HAL
open science

Colored fused filament fabrication

Haichuan Song, Jonàs Martínez, Pierre Bedell, Noemie Vennin, Sylvain Lefebvre

To cite this version:

Haichuan Song, Jonàs Martínez, Pierre Bedell, Noemie Vennin, Sylvain Lefebvre. Colored fused filament fabrication. ACM Transactions on Graphics, 2019, 38 (5), pp.1-11. 10.1145/3183793 . hal-01660621

HAL Id: hal-01660621

https://inria.hal.science/hal-01660621

Submitted on 19 Apr 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Colored Fused Filament Fabrication

HAICHUAN SONG, JONÀS MARTÍNEZ, PIERRE BEDELL, NOÉMIE VENNIN, and SYLVAIN LEFEBVRE, Université de Lorraine, CNRS, Inria, LORIA

Fig. 1. Our technique enables color printing with precisely controlled gradients using a filament printer. **Left:** The printer is equipped with an off-the-shelf nozzle that inputs multiple filaments (three in the picture) and outputs molten mixed plastic through a single exit hole. Such devices cannot directly be used to print gradients: the transition between mixtures takes time, and this timing varies unpredictably. **Middle:** Using our technique, a user can 3D paint a virtual model and print an accurate, colored physical realization of her design. Printing uses standard, widely available materials (e.g. PLA, PET). **Right:** Our approach divides each layer into strata of varying heights to reproduce colors, printing them one on top of another. The color of each stratum is optimized to reduce visible stripes and print time.

Fused filament fabrication is the method of choice for printing 3D models at low cost and is the de-facto standard for hobbyists, makers, and schools. Unfortunately, filament printers cannot truly reproduce colored objects. The best current techniques rely on a form of dithering exploiting occlusion, that was only demonstrated for shades of two base colors and that behaves differently depending on surface slope.

We explore a novel approach for 3D printing colored objects, capable of creating controlled gradients of varying sharpness. Our technique exploits off-the-shelves nozzles that are designed to mix multiple filaments in a small melting chamber, obtaining intermediate colors once the mix is stabilized.

We apply this property to produce color gradients. We divide each input layer into a set of strata, each having a different constant color. By locally changing the thickness of the stratum, we change the perceived color at a given location. By optimizing the choice of colors of each stratum, we further improve quality and allow the use of different numbers of input filaments.

We demonstrate our results by building a functional color printer using low cost, off-the-shelves components. Using our tool a user can paint a 3D model and directly produce its physical counterpart, using any material and color available for fused filament fabrication.

CCS Concepts: • **Computing methodologies** → **Shape modeling**.

ACM Reference Format:
Haichuan Song, Jonàs Martínez, Pierre Bedell, Noémie Vennin, and Sylvain Lefebvre. 2019. Colored Fused Filament Fabrication. *ACM Trans. Graph.* 1, 1 (July 2019), 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Filament 3D printers have been widely adopted by hobbyists, enthusiasts, FabLabs, schools and small companies alike in the past few years. These printers fabricate a physical object from its numerical counterpart by accumulating melted material in successive layers: a filament is pushed through a hot nozzle where it melts and exits by a small hole. The molten material fuses with the layer just below and solidifies quickly as it cools down. By moving the extruder in the XY plane – the build direction being along Z – a solid layer of material is constructed. This principle can be implemented with widely available, inexpensive parts and electronics, and the RepRap movement produced many open source 3D printer designs. This, in turn, triggered a vast community effort to push the boundaries of fused filament fabrication: better software, new

This work is supported by ERC grant ShapeForge (StG-2012-307877).
Authors' address: Haichuan Song; Jonàs Martínez; Pierre Bedell; Noémie Vennin; Sylvain Lefebvre, F-54000, Nancy, France, Université de Lorraine, CNRS, Inria, LORIA.

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.
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
0730-0301/2019/7-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

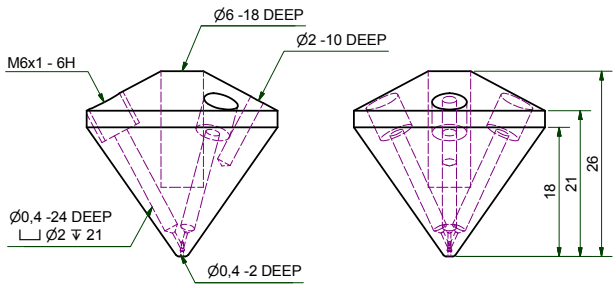


Fig. 2. Diamond nozzle design, blueprint by [RepRap.me 2015]. Three filaments are pushed into the nozzle which has a single exit hole. A five filaments version is also available.

materials with specific properties (e.g. colored, transparent, conductive, ferromagnetic), and novel applications. Thanks to a constant improvement in software and modeling tools, the quality of printed parts has steadily increased.

While often regarded as a low-quality process, fused filament fabrication has several unique advantages. From the user perspective, it is inexpensive to operate (by a large margin) and involves less cleaning and precautions than resin and powder systems. Fabricated parts are also functional, as they retain many of the desirable properties of (e.g.) plastic materials. From the fabrication perspective, the process offers unique degrees of freedom: most notably the ability to use multiple filaments and materials within the same print and the ability to freely move up and down the nozzle during deposition – creating heightfield-like paths [Song et al. 2017].

Among the features that are expected from a 3D printer, and that are of special interest to Computer Graphics, is the ability to produce colorful prints. While this has been achieved for a few other technologies – for which it nevertheless remains a topic of active research [Babaei et al. 2017; Brunton et al. 2018] – this ability is still lacking for filament printers, despite several attempts (see Section 2). This is especially unfortunate since many users are hobbyists printing decorative objects, toys, parts for model cars and planes, and boardgame figures for which colors and aesthetics are important. Color is also an effective way to carry information in the context of scientific visualization and education.

Our work proposes a novel technique that prints colorful parts reliably from multiple filaments. We exploit *multi-in-one-out* nozzles that input multiple filaments and melt them in a unique chamber. These nozzles are inexpensive and available from several vendors (e.g. Diamond extruder [2015], Figure 2). It would seem that such a device is enough to provide color printing: one could expect to produce color gradients by mixing various ratios of colored filaments. Unfortunately, while this works for *constant* ratios, dynamically changing the mix requires a significant transition time and cannot be controlled in a way that would produce reliable, precise gradients of colors (see Section 2.2).

Instead, our method achieves colored prints by dividing each colored layer into a number of *strata*. Each *stratum* has a constant color, but this color is different from one stratum to the next (e.g. think of each stratum as being red, green, or blue). By slightly curving each stratum *within* the layer thickness, adapting the plastic flow, the ratio of each filament in the final result is precisely controlled. This produces gradients of colors. The idea is illustrated in Figure 1, right. Note that the initial layers remain unchanged, only the strata within are optimized.

Our approach is facilitated by the multi-in-one-out nozzle in two ways. First and foremost, the ability to mix reliably at *constant* ratios let us choose the color of each stratum. This allows to minimize defects such as visible stripe patterns and reduce print time as well as using much fewer strata than there are input filaments. To illustrate this, consider a constant color layer which is a mix of all filaments. Instead of using one stratum per filament, we can use a single stratum while mixing the filaments in the nozzle directly. Our approach generalizes this principle. Second, as there is a single nozzle mounted on the carriage, there are no calibration defects when depositing a different color for each stratum. This would be the case if switching between different nozzles mounted side by side [Hergel and Lefebvre 2014; Kuipers et al. 2018; Reiner et al. 2014].

