HANS3D: A Multi-Material, Volumetric, Voxel-By-Voxel Content Processing Pipeline for Color and Beyond

Peter Morovič, Ján Morovič, Ingeborg Tastl, Melanie Gottwals, Gary Dispoto, HP Inc., Spain

Abstract

3D printing enables the production of objects that combine materials in hitherto infeasible ways, with the potential of varying their use with a granularity of up to individual print-resolution voxels. By combining printing materials, such as powders, fluid agents, etc., in volumetrically different ways, a limited number of materials can be used to yield a great variety of object properties, such as stiffness, strength, density, translucency and color. A key challenge is to ensure that an input object with volumetrically-specified properties is appropriately transformed to yield per-voxel recipes for how to combine a printing system's materials. To ensure that such a transformation is performed in a way that maintains volumetric control with up to voxel precision, the present paper introduces a content processing pipeline that has individual printed voxel contents as its basic building blocks and that controls their placement in a natively volumetric domain. This pipeline is an evolution of the HANS print control paradigm [1]. First results, applied to control color of 3D printed objects are reported.

Introduction

Additive manufacturing or 3D printing is in the process of undergoing a revolution as a result of the emergence of new 3D printing technologies that promise control over a multitude of object properties at an unprecedented level of granularity and precision. The technology to enable this is driven both by the use of an ever growing set of materials (plastic, metal or ceramic powders, modifying liquids or inks, depositing and fusing technologies etc.), as well as by advances in the pipelines that control such systems.

An ambitious vision is that properties ranging from perceptual (e.g., color [2], sub-surface scattering [3], transparency), via mechanical (e.g., stress and strain or density [4]) and optical [5] through to biological [6] and chemical [7] will be variable within a single object and with print-resolution voxel accuracy, thanks to printing systems that combine a multitude of materials [8]. Instead of the resulting objects simply inheriting the properties of their constituent materials, the ability to control their use and spatial arrangement, sometimes referred to as Local Composition Control [9], results in "metamaterials" [10].

There are several challenges here, starting with the creation of 3D models where multiple properties are assigned to parts either in manual or computational ways [11]. The result is a 3D object that potentially specifies different properties at each location within its volume. The realization of those properties relies on several inter-related components, including the materials (e.g., powders, agents, fluids, filaments, etc.), the dispensing system and the software pipeline that receives volumetric multi-property content and transforms it into print-ready halftone data where it is their interplay that determines resulting printed object properties, instead of the materials alone.

In the context of controlling color in 3D printing, previous approaches have applied workflows from 2D printing, e.g., using the International Color Consortium's color profiles for mapping a color input to a 3D printer's device color space, e.g., RGB or CMYK [12-14]. More focus has been placed on subsequent halftoning, with the desire to extend 2D approaches to the needs of 3D printing, e.g., inter-layer connectivity [15, 16]. To enforce the constraint of using a single material per voxel in a multi-material printer, Brunton et al. [17] have developed a "tie-breaking" scheme, where each material is first independently error-diffused and the material that has the highest tie-breaker error is placed at a voxel if more than one material would have had to be placed there. Instead of halftoning data in a printing system's agent or colorant channels, as is customary in 2D printing, Sun and Sie [18] have proposed error diffusion halftoning in an input RGB color space that discretizes continuous-tone RGB slice images into the eight vertices of the RGB cube (or only to the RGBKW vertices), before printing the result using a 3D printer's internal imaging pipeline. They have later extended the approach to 3D by considering not only within-slice neighbors, but diffusing errors into a 4x4x4 voxel region [19].

In this paper the focus will be on presenting a framework that derives from a first-principles-based analysis of 3D objects.

Material objects have a variety of properties, e.g., weight, stiffness, color, density, transparency, etc., and can therefore be characterized in such terms as a whole or part-by-part. In parallel, an object's material composition can also be expressed. E.g., taking a chair, some of its parts may be made of wood, others of leather, metal, plastic, latex foam, fabrics, etc. Furthermore, these two views are not independent and it is, in fact, material composition that is the root cause of an object's properties: the parts made of metal are harder than those made of leather. Conversely, different material compositions can also yield matching properties, e.g., the wooden parts of a chair may be painted to match the color of the leather parts.

Such a natural-language approach can also be expressed in formal, quantitative terms, where properties are characterized through measured quantities (e.g., kg/m^3 for density, CIELAB coordinates for color, etc.) and where material use can be expressed in volumetric terms. Taking a chair, its total volume (e.g., 0.12 m^3) can be expressed as the sum of its component parts' volumes: 0.08 m^3 wood, 0.03 m^3 polyfoam, 0.01 m^3 leather and a negligible volume taken up by metal parts.

