Color Contoning for 3D Printing

VAHID BABAEI, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
KIRIL VIDIMČE, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
MICHAEL FOSHEY, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
ALEXANDRE KASPAR, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
PIOTR DIDYK, Saarland University, MMCI and MPI Informatik
WOJCIECH MATUSIK, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology

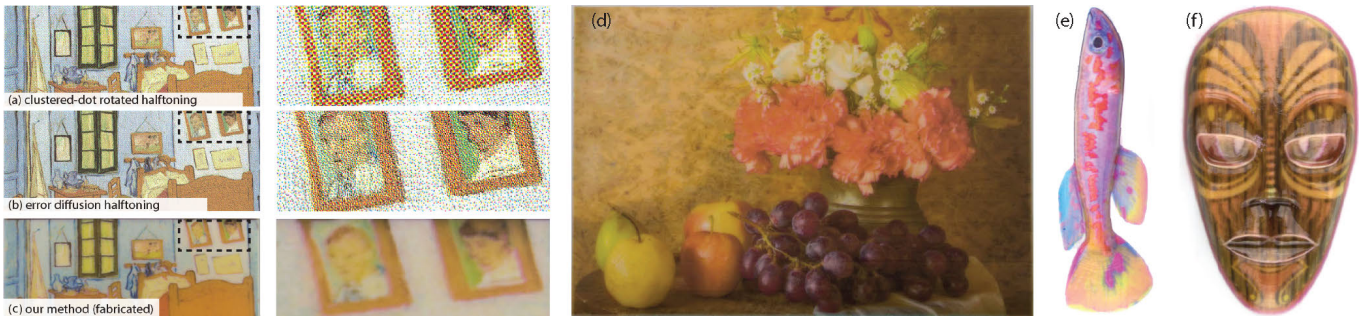


Fig. 1. We introduce a novel color reproduction method for 3D printing. In contrast to halftoning techniques that create colors by spatial combination of inks on the surface, our method creates colors by superposing inks inside the printed volume. This mitigates a variety of artifacts associated with halftoning. Parts (a) and (b) of the figure show the preview of an image generated using clustered-dot rotated halftoning and error-diffusion halftoning, respectively. Part (c) shows a photograph of the 3D printed image using our technique. Parts (d), (e) and (f) present three examples fabricated with our technique on a custom, multi-material 3D printer.

Appearance reproduction is an important aspect of 3D printing. Current color reproduction systems use halftoning methods that create colors through a spatial combination of different inks at the object's surface. This introduces a variety of artifacts to the object, especially when viewed from a closer distance. In this work, we propose an alternative color reproduction method for 3D printing. Inspired by the inherent ability of 3D printers to layer different materials on top of each other, 3D *color contoning* creates colors by combining inks with various thicknesses inside the object's volume. Since inks are inside the volume, our technique results in a uniform color surface with virtually invisible spatial patterns on the surface. For color prediction, we introduce a simple and highly accurate spectral model that relies on a weighted regression of spectral absorptions. We fully characterize the proposed framework by addressing a number of problems, such as material arrangement, calculation of ink concentration, and 3D dot gain. We use a custom 3D printer to fabricate and validate our results.

CCS Concepts: • **Computing methodologies** → **Reflectance modeling**; *Image processing*; *Perception*; • **Hardware** → **Printers**; *Displays and imagers*;

Additional Key Words and Phrases: Fabrication, 3D Printing, Color, Halftoning, Color reproduction

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2017 Copyright held by the owner/author(s). Publication rights licensed to Association for Computing Machinery.
0730-0301/2017/7-ART124 $15.00
https://doi.org/10.1145/3072959.3073605

ACM Reference format:
Vahid Babaei, Kiril Vidimče, Michael Foshey, Alexandre Kaspar, Piotr Didyk, and Wojciech Matusik. 2017. Color Contoning for 3D Printing. *ACM Trans. Graph.* 36, 4, Article 124 (July 2017), 15 pages.
https://doi.org/10.1145/3072959.3073605

1 INTRODUCTION

Additive manufacturing enables the design and reproduction of intricate geometries, and fabrication of novel multi-material composites which cannot be created using existing manufacturing techniques. A novel research area enabled by additive manufacturing is the fabrication of objects with a desired appearance. For many applications, reproducing the diffuse color is sufficient to guarantee an appearance reproduction without objectionable differences.

Color reproduction is a mature field in 2D printing. Almost all 2D printers are restricted to depositing discrete choices of materials at any given point on the surface. Therefore, halftoning techniques are employed to create an illusion of continuous-tone colors. Halftoning takes advantage of the human visual system that, due to its low-pass filtering property, integrates a set of discrete dispersed dots across a surface into a single color [Baqai et al. 2005]. Although halftoning is a clever solution for devices with binary tone reproduction, it has some drawbacks. Most importantly, the spatial halftone patterns can introduce visual artifacts which are particularly noticeable when the printed image is viewed from a close distance.

Similarly to 2D printers, 3D printers are also "discrete" devices and share the same restrictions. Instead of adapting 2D color reproduction methods to 3D printing [Brunton et al. 2015; Reiner et al.

2014], we propose to utilize the inherent ability of 3D printers to layer different materials on top of each other. We propose a "contoning" technique which prints layers of color inks around a white, diffuse material. In contrast to halftoning, our technique synthesizes colors by combining the inks within the volume of the object. Hence, regardless of the viewing distance, the decomposition of the color into multiple base materials remains invisible to the eye of the observer. Contoning mimics much better traditional materials which are often built from layers, such as wood, skin, marble, etc.

We use a custom, inkjet 3D printer with UV curable inks to explore different problems and trade-offs in the design of our color reproduction workflow, such as material design, and the layering order. We show that, compared to the halftoning algorithms, our method is free of spatial artifacts and has a much more scalable color prediction. Furthermore, our technique can create plausible color prints at very low resolutions. On the other hand, contoning requires a larger number of color layers than halftoning. Therefore, it is more prone to dot gain. We propose a simple and effective method to counteract this effect.

The main contributions of this work are:

- A complete color 3D printing framework for printing halftone-less textures suitable for 2.5D and 3D printers,
- A color prediction model for 3D printing,
- A simple and elegant pipeline for color reproduction of textured 3D objects,
- Fabrication of compelling prints at high and low printer resolution and comparison with halftoning.

2 RELATED WORK

Appearance reproduction for 3D printing is an emerging research area. Here, we review the research areas closely related to this paper. Readers are referred to [Hullin et al. 2013] for an extensive review of physical appearance reproduction literature.

Color matching. Color matching is a mature field, and the existing methods are capable of high accuracy reproduction. The equations introduced by Allen [1966; 1974] are the basis for many commercial systems for color matching of paints, plastics, and textiles. These equations benefit from the linear relationship between Kubelka-Munk constants and colorant concentrations [Kubelka and Munk 1931]. We use the same principles to predict the color of a stack of 3D printed inks.

Halftoning. Halftoning creates an illusion of a continuous-tone image by printing ink microdots with certain areas and structures [Baqai et al. 2005]. Two fundamental halftoning techniques are rotated clustered-dot and error diffusion. Clustered-dot halftoning [Sharma and Bala 2002] is designed such that individual printer microdots are grouped into clusters. It is used by printing technologies that have difficulties in rendering single dots. Error-diffusion halftoning [Ostromoukhov 2001] works within a local neighborhood and produces halftone prints with less dot-visibility than clustered-dot halftones. For creating color halftones, usually three or four halftone layers are superposed on top of each other.

There are several difficulties associated with halftoning methods. Clustered-dot halftones exhibit moiré [Amidror 2009] due to the interference between periodic structures of different layers. Error diffusion can suffer from divergent artifacts from worm-like structures to patterns with regular structures. These problems intensify with an increased number of superposed inks and limit the application of halftoning in high-end 2.5D or 3D printing. A further problem is the color predictability of halftones. It scales poorly with the number of inks and/or number of halftone layers [Babaei and Hersch 2016].

Continuous color printing. Collotype, a century-old technique, is a continuous-tone photomechanical process that produces photographic quality prints [Defibaugh 1997]. Dye sublimation, mostly used in printing textiles, is another printing process that transfers the colorants continuously from the carrier into the substrate using heat [Durbeck 2012]. Berns [1993] introduces a color prediction model for such printers. Our current effort is in the same vein except that here we focus on producing continuous-tone colors on 3D objects.

Appearance for 3D objects. There are several commercial color 3D printers on the market including several powder-binder systems from 3D Systems, photopolymer, phase-change printers from Stratasys and paper lamination printers from Mcor. Traditionally these printers have not provided a formal, color-managed pipeline and not much is known about their color reproduction strategy.

Recently Brunton et al. [2015] proposed a 3D halftoning algorithm for creating 3D color prints. Their error-diffusion method extends the capabilities of 2D halftoning to 3D objects and copes well with challenges of 3D halftoning. To obtain a good color reproduction, they rely on a fully data-driven color prediction scheme which requires per layer calibration that requires extensive sample measurements. Although their technique provides faithful color reproduction and reproduces well fine-detail structures, it suffers from inherent limitations of halftoning techniques. In our work, we follow the same goal, but we advocate for a fundamentally different strategy. We generate colors through a combination of our materials inside the print and not on its surface. This allows us to mitigate many halftoning artifacts at the cost of losing details in regions of sharp color discontinuities. We show that this effect can be compensated for using our pre-processing step.

