

リジッドルームのための織物組織図の近似学習

吉田 哲也^{1,a)}

概要：経糸（たて糸）と緯糸（よこ糸）を絡みあわせて制作する布は織物と呼ばれ、糸の交差は織物組織図として表現される。綜統の数が少ない卓上織機であるリジッドルームを用いて複雑な模様を制作するためには緯糸ごとに経糸を手作業で拾う操作が必要となるが、拾う糸の数が規則的でない場合は間違いやすく負担が大きいという課題がある。リジッドルームで複雑な模様を制作する際の負担の軽減に向けて、本稿では経糸の拾い方の割り当てに基づく織物組織図の近似を提案する。織物の模様における組織図の敷き詰めと画像特徴量に基づいて組織図の差異を表現し、経糸の拾い方の逐次的な再割り当てを行う組織図生成アルゴリズムを提案する。

キーワード：リジッドルーム、手織り、織物組織図、近似

Approximate Learning of Weave Diagram for Rigid Loom

TETSUYA YOSHIDA^{1,a)}

Abstract: A textile is woven using warp and weft, and their relationship is represented as a weave diagram. Since the number of healds is small in a rigid loom, it is necessary to pick up warp for each weft by hand. This operation is rather cumbersome when the number of warp to be picked up is irregular. This paper proposes a method for approximating the specified weave diagram based on the assignment of warp pick up pattern. By representing the difference of weave diagrams based on their tiling and image features, a sequential re-assignment algorithm is proposed to generate an approximated weave diagram.

Keywords: rigid loom, hand weaving, weave diagram, approximation

1. はじめに

経糸（たて糸）と緯糸（よこ糸）を絡み合わせて作られた布は織物と呼ばれ、布を織るための装置は織機と呼ばれる [5], [7]. 製品として販売される織物は自動織機で製造されることが多いが、工房での制作や家庭で手芸を楽しむための手織り機も普及している。卓上織機のうちリジッドルームは比較的安価で家庭でも使いやすいが、綜統の数が少ないため制作可能な模様の種類に限られる。リジッドルームを用いて複雑な模様を制作することもできるが、緯糸ごとに経糸を手作業で拾う操作が必要となる [6]. しかし、拾う糸の数が規則的でない場合は間違いやすく負担が

大きいという課題がある。

リジッドルームを用いて複雑な模様を制作する際の負担の軽減に向けて、本稿では経糸の拾い方の割り当てに基づく織物組織図の近似を提案する。織物の模様における組織図の敷き詰めと画像特徴量に基づいて組織図の差異を表現し、経糸の拾い方の逐次的な再割り当てを行う組織図生成アルゴリズムを提案する。

2. 織物の制作

2.1 織り

織物は経糸に緯糸を通すことで作られ、基本の織り方である平織、綾織、縐子織は三原組織と呼ばれる [5], [7]. 平織の例を図 1、綾織の例を図 2 に示す。織りの構造を表す構造図で経糸と緯糸の交差点ごとにどちらの糸が上かを指定する必要があるため、織物の組織は織物組織図（以下、

¹ 奈良女子大学
Nara Womens's University, Nara, 630-8506, Japan
^{a)} tyoshida@cc.nara-wu.ac.jp

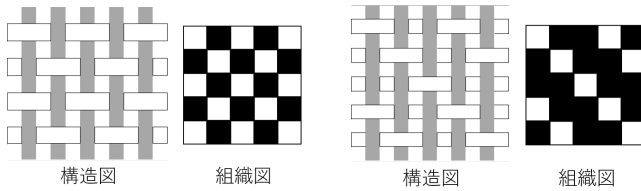


図 1 平織

Fig. 1 Plain weave

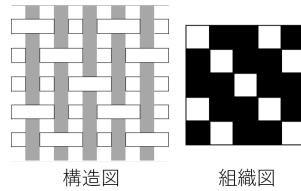


図 2 綾織

Fig. 2 Twill weave

組織図と呼ぶ)として表される。意匠紙のマス(方眼)の一目を交差点とし、経糸が上に現れる点を黒、その逆を白として表すことが多く、黒い点は組織点と呼ばれる。組織図には一般に規則的な繰り返しがあり、繰り返しの最小単位は完全組織と呼ばれる。以下では、組織図は織物の完全組織に対応することとする。