The material make-up of the example chair can also be seen in relative terms, where the following are probabilities of encountering any one of the component materials at a random location within its volume – wood: 67%, metal: ~0%, polyfoam: 25% and leather 8%. This analysis can also be repeated for parts or sub-volumes of an object. Importantly from the perspective of a print content processing pipeline, volumetric composition can be thought of in a dual way: both directly in volume coverage terms: e.g., 30% of a volume is occupied by material A, and in probabilistic terms: i.e., the likelihood of encountering material A when randomly sampling an object's volume is 30%.

In the case of 3D printing object properties are directly mapped to the volumetric probabilities of print-resolution voxel contents. This paper presents the HANS3D content processing pipeline that exercises control over a 3D printing system in precisely this domain, where volumetric probabilities are assigned to printed voxel types.

HANS3D basics

In this section, we outline the basics of the HANS3D framework, starting with the lowest level building blocks, followed by their volumetric, probabilistic combination into recipes associated with the sub-volumes of an object. We then show some of the properties of this domain and finally how the domain is halftoned into print-ready data.

Mvecs and Mvocs

Analogous to its 2D sibling, HANS [1], the basic building blocks for HANS3D are formed by all available at-voxel states of a given 3D printing system where all combinations of any controllable aspect of a single printable voxel define this domain. E.g., for a printing system like that of HP Jet Fusion where IR energy is used for fusing a layer of powder marked with one or more agents (inks), if a series of these agents is available and they can be independently controlled, then the set of atomic states are all combinations of the agents at their available amount states. If the system is binary (some agent, no agent) and there are *n* agents then 2^n atomic states result. If the system can deposit more discrete agent amounts (such as k amounts, including no agent, at a voxel) then the number of states is k^n. Note that this is akin to the set of Neugebauer Primaries in the 2D color printing domain. If the system can control additional aspects of the contents of a voxel or of how the system behaves at a voxel, then these result in additional dimensions of the combinatorial domain. If multiple powders are available, or if control over how voxels are fused can be exercised at voxel resolution level, then these become additional dimensions. Since all of these parameters together affect the material properties at a voxel (not only the powders, agents but also system parameters), we call these atomic states **M**aterial **vec**tors or **Mvec**s.

Tab. 1: An example of Mvecs for a 4-agent, binary system.

Label	Cyan	Magenta	Yellow	Neutral
Blank	0	0	0	0
N	0	0	0	1
Y	0	0	1	0
YN	0	0	1	1
M	0	1	0	0
MN	0	1	0	1
MY	0	1	1	0
MYN	0	1	1	1
C	1	0	0	0
CN	1	0	0	1
CY	1	0	1	0
CYN	1	0	1	1
CM	1	1	0	0
CMN	1	1	0	1
CMY	1	1	1	0
CMYN	1	1	1	1

Throughout this paper, a sample set-up will be used where a prototype printing testbed has a single powder and four agents that can be placed with some amount or no amount at each voxel independently: cyan (C), magenta (M), yellow (Y) and a transparent (N) agent. Assuming this set-up, with printing system parameters kept constant, the following list of 16 print resolution voxel contents (Tab. 1), where 0 indicates an agent's absence and 1 its presence, defines atomic states.

There are 16 basic building blocks since each of the agents can either be absent or present (resulting in k=2 levels of use) and there are n=4 agents. Fig. 1 illustrates a subset of these Mvecs in pseudo-colors with their labels shown in each case.

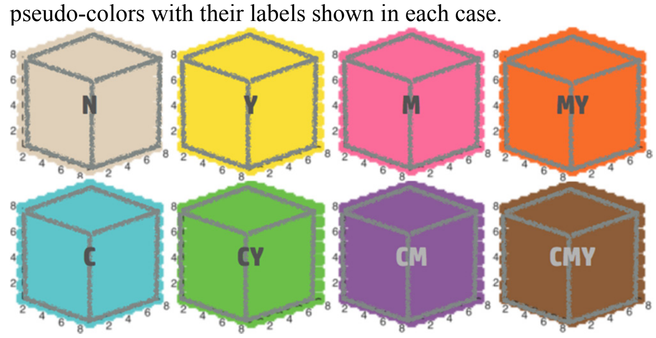

Fig. 1: *Example Mvecs – i.e., print-resolution voxel contents with different printing material combinations.*

Any pipeline that produces the input to a system, such as that described above, will produce a halftone whose voxel states are the set (or subset) of Mvecs above. However, while all halftones are in this domain, control is not exercised here, but in agent terms. It is then the consequence of choices about the use of agents (rather than Mvecs) and the halftoning strategy that determine the composition of print-ready voxels.

