

第9章 第5節 「高い表現力を実現するボクセルベースのデータフォーマット FAV」

富士ゼロックス株式会社 藤井 雅彦

9.5. はじめに

3D プリンタ、すなわち材料を付加することで立体物を造形する技術は Additive Manufacturing と呼ばれ、現在 7 つの方式が ISO/ASTM による標準化の中で分類されている¹⁾。これら 7 方式に属さない新しい方式も提案され実用化されているが、ほとんどの方式は、一定の厚みを持った層を順次積み重ねることにより立体物を造形する。このため、立体物の外形形状だけでなく、射出成形や切削加工では困難であった複雑な内部構造を実現することが可能になっている。また、材料の混ぜ合わせにより部分的に物理特性(例えば硬さ)を制御可能な 3D プリンタや、造形物の表面のみならず内部までフルカラー化が可能な 3D プリンタも導入されている。

一方、3D プリンタに入力されるデータのフォーマットとして現在標準的に使用されている STL^{*1}は、1988年に 3D Systems から 3D CAD 用に提案されたものである。STL はポリゴンと呼ばれる三角形で立体物の外形のみを近似するものであり、複雑な内部構造を表現できないばかりか、材料情報や色情報を保持していない。STL が提案された当時は設計された 3D データを最終加工用のデータとして用いたとしても、単色、単一材料、外形のみの加工しか行なえない切削加工や射出成形では問題なかった。しかし、複雑な内部構造の実現やフルカラー色再現など、性能が進化した現在の 3D プリンタの能力をそのまま活かすデータ形式になっておらず、3D プリンタの能力を活かす出力(造形物)を得るためのプロセス(データフロー)は煩雑になっていた。

このため近年、3MF^{*2} や AMF^{*3} といった新しいデータフォ

ーマットが提案されている。3MF や AMF では内部の色や材料情報は、(外形の)表面情報からの演算により付与することができるものの、構造内部に独自に情報を持たせることはできない。また STL と同様に外形をポリゴンで近似するものであり、複雑な内部構造を表現することも難しい。表 1 にこれまで使われてきた代表的なポリゴンベースの 3D データフォーマットの特徴をまとめる²⁾。

本節では、これら従来のポリゴンベースの 3D データフォーマットが持つ課題を解決するため 2016 年に提案され³⁾、2019 年に JIS に制定された²⁾ボクセルベースの新しい 3D データフォーマット FAV^{*4}を解説するとともに、FAV による 3D データワークフロー上の課題解決や、新たな可能性を説明する。

9.5.1. ボクセルによる 3D 表現と FAV フォーマット

ボクセル(Voxel)とは立体物(3次元モデル)を構成する基本要素であり(図 1)、デジタル 2 次元画像を構成するピクセル(Pixel)と Volume を併せた造語である。

図 2 にこれまでの STL とボクセルにより表現された立体物(うさぎ)を示す。STL では三角形のポリゴンにより外形のみを近似表現するのに対し、ボクセルでは外形のみならず内部構造も容易に表現可能となる。各ボクセルに色情報や材料情報を保持すれば、3D プリンタで直接これらの情報を参照し、造形することが可能になる。

3D データフォーマット FAV は上述したボクセルベースのデータフォーマットであり、個々のボクセルに材料や色以外の情報も保持でき、立体物を表現する高い能力を備えている(図

表 1. 3D データフォーマットの特徴比較²⁾

	ポリゴンベース				ボクセルベース	
	STL	AMF	3MF	OBJ	DCM	FAV
3D 形状(外形)	○	○	○	○	○	○
表面情報	色情報	×	○	○	○	○
	材料情報	×	○	○	△	○
	複数材料	×	○	○	△	○
	微細構造	×	×	×	×	○
内部情報	色構造	×	△	△	×	○
	材料情報	×	△	△	×	○
	複数材料	×	△	△	×	○
	微細構造	×	×	×	×	○
リンク情報	×	×	×	×	×	○
ユーザー定義情報	×	△	△	×	×	○