Dong et al. [2010] and Hašan et al. [2010] fabricate slabs with predefined subsurface scattering. In this work, we ignore the translucency and focus on the accurate reproduction of diffuse, spectral color. Papas et al. [2013] design a framework for reproduction of homogenous materials with a given color and translucency.

An alternative, two-step approach is to print the texture separately, and then transfer it to the object. Hydrographic printing [Zhang et al. 2015] and thermoforming [Schüller et al. 2016; Zhang et al. 2016] are two recent examples.

3 OVERVIEW

We propose a technique for 3D printing color objects using a layering approach. The core of our method is a color reproduction workflow that for each color in an input color space, e.g., in sRGB, computes its equivalent printer control values. In our representation, the control values are *layer-layouts*: the number of layers of each printing ink in a printed stack. The steps for this task are shown

in Figure 2. First, we design a color prediction model that, for a given layer-layout, predicts the resulting reflectance spectrum in the visible range (Section 4). The color prediction model uses measured spectral reflectances of a set of prints made of primary inks and their sparse combination. We use the color prediction model to compute the concentrations of the primary inks used for printing (Section 5.2). Since our printer is limited to 5 materials, we choose to work with the cyan, magenta, yellow, black and white inks. Next step is color gamut mapping where we map the input colors to the color gamut of our printer (Appendix D). The gamut of the printer is the range of all achievable colors by the printer and is predicted using the color prediction model. Finally, we perform *layer separation* (equivalent to color separation in halftoning terminology) to find a layer-layout that best estimates a given gamut mapped color (Section 5.3). As our gamut is a set of discrete colors, the layer separation is a nearest neighbor search in the color space with an appropriate color difference formula that plays the role of the distance function.

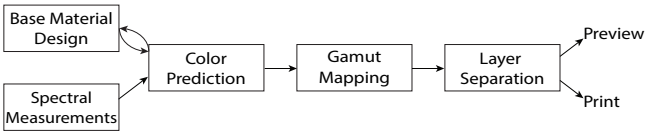


Fig. 2. Our color reproduction workflow. It takes an input color to its equivalent layer-layout.

For a fixed set of inks, the input-color to layer-layout workflow is run once and saved in a table. It can then immediately be used to generate flat prints. In Section 6, we show how these tables are used to generate color 3D objects. After evaluating our method in Section 7, we discuss the dot gain and propose a simple method to estimate it and compensate for it (Section 8). Finally, we present a comparison of halftoning and our method (Section 9).

4 COLOR PREDICTION MODEL

We propose a color prediction model that computes the resulting spectral reflectance (and consequently the perceived color) of a 3D printed stack as a function of its layer-layout. This model is the main component of our color reproduction workflow (Figure 2). Our prediction model uses a small set of measured prints to perform a linear regression in the *intrinsic absorption* space. The result of regression is a matrix that maps the layer-layout of any stack to its intrinsic absorption. We show that going from spectral reflectance to intrinsic absorption and vice versa is possible via a few closed-form formulas. In this section, we explain how to construct this model.

A common practice in color modeling is to find a space where there is a linear relationship between the "concentrations" of primary materials and the resulting color of their mixture [Berns 1997]. This explains the widespread use of the Kubelka-Munk (KM) equations [Kubelka and Munk 1931] in computational color science. The KM scattering and absorption coefficients scale linearly with the concentrations of coloring agents [Allen 1966, 1974]. The KM equations are the analytical solutions to the *radiative transfer equation* [Chandrasekhar 1960], subject to some approximations [Mudgett and Richards 1971].

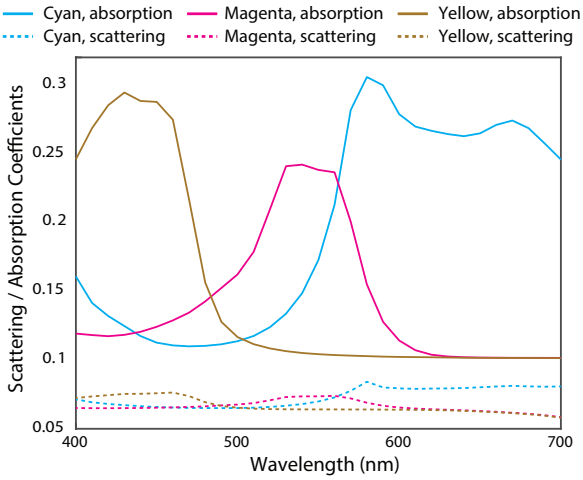


Fig. 3. The Kubelka-Munk absorption and scattering coefficients of our cyan, magenta and yellow inks at 0.5% concentration.

Our material formulation avoids color inks with significant light scattering. This assumption can be verified by computing the KM scattering and absorption coefficients of our inks for all visible wavelengths. We use the *black-white* method for calculating these coefficients [Zhao and Berns 2009]. As we see in Figure 3, the absorption is the more significant event occurring within our inks. Although, for some wavelengths, scattering reaches about 30% of absorption, this occurs only for low-absorption and low-scattering regions that do not contribute significantly to color perception. The absorption assumption also is supported by the high accuracy of our prediction model (Section 7). Consequently, we can use a simplified version of the KM theory (equivalent to Beer's law) which is based on absorption coefficients and build the regression using absorptions only.

In the absence of scattering, the absorption of an ink layer $\psi(\lambda)$ printed on a diffusing substrate with reflectance spectrum $R_g(\lambda)$ has the following relationship with the reflectance spectrum $R(\lambda)$ of stack of ink and substrate [Berns 1993, 1997]:

$$R(\lambda) = R_g(\lambda)\exp(-2\psi(\lambda)). \tag{1}$$

All above quantities are functions of the wavelength λ. As our materials do not show fluorescence, all wavelengths can be treated independently and the computations are done wavelength-by-wavelength. The KM theory assumes no discontinuity in the refractive index (neglecting the obvious air-ink interface). Therefore, we first apply a correction on the measured spectral reflectance to obtain the *intrinsic* reflectance. The intrinsic reflectance is the reflectance of a surface if there were no air-surface interface (Appendix A). We can then calculate the *intrinsic* absorption coefficient $\psi_s(\lambda)$ of a print using Equation 1 as:

$$\psi_s(\lambda) = -\frac{1}{2}\ln\left(\frac{\rho(\lambda)}{\rho_g(\lambda)}\right), \tag{2}$$

where $\rho(\lambda)$ and $\rho_g(\lambda)$ are the intrinsic reflectances of the sample $R(\lambda)$ and the substrate $R_g(\lambda)$, respectively.

According to the classic KM equations, the spectral absorption of every printed slab can be computed through convex combination of the spectral absorptions of its components. This gives large errors because of inaccuracies in KM theory assumptions, slight scattering of our materials, and the stray specular light captured during measurement. A compromise is to build a multilinear regression in the absorption space based on a small training dataset [Babaei et al. 2012]. The following relationship is built by concatenating the intrinsic absorption coefficients $\psi_s(\lambda)$ of the prints belonging to the training dataset in matrix $\Psi_\mathbf{s}$ and their corresponding layer-layout in matrix $\mathbf{C}$.

$$\Psi_\mathbf{s} = \mathbf{TC}. \tag{3}$$

Sampling the visible spectrum at n wavelength, using q prints in the training set, and having N inks, $\Psi_\mathbf{s}$, $\mathbf{T}$, and $\mathbf{C}$ are $n \times q$, $n \times N$ and $N \times q$ matrices, respectively. Matrix $\mathbf{T}$ maps any print with known layer-layout (thicknesses of different inks in matrix $\mathbf{C}$) to its intrinsic absorption coefficient $\psi_s(\lambda)$. From intrinsic absorption coefficient, we can use Equation 2 to obtain intrinsic reflectance $\rho(\lambda)$. Finally, we can add back the effect of discontinuity in refractive index to the intrinsic reflectance to obtain a predicted reflectance that is comparable to measured reflectances (Appendix A). Conceptually, matrix $\mathbf{T}$ contains the spectral absorptance bases for all inks at all wavelengths. Its size is therefore the number of wavelengths times the number of materials ($n \times q$). The training set is a set of spectral measurements of prints in which thicknesses of different inks are uniformly sampled from 0 to L layers, with a fixed interval, where L is the maximum thickness.

One limitation of this approach is that matrix $\mathbf{T}$ is identical for all test prints with different layer-layouts. To address this limitation, we use a weighted regression where a distinct parameter matrix $\mathbf{T_i}$ is tailored for the i-th test print. In the regression, we give higher weights to training samples with more similarity to a given test print. As a similarity metric, we use the inverse of the distance between the ink layer-layout of the i-th test print $\mathbf{c}_i$ and that of the j-th train print $\mathbf{c}_j$.

$$s_{ij} = \frac{1}{\left\|\mathbf{c}_i - \mathbf{c}_j\right\|_2}. \tag{4}$$

The division by zero does not happen as we do not allow $\mathbf{c}_i$ and $\mathbf{c}_j$ to be equal. In that case, the test print is in the training set and there is no need for predicting it. For p test prints to be predicted using q training prints, we compute a $p \times q$ similarity matrix $\mathbf{S}$ with entries s_{ij}:

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & s_{13} & \dots & s_{1q} \\ s_{21} & s_{22} & s_{23} & \dots & s_{2q} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & s_{p3} & \dots & s_{pq} \end{bmatrix}. \tag{5}$$

For the i-th test print, we define matrix $\mathbf{W_i}$ as a $q \times q$ diagonal matrix whose diagonal elements are the i-th row of $\mathbf{S}$.

