

Title	入力3Dモデルの近似形状を持つ3D歯車集合体モデルの生成
Author(s)	下中, 進
Citation	
Issue Date	2013-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/11287
Rights	
Description	Supervisor:宮田 一乗, 知識科学研究科, 修士

修 士 論 文

入力 3D モデルの近似形状を持つ 3D 歯車集合体モデルの生成

指導教員 宮田 一乗 教授

北陸先端科学技術大学院大学
知識科学研究科知識科学専攻

1050023 下中 進

審査委員： 宮田 一乗 教授（主査）
西本 一志 教授
藤波 努 准教授
林 幸雄 准教授

2013 年 2 月

目 次

第 1 章 序論	1
1.1 研究背景	1
1.1.1 Non-Photorealistic Rendering	1
1.1.2 Non-Realistic Expressive Modeling	2
1.1.3 シンボルとしての歯車	4
1.2 課題と本研究の目的	6
1.2.1 歯車を NREM に用いる場合の問題点	6
1.2.2 本研究の目的	8
1.3 関連研究.....	9
1.3.1 NREM に関する研究	9
1.3.2 歯車集合体の生成手法に関する研究	10
1.4 本研究の意義と独創的な点	13
第 2 章 歯車集合体の生成について	14
2.1 歯車の種類・規格	14
2.2 歯車集合体の生成手法	16
2.2.1 断面を得る処理	18
2.2.2 断面に歯車を配置する処理	22
2.2.2.1 1 つ目の歯車セットの S1 計算	25
2.2.2.2 2 つ目以降の歯車セットのスコア S2 計算	28
2.2.2.3 歯車セットスコアに関して	28
2.2.3 歯車が配置されていない領域の発生の抑制	29
2.2.4 描画のためのメッシュの生成	35
2.3 全数処理と簡易処理の比較実験	42
第 3 章 結果および考察	45
3.1 本手法の結果	45

3.2 結果の考察	62
第 4 章 まとめ	64
謝辞	66

図 目 次

図 1	NPR 研究の例	2
図 2	Arcimboldo の作品	3
図 3	Ran らの研究[1]の作品例：[1]からの引用図.....	4
図 4	歯車をシンボルとして用いている例	5
図 5	Office2010 に登録されている歯車を用いたクリップアートの例	6
図 6	Mike Libby[10]の作品例.....	6
図 7	歯車の特徴を明示したクリップアートの例	8
図 8	本研究の目的.....	9
図 9	NREM の関する研究の例.....	10
図 10	歯車集合体の生成手法の概略 入力モデルに複数の YZ 平面レイヤを設置，各レイヤ の位置における入力モデルの断面を抽出，抽出した各断面に歯車を配置する．	10
図 11	タイルモザイク生成に関する研究例.....	11
図 12	断面の 3 次元空間位置 断面の XYZ 座標軸は入力モデルの座標軸と同じである．	12
図 13	インボリュート標準平歯車	15
図 14	歯先円	15
図 15	歯車の歯幅.....	15
図 16	本手法の大まかな流れ 1)入力 3D モデルに YZ 平面レイヤを設置し，各レイヤ位置 の平面状断面を抽出する．2)抽出した断面に歯車セットをかみ合わせながら 1 つずつ配 置していく．	17
図 17	歯車セットの図 本研究では 1 つ以上の歯車が同じ歯車軸に固定された状態のもの を歯車セットと呼ぶ．	17
図 18	歯車セットの配置 歯車セットを構成する歯車 1 つをすでに置かれた歯車 1 つにか み合うように配置していく．	18
図 19	入力モデルの XYZ 空間への配置とサイズの正規化後，平行移動後の図 ZX 平面と ZY 平面上の一辺が F で原点を頂点として持つ正方形からなる直方体内に入力モデルが	

内包される．	19
図 20 XYZ 空間のレイヤ位置とテクセル化の概要図 入力モデルのレイヤで区切った部分 に対してテクセル化を行う．	20
図 21 各レイヤ位置での入力モデルのテクセル化の処理	21
図 22 XYZ 空間内のグリッド平面の空間位置	22
図 23 歯車を配置する手順を表すフローチャート	23
図 24 すべてのセルの中心位置で計算された歯車セットスコア S_1 で 1 つ目の歯車セットを 配置する手順の概要	24
図 25 レイヤ範囲 $[X_u, X_d]$ を算出する処理の概略 (a)(b)各レイヤ内で径 d_g の歯車軸 (図中の 橙丸) が断面内に配置できるかテクセル値で判定する (c)歯車軸が存在でき、 X_c と連続 するレイヤ内で終端に位置するレイヤを X_u, X_d とする．	26
図 26 F_{X_s} の計算方法 C_{X_s} が断面内でなくなるまで T_{X_s} に + 1 していく．	27
図 27 歯車集合体生成後の歯車が置かれなかった領域の発生例 (a)のような入力モデルとレイ ヤ位置から(b)のような断面が抽出される．その断面について前述の手法を用いると、 (c)のような歯車セットが 1 つ目に配置され、(d)のような歯車集合体が生成される．	30
図 28 「歯車集合体生成後の歯車の配置されていない領域」が発生する原因 (a):図 27(b) の断面における「跨ぎセル」の位置 (b): 図 27(b)の断面で異なるレイヤ間の領域を連結 できる歯車セットの例 (c): 図 27(b)の断面に図のような歯車セットを置いた場合の「ま だ歯車が配置されていない領域」 (d):(c)の「未跨ぎセル」の位置	31
図 29 「跨ぎセル」抽出の処理	33
図 30 歯車の外形	36
図 31 歯車の外形をなす各部位	37
図 32 歯車の外形の分割	38
図 33 歯車の中心点と接点で 3 角形メッシュを構築	38
図 34 歯車の形状メッシュ	39
図 35 直径 d_g の円の分割	40
図 36 直径 d_g の円の中心点と接点で 3 角形メッシュを構築	40
図 37 歯車軸の形状メッシュ	41
図 38 全数処理と簡易処理を用いた手法で生成した歯車集合体の例 1	43
図 39 全数処理と簡易処理を用いた手法で生成した歯車集合体の例 2	44

図 40	生成モデル例 1	48
図 41	生成モデル例 2	49
図 42	生成モデル例 3	50
図 43	生成モデル例 4	51
図 44	生成モデル例 5	52
図 45	生成モデル例 6	53
図 46	生成モデル例 7	54
図 47	生成モデル例 8	55
図 48	生成モデル例 9	56
図 49	生成モデル例 10	57
図 50	生成モデルの一部拡大図.....	58
図 51	生成モデルに金属の反射を適応して描画した例 1.....	59
図 52	生成モデルに金属の反射を適応して描画した例 2.....	60
図 53	生成モデルに金属の反射を適応して描画した例 3.....	61
図 54	断面に歯車が置かれなかった断面とのみ X 方向に連続する断面で起きる問題	62
図 55	歯車軸を保持する方法	65

表 目 次

表 1	全数処理と簡易処理を用いた手法での計算時間	44
表 2	生成モデルの計算時間と入力パラメータ	46

第 1 章 序論

この章では、まず本研究の背景について言及する。研究背景として、Computer Graphics 分野における表現手法の動向と歯車のシンボル性について言及する。次に研究背景から導きだされた問題点および、本研究の目的について述べる。つづいて、本研究の関連研究について触れ、最後に本研究の意義と本研究の独創的な点について言及する。

1.1 研究背景

この節では、本研究の背景として、「Non-Photorealistic Rendering」, 「Non-Realistic Expressive Modeling」, 「シンボルとしての歯車」について言及する。

1.1.1 Non-Photorealistic Rendering

Computer Graphics(CG)分野では、3D モデルなどの幾何情報を現実世界での見えに近くなるように表現することを目指す Photorealistic Rendering 分野と、制作者が意図する表現で描画することを目指す Non-Photorealistic Rendering(NPR)分野が存在する。NPR 分野の研究では、幾何情報を点描画風、油絵風などに描画する手法に代表される実在の芸術表現に関する手法の研究や、ユーザに理解されやすくする表現で幾何情報を描画する手法のような芸術的な表現を第一義としない研究などが行われている。

前者の研究の例として、

- ・ マルチエージェントシステムを用いて油絵風や点描画風、手描き風に入力画像を描画する手法[1]
- ・ 「representational calligraphy」と呼ばれる手描き風の文字の表現を持つように入力画像を描画する手法[2]
- ・ 入力画像を複数の任意のタイルを組み合わせて表現する手法[3]
- ・ 入力動画をタイルモザイク表現を持つアニメーションに変換する手法[4]

後者の研究の例として、

- ・ 機構を構成する幾何モデルを入力とし、その機構の動作に対するユーザの理解を支援するために矢印などを描画した静止画やアニメーションを作成する手法[5]

などの研究例が挙げられる。図 1にそれぞれの出力結果を示す。

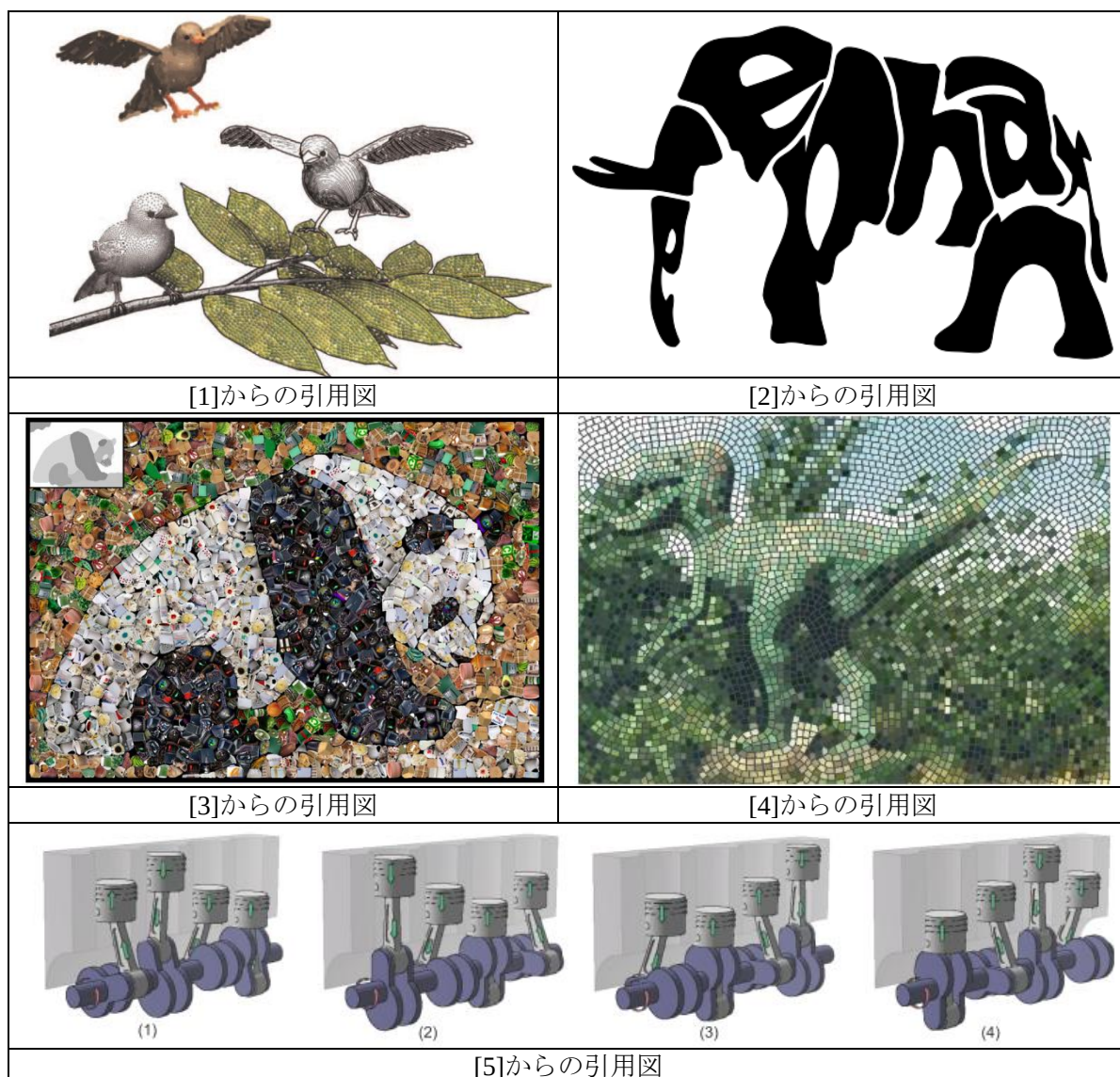


図 1 NPR 研究の例

1.1.2 Non-Realistic Expressive Modeling

一方, Ran らの研究[6]では, 描画で 3D モデルに意図した表現を持たせる一般的な NPR の手法ではなく, 3D モデルの形状自体で意図した表現を持つように 3D モデルをモデリングする手法: 「Non-Realistic Expressive Modeling(以降, NREM と称する)」を提唱している。

Ran らが提案したモデリング手法では, 16 世紀の画家 Arcimboldo[1527-1593]の画風を利用している。Arcimboldo の画風は, 図 2に示すように, 野菜や本などの構成要素を人物の形になるように組み合わせ, その集合体で人物を写術的でない表現で描画するものである。その画風は, 図 2

のように、異なる構成要素を用いると、異なる表現で描画できる。

この画風を参考に、Ran らの手法では、制作者は集合体用の 3D モデルと構成要素となる複数の 3D モデルを用意する。そして、構成要素用のモデルを組み合わせて、集合体用のモデルの形状を持つ集合体を生成する。この手法の作品例を図 3 に示す。この手法は、Arcimboldo の画風の画風と同じく、異なる構成要素用のモデルを用いることで異なる表現ができる。

以上で述べた Ran らの手法に関連する研究が、いくつか報告されている[7][8][9]。それらの研究は研究目的として NREM を掲げてはいないが、同様に Arcimboldo の画風のような表現を示せていることから、NREM に関するモデリング手法と言える。

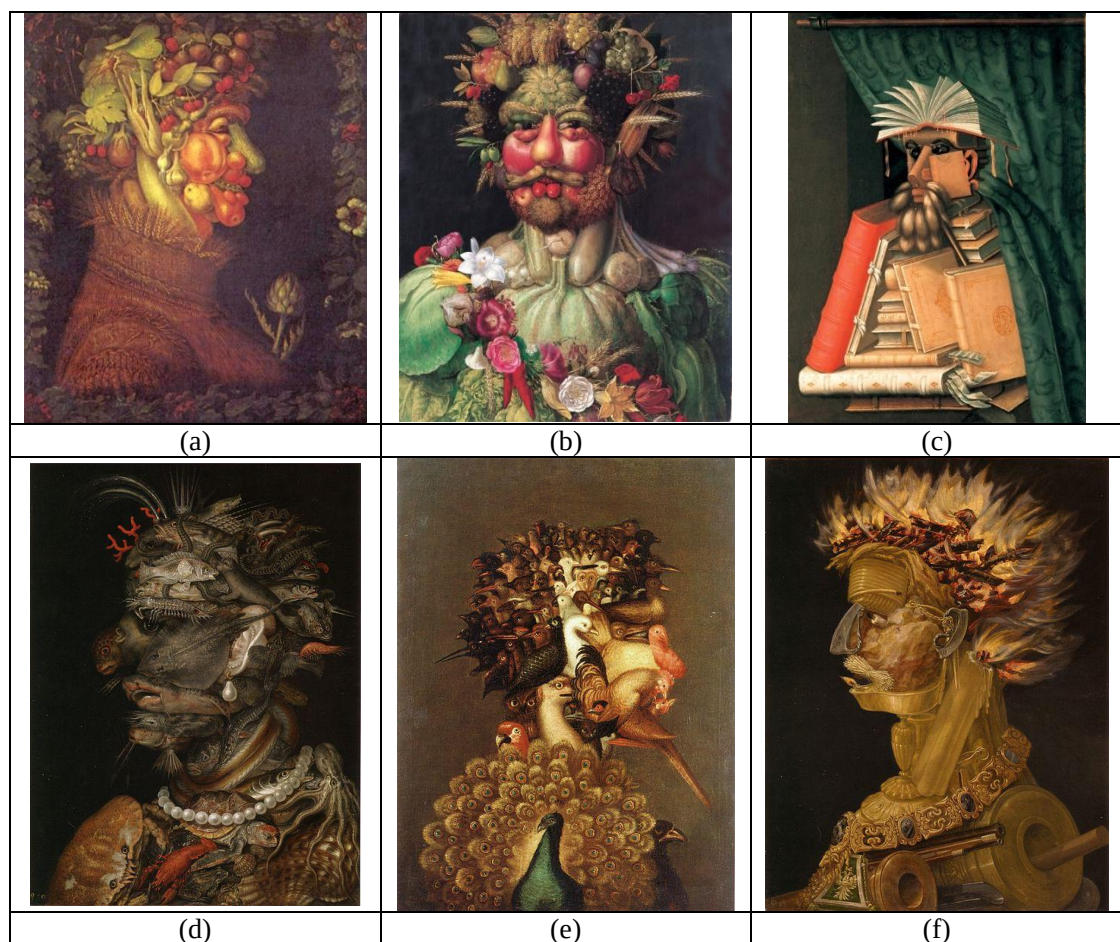


図 2 Arcimboldo の作品

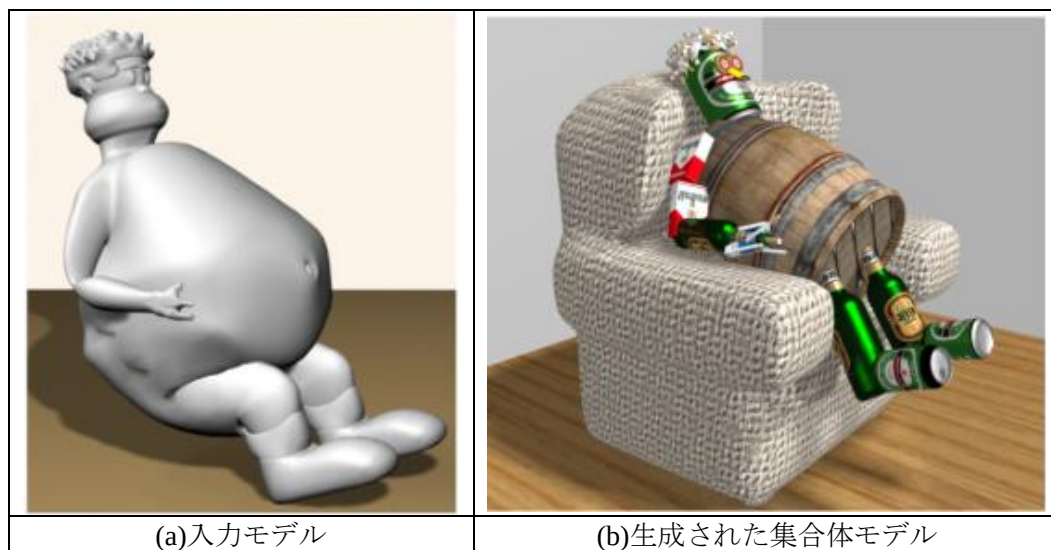


図 3 Ran らの研究[1]の作品例：[1]からの引用図

1.1.3 シンボルとしての歯車

歯車は機械要素の一種であり，主に動力の伝達に用いられる．歯車の周囲には歯が形づけられており，複数の歯車間の歯のかみ合いによって，1 軸から他の軸に回転という形で動力を伝達することができる．また，歯車の組み合わせによって，回転の減速，増速，回転方向の変換，動力の分割などを行うことができる．

歯車は機械を構成する要素として，工作機や加工機といった工業的な機械の他に，時計や簡単な機構を持つ玩具などの身近な機械にもよく用いられている．そのため，機械要素について馴染みのない一般人でも歯車を目にする機会がある．また，歯車が持つ独特で単純な形状から，一般人でも歯車の認識は容易である．一方，図 4に示すように歯車は国旗，国章，社章に「工業」「労働」「労働者」を表すシンボルとして利用されている．また，図 5に示すように，マイクロソフト社の商用オフィススイートである「Office 2010」に歯車を用いた構図のクリップアートが多数登録されている．

映画“モダン・タイムス“では，あるシーンが機械の内部であることを表現するために歯車を映画のセットの一部として使用している．また，アーティスト Mike Libby[10]は昆虫の標本に歯車やばねなどの機械要素を組み込むことにより，機械から受け取れる印象を昆虫に付加して図 6に示すような insect-like robot や robot-like insect を表現している．

以上で述べた事例のように，歯車はシンボルとしてさまざまな面で有用であると考えられる．

	
(a)五円硬貨	(b)イタリアの国章
	
(c)中華人民共和国の国章	(d)モンゴル人民共和国の国章
	
(e)アンゴラ共和国の国旗	(f)アンゴラ共和国の国章
	
(g)ベトナム社会主義共和国の国章	(h)ラオス人民民主共和国の国章
	
(i)ビルマ連邦社会主義共和国の国旗	(j)ビルマ連邦社会主義共和国の国章

図 4 歯車をシンボルとして用いている例



図 5 Office2010 に登録されている歯車を用いたクリップアートの例



図 6 Mike Libby[10]の作品例

1.2 課題と本研究の目的

この節では、まず 1.1 節で説明した研究背景から導きだせる課題について述べる．次にその課題を踏まえた本研究の目的について言及する．

1.2.1 歯車を NREM に用いる場合の問題点

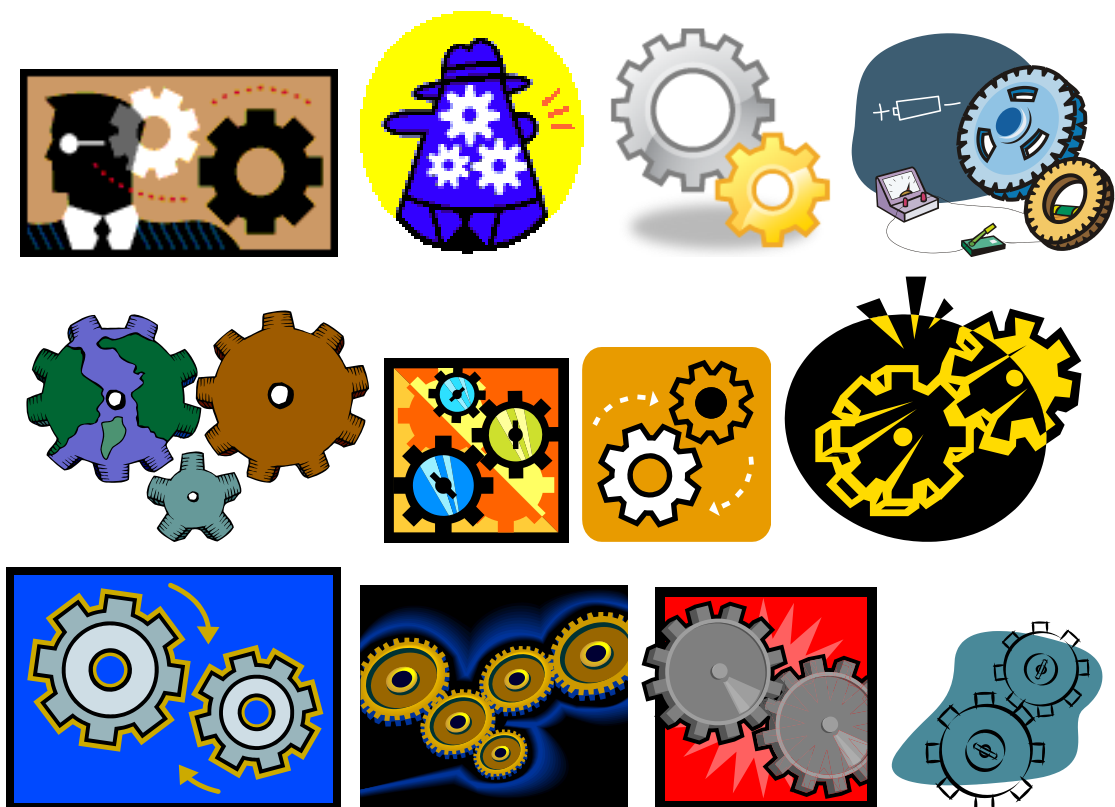
1.1.3 項で示したように、歯車はシンボルとして有用である．シンボルはそれが示唆する意図を