The next step from having defined the basic building blocks is to define their combined use over a volume. Here the HANS3D approach is conceptually one of reverse engineering. Given a 3D printed object, it is possible to express its printing material make-up by deconstructing it into a bill of Mvecs: how much of each of the printing system's Mvecs is used over that volume, and this can also be done for that object's parts *ad infinitum*. For example, an object may be made up of 70% blank voxels (i.e., voxels where no agent has been deposited), 20% of the C Mvec and 10% of the CMY Mvec. Picking a voxel at random from such an object's 3D halftone would make finding it blank most likely, finding it filled with powder that has the C agent would be the case 1/5th of the time and finding all three of the C, M and Y agents combined would only only have a 1 in 10 chance. Any volume within which the distribution of these probabilities does not change spatially (e.g. either it's uniformly random, or follows a specific spatial distribution) can, from the point of view of Mvec statistics be described by these probabilities alone. While this is a mechanism to describe a halftone, we use it as our control domain. Such expression of an Mvec distribution or probability (or relative volume coverages) will be referred to as an **Mvoc** – a **m**aterial (i.e., Mvec) **vo**lume **c**overage vector (Fig. 2). Mvocs are therefore the way to characterize or specify a distribution of Mvec volume coverages over some unit volume, or probabilities of a material choice.

Formally, for a set of independently controllable dimensions *P* (agents, powders, materials, parameters, …) where each can have *k* different states, the set of Mvecs is the set of ordered arrangements of k elements of length P, which results in $k^P = N$ elements in total. This is the set of permutations with repetition (sometimes referred to as n-tuples). Let this set be denoted as $\boldsymbol{S}$, with elements of the set as S_i. Then, Mvocs denoted as M_i are defined as vectors of length N of volume coverages (or volumetric probabilities) $[\alpha_1, \dots, \alpha_N]_i$ such that $0 \le \alpha_j \le 1$ and $\sum_{j=1}^{N} \alpha_j = 1$ where each α_j denotes the probability (or volume coverage) of S_j over the volume associated with M_i. Hence, Mvocs, as defined above, are convex weights: weighted combinations of Mvecs. Fig. 2 shows examples of Mvocs assigned to the volume of a cube.

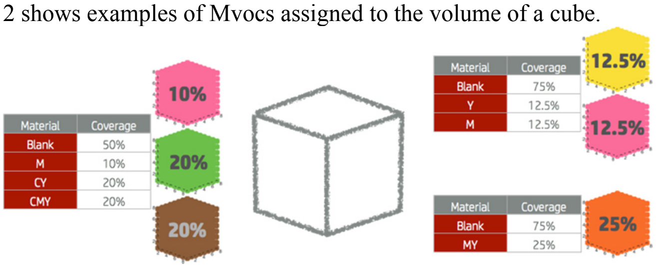

***Fig. 2:** Example Mvocs – i.e., Mvec probability / volume coverage distributions.*

Note that while the full set of Mvecs is the complete set of n-tuples and Mvocs are vectors that have the length of the size of this complete set, neither the initial Mvecs need to be complete (i.e. heuristics can be employed to reduce their number to those of interest), nor do all Mvocs need to use all Mvecs (i.e. α_i's can be 0).

To summarize, the essence of controlling printing materials natively is to identify the atomic building blocks that they give rise to (Mvecs) and to specify their use in volumetric probability or coverage terms (Mvocs). Such a volumetrically-probabilistic approach derives from a first-principles analysis of 3D object composition and is applicable to any 3D printing technology, any printing materials (e.g. powders, agents, substrates, filaments, ...) and printing system parameters, and in the pursuit of any printed object property that varies with printing material composition and system behavior.

Mvoc properties

The first, important property of Mvocs is their convexity. Since the probabilities associated with individual Mvecs add up to 100% (or 1), they can be thought of as convex weights (a third way of thinking about the volumetric domain presented here) or as barycentric coordinates in the volume coverage / probability space (a fourth way of looking at the same matter). Any set of convex weights associated with a 3D printing system's Mvecs is a valid way of addressing its atomic building blocks. Given that each Mvoc is defined in these terms, their combination (or interpolation) also has to be performed such that convexity is maintained. Given two Mvocs, M_1 and M_2, a natural way to transition or combine the two is to use convex combinations: convex combinations of Mvocs preserve convexity by way of associativity. For weights β_1 and β_2 associated with M_1 and M_2, a new Mvoc M_3 is formed as follows:

$$
\begin{gathered}
M_3 = \beta_1 * M_1 + \beta_2 * M_2 = \\
= \beta_1 * [\alpha_1, \dots, \alpha_N]_1 + \beta_2 * [\alpha_1, \dots, \alpha_N]_2 \\
= [\alpha_1, \dots, \alpha_N]_3 \; where: \\
[a_j]_3 = \beta_1 [a_j]_1 + \beta_2 [a_j]_2
\end{gathered}
$$

Since β_1 and β_2 are a pair of convex weights (i.e. scalars that satisfy the same constraints as α_j's), after applying the weights to the original two Mvocs, M_1 and M_2, M_3 is a new valid Mvoc, a convex combination of Mvecs (see examples in Fig. 3 where A and C are combined at 50% each as well as A, B and C combined at 33% each, forming new Mvocs that are the convex combination of the constituent Mvocs).

