

染料移動モデルを利用したタイダイ染めにおける じゃばら絞りシミュレーション

川添和泉^{†1} 濱川礼^{†2}

本論文では、染色のシミュレーションを用いて専門的な技術を必要とせずに、タイダイ染めじゃばら絞りの染色模様を簡単に作成できる方法を提案する。染色は紙や布を染める技術として古くから発展してきた。現在でも繊維産業、伝統文化、趣味として、幅広い価値がある。その種類は多岐に渡り、その中でタイダイ染めは、染料を付ける際に布を縛ることのできる様々な味の色合いを魅力とした特殊な技法である。染めの段階が手作業なため、酷似していても同じ物は一本とない。そのため染色には経験や技術が必要とされ、さらには科学的な要素にも影響され、複雑な模様が作り出される。しかし、これまでの IT における研究では絵画の質感表現が多く、特に画材や紙のモデリング・レンダリングの技法に関するものが多く、染色を題材としたものは極めて少ない。そこで、本研究では専門的な技術や労力を必要とせずに染色模様をデザインするための支援として、シミュレータの開発を行う。絞り染めにおける染料移動モデルを用いて染料供給分布を計算し、単色の染料を複数色の表現に変え、染物独特の手作業による不確実な要素をランダム性によって表現し、タイダイ染めのじゃばら絞りに特化したシミュレーションを行う。

The bellows iris diaphragm simulation in the tie dyeing using a dye move model

IZUMI KAWAZOE^{†1} REI HAMAKAWA^{†2}

In this paper, the method of creating easily the dyeing pattern of a tie dyeing bellows iris diaphragm is proposed, without needing special technology using the simulation of dyeing. Dyeing has developed as technology of dyeing paper and cloth for many years. There is broad value as a textile industry, traditional culture, and a hobby even now. It goes across the kind variably, and in it, when tie dyeing attaches dye, it is a special technique which made charm various clever tone made by binding cloth. Since manual labor [the stage of dyeing], even if it resembles closely, there is no same thing in one. Therefore, experience and technology are needed for dyeing, it is influenced by the still more scientific element, and a complicated pattern is made. However, at the research in old IT, there is much textures expression of pictures, especially there are many which relate to the technique of the modeling rendering of painting tools or paper, and there are very few things dealing with dyeing. So, in this research, a simulator is developed as support for designing a dyeing pattern, without needing special technology and labor. This simulator performs the simulation which specialized in the bellows iris diaphragm of tie dyeing, by calculating using the dye move model in variegation, and changing monochromatic dye into expression of two or more colors, and expressing by random nature, the uncertain element by manual labor peculiar to dyed goods.

1. はじめに

本論文では、染色のシミュレーションを用いて専門的な技術や労力を必要とせずに、タイダイ染めのじゃばら絞りにおける染色模様を簡単に作成できるシミュレータを提案する。森本らの絞り染めにおける染料移動モデル[1]を用いて、元画像の RGB 値から染料量を算出し、供給分布を計算する。これによって、染料の移動をシミュレートする。さらに、我々が既に[2]で提案した、いくつかのランダム性の付加を応用して、染色をする際に起きる人の手作業による不確実な要素を表現する。単色の染料を複数色の表現に変え、タイダイ染めに特化した技法のシミュレーションを行う。

2. 背景・目的

染色とは、布、革などの繊維質に色素を吸着、結合させることである。染色は世界各地で古くから行われてきた繊維を着色する方法であり、現在でも繊維産業や伝統文化、趣味として、幅広い価値がある。一般に繊維は非常に多種類の方法で染色されるが、染色法は大きく浸染および捺染に分けられる。浸染は染色物全体を戦役に浸して染める無地染の染色法である。捺染は主に糊に染料を混ぜて布に直接すり付けて染める染色法であるが、柄染めとも呼ばれ、抜染や防染もこれに含まれる。捺染による染色方法は、絞り染めや、ろうけつ染め、友禅染めなど、多くの技法が知られている。特にろうけつ染めや絞り染めのように染料の拡散を防ぐことで染色模様を作る技法のことを防染技法という。それらは更に細かな技法に枝分かれしており、染色技法は細かく分類すると何百種類以上に及ぶ[3]。

†1 中京大学大学院
Chukyo University

†2 中京大学
Chukyo University

以上のような手工芸による染色は、布形状の変化と染織の物理的な要素が組み合わさることによって起こる複雑な現象であり、出来上がりの模様をデザインするのは熟練の染色家でも難しい。染色家は長年の経験と知識に基づいて出来上がる模様を予測するが、それでも予測とは異なる染色結果になることも少なくはない。また、染色工程は非常に時間を要し、道具や場所も必要である。