表現する用途で用いられることが多いため、NREM で歯車を用いることは、多様な表現に有益であると言える。

NREM に関する先行研究[6][7]を用いれば、構成要素として歯車を用いた歯車集合体を生成することができる。しかしながら、生成されたモデルは、歯車同士のかみ合いを考慮せずに、乱雑に歯車を配置、もしくは堆積された状態である。

一方、シンボルとして用いられる歯車のいくつかは、歯車同士のかみ合いなどの歯車の特徴を明示した構図で用いられている。「Office 2010」のクリップアート内から、その構図の例を図 7に示す。これらの例から、歯車のかみ合いを考慮した構図は、NREM にとっても有用な表現であると考ええる。

本論文では、先行手法では表現が困難である歯車のかみ合いを考慮した NREM 手法の提案を研究の目的とする。



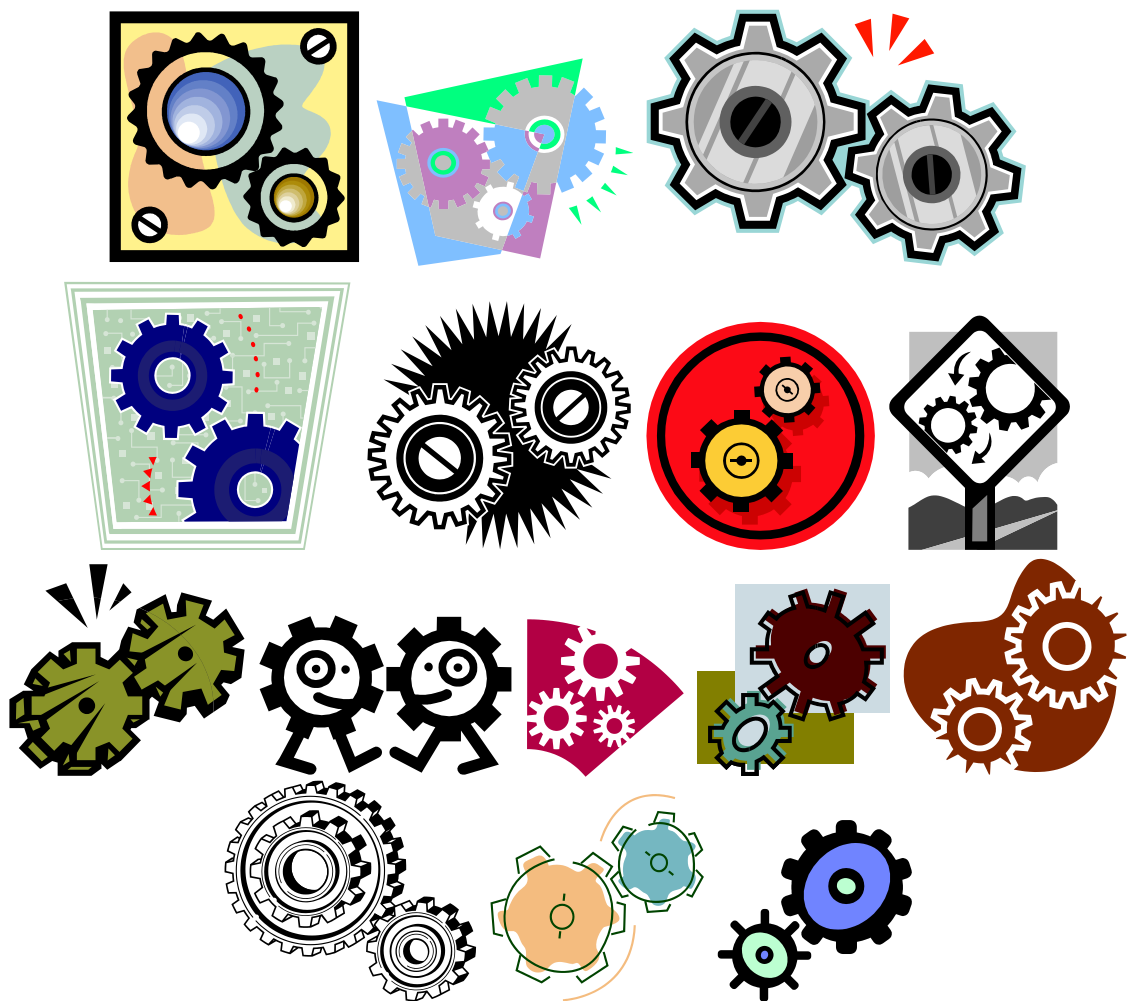


図 7 歯車の特徴を明示したクリップアートの例

1.2.2 本研究の目的

図 8に示すように、本手法では、Ran らの手法[6]と同様に、入力した 3D モデルの近似形状を形作るような構成要素の集合体を生成する。

本研究では構成要素を歯車に限定し、以下に示すような歯車の特徴を表現する。

1. 歯車は歯車同士のかみ合いによって回転を伝達する
2. 1つの歯車の回転を次々と他の歯車へ伝達する
3. 異なる大きさの歯車同士でもかみ合せ、大きさの比率によって回転速度を制御する

これらの特徴を視覚的に表現するため、歯車集合体を以下の状態にする。

1. 構成する歯車間にかみ合いがある
2. 構成する歯車を 1つ回転させるとすべての歯車が回転可能な状態である

3. 集合体を大きさの異なる歯車で構成する

ユーザが歯車の性質や特徴を考慮しながら手動で歯車集合体をモデリングするには、大きな労力が必要になる。この問題に対して、本研究では自動的に歯車集合体を生成することで、ユーザの労力の軽減を図る。

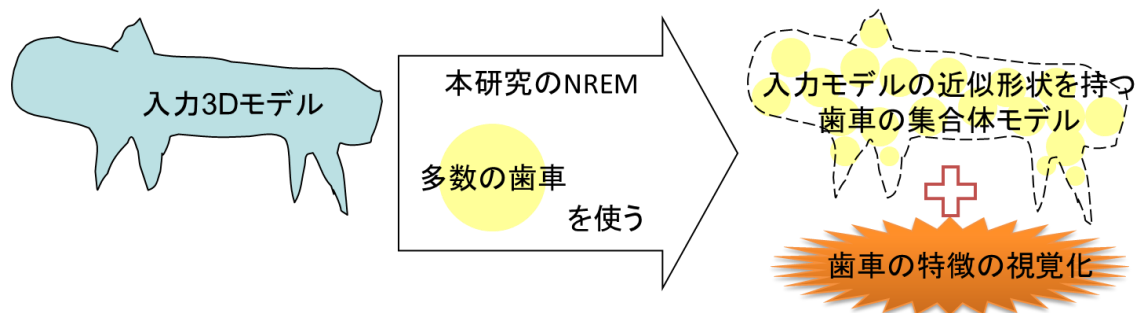


図 8 本研究の目的

1.3 関連研究

本研究の関連研究として、NREM に関する研究および、歯車集合体の生成手法に関する研究について言及する。

1.3.1 NREM に関する研究

NREM に関する研究は、その定義から、指定した 3D モデルを別な 3D モデルへと変換する研究と言える。関連する研究として、3D モデルを物理的に制作できる形に生成することを目指す fabrication 分野、対象となる 3D モデル通りに構成要素を堆積させることを目指す photorealistic modeling 分野などを含めると多数提案されている。例えば、実際に配置可能な缶の積み上げ集合体に変換する手法[8]、複数の折り紙から構成されるユニット折り紙に変換する手法[9]、入力モデルの形状を保持した Polyomino Puzzle を生成する手法[11]、複数の任意の 3D モデルを指定した 3D モデルの形状に合うように堆積させる手法[7]、などが先行研究として挙げられる。図 9にそれぞれの出力を示す。

しかしながら、先行研究の手法では歯車のかみ合わせのような他の構成要素との依存関係を考慮していない。

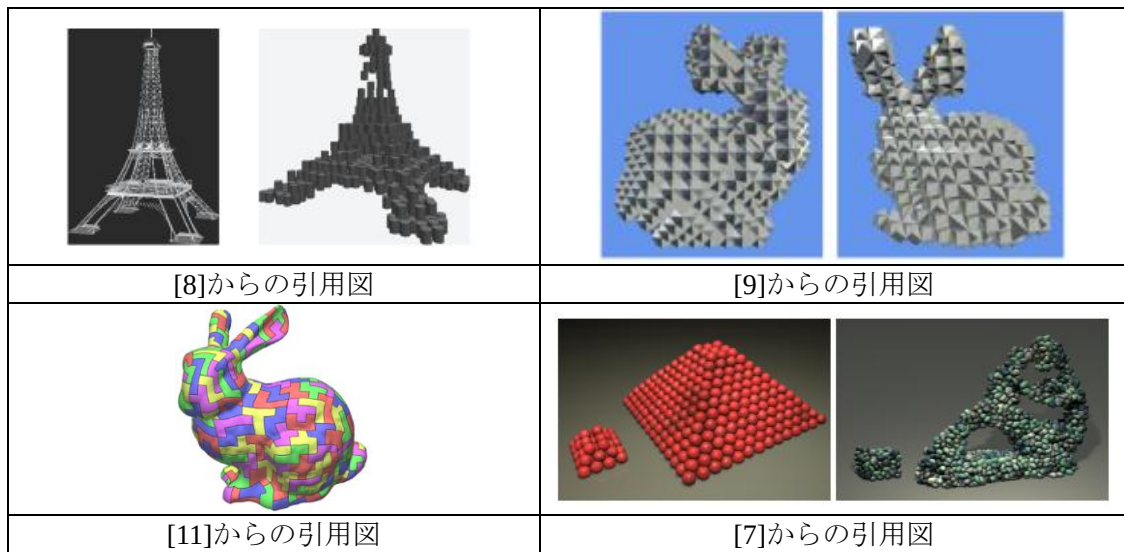


図 9 NREM の関する研究の例

1.3.2 歯車集合体の生成手法に関する研究

本手法では、図 10に示すように、入力 3D モデルを平行な平面レイヤで区切り、各断面上に歯車を敷き詰める。この処理により、近似形状を持つ歯車集合体を生成する。

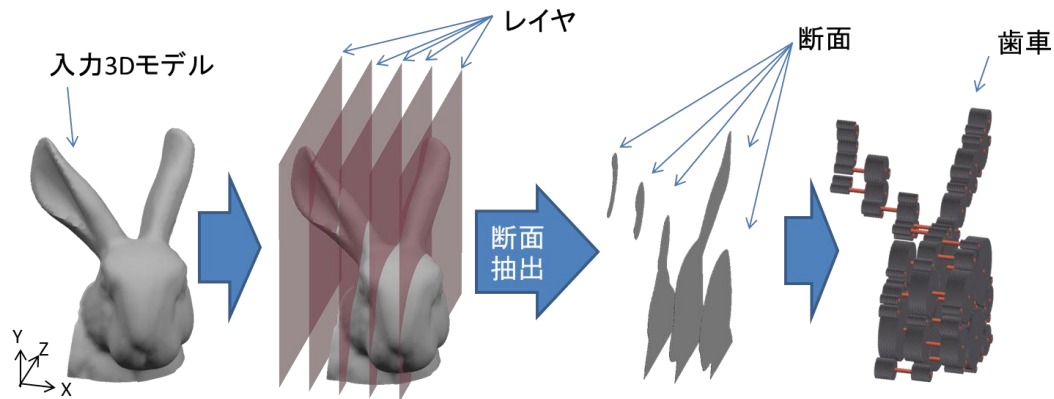


図 10 歯車集合体の生成手法の概略 入力モデルに複数の YZ 平面レイヤを設置、各レイヤの位置における入力モデルの断面を抽出、抽出した各断面に歯車を配置する。

平面内での要素の敷き詰め法として、タイルモザイク生成手法に関する研究[12][13][14][15]が多数報告されている。これらの手法では、任意の境界形状を持つ 2 次元平面内にタイルを所望のタイルモザイクになるように敷き詰める。

タイルモザイクの生成手法においては、以下に示す敷き詰め方法を主として用いている。

1. 多数のタイルを任意の境界形状を持つ 2 次元平面内にランダムに敷き詰める。
2. 所望のタイルモザイクになるようにタイルの敷き詰めを定義した評価基準をもとに各タイルを移動、または回転させる。
3. 2 の処理を各タイルの位置・姿勢が収束するまで繰り返す

上記の方法に関する研究で生成されたタイルモザイクの例を図 11に示す。これらの研究では、構成要素間の隙間を考慮しているが、本研究では、歯車のかみ合い、すなわち構成要素間の接触といった局所的な関係と、1つの歯車の回転ですべての歯車が回転するといった構成要素間の大域的な関係を考慮する必要がある。したがって、タイルモザイク生成の手法をそのまま本研究に適用することはできない。また、1つの歯車の回転ですべての歯車を回転させるために、別レイヤに配置した歯車に軸を介して回転を伝える必要がある。このため、図 12に示すような 3 次元空間での各断面の位置関係を本手法は考慮して歯車を配置する。3 次元空間にタイルモザイクを生成する研究もおこなわれているが、それは 3D モデルの表面上にのみタイルモザイクを生成する手法[17][18]であり、本研究のように 3D モデルの内部まで考慮した空間構造とは異なるものである。

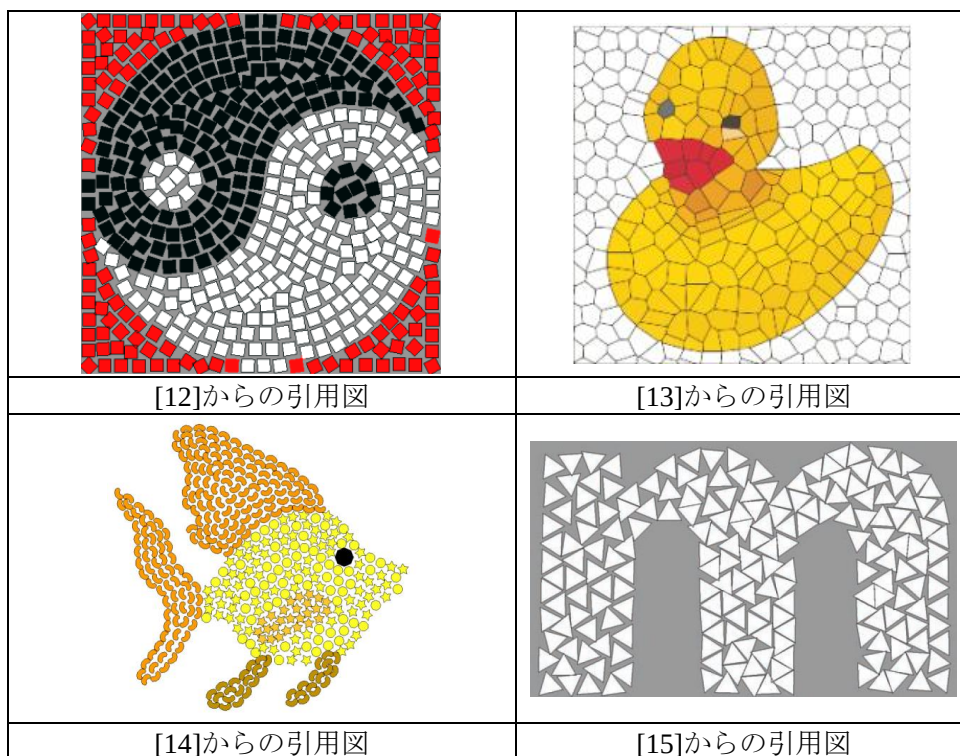


図 11 タイルモザイク生成に関する研究例

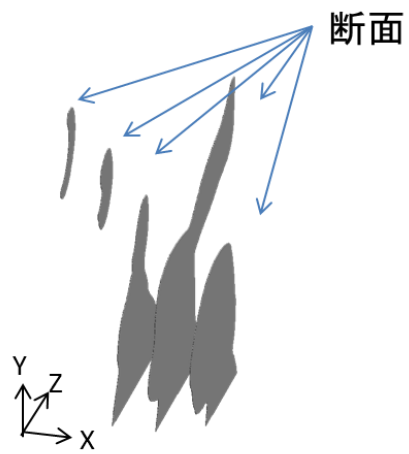


図 12 断面の 3 次元空間位置 断面の XYZ 座標軸は入力モデルの座標軸と同じである。

1.4 本研究の意義と独創的な点

本研究の独創的な点として、部品間の依存関係を考慮し、回転を伝えられるような構造を持つ複雑な歯車集合体モデルを生成できる点が挙げられる。既存研究でも、入力モデルの近似形状を持つ歯車集合体を生成できるが、その集合体は歯車が堆積したような乱雑な状態や、物理的干渉をさけることのみ考慮した歯車の集合体である。

本研究の成果を用いることで表現の幅を広げることにより貢献でき、この点に本研究の意義があると考えられる。

第 2 章 歯車集合体の生成について

この章では、まず本手法で用いる歯車の種類・規格について述べ、つづいて、その規格を利用した歯車集合体の生成手法について言及する。歯車の規格を利用することで歯車のかみ合いなどを考慮する。最後に、計算時間の比較実験を示す。

2.1 歯車の種類・規格

本手法では、歯車集合体を構成する歯車として、日本工業規格[23]で規格化された工業で一般的に使用されている「インボリュート標準平歯車」を用いる。ここで、インボリュート標準平歯車とは、インボリュート曲線による歯形を持つ平歯車を指す。平歯車は、歯車軸に平行な他の歯車にかみ合い動力を伝達する歯車である。図 13にインボリュート標準平歯車を示す。本論文では、以降断りがない限り、歯車とはインボリュート標準平歯車を指すものとする。

本手法では、使用に耐えうる歯車の強度や工業的に使用される寸法など、製造に必要な物理値は考慮せず、歯数や歯の大きさ、かみ合わせの位置の規格のみを考慮する。したがって、歯車のピッチ円直径 d 、モジュール m 、歯数 T の 3 つのパラメータで、歯車の形状を定義する。ピッチ円直径は歯車をかみ合あわせるための条件に用いられるピッチ円の直径を、モジュールは歯の大きさを、歯数は歯の数をそれぞれ示しており、これらの関係は $d = m T$ で表すことができる。ここで、歯数は自然数でなければならない。また規格より、図 14に示す歯車の歯先円直径(最外径) C は $C = d + 2m$ で定義され、本研究では歯車の干渉判定に使用する。同じく規格より、歯の根元を通る歯底円の直径 d_f は $d_f = d - 2.5m$ で定義される。ここで、ピッチ円、歯先円、歯底円の中心位置は同じである。

2 つの歯車がかみ合うには互いのピッチ円が一点で接触しており、かつ互いにモジュールが同じであるという条件を満たす必要がある。本手法では、モジュールの値はユーザが初めに指定した 1 つの値（正の数）をすべての歯車で使用する。これにより、歯車がかみ合うために必要な条件は「互いのピッチ円が接触する」だけになり、円の敷詰め問題に帰着する。

歯車が他の歯車と噛み合うことで回転を伝える場合、歯車同士の歯面の接触によって回転を伝達している。このとき、常に互いの 1 つ以上の歯ごとに接触しているので、回転を連続的に伝えることができる[16]。しかし、歯数が小さい歯車を加工する際に、加工工具によって歯の根元部分

が削られ、歯の形が変形する切り下げ[16]と呼ばれる不都合が発生し、かみ合うことのできる歯面が減りかみ合いが悪くなる、強度が減るなど歯車にとって好ましくないことが生じる。そのため、切り下げを防ぐ目的で転位と呼ばれる歯形の調整が行われている。転位を考慮すると解くべきパラメータが増えるため、問題の単純化のために本研究では転位は扱わず、歯車の最少歯数を切り下げが発生しない値と同じ”17 枚”[16]に設定する。また、図 15に示す歯幅(歯車の厚さ)はすべての歯車で同じ厚みとし、ユーザが 0 以外の正の数を設定する。

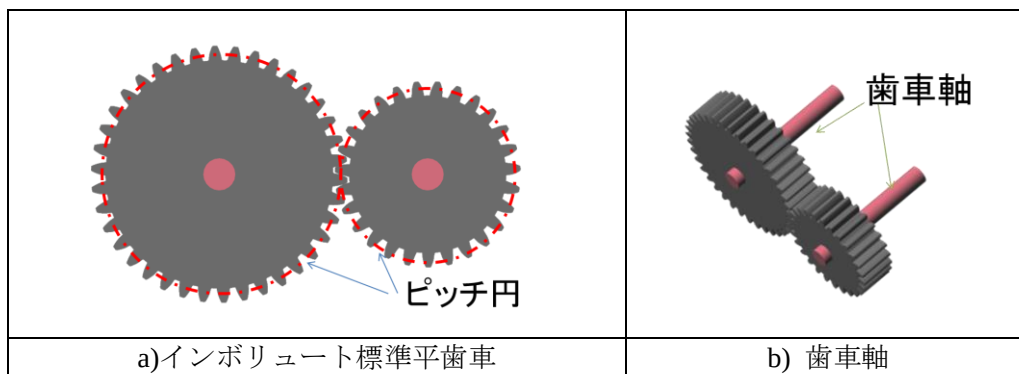


図 13 インボリュート標準平歯車

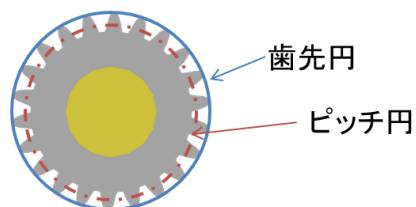


図 14 歯先円

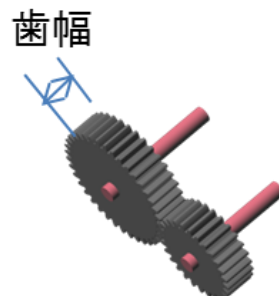


図 15 歯車の歯幅

2.2 歯車集合体の生成手法

歯車の集合体の生成には、以下の2つの手順を踏む。図 16に手順の概略図を示す。

- 1) 図 16(a)に示すような入力 3D モデルに対して、複数の平行した YZ 平面レイヤを図 16(b)に示す等間隔に位置に設置することで、図 16(c)のような各レイヤ位置における断面を得る。
- 2) 得られた各断面内に、図 16(a)-(f)のように複数の歯車を歯車の平面部と断面が平行になる姿勢で、断面内に歯車が配置できなくなるまで配置していく。その結果、図 16(g)に示すような歯車集合体が生成される。

ここで、目的に挙げた「1つの歯車を回転させるとすべての歯車が回転する」という条件がある。このため、配置する歯車は、同レイヤ内での他の歯車への回転の伝達だけでなく、他レイヤへも回転を伝達する必要がある。この条件を満たすため、本手法では歯車単体を1個ずつではなく、図 17に示すような歯車セットを1セットずつ配置する。歯車セットは、1つ以上の歯車と同じ歯車軸に固定しており、構成するいずれかの歯車1つが回転すると歯車軸に固定された他の歯車も回転する。したがって、歯車セットを構成する歯車を異なるレイヤの断面に置くことにより、他レイヤの歯車セットへ回転を伝達することができる。

歯車セットが他の歯車セットに回転を伝達できるように、歯車セットを構成するいずれかの歯車1つを、図 18で示すようにすでに配置された歯車1つとかみ合わせながら、1つずつ配置していく。この配置方法により、配置した歯車セットは必ずかみ合いによって回転するようになり、「1つの歯車の回転ですべての歯車が回転する」状態で歯車を配置することができる。

以降、2.2.1 項で入力モデルから断面を抽出する方法を、2.2.2 項で抽出した断面に歯車を配置する方法について言及する。最後に、2.2.3 項では、歯車集合体を描画するために、計算したパラメータから歯車セットの幾何形状を生成する手法について述べる。