布を織る装置は織機と呼ばれ、製品として販売される織物を製造する織機にはタベット織機、ドビー織機、ジャガード織機などがあるが、本稿では工房や手芸などでの手織りに用いられる織機を扱う。図 3 にその模式図を示す。経糸を通す綜統が踏み木とつながっており、踏み木を踏むと綜統が開いて経糸が上下 2 層に分離し、その間に緯糸を通して織物を作る。

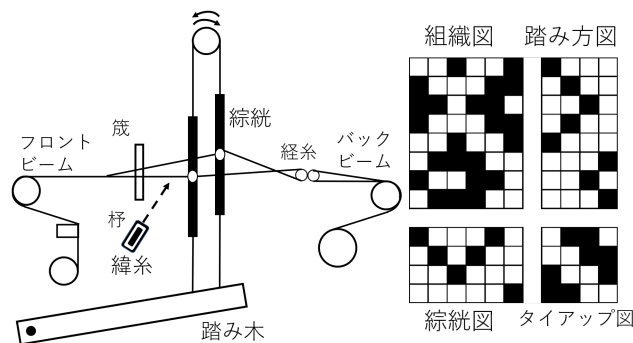


図 3 手織り織機の模式図

Fig. 3 Loom for hand weaving

図 4 織方図

Fig. 4 Structural draft

図 3 に示すように、織物の制作では 1) 踏み木の踏み方、2) 綜統と踏み木のつなげ方、3) 綜統への経糸の通し方、を指定する必要がある。これらを表す図の呼び方は文献によっても異なるが、本稿ではそれぞれ踏み方図、タイアップ図、綜統図と呼ぶ。また、組織図、踏み方図、タイアップ図、綜統図を図 4 のように配置して表すものを織方図と呼ぶ [2]。

2.2 織物組織図の行列表現

組織図の交差ごとに、経糸が上である組織点を 1 で表し、それ以外を 0 で表すことにより、組織図を行列を用いて表現できる [2]。同様に、織方図の他の図でもマス目ごとに黒と白が指定されるため、組織図と同様に黒の箇所を 1、それ以外を 0 とすることで行列として表現できる。

以下では、行列は太字の大文字、ベクトルは太字のイタリック小文字で表し、行列 $\mathbf{A}$ の第 ij 要素を a_{ij} 、ベクトル $\mathbf{a}$ の第 i 要素を a_i で表す。行列 $\mathbf{A}$ の転置を $\mathbf{A}^T$ 、ベクトル $\mathbf{a}$ の転置を $\mathbf{a}^T$ で表し、 $\mathbf{A}$ が正則行列のときはその逆行列を $\mathbf{A}^{-1}$ で表す。また、行列 $\mathbf{A}$ の第 i 行ベクトルを $\mathbf{a}_{i+}$ 、第 j 列ベクトルを $\mathbf{a}_{+j}$ で表す。

本稿では、組織図を行列 $\mathbf{W} \in \{0, 1\}^{m \times n}$ 、踏み方図を行列 $\mathbf{A} \in \{0, 1\}^{m \times p}$ 、タイアップ図を行列 $\mathbf{B} \in \{0, 1\}^{h \times p}$ 、綜統図を行列 $\mathbf{C} \in \{0, 1\}^{h \times n}$ と表す。このとき、組織図 $\mathbf{W}$ は行列演算を用いて

$$\mathbf{W} = \mathbf{A}\mathbf{B}^T\mathbf{C} \quad (1)$$

と表すことができる [2]。ただし、経糸 1 本がそれぞれ 1 枚の綜統に通る、同時に踏める踏み木は 1 本であるため、 $\mathbf{A}$ の行ごと、 $\mathbf{C}$ の列ごとに、それぞれただ一つの 1 を持つという制約を満たす必要がある。

3. 織物組織図の近似学習

3.1 リジッドルームを用いた手織り制作



図 5 リジッドルーム「咲きおり」

Fig. 5 Loom “Sakiori”

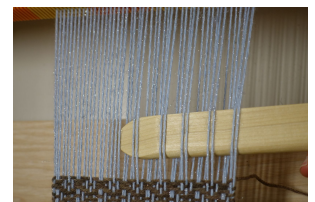


図 6 経糸を拾う操作

Fig. 6 Warp pick up

リジッドルームは比較的安価で家庭でも使いやすいが、綜統の数が少ないため制作可能な模様の種類が限られる。たとえばクロバー社の「咲きおり」(図 5 参照) [10] は綜統数が 2 枚の織機に対応するため、経糸と緯糸が 1 本ずつ交互に上となる平織りしか制作できない。リジッドルームを用いて複雑な模様の織物を作ることもできるが、ピックアップスティックを用いて経糸ごとに必要な緯糸を拾うことが必要となる(図 6 参照)。しかし、拾う糸の数が規則的でない場合は間違いが生じやすいという課題がある。そこで、リジッドルームを用いて複雑な模様の織物を作る際の負担の軽減に向けて、経糸の拾い方も考慮した組織図の近似を考える。