○：データの損失無く管理可能、または管理データの活用が容易

△：データの読み替えが必要、または管理データの活用時に計算処理などが必要

×：データの取り扱いが不可能

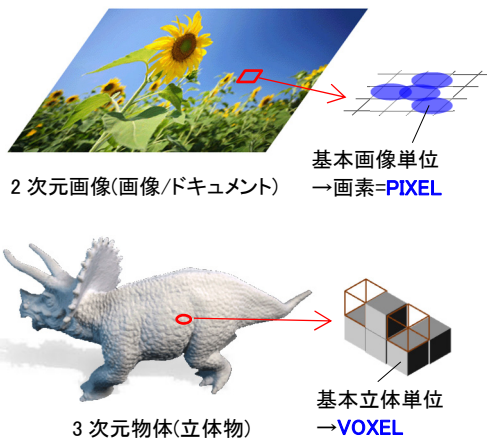


図 1. ピクセル(Pixel)とボクセル(Voxel)

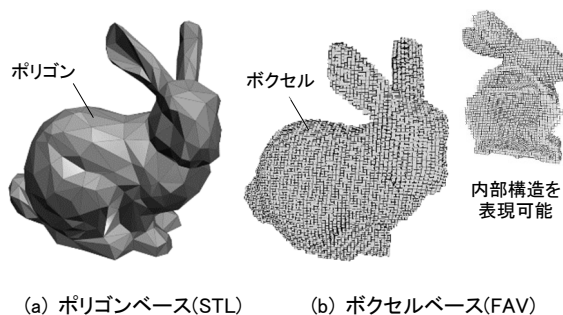


図 2. ポリゴンとボクセルによる立体物の表現

3)(表 1). FAV では立体物情報が図 4 に示すように XML 構造によって管理される^{2,4)}. 以下, FAV における立体物表現情報, および XML 構造の各要素を結びつけて説明する.

<fav>は, FAV フォーマットのルート要素を示しており, FAV フォーマットに従って定義した 3D モデルデータは<fav>という要素から開始する. 2020 年 1 月現在, 最新の仕様(version)は 1.1a である.

<metadata>にはタイトルやデータの作成者情報などを記載する. データの使用に関するライセンス方針などを, クリエイティブ・コモンズ・ライセンスに基づいてここに記載することもできる.

<palette>には 3D モデルデータを FAV フォーマットに基づいて構成するための準備として, 使用する各ボクセルの形状や材料などの基本情報を登録する. <geometry>ではボクセルの形状や倍率を定義するが, ボクセルとして図 1, 2, 3 に示した立方体のみならず, 円柱や球など様々な形, 大きさを使

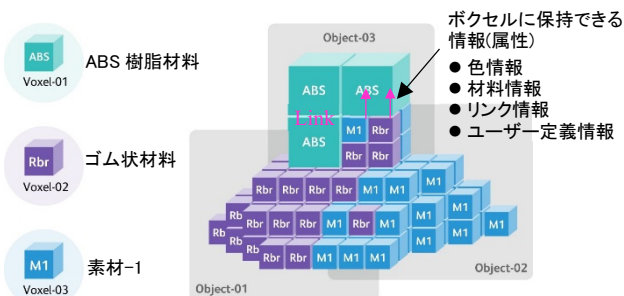


図 3. FAV による立体物の表現

用することが出来る(図 5)^{2,4)}. さらにユーザーが 3D プリンタ方式や用途に合わせて独自にボクセル形状やサイズを外部ファイルで定義し, <shape>以下の<reference>から参照することも可能である. ボクセルの材料情報は<material>で管理するが, 製品名や各種標準に登録された名前のみならず, 物理特性を記述することもできる.

