional red, green and blue and systems with up to six primaries are now available.

While the benefits of using more primaries in an imaging system are clear, doing so also brings with it new challenges on a number of fronts. The most critical of them is how to make use of all the primaries so as to obtain the best result, whereby 'best' includes: having the largest color gamut, being least metameric and/or most color constant, providing the smoothest transitions and being most stable. Another challenge that using a greater number of primaries introduces is the question of how to compute their color gamut. In general gamut computation is based on sampling all possible combinations of varying amounts of the primaries used in a system, predicting the color appearances of these combinations and finally describing the volume in color space that these color appearances inhabit.

Being able to determine the color gamut of a multi–primary system plays a key role first in being able to compare alternative sets of primaries in terms of their ability to address color space and second in then being able to determine how well a particular separation covers the color gamut possible with a given primary set. What is meant by color separation here, is the process where primary combinations are assigned to color space coordinates, which especially in the case of a multi–primary system involves making a choice among metameric combinations that address the same color space coordinates.

To see what implications the use of an increased primary set has on gamut computation, the following section will describe what would happen if current approaches (used for three or four primary imaging systems) were applied to an eight or eleven primary system. Following this illustration of the problem, a novel primary space sampling technique will be presented that dramatically reduces gamut computation times.

Exhaustive sampling

If we take the case of a three primary imaging device and use k=2 samples along each primary's dimension (i.e. 0% and 100%) the result are the following s=8 samples: [0, 0, 0], [0, 0,1], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 1]. Outputting them, measuring their colors and obtaining a gamut boundary from the result would not require much in terms of resources and would be quick. However, in practice more than two samples need to be used per primary dimension to get greater accuracy.

Typically values of k=9 or 11 would be used, resulting in s=729 or 1331 combinations, which could be measured in a matter of some minutes. In general s=k^n where n is the number of primaries, k is the number of samples per primary dimension and s is the total number of samples.

While the above picture presents no challenges, adding more primaries to an imaging system soon results in an exponential explosion and the number of samples that need to be output and measured gets quickly out of hand. Taking, for example, a printer like the HP Designjet Z3100 that has 11 inks, would result in s=2048 samples for the coarsest sampling of k=2 and over 31 billion samples for the more reasonable sampling of k=9. To print this latter sampling would require more than 31 million sheets of A4 and their measurement would take just under 12 centuries.

To speed up the process, the first thing that is typically done is to use a mathematical model that, given a specific primary space vector predicts the color that would be obtained if it were printed, displayed or projected. The use of such models speeds things up dramatically and having k=9 samples for the above 11 primary system now takes 'only' five days to compute on a dual 2 GHz G5 system as opposed to the 12 centuries it would take to print and measure the samples. However, even substituting modeling for printing and measurement still makes sampling primary combinations exhaustively take up an impractical amount of time (Fig. 1).

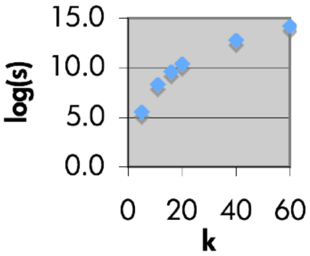

Fig. 1. Relationship of number of samples per primary dimension (k) and total number of samples (s) for exhaustively sampling an 8 primary system.

A further complication that is added when more than the typical three or four primaries are used is that, with the addition of new primaries, it is no longer safe to assume that the system's color gamut will be convex (or near–convex) in color appearance space. With the convexity assumption no longer valid, larger values of *k* are needed (e.g. 40 or more as opposed to around 10) to allow for a correct distinction between gamut surface concavities and sampling gaps. Taking even just *k*=25 for the 11 primary system mentioned above and computing all possible resulting combinations would again take close to 12 centuries to complete.

What is therefore required is a new kind of sampling that gives the same result as the exhaustive technique while considering only a fraction of the possible combinations.