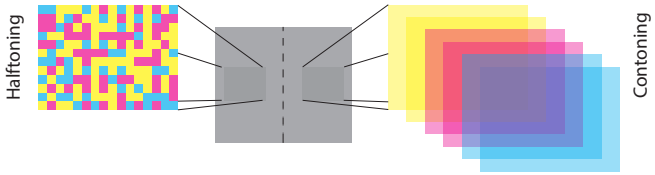


Fig. 4. Halftoning synthesizes colors by surface mixing of inks; increasing the amount of an ink is achieved by increasing its relative area. Contoning synthesizes colors by volumetric mixing of inks; increasing the amount of an ink is achieved by increasing its thickness.

$$\mathbf{W_i} = \begin{bmatrix} s_{i1} & 0 & \dots & 0 \\ 0 & s_{i2} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & \dots & 0 & s_{iq} \end{bmatrix}. \tag{6}$$

We can then rewrite Equation 3 as a weighted multilinear regression for each test print separately:

$$\Psi_\mathbf{s}\mathbf{W_i} = \mathbf{T_i}\mathbf{C}\mathbf{W_i}. \tag{7}$$

In order to obtain the parameter matrix $\mathbf{T_i}$, we need to perform the following optimization:

$$\underset{\mathbf{T_i}}{\operatorname{argmin}} \quad \|\Psi_s\mathbf{W_i} - \mathbf{T_i}\mathbf{C}\mathbf{W_i}\|_F, \tag{8}$$

where $\|.\|_F$ denotes the Frobenius norm. We need to solve this equation to predict the spectrum of every test print. The least square solution to this equation is the pseudo-inverse solution that can be computed as:

$$\mathbf{T_i} = \Psi_\mathbf{s}\mathbf{W_i}^2\mathbf{C}^\mathbf{t}(\mathbf{C}\mathbf{W_i}^2\mathbf{C}^t)^{-1} \tag{9}$$

Matrix $\mathbf{T_i}$ is then used, exactly like $\mathbf{T}$ as explained above, to calculate the intrinsic absorption of its corresponding 3D printed slab. We then again compute the intrinsic reflectance from intrinsic absorption and, finally convert the intrinsic reflectance to spectral reflectance. We further discuss the performance of our proposed color prediction model when put in practice in Section 7.

5 COLOR CONTONING

Inspired by additive layered manufacturing, we propose to synthesize colors by layering inks on top of each other (a brief study of ink layering is presented in Appendix B). Unlike halftoning that creates the perception of continuous colors by changing the area of dots, contoning superposes available inks with proper thickness at each printing location. Figure 4 demonstrates the schematic difference between these two methods. In this section, we present an algorithm for computing the ink concentrations. Then we show that layers can have a strict order of inks. This reduces the complexity of workflow significantly. Finally, we show how to invert the color prediction model in order to generate layer-layouts for any input color.

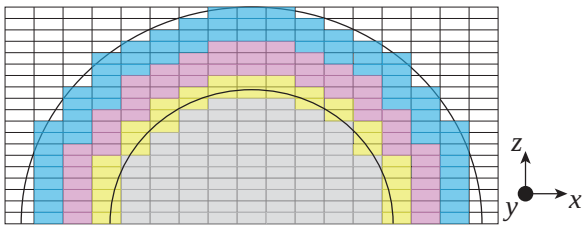


Fig. 5. Our printer grid. The layer thickness is equal in X and Y (XY-layer) but is different in Z direction (Z-layer).

5.1 Material Design

Like any other subtractive color mixing, color 3D printing needs the triple of cyan (C), magenta (M) and yellow (Y) to produce a reasonable color gamut (refer to [Yule 1967] for an excellent explanation behind this choice). In addition, a white (W) reflecting "background" is required, comparable to the role of the paper in 2D printing. We also add a black (K) ink to the palette of our 3D printer to improve black tones. Here, we discuss how we compute the optimal concentrations of these primary inks.

In our method, we generate colors by arranging color layers on top of a white background. The color layers are formed using low-concentration inks to avoid abrupt color changes and thus color quantization. The trade off is that with lower ink concentrations, more layers are needed to produce the same color which results in larger minimum feature size and increased optical and physical dot gain (Section 8). Minimum feature size is the smallest print feature whose color can be faithfully reproduced. The feature size corresponds to different number of layers in Z (Z-layers), or in X or Y (XY-layers), as our grid is not uniform (Figure 5). In this work, our strategy is to set a strict threshold on color quantization error and allow for a maximum number of layers that results in a reasonable feature size.

To calculate the ink concentrations, we begin with formulating the white ink. We carefully selected the highest white pigment concentration that was feasible to print on our system in order to make the background slab as opaque as possible with the fewest printed layers. We experimentally found that 5.0% white ink is reliable enough for printing. Higher concentration inks are generally less stable since the white pigments (typically made of TiO_2) are prone to settling and can easily clog the printhead. After measuring the spectral reflectance of different number of white layers on a black background, we observe that at 20 Z-layers of 5.0% white, the change in the white ink reflectance becomes negligible.

For colored inks (CMYK), we seek a set of ink concentrations that in the CIELAB color space [Morovic and Luo 2001] generate a color-quantization error less than a predefined threshold. Fortunately, we find that in our color prediction model, concentrations and thicknesses correlate well enough. Simply put, a color generated by doubling the concentration is reasonably close to a color printed with half concentration using two layers. This exempts us from making new inks for each iteration when searching for optimized concentrations. Therefore, we print only one set of ink concentrations, calibrate the model, and computationally change the concentration by changing the thicknesses in the model. We start by calibrating a color prediction model (see Section 4) with 1% pigment content for CMYK. Since the quantization is larger in the thicker XY-layers compared to Z-layers, the model is calibrated with training prints layered in XY.

After calibrating the model, we build the gamut of the printer and perform the gamut mapping (Appendix D) and layer separation (Section 5.3) for *all* sRGB colors. The color-quantization error is defined as the color error difference between a layer-separated color and its nearest neighbor in all gamut-mapped sRGB colors. As a quantization-error threshold, we choose the quantile 0.99999 of the layer separation error for all gamut-mapped sRGB colors. We perform a grid search by changing the initial concentrations in 0.05% steps until we satisfy the considered accuracy. Setting the quantization error to 3.0 CIEDE2000 (ΔE_{00}) units [Sharma et al. 2005], we get $c = 0.25\%$, $m = 0.25\%$, $y = 0.25\%$ and $k = 0.15\%$. After printing with these concentrations, we verify that the quantization error is 3.33 ΔE_{00} for XY-layering and 1.57 for Z-layering.

5.2 Ink Order

When permuting N inks within 1 to L layers, the number of possible layer-layouts Υ grows according to the following formula (Appendix C):

$$\Upsilon = \frac{N^{L+1} - 1}{N - 1} - 1. \tag{10}$$

Performing any computation for all possible layer-layouts would be impractical given the very large number of configurations. For a realistic scenario with maximum 40 layers, this number is almost $1.6{\times}10^{24}$. Given the complexity that free permutation of layers brings, allowing for it makes sense primarily if it produces a larger color gamut. Since our material formulation exhibits minimal scattering (Figure 3), we conjecture that the *volume of the color gamut* is mostly a function of the thicknesses of different inks in the layers and does not change considerably with changing their specific order. Therefore, we opt for a *stratified* layering scheme: identical inks are grouped together and are sorted in a strict order. This results in a significant reduction in the number of possible layer-layouts (Appendix C):

$$\Upsilon = \binom{L+N}{N} - 1. \tag{11}$$

For example, stratified layering of 4 inks allowing a maximum of 40 layers gives only 135,750 possible configurations.

Ideally, to compare the two layering modes (free-permutations and stratified), we should sample both color gamuts and compute their volumes. However, our color prediction model would need a very large calibration set in order to predict stacks with arbitrary order. Instead, we verified our assumption with a simple experiment where we printed two inks (magenta and yellow) and allowed 7 layers at most. This choice kept the number of prints and measurements manageable. We printed and measured all possible permutations of these two inks on a white background resulting in 254 colors, out of which 35 corresponded to the stratified layering category. Figure 6 shows the colors of these samples in a hue-angle, chroma diagram (Appendix B). Observe that colors with the same amount of inks but different orders form small clusters in this figure.

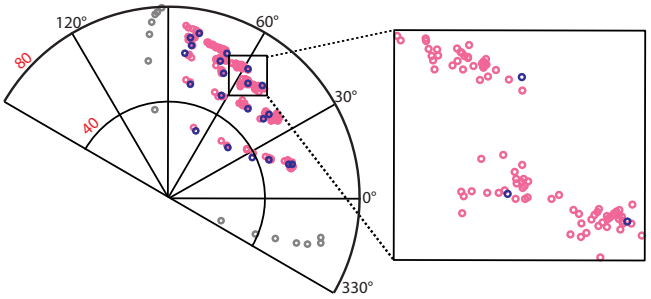


Fig. 6. The measured colors of a set of printed slabs in the hue-angle, chroma diagram. Slabs have up to 7 layers of magenta and yellow inks, where they freely permute. Among them, prints corresponding to stratified layering are shown with blue circles (the gray circles represent colors of stacks with either only magenta or only yellow inks from 1 to 7 layers, therefore they have one possible permutation).

Since lightness is absent in the hue-chroma representation, we also measured the color differences between all pairs within each individual cluster. The color difference formula accounts for luminance and chrominance. The average of the maximum color differences of members of each cluster is only 5.8 ΔE_{00}. Therefore, we observe that while the color of an ink stack is a function of its order, the color change due to the inks' order is not significant enough to make a change in the gamut volume. This is a key observation that leads us to adopt an ink grouping strategy.

