ALGORITHMS FOR COLOR ANALYSIS

J. A. Hoskins, W. D. Hoskins and J. L. W. May

Dept. of Computer Science, University of Manitoba

Winnipeg, Manitoba, Canada

Abstract

The usual algorithms for the analysis and synthesis of woven textile structures assume, in general, that the interlacement data is binary, and that this data corresponds on a one-to-one basis with the visible pattern exhibited by a cloth constructed of just two colors. As soon as this two-valued restriction is lifted and the data array is permitted to contain an arbitrary number of colors, the analysis algorithm becomes considerably more complicated. Three classes of solutions now become possible. The form of this algorithm, the solutions, and the necessary implications for the architecture of the user interface are examined and an implementation is discussed.

1. <u>Introduction</u>

The production of shuttle-woven fabrics involves the interlacement of one set of yarns, called the weft picks, with an orthogonal set of yarns, called the warp ends. Numerous descriptions of this process within a computer environment exist, however, [7], [10], [11], [12], [13], [14] and address the algorithms and problems arising when the visible pattern (see figure 1) is congruent to the <u>point diagram</u> of interlacements (see figure 2). Reference [7] also develops processes for the display of fabrics which are not formed in a single layer, but which are composed of two layers. These layers can either be disjoint or can exhibit some partial interchange. Further, more complex, forms, with up to four layers are examined in [6]. The size of the data arrays can be of an impressive size, and it is not uncommon for fine silk fabrics to contain 100 threads to the lineal inch, to be perhaps 30 inches wide, and to have a repeat size of 30 inches in the lengthwise direction. The implied data array is thus 3000 by 3000, or 9×10^6 bits, for the interlacement array. If a color representation, with 16 colors, is required then 144×10^6 are needed instead.

This paper discusses the form, and implementation, of the textile system needed to support the development of the color factoring algorithms. Appropriate examples and illustrations are provided.

2. <u>Data Entry Requirements</u>

In [7], the general aspects of data entry for the binary problem are outlined; however, in the color entry case, the potential size of the data structures means that it is no longer possible for the textile design environment to store more than a small number of these arrays on a reasonably sized hard disk. It is therefore necessary to devise some method of limiting storage.

We now introduce the following terms. A <u>k-color array</u> W is an nxm rectangular array of integers, drawn from the set **S** = {1,2,3,----,k}. The <u>k-color vectors</u> V and U are two vectors with entries drawn from the set **S**. The V vector has n elements and is designated the <u>weft color vector</u>, while the U vector has m elements and is designated the <u>warp color vector</u>. There are, therefore, two color modes to accommodate in the examination of a physical textile; one mode corresponding to synthesis of the