Pair–Surface Sequential (PSS) Sampling

With the aim of providing a significantly smaller number of total samples (*s*) the following two properties of gamut computation are identified and exploited:

First, the anatomy of a device color gamut surface makes one of its parts have specifically different properties in terms of primary combinations, whereby the two parts – a lighter and a darker one – join along the line of cusps (i.e. colors at each hue that have maximum chroma) (Fig. 2). In particular the part of a gamut surface, which contains the color obtained by setting all primaries to zero and consists of colors obtained by mixing at most two of the *n* primaries. For subtractive systems, it is a gamut surface's lighter part where all colors are the result of mixing at most two primaries – adding a third primary to any two–primary combination only results in a color that has lower lightness and is therefore not on the lighter part of the gamut surface. Conversely for additive systems, it is a gamut surface's darker part that has this at–most–two–primaries property as adding a third primary results in a lighter color that is no longer on the darker part of the gamut surface.

Second, the gamut boundary is a three–dimensional surface. Since the boundary is three dimensional in color appearance terms it can also be represented by a 3D subspace of the *n* dimensional primary space (i.e. a 3D subspace of *n*D exists in which the gamut can be represented and will biject onto the gamut in color appearance space). The consequence of this is that parts of the nD space can be discarded by looking at what they map to in color appearance without having to sample the primary space exhaustively.

Algorithm details and pseudo–code

By exploiting the above insights, a sampling technique has been developed that consists of two stages:

Stage 1: color gamut of primary combinations where only up to two primaries have non–zero values. This involves generating $k^2 * n * (n-1)/2$ samples (i.e. exhaustively sample all the 2D faces of the nD hypercube that have the origin as one of the vertices) and computing their color appearance gamut (via predicting color appearance for each primary vector). A key here is that not only color appearance coordinates are stored for gamut boundary colors but also corresponding primary combinations. The result of this first stage is a set of nD vectors that are candidates for being on the gamut boundary of the primary set in color appearance space and where each vector has at most two non–zero values.

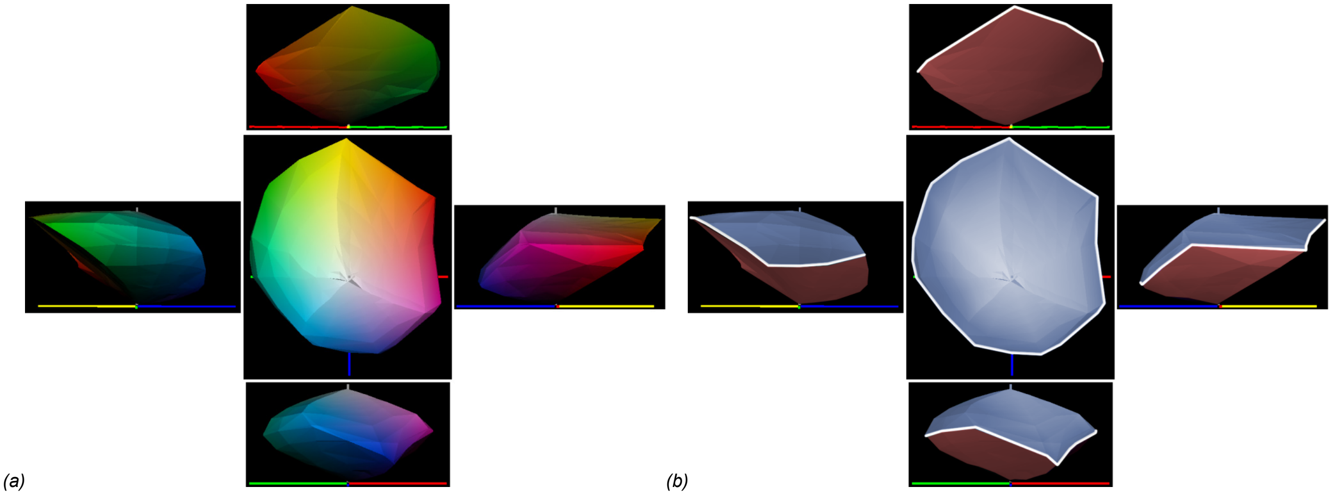

Fig. 2. . Lighter and darker part of a color gamut: (a) projections of color gamut along L axis and a* and b* half–axes, (b) colorized projections: blue – lighter gamut part, red – darker gamut part, white – line of cusps.*