5.3 Layer Separation

Now we show how to find a layer-layout that best represents an input colors. Before that, we must ensure that these input colors are reproducible by the printer and its available materials. For this reason, in the gamut mapping stage, we map the input colors into the color gamut of the printer (Appendix D).

In halftone color reproduction, the *color separation* step deduces the amounts of inks that are needed to print a target color C_t. This is usually formulated as the following [Pjanic and Hersch 2015]

$$\underset{c,m,y,k}{\operatorname{argmin}} \quad \Delta(C(c,m,y,k), C_t) \tag{12}$$

where $C(c,m,y,k)$ is a color computed using a prediction model and the Δ function is a color difference formula. Color separation is the most computationally expensive part of a color reproduction workflow as most color difference formulas are highly non-differentiable [Kauvar et al. 2015].

In contone color reproduction, we need to solve the similar inverse problem of going from a color to a layer-layout. We call it *layer separation*. Color contoning allows for a very efficient layer separation. The contone color gamut is made of a set of discrete 3-dimensional CIELAB points corresponding to distinct layer-layouts. The number and dimension of this set of points lend themselves very well to the nearest neighbor search algorithms. Using a KD-tree data structure, we can find the nearest neighbor in the color gamut set to a query point very quickly. Then, we find the layer-layout corresponding to that color. The Euclidean distance offered by KD-tree nearest neighbor search is equivalent to the classic ΔE_{76} color

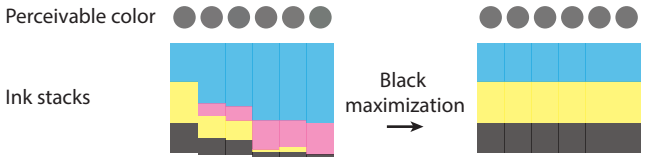


Fig. 7. Completely different ink stacks can generate almost identical colors. Our black maximization strategy chooses the stack with largest amount of black ink. Therefore, we produce a smooth transition between layer-layouts of neighboring printer pixels with similar colors.

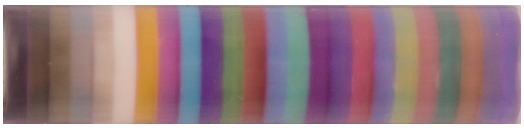


Fig. 8. A top-down photograph of a semi-cylinder printed with color separation tables from Z-layered calibration set.

difference formula [Wyszecki and Stiles 1982]. This formula gives reasonable results but is known to be perceptually non-uniform in some regions of the color space. In order to solve this challenge, we perform a rapid K nearest neighbor search for a relatively large K. Having obtained the K nearest colors for the target color in a ΔE_{76} sense, we refine them with a more advanced color difference formula, i.e. ΔE_{00} [Sharma et al. 2005].

The refinement stage gives us an opportunity to optimize the layer-layouts. Many target colors have layer-layouts whose colors are within a very close distance. These colors are usually a result of the trade-off between black ink on one hand and the three chromatic CMY inks on the other hand. We can reduce the amount of CMY inks in the stack and replace it with black without any noticeable change in the color. In our work, inspired by black maximizing color separation strategies [Sharma and Bala 2002], we select the layouts with the thickest black layer during layer separation. Apart from classic benefits , this method has some advantages specific to 3D color contoning. It enforces inter-layer smoothing by preventing sudden changes in the layer-layout of neighboring pixels with close colors. Figure 7 demonstrates this property.

6 3D CONTONING

Generating contone colors on 3D surfaces is straightforward. We use a lookup table, precomputed with our color reproduction workflow, to simplify and accelerate the color assignment process. The lookup table exhaustively generates a mapping from all 24-bit sRGB color space to layer stack.

During our experiments, we observed that it is sufficient to use the color prediction model calibrated with only Z-layered patches. This relieves us from calibrating the model in other directions and therefore reduces the required number of printed patterns significantly. To verify this, we printed a semi-cylinder using a layer-separation table resulting from the color prediction model calibrated with Z-layers only. As we see in Figure 8, this generates consistent colors over the curvature.

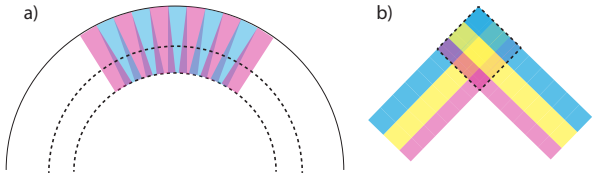


Fig. 9. Ink assignment conflict. Areas shown within dashed lines are in conflict. a) On a convex surface, if the color of the texture changes, the ink stacks of different colors collide inside the color shell. b) On geometric edges, even if there is a constant color, collisions are inevitable.

Given an input 3D model with its texture and surface parameterization, one naive approach is to voxelize it, store color and normal information on the surface voxels, and then follow the normals to assign the corresponding color stack to the internal voxels. While careful sampling on the surface is required to avoid missing internal voxels, two main problems arise: (1) convex regions lead to neighboring color stacks colliding into each others and (2) a given color stack height may not be available, as illustrated in Figure 9.

Our approach is instead to do per-voxel color assignment using distance queries to the surface. For every voxel, we query the closest surface point, extract the corresponding color stack, and sample it using the surface distance to get our material assignment. This strategy avoids sampling issues of ray marching from the surface, as well as conflicting color stacks. However this local approach does not handle the available color stack budget which leads to invalid colors because any stack whose height is too large gets cut instead of getting shrunk. To solve this last budget problem, we can precompute the maximum available stack height per voxel. In a first pass, we accumulate the maximum distances to the queried surface voxels and store it with the color information on the surface. In the second pass, we compute the color as before, but use an adaptive color stack fitting the stack budget (we precompute color stacks for heights up to $h = 48$ and assume further voxels use white). Effectively this shrinks individual color stacks to fit the voxel budget while being spatially coherent.

Figure 10 illustrates the effect of adaptive stack height for a model with edges. The left-hand example shows dramatic change in color of four rows along the thick to the very thin triangles, whereas the adaptive strategy achieves much more consistent colors even for very thin triangles. The top-edge artifact (unprinted or unevenly printed top layers), shared by both examples, are printer artifacts. This is a typical problem of current printers which have difficulties in reproducing such small geometric features.

7 EVALUATION OF CONTONING

We use a custom, inkjet 3D printer (Appendix E.1). As the printhead has only 5 channels allotted to CMYKW, we cannot also print support material and thus our printed examples are all height fields. The printer achieves resolution of 720 dpi in X and Y and between 1,200 and 3,600 dpi in Z. The main limiting factor of our 3D printer is the print speed and jetting inconsistency.

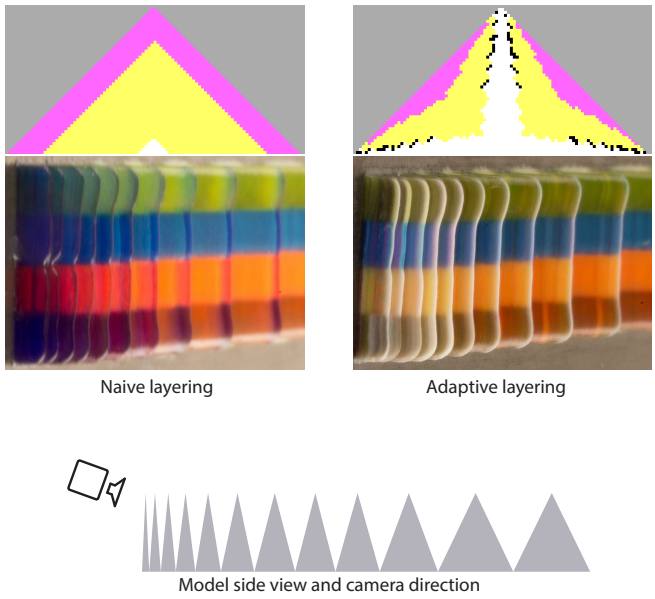


Fig. 10. Contoning at sharp edges. The naive implementation allows inks to collide and results in wrong color at edges. With adaptive ink assignment, we are able to change the color gradually towards the tip, such that the color change becomes less noticeable. A side-view illustration of color assignment of the profile section of the triangles is provided (top row). Photographs of the models (middle row) show that, due to printer difficulty, the heights of narrower triangles are not fully printed. The schematic drawing of the model, i.e., a number of triangles having fixed height and variable base, and the direction of camera for photographs are also shown (bottom row).

Print speed is limited due to the small number of nozzles per ink (only 60 for CMY) and the inefficient per-material nozzle arrangement (CMY inks are lined up along the Y axis which is perpendicular to our print motion, thus requiring multiple passes over the same area).

The inconsistency comes from jetting our highly viscous photopolymer inks (see Appendix E.2) using a consumer-level printhead originally designed for low-viscosity aqueous inks. The printhead is also mounted on an inexpensive gantry that has not been optimized for precision placement.

Our material formulation results in perfectly clear, mechanically rigid material to which we can add CMYKW pigments. The surface finish of our prints is highly specular. In order to remove the specularity from the prints, we print all flat prints upside-down on the printer metal platform that has fine microstructures which adds some microgeometry to the surface of our prints.