Our contributions are:

- A novel approach to produce reliable color gradients in a 3D printed part, following an input color field, by depositing successive strata having each a constant color.
- An optimization procedure to choose the colors of strata such as to reduce color reproduction artifacts, and to allow the use of an arbitrary number of input filaments.
- A complete implementation running on inexpensive, widely available filament printer hardware.

Our technique is implemented in a standard slicer software. It supports a varying number of input filaments, making it compatible with existing multi-in-one-out nozzles that input from two (e.g. *E3D Cyclop*) to five input filaments (e.g. *Full Color Diamond extruder*).

Limitations. In this work we assume that the input specifies *ratios* of the base filaments. The ratios are specified by the user, from e.g. a 3D printed palette, where each color corresponds to known mixing ratios (see Figure 15). We do *not* consider the color separation problem – how to best approximate an arbitrary color field using the *available* filaments – this is left for future work. We concentrate on the lower level challenge of depositing specified input ratios accurately while minimizing print time and defects.

2 RELATED WORK

The ability to print with multiple colors depends on the technology and is often linked to the ability to print with multiple materials. We focus on color printing, with a particular emphasis on filament printers, since that is our target. We only

briefly mention other technologies and multi-material printing. We refer the interested reader to recent survey papers for more details [Livesu et al. 2017].

2.1 Color printing in additive manufacturing

The main technologies capable of color printing apply colors on entire layers before solidification, e.g. inkjet on powder [Sachs et al. 1994] and inkjet on paper [Mcor 2005] for laminated fabrication. DLP resin printers, which solidify layers by selectively exposing them to light, can perform multi-material printing through the use of multiple resin tanks [Zhou et al. 2011]. Multi-jet technologies [PolyJet 1998; Sitthi-Amorn et al. 2015], which locally deposit and solidify droplets of different resins, also enable color printing. The resins are tinted and tiny droplets of e.g. cyan, magenta, yellow are deposited following a dithering procedure. Most processes only use a discrete number of base materials. Several approaches [Cho et al. 2001; Wu et al. 2000] discuss process planning to convert continuously varying material information into a limited set of base materials by a half-toning technique. Cignoni et al. [2008] exploit color printing to improve the visual perception of geometric details of a printed object, taking into account sub-surface scattering. Brunton et al. [2015] rely on half-toning and translucency in the context of color reproduction – producing highly realistic colored 3D objects, including translucent appearances [2018]. Elek et al. [2017] introduce a technique to counteract heterogeneous scattering and obtain sharp details along 3D prints. Auzinger et al. [2018] explore how to manufacture structural colors at the nanoscale, without requiring pigments. Babaei et al. [2017] propose to layer colored resin droplets in small stacks normal to the surface. By varying the thickness of each colored layer within the stack, precise color reproduction is achieved. This exploits translucency through individual stacks rather than traditional dithering.

The major challenge we face is that filament deposition requires continuity when extruding filament, while the aforementioned high-end technologies can address (solidify) voxels individually, switching materials instantaneously.

2.2 Color printing with filaments

There is a strong interest in filament color printing. This is witnessed in particular in the hobbyist community, with several successfully funded community projects [Chan 2017; ORD-Solutions 2016; Prusa 2016; RepRap.me 2015; Sammut 2016]. The main idea to achieve color printing with filament is to somehow blend filaments of different colors. This can be achieved in several ways.

A first approach is to use a printer having multiple hotends, typically mounted side by side on the same carriage, and loaded with filaments of different colors. Colors are deposited alongside another in each layer. This is made challenging by calibration issues and filament oozing. Hergel and Lefebvre [2014] specifically target these issues through path planning and part orientation.

Another approach is to use a single nozzle and switch filaments during printing. This can be done manually[1] with limited complexity (as this requires constant supervision/intervention). The *Palette* project [Mosaic 2015] automates the process with a device that splices and joins multiple filaments into a single, continuous, multi-color filament. Another automated approach is to rely on a *switching extruder* [Printers 2017; Prusa 2016; Sammut 2016], where a mechanical system allows for quick selection between different filaments. Using these devices reveals an interesting effect. There is always a transition between filaments: when a different material is pushed into the melting chamber, it mixes with the previous one. During this transition, the output material takes an intermediate color. To hide this transition, a *purge tower* is printed alongside the print and disposed of afterward.

The aforementioned methods select one filament at a time, and are not designed to produce mixtures. To create the perception of gradients, [Reiner et al. 2014] interweave filaments of different colors along the outer perimeter of a print. By changing which filament (color) is the most visible, a gradient effect similar to dithering is achieved. This approach, implemented in Photoshop CC [Adobe 2014], can produce impressive results. However, the surface is covered by a high-frequency weaving pattern (sine wave), and results exhibit color deviations along slanted surfaces. A similar approach is proposed by *Voxelizer* [Zmorph 2016]. Kuiper et al. [2018] revisited this idea, generating dithering by pushing more or less flow on layers, which makes them protrude more or less on side surfaces (see Figure 3a). Dithering has to be performed differently depending on the surface slope (vertical/horizontal). With both methods, the perceived color is slightly impacted by the view angle, as occlusions play an important (see Section 5.4 of [Kuipers et al. 2018]). Instead of dithering, we rely on translucent filaments, depositing them in thin, slightly curved strata (see Figure 3b). This limits view dependency, does not require a slope-dependent algorithm and the effect extends beyond the surface, within the object volume.

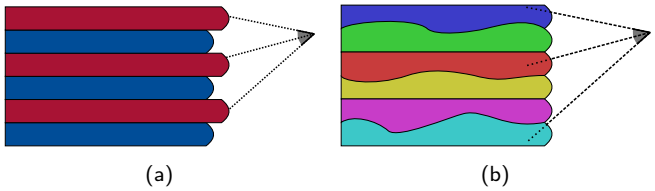


Fig. 3. **(a)** Existing methods rely on dithering along the object surface. This is achieved by having layers protrude more or less from the side, changing the perceived color ratio. **(b)** In contrast our method divides each layer into a number of strata, each having a different base color. By changing the thickness of each stratum, different ratios of the base colors are perceived.

[1] For impressive results of manual switching see https://www.thingiverse.com/thing:25612

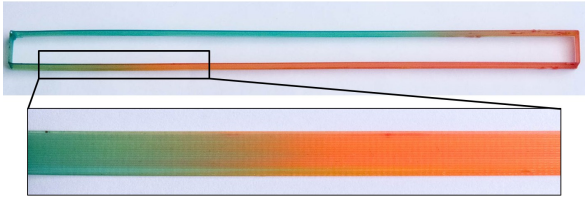


Fig. 4. This vertically extruded rectangle (170 × 10mm) is printed with the same orientation and exact same parameters at all layers, in a single continuous deposition path. The mix of filaments being pushed in the extruder is modified exactly halfway, switching from orange to green, and then from green to orange on the other side. It takes at least 57 mm for the deposited color to change to the target color. Also, note how the gradients obtained at every layer vary unpredictably.

A final approach for color mixing is to rely on multi-in-one-out nozzles [Chan 2017; Corbett 2012; Horne 2012; ORD-Solutions 2016; RepRap.me 2015], which allow to explicitly mix filaments. Filaments are pushed in the melting chamber according to a ratio. While this sounds very promising to create gradients, there is a catch: the transition between different mixtures takes a non-negligible time. Our tests revealed that while we can easily determine after which volume a stable mix is achieved, the transition time is not stable and repeatable. Therefore, these extruders cannot produce a controlled gradient, and the results remain unsatisfactory. This is clearly visible in prints using mixing extruders, as shown in Figure 4. In addition, even if the transition time was precise, it remains too long compared to what would be required to reproduce spatially varying details at the scale of a 3D printed object.