このように、手工芸としての染色技法では、染色家としての経験、技術、労力、知識が必要であり、材料の科学的・物理的な状態にも影響されて、独特で唯一の模様が作り出される。

しかし、現在、染色を題材にした Computer Graphics(CG) 研究は少ない。絵画調の技法や手書き調の質感の表現を扱う Non-Photorealistic Rendering (NPR) の分野では、画材のシミュレーションやストローク描画技法の研究などが進められている[4][5][6][7]。

Curtis らは、水彩絵具を想定し、かすれ、重ね塗り、染み込み、にじみなどの表現を提案している[8]。にじみのモデルは“絵の具の流れ”、“紙に対する色素の着脱”、“絵の具の紙への染み込み”の三つの要素で表現されており、支持体である紙の影響も考慮しながら、水彩画調の“にじみ”をリアルに表現している。水墨画においても染み込みやにじみによる濃淡は質感を表現する上で重要である。

これらはすべて支持体が紙である。紙以外に染み込みの現象が発生する支持体としてよく使われるものに布がある。布は糸によってできており、染み込みの計算をする場合には糸の織り構造を考える必要がある。布への染み込みを想定した工芸の代表は“染色”である。NPR において“染色”の表現は極めて少なく、若干例報告されているのみである[1][2][9][10]。しかし、染物全般を表現するものは現時点ではその他に見当たらない。

そこで、本研究では経験や技術がなく、染色を行ったことがない初心者でも、容易に染色後のデザインを想像できるよう、染色前の簡易なデザイン画像やパラメータの入力から、染色後の布画像を表示するシミュレータの開発を研究目的とする。

我々はすでにタイダイ染めのタイル模様絞りを表現するシミュレータの開発を行っている[2](図 1)。ここでは、森本らの appearance ベース染色シミュレーションモデル[1]を応用して、染料量の拡散を行っている。染料量の拡散は移動率(重み)によって左右される。[2]では、タイル模様絞りによって生じる、タイルを並べたような模様をサンプル画像(図 2)の複製にランダム性を加えて描写することで表現した。ランダム性には三つの要素を提案している。一つは染料拡散における移動率のランダム性。そして、タイル模様絞りを表現する際の折り幅のランダム性。最後に、染料の付与におけるランダム性である。これらは実際の染色作業において、人の手で行われ、それゆえに常に同様な

値によって表現されるとは限らない。そのため、乱数を用いてランダム性を与えることで、染色ならではの人の手作業による不確実性を付加している。

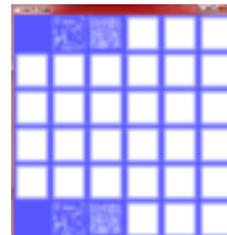


図 1 タイル模様絞りシミュレータ出力結果

Figure 1 Tiled-patterns iris diaphragm simulator output result

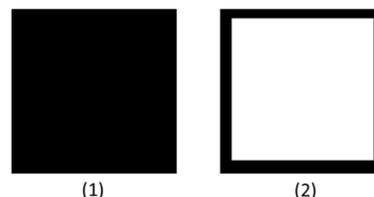


図 2 サンプル画像

Figure 2 Sample pictures

本研究では、[2]を応用し、タイダイ染めのじゃばら絞りにおける模様を表現するシミュレータの開発を行った。染色技法の中で、タイダイ染めは、染料を付ける際、布に施す折り、縛り、縫いなどの様々な手法による布形状の変化と、複数色の染料を用いた様々な味の色合いを魅力とした特殊な技法である。染めの段階が手作業なため、酷似していても同じ物は一本とない。そのため、染色後のデザインを想像することは初心者でなくとも難しい。

じゃばら絞りはタイル模様絞り同様、タイダイ染めの中でも基本になる染色技法である。その方法は、布をじゃばらに折り、数箇所を紐で縛って固定するだけのシンプルなものである。しかし、タイル模様絞りに比べ、折り方の丁寧さ、紐による圧迫の強弱による模様への影響が大きい。そのため、タイル模様絞りよりも、はっきりとしたランダム性を付加する必要がある。