In this work, we allow for color stacks that are 48-layer deep. This generates a reasonable color gamut and corresponds to a feature size of about 1 mm. When using 48 color layers, our table includes 270,725 printer colors and thus ink stack entries. Figure 11 and Figure 12 show two images printed with our method. We start with sRGB source images that represent different colors and textures. We first generate the previews using our color reproduction framework. There is a considerable difference between original and preview images since the sRGB gamut is typically about two times larger

than most printer gamuts. We also show the photographs of our fabricated results. Note that part of the color difference between the preview and the print photograph is inherent to the challenges of print soft proofing. Proper soft proofing needs to be seen using a calibrated, colorimetric displays (e.g. Eizo CG245W). That said, the prints do not show significant artifacts, preserve the edges, and reproduce the colors well. On the other hand, the prints are somewhat blurry, largely because of dot gain. We analyze dot gain related issues in depth in Section 8.

Figure 1 e and f, and Figure 13 are 3D models printed using 3D contoning. Since 3D prints need a large number of layers, we limit their size in the favor of print time. Although the prints are small (maximum 1 inch wide), the details are pronounced and texture appears high definition.

Color prediction accuracy. In order to examine the prediction accuracy of our proposed color prediction model, we calibrate the model by printing all layer-layouts of the CMYK inks in 12 Z-layer intervals, while allowing a maximum of 48 layers. This yields 70 calibration (train) patches to print. For the test set, we print two test patches per each possible ink-combinations. Since there are 15 possible combinations for four inks, the size of our test set is 30 patches. The thicknesses of individual inks for each test sample are chosen randomly. This way we ensure that the test samples are within different regions of the color space.

We run the same experiment for prints with XY-layering. The obtained accuracy for the color prediction model calibrated in this mode is lower. There are significantly more inconsistencies in the prints because it is difficult for the printer to generate uniform vertical patches. This is not however an important problem since we use the more accurate color prediction model calibrated for Z-layering to print all results. The XY model is only used for estimation of the ink concentrations.

Following standards in color reproduction, we summarize the prediction accuracies in Table 1, with and without the weighting strategy. The prediction accuracy is expressed using the ΔE_{00} color difference metric as well as the RMS% difference between the measured and predicted spectral reflectances. The model accuracy is excellent for Z-layered prints and comparable to the accuracy of state of the art CMYK models for 2D printing. Both train and test patches are printed simultaneously and with the same batch of inks, thus repeatability errors are minimized in this experiment to show the full capacity of the color model. The weighting scheme is proved to be quite helpful by reducing the prediction error. Furthermore, Figure 14 shows a very good match between the measured and predicted spectral reflectance curves of some randomly selected test patches.

Contone color gamut. The gamut of our 48 Z-layer contone printer is 325 kilo CIELAB, comparable with the gamut volume for a typical inkjet CMYK printer [Rossier 2013]. The color gamut is directly influenced by the layer thickness. We expect that after a certain number of layers the gamut of color contoning saturates. The gamut saturates when adding a layer to the ink stack does not change the reflectance. In the literature, this is sometimes referred to as the infinite reflectance (R_∞) [Hébert et al. 2008].

Prediction mode	mean ΔE_{00}	max ΔE_{00}	mean RMS%
normal Z	1.79	3.61	1.71
weighted Z	1.24	2.54	1.19
normal XY	4.11	8.33	3.04
weighted XY	2.90	7.53	2.41

Table 1. The prediction accuracy of our color prediction model under the D65 illuminant for a test set of 30 samples. The results are listed with and without the weighted regression scheme.

Performance. Our Matlab implementation takes less than 5 minutes to generate an exhaustive lookup table that maps every entry in the sRGB gamut to layer-layouts. The ray-tracing in the gamut mapping step was accelerated using a native C implementation. Generating print data is instant for our 2D results. For 3D prints, our current implementation is bound by the signed distance function queries but is sufficiently fast that print data generation can be done upfront or interleaved with the printing process. We have also implemented a simple preview procedure based on ray casting and integrating through the color stacks. A real-time 3D preview is feasible and left for future work.

8 DOT GAIN

Dot gain is a well-known problem in 2D printing. In most drop-on-demand printing processes, the dot that gets deposited spreads within the substrate resulting in *physical* or *mechanical* dot gain. Furthermore, light can penetrate in an inked area, travel in the translucent paper, and exit from non-inked areas; this is called *optical* dot gain. Separating these two effects is an unsolved problem. Instead, in 2D printing the *total* dot gain is measured and compensated for through a set of color measurements [Hersch and Crété 2005].

Mechanical dot gain in our system results from inks merging in the substrate before they undergo phase change to a solid material and because of imprecisions in the inexpensive gantry, that introduces mechanical dot-gain as it cannot precisely register different layers on top of each other.

Optical dot gain in contoning is somewhat different from 2D optical dot gain. Transparent CMYK inks are stacked on a diffuse white. The light travels in the color stack and is reflected back at all directions. Therefore optical mixing between adjacent ink columns does occur.

Both physical and optical dot gain reduce the resolution and blur the results. Similarly to 2D printing, we can compensate for the total dot gain in 3D contoning without trying to distinguish between physical and optical dot gain. If we consider the total dot gain as a Gaussian filter, we can estimate the kernel parameters using a calibration print. This print consists of representative color rectangles that generates horizontal and vertical edges. Then, we take a photo of the blur calibration chart, convert it to linear RGB by an inverse gamma correction, and combine all color channels to produce a grayscale image. We fit a gaussian blur kernel that best reproduces the effect of dot gain on all edges. Having found the blur kernel, we apply the Richardson-Lucy algorithm [Lucy 1974; Richardson 1972] to produce a deconvoluted texture which



Fig. 11. An image printed with our contoning technique at 720 dpi. From left to right: sRGB original image, color preview after gamut mapping and layer separation, and fabricated result, respectively. The prints have 68 Z-layers, at least 20 of which are white backing layers. (print size (xyz): 2108×2560×68; print time: 9hrs).



Fig. 12. An image printed with our contoning technique at 720 dpi. From left to right: sRGB original image, color preview after gamut mapping and layer separation, and fabricated result, respectively. The prints have 68 Z-layers, at least 20 of which are white backing layers. (print size (xyz): 3825×2550×68; print time: 15hrs).



Fig. 13. The preview of our 3D printed models (respective left side of each model) and a photograph of the final print (respective right side of each model). The preview includes the gamut mapping and the layer separation steps. Note that our preview shows an orthogonal top-down view of the 3D printed models, without including the ambient occlusions and/or the blur kernel. Adding these effects for 3D prints can improve the preview significantly.

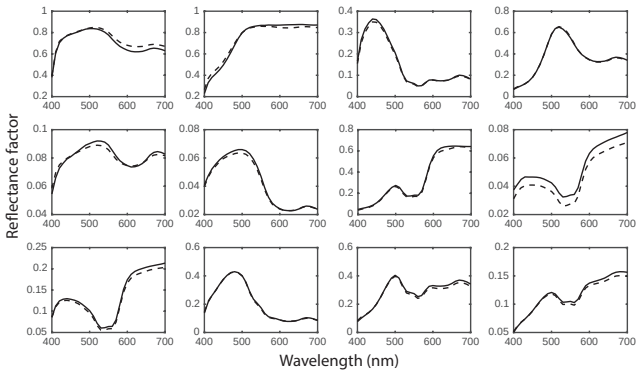


Fig. 14. The measured (solid-line) and predicted (dashed-line) spectral reflectance curves of a number of randomly selected test prints.

compensates for the blur effect. We assume that the dot gain kernel is the same regardless of the print orientation. Therefore for 3D objects, we use the physical size of the kernel obtained for flat prints. Figure 15 shows two image printed using contoning, with and without compensating for dot gain. Applying the kernel successfully recovers the lost details in the print due to the dot gain.

9 COMPARISON WITH HALFTONING

Halftoning is the only solution for most 2D printers as they print only a few layers. Contoning is an alternative solution when one can use depth for color mixing. The two methods can be thought as two extreme color reproduction techniques. We found it challenging to have a fair comparison of contoning with color 3D halftoning [Brunton et al. 2015]. One particular challenge is to determine the balance between the number of layers and inks concentrations for 3D halftoning. One can use fewer layers of more concentrated inks or more layers of less concentrated inks, with different consequences on the print quality. In the future, we would like to perform a comprehensive comparison involving either finding the optimized set of parameters (number of layers and ink concentrations) or showing examples that span them. Therefore, we only print flat images for comparison of halftoning and contoning and employ standard, state-of-the-art error diffusion halftoning (with only 4 CMYK layers). The only exception where we use multi-layer halftoning is the comparison of gamut volumes.

Figure 16 shows an image printed with contoning and halftoning. The halftone image has been printed using a state of the art color reproduction workflow [Rossier 2013]. The framework compensates for the total dot gain occurring in the halftone prints. To find the ink concentration for halftoning, we try to compute them such that they generate the same gamut volume as the gamut of contoning. We use a simple Neugebauer model [Hersch and Crété 2005] because it requires a small number of calibration samples. The number of measurements would become unwieldy, if we wanted to use a full ink spreading Yule-Nielsen and for each concentration. This is the reason of gamut mismatch between contone and halftone prints in Figure 16.

Color gamut. In order to have a fair comparison between the color gamut of contoning and halftoning, we use identical concentrations for CMY inks ($c = m = y = 0.25\%$), and identical number of layers ($L = 24$). For halftone color prediction, we use the ink-spreading Yule-Nielsen spectral Neugebauer model [Hersch and Crété 2005] calibrated for 24 layers. We use only CMY inks and up to 24 layers because with this setup the halftone prediction model shows a sufficient accuracy ($\Delta E_{00} = 4.9$) for gamut volume computation. Higher number of layers would result in unacceptable halftone model accuracy because halftone color prediction models have not been designed to account for the ink spreading only in 2D and not in 3D. For this identical setup, the gamut of color contoning is 10% larger than the color gamut obtained with color halftoning. As we discuss in Appendix B, layer superposition is known to generate more saturated colors than spatial mixing.