This convexity, and the fact that convex weights are specified for a 3D printing system's atomic, print-resolution-voxel contents has important consequences, including the following four:

1. A local volumetric neighborhood, for which an Mvoc is specified, will only contain Mvecs that have non-zero probabilities α_j specified for them. This provides direct and explicit control over how a system's materials are combined (or kept apart) at voxel level, where both **separation** and **combination** can have important consequences. E.g., in early tests large differences were found in terms of color depending on whether agents are used separately or combined in individual voxels (Fig. 6). In other cases keeping certain materials apart may be important, e.g., if those materials have undesirable interactions, or, conversely, if their combination is essential (e.g., co-locating cells and nutrients in bio-printing or reactants in chemical 3D printing).
2. A transition between two Mvocs will be **homogeneous**, since, as the probabilities of one end-point decrease smoothly, the probabilities of the other endpoint's Mvecs increase. This has consequences on perceptual attributes such as color, where on the surface of an object the convex transitioning in volume coverage terms is smooth. It also applies to non-perceptual properties in their respective domains to provide continuous variation. It does not however replace the need for a linearization (e.g. in color terms for the smoothness to be also perceptually uniform).
3. Transitions between two Mvocs are also **closed**. Since the entire transition only ever involves the endpoints' non-zero Mvecs, at no point in a transition will "new" Mvecs be introduced. This is in contrast to transitioning directly in agent channels, where, at some point agent overlaps take place, and a discontinuity occurs due to the introduction of an at-voxel state (a new Mvec) whose properties are often non-linearly related to its constituent agents.
4. Finally, thanks to the **associativity** of Mvocs, convex combination applies not only to Mvecs but to Mvocs too – as shown above – where new Mvocs can be formed by convexly combining component Mvocs [20].

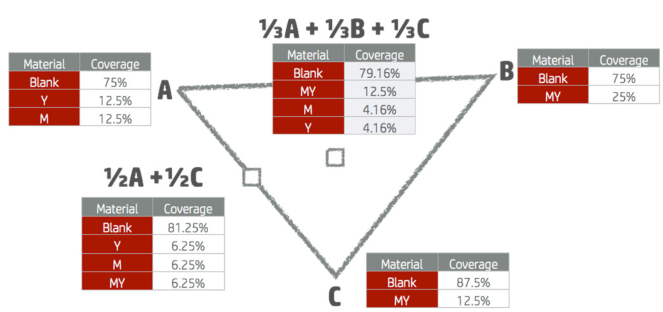

***Fig. 3:** Convexly combining Mvocs.*

The result of the above properties is a well-behaved domain in that a relationship is established between Mvocs and some object property. The aforementioned smoothness of transitioning, homogeneity and closedness are ensured. Another direct consequence of convexity is that, assuming properties relate to Mvocs monotonically, the gamut of such properties can be assumed to be convex as well. If this relationship is furthermore linear then convexity is directly related between Mvocs and the property domain and as a consequence the convex hull of the object properties is the domain that is accessible. A non-linearly modified colorimetric domain has approximately this property and HANS3D therefore results in convex color gamuts.

Halftoning

Given an input object processed through a pipeline to the point where Mvocs are available for each voxel, it is necessary to perform a transformation to Mvecs, since a single print-resolution voxel cannot contain an Mvoc, but only an Mvec (which is how Mvocs are defined – i.e., the weighted combination of single, print-resolution voxels). The next challenge is to make a choice of Mvec at a given voxel such that it is still the Mvoc specified for it that results over a local volume in the final, printed object.

A mechanism that selects Mvecs from Mvocs, while preserving probabilities, is therefore needed and two types of strategies can be pursued: error diffusion or an evolution of the PARAWACS (Parallel Random Area Weighted Coverage Selection) approach originally developed for 2D [21].

The key to halftoning Mvocs is the insight that the role of halftoning is the Mvec selection, so that the likelihood of an Mvec being selected from an Mvoc is that Mvec's volume coverage. In other words, halftoning in HANS3D is a problem of sampling a probability distribution. PARAWACS3D starts with re-expressing an Mvoc in a coverage-cumulative way. Instead of assigning the n-th Mvec the probability with which it is to be used, it receives the probability with which the first n Mvecs are to be used. Hence M_i, previously defined as a vector of length N of volumetric probabilities $[\alpha_1, \dots, \alpha_N]_i$, is first re-expressed as:

$$\mathrm{M}_i^C = [\alpha_1, \alpha_1 + \alpha_2, \dots, \sum_{j=1}^{N-1} \alpha_j, 1]_i$$

This results in a vector of length N, as before, where the value at j is the cumulative sum of volumetric probabilities from 1 to j, with the last element, the sum of all probabilities of all Mvecs always being 1. A first, simple approach to the selection of a single Mvec S_j from the Mvoc M_i^C is to use values between 0 and 1 that come from a uniform distribution. These values then act as selectors, where the Mvec from an Mvoc is selected that has the first cumulative probability above the selector value.