Finally, a hybrid approach consists in tinting the filament during extrusion. A low-cost approach is to use colored markers [Ajima 2014]. Company XYZ made public a prototype of a printer combining extruded filament with standard inkjet heads to produce colored 3D prints [XYZ 2017]. A special filament is required to enable the absorption of colored inks, while we seek to enable colored printing using a wide variety of available filaments as input (e.g. PLA, PET).

3 COLORED FUSED FILAMENT FABRICATION

Our approach is inspired by the way computer screens produce images. Each pixel physically corresponds to three tiny light emitting elements (red, green, blue). By varying the amount of light emitted by each, a different color is perceived. Similarly, we decompose each layer into smaller strata – a row of 'pixels' – and vary the contribution of each stratum in the final result by changing its thickness, hence producing different colors. However, contrary to a regular screen, and thanks to the use of a multi-in-one-out nozzle, we may choose the base colors of each stratum. We exploit this additional degree of freedom to improve print quality and reduce print time.

While our technique is geared toward producing colored prints, we do not tackle the color separation problem (how to find the mixing ratios to best approximate a color [Babaei et al. 2017]). Thus, we assume the user specifies colors as mixing ratios of the base filament mounted on the printer (e.g. picking them from a 3D printed palette, see Figure 15). Our goal is to provide the low-level technology that will enable full-color printing in the future. However, more work will be required to elaborate filaments and to model which colors are perceived for a given mixture. We leave these challenges as future work. Thus, in the remainder of the document, we will no longer refer to *colors*, but instead to *mixing ratios*.

3.1 Input

The input to our approach is a set of slices to be printed (toolpaths), as well as a volume field specifying mixing ratios. The toolpaths are obtained from a standard slicer, which computes deposition paths after slicing the input object into a set of layers. The volume field is a function which returns the mixing ratios producing the desired color at any point in space. This function may be a procedure, an access into a volume texture, or any other similar mechanism. In our implementation, we use a volume texture enclosing the object and let the user paint the model with a 3D brush.

When using K base filaments ($K > 1$) we denote a mixing ratio as a vector of K weights $c \in \mathbb{R}^K$ such that $\sum_{i=1}^{K} c_i = 1$ and $c_i \geqslant 0$. This ensures the input ratios can be physically reproduced by mixing the base filaments on the target printer. Note that this is a requirement: we do not support converting mixing ratios for e.g. a five filament printer into mixing ratios for a printer having fewer filaments.

3.2 Output

The output of our technique is a G-code file for a printer equipped with a multi-in-one-out nozzle. For the sake of clarity, let us assume a nozzle with three input filaments (we use both three and five filament printers). Extrusion is controlled through modified G-code commands that allow specifying the ratios of each filament in the mix, e.g. G-code command `G1 X10.0 Y12.0 Z3.0 E20.5 A0.2 B0.3 C0.5` moves the nozzle to position (x=10.0mm, y=12.0mm, z=3.0mm) and extrudes filament up to length 20.5 mm, using ratios of 20%, 30% and 50% for the filaments in extruders A, B and C. The ratios have to be positive and have to sum to one.

The strata are slightly curved to produce the mixing ratio variations. Thus, along a toolpath we vary the z coordinate by small amounts, and we adjust the flow accordingly – by changing motion speed – to maintain a constant track width. Each stratum is a sequence of G-code instructions with varying Z coordinates and E values (amount of material), where the variations are given by the mixing ratios and the shape of the stratum below, see Figure 1, right. The strata, taken together, reproduce the shape of the original layer.

3.3 A first approach and its limitations

Given the idea of using strata, a first approach would be as follows. Each base color would be mapped to one stratum, following an arbitrary fixed order (e.g. red first, then blue, then green). Then, strata would be printed in sequence one on

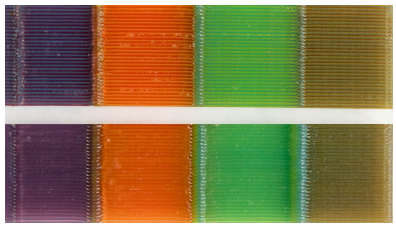


Fig. 5. **Top:** Using pure filaments as base mixtures produces visible stripes. **Bottom:** A better choice of strata mixtures makes stripes barely noticeable.

top of another, following the geometry of the original toolpath, but with slight height variations. This ideally creates curved strata that produce the desired effect.

Unfortunately, this simple approach does not produce satisfactory results and suffers from three issues: *stripes*, *interferences*, and *non-scalability*.

Stripes (Figure 5). The typical layer thickness we use is $300\mu m$. Thus, the strata are relatively thick, and the stripe pattern produced by interleaving the base colors may become visible in the final result. This is especially the case when mixing two colors with large ratios (e.g. a 50% blend of two base colors).

Interferences (Figure 8). Undesired colors may appear between layers. This is akin to the color fringing artifacts around white text on computer screens. The reason is that the color is perceived as the average of surrounding strata, regardless of layer boundaries. Thus, the last stratum of layer i will perceptually mix with the first stratum of layer $i+1$, possibly resulting in a spurious color.

Non-scalability. Finally, the simple approach described here always requires one stratum per base filament. Thus, each additional source filament requires an additional stratum: the method does not scale with the number of source filaments. The number of stratum should be kept low. Each added stratum increases print time: a purge sequence is required to stabilize the new mixture in the nozzle, and then the stratum has to be printed. In addition, using many strata requires careful calibration as average deposition thickness decreases. We successfully print using five strata on 0.4 mm layers (see Figure 21), but going beyond seems unreasonable.

To address all three issues, we propose to optimize both the number of stratum per layer and their mixtures, using the ability of the mixing extruder to reliably produce *constant* mixes of base filaments. This is based on the observation that the perceived color is similar whether the mixture is done within the nozzle or through strata of varying heights. As most objects exhibit a strong spatial coherency within a layer, it is rare for a painted object to contain all possible colors [1]. This implies that we can optimize a choice of mixtures that will 1) minimize the stripes by using base mixtures closer to that used within the layer and 2) decouple the number of strata from the number of base filaments.

[1]This fact is at the root of many texture compression algorithms, such as DXTC and S3TC.

3.4 Optimization of strata base mixtures

We now detail our optimization process. The objective is to compute the base mixture used for each stratum as well as reduce the number of strata, exploiting the mixing extruder as much as possible. Our formulation allows for arbitrary numbers of filaments, but in practice our most complex printer takes only five. At most, using K filaments would require printing K strata per layer. Our optimization reduces this number automatically, when possible.

We formulate the optimization in two steps. First, we optimize the number of strata and their base mixing ratios (the ones used with the nozzle, when printing each stratum). Second, we compute an ordering of the strata within each layer to obtain the final result.

The optimizer does not change the quantity of each filament deposited at any given location in the print: it simply makes a compromise between mixing in the nozzle or mixing using curved strata.

3.4.1 Optimizing number of strata and their base mixtures. This optimization step is performed independently for each layer. We, therefore, consider a single layer in the following.

Before processing, we re-sample the layer toolpaths: we increase the number of vertices to properly capture variations of the input field. The sampling ratio is constant.

We denote the mixing ratio associated with each toolpath vertex by $c \in \mathbb{R}^K$ such that $\sum_{i=1}^{K} c_i = 1$ and $c_i \geqslant 0$. Note that they are akin to barycentric coordinates.