Print at low resolution. Figure 17 shows an image printed at 180 dpi printer resolution, using our contoning method and also with the error diffusion algorithm. The contone prints does not show any signs of spatial subsampling or other artifacts whereas the same image printed with error diffusion halftoning exhibits unacceptable dot patterns.

Color predictability. There are established models to accurately predict the color of CMYK halftones [Hébert and Hersch 2015]. However, if more inks are involved [Babaei and Hersch 2016] or if multi-layer halftones are required [Brunton et al. 2015], color prediction becomes very challenging. The prior work in color prediction of multi-layer 3D halftone [Brunton et al. 2015] uses a fully data-driven color prediction scheme. Such an empirical approach requires a large number of measurements and is only applicable to a fixed number of layers.

Our proposed color prediction model is the first model for predicting the color of a 3D printed stack of different inks. For CMYK prints, it needs a small number of calibration samples, and can predict the color of any thickness from 1 to L layers once calibrated for L layers. However, our sampling of the training samples of our color prediction model, has an exponential growth with the number of inks.

10 DISCUSSION AND FUTURE WORK

While advantages of contoning are due to the use of depth for creating colors, advantages of halftoning come as a consequence of it using the surface (and less depth) for color synthesis. Halftoning can reproduce colors with the significantly lower number of layers. This will guarantee a smaller feature size and less dot gain. A possible extension is to combine both methods in a complementary manner. One way is to use halftoning for high spatial frequency areas, and contoning for uniform regions where the halftone dots are most objectionable.

Future advances in 3D printers that enable printing with a larger number of materials can release the full potential of contoning by allowing it to create colors with a minimal height of the ink stack. This will guarantee a smaller feature size and less dot gain. With a larger number of inks, contoning generates a larger color gamut while keeping the ink stack height low. With the current trends in the market, such 3D printers are becoming more and more



Fig. 15. Left: The original image (gamut mapped). Middle: Photograph of the print of the original image with standard contoning. Right: Photograph of the print of the presharpened image. The estimated blur kernel is a Gaussian with filter size 30 XY-voxels and standard deviation 10, estimated from a blur calibration print.

accessible. The most recent industrial inkjet 3D printer can print with 7 materials [Stratasys 2016]. With a larger number of inks, we should strive for a more physical and scalable color prediction model.

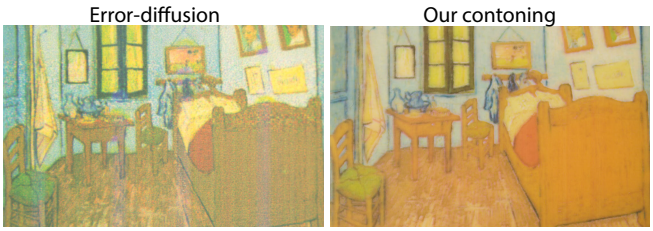


Fig. 16. An image printed on our printer with error-diffusion halftoning, and our contoning method.

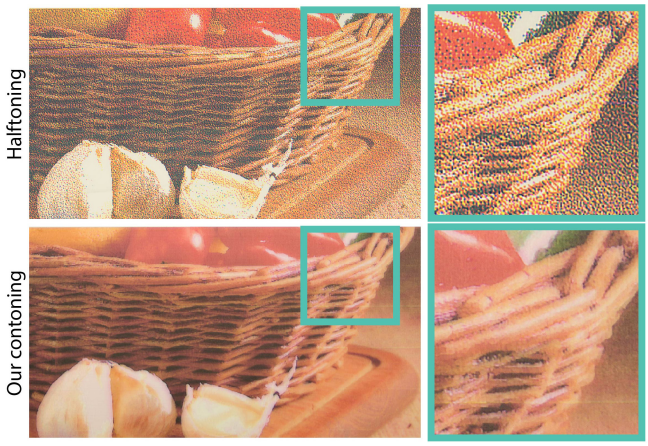


Fig. 17. An image printed with halftoning (top) and contoning (below) at 180 dpi printer resolution (print size (xy) = 2280×1300).

A main simplifying assumption in our model is that we treat each ink column independently of its neighbors. In reality, these columns interact with each other due to both optical and mechanical mixing. The interaction is more complex for more difficult geometries, where factors such as interreflection come into play. Our blur kernel estimation and compensation method is a first solution carried out in the texture domain. A more promising approach is to modify the material arrangement, and account for the mechanical mixing and the non-local light transport. Instead of finding the ink composition of a single column, optimizing for the materials within a neighborhood, while considering its geometry, is most interesting. One can predict these interactions by precomputing them within small neighborhoods through a combination of measurement and simulation. Then, similar to structure preserving halftoning [Pang et al. 2008], the difference between the original appearance and the print appearance can be minimized. Another exciting solution to deal with this problem is to design techniques that compensate for the unwanted light transport or ink spreading within printed volume. For example, it could be addressed by printing thin, opaque walls at color edges.

A user study of contone prints in order to understand their aesthetic aspects and to compare them with halftone prints is left for future work. For sensitive applications, such as fine-art reproduction, this would be necessary. We believe that the current workflow lends itself well to reproduction of other appearance attributes, such as gloss and translucency. Finally, it would be exciting to print with our workflow on an industrial 3D printer. The only requirement is the ability to control the ink concentrations.

Concluding remarks. The method presented in this article is simple. While we believe this to be one of its main virtues, it is largely thanks to eliminating the need for halftoning. The new color reproduction system proposed in this paper paves the way for high-definition color reproduction for 2.5D and 3D printers.

A SAUNDERSON CORRECTION

The Kubelka-Munk theory assumes no discontinuity in the refractive index. In reality, when light enters the material from the air side, a portion r_s is reflected back in the specular direction. When light exits the material, a portion r_i of light is reflected back into the material and bounces infinitely between the ink side of material-air interface and the substrate. Based on these observations, Saunderson [1942] proposed a correction on the measured reflectance spectrum $R_m(\lambda)$, in order to infer the *intrinsic* reflectance $\rho(\lambda)$ as if there was no interface:

$$\rho(\lambda) = \frac{R_m(\lambda) - Kr_s}{1 + (1-K)r_i r_s + r_i R_m(\lambda) - r_i - r_s}, \tag{13}$$

where K is set to 1 if the device measures the specular reflection, otherwise it is set to 0. r_i and r_s are computed using the Fresnel equations and their values depend on the angle of incident illumination θ and the relative index of refraction between ink and air. For our measuring geometry with hemispherical light (see Appendix E.3) and assuming uniform illumination, r_s is computed by integrating the Fresnel component $F(\theta)$ over the illumination hemisphere [Simonot et al. 2006]:

$$r_s = \int_0^{2\pi} \int_0^{\pi/2} \frac{F(\theta)}{\pi} \cos\theta \sin\theta d\theta d\phi, \tag{14}$$

which simplifies to

$$r_s = \int_0^{\pi/2} F(\theta) \sin(2\theta) d\theta. \tag{15}$$

The same equation is used to compute r_i assuming the light coming from the substrate (and hitting the non-air side of the interface) is Lambertian. The only difference is that the refractive index is inverted as the light goes from denser media to the media with lower index of refraction. Having set the refractive index of our polymer carrier to 1.5, we obtain $r_s = 0.092$ and $r_i = 0.596$.

B 3D PRINTED GLAZE

In painting, glaze is a technique where transparent layers of paint are applied on top of each other. Each layer is made of pigments scattered in a thin transparent medium. The artist creates different colors by varying the number and color of glaze layers. Before applying a new layer, the bottom layer should be dry. Glazes are known to produce most saturated colors [Elias and Simonot 2006]. A very similar process also occurs in our prints; every time a layer is deposited, it is instantly dried by UV-radiation and then the next layer is applied.

Here, we study the color and spectral changes when increasing the layers of an ink. Figure 18 (left) shows the spectral reflectance factor of 1 to 50 layers of a yellow ink printed with our 3D printer. As expected, the reflectance decreases with increasing the number of layers. In theory, if we continue adding more layers, there will be a point after which the reflectance does not change [Hébert et al. 2008].

For showing the colorimetric changes, we use hue-angle, chroma diagram computed in CIECH color space [Schanda 2007]. From CIELAB a^* and b^*, we compute the hue-angel as $\arctan(b^*/a^*)$, and

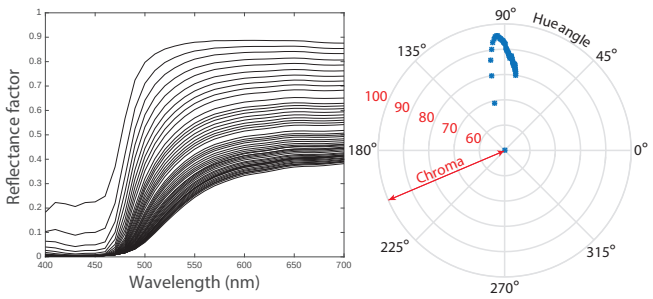


Fig. 18. Spectral and color changes due to contoning; Left: the measured spectral reflectance factor of 1 to 50 layers of a yellow ink with 0.5% ink concentration. Right: The hue-angle, chroma diagram of 1 to 50 layers of a yellow ink with 0.5% pigment concentration.

chroma as $\sqrt{a^{*2} + b^{*2}}$. In this polar diagram, different angles represent different hues and larger radii correspond to more saturated colors. Figure 18 (right) shows the effect of increasing the number of layers in this graph. By adding more layers, the chroma first increases and then decreases with a shift in the hue-angle. This phenomenon has already been studied in glazes [Elias and Simonot 2006] and also in piling nonscattering colored sheets [Hébert et al. 2008]. These two processes are very similar to UV-ink color 3D printing.