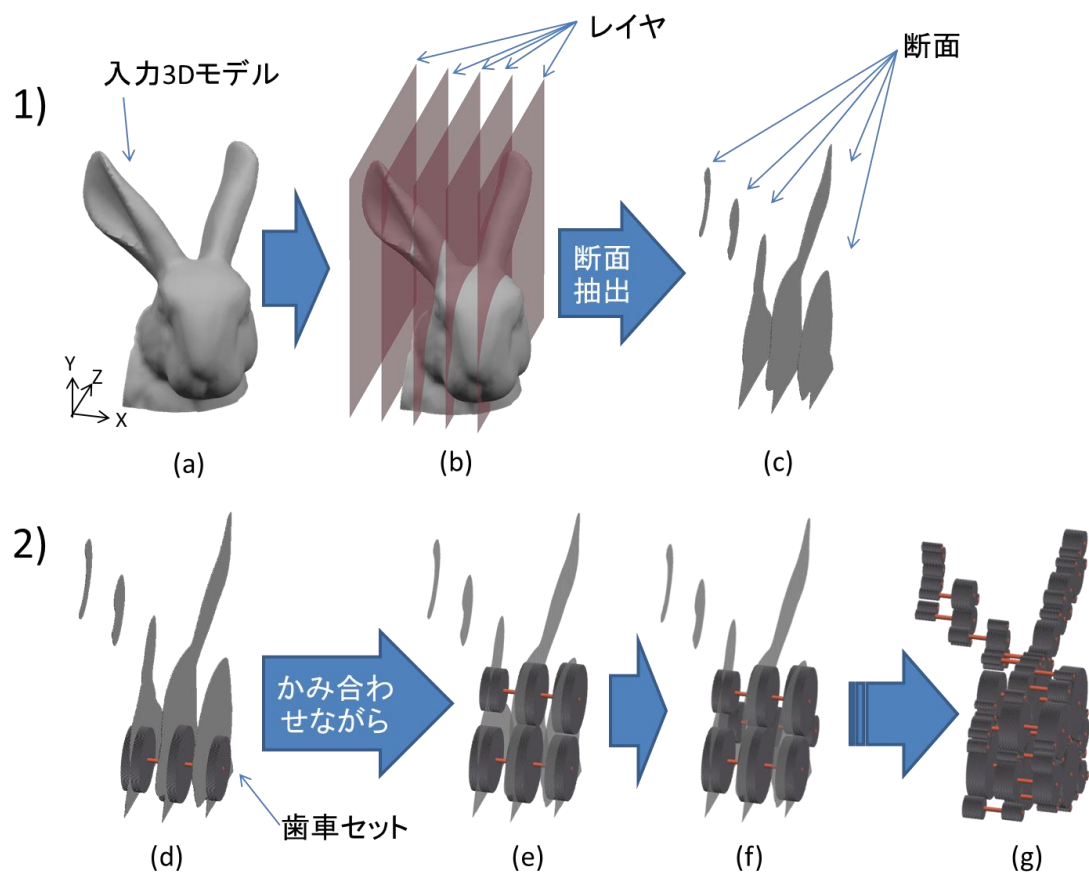


図 16 本手法の大まかな流れ 1)入力 3D モデルに YZ 平面レイヤを設置し，各レイヤ位置の平面状断面を抽出する．2)抽出した断面に歯車セットをかみ合わせながら 1 つずつ配置していく．

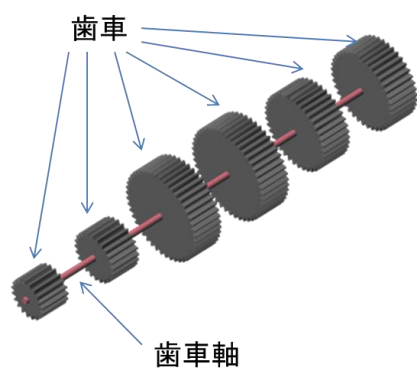


図 17 歯車セットの図 本研究では 1 つ以上の歯車が同じ歯車軸に固定された状態のものを歯車セットと呼ぶ．

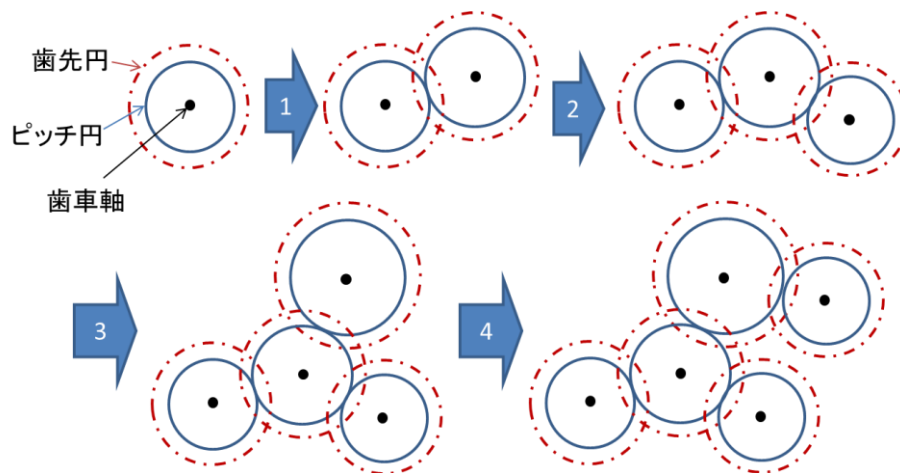


図 18 歯車セットの配置 歯車セットを構成する歯車 1 つをすでに置かれた歯車 1 つにかみ合うように配置していく。

2.2.1 断面を得る処理

入力した 3D モデルを平行なレイヤに分割し、断面を得る方法について説明する。まず前処理として、入力モデルを XYZ 空間に配置し、入力モデルのサイズの正規化、平行移動を以下の手順で行う。ここで、入力モデルの幾何形状は、互いに直交する 3 つの座標軸をもつ 3 次元空間内で、3 次元座標を持つ点群とその点を頂点として持つ面群から構成される 3D サーフェスモデルと定義する。

- 1) 入力モデルの幾何形状を定義している 3 次元空間と同じ座標軸の向き、原点を持つ 3 次元空間を生成する。生成した 3 次元空間の座標軸はそれぞれ X, Y, Z 座標軸とし、生成した空間を XYZ 空間と定義する。そして、XYZ 空間内に入力モデルの幾何形状を定義する点群を配置、面群を構築することで、XYZ 空間内に入力モデルの幾何形状を構築する。
- 2) 入力モデルの Z 軸方向および、Y 軸方向の全長のうち、長い方が正規化サイズ F になるように入力モデルをスケーリングする。F は 0 以外の任意の値を設定するものとし、本研究では $F=5.0$ としている。
- 3) XYZ 空間内に入力モデルを内包する境界ボックスを生成する。境界ボックスは以下の定義を持つ。
 - ・YZ 平面と ZX 平面とからなる直方体である。
 - ・入力モデルを内包する。
 - ・各面は内包する入力モデルの一部と内接する。

- 4) 生成した境界ボックスの最少のX,Y,Z座標値をもつ頂点にXYZ空間の原点を平行移動させる.

以上の手順により, 図 19に示すような XYZ 空間位置にサイズを正規化された入力モデルが配置される.

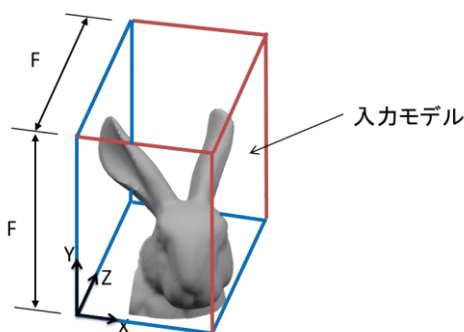


図 19 入力モデルの XYZ 空間への配置とサイズの正規化後, 平行移動後の図 ZX 平面と ZY 平面上の一辺が F で原点を頂点として持つ正方形からなる直方体内に入力モデルが内包される.

次に図 20(a)に示すように, 入力モデルに N 枚の YZ 平面レイヤを設置する. 設置するレイヤ枚数 N は自然数とし, ユーザが設定する. ここで, 原点から i 番目の位置のレイヤは図 20(a)のようにレイヤ番号 $X_r = (i-1)$ を持ち, 各レイヤは X 座標が $(X_r+1) * L_D$ の位置に設置される. L_D は入力モデルの X 軸方向の全長を $(N+1)$ で割った値とする.

そして図 20(b)に示すように, 各レイヤの位置で入力モデルの断面を 2D 平面テクスチャと見立ててテクセル化を行うことで断面の形状を得る.

本研究のテクセル化の処理は以下の手順で行なわれる. 手順について図 21を用いて説明する.

- 1) 各レイヤの X 座標の YZ 平面上で, 一辺が Y 軸に平行, 対角の YZ 座標値が $(0,0) - (F,F)$ の正方形平面を XYZ 空間内に生成する. ここで, 正方形平面は図 21(a)に示すように 1 辺が L_b の正方形セルで構成される正方形グリッドを持つ. L_b はテクセル数 B で F を割った値とする. B の値は自然数であり, ユーザが設定する.
- 2) 図 21(b)に示すように, $Z=0$, 各セルの中心の Y 座標, 各レイヤの X 座標を持つ各点からレイを Z 軸の正の方向へ飛ばす. そして, レイごとに以下の手順の処理を行う.
 - ①) レイと入力モデルとの衝突点の座標を衝突した順に記録する. 記録はレイが正方形平面を超過するまで行う.

②) レイが正方形平面を超過した後，記録した衝突点を衝突した順の 2 点ごとに，図 21(c)に示すようなその 2 点を端点に持つ直線である衝突線を生成する．

③) 図 21(d)に示すように，衝突線を内包するセルにテクセル値として **true** を持たせる．衝突線を内包しないセルにはテクセル値として **false** を持たせる．

以上の処理を行うと，2 値のテクセル値を持つセルの集合が得られる．そして，その集合を用いると，以下のことが言える．

- ・ **true** のテクセル値を持つセルが入力モデルの断面を構成するとした場合，そのセルの集合は図 21(d)のように入力モデルの断面の近似形状を持つ．

以上のことから，本手法では，**true** のテクセル値をもつ各セル内の XYZ 空間位置を入力モデルの断面とする．

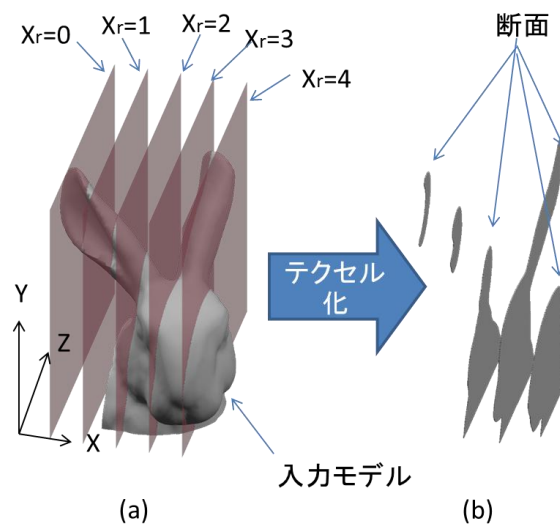


図 20 XYZ 空間のレイヤ位置とテクセル化の概要図 入力モデルのレイヤで区切った部分に対してテクセル化を行う．

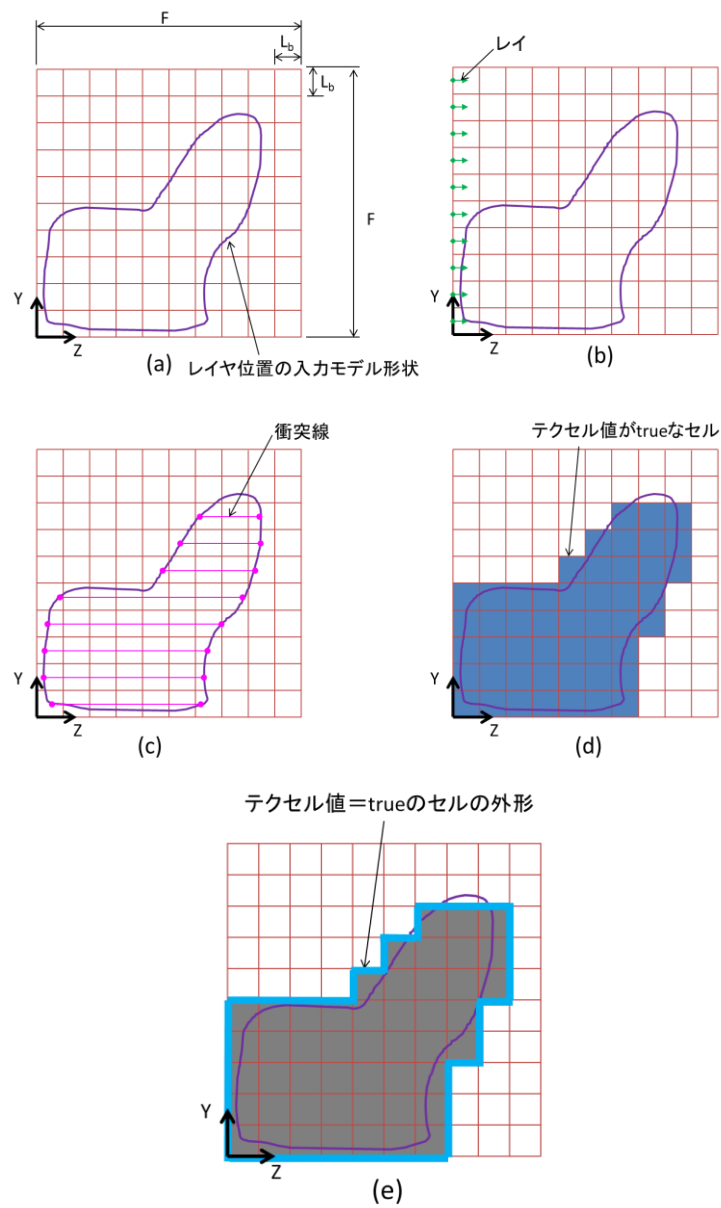


図 21 各レイヤ位置での入力モデルのテクセル化の処理

2.2.2 断面に歯車を配置する処理

次に、2.2.1 項の処理で得た断面に対して歯車を配置する処理について説明する。

まず、XYZ 空間に、図 22 のように 2.2.1 項のテクセル化の処理時に作成した正方形グリッド構造を持つ正方形平面を、テクセル化の処理時と同じサイズで同じ XYZ 位置に生成する。そして、テクセル化で得た各セルのテクセル値を、その生成した正方形グリッド構造平面内の対応する XYZ 位置のセルに持たせる。これによって、各レイヤ位置の YZ 平面内の任意の XYZ 位置が入力モデルの断面内かどうかは、その XYZ 位置を内包するセルのテクセル値を調べることで判断できる。ここで、そのテクセル値が true であった場合は断面内で、それ以外の場合は断面外と定義する。また、true を持つセルと false を持つセルの境界上の XYZ 位置は断面外とする。

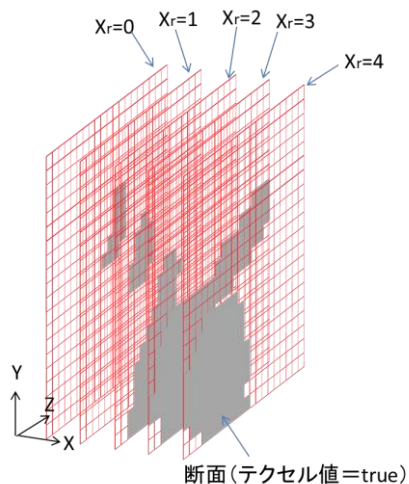


図 22 XYZ 空間内のグリッド平面の空間位置

次に、以下の手順に従い、図 23 に示すように歯車セットを配置する。

- 1) 各セルの中心位置ごとに歯車セットスコア S_1 を計算する。歯車セットスコア S_1 の詳細については 2.2.2.1 で後述するが、計算位置に歯車セットが置けるときのみに、 S_1 は正の値になる。
- 2) すべてのセルの中心位置で S_1 が 0 だった場合、歯車が 1 つも置けないとして歯車配置を終了し、歯車集合体は生成されない。この場合、以下の原因が考えられる。
 - ・入力モデルの断面の大きさが、最小の歯車の径より小さい。
 - ・true をもつセルが存在しない。

以上の状況は B の値を増やす、N の値を増やすなど、ユーザが入力パラメータを見直すことで

対応できる。

- 3) 全セル内で最大 S_1 を持つセルの中心位置に、その S_1 算出時のパラメータの歯車セットを 1 つ配置する。図 24にこの処理の概要を示す。
- 4) 各セルの中心位置ごとに歯車セットスコア S_2 を計算する。 S_2 の計算では、 S_1 と違い以下の点を計算時に考慮している。詳細については 2.2.2.2 で後述する。
 - ・配置済みのいずれかの歯車とかみ合う歯車を持つ歯車セットが計算位置に置けるときのみ、 S_2 は正の数になる。
 - ・配置済みの歯車および、歯車軸と物理的干渉をしない歯車セットが計算位置に置けるときのみ、 S_2 は正の数になる。
- 5) すべてのセルの中心位置で S_2 が 0 だった場合、歯車がもう置けないとして歯車配置を終了する。
- 6) 全セル内で最大 S_2 を持つセルの中心位置に、その S_2 算出時のパラメータの歯車セットを 1 つ配置する。
- 7) 4)に戻る

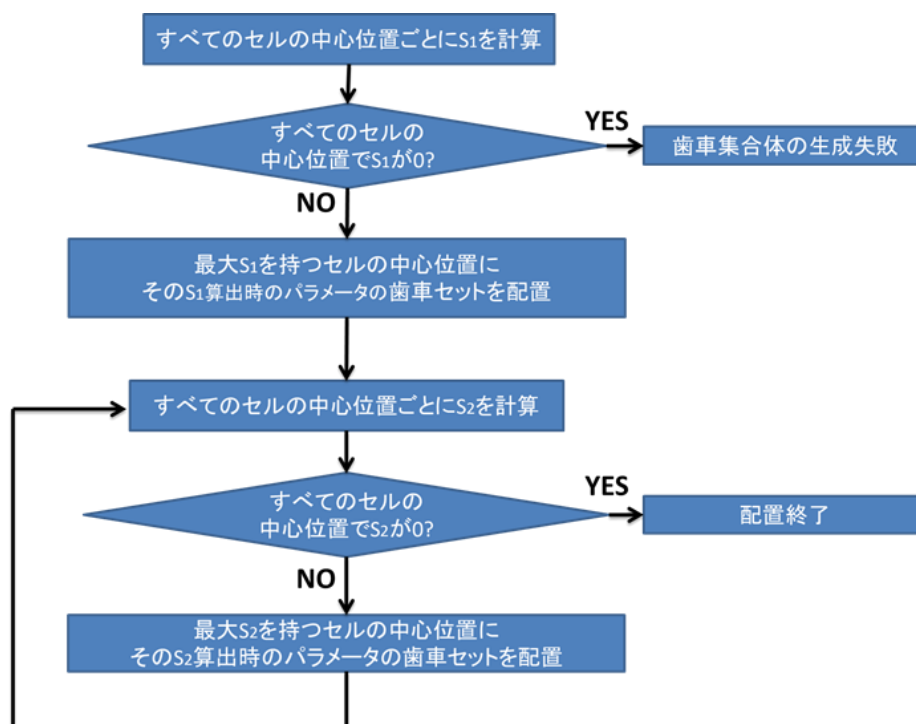


図 23 歯車を配置する手順を表すフローチャート

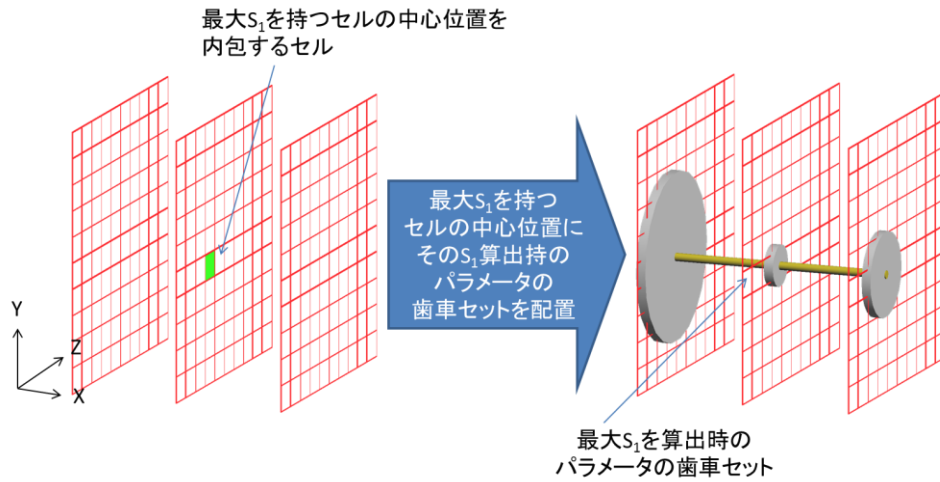


図 24 すべてのセルの中心位置で計算された歯車セットスコア S_1 で 1 つ目の歯車セットを配置する手順の概要

各セルの中心位置で計算する歯車セットスコア S は、詳細な定義は後述するが、基本方針として、以下に記述するように定義している。

- ・ S を計算する空間位置に、断面から歯車の形状の一部でもはみ出でる歯車セットしか置けないときは、 $S = 0$ 。
- ・ S を計算する空間位置に、すでに配置された歯車、歯車軸と物理的干渉する歯車セットしか置けないときは、 $S = 0$ 。
- ・ 上記以外は、 $S > 0$ 。

図 23で示したように、本手法では各セルの中心位置に歯車セットスコア $S > 0$ の歯車セットを、全セルの中心位置で S が 0 になるまで、配置していく。ここで、生成した歯車集合体が入力モデルの近似形状を持つために、本手法は以下の仮定を用いる。

- ① テクセル数 B が十分に大きければ、 ZX 平面上のセルの中心位置は離散的でなく、ほぼ連続的になる。したがって、歯車セットを置ける位置もほぼ連続的になる。
- ② モジュール m が十分に小さく、0 に近ければ、最少歯数の歯車の径は十分に小さくなる。十分に小さい歯車を断面内に配置する場合、配置済みの歯車や、歯車軸を物理的干渉しにくく、かつ断面外へ歯車の形状をはみ出しにくくなる。
- ③ レイヤ枚数 N が十分に多ければ、抽出される入力モデルの YZ 平面上の断面の位置は、 X 軸方向にほぼ連続的である。その入力モデルの断面の集合体は入力モデルの近似形状をもつ。

本手法は、①および②の仮定より、抽出した入力モデルの YZ 平面上の各断面内に、ほぼ隙間な

く歯車が敷き詰めることができ、その YZ 平面上の歯車の集合体はその断面の近似形状をもつ。そして、③の仮定より、入力モデルの各断面に近似した平面状歯車集合体は X 軸方向にほぼ連続であり、平面状歯車集合体の集合体は入力モデルの近似形状を持っている。

従って、本手法が入力モデルの近似形状をもつ歯車集合体を生成するためには、B が十分に大きく、m が 0 に近く、N が十分に大きければならない。

S_1 , S_2 について、以降、2.2.2.1, 2.2.2.2, 2.2.2.3 で詳述する。なお、本手法で求める S_1 , S_2 は局所解であるため、歯車を配置していない領域が発生する場合が生じる。この問題への対策は 2.2.3 項で説明する。

2.2.2.1 1 つ目の歯車セットの S_1 計算

1 つ目に配置する歯車セットに対する S_1 の計算方法を説明する。計算中のセルの中心位置のレイヤ番号を X_c とした場合、そのセルの位置を $P_c=(X_c, Y_c, Z_c)$ とする。

はじめに、直径 d_g で P_c を通り X 軸に平行な歯車軸が断面内に配置できるレイヤ範囲 $[X_u, X_d]$ を算出する。この処理の概略を図 25 に示す。ここで、歯車軸の直径 d_g の大きさはユーザが設定する。まず、 P_c で軸が断面内に配置できるか、図 25(a) に示すように P_c を中心とする直径 d_g の YZ 平面上の円板と接する各セルのテクセル値で判断する。接する各セルのテクセル値がすべて true の場合、軸は断面内に配置できる。配置できない場合、 P_c には歯車セットを配置できないため、 P_c の S_1 を 0 として S_1 の計算を終える。軸を配置できる場合、次に図 25(b)(c) に示すように X_u, X_d を算出する。歯車軸は X 軸と平行に配置するので P_c と同じ Y, Z 座標を通るとする。