<voxel>では, 実際に立体構造を表現するためのボクセルに, <palette>に登録されたボクセルの形状や縮尺, あるいは材料情報(混合比率)を割り当てて定義し, 管理する. <display>はボクセルを(例えばディスプレイで)可視化する場合の色情報であり, 造形する際の色は以下に示す<color_map>で記述される. また, 外部の FAV ファイルを<reference>で指定することで, 任意のボクセル集合を 1 ボクセルとして使用することもできる. 例えば, 外部の FAV ファイルに硬さの異なる材料

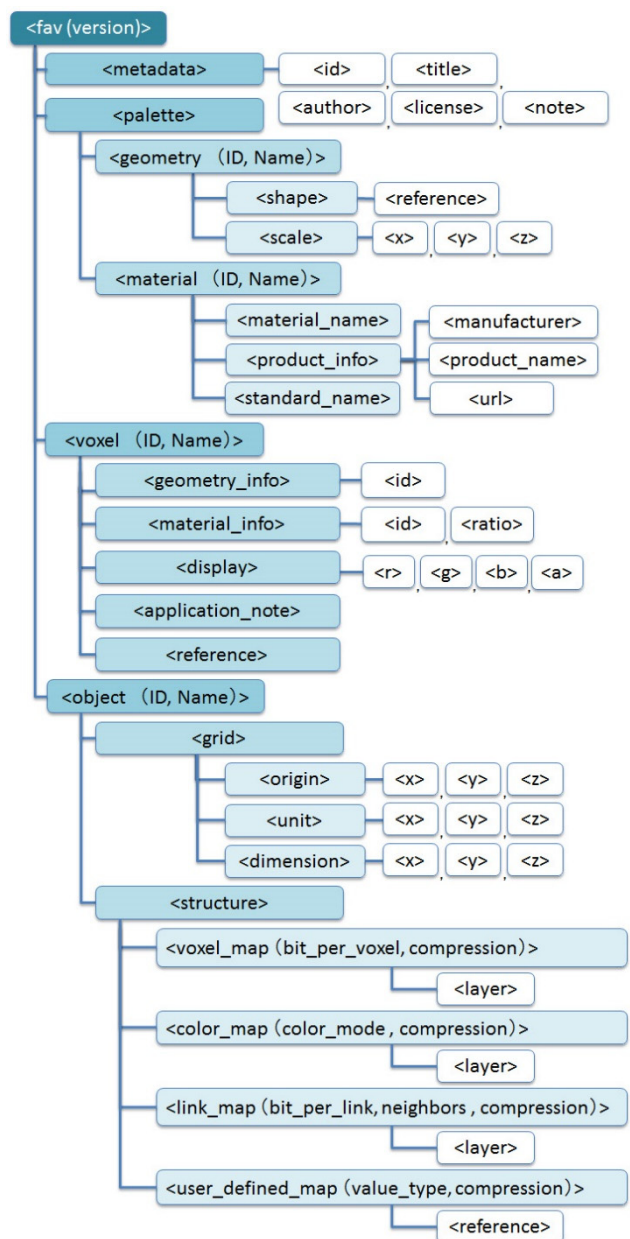


図 4. FAV フォーマット構成^{2,4)}

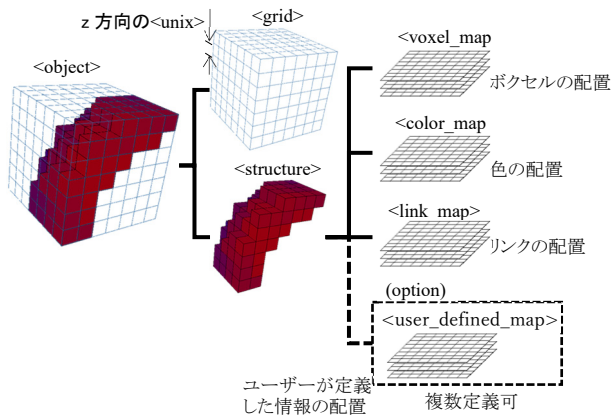


図 6. 立体モデルにおける各種情報の表現⁴⁾

(ボクセル)で構成される複合材を構成し、これを1ボクセルとして用いることも可能になる。<voxel>(id)に”0”を指定することで空いたセルを予約(定義)することもできる。ボクセル配置を定義する<voxel_map>で0を指定することでボクセルが存在しないセルを表現することもできるが、この場合は編集によりこのセルに新たなボクセルを形成することが可能になる。しかし、例えば立体物の造形後にこのセルに何かを充填するために’空’にしておく必要がある場合、この予約による空いたセルを利用することができる。

<object>は、ボクセルや様々な情報の立体配置を記述する要素である(図 6)^{2,4)}。<grid>でボクセルを配置する空間を定義し、<unit>は1つのボクセルを置ける格子(セル)1つのサイズであり、<dimension>はこのセルを並べる数である。すなわち<unit>*<dimension>でボクセルを配置する空間全体の大きさを示すことになる。定義されたボクセルの立体配置を<structure>下の<voxel_map>で記述することで立体構造が表現される。<color_map>ではボクセルに割り当てる色分布を記述する。また、近傍(6近傍/18近傍/26近傍)ボクセル同士の関係を示すリンク情報も保持することができ、その配置を<link_map>で記述する。リンクの活用として、例えば造形の順序(ツールパス)や材料物性の異方性などを情報として持たせることが考えられる。積層して造形する3Dプリンタによって作られた立体物は強度の異方性を持つものが多い。例えばリンクにボクセル同士の接合強度を保持すれば、造形物の物理強度の予測(シミュレーション)にも有効となる。ユーザーが独自に定義した(複数の)特性(属性)を<user_defined_map>で各ボ