In pseudo–code this stage is as follows:

```
create empty sample set
for i from 1 to (n-1)
    for j from i+1 to n
        set all primaries to 0
        for l from 0 to 1 in k steps
            for m from 0 to 1 in k steps
                set primary i to l
                set primary j to m
                compute color of primary combination
                add color and primary values to sample set
            end for
        end for
        set primaries i and j back to zero
    end for
end for
compute gamut boundary of sample set
```

Here the computation of the color of a primary combination involves using a characterization model to predict the stimulus corresponding to a combination of primary amounts followed by color appearance prediction from stimulus colorimetry and viewing condition parameters.

Stage 2: Sequentially extruding gamut boundary primary vectors along each dimension in turn. The second stage then is a sequential taking into account of the effect of the system's primaries. For the first primary this is done by taking the primary combinations that resulted in the gamut boundary at the end of the first stage and varying the first primary's value while keeping the other primaries' values unchanged. The result is an intermediate approximation of the gamut where the gamut boundary candidate ink vectors now have up to three non–zero values. Next the same is done for the second and all subsequent primaries, whereby it is always the outcome of having varied the previous primary that is further refined.

In pseudo–code this stage is as follows:

```
set current gamut boundary to gamut boundary from stage 1
for i from 1 to n
    set previous gamut boundary to current gamut boundary
    empty current gamut boundary
    create empty sample set
    for all vertices of previous gamut boundary
        for m from 0 to 1 in k steps
            set vertex primary i to m
            compute color of primary combination
            add color and primary values to sample set
        end for
    end for
    compute current gamut boundary as gamut of sample set
end for
return current gamut boundary
```

Extruding primary vectors

One of the key concepts here (in addition to the realization that device/medium gamuts have two distinct parts joined along the line of cusps) is the idea of taking a set of primary vectors and extruding them along the primary space's dimensions one at a time (i.e. stage 2 of PSS). To illustrate this idea, let us consider a set of vectors in 4D that have at most two non-zero values – i.e., like the kind of primary vectors from stage 1 of PSS. What extruding them means, e.g., along the first dimension, is to take the value of each vector in that dimension and vary it across the whole valid range. In Fig. 3 the left side shows three sets of vectors with at most two non-zero values – each set having zeros in at least the two dimensions indicated (i.e. always in the first and then on one each of the remaining three). The right side of Fig. 3 then shows lines along which the extruded versions of these vectors lie in 4D. Note that these extruded versions of the initial vectors have at most three non-zero values (i.e. the at most two non–zero values they started with plus the non-zero value in the first dimension). This process of extrusion can then proceed along any of the dimensions – always varying the values in only one member of a set of vectors.

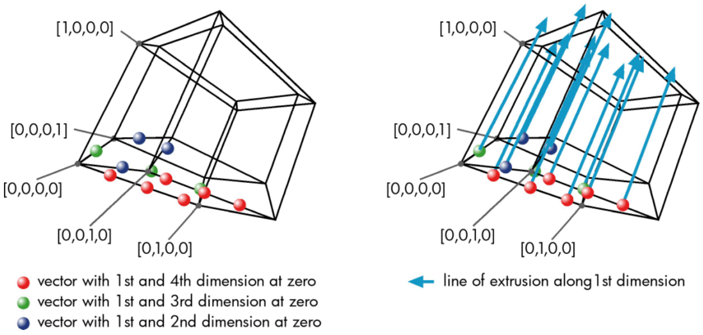
[1,0,0,0]
[0,0,0,1]
[0,0,0,0]
[0,0,1,0]
[0,1,0,0]
[1,0,0,0]
[0,0,0,1]
[0,0,0,0]
[0,0,1,0]
[0,1,0,0]
vector with 1st and 4th dimension at zero
vector with 1st and 3rd dimension at zero
vector with 1st and 2nd dimension at zero
line of extrusion along1st dimension

Fig. 3. 4D vectors with at most two non-zero members in three of the four dimension (left) and their extrusion along the first dimension in which they all had a zero value initially.