3.2 組織図の近似モデル

3.2.1 織りのデザインの枠組み

本稿の枠組みを図 7 に示す。まず、織物の模様をデザインする。その際、経糸と緯糸に対して異なる色を割り当て、経糸と緯糸の交差点ごとに織物で上に浮く色を指定する。経糸と緯糸に使用した色の情報をもとに、デザインした模

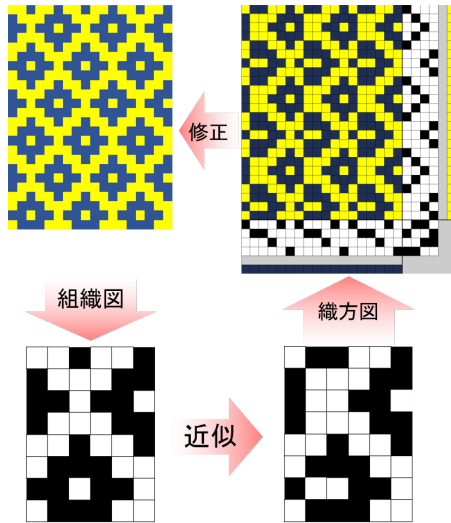


図 7 織りのデザインの枠組み

Fig. 7 Weave design framework

様を組織図 W_0 に変換し、3.3 節で述べる経糸のすくい方のバリエーションに基づいて組織図 W を生成する。生成した組織図に対応する模様とその織方図を確認し、デザインした模様と比較する。デザイン上の必要性があれば模様を修正し、このプロセスを繰り返す。

3.2.2 組織図の差異の表現

組織図を 0 と 1 を要素とする行列として表現するため、 W_0 と W の差異 ΔW を以下で定義する。

$$\Delta W = (W_0 + W) \bmod 2 \quad (2)$$

ただし、式 (2) で和 $+$ と $\bmod$ の演算は要素ごとに行う。式 (2) の ΔW は、 W_0 と W で経糸が上であるかが異なる交差点を 1、一致する交差点を 0 と表す。

式 (2) の ΔW は組織図の差異を表すが、織物の模様は組織図を繰り返したものであるため、 W_0 と W の違いだけではなく、組織図を規則的に繰り返して敷き詰めた模様としての差異を考慮する必要がある。組織図の敷き詰めはタイリングとみなすことができるため、ひとつの組織図を取り囲む範囲だけを考慮すれば十分である [9]。そこで、 ΔW を上下左右に 3 つずつ敷き詰めた分割行列 ΔW_t を、要素がすべて 1 である 3×3 行列 H とのクロネッカー積 $\otimes$ を用いて以下で生成する。

$$\Delta W_t = H \otimes \Delta W \quad (3)$$

$$= \begin{pmatrix} \Delta W & \Delta W & \Delta W \\ \Delta W & \Delta W & \Delta W \\ \Delta W & \Delta W & \Delta W \end{pmatrix} \quad (4)$$

式 (4) の ΔW_t は繰り返しも考慮した組織図の差異を表すが、織物の模様は 2 次元の拡がりをもつため、交差点に対応する個々の要素だけではなく、要素ごとにその近傍の情報も考慮して模様としての差異を表すことが重要となる。その際、等価組織と同様に平行移動に不変な表現であ

ることが望ましい。

本稿では、組織図を 2 値画像とみなし、画像における近傍情報を考慮できる高次局所自己相関特徴 [8] (以下では HLAC と呼ぶ) を式 (4) の ΔW_t に適用して組織図の差異を表現することを提案する。HLAC は画像に対する位置不変な特徴量であり、部分画像に対する加法性が成り立つことが知られている。そこで、 ΔW_t の要素ごとに、HLAC で定義される 3×3 のサイズの 25 個のマスクパターン (図 8 参照) とのマッチングを行い、パターンの生起頻度に基づいて組織図の差異を表現する。

