Volume xx (200y), Number z, pp. 1–17

Modeling Color Properties of Tiled Displays

Aditi Majumder[†], M. Gopi[‡]

Abstract

The concept of tiled displays can be successful only if such displays are made to look like a single display perceptually. The two issues that need to be solved to achieve this goal are geometric correction and color seamlessness of images spanning across tiles. Geometric correction algorithms borrow pin-hole camera models to model projector display geometry. In this paper, we introduce an abstract modeling function that describes the color seen by a viewer when displayed by a display device.
Though this function can be used to model color displayed by any common display device, in this paper, we use it to model color in multi-projector display systems. We use the model to explain the reasons for different types of color variations in a multi-projector display, to compare different color correction algorithms, and to derive such algorithms directly from the model.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation[display algorithms] I.4.0 [Image Processing and Computer Vision]: General[image displays] I.4.8 [Image Processing and Computer Vision]: Scene Analysis[color and photometry] H.1.2 [Models and Principles]: User/Machine Systems[human factors]

Keywords: Projection-Based Displays, Tiled Displays, Color Model

1. Introduction

Traditional 21-inch desktop monitors have resolution of 60 pixels per inch and $20 - 30$ degree field-of-view. But life-size displays used in scientific visualization require display area in the order of hundreds of square feet, high resolution of hundreds of pixels per inch, and wide field-of-view of more than 120 degrees. Such large displays can be used in defense training applications, in life-size teleconferencing, and also to create high quality immersive virtual reality (VR) environments.

A scalable approach to build large displays is to tile multiple projectors to create a single large display. There are three primary issues that need to be addressed in building multi-projector display systems.

(a) *Data management, data distribution and driving architecture*: This deals with handling large 2D or 3D data content, distributing them to individual machines that drive the projectors, and the configuration of the connectivity of these machines.
(b) *Geometric seamlessness:* The image displayed on a multi-projector display comes from several different physical devices and hence may not be aligned geometrically along the projector boundaries. Further, the image within the same projector may be radially distorted.
(c) *Color seamlessness:* Multi-projector displays can be arranged in abutting or overlapping configurations. In both these cases, colors and/or brightness varies within and across multiple projectors. In the overlapping configuration, higher brightness in the overlap regions are also introduced. Thus, even with the same input value at every pixel of a multi-projector display, the output colors are not the same for all pixels.

Solutions to the data management and driving architecture problems in multi-projector systems ([70, 33, 8, 34, 32]) use existing parallel and streaming data models ([55, 57, 58, 60, 56]). Solu-

[†] Department of Computer Science, University of California, Irvine. Email: majumder@ics.uci.edu .
[‡] Department of Computer Science, University of California, Irvine. Email: gopi@ics.uci.edu.

tions to correct the geometric misalignment problem ([79, 67, 65, 64, 78, 31, 66, 14, 11, 68, 28]) use the pin-hole camera model ([22]) to model individual projectors. However, there exists no model that describes the color variation in multi-projector displays that can be used to address the color seamlessness problem. Hence, the solutions presented for this problem in [45, 72, 73, 77, 59, 47, 65, 10, 37, 46, 79] provide only partial solutions under restricting assumptions and with varying degrees of success.

Solutions for other issues related to multi-projector systems were borrowed from existing models for other systems and devices. For the color variation problem, we cannot borrow models developed for other display devices due to many reasons. For example, there exist models for colorimetry of single CRT monitors [6, 5, 7]. But these models assume that for a given input, color remains spatially constant over the entire display. This assumption is not valid for projectors. Unlike CRTs, the physical space of a projector is decoupled from its display space. This introduces several physical phenomena like distance attenuation of light, lens magnification and non-Lambertian reflection of the display screen that cause severe spatial variation. For the same reason, projectors are different from LCD panels also. Further, there exists no model that describes color variations in multi-device displays with either CRT or LCD panels. Hence, to fill this gap, we need a color model for multi-projector displays.

A color model for multi-projector displays can be useful in many ways. First, this can help us represent the color variation in such displays in a compact manner using just a few parameters. Second, it will provide us with a tool to compare and evaluate different algorithms to solve the color variation. Finally, we can use the modeling function to systematically develop new methods that address the problem in an organized manner.

1.1. Main Contributions

In this paper, we introduce a function that models the color variation in multi-projector displays comprehensively, compactly, and accurately. The color properties of such displays have been studied thoroughly in [42, 49] to identify a small set of parameters that cause color variation and the relationships between them. With these parameters, we develop a function to represent the color property of the display compactly. We verify this function by comparing the color response predicted by it with the actual response of the display, for an image projected on the display.

We also show that this modeling function is more general and can be used to model other image display or capturing devices. Next, we demonstrate the different uses of this modeling function in explaining the different color variations in multi-projector displays. In addition to its use in modeling, this function also provides us with a framework to compare different color correction algorithms by formally defining the common goals that these algorithms try to achieve.

Further, we demonstrate that this model can be used to design color correction algorithms by directly deriving one of the existing solutions from it. Finally, we use this model to provide a few insights into the color correction problem to help us design new algorithms in the future.

1.2. Outline of the paper

In Section 2 we develop the function that models the color variation in multi-projector displays. In Section 3, we demonstrate the generality of the model by showing that existing models for other display devices is a special case of the proposed function. In Section 4, we explain the various color varations using different parameters of the model. In Section 5, we present the several ways in which this model can be used for *analysis* of various color correction methods, *reconstruction* of the different model parameters, and in *derivation* of an existing algorithms from this model. Finally we conclude by mentioning a few insights we get from this work that would help us design future color correction algorithms.

2. Modeling Color Variation

In this section, we first present some basic definitions and then derive the function modeling the color variation is a multi-projector display.

2.1. Color Operators

We define a color K at a point on the display by the tuple (L,C) where L is the *total tristumulus value (TTV)* defined by the sum of CIE tristimulus values of the color, $X + Y + Z$, and C is the chrominance defined by the *chromaticity coordinates* (x, y):

$$x = \frac{X}{X+Y+Z}; \quad y = \frac{Y}{X+Y+Z}.$$

The TTV L is the indicator of power/intensity and chrominance C, the hue and saturation of the color. We use the notation $L = ttv(K)$ and $C = chr(K)$. Note that L is different from the luminance Y of the CIE tristimulus values of the color.

We introduce two operators for colors: the addition operator and the intensity scaling operator. The addition operator is used when light from multiple sources superimpose in the overlap region.

Addition Operator: The addition of two colors $K_1 = (L_1, C_1)$ and $K_2 = (L_2, C_2)$ is defined as

$$K = (L, C) = K_1 \oplus K_2,$$

where

$$L = \sum_{i=1}^{2} L_i,$$

$$C = \frac{\sum_{i=1}^{2} L_i C_i}{L}.$$

So, by adding two colors, the TTV of all the constituting colors add up, and the chrominance of the different colors are weighted by the TTV proportion of each of them (given by L_i/L). Note that $\oplus$ is both commutative and associative.

We make an important observation here. If $K_1 = (L_1, C_1)$, $K_2 = (L_2, C_2)$, and $C_1 = C_2$ then the chrominance C of $K = K_1 \oplus K_2$ is $C = C_1 = C_2$.

Intensity Scaling: The intensity scaling of a color $K_1 = (L_1, C_1)$ by a scalar k is defined as

$$K = (L, C) = k \otimes K_1,$$

where

$$L = kL_1,$$

$$C = C_1.$$

Note that $\otimes$ does not change the chrominance of a color and distributes over $\oplus$.

2.2. Definitions

A *multi-projector display* is a display made of N projectors, projecting on a planar or non-planar display screen. Each projector is denoted by $P_j, 1 \leq j \leq N$. The coordinates of the 2D display screen (not necessarily planar) are denoted by (u, v) and individual projector's display region within this display screen is parameterized by (s, t). The geometry of the display screen S, and the positions of the projectors (p_x, p_y, p_z) and the viewer $e = (e_x, e_y, e_z)$ are described in a 3D world coordinate system.