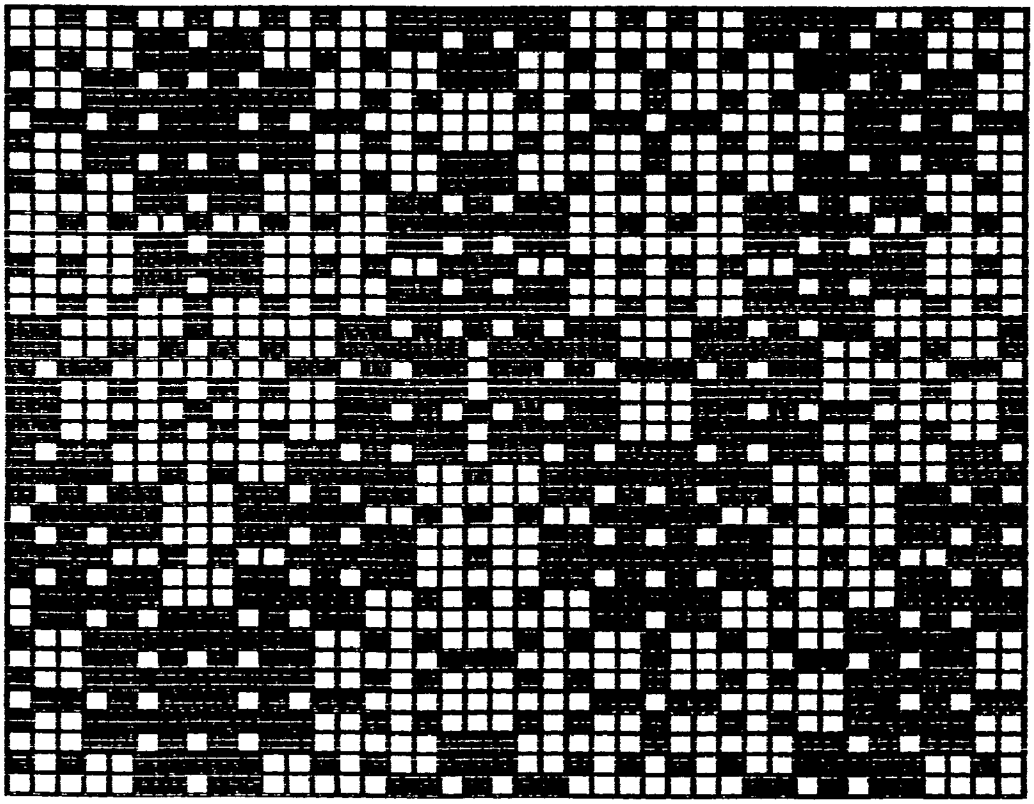


Figure 1

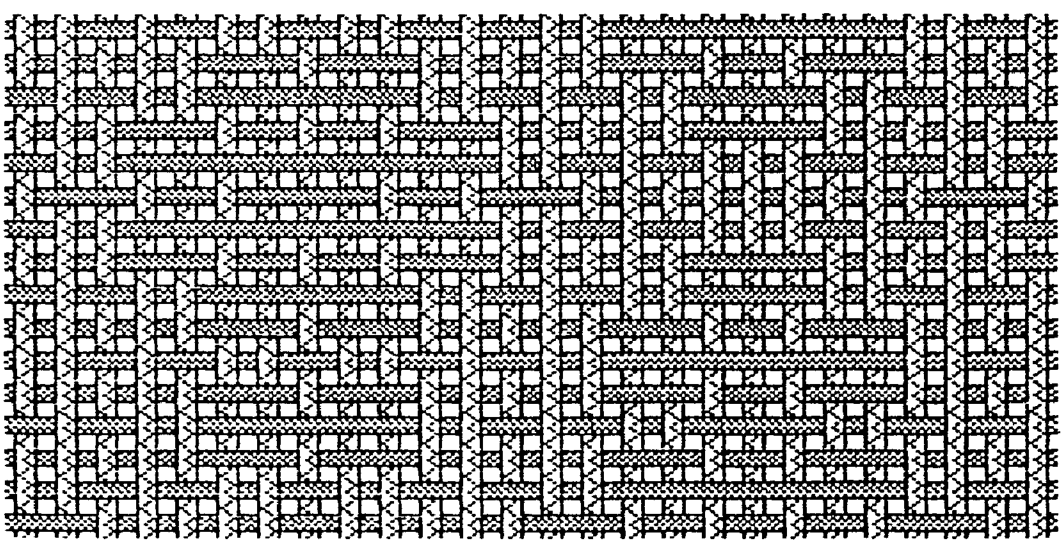


Figure 2

color array from its color factors. The other mode is the input of a given color array, for later analysis into its subordinate color factors and point diagram.

First mode

In this case, the underlying structure is known a priori, and the fast factoring algorithm described in [8], or the later improved version of [9], is used to factor the given interlacement array D (see figure 3) into the three required loom factors, namely the threading (A), tie-up (B) and shed sequence (C) (illustrated in figure 4). Up to this point, it is tacitly assumed that the warp/weft colors are, for example, black/white. The interlacement array D echoes this fact, by containing a one to represent an over crossing and a zero to represent an under crossing (the design and interlacement are congruent). The display for entry or editing of this information can therefore correspond, essentially, to a scaled one-one multiple-window mapping of the data structure(s). However, as soon as data entry to the warp and weft color vectors U, V is enabled, the display which reflects the present state of the subordinate color array W, becomes, with every change, less and less visually congruent to the original interlacement array D. It also becomes a less amenable design environment for the obvious and intuitive manipulation by the textile designer, towards a purposeful target. One clear advantage of this synthesis-only approach is that, at all times, a completely determined form for the textile is available. The interlacement array D is known; the loom set up factors A, B and C are known; the warp and weft color vectors U and V are known; and the colored array W can be determined by the process described in section 3.

Another important, but less obvious problem, is to devise an appropriate, and functionally meaningful, method of binding the input device to the color selection process at the time that the two color vectors are constructed. Since single character abbreviations are often awkward to construct (for example, for grey and green), and some electronic colors are not reliably identifiable by untrained colorists, an on-screen palette, cued with color to keys, has been used.

Sixteen colors are permitted in this particular implementation, in two different palettes, with two intensities, and with the maximum size for the array D being set to 512 x 512. The color array thus contains 512 x 512 x 4 bits (bytes rather than four bits have been used, to allow for future expansion). Storage requirements for this form involve filing the interlacement array D, the arrays A, B, C, and possibly the vectors U, V, for each named design.

Second Mode

In this mode, no input to the color vectors is appropriate, since the user is only concerned with constructing a color array which reflects the required design in the given palette of colors and intensities. Functionally, the input requirement has been to provide a windowed, scrollable display. In this case, the appropriate binding of the input device to the colors is of even greater importance than in the first mode, since data entry now involves painting the design, either broadly, or at the individual pixel level, with correspondingly heavier input requirements.

The storage requirements for this method of design development, require that the color array be maintained in memory and, for reasons of space, only one unfactored colored array is permitted to be maintained in the permanent filing module of the system, at any one time. This allows the textile designer who is working with a draft in development, to store at least one colored array which has not been analysed, or which can not be analysed, within the constraints enforced by the algorithm. Storage of color-analysed designs has been arbitrarily limited to 80 designs.

3. Computational Algorithms

The problems created by the first data entry mode are unfortunate, but it can be seen from figures 5, 6, 7 that little similarity between these figures is identifiable to the casual observer. However, the structure of these "cloths" is the same and only the color vectors differ.