クセルに割り当てることも出来る。例えばシミュレーションによる解析結果などの割り当て(表示)が考えられる。

なお FAV における座標単位は mm を用い、1mm 当りに配置可能なボクセルの数(<unit>で記述されるセルの数)、すなわち解像度を vpm(voxel/mm)で表す。

9.5.2. ボクセルベースデータの効用と活用

ボクセルベースの 3D フォーマットを立体物の表現に用いることで、上述したように進化した 3D プリンタの能力を直接活かして立体物を造形できるだけでなく、次のような効果も期待できる⁵⁾。

(a) 3D データフローの整流化

現在、多くの 3D データは 3D CAD で作成されている。CAD で作成された 3D データを STL に変換する際に、ポリゴンの抜けや重複などのエラーが発生することあり、3D プリンタで造形する前にこのエラーを修復する必要がある。これらのエラーを自動で修復するソフトウェアもあるが、細かいエラーは手作業で修復することもあり手間がかかっている。CAD からの出力を直接ボクセルに変換すれば、3D 表現としてのデータエラーは起きにくい。

また、一旦 STL に変換すると、STL 形式のままでは形状の変更や、他の 3D データと合わせることなどの編集が非常に難しくなる。ボクセルで表現されていれば、ボクセルの足し算、引き算、あるいは掛け算などのいわゆるブーリアン演算が可能であり、編集作業を容易に行なうことができる。これは 3D プリンタが得意とする少量多品種対応に必要な 3D データのカスタマイズを行い易くすることにつながる。

(b) シミュレーションとデザインの境界排除

CAD でデザインした形状が 3D プリンタで造形され、それが使用条件下でどのような変形や破壊が生じるかをシミュレーションで確認できれば、3D プリンタで造形する前に最適なデザインを得ることができる。STL に変換されたデータは、シミュレーションソフトに合わせて再メッシングが必要であり、シミュレーション結果をデザインに反映させるためには、再度 CAD まで戻る必要がある。立体形状がボクセルで表現されていれば、(ボクセルは有限要素であり)再メッシングせずにシミュレーションへの適用が可能であり、ボクセルのままシミュレーション可能なツールも開発されている⁶⁾。またボクセルであれば上述したようにブーリアン演算が出来るため、容易にシミュレーション結果をデザイン変更で反映することができる。このようにシミュレ