The display and projector coordinate pairs can be related by a *geometric warp*, G,

$$(u, v) = G(s, t, p), \tag{1}$$

where $p = (p_x, p_y, p_z, \theta, \phi, f, S)$. The parameters f and (θ, ϕ) are the focal length and orientation of P respectively, expressed in the world coordinate space. For all practical systems, p does not change because projectors and screen do not move relative to each other. Hence,

$$(u, v) = G(s, t). \tag{2}$$

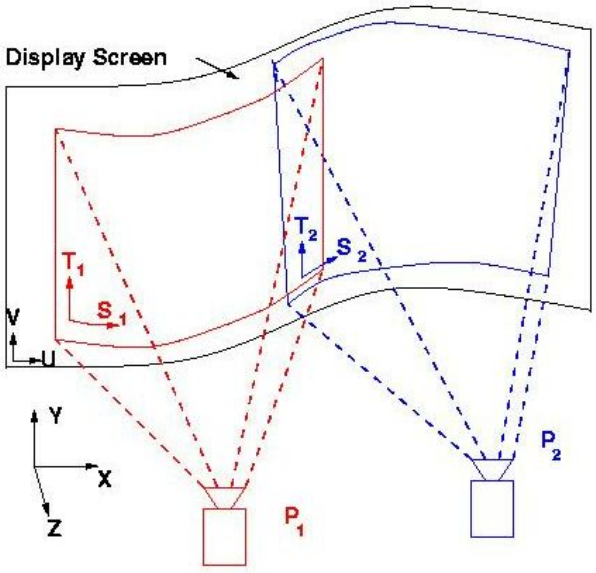

Figure 1: *Projector and display coordinate space.*

Figure 1 shows a two-projector display wall. The blue and red quadrilaterals show the areas of projection of projectors P_1 and P_2 on the display screen.

A projector has three channels, $\{r, g, b\}$. A channel is denoted by $l \in \{r, g, b\}$ and the corresponding input by $i_l \in \{i_r, i_g, i_b\}, 0.0 \leq i_l \leq 1.0$.

The Bidirectional Reflectance Distribution Function (for the front projection display) or the Bidirectional Transfer Distribution Function (for the back projection display) of the screen is dependent on the display coordinates (u, v) and the viewer location e, and is denoted by $\Lambda(u, v, e)$. We assume that the BRDF/BTDF is independent of chrominance.

2.3. The Model

In this section we introduce the function to model display devices in general, and multi-projector display system in particular.

Definition: The function $E(u, v, i, e)$ that models the color characteristics of a display is defined as the color reaching the viewer e from the display coordinate (u, v) when the input to the device coordinates of all the display devices that contributes directly to the color at the display coordinate (u, v) is $i = (i_r, i_g, i_b)$.

Note that more than one projector can contribute to the display coordinate (u, v) in the overlap region. Further, the function is defined only for devices with three

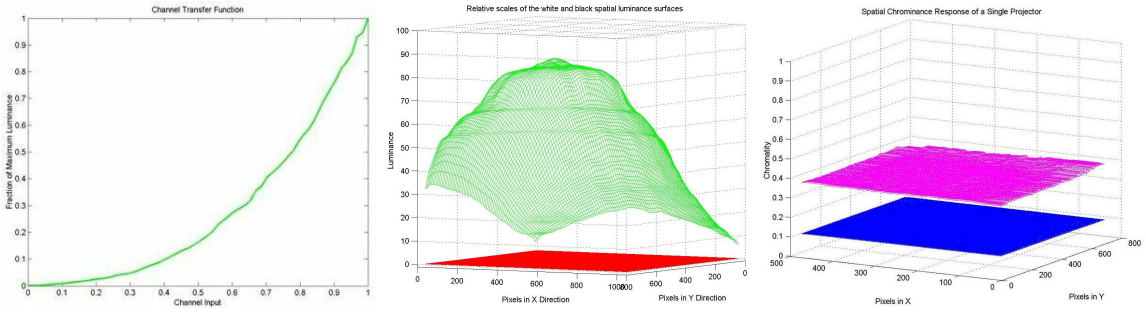

Figure 2: *Reconstructing the model parameters for a single projector. Left: The input transfer function for green channel. Middle: The TTV function for green channel and the black TTV function. Right: The chrominance function (x, y) for the green channel.*

primaries[†]. It is also clear from the definition that the function considers only the direct lighting and ignores the indirect lighting due to reflections and secondary scattering at the display coordinate.

In this section we will derive $E(u, v, i, e)$ for a single projector display followed by the same for a multi-projector display.

2.3.1. Single Projector Display

Let us consider one projector coordinate (s, t). Let $Q_l(s, t)$ be the maximum intensity that can be projected at that coordinate from channel l. For any input i_l, the intensity projected is a fraction of $Q_l(s, t)$ and is given by $h_l(s, t, i_l)$, where $0.0 \leq h_l(s, t, i_l) \leq 1.0$. Let the chrominance projected at that coordinate for input i_l be $c_l(s, t, i_l)$. Thus, the color projected at this coordinate (s, t) for input i_l in channel l is given by

$$D_l(s, t, i_l) = (h_l(s, t, i_l) Q_l(s, t) , c_l(s, t, i_l)) \quad (3)$$
$$= h_l(s, t, i_l) \otimes (Q_l(s, t) , c_l(s, t, i_l)) \quad (4)$$

We call c_l as the *chrominance function* and Q_l as the *intensity function* of the channel l.

An ideal projector satisfies the property of *channel constancy* which says that the colors projected by all inputs i_l of channel l have a constant chrominance and differ only in intensity. Thus, $c_l(s, t, i_l)$, which depends on the projector color filters and lamp characteristics, is independent of the channel input i_l. Hence,

$$D_l(s, t, i_l) = h_l(s, t, i_l) \otimes (Q_l(s, t) , c_l(s, t)). \quad (5)$$

In [42, 62], h_l is shown to be independent of (s, t). Hence, it is denoted by just $h_l(i_l)$ and is called the *transfer function* for channel l. Hence, we have

$$D_l(s, t, i_l) = h_l(i_l) \otimes (Q_l(s, t) , c_l(s, t)). \quad (6)$$

Thus, D_l is now expressed as a multiplication of two independent functions: h_l depends only on the input i_l, and (Q_l, c_l) depend only on the spatial coordinates (s, t). Note that h_l is similar to the gamma function in other displays. It has been shown in [42, 62] that in projectors, this function can be non-monotonic and often cannot be expressed by a power (γ) function. Hence, for generality, we call it *transfer function* for channel l. Figure 2 illustrates the parameters Q_l, h_l and c_l.

An ideal projector satisfies the property of *channel independency* by which the color projected from different channels of a projector are independent of each other and the total color $T(s, t, i)$ projected at (s, t) for input $i = (i_r, i_g, i_b)$ is the superposition of colors from individual channels. That is,

$$T(s, t, i) = D_r(s, t, i_r) \oplus D_g(s, t, i_g) \oplus D_b(s, t, i_b) \quad (7)$$

In practice, however, some extra leakage light is projected at all times, in addition to the light projected from input i. This leakage light is called the *black offset* and is represented by $(B(s, t), c_B(s, t))$, where $B(s, t)$ is the spatially varying TTV component called the *black intensity function* and $c_B(s, t)$ is the *black chrominance function*. Thus, the light projected by a projector is

$$T(s, t, i) = D_r \oplus D_g \oplus D_b \oplus (B(s, t), c_B(s, t)). \quad (8)$$

Finally, for a viewer at e, the function $E(u, v, i, e)$ for a single projector is given by

$$E(u, v, i, e) = \Lambda(u, v, e) \otimes T(s, t, i), \quad (9)$$

[†] The DLP projectors that use a clear filter for projecting the grays behave like a four primary system.