本研究でシミュレータに付加するランダム性は以下の三つを提案する。一つは染料拡散における移動率のランダム性。そして、じゃばら絞りによって生じる模様の擦れのランダム性。最後に、染料の付与におけるランダム性である。タイダイ染めは布に染料を与える際、藍染や草木染めのように染料に布を漬ける方法ではなく、人の手で布に染料をかける方法をとる。また、じゃばら絞りは折りと紐による圧迫によって模様が表現される技法である。じゃばらに折った布の数箇所を紐で縛り、形を固定して圧迫する。そして、そこに染料をかける。その際の縛り方の強弱が、布の圧迫される範囲を決定し、染料拡散に影響を及ぼす。また、縛り方以外に折り方も重要な要素となり、これは模様の位置に影響を与える。タイダイ染めの染料を上からかける方法は、藍染めや草木染めのような漬ける方法に比べ

てムラが出る。これらの要素は人の手で行う以上、ある程度の不正確さが生じる。そこで、圧迫の強弱、模様の擦れ、染料付与にランダム性を取り入れ、染色過程における人の手による紐の縛り方、布の折り方、染料液のかけ方の不確実性を表現する。

3. 提案手法

3.1 概要

本シミュレータでは、まずユーザが用意した染色デザイン画像とユーザが指定した染色のための簡単なパラメータから、じゃばら絞りによって生じる模様をデザイン画像に描画し、描画後の画像から RGB 値を取得し、仮定の染料量を算出する。この染料量から、森本らの appearance ベース染色シミュレーションモデル[1]を応用して、染料量の拡散を行う。染料量の拡散は移動率（重み）を決める。本研究では、この移動率に乱数を用いてランダム性を与えることで、染色ならではの不確実性を付加している。拡散と同時に、隣接セルに移動した染料の色をコピーしておく。拡散後の染料量とコピーした RGB 値を元に平均化処理を行い、複数色の色付けと染色後のデザインを表現し、ユーザに提示する。

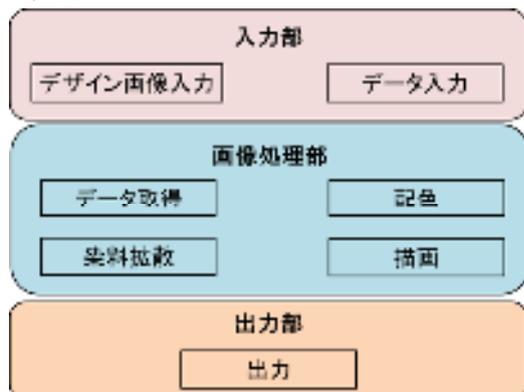


図 3 シミュレータの全体構成図

Figure 3 The whole simulator block diagram

3.2 ランダム性

ここでは、シミュレータに付加したランダム性について記述する。

(1) 紐による圧迫の強弱

本研究では、乱数 R_1 を用いて染料移動率 α にランダム性を与えることで、染色ならではの不確実性を増す。染料移動率については、3.3(3)で詳細を述べる。乱数 R_1 は、0～1の間をとる。このランダム性については、ユーザ入力の「紐による布の縛り方の強弱」を反映させる。ユーザは「紐による布の縛り方の強弱」を「強め」、「やや強め」、「やや弱め」、「弱め」の中から選択する。「強め」は紐による圧迫が強くなり、より確実に布が固定されるため、ランダム性がなくなる。逆に「弱め」は紐による圧迫が弱くなり、

布が固定されないため、ランダム性が大きくなる。これらの選択肢にはそれぞれ値 m (表 1)が入力されており、 m を乱数 R_1 に乗算することでランダム性の有無を調整している。その式を以下に示す。

$$\alpha = R_1 \div 100 \times m$$

m の値の大小も R_1 の値次第で、 α の数値の大小が反転することもある。

ただし同行同列の移動率は均一に定義する。

| 縛り方の強弱 | m値 |
|--------|-----|
| 強め | 0 |
| やや強め | 0.4 |
| やや弱め | 0.6 |
| 弱め | 1 |

表 1 縛り方の強弱における値

Table 1 The value in the strength of a way to bind

(2) 布の折りによる模様の位置

じゃばら絞りでは、布の折り方によって模様の位置に影響が出る。折り目は人の手で折る以上、ある程度の不正確さが生じる。そこで、布の折り方によるランダム性を表現するため、模様の描画位置に乱数を用いて、人の手による折り目の擦れを表現する。実際のじゃばら絞りの模様は、楕円のような形で生じることが多い。本シミュレータでは、この楕円をデザイン画像に描画し、その後染料拡散処理を行う。楕円の描画位置は、実際のじゃばら絞りにおける折り目の数と模様の数の関係式から導くが、詳細は 3.3(2)で述べる。