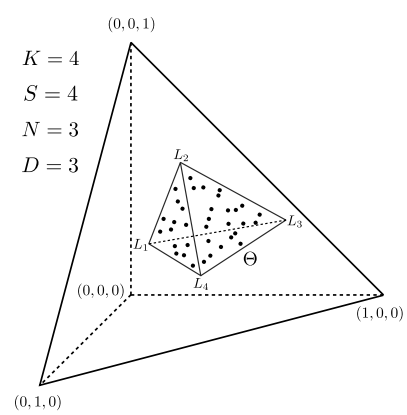


Fig. 6. Four filaments case. The mixing ratios are embedded into 3D using unit vectors and the origin, forming the set $\mathcal{C}$ (black dots). Our optimizer finds a simplex Θ (here tetrahedra) enclosing $\mathcal{C}$ while having a small volume. The vertices of the simplex (L_1, L_2, L_3, L_4) are the new strata base mixtures. Since they are closer to the enclosed mixing ratios, the stripe effect is reduced.

Our objective is to determine the number of required strata S for the layer ($1 \leqslant S \leqslant K$), as well as the base mixtures of each stratum. We denote each of the S base mixture by

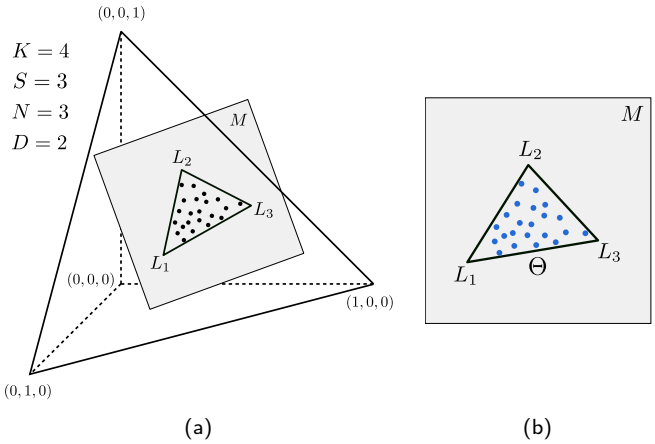


Fig. 7. Four filaments case. (7a) Here, the mixing ratios $\mathcal{C}$ lie on a lower dimensional subspace. (7b) Our algorithm first determines the intrinsic dimensionality of the data (2D) and computes a dimensionality reduction to obtain the set $M \otimes \mathcal{C}$ (in blue). It then optimizes for a simplex in the lower dimension space (here, a triangle). Only three strata will be used in this case to accurately reproduce the input mixtures.

$L^i \in \mathbb{R}^K$, with $1 \leqslant i \leqslant S$. These mixtures follow barycentric constraints: $\sum_{j=1}^{K} L^i_j = 1$ and $L^i_j \geqslant 0$. Ideally, we would like the L^i to be close to the input ratios, such that the stripe effect is minimized.

After optimization, each input mixing ratio c will be converted into a new mixing ratio α (also under barycentric constraints). The L^i are used as mixing ratios with the nozzle, while the α are used to change the thicknesses of the strata during printing.

A key observation, implied by the barycentric constraints, is that all the input ratios c lie within a simplex defined by the L_i. Therefore, optimizing for the L_i amounts to finding a simplex Θ enclosing all input points, and such that its vertices follow barycentric constraints. This is illustrated in Figure 6. To minimize the stripe defect we minimize the volume of Θ, which ensures the L_i are close to the points they enclose.

Since the mixing ratios represent barycentric coordinates, we embed them into a space of dimension $N = K - 1$ ($N > 0$) using the origin and unit vectors. The loss of one dimension is expected, as for instance when using two filaments a single number is enough to represent the mixture. We denote by $\mathcal{C} \subset \mathbb{R}^N$ the set of all embedded mixing ratios for the layer.

The number of required strata is directly impacted by the intrinsic dimensionality of $\mathcal{C}$. For instance, if all points lie on a line, only two strata will suffice regardless of the value of K: all points will be linear combination of two base mixtures. Therefore, if possible, we seek to compute a dimensionality reduction of $\mathcal{C}$. Let us denote by M and M^{-1} the dimensionality reduction operator and its inverse, as well as $D \leqslant N$ the intrinsic dimensionality of the data.

In dimension D, we search for the smallest simplex Θ enclosing the dimensionally reduced points $M \otimes \mathcal{C}$, ensuring the simplex vertices mapped back through M^{-1} (the L_i) follow barycentric constraints. This is illustrated in Figure 7.

Solver. For dimensionality reduction (M and M^{-1}), we rely on Principal Component Analysis (PCA). This provides us with a transformation (defined by the normalized eigen vectors) aligning the first coordinates with the dimensions having the largest variance. We select the number of dimensions using a threshold ϵ on the eigen values (variance). We set $\epsilon = 10^{-4}$ – it is defined in ratio space – to keep the approximation error low.

Minimizing the volume of Θ corresponds to a known, hard combinatorial optimization problem [Hendrix et al. 2013]. A known property of the optimal convex hull Θ is that each of its N-dimensional facets must touch the convex hull of $\mathcal{C}$. This property was used in [Zhou and Suri 2002] to obtain an algorithm enumerating all feasible contacts for $N = 3$.

We follow a similar idea to propose an approximate algorithm that extends to any dimension. Our algorithm takes into account only one type of contact: we enumerate all combinations of $N + 1$ hyperplanes of the convex hull of $\mathcal{C}$ that enclose a finite volume. We compute the volume of each valid combination – that is those which vertices enforce barycentric constraints through M^{-1} – and return the one with minimal volume. To help finding solutions we allow a small tolerance when checking for barycentric constraints, using a threshold $\lambda = 10^{-2}$ on all checks (sum to one and positivity). In case no solution is found, we increase D. At worst, we end with $D = N$. If still no solution is found, we use the trivial solution of setting L_i to be unit vectors. This process is fast. On average, the number of convex hull hyperplanes grows slowly with respect to the number of points $|\mathcal{C}|$. For instance, for a random distribution of points inside a polytope, the average number of hyperplanes is $\Theta(\log^{N-1} |\mathcal{C}|)$ [Dwyer 1988]. Optionally, we could randomize the enumeration and limit the maximum amount of tests to improve efficiency. We use QHull [Barber et al. 1996] for computing the initial convex hull in N-dimensions.

3.4.2 Optimizing the ordering of the strata. After optimizing for the strata mixtures we need to determine their ordering in the final print. Some orderings are better than others: an ordering where strata with similar mixtures in successive layers end up side by side will likely produce a visible interference (see Figure 8). Instead, we encourage a sequence of as-different-as-possible strata, to obtain an interleaving pattern akin to LED screens subpixels (RGB,RGB,...). As the number of strata and their mixtures change with each layer, we rely on a combinatorial optimization process.

Our objective function evaluates whether similar mixtures, across layers, end up close to each other in the final result. Since mixtures are used in varying amounts along and within the object, we take the mixture volume into account: a mixture that is used only in a few places will have a limited influence on the ordering.

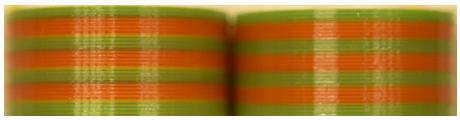
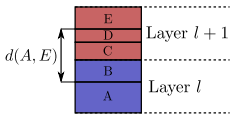


Fig. 8. **Left:** Some ordering produce visible interferences between neighboring layers. **Right:** Reordering the strata removes these defects.

