

コンピュータによる浴衣の仕上り画像の作成

田中 早苗

(平成 14 年 10 月 3 日受理)

Computer generated Yukata Figures

TANAKA, Sanae

(Received on October 3, 2002)

キーワード：浴衣，柄合わせ，相談型システム，雛型，ウェブ教材

Key words：yukata, pattern matching, consultation system, final figure, web teaching materials

1. はじめに

浴衣は和裁初心者に適した教材であるが，用いられる反物の多くが柄合わせの作業を要する．学生が実習等で浴衣を製作する場合，各自が持ち寄った反物で柄合わせを行うが，柄合わせ学習に充当する時間が限られるため様々な柄の柄合わせを体験することは難しい．

本研究ではパソコンのモニタ上に幾通りもの柄合わせ状態を提示して，柄合わせ学習を支援するシステムの構築を目指した．

前報¹⁾で述べた柄合わせ方法のアルゴリズムは，着丈寸法，裁断順序，柄の向き，一型の長さとは柄までの距離等をパラメータとし，文献から抽出した柄合わせの要点から柄配置・非柄配置の位置を領域分けして，柄配置に該当した柄が最も多い裁断順序を提示した．

本報では和裁熟練者や一般の和裁授業においても行われる‘背中心で左右身頃の柄を交互に配置する’方法を取り入れ，これによって生成される柄配置のシミュレーション画像を作成する方法について報告する．

2. アルゴリズム

伝統的な浴衣地の柄の配列は，反物の 1 型が丈方向に鏡像で繰り返されている．図 1 (1) に示すように，1 型の画像ファイル（以後パターン）の緯方向の鏡像をストロークとして作成し，ストローク画像の左端をベース

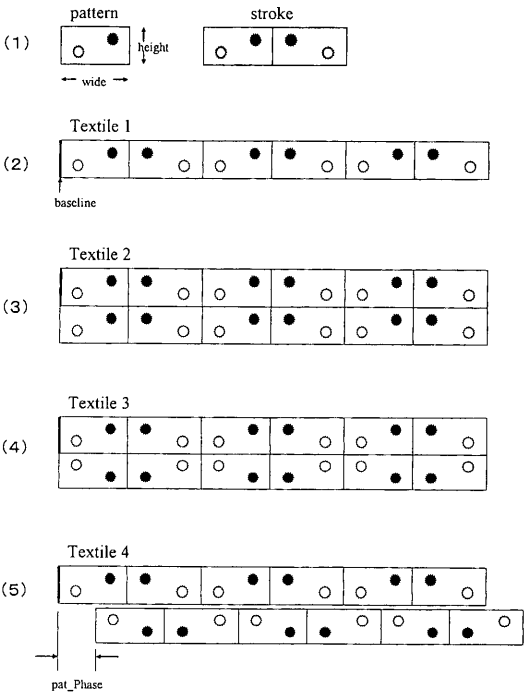


図 1 テキスタイル画像の作成

インにして繰り返し画像を作成すると図 1 (2) のような Textile1 が得られる．Textile1 をコピーして下にベースとした状態が図 1 (3) の Textile 2 である．これを着物に置き換えると，左後身頃の裾を起点として反物を延反し，延反先で折り返して布端が再び起点に戻ってきたときに，布の表裏を逆さにした状態である．Textile 3 は

表裏を変えずに折り返した状態で、これはTextile 1の縦方向の鏡像である。柄合わせでは、柄の種類によっていずれの方法も用いられる。

このように並行させた反物の柄は、後中心を挟んで同種の柄や大きな柄が隣接しないことが好ましく、また裾から70～80cmの後腰位置には大柄が並ばないことが望ましい。これらの要領を踏まえて並行した反物の柄を交互に配置させるために、片方の反物を丈方向に位相を加える(図1(4)Textile 4)。浴衣の柄は位相がパターン長の1/2のときに等間隔な配置となる。しかし、柄の種類によっては柄が変則に配置されていたほうが柄の流れがリズムカルな場合もあるので位相は任意に決められるようにした。