本来等分されている楕円の中心点に、乱数 R_2 を加算することで折り方による擦れを発生させる。 R_2 は実験を行い、10～20pixelの間と一様に定義した。

この擦れの生成には、ユーザが入力した「折り方の丁寧さ」が影響する。ユーザは「折り方の丁寧さ」を「丁寧」、「やや丁寧」、「やや粗雑」、「粗雑」の中から選択する。丁寧であるほど、ランダム性は低くなり、粗雑なほどランダム性は高くなる。これらの選択肢にはそれぞれ値 n (表 2)が入力されており、 n を乱数 R_2 に乗算することでランダム性の有無を調整する。

| 折り方の丁寧さ | n値 |
|---------|-----|
| 丁寧 | 0 |
| やや丁寧 | 0.4 |
| やや粗雑 | 0.6 |
| 粗雑 | 1 |

表 2 折り方の丁寧さにおける値

Table 2 The value in the care of a way to fold

(3) 染料付与

タイダイ染めの特徴の一つは、布への染料の与え方である。友禅染は染料をつけた筆で絵を描くように、藍染や草木染めは染料の溶液を漬けることで染色を行う。しかし、タイダイ染めはボトルに入れた染料の溶液を布に垂らすことで染色する。この作業は人の手によって行われ、その加減で付与のされ方は不均一となる。そのためムラになったり、本来作り手が予想していなかった部分に染料が与えられたりすることがある。

そこで、本研究では染料拡散処理前の入力画像に対してランダムに斑点を描画し、染料付与のランダム性を付与することを提案する。「紐による布の縛り方の強弱」や「折り方の丁寧さ」同様、ユーザが入力した「染料付与の丁寧さ」によってランダム性の有無を調整する。ユーザが選択する選択肢と、それに対する値 l を以下の表に示す。

| 染料付与の丁寧さ | l値 |
|----------|-----|
| 丁寧 | 0 |
| やや丁寧 | 0.4 |
| やや粗雑 | 0.6 |
| 粗雑 | 1 |

表 3 染料付与の丁寧さ

Table 3 Care of dye grant

今回、本シミュレータにはこの染料付与のランダム性を実装することができなかった。

3.3 内部処理

図 1 に実装したシステムでの処理手順を示す。各処理について以下に説明する。

(1) データ入力

入力部ではユーザによるパラメータ入力と、ユーザが別途用意したデザイン画像(図 4)の入力を行う。ユーザが入力を行うパラメータは、じゃばら折りの回数、折り方の丁寧さ、紐による布の縛り方の強弱である。

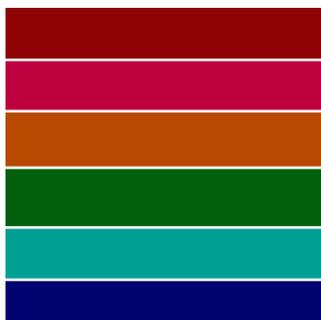


図 4 デザイン画像例

Figure 4 The example of a design picture

今回、デザイン画像は 600×600 pixel を標準としている。

このデザイン画像を元に染料拡散を行う。

デザイン画像は、ユーザが別途用意する。ユーザはデザイン画像を用意する際、以下の条件を守る必要がある。

- 紐で布を縛る位置に白線をひく
- 布に与える染料の色で画像を描く

(2) データ取得

データ取得では、まず出力画像の生成を行う。出力画像は実際の染色における布の大きさである。本システムでは、 600×600 pixel を布サイズとし、1 pixel を一本の糸幅として定義する。また、本研究では織構造を平織りと定義して染料拡散を行うものとする。

次に、デザイン画像にじゃばら絞りによって表現される模様を描画する。模様は縦長の楕円を基本としている。画像サイズとユーザ入力のじゃばら折り回数を基に、描画する楕円の数と大きさ、中心点の位置を決定する。以下にその式を示す。