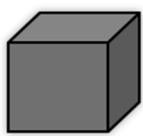





イメージ						
(id)	01	02	03	04	05	06
(name)	Cube01	Cube02	Plate	BigSphere	SmallSphere	Cylinder
<shape>	cube	cube	cube	sphere	sphere	user_defined
<scale>	2×2×2	1×1×1	1×1×0.3	1.5×1.5×1.5	0.25×0.25×0.25	3×1×1

図 5. ボクセル形状(<geometry>パターン)例^{2,4)}

ーションとデザインがシームレスに実現可能な新しいデザイン環境を構築することが出来る。

(c) 他の 3D データフォーマットとの連携

3D プリンタ以外にも、医療用の 3D データ(DICOM^{*5})や 3D スキャナーの点群データなど、様々な 3D データ形式が存在する。CAD で作成した 3D データや、DICOM データ、点群データを合わせて(編集して)新しいデザインをすることは、それぞれのデータ互換性が十分でないため、現在では困難である。全ての 3D データをボクセルに変換(ボクセルサイズ)すれば、ボクセルでの編集は容易であり、様々な形式の 3D データを活用して新しい価値を提供するデザインを行うことができる。

(d) モデリング(3D デザイン)の容易化

図 3 からわかるようにボクセルで立体物を表現することは、ブロックを積み上げて遊ぶ[積み木]であり、同様の手法を用いて建物を作るゲームソフトもある。3D CAD を習得して自分で立体物をデザインすることは多くの人にとって難易度が高いが、積み木のように立体物を設計できる簡単なデザインツールがあれば、誰でも 3D デザインが可能になるはずである。新しいビジネスチャンスとして少量多品種の販売であるロングテールが注目されており、自分の欲しい(自分に合った)カスタマイズされたものを手に入れることも、豊かな生活実現のために望まれている。しかし、これらを実現する上で、このマスカスタマイズデザインを誰がするのかという課題がある。ボクセルデータの編集容易性ととも、ボクセルによる容易なデザインツールは、ユーザー自らのカスタマイズデザイン(3D データ作成)を実現可能にするだろう。

(c) 製造工程における 3D データ管理の一元化

3D プリンタを出力先とした製造工程はもちろん、CAD による部品設計から金型作成、金型による部品の量産といった 3D プリンタを使用しない従来の製造工程においても、3D データの一元管理を行う上で FAV の活用が考えられる。前項で説明したように、FAV では<user_defined_map>によってユーザー独自に定義した属性を、ボクセルに割り当てることができる。製造の各工程で発生する様々な 3D データ、例えば CAD で作成した「部品形状データ」だけでなく、シミュレーションによる部品の強度解析結果である「構造解析データ」、金型による製造時の「熱流動解析データ」、あるいは製造された部品の「3D 計測データ」などを FAV の仕様に基づいて変換し、それぞれ属性として 1 つの FAV ファイルに保持することができる。これにより従来各工程に分散していた他の工程から見えなかった 3D データや技術者のノウハウを統合・共有し、効率的に一元管理することが可能になり、生産性の向上や技術伝承が見込める。

9.5.3. ボクセルデータの課題と対応

ボクセルベースのデータフォーマットでは、きめ細かな表現をするためにボクセルサイズを小さくすると(解像度を上げれば)データサイズが大きくなる。また、FAV では材料情報、色情報、リンク情報、ユーザー定義情報などこれまでの 3D データより多くの情報を保持するため、さらにデータサイズが大きくなる要因を持つ。