Specifically, the process for determining an entry $w_{i,j}$ in the colored array W, given the color vectors $U = \{ u_k \}$, $V = \{ v_r \}$ and the interlacement array $D = \{ d_{s,t} \}$, is given most efficiently by the equation:

$$w_{i,j} = d_{i,j}\, u_i + (\sim d_{i,j})\, v_j$$

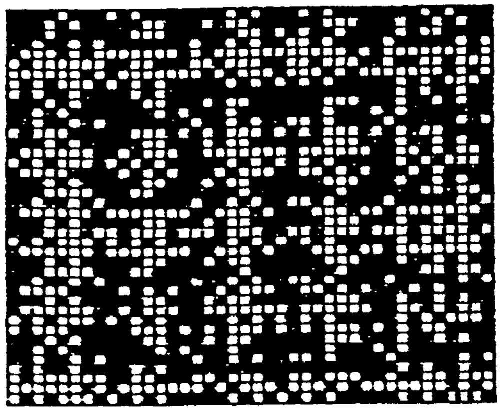


Figure 3

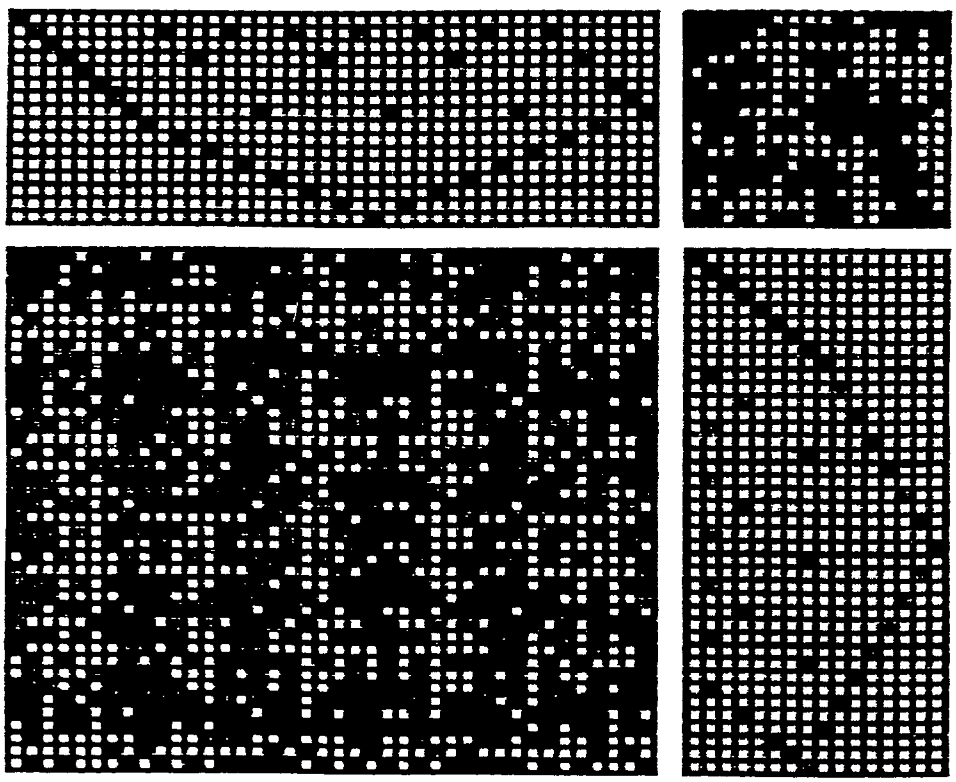


Figure 4

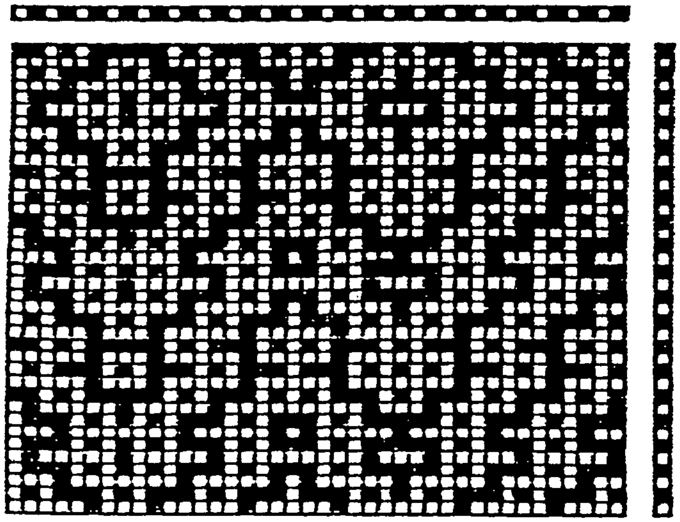


Figure 5

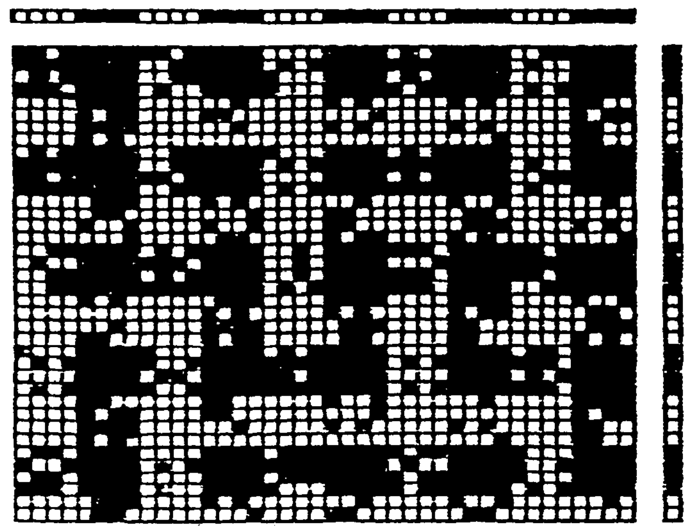


Figure 6

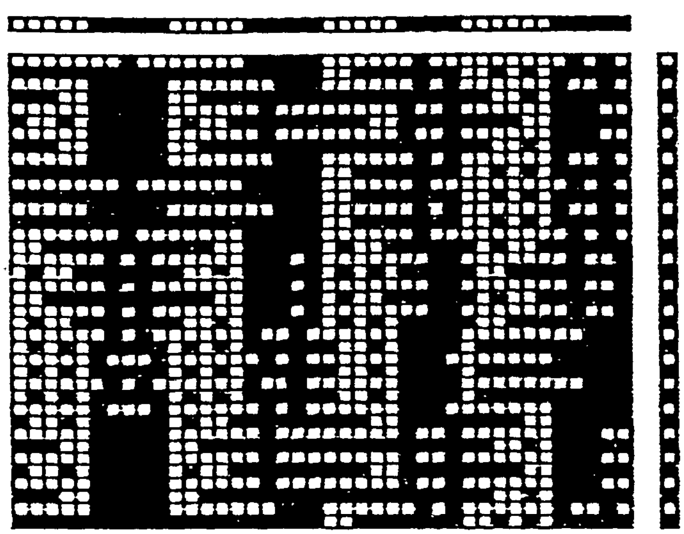


Figure 7

where $i = 1, 2, 3, \cdots, n$; and $j = 1, 2, 3, \cdots, m$ and $\sim$ is the usual logical operator "not" operating on boolean values represented by ones and zeroes.

A more compact representation of the above relationship is given by the matrix equation

$$W = E\,D + (\,D - uu^T\,)\,F$$

where E is a diagonal matrix with the elements of U down the diagonal, F is a diagonal matrix with the elements of V down the diagonal and u is a column vector, of appropriate dimension, containing a one in every position. D and W are the interlacement and color arrays, respectively.

The reverse process, namely the factoring of a color array, which is perhaps even a digitized image, into color factors, and then into the structural factors required for physical construction of a final fabric, is not as straightforward as the synthesis, and may not yield the "essential uniqueness" characteristic of the usual binary interlacement analysis [4]. Moreover, the admission of color to the analysis process means that the final solution to the process may be an indeterminate t-solution (that is, where the image only factors into a tapestry solution or weaker forms of it, and the color vectors U, V are matrices). The structure of the obverse fabric(s) will, in this case, remain unknown and is, of course, arbitrary.