C COUNTING LAYER-LAYOUTS

First we compute the number of all possible layer-layouts with N inks and maximum L layers. For 1 layer, there are N, for 2 layers there are N^2 and for L layers there are N^L possible layer-layouts. This leads to a geometric series:

$$N + N^2 + \ldots + N^L = \frac{N^{L+1} - 1}{N - 1} - 1. \tag{16}$$

For stratified layering, number of all possible layer-layouts with N inks and maximum L layers is equal to the number of non-negative solutions of the following equation:

$$I_1 + I_2 + \ldots + I_N + I_\Phi = L. \tag{17}$$

where I_i is the number of layers of the i^{th} ink and I_Φ is the number of void layers. The number of non-negative solutions to Equation 17 is know to be [Rosen and Krithivasan 2007]:

$$\Upsilon = \binom{L + N + 1 - 1}{N + 1 - 1} \tag{18}$$

which after excluding the single all-void layer from our counting gives:

$$\Upsilon = \binom{L + N}{N} - 1. \tag{19}$$

D GAMUT MAPPING

The gamut mapping takes place in a device-independent color space which is usually the CIELAB [Morovic and Luo 2001]. We therefore convert colors from different spaces into the CIELAB color

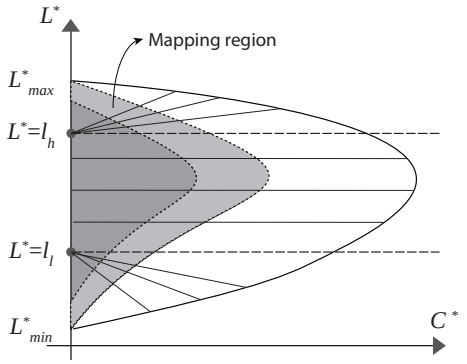


Fig. 19. Lightness-chroma plot of the sRGB gamut (union of white, light gray and dark gray areas) and a printer gamut (union of two light and dark gray areas) in a constant hue-angle plane. In this figure, the lightness mapping has already taken place such that the minimum and maximum lightnesses of both gamuts coincide. In this step, the colors with chroma outside the printer gamut are mapped into the mapping region (light gray region) of the printer gamut . Colors with $L^* \leq l_l$ or $L^* \geq l_h$ are mapped along the line segment made of those colors and l_l or l_h. Other colors are mapped perpendicularly to the L^* axis. The choice of l_l, l_h and the mapping region is not deterministic and can change from gamut to gamut.

space. This is particularly straightforward for input colors in standard RGB (sRGB) [Morovic and Luo 2001]. In order to generate the CIELAB gamut of the printer, the resulting reflectance spectra from all possible stratified layer-layouts are computed using our developed color prediction model. Each spectrum is then converted to CIEXYZ tristimulus values with a D65 illuminant and in respect to the 2°standard observer [Wyszecki and Stiles 1982]. The CIE-XYZ values are then converted into CIELAB color coordinates using the reflectance spectrum of our 3D printed white material as a white reference. The non-convex gamut boundary is obtained by performing a ball pivoting surface reconstruction on the CIELAB color points [Bernardini et al. 1999].

For the gamut mapping, we follow the standard methods in the color reproduction field [Morovic and Luo 2001]. As a first step, we map the lightness range of the sRGB gamut into the lightness range of the printer gamut. The lightness-mapped gamut is then chroma mapped into the gamut of the printer for each hue-angle slice. See Figure 19 for more details [Rossier 2013].

E EXPERIMENTAL SETUP

E.1 3D Printer

We use MultiFab [2015], a laboratory-scale, custom-built, multi-material 3D printer that uses an inkjet-based, phase-shift, photopolymer printing process. The system is similar to commercial printers developed by Stratasys [2016] and 3D Systems [2013]. We use a consumer-level, drop-on-demand (DoD) Epson Workforce 30 piezo inkjet printhead that consists of several nozzles. Each nozzle is provided with a piezo-actuator, which on application of a voltage pulse (or waveform) generates pressure oscillations inside the ink channel. These pressure oscillations push the ink drop out of the nozzle [Khalate et al. 2012]. The printhead accommodates 5 channels, allowing

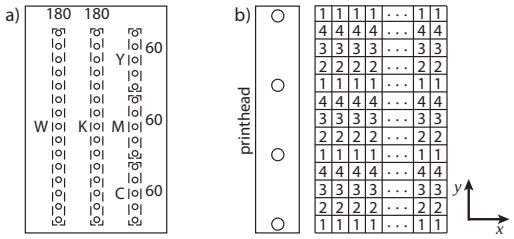


Fig. 20. a) The schematic orientation of the nozzles in Epson Workforce 30 printhead. There are three lines of nozzles. The first and second lines are connected to two ink channels and they have 180 nozzles each, while the third line is connected to three channels and each channel has 60 nozzles. b) The printhead moves with the maximum speed in X direction and traverses the Y direction in four passes.

5 materials to be printed at the same time. As we use 5 materials (CMYKW), we cannot have support material and our 3D prints are height field examples. The inks are heated and kept at elevated operational temperature to reduce their viscosity, improve ink flow and jetting. We print with droplet volume of between 10 and 15 picoliters and achieve resolution of 720 dpi in X and Y and about 2000 dpi in Z, subject to the individual material characteristics. The printer additively builds objects by printing one layer at a time; a UV LED module is affixed immediately to the printhead and is activated concurrently while the printhead is printing in either direction. The emitted UV radiation pins and solidifies the jetted droplets.

The amount of the overlapping is primarily a function of the drop volume which itself is a function of the waveform applied to the piezo-electronic actuators at each nozzle channel. We observed that the default waveform of the printhead results in very large drop volumes. This causes sever overlapping of the adjacent contone columns. Therefore, we experimentally designed a new waveform for our materials that corresponds to a smaller drop volume.

The main limiting factor of our 3D printer is the print speed. The slow process is due to the small print swathe of the printhead (1 inch) and the limited number of nozzles (540). There are three lines of 180 nozzles on the printhead one of which is connected to three ink channels and therefore three out of five channels are 1/3" wide and have only 60 nozzles (Figure 20 a). Thus when printing with more than two inks, our speed is reduced. The second factor influencing the print speed is the nozzle density on the printhead. The rows of the printhead are not fully populated with nozzles. The printhead needs to traverse the same the Y direction multiple times in order to fire at all designated pixels and fulfill the resolution requirement. In our case, the printhead needs 4 passes (Figure 20 b). The other factor is the speed of the printhead movement in X direction which itself depends on the firing frequency of the printhead. With higher firing frequency, we can move faster in X and increase the overall printing speed. Finally, the employed inkjet printhead was not designed to work with highly viscous photopolymer ink such as ours; the jetting consistency is such that it requires continuous nozzle rotation which can further slow down printing of certain geometries.

Industrial printheads have significantly larger swathes (2.55" is considered standard), and are designed to work in one material

per printhead configurations. Further, their firing frequency can be several times faster than a typical consumer-level printhead such the one employed in our system.

E.2 Materials

All materials that we use are UV-curable photopolymers. We mix two monomers and one olygomer to make our base carrier. We add a small amount of photoinitiator that triggers the polymerization when exposed to the UV light. Before adding color pigments to these materials, we measure and test various properties including density, surface tension, viscosity, thermal stability, shelf-life, and phase change speed. Additionally, we optimize the jetting of each ink by adjusting the waveform applied to the printhead piezos.

We expect our formulation to result in a perfectly clear, mechanically rigid print. However, this depends on the careful selection of the photoinitiator and its concentration. We observe that some photoinitiators introduce an objectionable yellowing tint to prints. This is because the unreacted photoinitiators inside the polymer tend to reduce the intensity of the blue region of the spectrum. Photoinitiators have their peak absorption in the UV range. The location of this peak varies in different photoinitiators and usually shifts toward the blue region when increasing the photoinitiator concentration. Yellowing occurs when this peak is close to the visible blue range. Formulating an optimal ink involves selecting a photoinitiator that works well with the given UV curing unit and allows curing to occur in low enough concentrations that limit the yellowing effect while still promoting full polymerization.

E.3 Spectral Measurements

For calibration of our color prediction model, we measure all train and test samples using a sphere spectrophotometer, namely the X-Rite Color i7 instrument. This instrument provides a diffuse incident illumination through an integrating sphere. The diffuse incident light falls on the measured sample at all possible angles and the sensor captures the reflected light at 8°from the normal incidence. The response is in the form of reflectance factor sampled at 10 nm intervals from 350 nm to 740 nm. We work with the visible range from 400 nm to 700 nm, therefore every measurement is a 31-vector of reflectance factors. The device gives the possibility to include or exclude the specular component reflected off the sample. We use the device in the specular component excluded mode.