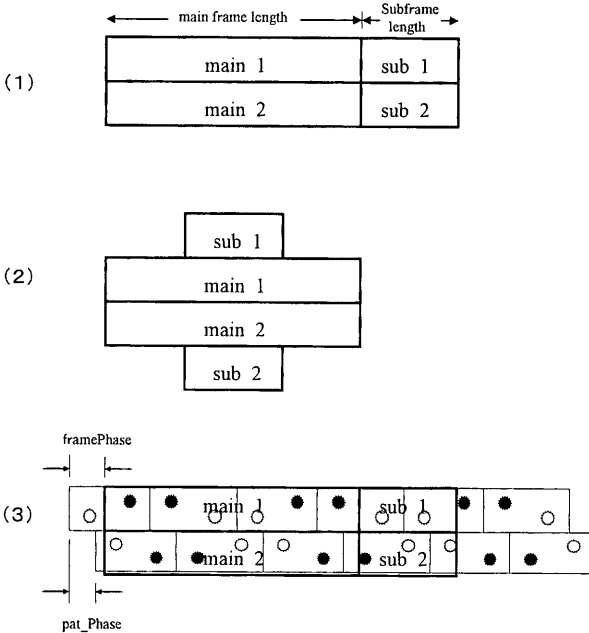


図2 雛型フレームの作成

着物は前後の身頃と袖が肩山でつながった状態で裁断される。したがって、図2(1)のように身丈の2倍をメインフレーム長さとし、袖丈の2倍をサブフレーム長さとし、フレームのmain1は左身頃、main2は右身頃、sub1、sub2はそれぞれ左前後袖、右前後袖として図2(2)のような雛型を形成した。Textile2またはTextile3からこのフレームで切り出す位置は、ベースラインからの位相によって決める。これをフレーム位相とした。柄合わせにおいて切り出し位置となる左裾は大きな柄が途切れないことが望ましいので、フレーム位相は数値入力

か、あるいは画像にクリックボタンとなる領域を設定して裾の位置を決められるようにした。

3. システム構成

システムはネットワークにつながったパーソナルコンピュータで運用できることを前提として作成した。容量の嵩む画像ファイルを扱うために、LANにWebサーバを立ち上げて実行スクリプトファイルを置いた。サーバソフトウェアはApacheでTurbolinux 7.0 Servrue上で利用している。フレーム作成等の計算処理のためにPerlによるCGI(Common Gateway Interface)プログラムを有する。実行スクリプトはRubyで作成し、画像編集にはImage Magicのライブラリを用いた。一型画像はデジタルカメラで撮影し、1/20のサイズに縮小して30～40KBのPNG(Portable Network graphics)形式のファイルを作成した。図3にTextile 2画像を作成するスクリプトを例示した。はじめに個々の画像ファイルのm当たりのpixel数をlunitとして各フレームのunitをpixel単位に直している。これによってcm単位で入力した身丈、袖丈等と画像サイズの整合が図られる。画像ファイルのm当たりのpixel数は約195～200pixelsであった。図4に身丈160cm、袖丈55cmでフレーム位相を0cmで変化させずにパターン位相のみを0、20、40cmに変化させた場合の画像を示した。