Requirements

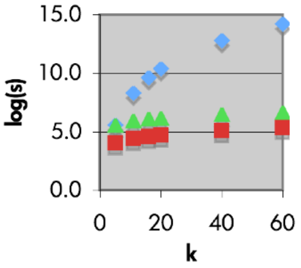

Fig. 4. Relationship of number of samples per primary dimension (k) and total number of samples (s) for an 8 primary system. In the plot cyan diamonds represent exhaustive sampling, and red squares and green triangles show PSS sampling using 256 and 10000 gamut boundary samples (g) respectively.

A key requirement of this PSS approach is also the use of a gamut boundary descriptor that allows for the description of concavities (e.g. modified convex hull,[2] alpha shapes,[3] segment maxima,[4] spline surface[5]). The total number of samples used by PSS for describing a multi–primary gamut is $k^2 * n * (n-1)/2 + n * g *$k where g is the number of vertices used for describing a gamut boundary (for techniques that use a fixed number of gamut boundary vertices). This is a dramatically smaller number than the k^n of exhaustive sampling (Fig. 4) and even for k=60 PSS sampling takes only three seconds to compute.

Given that the above computation relies on a mathematical model of the color imaging device, which is likely to have some differences from actual output, the result of the PSS sampling can also be used to obtain direct measurements of the device's gamut using a small number of color patches. As the computation results in a small number of gamut boundary points (e.g. 200–300 are typically sufficient) and as the primary combinations that resulted in them are known, these primary combinations can simply be output and measured. The result then is measured gamut boundary information based on PSS sampling.

Evaluating PSS sampling

The two key aspects of PSS that need to be tested are: first, dependency of result on order in which primaries are considered in the sequential part of the algorithm and second, overall accuracy as compared with exhaustive sampling.

In both these cases CIECAM02[6] was used as the color appearance space and the single constant Kubelka–Munk model[7] was used to predict printed reflectance from primary vectors and predictions for an 8–ink inkjet system on a glossy substrate. The gamut boundary computation was performed using the segment maxima technique[4] using 256 gamut boundary samples. All differences are in CIECAM02 *Jab* space, where *a* and *b* are orthogonal equivalents of chroma (*C*) and hue (*h*).

Effect of primary order

To test the effect of the order in which primaries are considered by the PSS technique, the gamut boundary was computed for all permutations of the 8 inks (i.e. 8! = 40320). The volumes of these more than 40 000 gamuts were compared with their mean and it was found that the range of differences was from –0.65% to +0.51%. In other words, on average the effect of primary order on gamut volume is only about ±0.5%. I.e. for a gamut with a volume of 600 000 cubic CIECAM02 Jab units this corresponds to ±3000. The effect of primary order can therefore be considered as negligible and can be ignored.

PSS accuracy

The key requirement for PSS is to provide samples that result in a gamut boundary that is very similar to the one obtained by using exhaustive sampling of all primary value combinations. To test this property the difference between an exhaustively computed (Ge) and a PSS (Gpss) gamut boundary will be computed by taking all the gamut boundary vertices of Ge and computing the minimum color difference between them and the Gpss boundary if they are outside it. Difference will be se to zero when the Ge vertex is inside the Gpss boundary. The same will be done for Gpss points relative to Ge – only these color differences will be made negative as they represent cases where the PSS gamut exceeds the exhaustive gamut and in which case they are not errors of the PSS sampling but of the exhaustive technique.

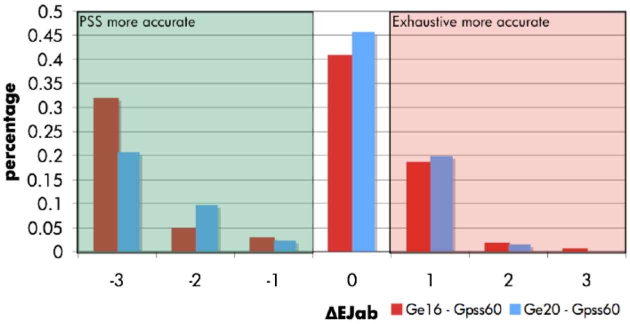

Fig. 5. Differences of Ge–Gpss for exhaustive k=16 or 20 and PSS k=60.