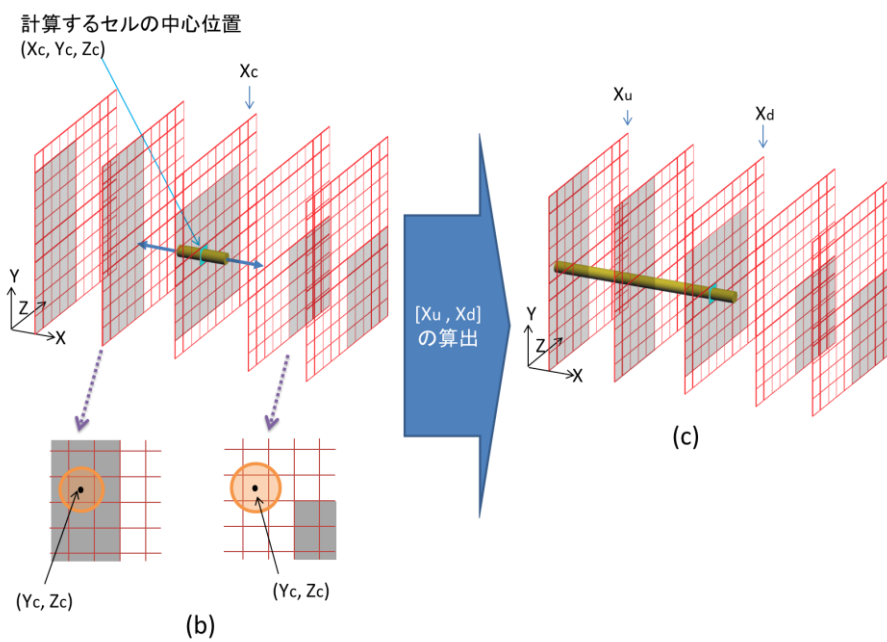
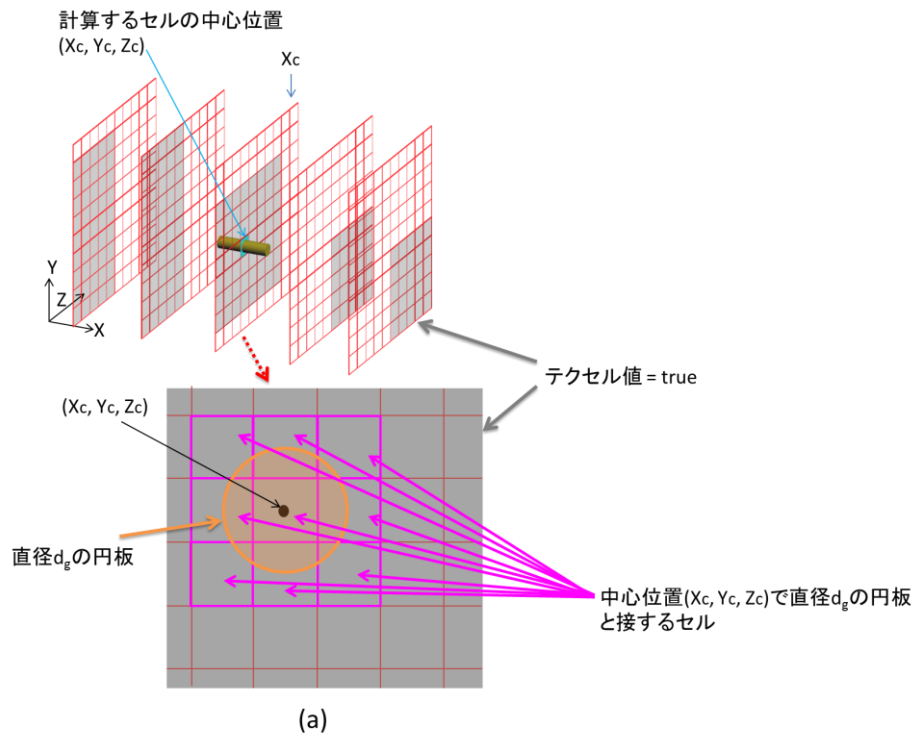


図 25 レイヤ範囲 $[X_u, X_d]$ を算出する処理の概略 (a)(b)各レイヤ内で径 d_g の歯車軸（図中の橙丸）が断面内に配置できるかテクセル値で判定する (c)歯車軸が存在でき、 X_c と連続するレイヤ内で終端に位置するレイヤを X_u, X_d とする。

X_u を算出するために、 X_c から0までの各レイヤで軸が断面内に配置できるか順に判定する。 X_c から連続して軸が断面内に配置できる最後のレイヤを X_u とする。

X_d を算出するために、 X_c からN-1までの各レイヤで軸が断面内に配置できるか順に判定する。 X_c から連続して軸が断面内に配置できる最後のレイヤを X_d とする。

次に、歯車セットを構成する歯車の歯車スコア F_i を定義する。添え字 i はレイヤ番号を表す。そして、以下の式(1)のように S_1 は $[X_u, X_d]$ 範囲内の F_i の総和とする：

$$S_1 = \sum_{i=X_u}^{X_d} F_i \quad \dots(1).$$

式(1)に示す通り、 X_u から X_d まで順に F_i を計算する。計算中のレイヤ番号を X_s , $P_{X_s}=(X_s, Y_c, Z_c)$ とし、 X_s での F_i を F_{X_s} とする。 P_{X_s} に配置する歯車の歯数を T_{X_s} , ピッチ円直径を d_{X_s} , 歯先円直径を C_{X_s} とする。これらを用いて以下の式を定義し、計算方法を図 26を用いて説明する：

$$d_{X_s} = m T_{X_s} \quad \dots(2),$$

$$C_{X_s} = d_{X_s} + 2 m \quad \dots(3),$$

$$F_{X_s} = \max(C_{X_s}) \quad \dots(4).$$

図 26 (a)のように式(2)における歯数 T_{X_s} の初期値を最少の歯数の 17 とし、図 26 (a)から(b)のように 1 ずつ増加させていきながら、式(3)の C_{X_s} を持つ歯車が断面内に存在できるか直径 C_{X_s} の円板に接触するセルのテクセル値を参照し判定する。そして式(4)のように断面内に存在できる最大の C_{X_s} を F_{X_s} とする。式(3)の計算において、 T_{X_s} の初期値で式(3)の C_{X_s} を持つ歯車が断面内に存在できない場合、 P_{X_s} には歯車が配置できないとして $F_{X_s} = 0$ とする。

以上の計算で、セルの中心位置 P_c の S_1 を求めることができる。

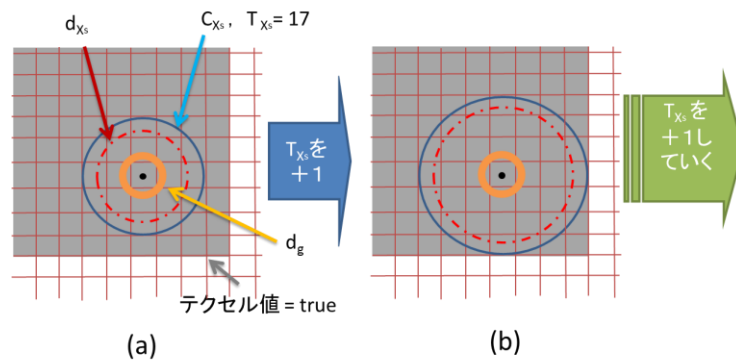


図 26 F_{X_s} の計算方法 C_{X_s} が断面内でなくなるまで T_{X_s} に +1 していく。

2.2.2.2 2つ目以降の歯車セットのスコア S_2 計算

2つ目以降の歯車セットに対する S_2 の計算は基本的には 2.2.2.1 と同じ処理を行うが、配置済の歯車との関係から以下に示す処理を加える

1) 計算中のレイヤ番号 X_c のセルの中心位置 P_c を (X_c, Y_c, Z_c) とする.

P_c に配置済の歯車 1 つとかみ合える歯数 T_k の歯車が配置できる場合のみ S_2 を計算する. ここで、配置済みのある歯車のピッチ円と 1 点で接触できるピッチ円をもつ歯車が断面内に配置できる場合、その歯車がかみ合える歯車となる. そして S_2 の計算では、 F_{Xc} は以下の 3 つの式を用いる：

$$d_{Xc} = m T_k \quad \dots(5),$$

$$C_{Xc} = d_{Xc} + 2 m \quad \dots(6),$$

$$F_{Xc} = C_{Xc} \quad \dots(7).$$

C_{Xc} を持つ歯車が断面内に配置できない場合、 X_c でかみ合える歯車を配置できないため、 S_2 は 0 とする.

2) 2.2.2.1 の S_1 計算時のセルのテクセル値を参照して歯車や歯車軸を断面内に配置できるかの判定に、 S_2 の計算では配置済みの歯車の歯先円や歯車軸と接触するかの判定を追加する. そして、接触する場合、断面内に配置できないと判定する. 例外として、1)の処理での、接触判定では X_c のレイヤでかみ合う配置済みの歯車の歯先円との接触は考慮しない.

以上の処理で、セルの中心位置 P_c の S_2 を求めることができる.

2.2.2.3 歯車セットスコアに関して

2.2.2.1, 2.2.2.2 の式(4)(7)に示すように、歯車の最外径である歯先円の直径が大きいほど F_i が大きくなる. 従って、その総和である S_1 , S_2 は歯車セットが大きい歯車で構成されているほど値が大きくなる. 本手法では、 S_1, S_2 の大きい歯車セットから順に配置するため、大きい歯車で構成されている歯車セットが優先的に配置される. したがって、断面が大きい部分には、大きい歯車が配置され、大きい歯車が配置されないような断面が小さい部分、または配置された歯車や歯車軸の隙間に小さい歯車が配置される. これにより、生成される歯車集合体を異なるサイズの歯車で構成する目的を満たせる. しかしながら、本手法は、配置済みの歯車とかみ合える歯車を持つ歯車セットの中で最大の歯車セットを 1 つずつ配置するという局所的な解法のため、出力モデル内で取りうる最大の歯車の組み合わせを求めることは困難である.

2.2.3 歯車が配置されていない領域の発生の抑制

図 27に示す例を用いて、「歯車集合体生成後の歯車が配置されていない領域」が発生する場合について説明する。

図 27(a)に示すように入力モデルに対して $X_r=0$, $X_r=1$ のレイヤを設置した場合、図 27(b)のような断面を得る。得られた断面に逐次歯車セットを配置する場合、図 27(c)のような最大 S_1 を持つ歯車セットを配置し、新たな歯車セットを既配置の歯車セットとかみ合うように配置していく。そして最終的に、図 27(d)のような歯車集合体を生成する。生成した歯車集合体は図 27(d)に示すように $X_r=1$ の断面内の領域には歯車が配置されていない。すなわち、「歯車集合体生成後の歯車の配置されていない領域」が発生しているため、入力モデルの近似形状を十分には得られていないことになる。

この原因について、図 28を用いて説明する。図 27(b)の断面には、図 28(b)のような異なるレイヤ間の両領域を連結できる歯車セットが存在している。その領域を構成するセルを本研究では「跨ぎセル」と呼ぶ。図 27(b)の断面内の「跨ぎセル」を図 28(a)に示す。そして、「まだ歯車が配置されていない領域へ連結できる」跨ぎセルを「未跨ぎセル」と呼ぶ。図 27(b)の断面内に、図 28(c)に示すような歯車セットを配置した後の「まだ歯車が配置されていない領域」を図 28(c)に、「未跨ぎセル」を図 28(d)にそれぞれ示す。

図 27の例では、この「未跨ぎセル」を歯車が塞いでしまい、 $X_r=1$ へ連結できる歯車セットが配置できなくなったため、「歯車集合体生成後の歯車の配置されていない領域」が発生した。この例のように、歯車セットを配置する過程で「未跨ぎセル」を塞ぐような歯車を配置した場合は、「歯車集合体生成後の歯車の配置されていない領域」が発生することがある。

一方、本手法では、入力モデルの断面をテクセル化する際に断面の面積の大きさを考慮していないので、断面内に「最少歯数の歯車が配置できないほど面積の小さい領域」が存在する可能性がある。本手法は歯車を置くことのできる大きさを持つ面積の領域に歯車を配置していく手法なので、「最少歯数の歯車が配置できないほど面積の小さい領域」には、終始歯車が配置されない。従って、最少歯数の歯車が配置できないほど面積の小さい「歯車集合体生成後の歯車の配置されていない領域」が発生する。その領域を「歯車未配置のとても小さい領域」と呼ぶ。しかしながら、「歯車未配置のとても小さい領域」が発生しても、最少歯数の歯車が配置できないほど面積が小さいため、その領域に歯車が置かれなかったとしても、入力モデルを近似する歯車の集合体を得られると考えられる。

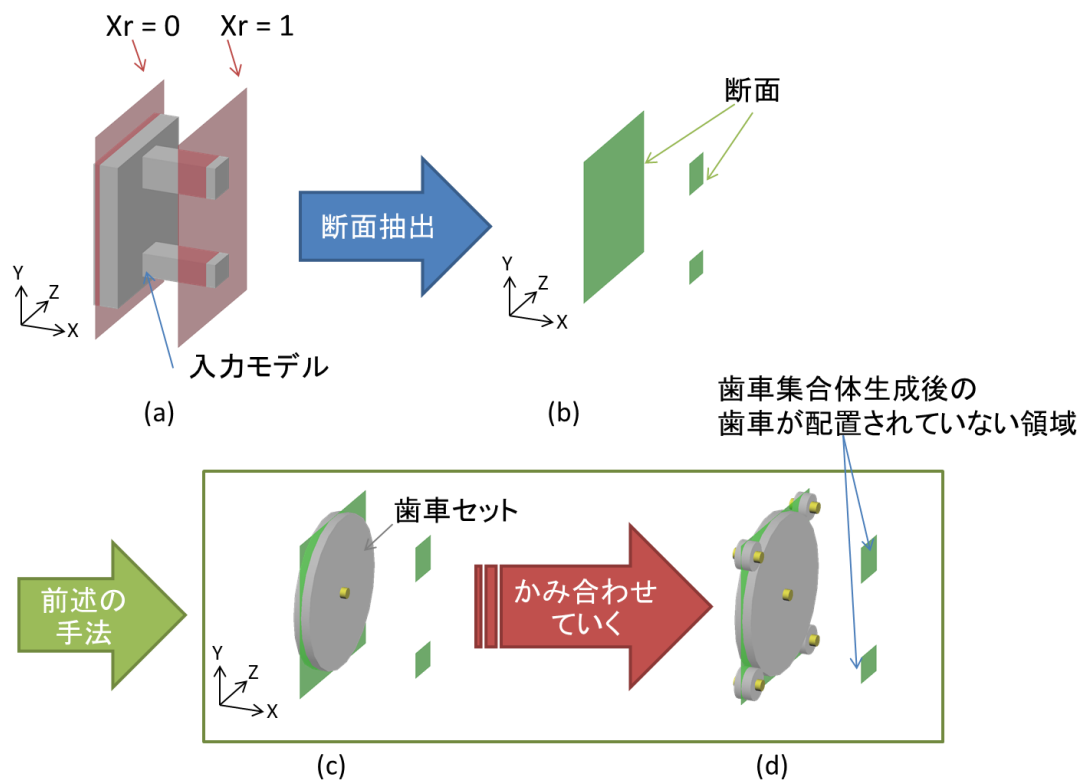


図 27 歯車集合体生成後の歯車が置かれない領域の発生の例 (a)のような入力モデルとレイヤ位置から(b)のような断面が抽出される。その断面について前述の手法を用いると、(c)のような歯車セットが1つ目に配置され、(d)のような歯車集合体が生成される。

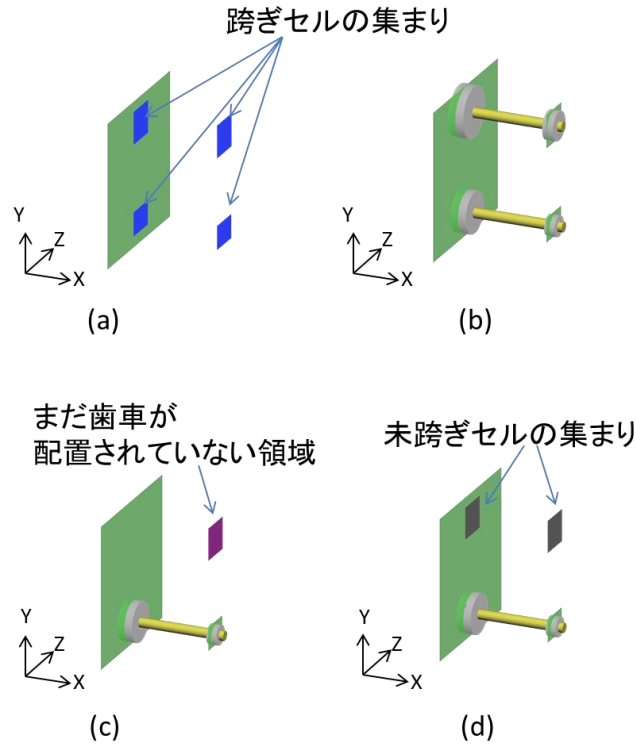


図 28 「歯車集合体生成後の歯車の配置されていない領域」が発生する原因 (a):図 27(b)の断面における「跨ぎセル」の位置 (b): 図 27(b)の断面で異なるレイヤ間の領域を連結できる歯車セットの例 (c): 図 27(b)の断面に図のような歯車セットを置いた場合の「まだ歯車が配置されていない領域」 (d):(c)の「未跨ぎセル」の位置

本手法では，入力モデルの近似形状を得るために，入力モデルの断面内の領域に歯車を置いていき，置いた歯車の集合体で入力モデルの近似形状を得る手法と言える．このことから，前述したように「歯車集合体生成後の歯車の配置されていない領域」が発生すると，生成した歯車集合体が入力モデルの近似形状を得られないとして，発生を抑止する対策を行う．

「歯車集合体生成後の歯車の配置されていない領域」が発生する問題への対策として，2.2.2.1 および，2.2.2.2 で述べた S_1, S_2 の計算内に「未跨ぎセル」を塞ぐような歯車を配置しない処理を加える．その処理の準備として， S_1, S_2 の計算の前に以下の 2 つの処理を行う．

まず 1 つ目の処理として，断面を構成する領域に関する以下の手順の処理を行う．

- 1) テクセル値が **true** であるセルを各領域に分ける．領域の定義として，YZ 平面上で互いに 4 連結しているセルの集合を 1 つの領域と定義する．
- 2) 各領域に ID を振る．その ID を本研究では「断面 ID」と呼ぶ．

3) 各セルに自身が属する領域の断面 ID を持たせる.

次に2つ目の処理として,「跨ぎセル」の抽出に関する処理を行う. ここで,異なるレイヤ間の領域を連結する歯車セットを配置できる位置は,以下の2つの条件を満たす.

- ・歯車を配置するために,配置できる位置を中心とした最少歯数の歯車の歯先円の大きさ以上の領域が確保されている.
- ・異なるレイヤに歯車を置くために,配置できる位置を中心とした最少歯数の歯車の歯先円の大きさの領域全体が歯車軸の方向の断面と連続している.

そして,その領域を構成するセルが「跨ぎセル」である.

「跨ぎセル」の抽出は以下の手順の処理を行う. 手順の説明に図 29を用いる. 以下の処理は画像処理のオープニング処理[21]を応用している.

- 1) X 方向に断面が連続するセルを,後述する式(8)を用いて抽出する. そして抽出した各セルに「フラグ A」を持たせる.
- 2) 「フラグ A」を持つ各セルに対して,以下の手順を行う.
 - ① 図 29(a)のように,セルの中心位置に,その位置を中心とした最少歯数の歯車の歯先円直径を持つ円板を YZ 平面に置く.
 - ② 図 29(b)のように,置いた円板に接触する各セルが「フラグ A」を持つセルのみ場合,円板の中心位置のセルに「フラグ B」を持たせる.
- 3) 「フラグ B」を持つ各セルに対して,以下の手順を行う.
 - ① 図 29(c)のように,そのセルの中心位置に,その位置を中心とした最少歯数の歯車の歯先円直径を持つ円板を YZ 平面に置く.
 - ② 図 29(d)のように,置いた円板に接触する各セルに「フラグ C」を持たせる.
- 4) 2) 3) の処理により, X 方向に断面が連続するセルの集合内から,セルの中心位置に最少歯数の歯車の歯先円の大きさの領域を持たないセルが取り除かれる. そして「フラグ C」を持つセルが「跨ぎセル」である.

レイヤ番号が U, Y 座標が V, Z 座標が W の XYZ 空間座標 (U,V,W)において,式(8)が TRUE であれば,(U,V,W)を内包するセルは X 方向に断面が連続するセルである.

$$(G(U-1,V,W) \vee G(U+1,V,W)) \wedge G(U,V,W) \quad \dots(8).$$

ここで, G(I, J, K)関数は座標(I, J, K)を内包するセルのテクセル値(TRUE or FALSE)を返す.