- 楕円の数

$$d = j - 1$$

d : 楕円描画回数

j : じゃばら折り回数

- 楕円中心点

$$x = (n + 1) \times (\text{width} \div (j + 1)) + (\text{width} \div (j + 1) \div 2) + R_2$$

$$y = \text{height} \div 2$$

x : 楕円の中心点の x 座標

y : 楕円の中心点の y 座標

n : 現在の楕円描画回数

j : じゃばら折り回数

R_2 : 「折り方の丁寧さ」から生成した乱数

width : デザイン画像サイズ横

height : デザイン画像サイズ縦

- 楕円サイズ

$$w = (\text{width} \div (j + 1) \div 2) + (\text{width} \div (j + 1) \div 5) \div 2$$

$$h = \text{height} - (2 \times (\text{height} \div (j + 1) \div 2) + ((\text{height} \div (j + 1) \div 5) \div 2)$$

j : じゃばら折り回数

width : デザイン画像サイズ横

height : デザイン画像サイズ縦

これらの式によって決定したパラメータを基に、模様の描画を行い、図 5 のようにデザイン画像を変更する。

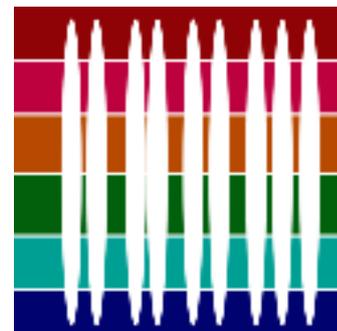


図 5 模様描画後のデザイン画像

Figure 5 The design picture after drawing encaustic

変更したデザイン画像のデータを取得し、さらに初期値の計算を行って、構造体へ保存する。取得するデータは、1pixel ごとの RGB 値である。この RGB 値を用いて、1pixel 内に与えられた初期染料量を算出する(図 6)。

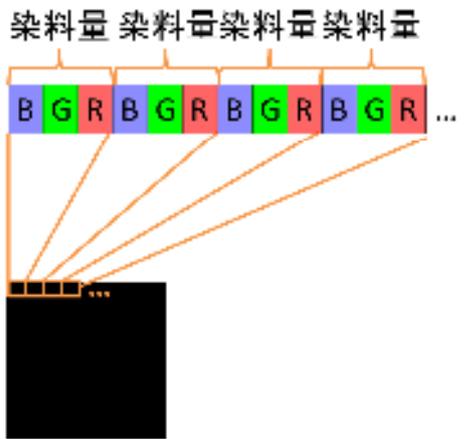


図 6 初期染料量算出

Figure 6 Calculation of the amount of initial quantity of dye

取得した RGB 値と算出した初期染料量を、1pixel ごとに画像データ用の構造体へ保存し、データ取得は終了する。

(3) 染料拡散

染料拡散は、森本らが提案した appearance ベースの染色モデル[1]を応用し、セルオートマトンを用いた染料量の平均化処理を用いる。このモデルは、計算セルの単位を糸の幅で定義しているため全体の計算点が少なく、高速に処理を行うことができる。

表 4 に染料拡散に必要な各セルが持つ要素を示す。

| 項目 | 変数 | 説明 |
|------------------|------------|---------------------|
| 染料の量 | s | 現在保有している染料の総量 |
| 染料の移動に必要な最小量の染料量 | ϵ | |
| これ以上受け取れない染料の量 | c | |
| 位置 | i, j | セルの位置 |
| セルにおける染料の量 | $s_{i,j}$ | 現在保有している染料の総量 |
| 染料移動率の係数 | α | 糸の織り方の種類がパラメータ化したもの |

表 4 各セルが持つ要素

Table 4 The element which each cell has

本研究では簡単なセルオートマトンを用いて布内での染料の拡散を表現する。染料拡散モデルは染色の表現研究である参考文献[1]で森本らが提案した appearance ベースモデルを参考にして染料の浸透を表現する。appearance ベースモデルは、布に与えた染料を、隣のセルとの染料の量の差分を近傍数で割った分だけ着目セルに移動するものである。布の構造に従って定めたセルの三つのパターンとの位置関係(図 7)によって、近傍セルとの染料移動率 α (重み)

を決める。森本らの研究では、染料移動率はシミュレータを実行し、実際の染物と比較することで調整しているが、我々はこれに 3.2(1) で前述した乱数 R_1 を用い、シミュレーション内で変化が起こるよう設定している。

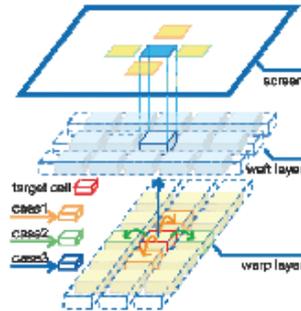


図 7 拡散におけるセルとセルの関係[1]