Fig. 5 shows the distribution of differences computed in the above way. First it can be seen that the range of differences is [–3,2] ΔEJab (i.e. Euclidean distance in CIECAM02 Jab space) whereby in the vast majority of cases (80%) PSS is more accurate than or equal to the exhaustive computation and only in 2% of cases does the PSS boundary under–predict the exhaustive boundary my more than 1 ΔEJab.

Also useful is to see how the exhaustive and PSS techniques compare when the number of samples per channel is the same for both and Fig. 6 shows this when k=20. As can be seen the accuracy of the PSS technique is virtually the same when k=20 as when k=60 and this is the case due to the sampling approach PSS takes (i.e. where it ends up having more samples on the actual gamut boundary and therefore results in less risk of false concavities).

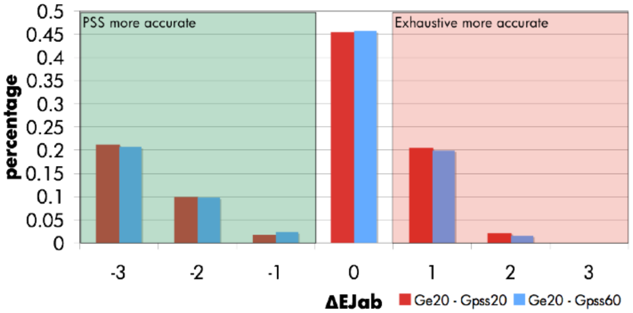

Fig. 6. Differences of Ge–Gpss for k=20.

It can be seen that in this case PSS gives a very accurate prediction of the printing systems' color gamut in a way virtually invariant of sampling sequence order and even for 60 samples per channel it completes the computation in 1/8000000th of the exhaustive computation's time while the exhaustive computation only uses 20 samples per channel and in many cases under–predicts the gamut.

Conclusions

The pair–surface sequential (PSS) sampling technique introduced here has been shown to provide accurate predictions of *n*–primary output imaging systems in a matter of seconds, as compared to the days or even years it would take an exhaustive computation to give an equivalent result.

The advantages of PSS are that it allows for accurate, non–convex, on–the–fly, *n*–primary gamut computation at high speed and its results can be used both in the development of multi–color separation and for the visualization of color gamuts possible with arbitrary sets of additive or subtractive primaries.

Acknowledgments

I would like to thank Dr. Peter Morovič for his advice and Virginia Palacios and Jordi Arnabat (Hewlett–Packard Company) for supporting this work.

References

1. Bourges J. (1997) *Color Bytes. Blending the Art and Science of Color*, Chromatics Press, Forest Hills, New York, USA.
2. Balasubramanian R., Dalal E. (1997) A method for quantifying the color gamut of an output device, *Proceedings of SPIE*, **3018**:110–116.
3. Cholewo T. J. and Love S. (1999) Gamut Boundary Determination Using Alpha–Shapes, *Proceedings of 7th IS&T/SID Color Imaging Conference*, 200–204.
4. Morovic J. and Luo M. R. (2000) Calculating Medium and Image Gamut Boundaries for Gamut Mapping, *Color Research and Application*, **25**:394–401.
5. Shaw M. (2006) Gamut estimation using 2D surface splines, *Proceedings of SPIE*, **6058**:605807.
6. CIE (2004) *CIE 159:2004: A colour appearance model for colour management systems: CIECAM02*, CIE Central Bureau, Vienna
7. Kubelka P. and Munk F. (1931) Ein Beitrag zur Optik der Farbanstriche, *Zeitschrift für technische Physik*, Germany, **12**:593–601.

Author Biography

Ján Morovič received his Ph.D. in color science from the Colour & Imaging Institute (CII) of the University of Derby (UK) in 1998, where the title of his thesis was To Develop a Universal Gamut Mapping Algorithm. After working as a lecturer in digital color reproduction at the CII he became senior color scientist at Hewlett–Packard in Barcelona, Spain. He is also the chairman of the CIE's Technical Committee 8–03 on Gamut Mapping.