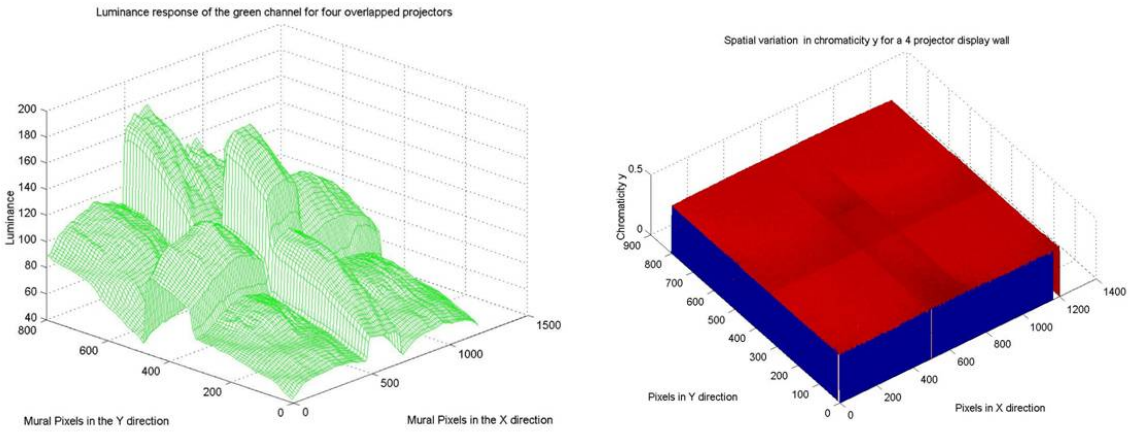

Figure 3: *This shows the the function E at input $(1, 1, 0)$ for an tiled display made of 2×2 array of four-projectors. The $ttv(E)$, (left) and y component of chrominance, $chr(E)$ (right) is reconstructed from a camera image. The details of the reconstruction process is presented in Section 5.3.*

where $(u, v) = G(s, t)$.‡

2.3.2. Multi-projector Display

Using the function derived for a single projector display, we now derive the function for a multi-projector display. Let N_P denote the set of projectors overlapping at (u, v). The function $E(u, v, i, e)$ for a tiled display is then given by the superposition of colors projected for the same input i from all the projectors overlapping at the display coordinate (u, v).

$$E(u, v, i, e) = \oplus_{j\in N_P} E_j(u, v, i, e). \tag{10}$$

For Lambertian screen, Equation 9 and hence 10 become independent of e, and E is reduced to

$$E(u, v, i) = \oplus_{j\in N_P} T_j(s_j, t_j, i). \tag{11}$$

Figure 3 shows the E for a four projector display for input $(1, 1, 0)$.

3. Modeling Other Devices

The function E, though designed for multi-projector displays, can be used to model any image-display device (single or tiled) or any image-capture sensor (single). The parameters of the function, h_l,Q_l,c_l, can be different for different devices. In this section we show that the existing models of different color devices are a special case of our general model.

‡ When p used in Equation 1 is not assumed to be static, G, Q_l, B, c_l and c_B are dependent on p also.

3.1. Non-Linear Device

Let us first consider single non-projection displays like CRT monitors that do not involve screens. Existing models of such displays [6, 5, 7] make the following assumptions.

1. There is no black offset, i.e. $B(s, t) = 0$.
2. There is no spatial variation of intensity or chrominance, i.e. $Q_l(s, t)$ and $c_l(s, t)$ are spatially constant. Let these constant values which are independent of (s, t) be M_l and C_l respectively.
3. $h_l(i_l)$ can be expressed by a power function, i.e $h_l(i_l) = (A\ i_l)^\gamma$, where A and γ are constants.

When using our model for such displays, since we are dealing with a single display, we use Equation 8. When the above assumptions are imposed on our model, we find that the color generated for input $i = (i_r, i_g, i_b)$ by a CRT monitor is

$$\begin{aligned} E(i) &= \oplus_{l\in r,g,b} \left((A\ i_l)^{\gamma_l} \otimes (M_l, C_l)\right) &(12)\\ &= \oplus_{l\in r,g,b} ((A\ i_l)^{\gamma_l} M_l, C_l) &(13)\end{aligned}$$

This model of the CRT monitor arrived at from E matches with the model presented in [6, 5, 7] and is achieved by substituting device-specific functions and assumptions for the parameters to our model.

3.2. Linear Device

Another popular model that has been used extensively in color matching and color profiling applications for modeling various devices, like cameras, scanners, printers and monitors, assume linear devices (i.e. $h_l(i_l) = i_l$) and hence use a matrix that relates the

RGB space of a device with the standard CIE XYZ space [23, 24, 74]. Further, such models assume spatial constancy in color properties and hence only one matrix is used to characterize the device. Thus, the XYZ response of a color generated by input $i = (i_r, i_g, i_b)$ is given by

$$\begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} = \begin{pmatrix} X_r & X_g & X_b \\ Y_r & Y_g & Y_b \\ Z_r & Z_g & Z_b \end{pmatrix} \begin{pmatrix} i_r \\ i_g \\ i_b \end{pmatrix} \quad (14)$$

where (X_l, Y_l, Z_l) denotes the CIE XYZ response of the maximum intensity input of channel l. Hence, L and C for input i is given by

$$(L_i, C_i) = \left(X_i + Y_i + Z_i, \left(\frac{X_i}{L_i}, \frac{Y_i}{L_i} \right) \right) \quad (15)$$

Now, we predict the color of the input i for such linear devices using our model. We use Equation 7 since black offset is non-existent for such devices. Replacing $h(i_l) = i_l$ to account for linearity, the color predicted by our model for input i is given by

$$(L_i, C_i) = \left(\sum_l i_l Q_l, \frac{\sum_l i_l Q_l c_l}{\sum_l i_l Q_l} \right) \quad (16)$$

Q_l and c_l, being the TTV and chrominance of the maximum intensity color for channel l defined by (X_l, Y_l, Z_l), are given by

$$Q_l = X_l + Y_l + Z_l; \quad c_l = \left(\frac{X_l}{Q_l}, \frac{Y_l}{Q_l} \right) \quad (17)$$

Replacing Equation 17 in Equation 16 which predicts the color of input i from our model gives us Equation 15 which predicts the color from the color transformation matrix.

This shows that our general model can accommodate existing models, which becomes a special case of our model. In essence, our model should be thought of as an abstract color model for imaging devices whose parameters should be evaluated using device-specific methods.

4. Deriving Color Variation Properties

Our model defines the color seen by a viewer from a display coordinate for a particular input. However, this color varies across the display. In this section, we show that this model can be used to explain all these color variations and artifacts that are visible on the display in practice.

In [49, 42] the color variation in multi-projector displays is classified in three different categories: the *intra-projector variation* (variation within a single projector), the *inter-projector variation* (variation across different projectors), and the *overlap variation* (variation in the overlap region). Here we explain these different kinds of variations using our model.

4.1. Intra-projector Variation

The intra-projector variation defines the color variation within a single projector. [49, 42] show that the spatial variation in intensity within a single projector (commonly referred to as the hot-spot effect) is much more significant than the spatial variation in chrominance. However, some regions within a single projector can have chrominance variations in the form of visible color blotches, as shown in Figure 4.

The intra-projector *intensity* variation is modeled by the spatially varying intensity function, Q_l, for each channel l, as illustrated in Figure 2.

The intra-projector *chrominance* variation (the color blotches in Figure 4) are due to following reasons. The first obvious reason is the small spatial variations in the chrominance function, c_l, as shown in Figure 2. However, sometimes these color blotches are visible even if there is no spatial variation of the chrominance function. We found that this is due to difference in the *shape* of the channel intensity functions Q_l across different channels (Figure 3). (Note that the *absolute* values of the intensity functions, Q_r, Q_g and Q_b, can be different from each other, and this will not lead to chrominance variation.) This can be proved as follows.