The primary goal of the color analysis algorithm is therefore to factor a given color array into two unique color vectors, the U and V vectors previously defined, and a unique interlacement array. Permutations of any of the factors which satisfy the interlacement and design conditions are considered equivalent to the original factor, or factors, and correspond to the essential uniqueness discussed in [8].

The first stage of this analysis involves determining a vector of warp colors U, a vector of weft colors V, and the binary interlacement array $D = \{\, d_{i,j} \,\}$, corresponding to a given color interlacement array $W = \{\, w_{i,j} \,\}$. Implicit in this description is the verification that the array W is, indeed, a simple color interlacement array.

The following finite series of steps will produce the required vectors of colors (encoded as integers):

1. Set $U_j = w_{1,j}$ for all columns where $w_{1,j} = w_{i,j}$, for all i.
2. Set $V_i = w_{i,1}$ for all rows where $w_{i,1} = w_{i,j}$, for all i.
3. Terminate the process if the U and V vectors are complete.
4. Choose a value for j ($j = 1,2,\ldots,m$) such that there exists some $w_{i,j} \neq w_{1,j}$ ($i = 1,2,\ldots,n$).
5. Let $w_{1,j}$ be a warp over weft intersection and set $U_j = w_{1,j}$.
6. All $w_{i,j} \neq U_j$ are weft over warp intersections. Set $V_i = w_{i,j}$ for all of these values of i.
7. For all V_i determined in step 6, locate all $w_{i,j} \neq V_i$ and set U_j equal to these $w_{i,j}$.
8. Repeat steps 6 and 7 until all values for U and V have been defined.

N.B. If, at any point in this process, an inconsistency develops, the array W <u>is not</u> a simple colored interlacement array.

Obviously, as far as the color analysis is concerned, any rectangular region in a colored interlacement array corresponding to intersections between warp and weft yarns of the same color is structurally indeterminate.