We denote L_i^l the i-th base mixture of layer l, and denote by $d(A, B)$ the *stack distance* between two base mixtures. Considering an imaginary stack of all the ordered base mixtures, where the volume is used as height, the stack distance is defined as the distance between the top of A and the bottom of B (see inset). Then, a current ordering is scored as:



$$\sum_{i,j} \left(\frac{\|L_j^l - L_i^{l+1}\|}{1 + d(L_j^l, L_i^{l+1})} \right)$$

where i, j pairs are pairs of indices for respectively strata in layer $l+1$ and l. We seek an ordering with a maximal score. The principle is that pairs which have similar mixtures (numerator) produce a little score, while pairs with dissimilar mixtures produce an increasing score when further apart in the stack (denominator). The volume is taken into account through the stack distance.

Our approach is a simple bottom-up sweep, determining the ordering of layer l assuming the ordering of layer $l-1$ is known. At the current layer l we enumerate all possible permutations of strata (there are five strata maximum in our implementation), and select the one maximizing the objective. The first layer is initialized with a random ordering.

4 IMPLEMENTATION

We implemented our approach in a custom slicer. After generating the toolpaths, we optimize for the per-layer strata, and generate new path planning instructions. Table 1 gives processing times for our main results ; all complete in less than four seconds.

We produce the G-code by outputting instructions for each layer as many times as there are strata, adjusting the height and the flow along the paths. The height at each vertex is directly obtained from the barycentric coordinates α. We then simplify the geometry of the obtained curved paths to reduce the number of vertices. Note that when the height of a path vanishes, we interrupt deposition and proceed to the next path.

An important factor in achieving successful prints while using several strata and thin layers is to implement what is known as *linear advance*[1]. This is a mechanism to deal with internal pressure within the nozzle, that stops pushing plastic before the end of a path, such that pressure is reduced when travel motions start.

[1] http://marlinfw.org/docs/features/lin_advance.html

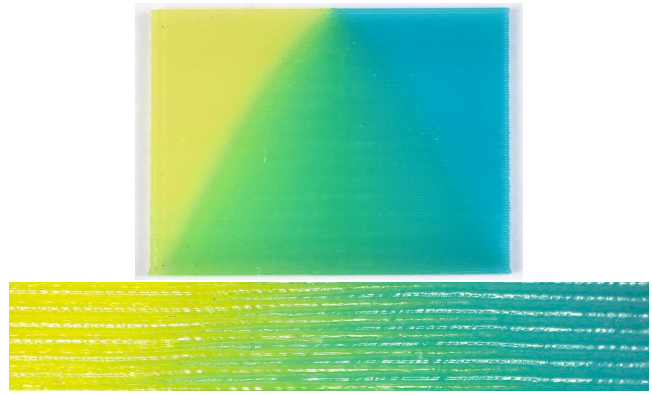


Fig. 9. Gradient of varying sharpness, from smooth (bottom) to sharp (top). Our approach affords for a progressive transition. The bottom microscope close-up of the gradient tip reveals the interleaved strata. Please refer to the text for details regarding the small discontinuity on the left/right sides.

Before each stratum, we print an auxiliary structure that allows the new filament mix to stabilize. We chose to use an ooze shield structure [Hornus et al. 2016; Ultimaker 2016] as it remains close to the print. We adjust print speed to maintain a constant extrusion speed, which avoids abrupt pressure changes in the nozzle [Kuipers et al. 2018].

5 RESULTS

To evaluate our results, we built several custom printers using a Prusa i3 kit and a Diamond extruder (three and five filaments). One of the printers is shown in Figure 1, left. We perform all computations on an Intel Core i7, CPU 4.0 GHz, with 32 GB memory. Unless otherwise specified we use a layer thickness of 0.3 mm.

Gradients. Figure 9 shows a part with a gradient that starts smooth at the bottom and becomes progressively sharper until becoming an edge. As can be seen, our method successfully copes with the transition. To the best of our knowledge, no other method can achieve similarly smooth and progressive results on filament printers.

A small discontinuity can be seen just before transitioning to a pure color, on the left and right sides. This is due to deposition interruption when height vanishes, explained in Section 4. A very careful calibration of linear advance could reduce this effect further.

Impact of the surface slope. The strata are printed one on top of another. While this works well along vertical surfaces, one concern is that on near-flat surfaces the last printed stratum might hide those below, biasing the perceived colors [Kuipers et al. 2018; Reiner et al. 2014]. As mentioned in Section 2 we rely on slightly translucent filaments to avoid view dependency. A positive side effect is that this also alleviates the visibility issue on top layers, as can be see in

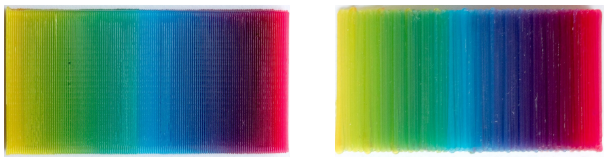


Fig. 10. Grading from yellow, cyan, to magenta. **Left:** Vertical print with gradient along the Z build direction. **Right:** Print with 27° slope, gradient along X. Thanks to translucency, the slope has little impact on the result.

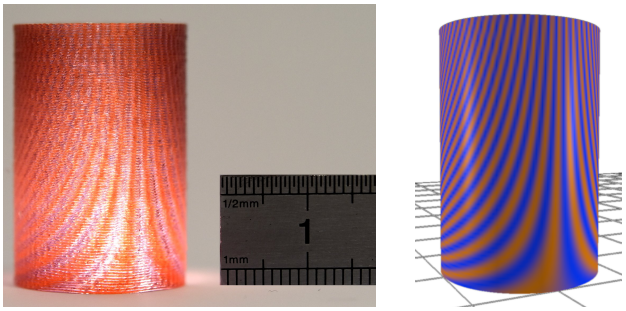


Fig. 11. Testing achievable detail frequencies. This pattern is produced by a sine wave around the cylinder that has increasing frequency towards the top. The nozzle diameter is 0.4 mm. The smallest features being properly captured are around 0.6 mm.

Figure 10, right.

Using translucent filaments however introduces a drawback: a thicker colored shell has to be printed to hide the object interior, and the final result remains slightly translucent. In our experience, most users actually find this effect pleasing and qualitative compared to opaque plastic prints.

Frequency. Figure 11 shows a cylinder textured with a pattern of increasing frequency (a sine wave). As can be seen, high frequencies can be reproduced – the main limitations stemming from the nozzle deposition diameter (0.4 mm in our setup). In this print, the thinnest visible stripes – before aliasing occurs – have a width of approximatively 0.6 mm.

Colored prints. We painted and 3D printed a variety of objects using our approach using PLA filament. Unless otherwise specified prints use three source filaments and 0.3 mm layers.

When considering the results, please keep in mind that there is no calibration between the colors displayed in our paint tool and the colors on the print: we faithfully reproduce the *mixing ratios* selected by the user but have currently no way to display matching colors on the screen. In the results, however, regions of smooth gradients and sharp transitions should match accurately between the paint and the print. The filaments used vary between prints, therefore the obtained colors differ.

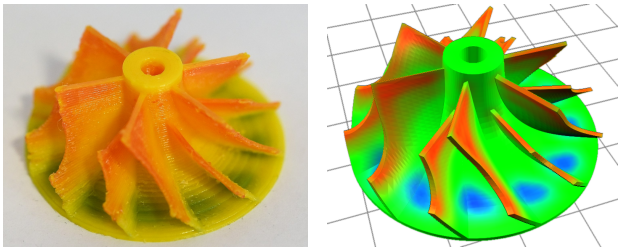


Fig. 12. 3D printed turbine (Thing: 398841 by bob345). **Left:** 3D painted model in our tool. **Right:** photographs of the 3D printed model with a matching viewpoint.