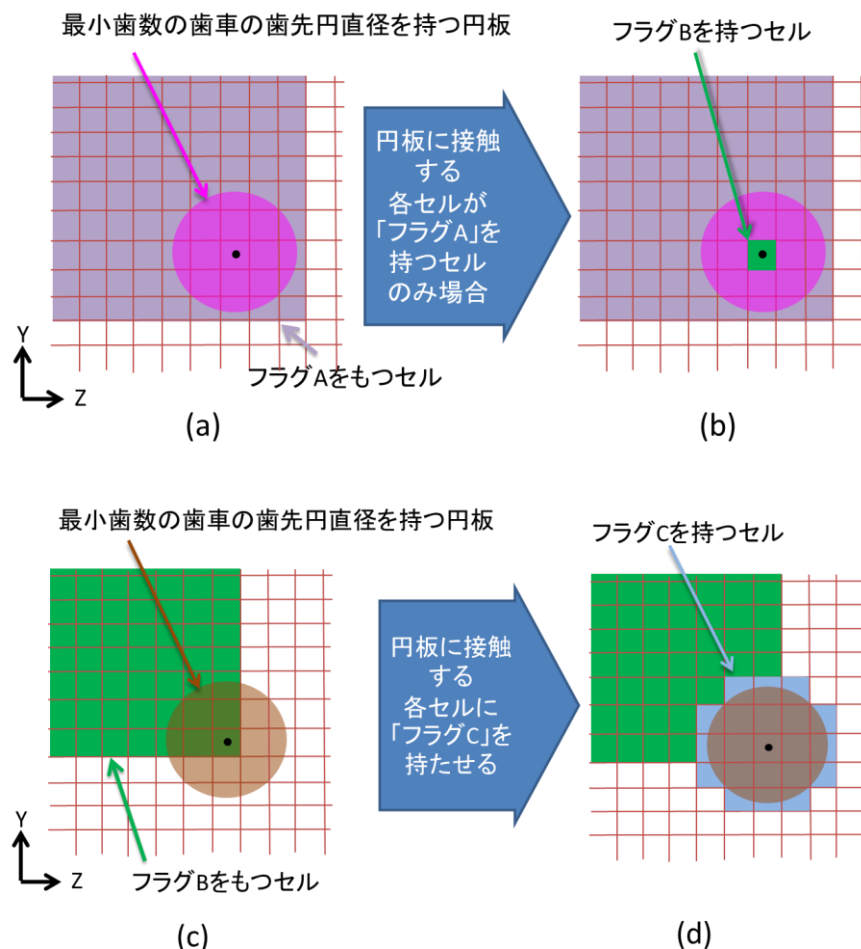


図 29 「跨ぎセル」抽出の処理

つづいて、 S_1, S_2 の計算内に追加する処理について説明する．前述した S_1, S_2 の計算における歯車軸の範囲 $[X_u, X_d]$ の計算後に，以下の処理を追加する．

- ・ 配置済みの歯車が置かれている領域の断面 ID を「歯車配置済み領域リスト」に登録する．
- ・ $[X_u, X_d]$ の範囲内の X_r で，YZ 座標が (Y_c, Z_c) の位置を内包する各セルが持つ断面 ID を「軸リスト」に登録する．そして以下の手順を行う．

1) $[X_u, X_d]$ の範囲内の X_r で，YZ 座標が (Y_c, Z_c) の各位置で以下の処理を行う．

- ・ (X_r, Y_c, Z_c) の位置に最少歯数の歯先円直径の円板を中心にした場合，円板に接触する各セルが，後述する式(9)を用いて，「未跨ぎセル」かどうか判定する．「未跨ぎセル」だった場合， (X_r, Y_c, Z_c) の位置を内包するセルが持つ断面 ID を「軸リスト」が取り除く．

2) 1)の処理で断面 ID が「軸リスト」から取り除かれなくなるまで，1)を繰り返す．ここで，

「軸リスト」内に残った断面 ID を持つ領域は、計算している S の歯車セットで置かれる領域を表す。

式(9)について、式(9)が TRUE なら(U,V, W)を内包するセルは「未跨ぎセル」である。

$$(M(U-1,V,W) \wedge H(U-1,V,W)) \vee (M(U+1,V,W) \wedge H(U+1,V,W)) \wedge M(U,V,W) \dots(9),$$

M(I, J, K)関数は座標(I, J, K)を内包するセルが「跨ぎセル」なら TRUE を返す。H(I, J, K)関数は座標(I, J, K)を内包するセルがもつ断面 ID が「歯車配置済み領域リスト」、または「軸リスト」内にないなら TRUE を返す。

そして、 F_i 計算時の「歯車が断面内に置けるか」の判定に以下の判定を加える。

- ・直径 C_{xs} の円板に接触するセルが「未跨ぎセル」場合、 C_{xs} を持つ歯車は断面内に配置できないと判定する。

以上の処理により、「歯車集合体生成後の歯車の配置されていない領域」の発生を抑制できる。

2.2.4 描画のためのメッシュの生成

2.2.1 項から 2.2.3 項までの方法で、歯車集合体を構成する歯車セットのパラメータを算出できる。

算出した歯車セットをレンダリングするために、歯車セットを構成する歯車と歯車軸の形状メッシュを以下のように生成する。

図 30に示す歯車の外形は、モジュール m 、ピッチ円直径 d 、歯数 T を用いた歯車の規格[16][24]で定義される。歯車の形状メッシュは、算出したパラメータ m , d , T , をもつ歯車の外形をなす各部位の寸法を歯車の規格[16][24]に当てはめて決定する。以下に決定する部位の一覧を、図 31に各部の位置を示す。

①円ピッチ：

ピッチ円上の歯の間隔の円弧長さ

$$\text{円ピッチ } p = \pi * m$$

②円弧歯厚：

ピッチ円上の歯の厚みの円弧長さ

$$\text{円弧歯厚 } s = p / 2$$

③歯先円直径：

歯車の最外径の円の直径

$$\text{歯先円直径 } C = d + 2m$$

④歯底円直径：

歯車の最内径の円の直径

$$\text{歯底円直径 } d_f = d - 2.5m$$

⑤基礎円直径：

歯底から歯先までの形状であるインボリュート曲線[22][24]の基礎円の直径

$$\text{基礎円直径 } d_b = d * \cos 20^\circ$$

以上の各部位の寸法で、本研究が対象とする歯車の外形が決まる。

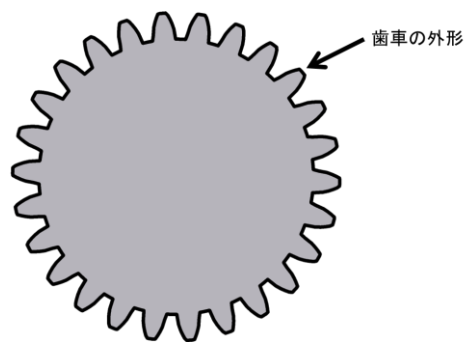
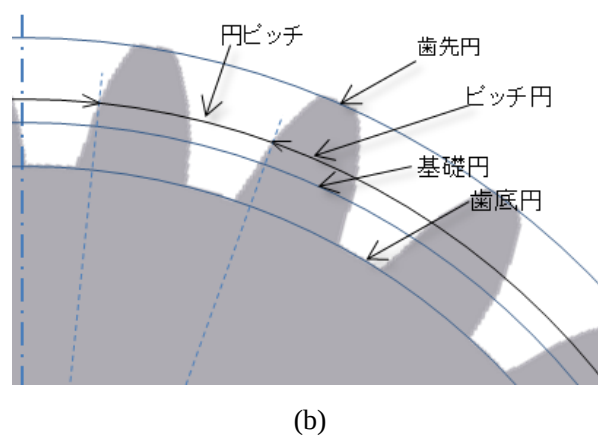
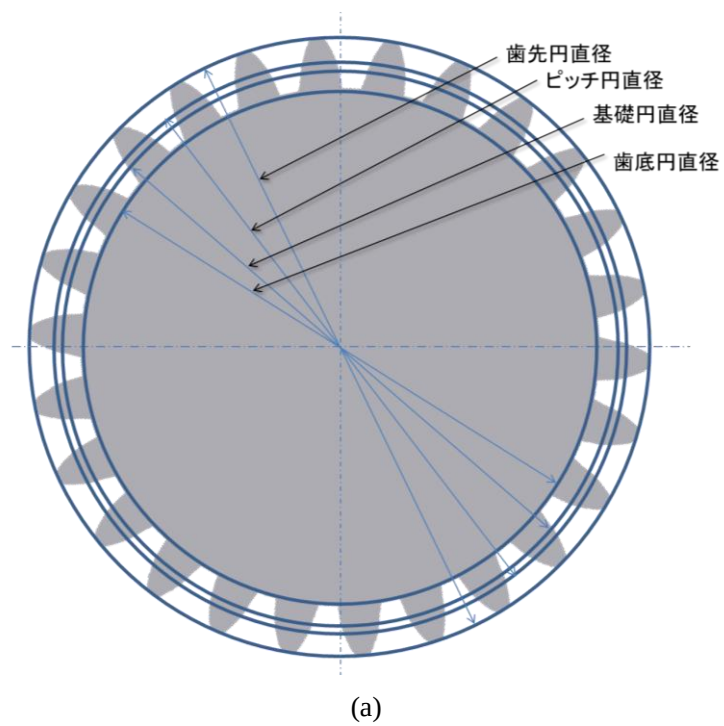
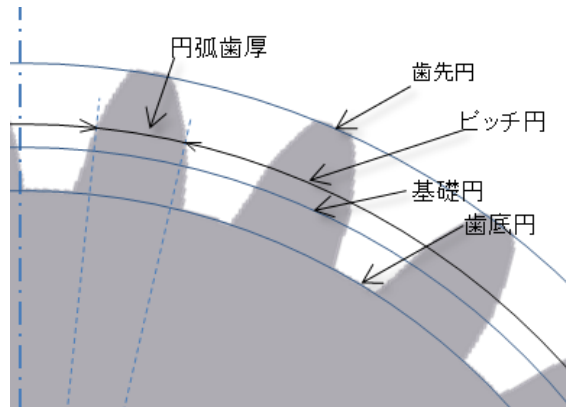


図 30 歯車の外形





(c)

図 31 歯車の外形をなす各部位

次に図 32のように，歯車の外形と，その歯車の中心を通り，歯車の歯先円の円周を N_α 等分する角度 α を持つ線群との接点を計算する． N_α はメッシュを生成する歯車の歯数 $T * M_\beta$ とする． M_β の値はユーザが自然数を指定する． M_β は大きいほど，生成する歯車の歯のメッシュが現実の歯車の歯の形状に近づく．本研究では， $M_\beta=15 \sim 20$ の値を使用している．

そして，図 33のように計算した接点群内の円周方向に連続する 2 点と歯車の中心点で 3 角形メッシュを構築していく．その結果，構築した 3 角形メッシュの集合が歯車の外形のメッシュになる．

生成した歯車の外形メッシュを中心軸方向にユーザが指定した歯幅分離れた位置にもう 1 つ生成し，2 つの歯車の外形メッシュを繋ぐ 3 角形メッシュを図 34のように作成する．

上記の処理により，図 34のような，任意のパラメータ m , d , T , 歯幅を持つ歯車の形状メッシュを生成する．

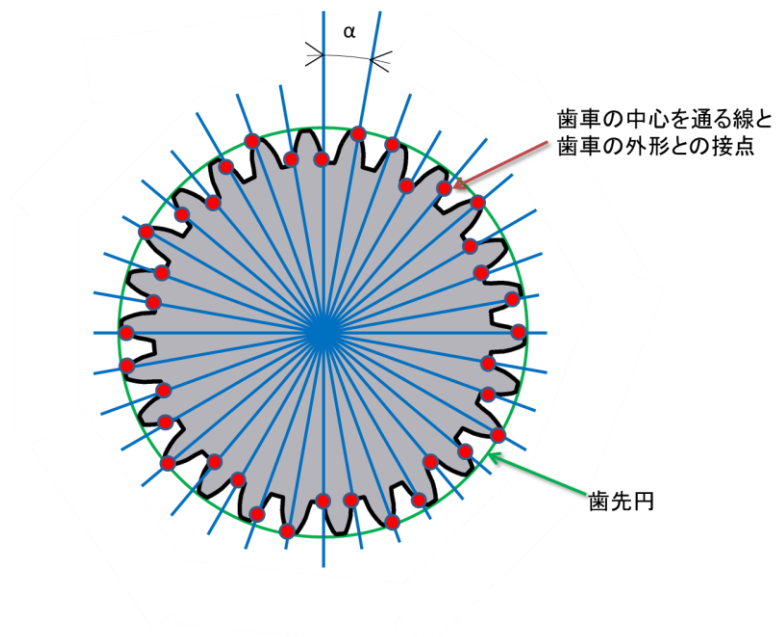


図 32 歯車の外形の分割

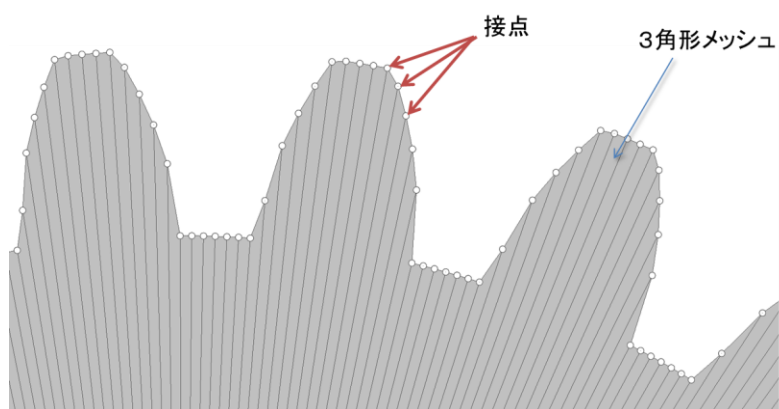


図 33 歯車の中心点と接点で 3 角形メッシュを構築

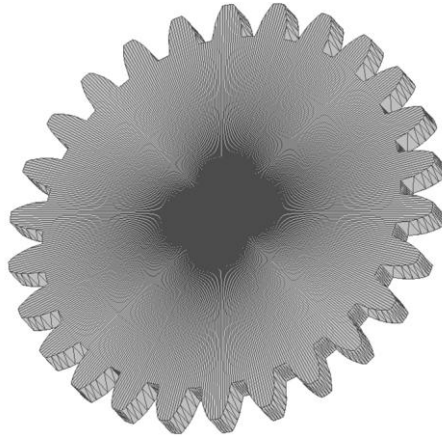


図 34 歯車の形状メッシュ

続いて、歯車軸の形状メッシュの生成について説明する。まず、ユーザが指定した歯車軸径 d_g を直径として持つ円を生成する。次に、図 35のように、円と、その円の中心を通り、円の円周を N_γ 等分する角度 γ を持つ線群との接点を計算する。 N_γ の値はユーザが 3 以上の自然数を指定する。 N_γ は大きいほど、生成された歯車軸の形状メッシュは、円柱に近づく。本研究では、 $N_\gamma = 16 \sim 20$ の値を使用している。そして、図 36のように計算した接点群内の円周方向に連続する 2 点と円の中心点で 3 角形メッシュを構築していく。その結果、構築した 3 角形メッシュの集合は図 36 のような円板に近似した形状メッシュになる。次に、生成した円板に近似した形状メッシュを円板の中心軸方向の、生成対象の歯車セットの全長分離れた位置にもう 1 つ生成し、図 37 のように 2 つの円板を繋ぐ 3 角形メッシュを生成する。ここで、本研究では、歯車軸を目立たせるため、もう 1 つの円板を生成する位置を上記の記述より微小量離している。

以上の処理で、図 37 のような任意のパタメータ d_g 、歯車セットの全長を持つ歯車軸の形状メッシュを生成することができる。

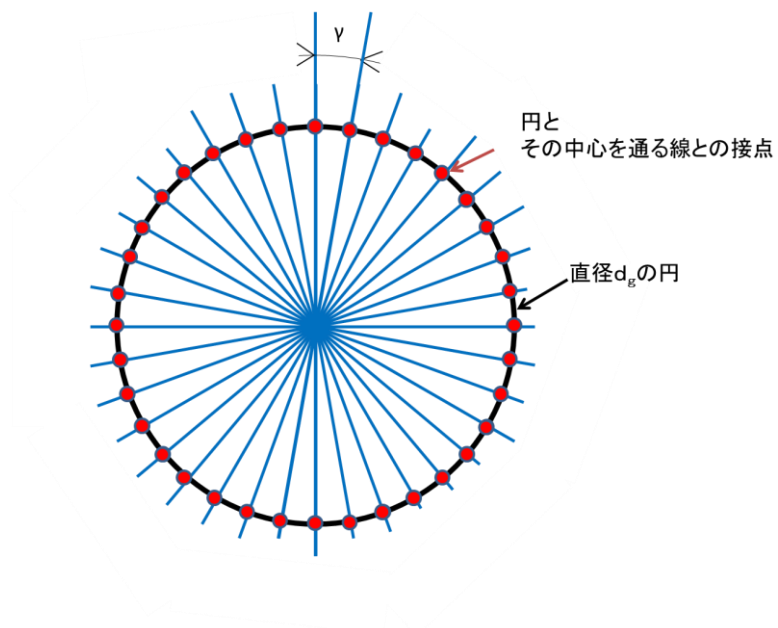


図 35 直径 d_g の円の分割

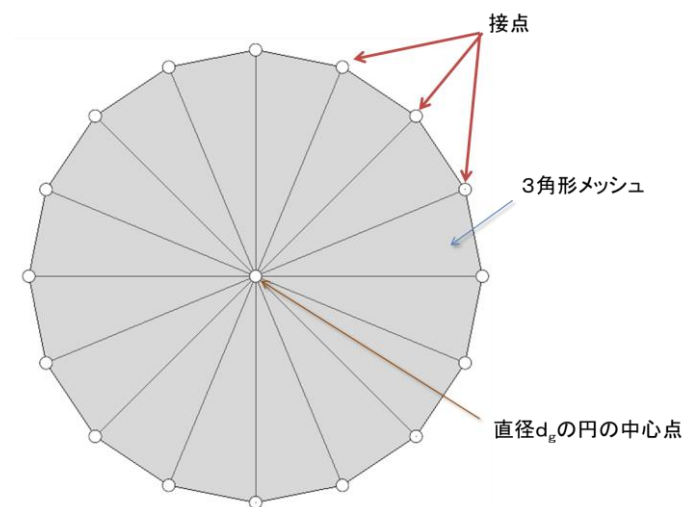


図 36 直径 d_g の円の中心点と接点で 3 角形メッシュを構築

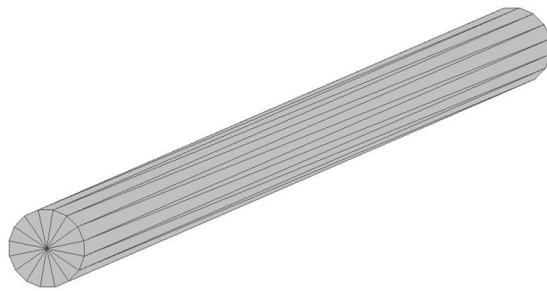


図 37 歯車軸の形状メッシュ

最後に、歯車および歯車軸の中心の XYZ 空間位置は S_1 , S_2 の計算時のセルの中心位置とする。
また、各歯車の角度に関しては、かみ合っている歯車同士の歯が干渉しない角度を用いる。

2.3 全数処理と簡易処理の比較実験

本手法の実行環境は CPU : Intel Core i7 CPU 930, メモリ: 6GB である.

ここで本手法では, 最大の S_1 を計算する処理に対し, 各セルの中心位置で S_1 を計算し, 全セル内で一番大きい S_1 を用いる全数処理と, 以下に記す簡易処理の 2 つの方法を適用し, 計算コストを比較した. 全数処理は, 各セルの中心位置で S_1 を計算するため, セルの総数分計算する必要がある. セルの総数はテクセル数 B , レイヤ数 N とした場合, B^2N であり, 2.2.2 項で述べたとおり本手法は入力モデルの近似形状を得るのに B と N は十分に大きい必要があるため, 全数処理は膨大な回数分 S_1 を計算しなければならない. 加えて, 各セルにおける S_1 の計算も, m が小さいほど, B が大きいほど, N が大きいほど計算コストが高くなるため, 全数処理の総計算コストは莫大になると予想できる. その計算コストを抑えつつ, 歯車の配置に必要な S_1 を計算するために以下の簡易処理を行う.

- 1) ランダムに選んだ 1 つのセルの中心位置で S_1 を計算する
- 2) $S_1 > 0$ を持つ位置を内包するセルが N_{S1} 個になるまで 1) を繰り返す. N_{S1} はユーザが設定する. 本手法では $N_{S1}=10$ を用いている.
- 3) N_{S1} 個内で最大の S_1 を用いる.

全数処理と簡易処理を用いた手法で同じパラメータを用いて歯車集合体を生成した結果を図 38, 図 39 に, その計算時間について表 1 に示す. 生成されたモデルのレンダリングには Mitsuba Renderer[20]を用いている.

表 1 に示めた計算時間をみると, この 2 つの例では 1 つ目の歯車セットを計算するのに全数処理では膨大な時間を要している. これに対して, 簡易処理では 1 つ目の歯車セットの計算時間は, 全数処理の計算時間の数万分の 1 である. そして, 全歯車セットの計算時間も全数処理に比べ簡易処理の方が極めて短い.

一方, 図 38, 図 39 に示すようにどちらとも入力モデルの近似形状をもつ歯車の集合体を出力できていることが視認できる. 生成結果が多少異なる原因は, 1 つ目に配置される歯車セットに差異が生じることが原因のひとつである. 加えて, 新たな歯車をすでに配置した歯車にかみ合わせながら逐次配置していく手法であるため, 配置する歯車セットが後に配置する歯車セットに影響を与えることとなる.

S_1 は, その値が大きいほど本研究の目的に合うように定義されているため, 全数処理の方が本

研究の目的には合致している．しかしながら，本手法は， S_1 で配置するのは歯車集合体を構成する無数の歯車セットの1つだけで，その他の歯車セットは S_2 に従って配置されるため，モデルを構成する歯車の傾向は主に S_2 に従うと考えられる．それは図 38，図 39のどちらの手法の歯車集合体でも，入力モデルの大きな部分に当たる部分には大きな歯車を配置できていることからわかる．このことから，計算時間を大幅に削減できる簡易処理を本手法では採用することとした．

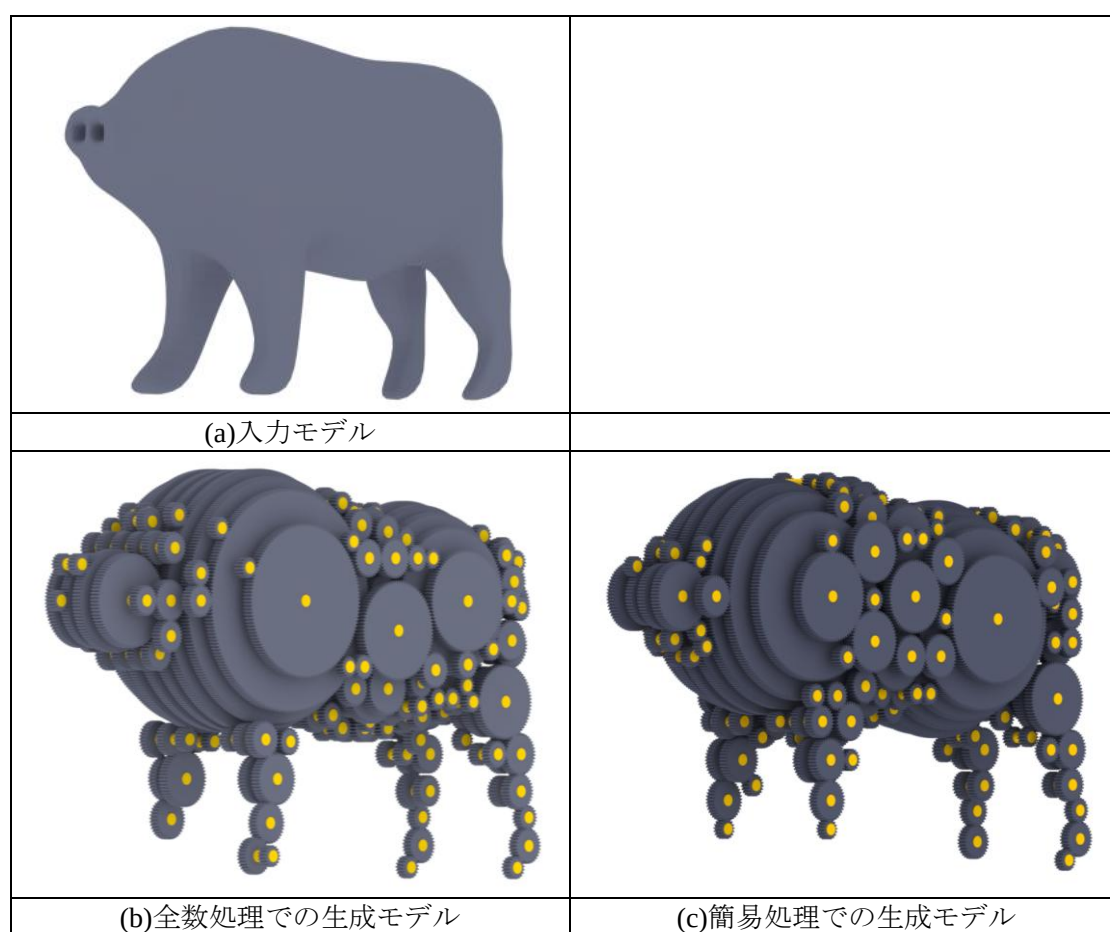


図 38 全数処理と簡易処理を用いた手法で生成した歯車集合体の例 1

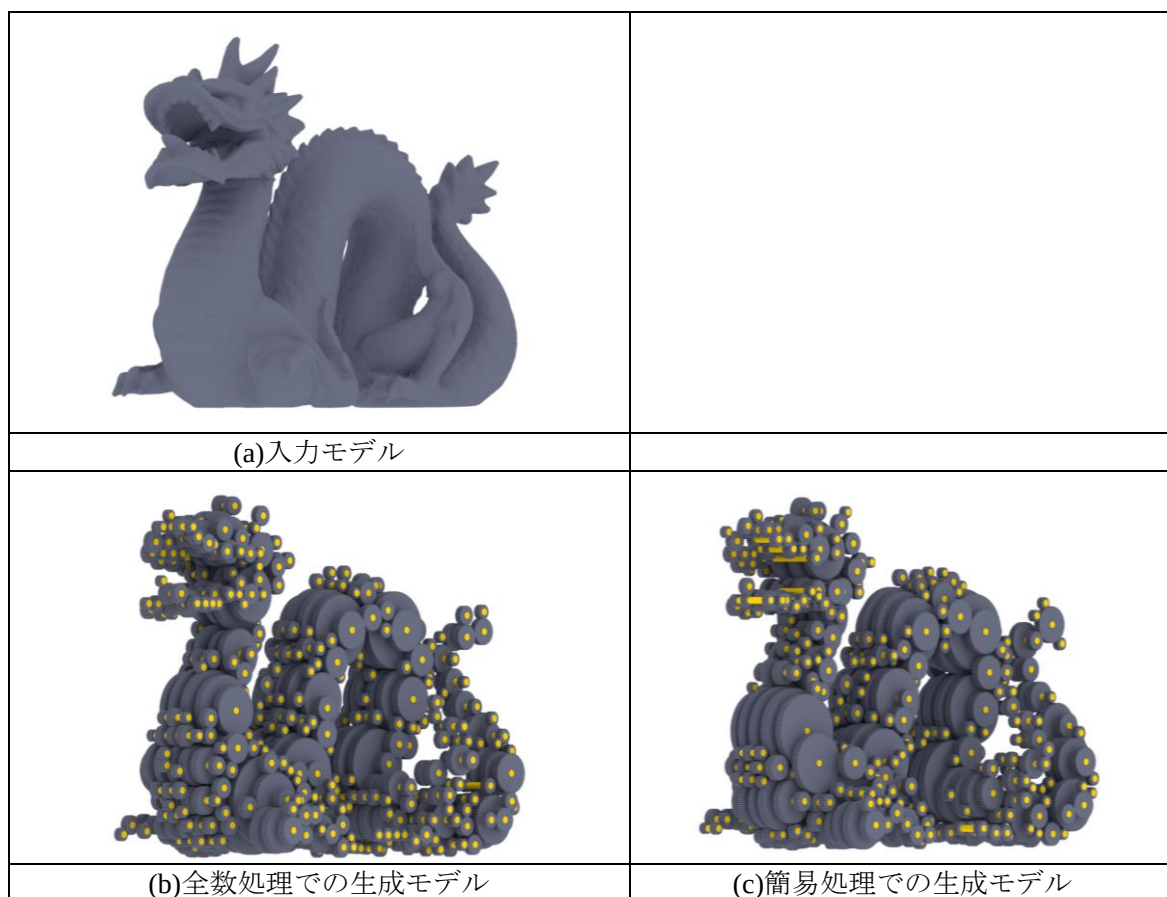


図 39 全数処理と簡易処理を用いた手法で生成した歯車集合体の例 2

表 1 全数処理と簡易処理を用いた手法での計算時間

	図 38		図 39	
処理の種類	全数処理	簡易処理	全数処理	簡易処理
テクセル数 B	600		900	
レイヤ数 N	10		15	
モジュール m	正規化サイズ F/600		正規化サイズ F/900	
歯車軸の太さ d_g	(最小歯数の歯車の d_f) * 4/5			
全歯車セット の計算時間	50,035s	965s	26,286s	2,527s
歯車セット 1 つ目を 配置するのに 要した時間	49,144s	1s	23,668s	1s

