# Maximal Superpositions of Grids and an Application

Christophe Fiorio, Gilles d'Andréa

# Maximal Superpositions of Grids and an Application

Gilles d'Andréa and Christophe Fiorio

LIRMM, 161 rue Ada, 34392 Montpellier Cedex 5

**Abstract.** In this paper, we present an efficient algorithm of maximals superpositions of two digital sets coded in grids. It represents the basis of an application which controls the printing quality of industrial products labels, in which we consider a grid as the representation of a character.

## 1   Introduction

The problem we would like to solve is to control the printing quality of information on labels of industrial products. This application is different from the classical character recognition applications. Indeed, the problem here is not to identify a text, but to compare a printed text with a pattern of the text to be printed. It concerns more a default detection application than a character recognition application.

In [dF97] we have presented an algorithm which determines maximal superpositions of two horizontally convex polyominoes. We call a polyomino, a connected finite set of adjacent dashed cells lying two by two along a side in a rectangular grid. Polyominoes were first introduced by Golomb in 1954 in a talk to the Harvard Mathematics Club, see [Gol54,Gol65]. It is clear that a polyomino can represent a character, but unfortunately all characters can't be coded with a polyomino. Thus, we will present a new algorithm based on the previous one which, given two grids $P$ and $P'$, determines the (horizontal and/or vertical) translation of $P$ on $P'$ such that the cardinal of the superposed dashed cells is maximal.

The application of printing quality control is based on this algorithm and prepare the comparison between printed characters to patterns to print.

## 2   Superposition of two Grids

Given $P$ et $P'$ two digital sets coded in two grids of $n$ (resp. $n'$) lines, and of $m$ (resp. $m'$) columns, given a vertical translation, there is at most

$(m+m'-1)$ possible horizontal translations of $P$ on $P'$ such that the two grids have at less one cell superposed. In the same way, given a horizontal translation, there is at most $(n+n'-1)$ possible vertical translations of $P$ on $P'$ such as $P$ and $P'$ have at less one superposed cell.

Thus, we can build a matrix $M_{P/P'}$ of $(m+m'-1) \times (n+n'-1)$ cells, where each element $M_{P/P'}[h,v]$ of the matrix gives the number of superposed dashed cells for this translation of $P$ on $P'$ of $h$ cells horizontally and of $v$ cells vertically[1]. Besides, for every translation, the cardinal of the superposed dashed cells can be computed in scanning the cells set of $P$. The complexity of this naive algorithm is therefore in $O\left((n+n') \times (m+m') \times (n \times m)\right)$.

## 3   Presentation of our Algorithm

Every grid is decomposed in horizontal lines segments. The algorithm is based on this decomposition, each superposition of segments is independently processed to compute the total of superposed dashed cells.

### 3.1   Grids Superposing and Coding

We use a horizontal line by line coding for the grids. For every horizontal line, of the grid, every set of consecutive dashed cells (segment) is coded by a pair marking the position of the first dashed cell and the last dashed cell (see figure 1). Thus, the number of pairs coding a line is equal to the number of segments composing this line.



| $l_1$ | (1,1), (3,3), (5,5) |
| $l_2$ | (1,1), (5,5) |
| $l_3$ | (2,4) |

| $l'_1$ | (2,5) |
| $l'_2$ | (1,2), (5,6) |
| $l'_3$ | (1,2), (5,6) |
| $l'_4$ | (2,5) |

P                                    P'

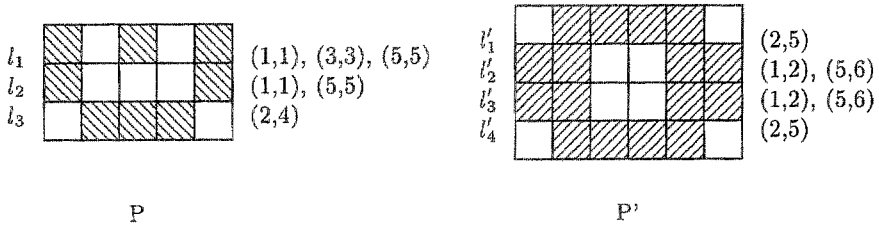**Fig. 1.** Example of 2 grids and their associated coding.