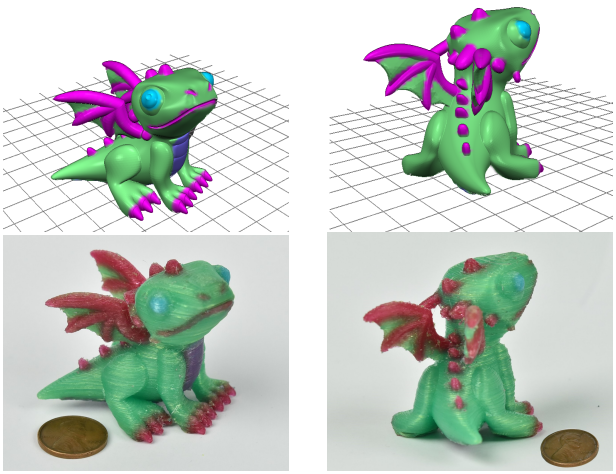


Fig. 13. 3D printed dragon (Thing: 1624412). **Top:** two views of the 3D painted model in our tool. **Bottom:** photographs of the 3D printed model using our technique with matching viewpoints.

The bird in Figure 1 shows how a variety of colors can be used within the same print, from pure, clear colors to more subtle mixes. Gradients and sharp transitions are properly captured. Note how the top of the bird head remains properly colored despite the change in slope. The turbine in Figure 12 illustrates how our technique can be used for educational and prototyping/engineering purposes, visualizing a fluid pressure field directly on a 3D printed prototype.

The dragon in Figure 13 reveals how printed colors closely follow the layout shown on the virtual model, with both gradients and sharp transitions faithfully reproduced. Figure 14 shows the prime shield used for transitioning between strata mixes.

The fish in Figure 15 is a case of near-flat printing. As can be seen, there is no perceivable difference in quality between vertical and quasi-flat surfaces.

The chameleon in Figure 16 has large regions of gradients and uses a variety of colors. The gradients are smooth and match the painted field closely, for example along the body

Fig. 14. 3D printed dragon with its supports and its priming shield.

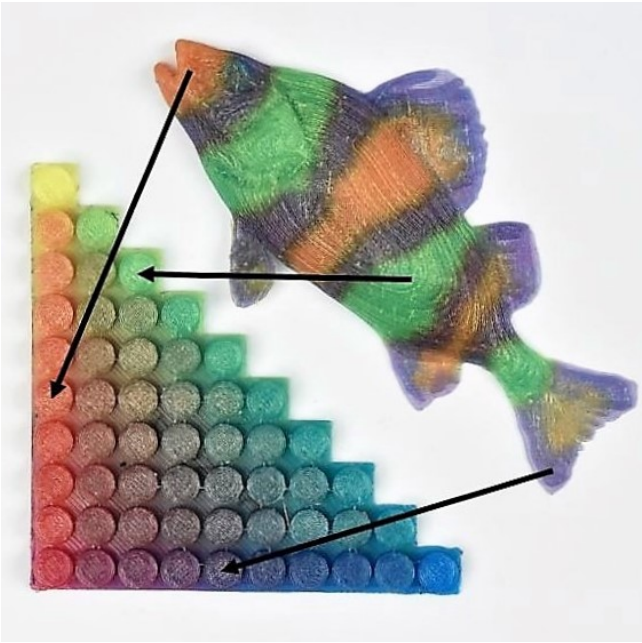


Fig. 15. 3D printed fish and palette used to select colors (Thing: 1701871).

around the legs.

The vases in Figure 17 are examples of having both global gradients (background) and detailed patterns (stripes and dots). Both patterns and gradients are well reproduced. In these cases the mixture fields are produced from procedural functions.

In Figure 18 we use an actual picture as a source for mixing ratios. Without color separation and tightly controlled filament pigmentation, we cannot currently achieve the ultimate goal of printing an accurate reproduction, or even something close to it. Instead, the user chose to create a stylized image. The resulting 3D printed lithophane exhibits little artifacts and a rich, colorfull appearance.

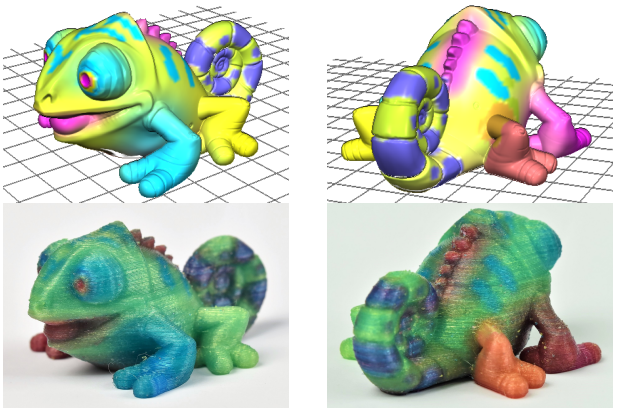


Fig. 16. 3D printed chameleon (Thing: 2303679). **Top:** two views of the 3D painted model in our tool. **Bottom:** photographs of the 3D printed model using our technique with matching viewpoints. The slight visual striping is due to varying highlights (zoom in to see the effect).

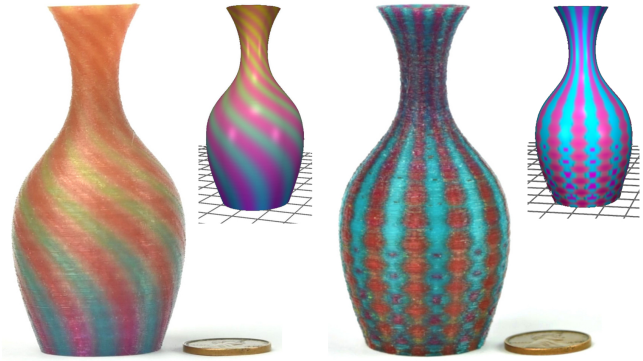


Fig. 17. 3D printed vases (Thing: 763726). Pictures of the print and 3D views in our paint tool in the insets, with matching viewpoints.

Benefits of optimization. Table 1 summarizes timings for our main results. As can be seen, the optimization reduces the number of strata, and hence the print time. The benefits, however, depend on the complexity of the mixing field. For instance on the bird model, the reduction is modest, while it is larger on the turbine model. This is explained by the fact that many turbine layers can be captured with only two strata.

Even when optimizing does not impact print time, it does impact quality by reducing the stripe defects. Figure 22 shows a rendering of the strata obtained with and without the optimizer. The number of strata goes down from 330 to 267. After optimization the mixes of the strata closely match those used in the layers, whereas without the stripes of the base filaments are obvious.

Towards full color printing. We tested our approach

Fig. 18. A low-resolution lion image is used to drive a mixing field and produce a colorfull lithophane (printed flat and backlit).

Model	Mixtures	Ordering	Est. print time w/ (w/o) opt.	# Layers	# Strata w/ (w/o) opt.
Dragon	1.9s	6 ms	9h20m (11h52m)	140	348 (423)
Chameleon	2.8s	10 ms	9h17m (10h11m)	130	333 (394)
Police car	2.0s	2 ms	1h56m (2h19m)	40	123 (164)
Turbine	0.81s	1 ms	2h29m (2h32m)	57	124 (174)
Bird	3.6s	5 ms	6h30m (6h48m)	239	651 (720)

Table 1. This table shows, for our main results: the time to optimize for per-strata mixtures, the time to optimize for the ordering of strata, the estimated print time for the optimized and unoptimized versions (in parenthesis), the number of layers, the total number of strata for the optimized and unoptimized versions (in parenthesis). Note that the estimated print time is underestimating the actual print time as it ignores acceleration effects.

using four filaments, in particular red, green, blue and white. All the prints using four or five filaments use a layer thickness of 0.4 mm.