第 3 章 結果および考察

この章では、まず、2 章で言及した歯車集合体の生成手法で生成した歯車集合体モデルとその入力パラメータ、計算時間について示し、次にその結果から導きだせる考察について言及する。

3.1 本手法の結果

2 章で説明した本手法で生成した歯車集合体を図 40-図 49 に示し、表 2 にそれぞれの計算時間と入力パラメータについて示す。入力パラメータの歯幅については、レイヤ間の歯車が接触しないような値を用いている。

図 40-図 41, 図 43-図 48 に示すように生成した歯車集合体は、入力モデルの近似形状を保持しており、異なったサイズの歯車で構成されていることが視認できる。また、図 50 の拡大図に示すように生成モデルは歯車同士のかみ合いを表現できている。

本手法は、2.2.2 項で述べた通り、入力モデルの近似形状を得るために、テクセル数 B 、レイヤ数 N は十分大きく、モジュール m は十分に小さい必要がある。しかし本手法では、 B, N が大きくなるほど、歯車スコアの総計算回数が増え、 m が小さいほど 1 つの歯車スコアの計算時間が長くなる。

本手法は、複数のレイヤの位置の入力モデルの形状を保持することで歯車集合体に入力モデルの近似形状を持たせているので、レイヤの枚数 N が多いほど、生成したモデルは入力モデルの近似形状を保持することができる。レイヤ枚数 N 以外に入力パラメータが同じ場合の生成モデル例を図 44-図 47 に示す。この例からもレイヤ枚数が多くなるほど近似形状を保つことができることがわかる。

ユーザが入力するモジュール m は、歯車の歯の大きさであり、値が大きいほど歯が大きくなる。しかし、本手法での最少の歯数は一定なので、ピッチ円直径 $d = m * \text{歯数 } T$ の関係から m を大きく設定するほど、配置される最小歯数の歯車が大きくなる。配置する歯車は大きくなるほど、断面に敷き詰めることが困難になるので、 m は大きいほど生成モデルが入力モデルの近似形状を得るのが困難になる。

入力モデルの大きさは有限なので、断面内に配置する際に、選択できる歯車のサイズには最大値が存在する。したがって、選択できる歯車の d には最大値が存在する。 d が最大値の時の T の

値は、 $d = m * T$ の関係から、 m が小さいほど大きくなる。また、本手法では、選択できる歯車の T には最小値が存在しており、 T は自然数なので、選択できる歯車の T の値の個数は T の最大値から T の最小値を引いたものになる。そして、その T の値の個数は、選択できる歯車の d の値の個数を示す。これらのことにより、 m が小さいほど、選択できる歯車の d の値の個数が増える。

ここで、本手法はかみ合う歯車を置いていく手法であり、配置済みの歯車のピッチ円と 1 点で接触するピッチ円を持つ歯車を離散した位置に置いていく手法である。したがって、その各位置で選択できる歯車の d の値の個数が多いほど、より多くの位置で配置済みの歯車のピッチ円と接触できる歯車を配置できる。これより、 m が小さい方が、かみ合う歯車の配置に有利になり、逆に m が大きいほどかみ合う歯車は配置しづらくなる。

一方、 m が小さくなるほど歯車の歯が小さくなり、生成モデルを見たときに歯車の歯が視認しづらくなる。

入力モデルの形状と入力パラメータが同じでも入力モデルが持つ座標軸の向きによって、レイヤの並びが変わるので生成される歯車集合体の見えは変わる。この例を図 43 と図 48、図 44 と図 47 に示す。

最後に、生成モデルに金属の反射を適応して描画した例を図 42-44 に示す。

表 2 生成モデルの計算時間と入力パラメータ

	図 40	図 41	図 42	図 43
テクセル数 B	800	800	800	800
レイヤ数 N	30	30	30	30
モジュール m	正規化サイズ F/1200	正規化サイズ F/1200	正規化サイズ F/800	正規化サイズ F/1200
軸の太さ dg	(最小歯数の歯車の d_f)*4/5			
全歯車セットの 計算時間	11,327 s	55,066 s	7,210 s	4,184 s
テクセル化と 領域分割の計算時間	2,631 s	3,131 s	2,569 s	3,112 s
跨ぎセルの オープニング処理	15 s	18 s	14 s	10 s
全体の計算時間 (メッシュ作成を含む)	14,300 s	59,871 s	10,041 s	7,608 s

	図 44	図 45	図 46
テクセル数 B	800	800	800
レイヤ数 N	30	10	5
モジュール m	正規化サイズ F/1200	正規化サイズ F/1200	正規化サイズ F/1200
軸の太さ d_g	(最小歯数の歯車の d_f)*4/5		
全歯車セットの 計算時間	13,816 s	1,541 s	476 s
テクセル化と 領域分割の計算時間	2,403 s	964 s	490 s
跨ぎセルの オープニング処理	10 s	3 s	1 s
全体の 計算時間 (メッシュ作成を含む)	16,690 s	2,672 s	1,052 s

	図 47	図 48	図 49
テクセル数 B	800	800	800
レイヤ数 N	30	30	30
モジュール m	正規化サイズ F/1200	正規化サイズ F/1200	正規化サイズ F/800
軸の太さ d_g	(最小歯数の歯車の d_f)*4/5		
全歯車セットの 計算時間	11,173 s	13,876 s	8,856 s
テクセル化と 領域分割の計算時間	2,547 s	5,281 s	2,659 s
跨ぎセルの オープニング処理	10 s	13 s	14 s
全体の 計算時間 (メッシュ作成を含む)	14,098 s	20,024 s	11,810 s

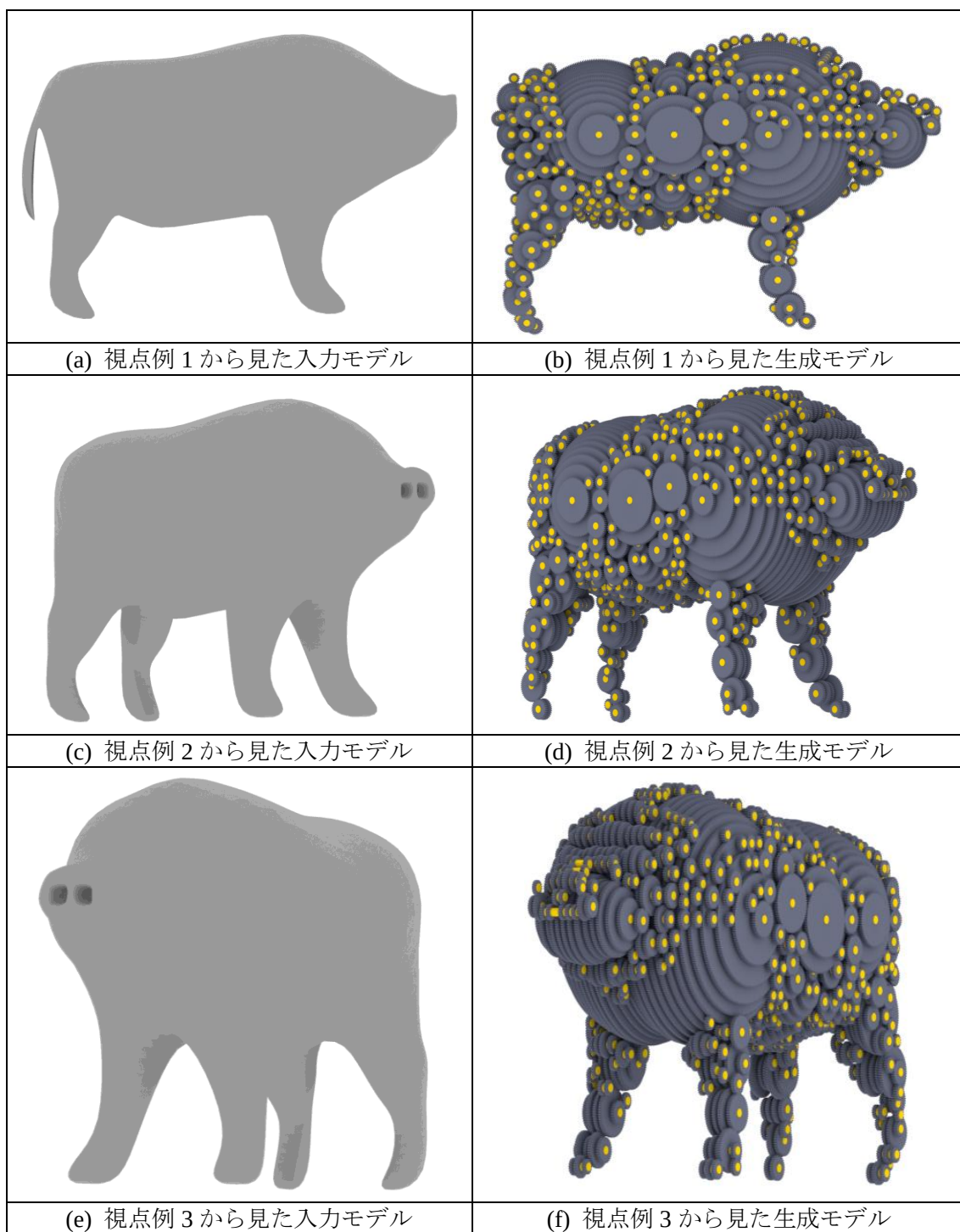


図 40 生成モデル例 1

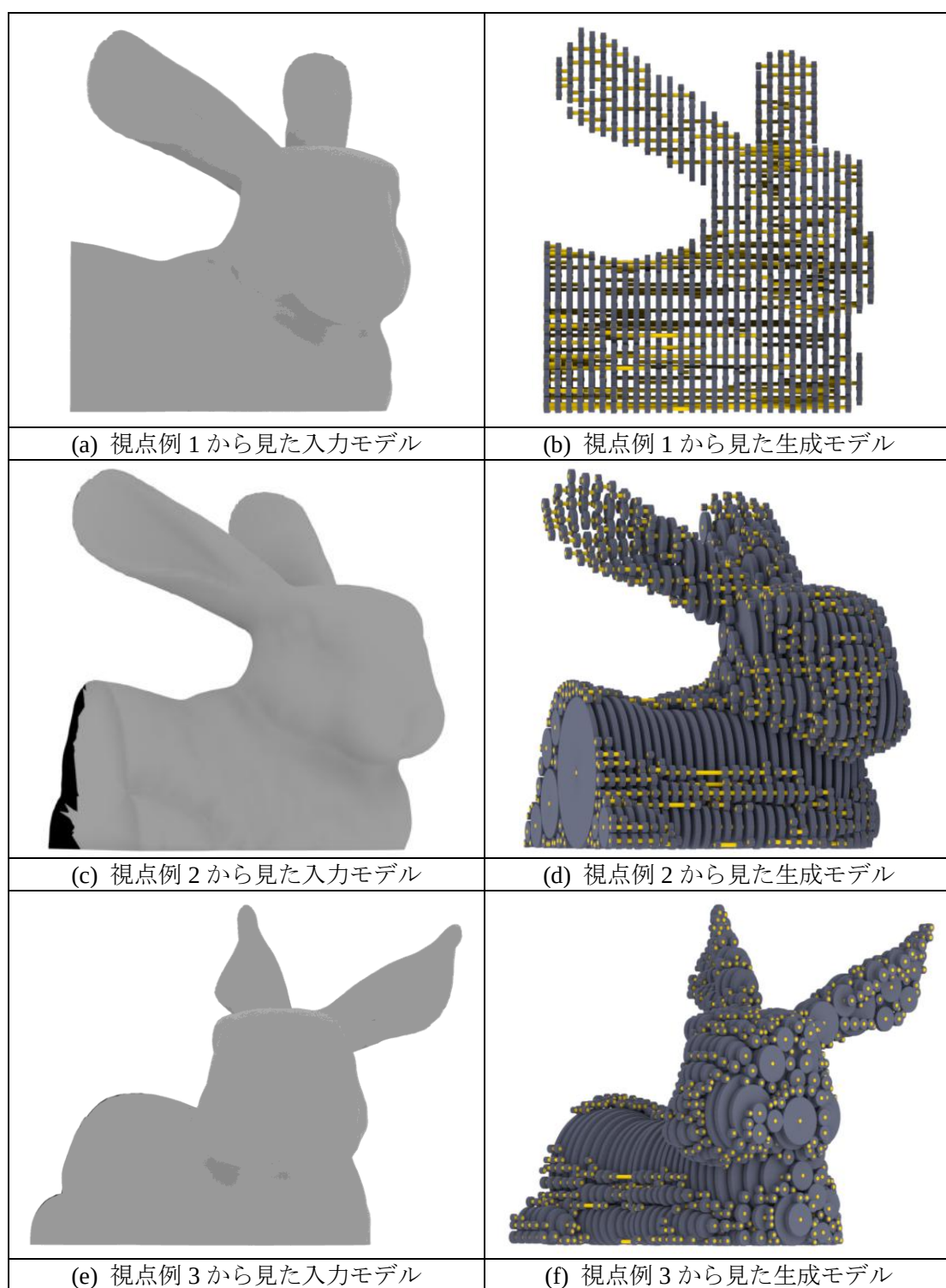


図 41 生成モデル例 2

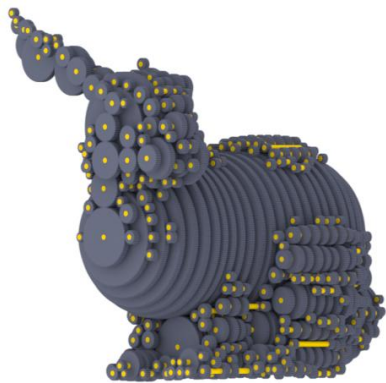
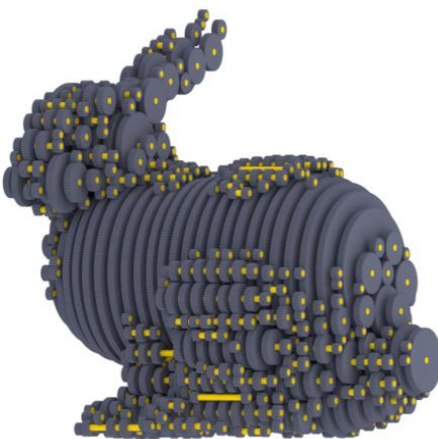
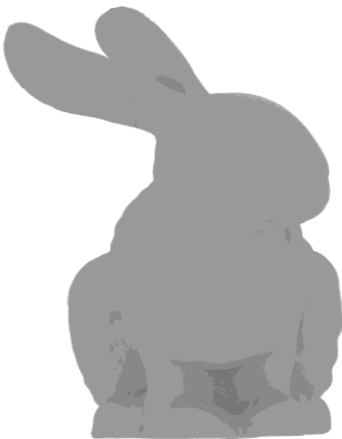
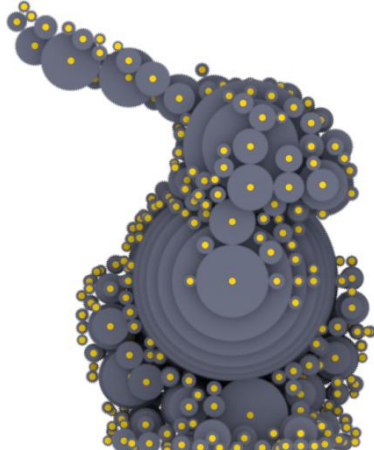
	