<u>EXAMPLE 1</u>

This example shows a colored interlacement array W with two colors (encoded as 2 and 3) and its corresponding U and V color vectors. The same array is also shown with its rows and columns permuted, with corresponding changes being made to the color vectors. It is clear that, although the color <u>sequences</u> have changed, the interlacement relationship between <u>individual</u> warp and weft strands has not.

$$
W = \begin{array}{cccccccccc}
3&3&2&2&3&3&3&3&3&3\\
3&3&2&3&3&3&3&2&3&3\\
3&3&3&3&3&3&3&3&3&3\\
3&3&3&2&3&3&2&3&3&3\\
2&2&2&2&3&2&2&2&3&3\\
2&3&2&2&3&3&2&2&3&2\\
3&3&2&2&2&3&2&2&2&2\\
3&3&3&2&3&3&2&3&3&3\\
2&3&2&2&2&3&2&2&2&2\\
2&3&2&2&3&3&2&2&3&2\\
2&2&2&2&2&3&2&2&2&3
\end{array}
\qquad
V = \begin{array}{c}
3\\3\\3\\3\\2\\2\\2\\3\\2\\2\\2
\end{array}
$$

$$U = \begin{array}{cccccccccc} 3&3&2&2&3&3&2&2&3&3 \end{array}$$

(see figure 8)

$$W^P = \begin{array}{c} 3\,3\,3\,3\,3\,3\,2\,2\,3\,3 \\ 3\,3\,3\,3\,3\,3\,2\,3\,3\,2 \\ 3\,3\,3\,3\,3\,3\,3\,3\,3\,3 \\ 3\,3\,3\,3\,3\,3\,3\,2\,2\,3 \\ 3\,3\,3\,3\,3\,3\,3\,2\,2\,3 \\ 2\,2\,3\,2\,3\,3\,2\,2\,2\,2 \\ 2\,3\,3\,3\,3\,2\,2\,2\,2\,2 \\ 3\,3\,2\,3\,2\,2\,2\,2\,2\,2 \\ 2\,3\,2\,3\,2\,2\,2\,2\,2\,2 \\ 2\,3\,3\,3\,3\,2\,2\,2\,2\,2 \\ 2\,2\,2\,3\,2\,3\,2\,2\,2\,2 \end{array} \qquad V^P = \begin{array}{c} 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \end{array}$$

$$U^P = 3\,3\,3\,3\,3\,3\,2\,2\,2\,2$$

(see figure 9)

Based on the preceding results, we now have an algorithm for factoring colored interlacement arrays, as follows:

1. Obtain the U and V color vectors, using the process outlined.
2. If the sets of U and V colors are disjoint, set elements in the interlacement array corresponding to U colors equal to 1 and V color elements equal to 0. Go to step 4.
3. If the sets of U and V colors are not disjoint, partition the interlacement array into regions which are color disjoint and regions which are not color disjoint. Assign 1's and 0's in disjoint regions, as in step 2. Areas which are undefined are structurally indeterminate and intersections can be freely specified according to whatever criteria you choose (for example, to correct reducibility using the algorithm elaborated in [2] improved by the depth first search described in[1]). Assign these values.
4. Determine the corresponding threading, tie-up and shed sequence matrices for the resulting binary interlacement array, using one of the algorithms from [7], [9].

EXAMPLE 2

This example shows a colored interlacement array W, with disjoint warp and weft coloring, along with its corresponding binary interlacement array D.

$$W = \begin{array}{c} 5\,2\,5\,2\,3\,3\,3\,5 \\ 5\,2\,5\,2\,2\,4\,4\,4 \\ 3\,2\,5\,2\,2\,5\,3\,3 \\ 4\,4\,5\,2\,2\,5\,2\,4 \\ 3\,3\,3\,2\,2\,5\,2\,5 \\ 5\,4\,4\,4\,2\,5\,2\,5 \\ 5\,2\,3\,3\,3\,5\,2\,5 \\ 5\,2\,5\,4\,4\,4\,2\,5 \end{array} \qquad V = \begin{array}{c} 3 \\ 4 \\ 3 \\ 4 \\ 3 \\ 4 \\ 3 \\ 4 \end{array}$$

$$U = 5\,2\,5\,2\,2\,5\,2\,5$$

(see figure 10)

$$D = \begin{array}{c} 1\,1\,1\,1\,0\,0\,0\,1 \\ 1\,1\,1\,1\,1\,0\,0\,0 \\ 0\,1\,1\,1\,1\,1\,0\,0 \\ 0\,0\,1\,1\,1\,1\,1\,0 \\ 0\,0\,0\,1\,1\,1\,1\,1 \\ 1\,0\,0\,0\,1\,1\,1\,1 \\ 1\,1\,0\,0\,0\,1\,1\,1 \\ 1\,1\,1\,0\,0\,0\,1\,1 \end{array} \qquad V = \begin{array}{c} 3 \\ 4 \\ 3 \\ 4 \\ 3 \\ 4 \\ 3 \\ 4 \end{array}$$

$$U = 5\,2\,5\,2\,2\,5\,2\,5$$

(see figure 11)