Please note that these tests are meant to demonstrate the future potential of the technique – accurate color reproduction will require solving additional challenges, namely the color separation problem and the design of filaments with precisely controlled pigmentations. We are currently limited to filaments available from resellers, and thus cannot precisely adjust the base colors.

Despite these issues, using four filaments we can significantly increase the range of colors, as demonstrated in Figure 19.

Figure 20 shows a car printed using the same four filaments. The gray wheels are obtained by mixing red, green, blue, while the white is used on the sides and top.

Finally, we tested our technique with up to five filaments, to demonstrate the feasibility of having five strata per layer, when required. This is shown in Figure 21.

6 DISCUSSION, LIMITATIONS AND FUTURE WORK

We now discuss some specific aspects of our technique, limitations, and remaining challenges.

Print time. One obvious drawback of our technique is that it increases print time significantly, compared to a standard print. At most we visit each layer once per strata, thus one can expect a proportional increase in print time.

In practice, this is mitigated by two factors. First, many strata take a zero thickness in large areas, allowing for fast forwarding to the next toolpaths. Second, our optimizer reduces the number of strata significantly (see Figure 22). At the extreme, if a model is painted with a constant mixture, the print time remains unchanged (a single stratum per layer suffices).

Print quality and reliability.

As can be seen throughout our Figures, our prints are good but not perfect. Beyond the traditional defects of filament printing (z-scars, stringing), the mixture quality also suffers from oozing and smears due to the nozzle catching some material on the stratum below. These defects produce slight striations on some prints (e.g. the left vase in Figure 17), as well as spurious material deposition (e.g. along the edges of the discs in Figure 19).

The choice of filament and the printer calibration have a direct impact on these defects. The filament we used here is quite shiny and tends to emphasize the defects (see Figure 16). Our printers are custom assembled from inexpensive off-the-shelves components (each printer costs around $600), and while we carefully set them up they have limited mechanical accuracy.

Printing using our technique requires careful calibration and setup. The parameters have to be fine-tuned, in particular those impacting oozing: linear advance, retraction settings, print temperature. We provide all setting and scripts for our slicing software on https://icesl.loria.fr/pages/features/color-mixing/.

Mixing field complexity.

There are limitations to the complexity of the mixture

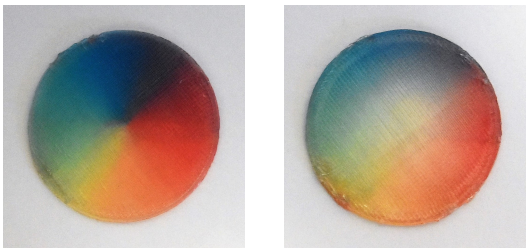


Fig. 19. 3D printed color discs. The left disc uses only red, green, blue mapped onto the disc radially, showing varying hues. The right disc adds white in the center, progressively changing saturation.

Fig. 20. A tiny police car printed on a four filaments printer (Thing: 1587558 by Vorpalia).

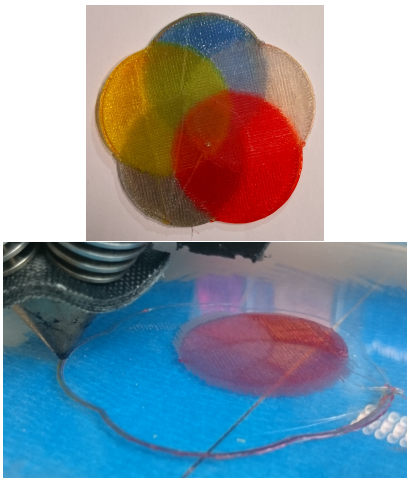


Fig. 21. Five discs using each one base filament intersect in this model. This demonstrates the feasibility of using five strata with 0.4 mm layers. The right image shows the first stratum of the first layer just after printing. The different areas corresponding to different ratios of red can be seen.

fields that can be reproduced. Some fields may produce many isolated, tiny paths which are difficult to print reliably. A dedicated filtering approach would be required to remove (or enlarge) these.

The mixing field may also contain regions which each use a different subspace of mixtures. Our current algorithm will attempt a tradeoff across each entire layer. An alternative would be to first cluster mixture ratios, and then optimize each cluster independently for base mixtures.

Layer thickness. Our technique is easier to use when layers are thicker – essentially making printing more robust to calibration errors. We use layer thicknesses of 0.3 and 0.4 mm. Using thicker layers would be possible, but we are limited by the nozzle output diameter (0.4 mm ; it is generally advised to use layer thicknesses below the nozzle diameter). In principle, our technique works with adaptive layer thicknesses, but we have yet to experiment with this. Of course, the range of adaptivity is limited by the thinnest achievable layers.

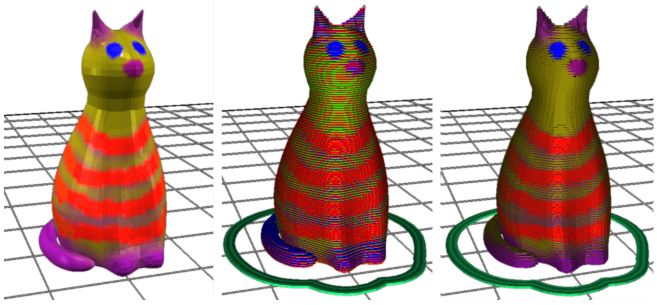


Fig. 22. Painted cat using three filaments (Thing: 24255 by Mere). **Left:** 3D painted model. **Middle:** Strata obtained without any optimization. **Right:** Strata obtained with the optimizer.

Nozzle shape and up/down z motions. Abrupt changes in Z (height) coordinates while printing paths can produce defects: the flat area at the nozzle tip interferes with already printed paths around. Song et al. [2017] address this issue for printing top surfaces with better accuracy, splitting and re-ordering paths to minimize interference. However, our case is less sensitive to this issue. While the first strata are indeed curved, the last stratum always exactly aligns with the (flat) layer top boundary. As a consequence, the last stratum does not suffer from nozzle interferences (see Figure 1, rightmost). While there may be small geometric defects in between strata, we found that in practice they have no impact on the final perceived quality, especially as the correct amount of (translucent) colored filament was deposited.

Filament availability.

The results of our technique depend on the source filament quality. In particular, it is important for the filament pigmentation to be well balanced. Otherwise, even tiny amounts of a filament with high pigment concentration will immediately overrule the other ones.

We found it very difficult to source well suited filament from existing vendors. We now produce our own filament using an off-the-shelf extruding machine designed for producing 3D printer filaments. We mix pigments with PLA pellets prior to feeding them to the machine. We use cyan, magenta and yellow pigments. We provide more details on https://icesl.loria.fr/pages/features/color-mixing/.

7 CONCLUSION

We introduced a technique for colored printing with controlled gradients using readily available low-cost printer components and filaments. Beyond the initial idea of mixing filaments through the use of per-layer strata, we propose an optimization of the mixture of each stratum to improve the quality and reduce print time, exploiting a mixing nozzle. One important advantage of our technique is that it keeps using standard materials (PVA, ABS, PET, etc.) and does not impose one particular type of filament.

As the technologies evolve we believe our approach will remain useful: by using constant color strata we improve deposition continuity. With more precise hardware and controllers, oozing could be reduced and higher quality deposition would be achieved. An interesting question is whether our approach could be used to grade properties beyond colors, such as elasticity using flexible filaments.