4. インタフェースと出力結果

HTMLで作成したユーザインタフェースの初めのステップで、画像ファイルを選択する。この度はサーバにストックした一型画像をそのままのサイズで表示し、画像データベースのように提示するためにスクロールバーを設けた。選択した画像をリンクボタンとしてステップ2を表示させる。ステップ2では身丈、袖丈を入力し、「柄を何cmずらしますか?」の問いでパターン位相を5cm刻みのプルダウンメニューから選択する。次に左後身頃の裾にする位置の決め方は、画像のイメージマップをリンクボタンとしてフレーム位相を決定する。また表示した一型画像の上下を逆さまに表示する機能を設けて柄の向きを変えることができる。図5にWebにおける出力結果を示した。雛型画像は複数表示することが可能であり、入力数値を変えることによって柄の配置状態を比較することができる。

```

if ( ARGV.size() < 6 || ARGV.size > 7 ) # 引数チェック
  Usage()
  exit
end

begin
  patfn = ARGV[0]      # パターン画像ファイル名    [pattern filename]
  outfn = ARGV[1]      # 出力画像ファイル名        [output filename]
  mfl_u = ARGV[2].to_f() # メインフレーム長 (unit) [main frame length by unit]
  sfl_u = ARGV[3].to_f() # サブフレーム長 (unit) [sub frame length by unit]
  pph_u = ARGV[4].to_f() # パターン位相 (unit) [pattern phase by unit]
  fph_u = ARGV[5].to_f() # フレーム位相 (unit) [frame phase by unit]
  ppu = ARGV[6]        # 長さ単位 (pixel/unit)
  if ( ppu == nil )    # [pixels per unit]
    ppu = 1.0
  else
    ppu = ppu.to_f()
  end

  raise "File #{patfn} not found." if ( ! FileTest.exists?( patfn ) );
  raise "Can't slide to left direction(pph<0)" if ( pph_u<0);
  raise "Can't slide to left direction(fph<0)" if ( fph_u<0);

rescue
  print $!, "\n"
  exit
end

# まず、 unit を pixel 単位に直す
mfl_px = (mfl_u * ppu).to_i() # メインフレーム長
sfl_px = (sfl_u * ppu).to_i() # サブフレーム長
pph_px = (pph_u * ppu).to_i() # パターン位相
fph_px = (fph_u * ppu).to_i() # フレーム位相
cfl_px = mfl_px + sfl_px      # カットフレーム長 ( メインフレーム+サブフレーム )

tmp_files = []
# パターン画像の情報を抽出
width_px, height_px = getImageSize( patfn )
# print "#{width_px},#{height_px}\n"

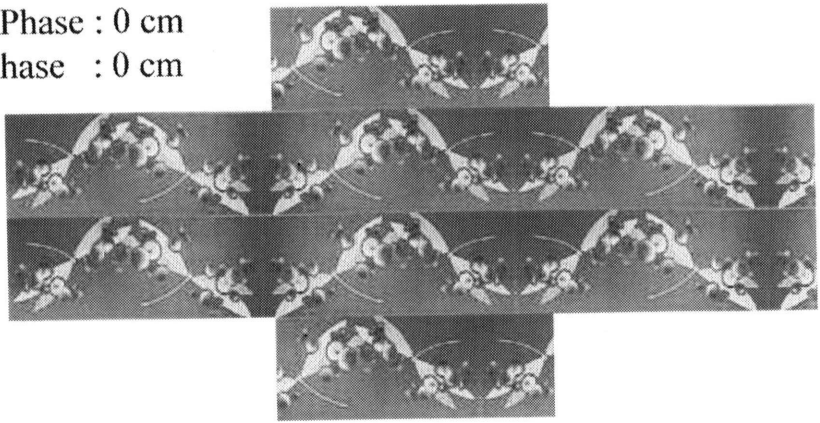
# パターンストローク画像を生成
stroke_img = makePatStrokeFile( patfn )
tmp_files.push(stroke_img)

# textile1 を作成。ストローク画像を繰り返して作成する
ite = cfl_px / (width_px*2) + 1 # ストロークの繰り返し回数を計算
textile1 = makeTextile1File( stroke_img, ite )
tmp_files.push(textile1)

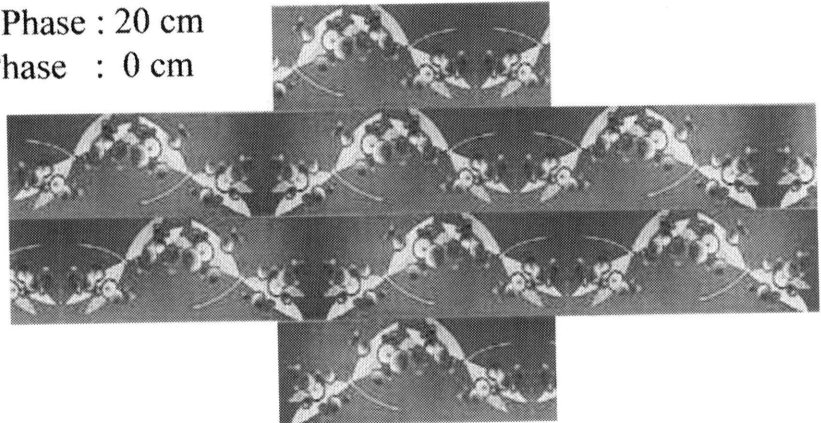
# textile2 を作成 ( 1 をずらしたものを作る )
textile2 = makeTextile2File( textile1, pph_px )
tmp_files.push(textile2)