(a) 視点例 1 から見た入力モデル	(b) 視点例 1 から見た生成モデル
	
(c) 視点例 2 から見た入力モデル	(d) 視点例 2 から見た生成モデル
	
(e) 視点例 3 から見た入力モデル	(f) 視点例 3 から見た生成モデル

図 42 生成モデル例 3

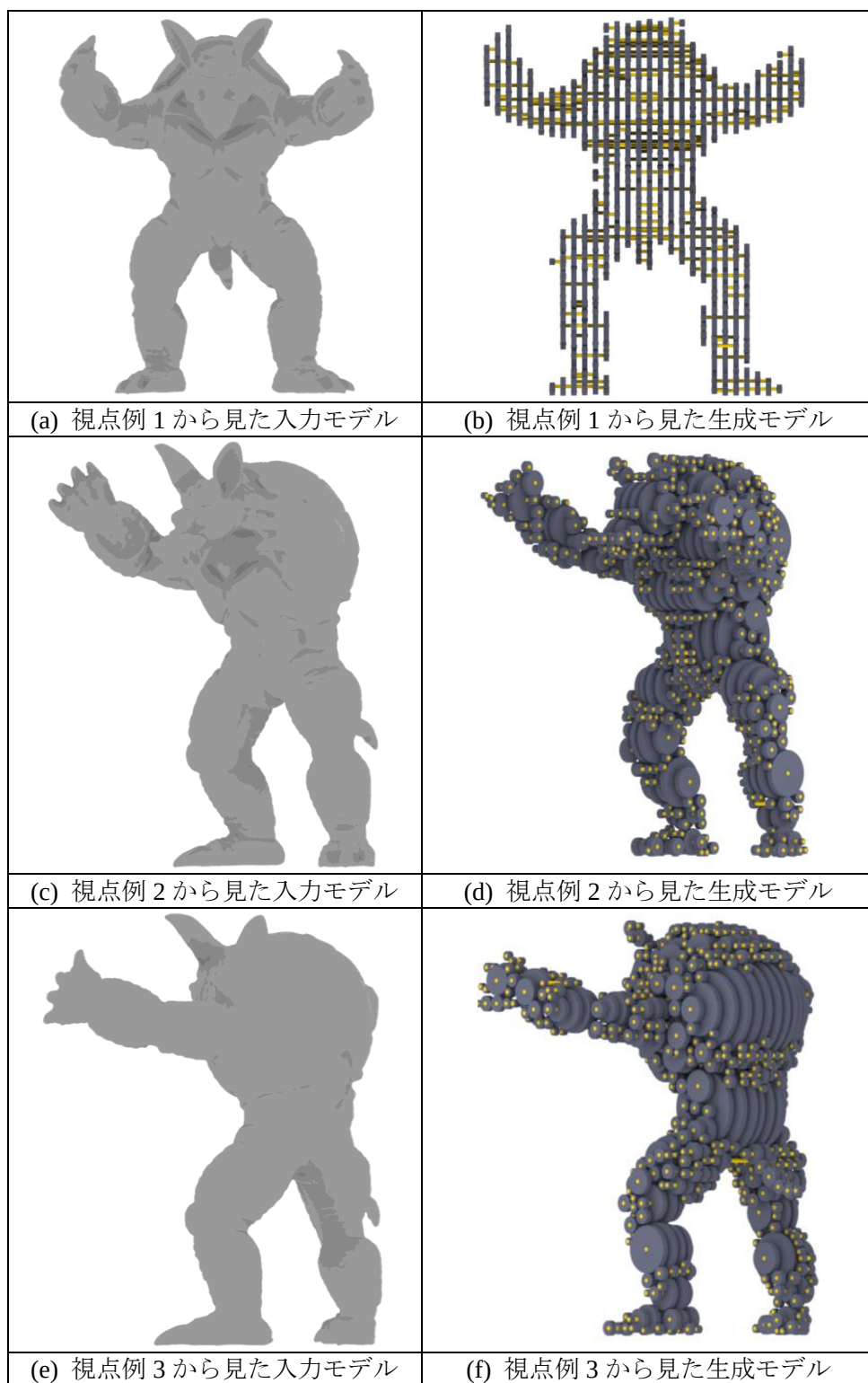


図 43 生成モデル例 4

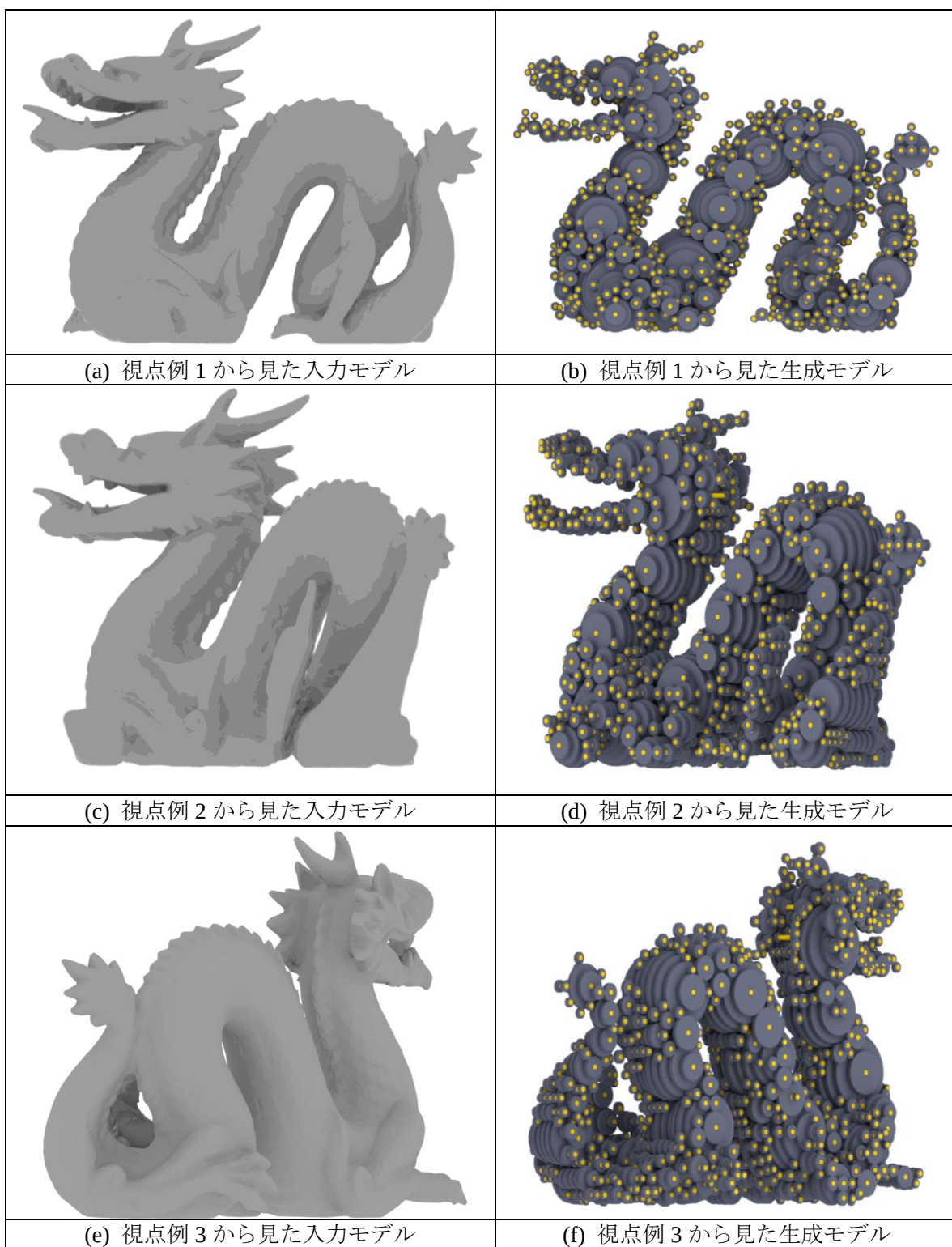


図 44 生成モデル例 5

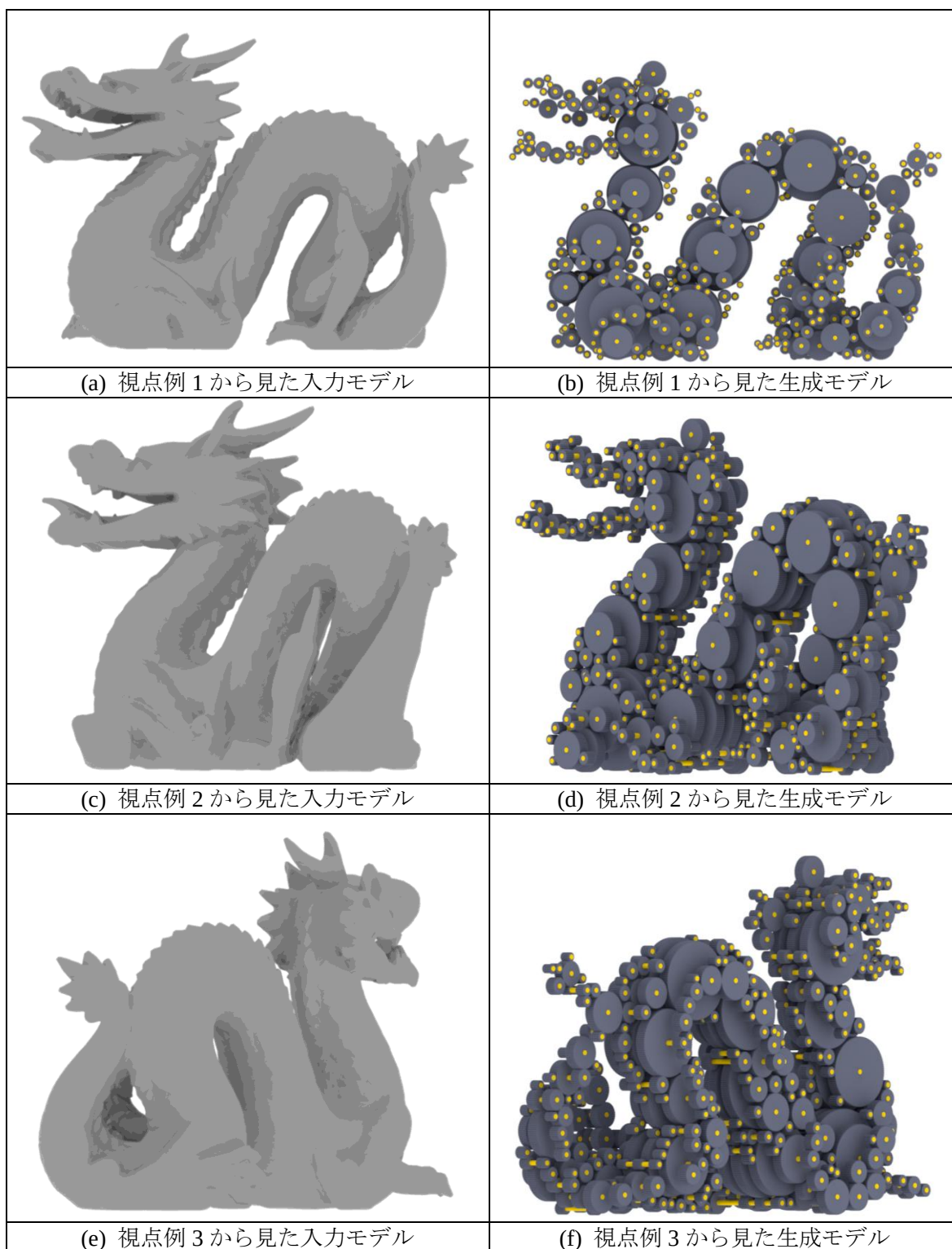


図 45 生成モデル例 6

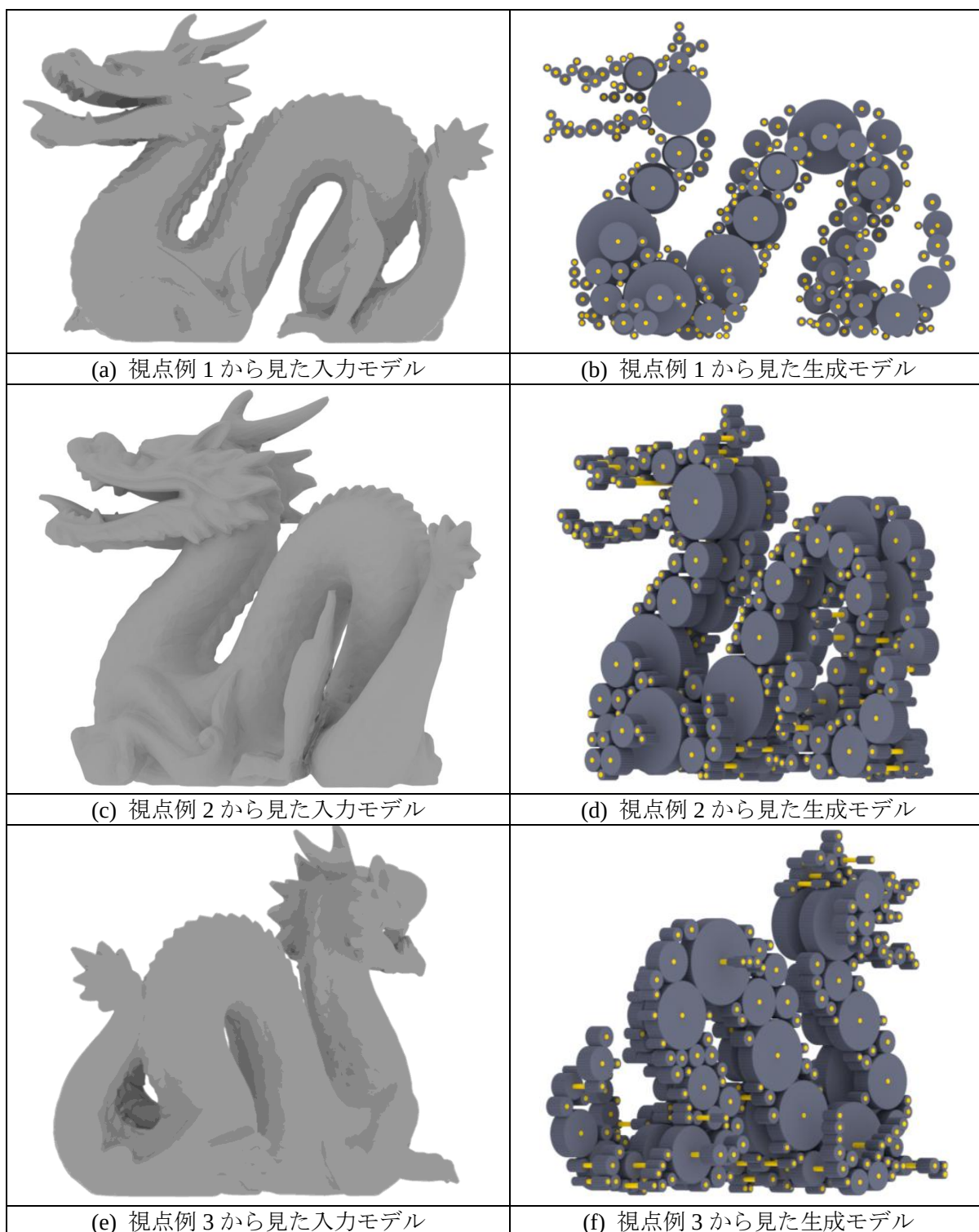


図 46 生成モデル例 7

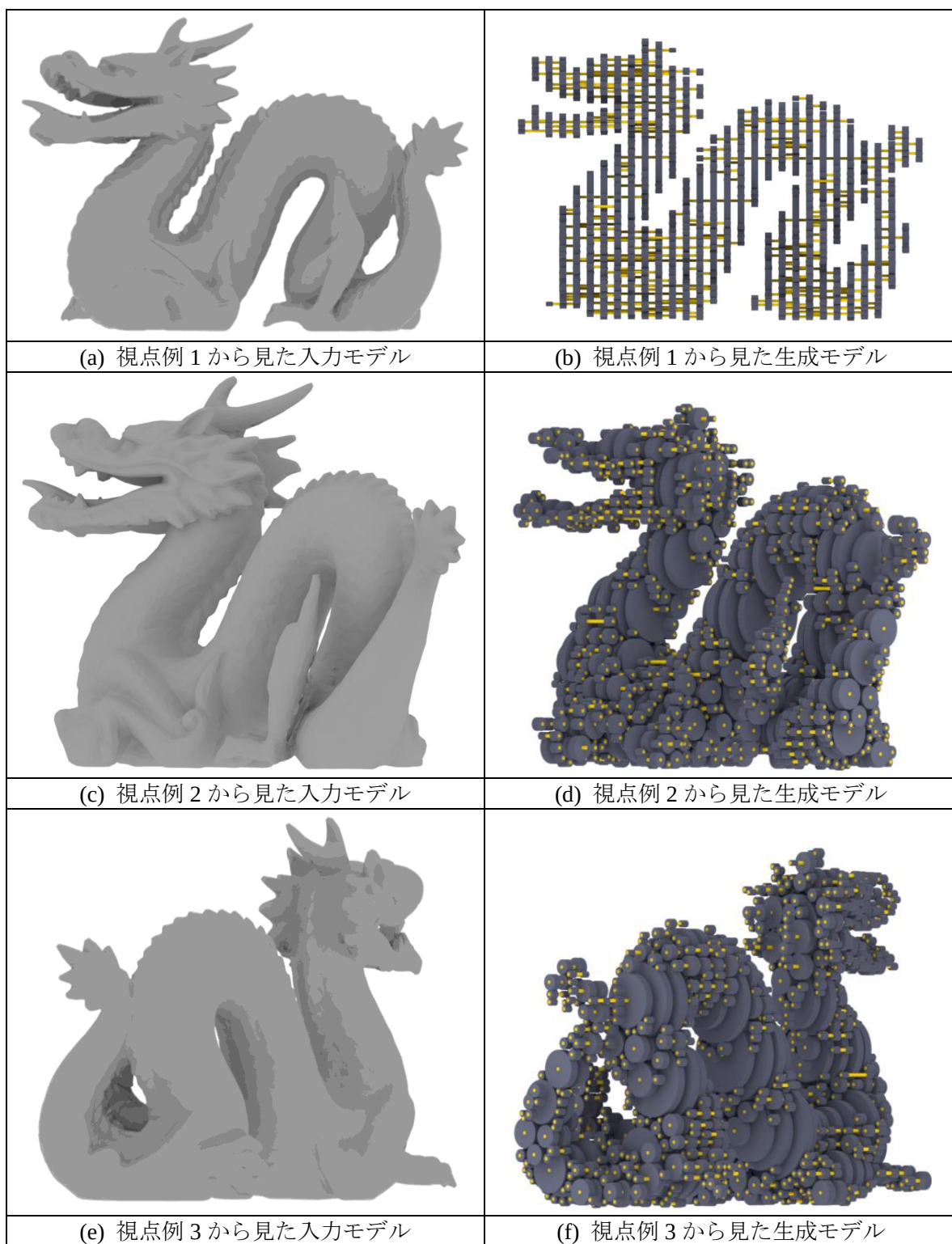


図 47 生成モデル例 8

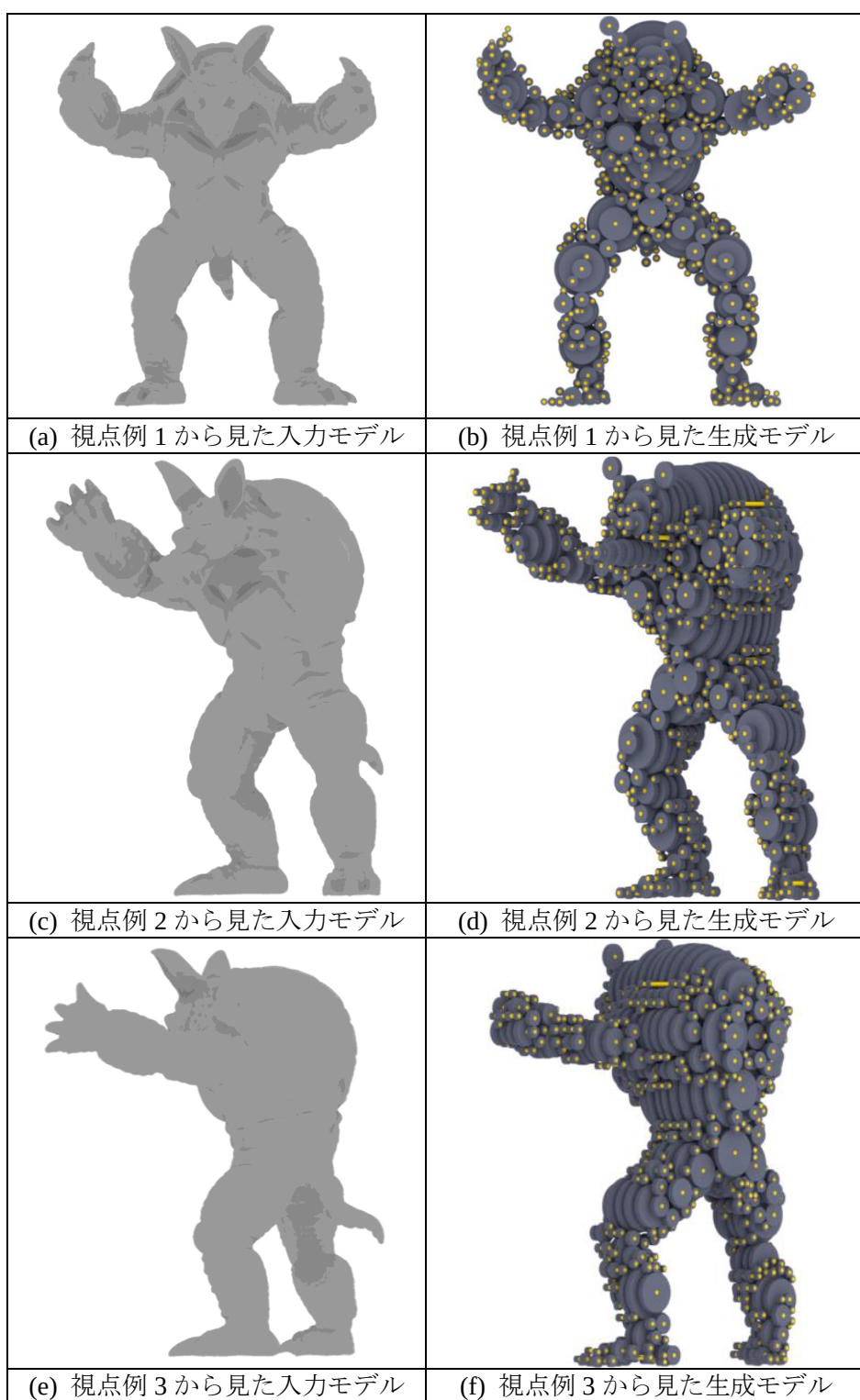


図 48 生成モデル例 9

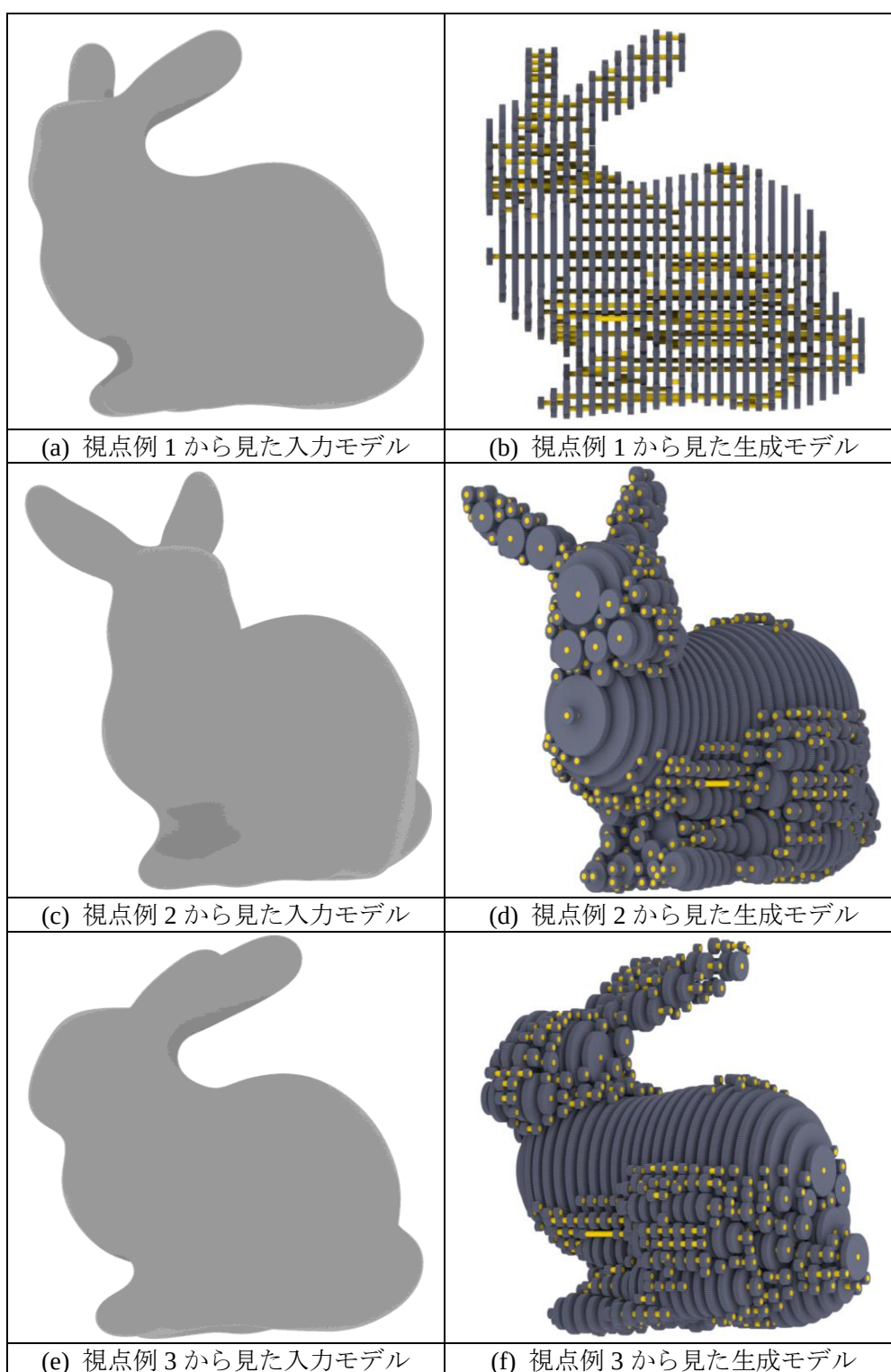


図 49 生成モデル例 10

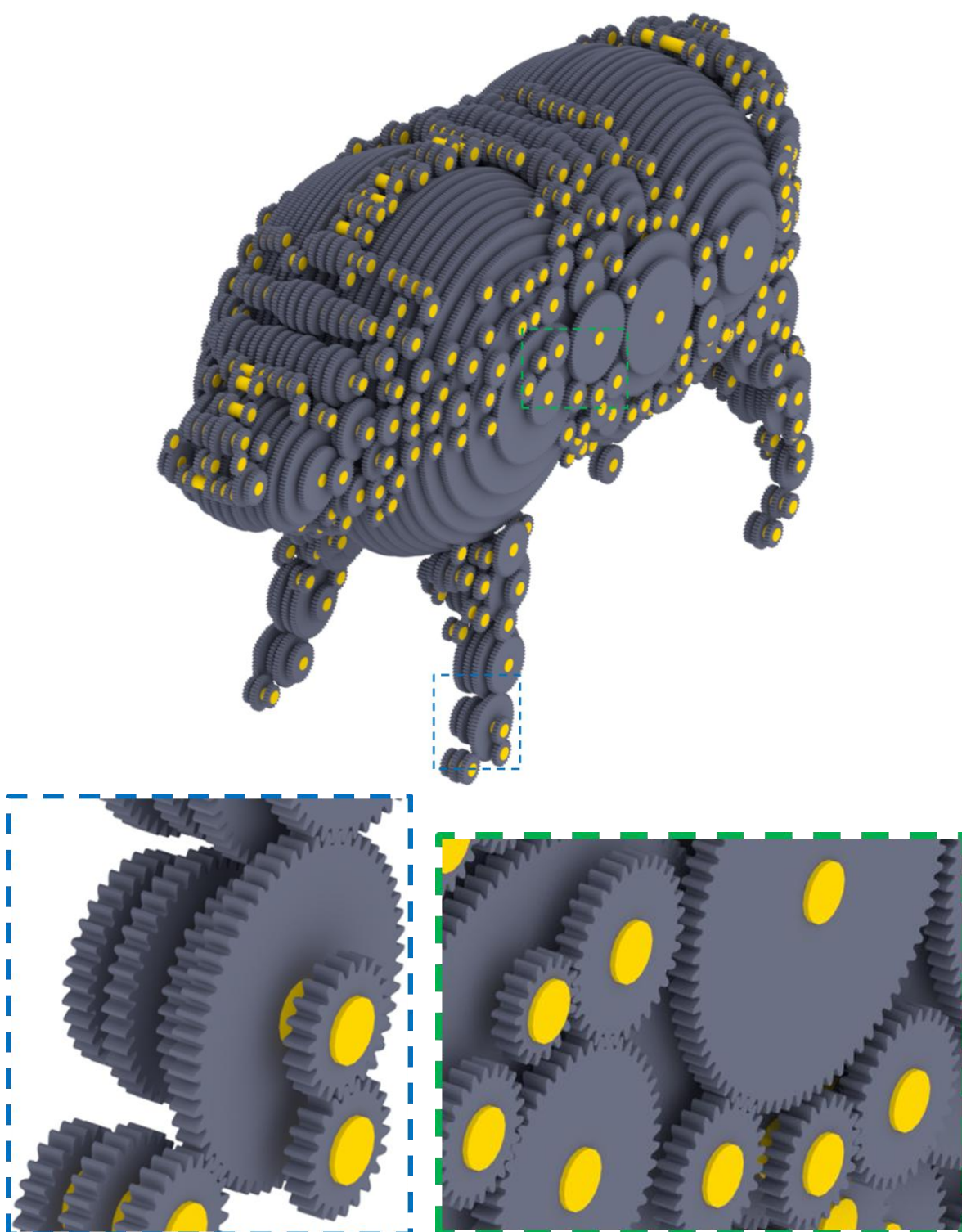


図 50 生成モデルの一部拡大図

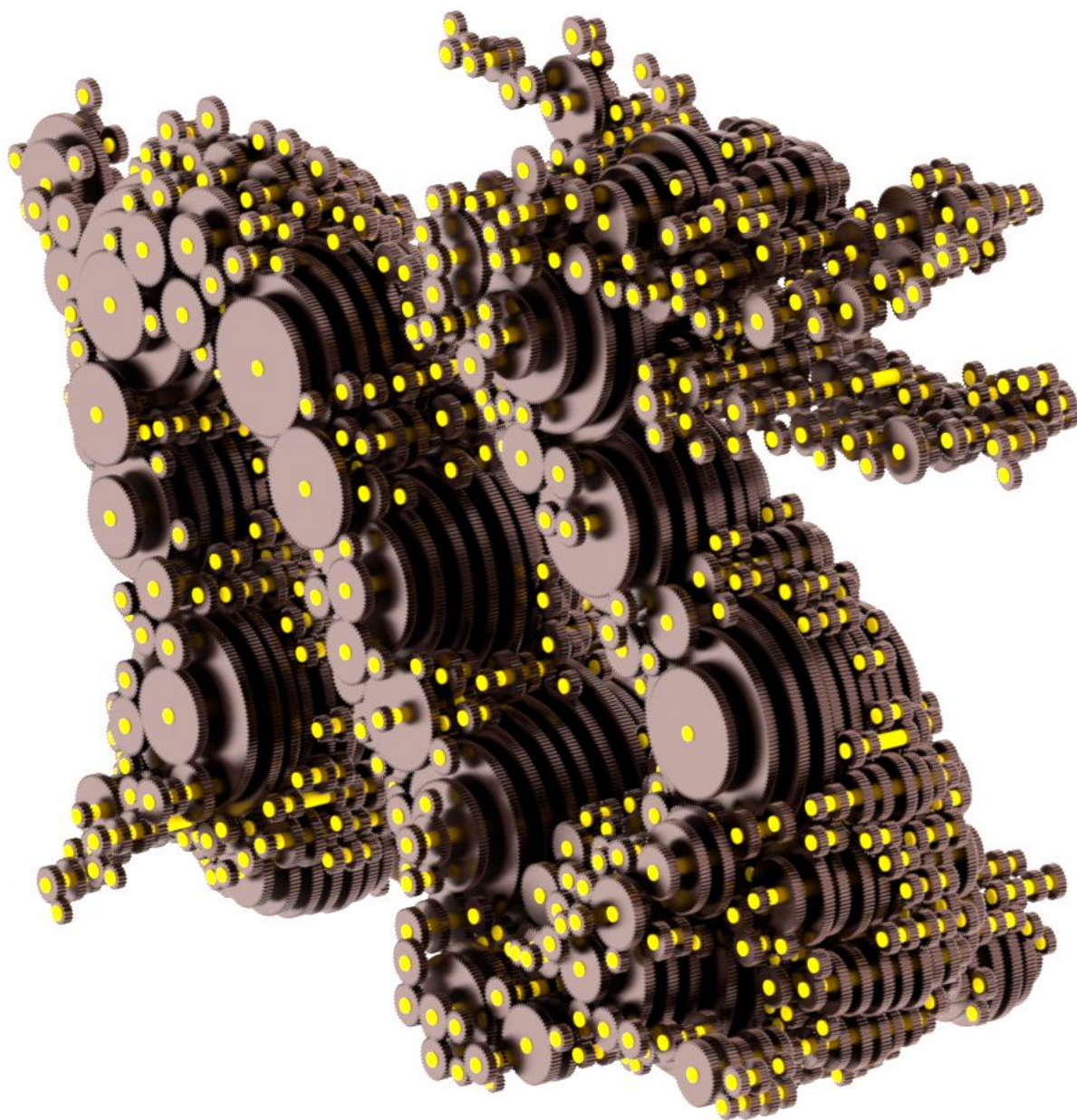


図 51 生成モデルに金属の反射を適応して描画した例 1

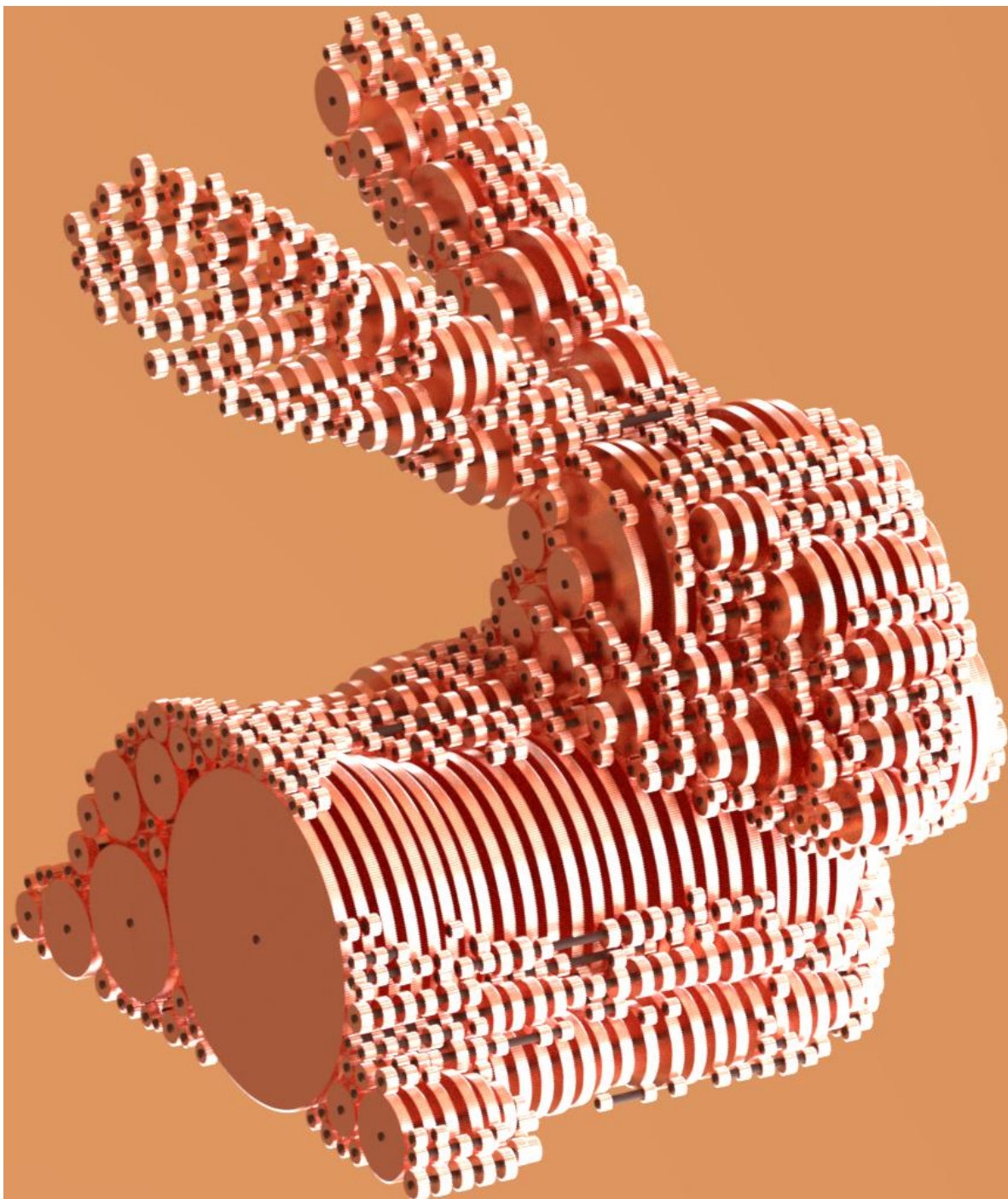


図 52 生成モデルに金属の反射を適応して描画した例 2

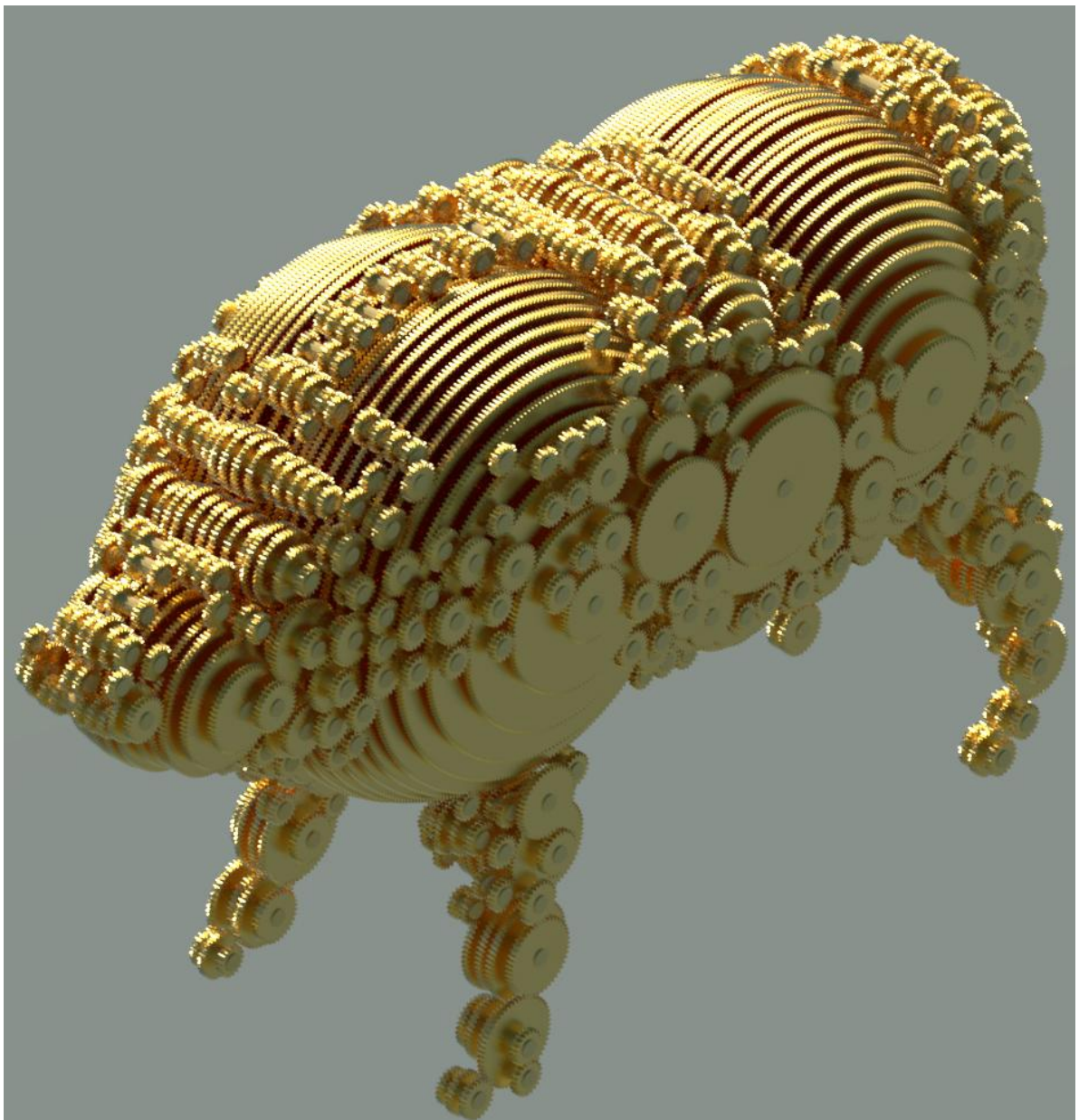


図 53 生成モデルに金属の反射を適応して描画した例 3

3.2 結果の考察

図 42のウサギ形の歯車集合体において、入力モデルの片耳にあたる部分に歯車が配置されていない状態であることがわかる。これは、その部分の断面のセル数が少ないため、もしくは断面が小さすぎたためであると考えられる。これの対処法として、テクセル数の増加、モジュールを小さくするなどである程度対処できるが、本処理は離散的であるセルの中心位置ごとに計算するため、断面が歯車を置けるのに十分な大きさを持っていたとしても歯車が置けないという状況を完全になくすことは不可能である。

さらに本手法は、配置済みの歯車に歯車セットを必ずかみ合わせて置いていく方法である。このことから、特定の断面に歯車が置かれなかった場合、その特定の断面とのみ X 方向に連続する断面にも歯車が置けないといった問題がある。この問題が発生していると思われるのが図 42で、その問題の箇所を図 54に示す。図 54ではマークしているウサギの耳の付け根に当たる断面に歯車が配置されなかったため、その断面としか X 方向に連続しなかったと思われる耳の先方向の断面にも、歯車が置かれなかったと考えられる。

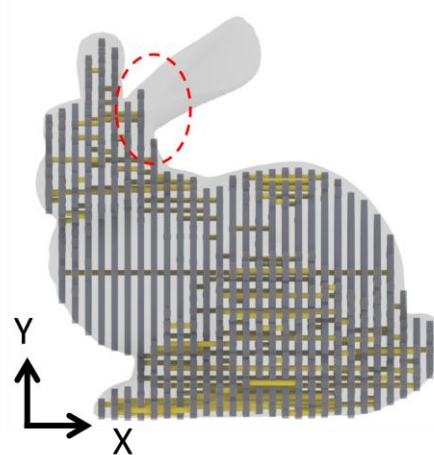


図 54 断面に歯車が置かれなかった断面とのみ X 方向に連続する断面で起きる問題

また、生成されたモデルの傾向として、しわや眼球・口周辺の凹凸、小さい突起などの細かい形状的特徴が消失しているが、胴体の形状など大きい形状的特徴を保持していることが確認できる。このことから本手法は、入力モデルの大局的な特徴を保持する歯車集合体を生成できる手法

であるといえる。

図 42のウサギの耳の付け根のような、特定の断面が他の断面に比べ小さい入力モデルで、大きな形状的特徴を保持する歯車集合体を生成する場合には、前処理として入力モデルのメッシュスムーズ化が有効である。メッシュスムーズ化は、入力したモデル表面の曲率を緩やかにする処理で、メッシュのスムーズ化の例として、メッシュスムーズ化前のモデルを図 42(a)(c)(e)に、そのモデルのメッシュスムーズ化後のモデルを図 49(a)(c)(e)に示す。

図 42の入力モデルに MeshLab[19]でメッシュスムーズ化を行ったモデルを入力モデルとして、図 42と同じ入力パラメータで歯車集合体を生成した結果を図 49に示す。メッシュスムーズを行った歯車集合体は図 42とは違い両耳に歯車が配置され、入力モデルの大局的な特徴を保持できている。前処理としてメッシュスムーズ化を行うことにより、形状の大局的な特徴を残しつつ、図 42の入力モデルの耳の付け根のような、入力モデルのくびれている部分の断面を大きくすることができる。

また、メッシュスムーズ化を行うことで入力モデルは皺や眼球などの微細な形状的特徴を消失してしまう問題があるが、本手法で生成した歯車集合体は細かい形状的特徴を保持できないので、結果として、生成した歯車集合体が変わることはないと考えられる。

第 4 章 まとめ

本論文では、入力モデルの大局的な特徴を保持し、歯車の特性を表現した歯車集合体を自動生成する手法を提案した。提案手法は、入力モデルから複数の平行な断面を計算し、断面ごとにその断面形状を近似するように歯車を敷き詰めることで、入力モデルの大局な特徴に当たる近似形状を得ている。そして、歯車の特性を表しながら歯車を敷き詰める手法として、歯車の規格を用いながら歯車セットを1つずつ置いていく局所的な手法を採用している。また、1つずつ置いていく局所的な手法のために起こる問題を防ぐべく、好ましくない歯車のサイズ・配置に制限をかける対策を行った。

今後の課題として以下の点が挙げられる。

- ・現状のシステムでは図 54に示すウサギの耳のような、入力モデルの近似形状を得るのに必要な部分でも断面が小さければ、歯車が配置されない問題がある。この問題を解決するため、3.2 節でのような前処理や断面が小さい場合でも、その断面が入力モデルの近似形状を得るのに必要ならば、断面からはみ出ても歯車を置くなどの断面の大きさの制約にとらわれないで歯車を配置する仕組みが必要である。
- ・本手法はテクセル数が大きいほど、レイヤ数が多いほど、モジュールが小さいほど計算コストが高くなる。これらの計算コストを並列化や GPU 化などで低減を図る必要がある。

また、本研究の有用性を増す工夫として以下の点が考えられる。

- ・本手法の歯車セットスコアの基準は、断面の形状を考慮して大きい歯車が優先的に配置されるように定義している。歯車セットスコアの基準をモデルの表面からの位置も考慮するように定義すると、モデル表面に近い位置に歯車が優先的に配置されるようになるなど、歯車セットスコアの定義を変更することで出力する歯車セットの傾向を変えることができると考えられる。このように歯車セットスコアの定義を編集可能にすることで、ユーザの所望の見えるような歯車集合体を生成でき、ユーザの希望に則した表現手法として有用性が増すと考えられる。
- ・歯車集合体を実際に実体化できるような形で生成できれば、制作物の表現の幅を増やすことができ有用性が増すと考えられる。本手法を用いて生成モデルを実体化する場合に、現状の生成モデルでは歯車・歯車軸が宙に浮いているため、支持する仕組みが必要になる。仕組みとしては、図 55に示すように歯車軸方向の歯車の間に歯車軸を支える穴が開いた透明板を立

てて、歯車軸を保持する方法が考えられる。

その他、歯車と歯車軸を固定された歯車セットを制作する必要がある。また、かみ合った歯車をスムーズに回転させるため、回転する歯車の歯と回転させられる歯車の歯との隙間“バックラッシ”[22]を設ける必要がある。生成モデルを実体化する場合、これらのことを考慮した生成モデルの製造支援の仕組みが必要になる。

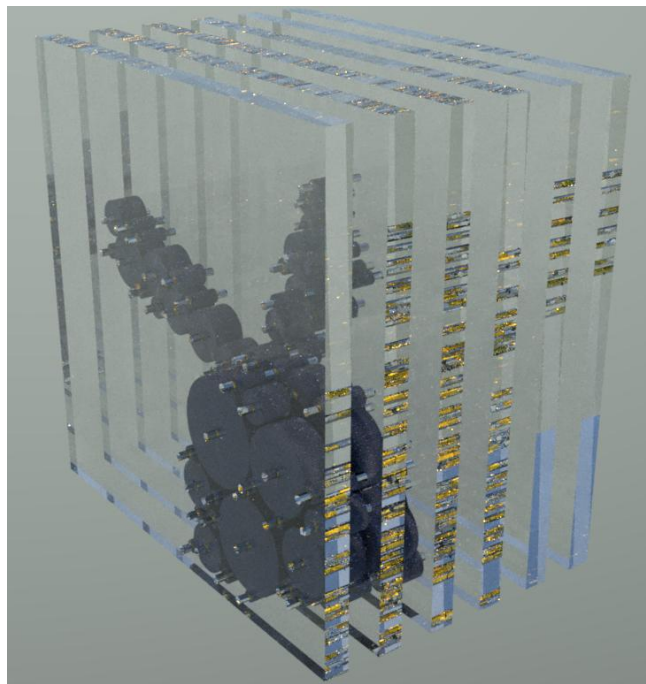


図 55 歯車軸を保持する方法

謝辞

研究活動を進めるにあたり，所属研究室の宮田一乗 教授から丁寧かつ熱心なご指導を受け賜りました．また，同研究室の杜 暁冬 氏，溝口 敦士 氏，櫻井 快勢 氏，石橋 賢 氏，謝 浩然 氏，北 直樹 氏から研究活動に対して多くの示唆を受け賜りました．研究室の皆様へ心から感謝の気持ちと御礼を申し上げたく、謝辞にかえさせていただきます．