ACKNOWLEDGMENTS

This work was funded by ERC ShapeForge (StG-2012-307877) and received support from Région Lorraine and FEDER. We thank Cédric Zanni for his help with preparing colored prints, as well as Sandrine Hoppe (LRGP) for discussions regarding filaments and materials.

REFERENCES

Adobe. 2014. Photoshop CC. http://www.adobe.com/products/photoshop.html.

Josh Ajima. 2014. Rainbow Extrusion: Coloring 3D Printer Filament. http://makezine.com/projects/make-42/rainbow-extrusion-coloring-3d-printer-filament/.

Thomas Auzinger, Wolfgang Heidrich, and Bernd Bickel. 2018. Computational Design of Nanostructural Color for Additive Manufacturing. *ACM Trans. Graph.* 37, 4 (2018), 159:1–159:16. https://doi.org/10.1145/3197517.3201376

Vahid Babaei, Kiril Vidimče, Michael Foshey, Alexandre Kaspar, Piotr Didyk, and Wojciech Matusik. 2017. Color Contoning for 3D Printing. *ACM Transactions on Graphics* 36, 4, Article 124 (July 2017), 15 pages.

C. Bradford Barber, David P. Dobkin, David P. Dobkin, and Hannu Huhdanpaa. 1996. The Quickhull Algorithm for Convex Hulls. *ACM Trans. Math. Softw.* 22, 4 (Dec. 1996), 469–483. https://doi.org/10.1145/235815.235821

Alan Brunton, Can Ates Arikan, Tejas Madan Tanksale, and Philipp Urban. 2018. 3D Printing Spatially Varying Color and Translucency. *ACM Trans. Graph.* 37, 4 (2018), 157:1–157:13. https://doi.org/10.1145/3197517.3201349

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* 35 (2015), 4:1–4:13.

Felix Chan. 2017. NIX color printer. https://www.kickstarter.com/projects/517014753/the-world-first-high-resolution-full-color-3d-prin?lang=fr.

W Cho, EM Sachs, NM Patrikalakis, MJ Cima, TR Jackson, H Liu, J Serdy, CC Stratton, H Wu, and R Resnick. 2001. Methods for Distributed Design and Fabrication of Parts With Local Composition Control. In *Proceedings of the 2001 NSF Design and Manufacturing Grantees Conference*.

Paolo Cignoni, Enrico Gobbetti, Ruggero Pintus, and Roberto Scopigno. 2008. Color Enhancement for Rapid Prototyping. In *The 9th International Symposium on VAST International Symposium on Virtual Reality, Archaeology and Cultural Heritage*. Eurographics, 9–16. http://vcg.isti.cnr.it/Publications/2008/CGPS08

James Corbett. 2012. Reprap colour mixing project. Final Year MEng Project, Department of Mechanical Engineering, Faculty of Engineering and Design, University of Bath..

Rex Allen Dwyer. 1988. *Average-case Analysis of Algorithms for Convex Hulls and Voronoi Diagrams*. Ph.D. Dissertation.

Oskar Elek, Denis Sumin, Ran Zhang, Tim Weyrich, Karol Myszkowski, Bernd Bickel, Alexander Wilkie, and Jaroslav Křivánek. 2017. Scattering-aware Texture Reproduction for 3D Printing. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2017)* 36, 6 (Nov. 2017), 241:1–241:15. https://doi.org/10.1145/3130800.3130890

E. M. T. Hendrix, I. García, J. Plaza, and A. Plaza. 2013. On the minimum volume simplex enclosure problem for estimating a linear mixing model. *Journal of Global Optimization* 56, 3 (2013), 957–970.

Jean Hergel and Sylvain Lefebvre. 2014. Clean Color: Improving Multi-filament 3D Prints. *Computer Graphics Forum* 33, 2 (2014), 469–478.

Richard Horne. 2012. 3-way Quick-fit Extruder and Colour Blending Nozzle. http://richrap.blogspot.fr/2012/08/3-way-quick-fit-extruder-and-colour.html.

Samuel Hornus, Sylvain Lefebvre, Jérémie Dumas, and Frédéric Claux. 2016. Tight Printable Enclosures and Support Structures for Additive Manufacturing. In *Eurographics Workshop on Graphics for Digital Fabrication*, A. Medeiros e Sa, N. Pietroni, and K. Rodriguez Echavarria (Eds.). The Eurographics Association. https://doi.org/10.2312/gdf.20161074

Tim Kuipers, Willemijn Elkhuizen, Jouke Verlinden, and Eugeni Doubrovski. 2018. Hatching for 3D prints: Line-based halftoning for dual extrusion fused deposition modeling. *Computers & Graphics* 74 (2018), 23 – 32. https://doi.org/10.1016/j.cag.2018.04.006

Marco Livesu, Stefano Ellero, Jonàs Martínez, Sylvain Lefebvre, and Marco Attene. 2017. From 3D Models to 3D Prints: An Overview of the Processing Pipeline. *Computer Graphics Forum* 36, 2 (2017), 537–564. https://doi.org/org:443/handle/10.1111/cgf13147

Mcor. 2005. Mcor technologies. http://mcortechnologies.com/.

Mosaic. 2015. Palette. https://www.kickstarter.com/projects/mosaic3d/the-palette-3d-printing-evolved.

ORD-Solutions. 2016. RoVa4D Full Color Blender 3D Printer. http://www.ordsolutions.com/rova4d-full-color-blender-3d-printer-pre-order/.

PolyJet. 1998. PolyJet Technology. https://en.wikipedia.org/wiki/Objet_Geometries.

Builder 3D Printers. 2017. Builder. http://builder3dprinters.com/color-mixing/.

Joseph Prusa. 2016. Prusa i3 MK2 multi-material upgrade. http://www.prusaprinters.org/original-prusa-i3-mk2-multi-material-upgrade-release/.

Tim Reiner, Nathan Carr, Radomír Měch, Ondřej Št'ava, 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.

RepRap.me. 2015. Diamond nozzle. http://reprap.org/wiki/Diamond_Hotend.

E.M. Sachs, J.S. Haggerty, M.J. Cima, and P.A. Williams. 1994. Three-dimensional printing techniques. https://www.google.com/patents/US5340656 US Patent 5,340,656.

Eric Sammut. 2016. The Prometheus System. https://www.kickstarter.com/projects/811909269/the-prometheus-system-intuitive-multi-filament-3d.

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

Hai-Chuan Song, Nicolas Ray, Dmitry Sokolov, and Sylvain Lefebvre. 2017. Anti-aliasing for fused filament deposition. *Computer-Aided Design* 89 (2017), 25 – 34.

Ultimaker. 2016. Cura Software, ooze shield. https://ultimaker.com/en/products/cura-software.

H Wu, EM Sachs, NM Patrikalakis, D Brancazio, J Serdy, and TR Jackson. 2000. Distributed Design and Fabrication of Parts With Local Composition Control. In *Proceedings of the 2000 NSF Design and Manufacturing Grantees Conference*.

XYZ. 2017. da Vinci Color printer. https://www.xyzprinting.com/.

Chi Zhou, Yong Chen, ZG Yang, and Behrokh Khoshnevis. 2011. Development of multi-material mask-image-projection-based stereolithography for the fabrication of digital materials. In *Annual solid freeform fabrication symposium*.

Yunhong Zhou and Subhash Suri. 2002. Algorithms for a minimum volume enclosing simplex in three dimensions. *SIAM J. Comput.* 31, 5 (2002), 1339–1357.

Zmorph. 2016. Voxelizer. https://support.zmorph3d.com/hc/en-us/categories/200909295-Voxelizer.