This process (Fig. 4) ensures that Mvoc probabilities are achieved over a local volume neighborhood. A naïve way to generate these selectors is randomly, with a uniform random distribution, where the result is a 3D white-noise halftone.

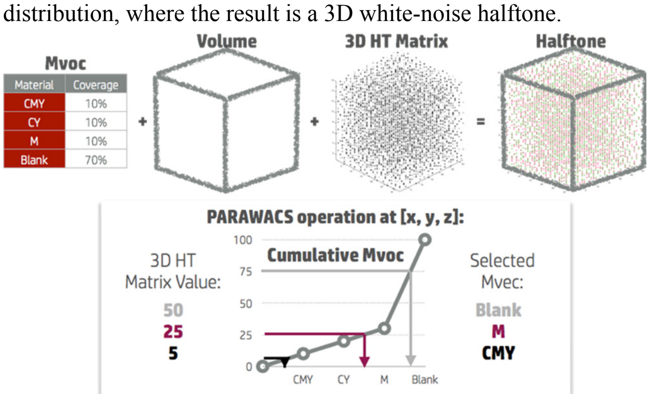

Fig. 4: Example application of PARAWACS3D.

Alternatively, 3D halftone matrices can be pre-computed with specific 3D distributions in mind, e.g., taking inter–level relationships into account, or allowing for perfect (or partial, or no) complementarity between the agent placements in slices – i.e., no subsequent slices place agents at the same spatial location.

Note that the order in which Mvecs are encoded, i.e. the order in which the cumulative probability distribution is constructed, also plays an important role and can have an impact on the final halftone quality and therefore the resulting printed object's properties.

Since the halftone matrices are pre-computed and their properties designed offline, the above halftoning strategy is also natively volumetric, both since the domain to which it is applied is such (Mvocs) and because it is fully parallel and does not need to be applied plane by plane (or following a path, like error diffusion). Hence, PARAWACS3D halftoning can directly be applied all at once over a volume, independently of the printing process. It also scales well with the complexity of printing systems and its application to a single agent, single property device is no different from a multi-agent, multi-property one. In agent vector based pipelines, using matrix based approaches, this is not the case since each agent vector plane is halftoned separately (with work having been done on establishing inter-plane relationships) and then combined. Hence, if a system has X agents, X planes need to be halftoned and combined into the final, combined halftone. Instead, with PARAWACS3D all agents (and agent combinations, that result in the Mvec domain) are halftoned at the same time via the Mvec selection mechanism described above.

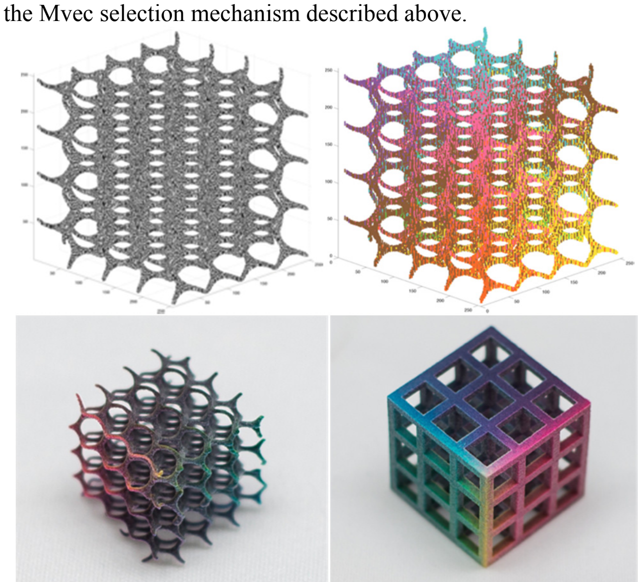

Fig. 5: Structural PARAWACS3D halftone matrices applied to a simple cube object with smooth color gradients. Top row: visualization of halftone matrix (left) and final halftone (right), bottom row: printed object photos.

Finally, this halftoning approach also lends itself naturally to controlling structure, without introducing complexity in the content processing pipeline. 3D halftone (selector) matrices can be designed that have a desired property or structure embedded, at macro level (for aesthetic or mechanical purposes) or micro level (varying the nature of the distribution of selector values). Fig. 5 shows two examples of a cube processed and halftoned in this way, with the halftone matrix in one case using the geometry of the

(10,3)–a Network (or the K4 crystal) [22] on the left and a simple regular lattice on the right.