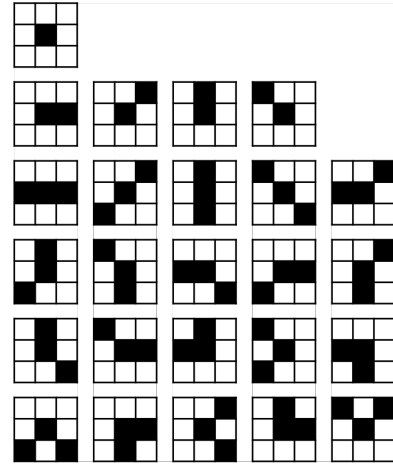


図 8 HLAC のマスクパターン

Fig. 8 HLAC mask pattern

3.3 リジッドルームのための組織図の近似

3.3.1 経糸の拾い方の割り当てを通じた組織図の生成

組織図は式 (1) のように行列演算を用いて表現できるが、これを以下のように書き換える。

$$\begin{aligned} W &= AB^T C \\ &= A(B^T C) \end{aligned} \quad (5)$$

式 (5) は、タイアップ図 B と綜統図 C を用いて行列 $(B^T C)$ を求め、2.2 節で述べた踏み方図 A における制約に基づき、組織図の各行ごとに $(B^T C)$ の行ベクトルを割り当てることができることを表す、とみなすことができる。

しかし、リジッドルームには踏み木はなく、また、複雑な模様を制作する際には綜統も用いない。そこで、式 (5) の考え方を一歩進めて、リジッドルームでの制作で用いる経糸の拾い方のバリエーションを考え、それを行列 $S \in \{0, 1\}^{s \times n}$ として表現し、組織図の各行に対して S における経糸の拾い方を割り当てることで組織図を生成することを考える。行列 S の行番号を経糸の拾い方の ID とみなし、組織図の各行への経糸の拾い方の割り当てを並べたベクトルを $a \in \mathbb{N}^m$ とすると、割り当てに従って S の行ベクトルをコピーすることで組織図 W を生成できる。この

```

Require:  $\mathbf{W}_0$ ; //組織図
Require:  $\mathbf{S}$ ; //経糸の拾い方

1: initialize  $\mathbf{a}$  //  $\{1, \dots, s\}$  から復元抽出
2: while not yet terminated do
3:   for each  $i \in \{1, \dots, m\}$  do
4:     式 (8) で  $a_i$  への割り当てを更新
5:   end for
6: end while
7: return  $\text{weave}(\mathbf{a}, \mathbf{S})$ 

```

図 9 経糸の拾い方の割り当て
Fig. 9 Warp pick up assignment

操作を

$$\mathbf{W} \leftarrow \text{weave}(\mathbf{a}, \mathbf{S}) \quad (6)$$

と表す。

行列 $\mathbf{S}$ の各行は経糸の拾い方に対応する 0 と 1 を並べたものとなるが、その並び方が不規則な場合には経糸の拾い間違いが生じやすくなる恐れがある。他方、規則的な並び方の場合には拾い間違いがおきにくいと期待される。どのような経糸の拾い方が望ましいかは制作者に依存するが、たとえばランレングス符号化のように同じ値の連続 (11 など) を単位と考え、これを交互に反転させた拾い方 (1100110011... など) は経糸の拾い方が規則的で間違いににくいと考えられる。

本稿では、指定された経糸の拾い方の集合のもとで組織図を近似する経糸の拾い方の割り当てを求めることを提案する。3.2.2 節で述べた組織図の差異の表現に対し、生起頻度の総和を用いて近似誤差を表す目的関数を以下で定義する。

$$J = \text{diff}_{HLAC}(\mathbf{W}_0, \mathbf{W}) \quad (7)$$

このため、式 (6) を用いて組織図 $\mathbf{W}$ を生成する際に式 (7) を最小化する割り当て $\mathbf{a}$ を求める。

3.3.2 アルゴリズム

組織図 $\mathbf{W} \in \{0, 1\}^{m \times n}$ の各行をデータ、 $\mathbf{S} \in \{0, 1\}^{s \times n}$ の各行をクラスとみなし、組織図の m 個の行を s 個の経糸の拾い方のどれかに逐次的に再割り当てすることを通じて式 (7) の最小化を実現する。具体的には、組織図の各行 $i \in \{1, \dots, m\}$ に対して

$$a_i = \arg \min_{q \in \{1, \dots, s\}} \text{diff}_{HLAC}(\mathbf{W}_0, \text{weave}(\mathbf{a}', \mathbf{S})) \quad (8)$$

$$s.t. \ a'_j = \begin{cases} q & j = i \\ a_j & \text{otherwise} \end{cases}$$

と式 (7) を最小化する経糸の拾い方を選ぶことで割り当て $\mathbf{a}$ を更新する。提案するアルゴリズムを図 9 に示す。1 行目で割り当て $\mathbf{a}$ を初期化し、4 行目で式 (8) に基づく再割り当てを行う。2 行目の終了条件は $\mathbf{a}$ の変更がなくなった