参 考 文 献

- [1] Stefan Schlechtweg, Stefan Schlechtweg, Thomas Strothotte, RenderBots—Multi-Agent Systems for Direct Image Generation, Computer Graphics Forum, volum24, pp.137-148(2005).
- [2] Jie Xu, Craig S. Kaplan, Calligraphic packing, GI '07 Proceedings of Graphics Interface 2007, pp.43-50(2007).
- [3] Junhwan Kim, Fabio Pellacini, Jigsaw Image Mosaics, SIGGRAPH '02 Proceedings of the 29th annual conference on Computer graphics and interactive techniques, pp.657-664(2002).
- [4] Dongwann Kang, Yong-Jin Ohn, Myoung-Hun Han, Kyung-Hyun Yoon, Animation for ancient tile mosaics, NPAR '11 Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering, pp.157-166(2011).
- [5] Niloy J. Mitra, Yong-Liang Yang, Dong-Ming Yan, Wilmot Li, Maneesh Agrawala, ILLUSTRATING HOW MECHANICAL ASSEMBLIES WORK, SIGGRAPH '10 ACM SIGGRAPH 2010 papers, Article No. 58(2010).
- [6] Ran Gal, Olga Sorkine, Tiberiu Popa, A Ila Sheffer, Daniel Cohen-Or, 3D collage: expressive non-realistic modeling, Proceedings of the 5th international symposium on Non-photorealistic animation and rendering, pp.7-14(2007).
- [7] Chongyang Ma, Li-Yi Wei, Xin Tong, Discrete element textures, SIGGRAPH '11 ACM SIGGRAPH 2011 papers, Article No. 62(2011).
- [8] 高橋 和茂, 高井 昌彰, 高井 那美, 近似形状の構築が可能な缶アート制作支援システム, 情報処理学会研究報告. グラフィクスと CAD 研究会報告, Vol.2011-CG-145 No.26(2011).
- [9] 田村 友和, 高井 昌彰, 高井 那美, ユニット折り紙を用いた3次元メッシュモデルの近似形状構築, 情報処理学会研究報告. グラフィクスと CAD 研究会報告, Vol.2010-CG-141 No.13(2010).
- [10] Insect Lab Studio, <http://insectlabstudio.com>
- [11] Kui-Yip Lo, Chi-Wing Fu, Hongwei Li, 3D polyomino puzzle, SIGGRAPH Asia '09 ACM

- SIGGRAPH Asia 2009 papers, Article No. 157(2009).
- [12] Alejo Hausner, Simulating decorative mosaics, SIGGRAPH '01 Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pp.573-580(2001).
 - [13] Yoshinori Dobashi, Toshiyuki Haga, Henry Johan, Tomoyuki Nishita, A Method for Creating Mosaic Images Using Voronoi Diagrams, Proceedings of Eurographics 2002 Short Presentations, pp.341-348(2002).
 - [14] Kaleigh Smith, Yunjun Liu, Allison Klein, Animosaics, SCA '05 Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation, pp.201-208(2005).
 - [15] Ketan Dalal, Allison W. Klein, Yunjun Liu, Kaleigh Smith, A spectral approach to NPR packing, NPAR '06 Proceedings of the 4th international symposium on Non-photorealistic animation and rendering, pp.71-78(2006).
 - [16] 根本 良三, 絵とき歯車の基礎のきそ, pp.20-21, pp.20-21, pp.30-31, pp.41-42, 日刊工業新聞社, 2007, ISBN 978-4-526-05814-1.
 - [17] Yu-Kun Lai, Shi-Min Hu, Ralph R. Martin, Surface mosaics, The Visual Computer: International Journal of Computer Graphics, volume22, pp.604-611(2006).
 - [18] V. A. Dos Passos, M. Walter, 3D virtual mosaics: Opus Palladium and mixed styles, The Visual Computer: International Journal of Computer Graphics, volume25, pp.939-946(2009).
 - [19] MeshLab , <http://meshlab.sourceforge.net>
 - [20] Mitsuba - physically based renderer , <http://www.mitsuba-renderer.org/>
 - [21] デジタル画像処理編集委員会, デジタル画像処理, pp.179-180, 財団法人 画像情報教育振興協会, 2004, ISBN978-4-903474-01-4.
 - [22] ジャパンマシニスト社編集部, JM ブックシリーズ 歯車, pp.67, pp.21-39, 株式会社ジャパンマシニスト社, 1969, ISBN978-4-88049-001-4.
 - [23] 技能士の友編集部, 技能ブックス(13) 歯車のハタラキ, pp.32-33, 株式会社大河出版, 2007, ISBN978-4-88661-413-1.
 - [24] 蓮見 義久, パソコンによる歯車の設計計算, pp.35-41, pp.20-32 理工学社, 1993, ISBN4-8445-2520-4.

発 表 論 文

- [1] 下中 進, 宮田 一乗, 入力モデルの近似形状を持つ歯車集合体モデルの生成, NICOGRAPH 2012, 京都大学 学生情報メディアセンター 南館 2 階, 2012/11/16. (優秀論文賞受賞)