```

- (1) pattern Phase : 0 cm
frame Phase : 0 cm



- (2) pattern Phase : 20 cm
frame Phase : 0 cm



- (3) pattern Phase : 40 cm
frame Phase : 0 cm

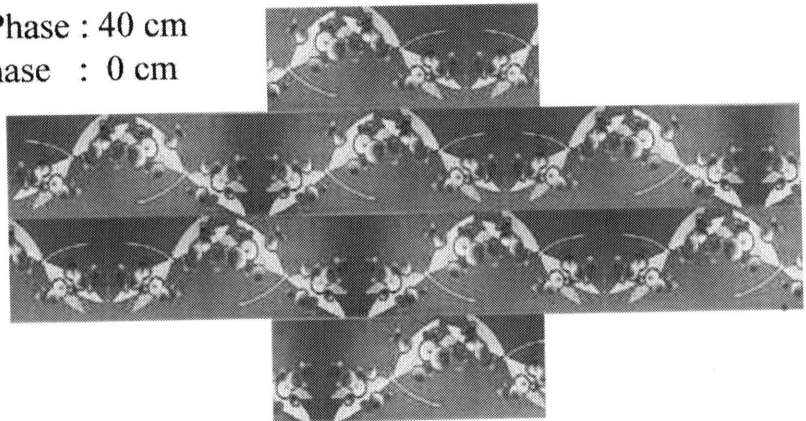


図3 パターン位相のみ変化させた画像出力例



図5 Webにおける画像表示例

5. まとめと今後の課題

一型画像の鏡像から繰り返し画像を作成し、これを並行に並べて位相を加えることにより、背中心で柄を交互に配置させるという柄合わせの手法で雛型を編集することができた。位相を少しずつ変えることによって編集された雛型の柄の配置や流れを確認できることは、和裁学習者に限らずゲーム感覚で試行できる。

今後この雛型に付け加えられるべき機能は、衿の表示である。衿は袖を切り出した部分に続く画像を縦に2分割、緯に2分割した合計8画像のいずれか1つを上前身頃に配置する。柄合わせの実際においては、布の上下を逆さにしたり、表裏を逆にする等して上前に最も適切な柄のパーツを選ぶ。これをコンピュータで行うとき、衿パーツをドラッグ・アンド・ドロップで移動させて、ユーザー自身が適切な柄合わせを検討することが理想的であ

る。また、前報のように柄のある位置を算出して特定のパーツのみ表示させること等も考えられる。

この論文は日本家政学会第54回大会において発表した。

参考文献

- 1) 田中早苗：学校教育における浴衣製作のための柄合わせ相談システム，東京家政大学研究紀要41 (2)，103-110 (2001)

Summary

The purpose of this paper is to construct a consultation system of pattern making for Yukata making in school education. In the previous paper, the author reported an algorithm of cutting procedure giving a largest number of cloth figures at an adequate position of Yukata. The present paper reports the simulation method of making images to arrange the cloth figures in desired regular manners, adopted often in a Yukata class and by experts, on the right and left sides of the back of Yukata. This system is constructed as a Web teaching material on the WWW server in HTML form. For server software, Apache is used on Turbolinux. The common gate way interface is written in Perl and the execution script is written in Ruby. The simulation images are developed by using the library of Imagemagic.