Let the maximum intensity for each channel of a projector be defined by M_l as

$$M_l = \max_{\forall s,t}(Q_l(s,t)). \quad (18)$$

Note that the parameter M_l is a constant and is not dependent on the spatial coordinates or input values. Hence, the *normalized intensity function* for each channel, $\bar{Q}_l$, is such that

$$Q_l(s,t) = M_l \bar{Q}_l(s,t).$$

Clearly, $\bar{Q}_l$ has the same shape of Q_l. From the definition of color operators in Section 2.1, the chrominance c^i for an input $i = (i_r, i_g, i_b)$, can be derived from Equation 7 as,

$$c^i(s,t) = \frac{\sum_{l \in \{r,g,b\}} c_l M_l \bar{Q}_l(s,t) h_l(i_l)}{\sum_{l \in \{r,g,b\}} M_l \bar{Q}_l(s,t) h_l(i_l)}. \quad (19)$$

Note that, even if c_l is assumed to be spatially constant, c^i can vary spatially due to the variation in $\bar{Q}_l$. But, if $\bar{Q}_l$ is same for all the three channels, not necessarily spatially constant, then the above equation reduces to,

$$c^i = \frac{\sum_{l \in \{r,g,b\}} c_l M_l h_l(i_l)}{\sum_{l \in \{r,g,b\}} M_l h_l(i_l)}. \quad (20)$$

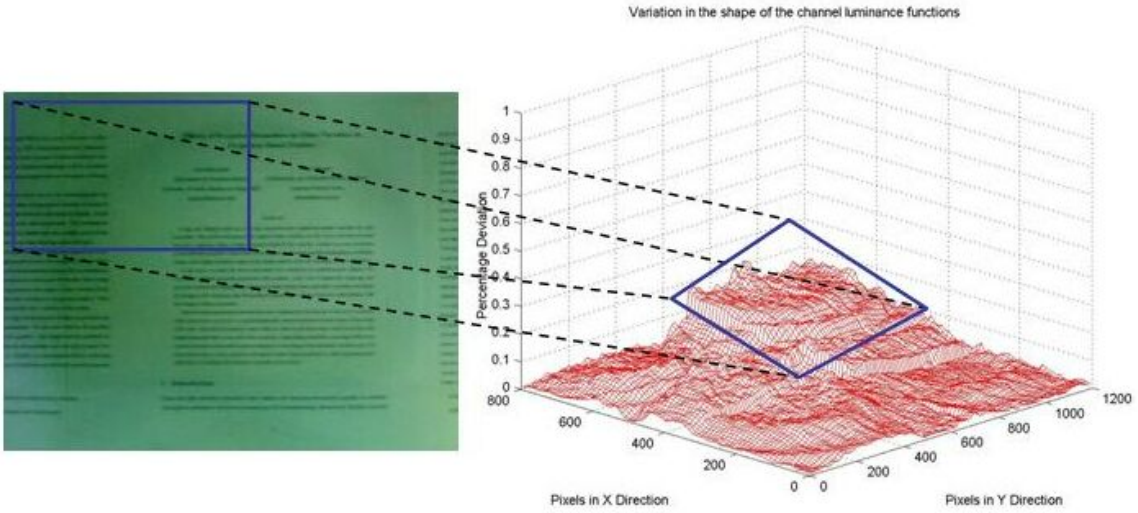

Figure 4: *Left: Color blotches on a single projector. Right: Corresponding percentage deviation in the shape of the blue and green channel intensity functions. Empirically, the maximum deviation in $\bar{Q}_l$ across different channels was* 20% *and the visible color blotches corresponded with the regions of large deviation.*

This equation is independent of spatial coordinates and hence c^i is a spatially constant function. This illustrates that the difference in shape of the intensity functions across different channels also contributes to intra-projector chrominance variations.

We verified the above result derived from our model, in projectors that had spatially constant channel chrominance but different $\bar{Q}_l$ across different channels (Figure 4).

4.2. Inter-projector Variation

Difference in the intra-projector color variations across different projectors is called the inter-projector color variation. Let us analyze the possible reasons for such variations.

The inter-projector *intensity* variation is due to two factors: the input transfer function h_l that is not same for all projectors; and the intensity function Q_l that is different for different projectors.

The inter-projector *chrominance* variation is due to differences in c_l (Equation 20) across different projectors. If two projectors differ in any of the parameters on the right hand side of the Equation 20, namely c_l, M_l and h_l, we will have inter-projector chrominance variation. It may seem from this equation that if c_l and h_l are identical, variation in absolute values of M_l cause the chrominance variation. However, note that since Equation 20 is a rational polynomial, even with variation in absolute values of M_l, the chrominance will not vary if there is no variation in the normalized values of M_l across different projectors. To show this, we divide the numerator and the denominator of Equation 20 by $\sum_{l\in\{r,g,b\}} M_l$ and normalize the values of M_l to

$$\bar{M}_l = \frac{M_l}{\sum_{l\in\{r,g,b\}} M_l}.$$

Note that these $\bar{M}_l$s describe the proportions of red, green and blue primaries used to generate colors in different projectors.

4.3. Overlap Region Variation

The overlap region variation defines the color difference between the overlap and non-overlap regions. The color in the overlap region is the superposition of the color from the individual projectors and is explicitly modeled by the addition operator in Equation 10. The cardinality of N_P ($|N_P|$) being different at every display coordinate causes this variation.

All the different kinds of color variations discussed in this section and the parameters causing them are summarized in Table 2. In addition, a quick reference to all parameters and their descriptions are also provided in Table 1.

5. Using the Model for Color Correction

We define a three step framework to design a color correction method.

Notation	Parameter Description
l	$l \in \{r, g, b\}$ and denotes the red, green and blue channels.
i_l	Input for channel l.
h_l	Spatially constant input transfer function of channel l, function of i_l.
c_l	Spatially varying chrominance of channel l, function of (s, t).
B	Spatially varying black offset, function of (s, t).
Q_l	Spatially varying maximum intensity function of channel l, function of (s, t).
M_l	$\max_{\forall(s,t)} Q_l(s, t)$.
$\bar{Q}_l$	Spatially varying *normalized* maximum intensity function of channel l, i.e. $\frac{Q_l(s,t)}{M_l}$.
$\bar{M}_l$	M_l normalized over all the three channels, i.e. $\frac{M_l}{\sum_{l \in \{r,g,b\}} M_l}$.
N_p	Number of projectors overlapping at any pixel, function of (u, v).

Table 1: *Legend of the notations and their descriptions of the model parameters.*

Type of Variation	TTV/Chrominance	Parameters Responsible
Intra Projector	TTV	1) Spatial Variation in Q_l
	Chrominance	1) Spatial variation in c_l 2) Difference in shapes of $\bar{Q}_l$ across different channels
Inter Projector	TTV	1) Difference in h_l across projectors 2) Difference in Q_l across projectors
	Chrominance	1) Difference in c_l across projectors 2) Difference in $\bar{M}_l$ across projectors
Overlap	TTV and Chrominance	1) Difference in $\lvert N_p \rvert$ at different spatial locations.

Table 2: *Summary of the parameters responsible for different kinds of color variation in a multi-projector display.*

1. *Reconstruction:* Our model defines the color that is actually seen by a viewer from the display. So, the first step of any color correction algorithm should be to reconstruct the function, $E(u, v, i)$, accurately. The accuracy and speed of the reconstruction depend on the sensors being used.
2. *Defining the Desired Function:* The reconstructed function shows perceivable color variation. Hence, the next step is to define the desired response of the display that the algorithm wants to achieve. We call this response as the *desired function*, $E'(u, v, i)$. E' can be formally defined by a set of conditions it should satisfy, called the *goals* of the color correction method.
3. *Modification:* Finally, a method should be devised that takes these goals as input and *modifies* a few parameters of E to generate the desired E'. The interactivity and the performance of the correction depend on the parameters that are chosen to be modified to achieve E'.

In this section, we analyze different existing color correction methods for multi-projector displays using the above framework. Finally, we demonstrate that a successful color correction algorithm can be designed and derived directly from our model by deriving an existing method [49, 47] from our model.

5.1. Common Goals

Several methods have been proposed to correct the color variation problem. We define a few common goals these algorithms implicitly or explicitly try to achieve.

Strict Color Uniformity: The goal here is to make the response of two different projectors look alike. The desired function, E', satisfies the property of *strict color uniformity* if the color (intensity and chrominance) of the light reaching the viewer from any two display coordinates, for the same input is the same. Hence, $\forall i,\ (u_1, v_1), (u_2, v_2)$,

$$|E'(u_1, v_1, i) - E'(u_2, v_2, i)| = 0. \qquad (21)$$

From Table 2, we note that the goal of strict color uniformity makes the parameters c_l, Q_l and h_l identical at every display coordinate.

Strict Photometric Uniformity: The goal here is to achieve identical *intensity* at every display pixel. In other words, E' satisfies the property of *strict photometric uniformity* if the intensity of light reaching the viewer from any two display coordinates for the same input is the same. That is $ttv(E')$ is spatially constant. Hence, $\forall i,\ (u_1, v_1), (u_2, v_2)$,

$$|ttv(E'(u_1, v_1, i)) - ttv(E'(u_2, v_2, i))| = 0 \qquad (22)$$

From Table 2, we note that the goal of strict photometric uniformity ignores the inter- and intra-projector *chrominance* variation and makes Q_l and h_l identical at every display coordinate.