場合とする。なお、kmeans 法 [3] などのクラスタリング手法と異なり、すべてのクラス (経糸の拾い方) に必ず 1 つ以上のデータ (組織図の行番号) が割り当てられるとは限らず、生成する組織図で行列 $\mathbf{S}$ として表す経糸の拾い方がすべて使用されるとは限らない。

式 (8) に示す割り当てを逐次的に繰り返すことで、EM 法 [1] や kmeans 法と同様に目的関数の単調減少性を保証できる。さらに、HLAC の 3×3 のマスクパターンの形状から、式 (8) に示す割り当ての影響は図 9 の 4 行目での更新対象に対応する行とその前後の行を含めた範囲に限定できる。前後の行におけるパターンの生起頻度にも再割り当ての影響が及ぶため、動的計画法のように図 9 の 4 行目で着目する行とそれ以外の部分とに問題を完全に分割できるわけではないが、HLAC の性質と生起頻度の和に基づく $\text{diff}_{HLAC}(\cdot, \cdot)$ の加法性により目的関数の大域的最小化が実現できると予想される。しかし、アルゴリズムの性質を厳密に示すことは今後の課題である。

組織図 $\mathbf{W} \in \{0, 1\}^{m \times n}$ では各要素が 0 と 1 の 2 値を取り得るため、等価組織などを考慮せず素朴に扱えば $O(2^{mn})$ と指数的な組み合わせが考えられる。本稿のアプローチでは経糸の拾い方を表す行列 $\mathbf{S} \in \{0, 1\}^{s \times n}$ における各行の織り方に限定するものの、やはり $O(s^m)$ と指数的な組み合わせとなる。しかし、上記で述べたように HLAC のマスクパターンとのマッチングは更新対象の行とその前後の行だけに局所化できる。パターンの数は定数であるため、式 (8) の $\text{diff}_{HLAC}(\cdot, \cdot)$ は組織図の行数 m 、列数 n に対して $O(n)$ で計算でき、これを $\ell \in \{1, \dots, s\}$ の s 回行う。このため、入力 ($\mathbf{W}_0$ と $\mathbf{S}$) と初期値に依存する while の繰り返し数を無視すれば、図 9 のアルゴリズムの計算量は $O(mns)$ と多項式のオーダーとなる。

3.4 組織図に対する織方図の生成

与えられた組織図 $\mathbf{W}$ に対し、織方図を構成する踏み方図 $\mathbf{A}$ 、タイアップ図 $\mathbf{B}$ 、綜統図 $\mathbf{C}$ を以下のように生成する。

まず、 $\mathbf{W}$ で相異なる行ベクトルを列挙し、それらを $\{\mathbf{r}_1^T, \dots, \mathbf{r}_p^T\}$ とする。次に、要素がすべて 0 の $m \times p$ 行列 $\mathbf{A}$ を準備する。最後に、 $\mathbf{W}$ の各行ベクトル $\mathbf{w}_{i+}^T$ ごとに、列挙した $\{\mathbf{r}_1^T, \dots, \mathbf{r}_p^T\}$ と一致するベクトルの番号を k とするとき、 $\mathbf{A}_{ik} = 1$ とする。

同様に、 $\mathbf{W}$ で相異なる列ベクトルを列挙し、それらを $\{\mathbf{c}_1, \dots, \mathbf{c}_h\}$ とする。次に、要素がすべて 0 の $h \times n$ 行列 $\mathbf{C}$ を準備する。最後に、 $\mathbf{W}$ の各列ベクトル $\mathbf{w}_{+j}$ ごとに、列挙した $\{\mathbf{c}_1, \dots, \mathbf{c}_h\}$ と一致するベクトルの番号を ℓ とするとき、 $\mathbf{C}_{\ell j} = 1$ とする。

最後に、行列 $\mathbf{B}$ を以下で定める。

$$\mathbf{B} = \{(\mathbf{A}^T \mathbf{A})^{-1} (\mathbf{A}^T \mathbf{W} \mathbf{C}^T) (\mathbf{C}^T \mathbf{C})^{-1}\}^T \quad (9)$$