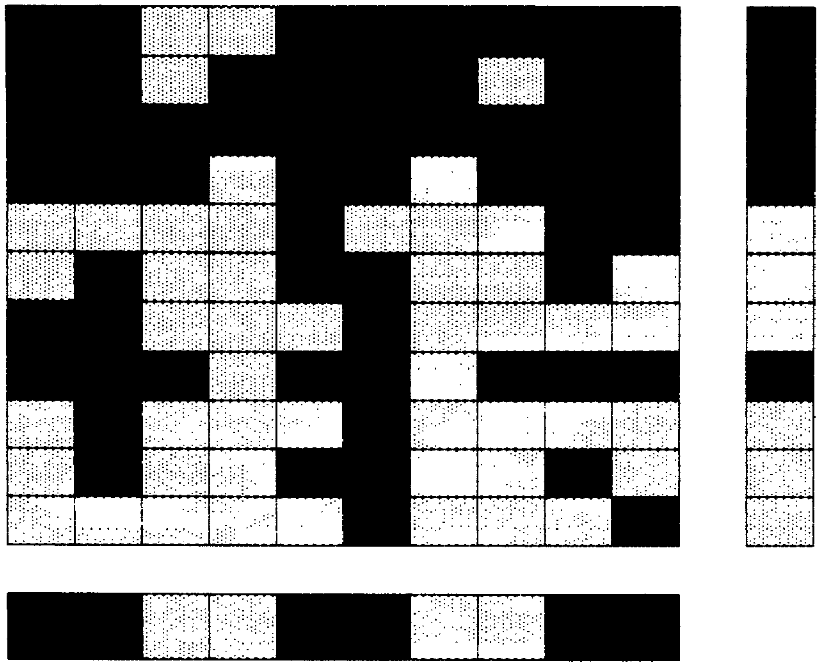
EXAMPLE 3

This example shows a colored interlacement array W with U and V colors which are not disjoint, along with the corresponding partially determinate binary interlacement array D. The array D^P is D, with its rows and columns permuted to show the indeterminate intersections in rectangular regions.

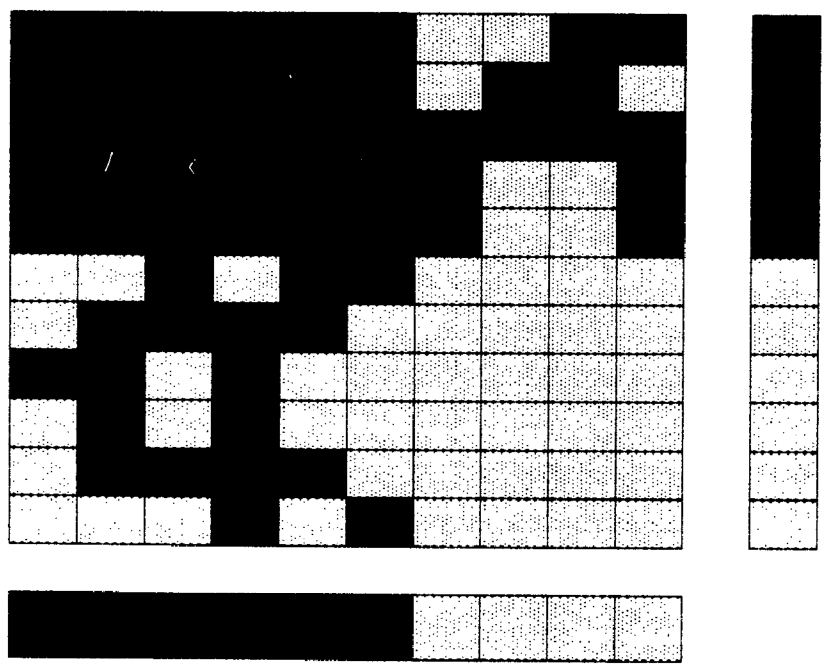
$$W = \begin{array}{c} 4\,2\,4\,3\,2\,2\,2\,3 \\ 4\,2\,4\,3\,4\,4\,4\,4 \\ 2\,2\,4\,3\,4\,2\,2\,2 \\ 3\,3\,4\,3\,4\,2\,4\,3 \\ 2\,2\,2\,3\,4\,2\,4\,3 \\ 4\,4\,4\,4\,4\,2\,4\,3 \\ 4\,2\,2\,2\,2\,2\,4\,3 \\ 4\,2\,3\,3\,3\,3\,4\,3 \end{array} \qquad V = \begin{array}{c} 2 \\ 4 \\ 2 \\ 3 \\ 2 \\ 4 \\ 2 \\ 3 \end{array}$$

$$U = 4\,2\,4\,3\,4\,2\,4\,3$$

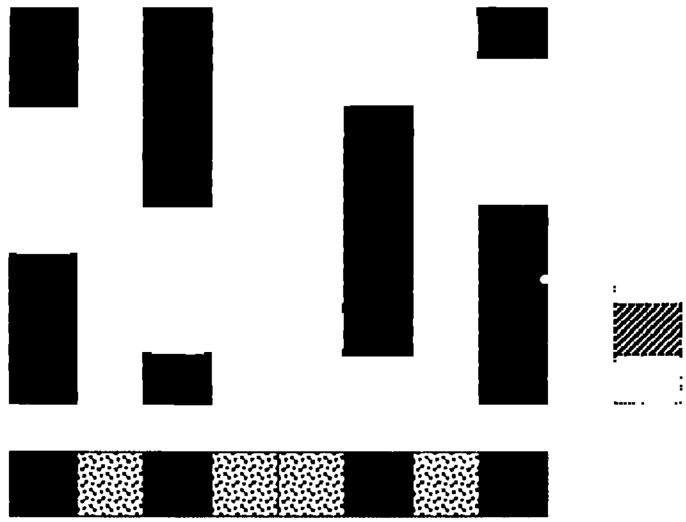
(see figure 12)