Perceptual Color Uniformity: The goal here is to achieve a color variation that is imperceptible to the human eye. In other words, E' satisfies the property of *perceptual color uniformity* if the color (intensity and chrominance) of light reaching the viewer from any two display coordinates is within a threshold δ. Hence, $\forall i,\ (u_1, v_1), (u_2, v_2)$,

$$|E'(u_1, v_1, i) - E'(u_2, v_2, i)| \leq \delta \qquad (23)$$

From Table 2, we note that the goal of perceptual color uniformity makes the spatial variations in c_l and Q_l imperceptible in order to remove the intra-projector variation. Since h_l does not vary spatially within a single projector [49] it can be ignored for this purpose. However, the variation in all three parameters, c_l, Q_l and h_l, across different projectors must be imperceptible to address the inter-projector variation.

Perceptual Photometric Uniformity: The goal here is to achieve an intensity variation that is imperceptible to the human eye, without addressing the chrominance. In other words, E' satisfies the property of *perceptual photometric uniformity* if the intensity of light reaching the viewer from any two display coordinates is within a threshold δ. Hence, $\forall i,\ (u_1, v_1), (u_2, v_2)$,

$$|ttv(E'(u_1, v_1, i)) - ttv(E'(u_2, v_2, i))| \leq \delta \qquad (24)$$

From Table 2, we note that the goal of perceptual photometric uniformity makes the spatial variations in Q_l imperceptible in order to remove the intra-projector intensity variation. Also, as in the case of perceptual color uniformity, h_l can be ignored, but the variation in the parameters, Q_l and h_l, across different projectors must be imperceptible to address the inter-projector variation.

5.2. Explaining and Comparing Algorithms

In this section, we present an analysis of all the existing methods using the framework provided by our model.

5.2.1. Manual Manipulation of Controls

The most common method to correct the color variation in multi-projector displays is to manually change the projector control settings (like contrast, intensity, and white balance) and lessen the inter-projector mismatch. This method assumes that there is no intra-projector variation and overlap region.

This method does not make any effort to reconstruct the function $E(u, v, i)$. The goal of this method is to generate an E' that satisfies the constraints of *perceptual color uniformity*. In the modification step, changes in projector controls introduce changes in projector parameters h_l and M_l [49]. Since these projector parameters are not the only reasons for color variation (Section 4), especially across the spatial domain, the desired color correction is not successfully achieved.

5.2.2. Gamut Matching

The colors that a device can reproduce is represented by a three dimensional volume (one dimension for intensity and two dimensions for chrominance) called the color gamut. Gamut matching methods ([72, 73, 77]) try to address only the inter-projector variation. They assume that the intra-projector variation is negligible and there is no overlap region. The correction is achieved in three steps.

1. An accurate light measuring sensor, like a photometer, is used to find the precise color gamut of each projector at one spatial location. This is equivalent to reconstructing the function E, but only at very few display coordinates (u, v), usually at the center of each projector. The parameters c_l, Q_l and h_l are estimated only at those sampled spatial locations.
2. Since the color gamuts are different for different projectors, there are colors that can be produced by one projector and not by another. Hence, a *common color gamut*, defined by the intersection of the 3D color gamuts of all the projectors, is found. This is the gamut that can be reproduced by all projectors. Thus, the goal is to achieve a *strict color uniformity* by making the Q_l and c_l same for all projectors, but only at the display coordinates where E was sampled.
3. Finally, in the modification step, a linear transformation is derived for each projector that maps the colors from the original gamut to this common color gamut. This linear transformation is applied to the input i to achieve the correction.

In Table 2, we found that Q_l is responsible for intra-projector variation. Since Q_l is not sampled accurately in the spatial domain, the goal of strict color uniformity cannot be achieved successfully by this method.

Further, the algorithm to find a common color gamut for n projectors is a computational geometry problem of complexity $O(n^6)$ [4], which makes this method unscalable for displays made of large number of projectors. So, heuristics are often used to match the color gamuts which cannot guarantee an optimal solution. To avoid this, another method [45] makes the assumption that chrominance across a display made of *same model* projectors does not vary significantly. Thus, this method tries to match just the intensity response of the different projectors instead of the color gamuts. The problem of identifying the common intensity response is simpler than identifying the common gamut response. Formally, it tries to achieve *strict photometric uniformity*, but only at the display coordinates where E was sampled.

5.2.3. Using the Same Bulb for All Projectors

Some other methods [59] explore a novel engineering option of using the same bulb for all projectors. This method does not reconstruct E. The goal is strict color uniformity, under the assumption that the bulb is the sole cause of all variations and hence having a common bulb would achieve this goal. The modification of using a common bulb, in effect, makes the parameters M_l and c_l identical for all projectors. However, since Q_l is not changed, the color uniformity cannot be achieved successfully. Further, this method assumes that there is no overlap region.

5.2.4. Blending

Blending, or feathering techniques, attempt to blend the higher intensity in the overlap regions by weighing the contribution from each pixel of each projector in the overlap region by a spatially dependent factor. Thus, blending methods address only the overlap region color variation assuming negligible intra and inter projector variation.

This method again does not reconstuct E. Each overlap region defines a zone of transition from one projector to another, and this method aims at making this transition perceptually smooth. Thus, the formal goal is to generate *perceptual color uniformity*, but only in the overlap region. However, since E is not estimated accurately, blending leads to softening of the seams due to overlaps, rather than removing them completely.

Blending can be achieved in three different ways – software, aperture mask, and optical mask. When done in software [65], the blending function can be carefully designed to be a linear or cosine ramp. The second method uses physical masks [37] on the optical path near the projector's aperture boundaries. The third method modifies the optical and analog signals to the projectors [10] near the boundaries to create a virtual mask for blending. Figure 5 shows some results.

In the modification step of software blending, the input i is changed to i' to achieve the desired response. i is modified to i' such that

$$i' = \alpha_j(u, v, s_j, t_j) \times i,$$

where α_j is a function of the relative positions of projector and display coordinates such that $\sum_{j=1}^{n} \alpha_j = 1.0$ and $0.0 \leq \alpha_j \leq 1.0$. So, E' is given by

$$E'(u, v, i) = \oplus_{j \in N_P} T_j(s_j, t_j, \alpha_j \times i). \qquad (25)$$

In the modification step of aperture or optical mask blending, the correction is achieved by changing the function T_j at (u, v) itself and is given by

$$E'(u, v, i) = \oplus_{j \in N_P} \alpha_j \times T_j(s_j, t_j, i). \qquad (26)$$

Ideally, $\alpha_j \leq 1.0$ and $\sum_{j=1}^{n} \alpha_j = 1.0$, though this cannot be guaranteed in aperture or optical blending due to imprecise control of light.

5.2.5. Intensity Manipulation

A few recent methods [47, 49, 50, 44] tries to manipulate the spatial intensity response of every pixel of the display. These methods addresses the inter, intra, and overlap intensity (not chrominance) variations by first reconstructing the intensity of E in a rigorous manner using a digital camera.

The *intensity matching* method [47, 49] tries to match the intensity response of every pixel of the display. Formally, the goal of this method is *strict photometric uniformity*. To achieve the desired function, a constant maximum per channel intensity is assigned at every display coordinate. This desired E' is achieved in the modification step by manipulating the input i at every pixel (s, t) of every projector using a per-pixel attenuation factor. However, since the intensity at every pixel is matched to the pixel with the most limited dynamic range, a shortcoming of this method is severe compression in the dynamic range of the display, as illustrated in Figure 6.