Figure 7 Relation between the cell and a cell in diffusion

このような条件のもとに、染料の 4 近傍拡散を行う。図 8 に[1]を応用した擬似コードを示す。これは 1 タイムステップに行う染料拡散の処理であり、これを任意のタイムステップ数の分だけ繰り返すことによって拡散が行われる。

```

s' ← s
for all cells(i, j) do
  for each cell(l, m) ∈ neighbors(i, j) do
    if si,j > ε and si,j > sl,m and sl,m > δ then
      Δs ← max(0, min(si,j - sl,m, cl,m - sl,m) α)
      s'i,j ← s'i,j - Δs
      s'l,m ← s'l,m + Δs
    end if
  end for
end for
s ← s'
    
```

図 8 染料拡散の擬似コード

Figure 8 The pseudo code of dye diffusion

拡散と同時に、ターゲットセルから隣接セルへ移動した染料の RGB 値を、複製して保存しておく。この値と染料量を基に、セルの色が決定する。

(4) 配色

染料の拡散モデルによって各セルの染料の量が決まると、その量に応じて色を決める。この色は実際に染料を用いて生成した染みの色素のデータから決定する。実際の画像から最も明度の高い部分と低い部分の色素データを取り出し、線形補間で間の色を決定する。必要であれば明度の中間値の部分のデータも取り出して補間を行う。色を決める線形補間の式は、赤成分のみを見たとき、以下の式のようにになる。

$$R_p = \frac{((255 - R_b) \div 255) \times ((1 - q)R_a + qR_b)}{+ R_b}$$

$$q = \frac{c_p - c_a}{c_b - c_a}$$

R_p : 赤成分の結果
 R_a : 元の布の色の赤成分
 R_b : 染料の色の赤成分
 c_p : 現在の染料量
 c_a : 保有できる最少染料量
 c_b : 保有できる最大染料量

青成分, 緑成分においても同様の線形補間を行う.

(5) 描画

描画は染料拡散の直後に行う. 出力画像に染料拡散処理と配色を施したデザイン画像を描画する.

(6) 出力

描画した出力画像を出力し, 画像ファイルとして保存する.

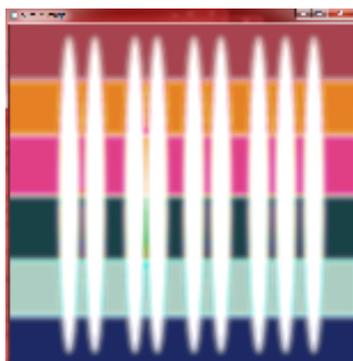


図 9 出力イメージ
Figure 9 Output image

4. 結果

経験や技術がなく, 染色を行ったことがない初心者でも, 容易にタイダイ染めのじゃばら絞りデザインを想像できるシミュレータを開発した.

4.1 実験

表 5 はランダム性の入力内容と出力結果画像の図の関係である.

| | 布の絞り方 丁寧 | 布の絞り方 やや丁寧 | 布の絞り方 やや粗雑 | 布の絞り方 粗雑 |
|---------------|-------------|---------------|---------------|-------------|
| 紐の締め方 弱め | 図8 | 図9 | 図10 | 図11 |
| 紐の締め方 やや強め | 図12 | 図13 | 図14 | 図15 |
| 紐の締め方 やや強め | 図16 | 図17 | 図18 | 図19 |
| 紐の締め方 弱め | 図20 | 図21 | 図22 | 図23 |