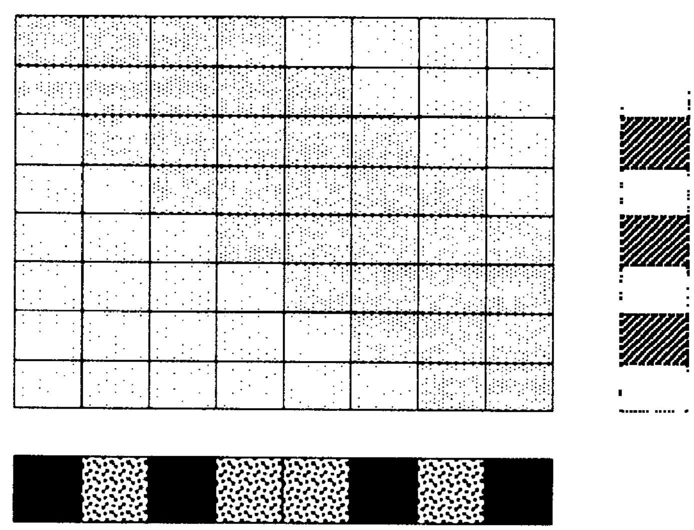
Color array and derived color vectors
figure 8



Permuted color array with derived color vectors
figure 9



Color array and derived color vectors
figure 10



Interlacement and color vectors

figure 11

```
        1  1 1 0   0 1        V =   2
          1   1   0   0             4
        0   1 1 1   0 0             2
        0 0 1   1 1 1               3
D =     0   0 1 1   1 1             2
          0   0   1   1             4
        1   0 0 0   1 1             2
        1 1 0   0 0 1               3

U =     4 2 4 3 4 3 4 3
```

(see figure 13)

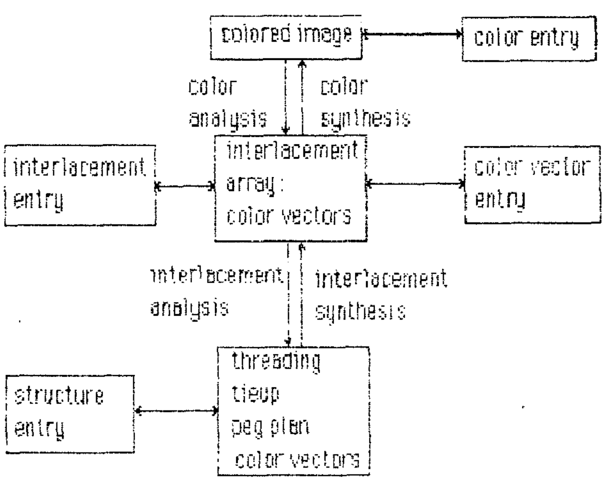
```
               |1 0|1 0       V^P =  4
         ______|0 1|0 1              4
        1 1 0 0|   |1 1              2
        0 1 1 0|   |1 0              2
D^P =   0 0 1 1|   |1 1              2
        1 0 0 1|___|____             2
        0 1 1 1|0 0|                 3
        1 0 0 1|1 0|                 3

U^P =   4 4 4 4|2 2|3 3
```

(see figure 14)

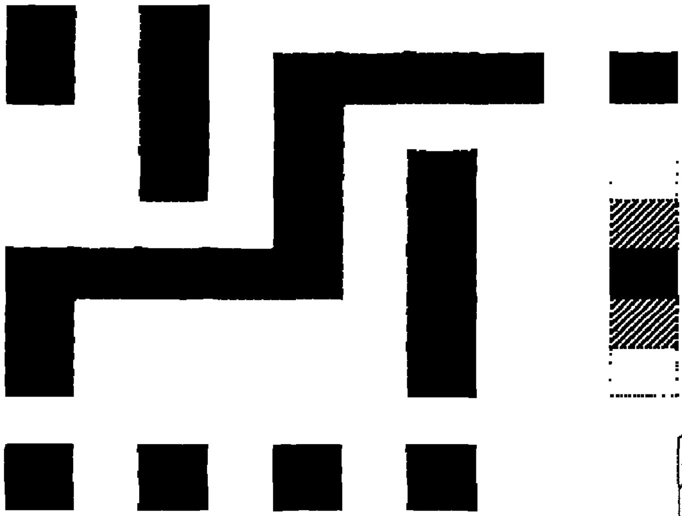
4. Observations

Addition of the color analysis and syntnesis modes to the design environment of [7], has meant that the development of proposed textile structures can now be approached along a variety of possible paths, and any image, no matter how acquired, can be captured by the system and become part of the process. Some sample sequences are illustrated by the following:

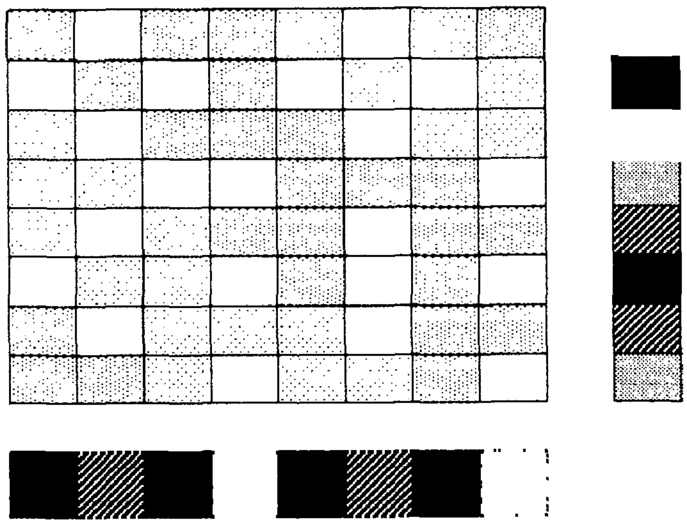


5. Conclusions

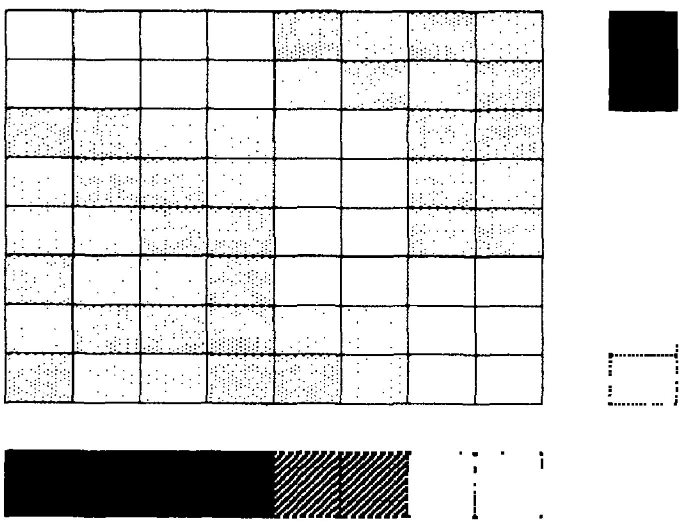
The prototype implementation on an IBM PC/XT has demonstrated that it is possible to provide the important elements of the color, and color analysis, environments for the textile designer without hopelessly compromising storage requirements, or degrading the response of the system described in [7]. The size-of-design limitations are realistic for most industrial requirements and the restricted color sets are almost acceptable. The iterative, interactive behaviour between the create interlacement, analyse interlacement, modify-color vectors, analyse-color array segment is very useful in design and simplifies the entire process dramatically.



Color array and derived color vectors
figure 12



Interlacement array, derived color vectors, and indeterminate texels
figure 13



Permuted interlacement array showing regions where arbitrary assignment of interlacing takes place

figure 14

REFERENCES

1 Aho, Alfred V., Hopcroft, John E. and Ullman, Jeffrey D., Data Structures and Algorithms, (Reading, Mass.: Addison-Wesley Publishing Company, 1983), 215-226

2 Clapham, C.R.J., When a Fabric Hangs Together, Bull. London Math. Soc. 12 (1980), 161-164

3 Foley, J.D. and Van Dam, A., Fundamentals of Interactive Computer Graphics, (Philippines: Addison-Wesley Publishing Company, 1982)

4 Grosicki, Z.J., Watson's Advanced Textile Design: Compound Woven Structures, 4th ed., (London: Butterworth & Co. (Publishers) Ltd., 1977)

5 Hauptmann, Bruno, Gewebetechnik, (Leipzig: Fachbuchverlag GMBH, 1952), p. 138

6 Hoskins, J.A., Binary Interlacement Arrays and Structural Cross-Sections, Cong. Num. 40 (1983), 63-76

7 Hoskins, J.A. and Hoskins, W.D., Algorithms for the Design and Analysis of Woven Textiles, Proceedings of the 1983 ACM Conference on Personal and Small Computers, 153-160

8 Hoskins, Janet A. and Hoskins W.D., The Solution of Certain Matrix Equations Arising from the Structural Analysis of Woven Fabrics, Ars Comb. 11 (1981), 51-59

9 Hoskins, J.A. and Hoskins, W.D., A Faster Algorithm for Factoring Binary Matrices, Ars Comb. 16b (1983), 341-350

10 Lourie, Janice R., Loom-Constrained Designs: An Algebraic Solution, Proceedings of the ACM National Conference (1969), 185-192

11 Lourie, Janice R., Textile Graphics/Computer Aided, (New York: Fairchild Publications, Inc., 1973)

12 Lourie, Janice R., The Textile Designer of the Future, Handweaver and Craftsman, Winter 1966, 8

13 Lourie, Janice R. and Bonin, Alice M., Computer-Controlled Textile Designing and Weaving, Proceedings - IFIPS (1968), 884

14 Lourie, Janice R., Lorenzo, John J. and Bomberault, Abel, On-Line Textile Designing, Proceedings of the ACM National Meeting (1966), 537