The key here is that the structure is introduced in the object at the point of halftoning without any change to the pipeline. The only difference is the use of particular 3D halftone matrices. Furthermore, regardless of the complexity of the structure, the halftone matrices occupy the same memory as any other halftone matrix (or even less) since they, by definition, are in a rasterized domain and therefore independent of the number of primitives needed to describe the geometry in vector form.

Building a color to Mvoc mapping

At its simplest, a mapping from color to Mvocs can start with choosing Mvocs *a priori*, based either on prior knowledge or assumptions of expected behavior. Then, objects of such constant Mvocs can be built, printed and color-measured. An example is shown in Fig. 9, where 14 Mvocs were designed, using first (top left to bottom right) two Mvecs at a time (the blank Mvec and Y, M, C, MY, CY, CM and CMY). In each part, the blank Mvec accounted for 75% of the volume coverage and 25% was occupied by each of the non-blank Mvecs. Finally, Mvocs that use three, four or more Mvecs were also included. The assumption to verify here was what color the use of single-agent and multi-agent Mvecs would yield. Of particular interest here are the pairs of object parts labelled “MY” and “M, Y”, “CY” and “C, Y” etc. which use the same agent-amounts (in fact the same agent-vectors) but in one case they are always combined (MY means that every non-blank voxel over that volume has both the M and the Y agent present), while in the other they are always kept away from each other (M, Y means that both M and Y are used over the volume but no voxel contains them both at the same time). As can be seen, the colors of these object parts differ from each other, which is a confirmation of the assumption that control over the exact Mvecs used to print objects does have an impact on object properties such as color.

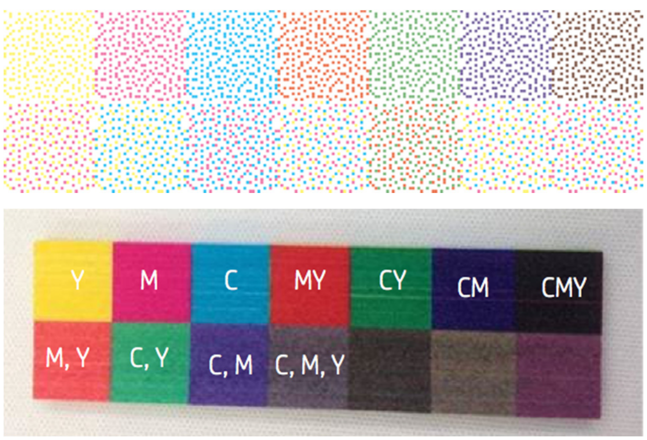

Fig. 6: *Simple Mvocs, shown as the input halftone (low resolution single slice, top) and as the object printed on a prototype testbed (bottom).*

Because of the convexity and associativity of Mvoc formation, one alternative is to hand-build a mapping from device color to Mvocs. For the case of a color pipeline we can assume the input space to be a device RGB domain, hence the objective is to build a mapping that goes from there to Mvocs. The hand-built example LUT shown in Tab. 2 – based on the data from Fig. 6 – is the mapping used for making the first continuous color Multi-Jet Fusion (MJF) prints (Fig. 7) using the prototype printing testbed to which HANS3D is being applied throughout this paper. Here dRGB denotes a device RGB space – i.e., one whose colorimetry is defined by a mapping to a printing system’s Mvocs.

Tab. 2: Hand built HANS3D dRGB to Mvoc LUT.

dR	dG	dB	Mvoc
0	0	0	B:0.75, CMY: 0.25
0	0	255	B:0.75, MY: 0.25
0	255	0	B:0.75, CY: 0.25
0	255	255	B:0.75, C: 0.25
255	0	0	B:0.75, MY: 0.25
255	0	255	B:0.75, M: 0.25
255	255	0	B:0.75, Y: 0.25
255	255	255	B:0.75, N: 0.25

Note that there is a very simple and direct relationship here between the vertices of the device RGB space and the Mvecs formed by the CMYN agents (e.g., the device Red primary is mapped to a 25% coverage of the Mvec that combines Magenta and Yellow and 75% of the volume being left blank). Note also that only eight choices, the extreme vertices of the dRGB cube, had to be made and that the interior of the cube is then obtained by interpolation [20]. This results in the smooth, well-behaved output in Fig. 7 because the Mvoc domain is convex, which takes care of filling in the whole RGB space.