ACKNOWLEDGMENTS

We would like to thank our reviewers for the helpful suggestions, X-Rite (especially Peter Ehbets) for lending us the spectrophotometer, David Kim and Wenshou Wang for help with materials, and Subra Subramanian, Javier Ramos, and Pitchaya Sitthi-amorn for help with printing. We thank Sergiu Gaman, Romain Rossier and Roger Hersch from EPFL for the gamut mapping software. We are grateful to Sylvain Paris for his feedback on the manuscript. The 3D models and images of physical materials were obtained from TurboSquid, GrabCAD, Wikimedia and ISO. Vahid Babaei acknowledges the Swiss National Science Foundation (SNSF) support, under grant number P2ELP2_161843.

REFERENCES

Eugene Allen. 1966. Basic equations used in computer color matching. *JOSA* 56, 9 (1966), 1256–1257.

Eugene Allen. 1974. Basic equations used in computer color matching, II. Tristimulus match, two-constant theory. *JOSA* 64, 7 (1974), 991–993.

Isaac Amidror. 2009. *The Theory of the Moiré Phenomenon, Volume I: Periodic Layers* (2nd ed.). Springer.

Vahid Babaei and Roger D Hersch. 2016. N-Ink printer characterization with barycentric subdivision. *IEEE Transactions on Image Processing* 25, 7 (2016), 3023–3031.

Vahid Babaei, Romain Rossier, and Roger D Hersch. 2012. Reducing the number of calibration patterns for the two-by-two dot centering model. In *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 829208–829208.

Farhan Baqai, Je-Ho Lee, Jan P Allebach, and others. 2005. Digital color halftoning. *Signal Processing Magazine, IEEE* 22, 1 (2005), 87–96.

Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Cláudio Silva, and Gabriel Taubin. 1999. The ball-pivoting algorithm for surface reconstruction. *Visualization and Computer Graphics, IEEE Transactions on* 5, 4 (1999), 349–359.

Roy S Berns. 1993. Spectral modeling of a dye diffusion thermal transfer printer. *Journal of Electronic Imaging* 2, 4 (1993), 359–370.

Roy S Berns. 1997. A generic approach to color modeling. *Color Research & Application* 22, 5 (1997), 318–325.

Alan Brunton, Can Ates Arikan, and Philipp Urban. 2015. Pushing the limits of 3d color printing: Error diffusion with translucent materials. *ACM Transactions on Graphics (TOG)* 35, 1 (2015), 4.

S Chandrasekhar. 1960. *Radiative Transfer*. Dover Publications, New York.

Denis Defibaugh. 1997. The Collotype: History, process & photographic documentation. *Rochester Institute of Technology* (1997).

Yue Dong, Jiaping Wang, Fabio Pellacini, Xin Tong, and Baining Guo. 2010. Fabricating Spatially-varying Subsurface Scattering. *ACM Trans. Graph.* 29, 4 (July 2010), 62:1–62:10.

Robert C. Durbeck. 2012. *Output Hardcopy Devices*. Academic Press.

Mady Elias and Lionel Simonot. 2006. Separation between the different fluxes scattered by art glazes: explanation of the special color saturation. *Applied optics* 45, 13 (2006), 3163–3172.

Miloš Hašan, Martin Fuchs, Wojciech Matusik, Hanspeter Pfister, and Szymon Rusinkiewicz. 2010. Physical Reproduction of Materials with Specified Subsurface Scattering. *ACM Trans. Graph.* 29, 4 (July 2010), 61:1–61:10.

Mathieu Hébert and Roger D Hersch. 2015. Review of spectral reflectance models for halftone prints: principles, calibration, and prediction accuracy. *Color Research & Application* 40, 4 (2015), 383–397.

Mathieu Hébert, Roger D Hersch, and Lionel Simonot. 2008. Spectral prediction model for piles of nonscattering sheets. *JOSA A* 25, 8 (2008), 2066–2077.

Roger David Hersch and Frédérique Crété. 2005. Improving the Yule-Nielsen modified Neugebauer model by dot surface coverages depending on the ink superposition conditions. In *Electronic Imaging 2005*. International Society for Optics and Photonics, 434–447.

Matthias B. Hullin, Ivo Ihrke, Wolfgang Heidrich, Tim Weyrich, Gerwin Damberg, and Martin Fuchs. 2013. Computational Fabrication and Display of Material Appearance. In *Eurographics 2013 - State of the Art Reports*, M. Sbert and L. Szirmay-Kalos (Eds.). The Eurographics Association. https://doi.org/10.2312/conf/EG2013/stars/137-153

Isaac Kauvar, Samuel J Yang, Liang Shi, Ian McDowall, and Gordon Wetzstein. 2015. Adaptive color display via perceptually-driven factored spectral projection. *ACM Transactions on Graphics (TOG)* 34, 6 (2015), 165.

Amol A Khalate, Xavier Bombois, Gérard Scorletti, Robert Babuska, Sjirk Koekebakker, and Wim De Zeeuw. 2012. A waveform design method for a piezo inkjet printhead based on robust feedforward control. *Journal of Microelectromechanical Systems* 21, 6 (2012), 1365–1374.

P. Kubelka and F. Munk. 1931. Ein Beitrag zur Optik der Farbanstriche. *Zeitschrift für technische Physik* 12 (1931), 593–601.

Leon B Lucy. 1974. An iterative technique for the rectification of observed distributions. *The astronomical journal* 79 (1974), 745.

Ján Morovic and M Ronnier Luo. 2001. The fundamentals of gamut mapping: A survey. *Journal of Imaging Science and Technology* 45, 3 (2001), 283–290.

PS Mudgett and LW Richards. 1971. Multiple scattering calculations for technology. *Applied Optics* 10, 7 (1971), 1485–1502.

Victor Ostromoukhov. 2001. A simple and efficient error-diffusion algorithm. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 567–572.

Wai-Man Pang, Yingge Qu, Tien-Tsin Wong, Daniel Cohen-Or, and Pheng-Ann Heng. 2008. Structure-aware halftoning. In *ACM Transactions on Graphics (TOG)*, Vol. 27. ACM, 89.

Marios Papas, Christian Regg, Wojciech Jarosz, Bernd Bickel, Philip Jackson, Wojciech Matusik, Steve Marschner, and Markus Gross. 2013. Fabricating Translucent Materials Using Continuous Pigment Mixtures. *ACM Trans. Graph.* 32, 4 (July 2013), 146:1–146:12.

Petar Pjanic and Roger D Hersch. 2015. Color imaging and pattern hiding on a metallic substrate. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 130.

Tim Reiner, Nathan Carr, Radomír Měch, Ondřej Šťava, Carsten Dachsbacher, and Gavin Miller. 2014. Dual-color mixing for fused deposition modeling printers. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, 479–486.

William Hadley Richardson. 1972. Bayesian-based iterative method of image restoration. *JOSA* 62, 1 (1972), 55–59.

Kenneth H Rosen and Kamala Krithivasan. 2007. *Discrete Mathematics and its Applications* (6 ed.). McGraw-Hill New York. 370–382 pages.

Romain Rossier. 2013. *Framework for Printing with Daylight Fluorescent Inks.* Ph.D. Dissertation. Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne.

JL Saunderson. 1942. Calculation of the color of pigmented plastics. *JOSA* 32, 12 (1942), 727–729.

János Schanda. 2007. *Colorimetry: Understanding the CIE System.* John Wiley & Sons.

Christian Schüller, Daniele Panozzo, Anselm Grundhöfer, Henning Zimmer, Evgeni Sorkine, and Olga Sorkine-Hornung. 2016. Computational thermoforming. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 43.

Gaurav Sharma and Raja Bala. 2002. *Digital Color Imaging Handbook.* CRC press.

Gaurav Sharma, Wencheng Wu, and Edul N Dalal. 2005. The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color research and application* 30, 1 (2005), 21–30.

Lionel Simonot, Mathieu Hébert, and Roger D Hersch. 2006. Extension of the Williams-Clapper model to stacked nondiffusing colored coatings with different refractive indices. *JOSA A* 23, 6 (2006), 1432–1441.

Pitchaya Sitthi-Amorn, Javier E. Ramos, Yuwang Wangy, Joyce Kwan, Justin Lan, Wenshou Wang, and Wojciech Matusik. 2015. MultiFab: A Machine Vision Assisted Platform for Multi-material 3D Printing. *ACM Trans. Graph.* 34, 4, Article 129 (July 2015), 11 pages. https://doi.org/10.1145/2766962

Stratasys. 2016. Stratasys J750 the ultimate full-color multi-material 3D printer. http://www.stratasys.com/3d-printers/production-series/stratasys-j750. (2016). [Online; accessed 15-October-2016].

3D Systems. 2013. ProJet 5500X Multi-Material 3D Printer. (2013). http://www.3dsystems.com/files/projet-5500x-1113-usen-web.pdf

Gunter Wyszecki and Walter Stanley Stiles. 1982. *Color Science.* Vol. 8. Wiley New York.

John AC Yule. 1967. *Principles of color reproduction: applied to photomechanical reproduction, color photography, and the ink, paper, and other related industries.* Wiley New York.

Yizhong Zhang, Yiying Tong, and Kun Zhou. 2016. Coloring 3D Printed Surfaces by Thermoforming. *IEEE Transactions on Visualization and Computer Graphics* (2016).

Yizhong Zhang, Chunji Yin, Changxi Zheng, and Kun Zhou. 2015. Computational hydrographic printing. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 131.

Yonghui Zhao and Roy S Berns. 2009. Predicting the spectral reflectance factor of translucent paints using Kubelka-Munk turbid media theory: Review and evaluation. *Color Research & Application* 34, 6 (2009), 417–431.