表 5 ランダム性の入力内容と出力結果

Table 5 The contents of an input of random nature and the output result

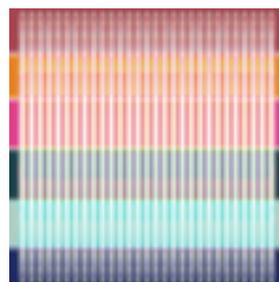


図 8

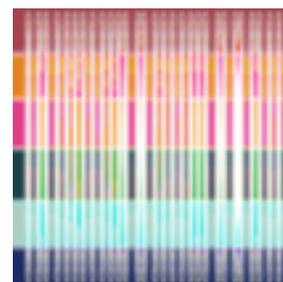


図 9

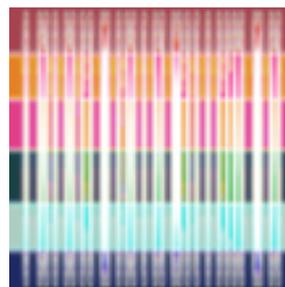


図 10

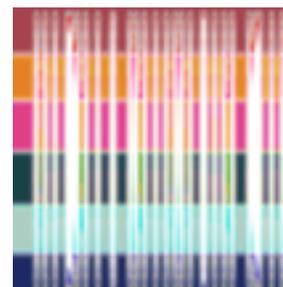


図 11

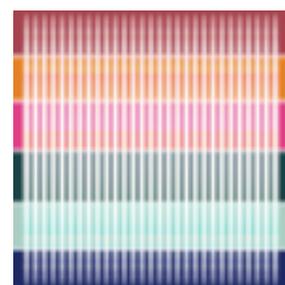


図 12

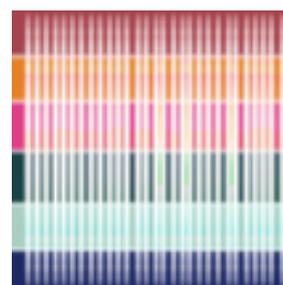


図 13

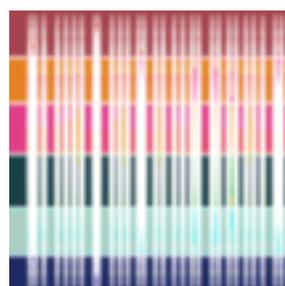


図 14

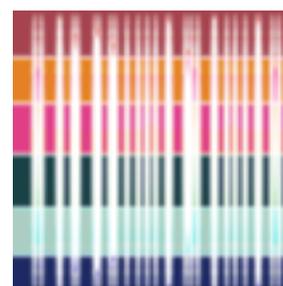


図 15

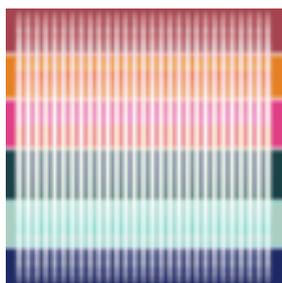


図 16

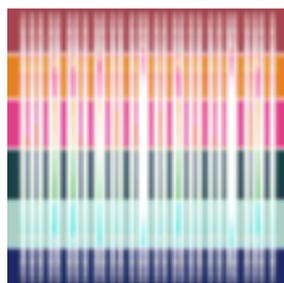


図 17

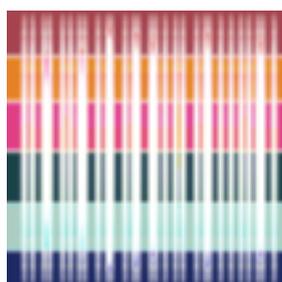


図 18

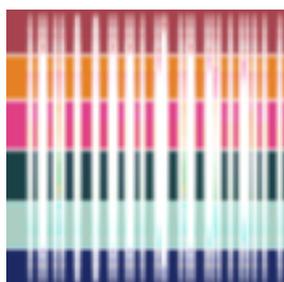


図 19



図 20



図 21

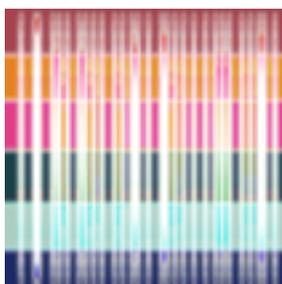


図 22

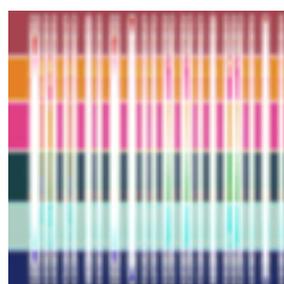


図 23

以下の図 24, 図 25, 図 26 は実際に染色を行った布である。図 25 はじゃばらを丁寧に折り, 上方は紐を強く, 下方は紐を弱く縛った。図 24 はじゃばらをやや雑に折り, 紐をやや強く縛った。図 26 はじゃばらを雑に折り, 図 24, 図 25 より強く紐を縛り, 圧迫したものである。



図 24



図 25



図 26

4.2 考察

実際に染色を行った図 24 から図 26 と本シミュレータによって得られたシミュレーション結果(図 8 から図 23)を比較する。

まず, 図 8 から図 11 の出力結果はいずれも紐による圧迫のランダム性がない状態, つまり染料移動のランダム性がない状態のシミュレーションである。これらの図は, 布の折り方のランダム性が表現されている。図 8 から図 11 のシミュレーション結果から, 折り方の差を表現することができた。これはその他のシミュレーション結果図からも伺える。実際に染色した布と比較すると, 「折り方の丁寧さ」でランダム性の強い「粗雑」を選択した図 11, 図 15, 図 19, 図 23 がこれに近い。ただし, 折りの粗雑さはシミュレーション結果より少ない。シミュレーション結果はいずれも縦の連結がはっきりしている。これに対して, 縦の連結がはっきりと現れるのは, 図 26 のみである。また, 図 24 は折り方を意識してやや雑に折ったため, 布に対して模様が斜めに入っている。これに値するシミュレーション結果は得ることができなかった。