To overcome this shortcoming, the *intensity smoothing* method [50, 44] smooths the spatial intensity response in a manner so that it is imperceptible to the human eye. The desired E' is seamless and also maintains a high average dynamic range. This desired E' is achieved by solving an optimization problem using dynamic programming techniques. Thus, formally, this

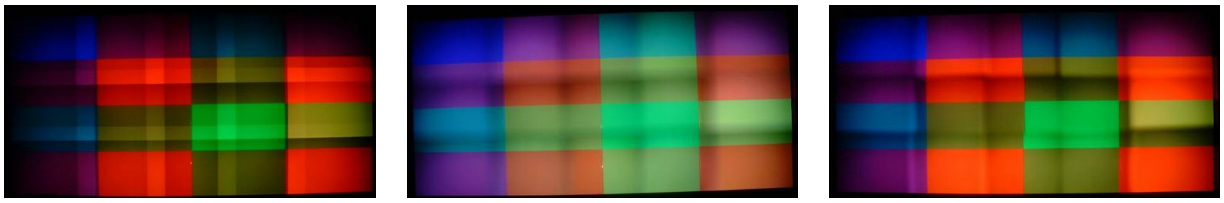

Figure 5: *Fifteen projector tiled display at Argonne National Laboratory: before blending (left), after software blending (middle), and after optical blending using physical mask (right).*

Figure 6: *Digital photograph of a* 5×3 *array of* 15 *projectors. Left: Before correction. Middle: After intensity matching. Right: After intensity smoothing.*

method aims for *perceptual photometric uniformity*, instead of a strict photometric uniformity. The modification step is still achieved by manipulating the input i at every pixel by an attenuation factor. The results of these methods are shown in Figure 6.

5.3. Deriving Algorithms from the Model

A convincing way to assure the effectiveness of the model lies in the success achieved by algorithms that are derived from this model In this section, we show how existing color correction methods, like the intensity matching methods [49, 47], can be derived from our model. This demonstrates that our model can be used directly to design solutions for correcting the color variation problem.

However, to use our model to design methods of color correction, we need to reconstruct the various model parameters. Reconstruction of the color variation of a multi-projector display requires reconstructing the color projected by individual projectors $T(s,t,i)$ (Equation 8). Reconstructing T for each projector involves reconstructing the transfer (h_l), the intensity (Q_l), and the chrominance(c_l) function for every channel and the black intensity (B) and chrominance (c_B).

The parameter h_l is spatially constant and a photometer can be used to measure this parameter at the center of every projector. To reconstruct the spatially varying components like Q_l, c_l, B and c_B, one can use a digital camera. The required geometric correspondences between projector, display, and camera coordinates are available from any standard geometric calibration method ([65, 64, 11, 31]). Readers are referred to [49, 48] for a more detailed discussion of this reconstruction process.

For a Lambertian screen, the parameters thus reconstructed are view-independent since $\Lambda(u,v,e) = 1$ for all view points. For non-Lambertian screens, the channel and black chrominance, c_l and c_B respectively, being independent of the BRDF/BTDF and h_l being a normalized function, can still be reconstructed using the above method. However, Q_l and B are view dependent and are given by $\Lambda(s,t,e)Q_l(s,t)$ and $\Lambda(s,t,e)B(s,t)$ respectively (Equation 9). In such cases, BRTF/BTDF information can be generated using different existing methods [2, 54, 13, 53, 17, 16, 26, 39, 80]. This information along with the geometric calibration information providing the position and orientation of the camera with respect to the screen [31] should be used to take into account the effect of the view dependent BRDF/BTDF to generate the view-independent Q_l and B.

Figure 2 shows the different reconstructed parameters of the model for a single projector. Figure 3 shows the reconstructed function for a 2×2 array of four projectors when the input $(1,1,0)$ is projected at every display coordinate as seen from a single fixed position..

To verify the validity of this model, we compare two responses: the first is called the *actual response* and is the response of a 'real' camera capturing a multi-projector display when it projects a particular input image I. The second is called the 'predicted response',

Figure 7: *A 2×2 array of four projectors. Left Column: Predicted response generated using our color variation model. Right Column: Actual response generated from a digital photograph.*

where the projected image for the input image I is predicted using the parameters of the multi-projector display reconstructed using the methods given above. Finally, the response of a 'hypothetical camera that captures this predicted image is generated. This is called the *predicted response*. If the 'hypothetical' camera is designed to have similar color properties as the 'real' camera, then the actual and the predicted response will be similar.

We note here that the framework of the same function, E, can be used to model an image capturing device, a camera, also and we use it to design our 'hypothetical' camera. However, to make this camera similar to the 'real' camera, we make some simplifying assumptions like spatially invariant channel intensity, chrominance and the transfer functions and no black offset. These lead to the difference between the actual and predicted responses in the results shown in Figure 7. Nevertheless, the *similarity* between the responses is the important feature to be noted. For example, the bottom left projector has more radial fall-off than other projectors and this can be noticed in both the actual and predicted responses.

Now that we know how to reconstruct the different model parameters, next we show the derivation of the intensity matching algorithm from our model. The intensity matching algorithm [47] makes four assumptions that simplify Equation 10 as follows.

1. *The screen is Lambertian.*
   So, we can use Equation 11 instead of Equation 10.
2. *The channel chrominance function is spatially constant and identical for all projectors.*
   With this assumption, the chrominance part of Equation 11 can be ignored and only the intensity given by $ttv(E(u, v, i)$ is considered.
3. *There is no black offset.*
   This means that $B(s, t) = 0$.
4. *The intensity response for each channel of each projector is linear and monotonic.*
   This means h_l for all channels of every projector is linear and monotonic. This, along with the fact that there is no black offset, indicates the $h(l)$ is identity function, i.e. $h_l(i_l) = i_l$ and hence same for all projectors.

Using all the above assumptions, $ttv(E(u, v, i)$ is de-

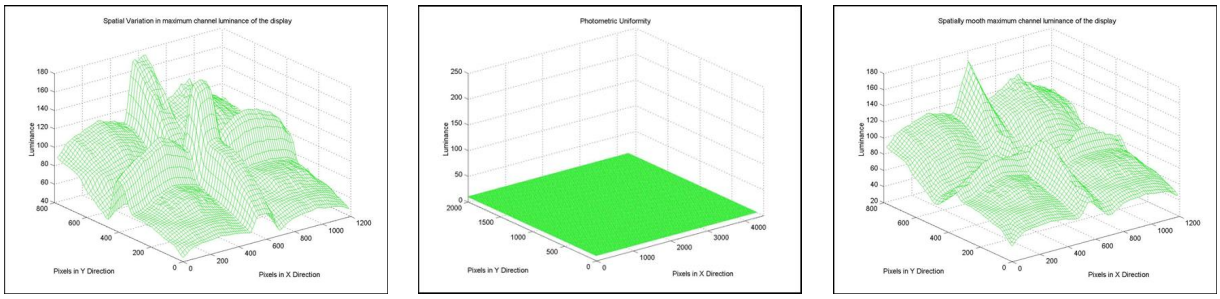

Figure 8: *The display intensity response surface 2×2 array of four projectors. Left: Before correction. Middle: After intensity matching. Right: After intensity smoothing.*

rived from Equation 10 as

$$ttv(E(u,v,i)) = \sum_{l\in\{r,g,b\}} \left(i_l \sum_{j\in N_P} Q_{l_j}(s_j,t_j) \right) \quad (27)$$

$$= \sum_{l\in\{r,g,b\}} (i_l \; Q_{d_l}(u,v)) \quad (28)$$

where $Q_{d_l}(u,v) = \sum_{j\in N_P} Q_{l_j}(s_j,t_j)$ is the spatial sum of intensity functions from all projectors for a channel and is called the *display intensity surface* in [49].

The algorithm consists of two steps: calibration step and the correction step. In the offline *calibration* step, the display intensity surface is captured using a digital camera. Since the goal of this method is to achieve strict photometric uniformity, the desired function, E' should have constant display intensity surface. This constant value is given by the $\min_{\forall(u,v)} Q_{d_l}(u,v)$. Thus, the intensity of the desired function can be derived from Equation 28 as,

$$ttv(E'(u,v,i)) = \sum_{l\in\{r,g,b\}} \left(i_l \left(\min_{\forall(u,v)} Q_{d_l}(u,v) \right) \right) \quad (29)$$

Note that the intensity smoothing method achieves a smooth display intensity surface instead of a constant display intensity surface by solving an optimization problem. The difference between the desired display intensity surface for these methods are illustrated in Figure 8.

The online *image correction* step achieves the above desired function by modifying the input i to i' as follows:

$$i'_l(u,v) = i_l(u,v) \times \frac{\min_{\forall(u,v)} Q_{d_l}(u,v)}{Q_{d_l}(u,v)} \quad (30)$$

Projectors can realize only the original function (Equation 28). Since we have changed the input, we have to substitute the value of new input given by Equation 30 in Equation 28 to find the actual response. We can see that it is same as the desired function given by Equation 29.