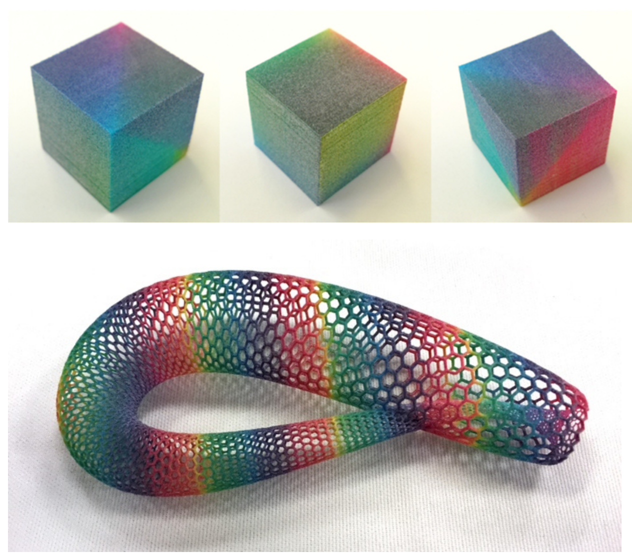

Fig. 7: *An RGB cube (top) and a Klein bottle model, processed using the LUT shown in Tab. 2. The input for this object in geometric terms is a plain cube and the Klein bottle [23], while in the case of the cube, every location [x, y, z] has a unique RGB color specified, proportionally representing all possible RGB values. The Klein bottle instead used a color transitioning map, also throughout the object.*

Summary

The ability to exercise control over all possible combinations of available materials on a voxel by voxel basis in a deliberate manner open up new possibilities in the 3D printing process. This domain gives both access to the full variety of possible properties obtained from a given system and at the same time, because of how well behaved the domain is, also allows for ultra-simple ways to build resources and control properties in isolation (e.g. using only 8 nodes in a LUT). The volumetrically probabilistic combinations then scale this domain naturally to be natively 3D. In this paper results were shown focusing on the application of this framework to 3D color printing, however the same mechanism also applies to characterizing, modelling, predicting and controlling any other object properties as well, such as physical properties like mechanical strength or density [24]. Furthermore, the HANS3D framework also scales to more complex systems with multiple properties being controlled concurrently and even needing to be co-optimized and in principle applies to any printing mechanism.

Acknowledgements

The authors would like to thank their colleagues at HP Inc. for their support, in particular Rafa Gimenez, Ramon Pastor, Albert Serra, Africa Real and Annarosa Multari.

References

[1] J. Morovič, P. Morovič, J. Arnabat "HANS – Controlling Inkjet Print Attributes Via Neugebauer Primary Area Coverages," *IEEE Transactions on Image Processing*, 21, 2, (2011)

[2] Ming, L. W. and Gibson, I. 1999. Possibility of colouring SLS prototypes using the ink-jet method, In *Rapid Prototyping Journal*, MCB University Press, *5*, 4, 152-153.

[3] Papas, M., Regg, C., Jarosz, W., Bickel, B., Jackson, P., Matusik, W., Marschner, S. and Gross, M. 2013. Fabricating Translucent Materials Using Continuous Pigment Mixtures, In In *ACM Tog (Proc. SIGGRAPH)*, 32, 3.

[4] Rodrigues, H., Guedes, J., And Bendsoe, M. 2002. Hierarchical Optimization Of Material And Structure. In *Structural and Multidisciplinary Optimization*, 24, 1, 1–10.

[5] Willis, K. D. D., Brockmeyer, E., Hudson, S. E., and Poupyrev, I. 2012. Printed optics: 3D printing of embedded optical elements for interactive devices. In *UIST*.

[6] Mironov, V., Visconti, R. P., Kasyanov, V., Forgacs, G., Drake, C. J., and Markwald, R. R. 2009. Organ printing: tissue spheroids as building blocks. In *Biomaterials,* 30, 12, 2164– 2174.

[7] Skorski, M. R., Esenther, J. M., Ahmed, Z., Miller, A. E. and Hartings, M. R. 2016. The chemical, mechanical, and physical properties of 3D printed materials composed of TiO2-ABS nanocomposites, In *Science and Technology of Advanced Materials*, 17:1, 89-97.

[8] Sitthi-Amorn, P., Ramos, J. E., Wangy, Y., Kwan, J., Lan, J., Wang, W. and Matusik W. 2015. MultiFab: a machine vision assisted platform for multi-material 3D printing, In *ACM Transactions on Graphics (TOG)*, ACM, *34*, 4.

[9] Dimitrov, D., Schreve, K. and de Beer, N. 2006, Advances in three dimensional printing – state of the art and future perspectives, In *Rapid Prototyping Journal*, 12, 3, 136-47.

[10] Schumacher, C., Bickel, B., Rys, J., Marschner, S., Daraio, C., and Gross, M. 2015. Microstructures to control elasticity in 3D printing. In *ACM Trans. Graph.*, 34, 4, 136:1–136:13.

[11] Vidimče, K., Wang, S.-P., Ragan-Kelley, J., and Matusik, W. 2013. OpenFab: A programmable pipeline for multi-material fabrication. In *ACM Trans. Graph.* 32, 4, 136:1–136:12.