ボクセルで立体物を表現することは、2 次元画像をピクセルで表現する考え方の延長であり、2 次元画像におけるデータ圧縮技術を、ボクセルベースのデータの圧縮に適用することが期待できる。また 2 章で述べたように、小さなボクセルにより細かな形状を表現する FAV ファイルを外部に作成し参照することで、本体 FAV ファイルのサイズを小さくすることが出来る。

滑らかな外形を得るためにボクセルサイズを小さくしなくても、マーチングキューブや SDF(Signed Distance Field)といった手法により、ボクセルで表現された外形から曲面を形成(再現)する方法が古くから提案されている⁷⁾。

現在、ボクセルデータを直接サポートしている 3D プリンタは無い(2020 年 1 月現在)。ボクセルベースのデータに、材料や色などのリッチな情報を保持し、複雑な内部構造が表現できたとしても、3D プリンタで出力するために 3D プリンタが対応しているフォーマット(例えば STL)に変換すると、多くの情報が欠落する可能性がある。近年、3D プリンタによる造形において、ボクセル表現の有用性をアピールする発表も増えてきている^{8,9)}。また、30 万点を超えるボクセルベース(FAV)の 3D データの蓄積や公開も進んでおり^{10,11)}、近い将来、ボクセルベースの 3D データの流通が増え、リッチな情報を持ったボクセルデータを直接出力できる 3D プリンタが登場し、3D データ、および 3D プリンタが豊かな社会生活の実現に貢献することを期待する。

脚注)

*1 STL: Stereolithography の略。

*2 AMF: Additive Manufacturing File Format の略。2011 に ver.1.0 が策定された。

*3 3MF: 3D Manufacturing Format の略。2015 年に ver.1.0 が策定された。

*4 FAV: Fabricatable Voxel の略。ver.1.1a が以下より公開されている。2019 年に JIS に制定された。
<https://www.fujixerox.co.jp/company/technical/communication/3d/fav.html>

*5 DICOM: Digital Imaging and Communications in Medicine の略。CT や MRI などで撮影された医療用画像用のフォーマット。

参考文献

- 1) ASTM International, 「Standard Terminology for Additive Manufacturing Technologies (F2792-12a)」。West Conshohocken, PA, (2012)
- 2) JIS B 9442, 「3D モデル用 FAV フォーマットの仕様」
- 3) 高橋 智也, 升森 敦士, 藤井 雅彦, 田中 浩也, 「ボクセルベース 3D データフォーマット FAV (Fab-able Voxel) による内部構造と属性の表現」, Proceeding of Imaging Conference Japan 2016, 33 (2016)。
- 4) Tomonari Takahashi, Atsushi Masumori, Masahiko Fujii and Hiroya Tanaka, 「FAV File Format Specification Version 1.1a」(2019) (<https://www.fujixerox.co.jp/company/technical/communication/3d/fav.html>)

- 5) 藤井 雅彦, 「3D プリンタ活用の鍵となるボクセルベースのデータフォーマット「FAV」」, プラスチックエージ Vol. 65 (2019)
- 6) 田中 浩也, 齋藤 和行, 守矢 拓海, 「3D プリンティングにまつわる情報学的研究課題」, 情報管理, 60(6), 403 (2017)
- 7) William E. Lorensen, Harvey E. Cline, 「Marching Cubes: A high resolution 3D surface construction algorithm」, Computer Graphics, 21(4) (1987)
- 8) Tim Weber, 「HP's Jet Fusion 3D Printing Technology—Enabling the Next Industrial Revolution」, Proceeding of Printing for Fabrication 2017, 2 (2016)
- 9) E.L. Doubrovski, E.Y. Tsai, D. Dikovsky, J.M.P. Geraedts, H. Herr, N. Oxman, 「Voxel-based fabrication through material property mapping: A design method for bitmap printing」, Computer-Aided Design 60 (March), 3 (2015)
- 10) 田中浩也, Fabbing Our City, 2019 Tanaka Lab Project Book.
- 11) <https://fab3d.cc/>