The scale factors to modify the given input in Equation 30 define the per-pixel attenuation map in [49]. To achieve an h_l that is an identity function, a color look-up-table is used to linearize the response of every projector. Thus, the intensity correction is achieved.

Note that, this method would work for non-Lambertian screen with no modification. When considering a non-Lambertian screen, the RHS of Equation 28 and 29 will be scaled by the BRDF/BTDF $\Lambda(u,v,e)$. And since the attenuation map is generated by dividing the RHS of these two equations, the BRDF/BTDF would cancel out generating the same correct attenuation map. However, this attenuation map will be view-dependent. Note that such insights are realized with the aid of our model.

This is just an instance of how a method can be derived from our model of color variation in multi-projector displays. We believe that better algorithms can be designed that can produce high quality color correction, using our model and the insights presented in the following Section 6.

6. Conclusion

Our model involves a five dimensional function (two dimensions for space and three for color). Thus, matching color gamuts at every display coordinate can be practically impossible. This is the reason we have only seen algorithms that deal with three of the five dimensions. Methods in Sections 5.2.1, 5.2.2 and 5.2.3, match the three dimensional color ignoring the spatial variation. On the other hand, methods in Sections 5.2.4 and 5.2.5 address intensity and two spatial dimensions, ignoring chrominance. However, we noted in Section 4 that a significant amount of the chrominance variation problem can be addressed by making

the shape of the intensity functions similar across different channels. This insight may help us to address considerable part of the chrominance variation problem, by staying within the three-dimensional domain and ignoring chrominance.

It is difficult to precisely and automatically control parameters like c_l, Q_l, h_l and M_l outside the projector manufacturing pipeline. Hence solutions which try to manipulate these parameters are not precise and are not scalable. On the other hand, methods that change the input i, are precise and scalable. Further such methods have the potential to be implemented using commodity graphics hardware and can be used for interactive rendering.

In conclusion, the color model presented in this paper provides a tool to study the color variation problem in multi projector displays. Further, this function can be used *directly* to design sophisticated algorithms that would yield truly seamless high quality displays. It is general and hence need not be restricted only to projectors. In fact, this can be used to model other display devices like CRTs, LCD panels, and image sensors like camera. In essence, this function provides an abstract framework to model all color properties of an image display/capture device. We believe that this model will be immensely useful in characterizing and designing color correction algorithms for small or large area displays.

Acknowledgements

We thank Kodak for providing us with high resolution test images. This work was supported in part by the U.S. Department of Energy under Contract W-31-109-ENG-38.

References

[1] Mercury: Realtime anti-aliasing 3d graphics sub-system. Technical report, http://www.quantum3D.com/products%20pages/mercury.html, [cited 2000].

[2] M. Ashikhmin, S. Premoze, and P. Shirley. A microfacet-based brdf generator. *Proceedings of SIGGRAPH*, pages 65–74, 2000.

[3] Rui Bastos. *Superposition Rendering : Increased Realism for Interactive Walkthroughs.* PhD thesis, University of North Carolina at Chapel Hill, Department of Computer Science, 1999.

[4] M. Bern and D. Eppstein. Optimized color gamuts for tiled displays. *ACM Computing Research Repository, cs.CG/0212007, 19th ACM Symposium on Computational Geometry*, 2003.

[5] R.S. Berns, M.E. Gorzynski, and R.J. Motta. Crt colorimetry, part ii: Metrology. *Color Research and Application*, 18(5):315–325, 1992.

[6] R.S. Berns, R.J. Motta, and M.E. Gorzynski. Crt colorimetry, part i: Theory and practice. *Color Research and Application*, 18(5):299–314, 1992.

[7] David H. Brainard. Calibration of a computer controlled color monitor. *Color Research and Application*, 14(1):23–34, 1989.

[8] Ian Buck, Greg Humphreys, and Pat Hanrahan. Tracking graphics state for networked rendering. *Proceedings of Eurographics/SIGGRAPH Workshop on Graphics Hardware*, 2000.

[9] Albert Cazes, Gordon Braudaway, Jim Christensen, Mike Cordes, Don DeCain, Alan Lien, Fred Mintzer, and Steve L. Wright. On the color calibration of liquid crystal displays. *SPIE Conference on Display Metrology*, pages 154–161, 1999.

[10] C. J. Chen and Mike Johnson. Fundamentals of scalable high resolution seamlessly tiled projection system. *Proceedings of SPIE Projection Displays VII*, 4294:67–74, 2001.

[11] Han Chen, Rahul Sukthankar, Grant Wallace, and Kai Li. Scalable alignment of large-format multi-projector displays using camera homography trees. *Proceedings of IEEE Visualization*, 2002.

[12] R.A. Chorley and J. Laylock. Human factor consideration for the interface between electro-optical display and the human visual system. In *Displays*, volume 4, 1981.

[13] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *Proceeding of SIGGRAPH*, pages 307–316, 1981.

[14] Carolina Cruz-Neira, Daniel J. Sandin, and Thomas A.Defanti. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *Proceedings of ACM Siggraph*, 1993.

[15] Scott Daly. The visual differences predictor : An algorithm for the assessment of image fidelity. *Digital Images and Human Vision, Editor: A.B.Watson, MIT Press*, pages 179–206, 1993.

[16] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real world surfaces. *IEEE Transactions on Graphics*, pages 1–34, January 1999.

[17] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. *Proceedings of SIGGRAPH*, pages 145–156, 2000.

[18] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. *Proceedings of ACM Siggraph*, pages 369–378, 1997.

[19] D.H.Brainard. Calibration of a computer controlled color monitor. *Color Research and Applications*, 14(1):23–34, 1989.

[20] P. Diefenbach and N. Badler. Pipeline rendering : Interactive refractions, reflections and shadows. In *Displays(Special Issue on Interactive Computer Graphics)*, 1994.

[21] D.L.MacAdam. Visual sensitivity to color differences in daylight. *Journal of Optical Society of America*, pages 247–274, 1942.

[22] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, Massachusetts, 1993.

[23] J.D. Foley, A. Van Dam, S.K. Feiner, and J.F.Hughes. *Computer Graphics Principles and Practice*. Addison Wesley, 1990.

[24] Edward J. Giorgianni and Thomas E. Madden. *Digital Color Management : Encoding Solutions*. Addison Wesley, 1998.

[25] E. Bruce Goldstein. *Sensation and Perception*. Wadsworth Publishing Company, 2001.

[26] M. Gsele and W. Heidrich et al. Building a photo studio for measurement purposes. *Proceedings of Vision, Modeling, and Visualization*, 2000.

[27] Paul E. Haeberli and Kurt Akeley. The accumulation buffer: Hardware support for high-quality rendering. *Computer Graphics (Proceedings of SIGGRAPH 90)*, 24(4):309–318, 1990.

[28] Mark Hereld. Local methods for measuring tiled display alignment. *Proceedings of IEEE International Workshop on Projector-Camera Systems*, 2003.

[29] Mark Hereld, Ivan Judson, and Rick Stevens. Introduction to building projection-based tiled display systems. *IEEE Computer Graphics and Applications*, 2000.

[30] Mark Hereld, Ivan R. Judson, Joseph Paris, and Rick L. Stevens. Developing tiled projection display systems. *Proceedings of the Fourth International Immersive Projection Technology Workshop*, 2000.

[31] Mark Hereld, Ivan R. Judson, and Rick Stevens. Dottytoto: A measurement engine for aligning multiprojector display systems. *Argonne National Laboratory preprint ANL/MCS-P958-0502*, 2002.

[32] G. Humphreys and P. Hanrahan. A distributed graphics system for large tiled displays. In *Proceedings of IEEE Visualization*, 1999.

[33] Greg Humphreys, Ian Buck, Matthew Eldridge, and Pat Hanrahan. Distributed rendering for scalable displays. *Proceedings of IEEE Supercomputing*, 2000.

[34] Greg Humphreys, Matthew Eldridge, Ian Buck, Gordon Stoll, Matthew Everett, and Pat Hanrahan. Wiregl: A scalable graphics system for clusters. *Proceedings of ACM SIGGRAPH*, 2001.

[35] Greg Humphreys, Mike Houston, Ren Ng, Randall Frank, Sean Ahem, Peter Kirchner, and James Klosowski. Chromium : A stream processing framework for interactive rendering on clusters. *ACM Transactions of Graphics*, 2002.