図 8, 図 12, 図 16, 図 20 は折り目幅のランダム性がない状態のシミュレーションである。これらの図は, 染料移動率のランダム性が異なる。一見すると, これらの図に差はないように思われる。しかし, これらを同倍率で拡大すると, 配色に僅かながら影響を及ぼしていることが分かる。図 8, 図 12, 図 16, 図 20 を同倍率で拡大し, 並べたものを図 27 に示す。



図27 図8, 図12, 図16, 図20の比較
Figure27 Comparison of Fig. 8, Fig. 12, Fig. 16,
and Fig. 20

我々は染料移動率の変化は染料の遷移に影響を及ぼし、この値をランダムに変化させることで、染料の移動範囲に差が出るものと想定していた。シミュレーション結果は想定と異なり、図27のように配色に影響を及ぼしている。

4.3 展望

以上のことから、今後の展望として染料移動率のランダム性を見直しが必要である。染料移動率のランダム性は想定外の影響を及ぼした。これらの影響は実際の染色にも起こりうるのか、また起こるとしたらそれは染色の手順の何を原因として発生しているのかを考慮する必要がある。実際には起こりえない現象であれば、このランダム性はシミュレーションから除外すべきである。

実際のじゃばら絞りでは、じゃばらが細かければ細かい模様に、大きければ大きい模様になる。また、紐をきつく縛れば、その模様のいくつかは縦に連結し、大きい楕円や線となって表現される。本シミュレータでは、その要素を取り入れることができなかった。よって、今後はこれらの要素を紐の縛り位置の取得やさらなるランダム性を用いて表現する必要がある。

また、今回実装できなかった染料付与のランダム性を取り入れ、さらなる不確実性を与えることで、で見られるようなランダムに染料が移った部分を表現することができる。

タイダイ染めじゃばら絞りには、縦方向のじゃばら折りによって模様を作る方法の他に、あらかじめ布に記した図の線に沿ってじゃばら折りをし、じゃばら絞りの模様を星や円の形に表現することができる。これらのじゃばら絞りの応用にも対応したい。

また、本研究において着目したタイダイ染めのじゃばら絞り以外にも染色には様々な技法がある。今後は他の技法もシミュレーションできるように付加していきたい。

参考文献

- 1) 森本有紀: 布染色系ノンフォトリアリスティックレンダリングに関する研究, 九州大学
- 2) 川添和泉, 濱川礼 “染料移動モデルを用いたタイダイ染めシミュレーション” グラフィクスとCAD研究発表会 2014
- 3) きもの風土記 <http://mustang.c-mash.co.jp/>
- 4) Guo Q., Kunii T. L.: Modeling the diffuse paintings of 'sumie', Modeling in Computer Graphics (1991).
- 5) Qing Zhang, Y. Sato, J. Takahashi, K. Muraoka and N. Chiba.: Simple Cellular-Automaton-based Simulation of Ink Behavior and Its Application to Suibokuga-like Rendering of Trees, The Journal of Visualization and Animation, pp. 27-37, 1999.
- 6) 佐藤 陽悦, 藤本 忠博, 村岡 一信 他 “ストロークベースの水彩画調レンダリング法” グラフィクスとCAD研究報告 2001
- 7) Paul Haerberli.: Paint By Numbers: Abstract Image Representations, pp. 477-484, 1996.
- 8) Cassidy J. Curtis, Sean E. Anderson, Joshua E. Seims, Kurt W. Fleischer, David H. Salesin: Compute Generated watercolor, proc.SIGGRAPH'97, pp.421-430, 1997
- 9) Wyvill B., Van Overveld K., Carpendale S.: Rendering cracks in batik. In NPAR '04: Proceedings of the 3rd international symposium on Nonphotorealistic animation and rendering (New York, NY, USA), ACM Press, pp. 61-149, 2004
- 10) 森本有紀, 小野謙二: 布形状を考慮した絞り染め模様のシミュレータ, Design シンポジウム 2010