[12] Parraman, C., Walters, P., Reid, B., and Huson, D. 2008. Specifying colour and maintaining colour accuracy for 3d printing, In *SPIE/IS&T Electronic Imaging Conference*, 6805, 68050L.

[13] Xiao, K., Zardawi, F., van Noort, R. and Yates, J. M. 2013. Color reproduction for advanced manufacture of soft tissue prostheses. In *J Dent.*, 41, S5, e15–e23

[14] Arikan, C., Brunton, A., Tanksale, T., and Urban, P. 2015. Color-Managed 3D-Printing with highly Translucent Printing Materials. In *Proc. SPIE 9398, Measuring, Modeling, and Reproducing Material Appearance*, 93980S

[15] Stucki, P. 1997. 3D halftoning, *In Proc. SPIE*, 2949: 314-317.

[16] Cho, W., Sachs, E., Patrikalakis, N. M., and Troxel, D. E. 2003. A Dithering Algorithm for Local Composition Control With Three-Dimensional Printing. In *Cad*, 35, 9, 851–867.

[17] Brunton, A., Arikan, C. A. and Urban, P. (2015). Pushing the Limits of 3D Color Printing: Error Diffusion with Translucent Materials. In *ACM Transactions on Graphics (TOG)*, 35, 1, 4.

[18] Sun, P. L. and Sie, Y. 2015. Color dithering methods for LEGO-like 3D printing, In *Proc. SPIE 9395, Color Imaging XX: Displaying, Processing, Hardcopy, and Applications*, 93950J

[19] Sun, P. L. and Sie, Y. 2016. Color Uniformity Improvement for an Inkjet Color 3D Printing System. In *Electronic Imaging, Color Imaging XXI: Displaying, Processing, Hardcopy, and Applications, 1-6(6).*

[20] Morovič, J., Morovič, P., Rius, M. and García–Reyero, J. M. 2013. 8 vertex HANS: An ultra-simple printer color architecture, In *21st IS&T Color and Imaging Conference*, Albuquerque, NM, 210-214.

[21] Morovič, P., Morovič, J., Gondek, J. and Ulichney, R. 2017. Direct Pattern Control Halftoning of Neugebauer Primaries, In *IEEE Transactions on Image Processing*, submitted for publication.

[22] Hart, G. W. 2017. The (10, 3)-a Network, http://www.georgehart.com/rp/10-3.html

[23] Rasul, K., Klein Bottle (STL 3D model at http://www.thingiverse.com/thing:121871), 2013.

[24] Morovič, P., Morovič, J., Tastl, I., Gottwals, M., 2017. HANS3D: a volumetric voxel-control print content processing pipeline, In *ACM Tog (Proc. SIGGRAPH)*, submitted for publication.

Author Biography

Peter Morovič received his Ph.D. in computing sciences from the University of East Anglia, UK in 2002 and holds a B.Sc. degree in theoretical computer science from Comenius University, Slovakia. He has been a Color and Imaging Scientist at HP Inc. since 2007, has published 50+ scientific articles and has 100+ US patents filed (25 granted) to date. His interests include 2D/3D image processing, color vision, computational photography, computational geometry and his Erdős number is 4.

Ján Morovič received his Ph.D. in color science from the University of Derby (UK) in 1998, where he then worked as a lecturer. Since 2003 he has been at Hewlett-Packard in Barcelona as a senior color scientist and later master technologist. He has also served as the director of CIE Division 8 on Image Technology and Wiley and Sons have published his 'Color Gamut Mapping' book. He is the author of over 100 papers and has filed 100+ US patents (26 granted).

Ingeborg Tastl is a principal scientist at HP Labs in Palo Alto California. Her current research focus is on appearance perception, measurement and reproduction for additive manufacturing. Ingeborg has worked in the area of color reproduction for commercial and industrial printing since she joined HP Labs in 2001. She got her Master and PhD degrees from the Technical University in Vienna, Austria and has been an active member of IS&T for many years.

Melanie Gottwals is a research engineer working at HP Labs in Palo Alto, USA. She joined HP in 1989 and started her career in storage technologies working on advanced read/write channels, media and heads to increase performance of disk and tape drives. In 2005 her research shifted to color technology where she worked on a low-cost colorimeter plus height sensor and color correction via mobile phone applications. Her current research focus is on color related to 3D printing. She has a B.S. degree in Electrical Engineering from the University of California at Davis.

Gary Dispoto is director of the Fabrication Solutions Team at HP Labs. His team designs and optimizes the short run production processes for print and additive manufacturing. Gary has been involved in various aspects of digital print R&D since joining HP Labs in 1985, including extensive work in the area of digital color reproduction. He holds B.S. and M.S. degrees in electrical engineering from Stanford University and an MBA from the Santa Clara University.