[36] Gregory Ward Larson. Overcoming gamut and dynamic range limitation in digital images. *Siggraph Course Notes*, 2001.

[37] Kai Li, Han Chen, Yuqun Chen, Douglas W. Clark, Perry Cook, Stefanos Damianakis, Georg Essl, Adam Finkelstein, Thomas Funkhouser, Allison Klein, Zhiyan Liu, Emil Praun, Rudrajit Samanta, Ben Shedd, Jaswinder Pal Singh, George Tzanetakis, and Jiannan Zheng. Early experiences and challenges in building and using a scalable display wall system. *IEEE Computer Graphics and Applications*, 20(4):671–680, 2000.

[38] M. J. Liaw and H.P.D. Shieh. Colorimetric calibration of crt monitors using modified bern's model. *Optik*, 104(1):15–20, 1996.

[39] Xinguo Liu, Yizhou Yu, and Heung-Yeung Shum. Synthesizing bidirectional texture functions for real-world surfaces. *Proceedings of SIGGRAPH*, pages 97 – 106, 2001.

[40] Charles J. Lloyd. Quantifying edge blend quality: Correlation with observed judgements. *Proceedings of Image Conference*, 2002.

[41] Jeffrey Lubin. A visual discrimination model for imaging system design and evaluation. *Vision Models for Target Detection and Recognition, Editor: E. Peli, World Scientific*, pages 245–283, 1995.

[42] Aditi Majumder. Properties of color variation across multi-projector displays. *Proceedings of SID Eurodisplay*, 2002.

[43] Aditi Majumder. *A Practical Framework to Achieve Perceptually Seamless Multi-Projector Displays*. PhD thesis, University of North Carolina at Chapel Hill, Department of Computer Science, 2003.

[44] Aditi Majumder. Contrast enhancement of multi-displays using human contrast sensitivity. *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[45] Aditi Majumder, Zue He, Herman Towles, and Greg Welch. Achieving color uniformity across multi-projector displays. *Proceedings of IEEE Visualization*, 2000.

[46] Aditi Majumder, David Jones, Matthew McCrory, Michael E Papke, and Rick Stevens. Using a camera to capture and correct spatial photometric variation in multi-projector displays. *Proceedings of IEEE International Workshop on Projector-Camera Systems*, 2003.

[47] Aditi Majumder and Rick Stevens. LAM: Luminance attenuation map for photometric uniformity in projection based displays. *Proceedings of ACM Virtual Reality and Software Technology*, 2002.

[48] Aditi Majumder and Rick Stevens. Identifying and optimizing the emineoptic function for color seamlessness in multi projector displays. Technical report, #260, Argonne National Laboratory, 2003.

[49] Aditi Majumder and Rick Stevens. Color nonuniformity in projection-based displays: Analysis and solutions. *IEEE Transactions on Visualization and Computer Graphics*, 10(2), March/April 2004.

[50] Aditi Majumder and Rick Stevens. Perceptual photometric seamlessness in tiled projection-based displays. *ACM Transactions on Graphics*, 24(1), January 2005.

[51] Aditi Majumder and Greg Welch. Computer graphics optique : Optical superposition of projected computer graphics. *Proceedings of Joint Eurographics Workshop on Virtual Environments and Immersive Projection Technology*, 2001.

[52] Thomas L. Martzall. Simultaneous raster and calligraphic crt projection system for flight simulation. In *SPIE Proceedings,Electroluminescent Materials, Devices, and Large-Screen Displays*, volume 1910, 01/31/1993 - 02/05/1993.

[53] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A data driven reflectance model. *ACM Transactions on Graphics (TOG)*, 22(3):759–769, July 2003.

[54] David K. McAllister. *A Generalized Surface Appearance Representation for Computer Graphics*. PhD thesis, University of North Carolina at Chapel Hill, Department of Computer Science, 2002.

[55] Steve Molnar, Michael Cox, David Ellsworth, and Henry Fuchs. A sorting classification of parallel rendering. *IEEE Computer Graphics and Algorithms*, pages 23 – 32, 1994.

[56] J. Montrym, D. Baum, D. Dignam, and C. Migdal. Infinitereality: A real-time graphics systems. *Proceedings of SIGGRAPH*, 1997.

[57] Liandan O'Callaghan, Nina Mishra, Adam Mayerson, Sudipto Guha, and Rajeev Motwani. Streaming-data algorithms for high-quality clustering. *IEEE International Conference on Data Engineering*, 2002.

[58] John Owens, William Dally, Ujval Kapasi, Scott Rixner, Peter Mattson, and Ben Mowery. Polygon rendering on stream architecture. *Proceedings of SIGGRAPH/Eurographics Workshop on Graphics Hardware*, pages 23 – 32, 2000.

[59] B. Pailthorpe, N. Bordes, W.P. Bleha, S. Reinsch, and J. Moreland. High-resolution display with uniform illumination. *Proceedings Asia Display IDW*, pages 1295–1298, 2001.

[60] Kenneth Perrine and Donald Jones. Parallel graphics and interactivity with scalable graphics engine. *IEEE Supercomputing*, 2001.

[61] Charles Poynton. *A Techical Introduction to Digital Video*. John Wiley and Sons, 1996.

[62] A. Raij, G. Gill, A. Majumder, H. Towles, and H. Fuchs. PixelFlex2: A Comprehensive, Automatic, Casually-Aligned Multi-Projector Display. *IEEE International Workshop on Projector-Camera Systems*, 2003.

[63] Mahesh Ramasubramanian, Sumanta N. Pattanaik, and Donald P. Greenberg. A perceptually based physical error metric for realistic image synthesis. *Proceedings of ACM SIGGRAPH*, pages 73–82, 1999.

[64] R. Raskar, M.S. Brown, R. Yang, W. Chen, H. Towles, B. Seales, and H. Fuchs. Multi projector displays using camera based registration. *Proceedings of IEEE Visualization*, 1999.

[65] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: A unified approach to image based modeling and spatially immersive display. In *Proceedings of ACM Siggraph*, pages 168–176, 1998.

[66] Ramesh Raskar. Immersive planar displays using roughly aligned projectors. In *Proceedings of IEEE Virtual Reality 2000*, 1999.

[67] Ramesh Raskar. Projector based three dimensional graphics. Technical Report TR02-046, University of North Carolina at Chapel Hill, 2001.

[68] Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM Transactions on Graphics*, 22(3), 2003.

[69] J. Rohlf and J. Helman. Iris performer: A high performance multiprocessing toolkit for real-time 3d graphics. *Preceedings of SIGGRAPH*, 1994.

[70] Rudro Samanta, Jiannan Zheng, Thomas Funkhouser, Kai Li, and Jaswinder Pal Singh. Load balancing for multi-projector rendering systems. In *SIGGRAPH/Eurographics Workshop on Graphics Hardware*, 1999.

[71] S.Molnar, J. Eyles, and J. Poulton. Pixelflow : High-speed rendering using image composition. In *Proceedings of SIGGPARH*, pages 231–240, 1992.

[72] Maureen C. Stone. Color balancing experimental projection displays. *9th IS&T/SID Color Imaging Conference*, 2001a.

[73] Maureen C. Stone. Color and brightness appearance issues in tiled displays. *IEEE Computer Graphics and Applications*, 2001b.

[74] Maureen C. Stone. *A Field Guide to Digital Color*. A.K. Peters, 2003.

[75] Russell L. De Valois and Karen K. De Valois. *Spatial Vision*. Oxford University Press, 1988.

[76] Russell L. De Valois and Karen K. De Valois. *Spatial Vision*. Oxford University Press, 1990.

[77] G. Wallace, H. Chen, and K. Li. Color gamut matching for tiled display walls. *Immersive Projection Technology Workshop*, 2003.

[78] Ruigang Yang, David Gotz, Justin Hensley, Herman Towles, and Michael S. Brown. Pixelflex: A reconfigurable multi-projector display system. *Proceedings of IEEE Visualization*, 2001.

[79] Ruigang Yang, Aditi Majumder, and Michael S. Brown. Camera based calibration techniques for seamless flexible multi-projector displays. *Applications of Computer Vision Workshop, Proceedings of European Conference in Computer Vision (ECCV)*, 2004.

[80] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. *Proceedings of SIGGRAPH*, pages 215–224, 1999.