Given $P$ and $P'$ two grids respectively coded in two sets of lines $L = \{l_1, l_2, ..., l_n\}$ and $L' = \{l'_1, l'_2, ..., l'_{n'}\}$, given $v$ a vertical translation and $h$ a

---

[1] $h$ (resp. $v$) is positive if $P'$ is translated to the right (resp. to the down), negative if $P'$ is translated to the left (resp. to the up).

horizontal translation of $P$ on $P'$, we define $S_{P/P'}$ the function computing the number of dashed cells of $P$ superposed to the dashed cells of $P'$ (*overlapping surface*). A superposition of $P$ on $P'$ is said maximal if the overlapping surface is maximal.

Given a vertical translation $v$, a lines subset of $P$ is superposed to a lines subset of $P'$. Given a horizontal translation $h$, let's define $S_{l/l'}$ the function computing the overlapping surface of two lines of the grids. Let $L_v$ the set of lines pairs superposed of $P$ on $P'$ for a vertical translation $v$, we get:

$$M_{P/P'}[h, v] = \sum_{(l,l') \in L_v} S_{l/l'}(h)$$

Given a translation, the overlapping surface of two grids is brought back to the sum of overlapping surfaces of the lines of $P$ superposed to the lines of $P'$. We are going now to discuss about the superposition of two lines.

## 3.2   Superposition of two Lines of two Grids

Given $l$ and $l'$ two grids lines, respectively defined by two sets of pairs $\{(x_1, y_1), (x_2, y_2), ..., (x_q, y_q)\}$ and $\{(x'_1, y'_1), (x'_2, y'_2), ..., (x_{q'}, y_{q'})\}$, for all the horizontal translations, every segment of $l$ is superposed to every segment of $l'$. We are lead to the study of the superpositions properties of a segment of $l$ on a segment of $l'$.

Given $t$ and $t'$ two segments of two grids lines respectively defined by pairs $(x, y)$ and $(x', y')$, let's defines the function $S_{t/t'}$ that computes the overlapping surface of $t$ on $t'$ for a horizontal translation $h$.

Dashed cells of $t$ and $t'$ are consecutive, so all the maximal superpositions of $t$ on $t'$ are also consecutive and may be coded by a triplet $(B_{t/t'}, E_{t/t'}, N_{t/t'})$ (see Figure 2). With $N_{t/t'}$ the number of superposed
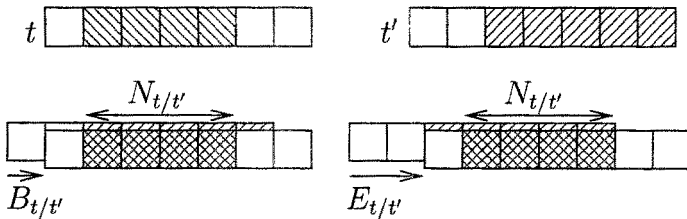


**Fig. 2.** maximal superpositions of 2 segments of 2 grids lines.

dashed cells, and $B_{t/t'}$ (resp. $E_{t/t'}$) the minimum shift (resp. maximum) of $t$ giving $N_{t/t'}$, we have:

$$B_{t/t'} = min(x' - x, y' - y) \tag{1}$$

$$E_{t/t'} = max(x' - x, y' - y) \tag{2}$$

$$N_{t/t'} = min(y - x + 1, y' - x' + 1) \tag{3}$$

The functions $S'_{t/t'}$, $S''_{t/t'}$ are defined by:

$$S'_{t/t'}(h) = S_{t/t'}(h + 1) - S_{t/t'}(h) \tag{4}$$

$$S''_{t/t'}(h) = S'_{t/t'}(h) - S'_{t/t'}(h - 1) \tag{5}$$

Let's studying these functions:

| $h$ | $...B-N-1$ | $B-N$ | $B-N+1$ | $...$ | $B-1$ | $B$ | $...$ | $E$ | $E+1$ | $...$ | $E+N-1$ | $E+N$ | $E+N+1 ...$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S(h)$ | 0 | 0 | 1 | ↗ | $N-1$ | $N$ | — | $N$ | $N-1$ | ↘ | 1 | 0 | 0 |
| $S'(h)$ | 0 | 1 | 1 | — | 1 | 0 | 0 | -1 | -1 | — | -1 | 0 | 0 |
| $S''(h)$ | 0 | 1 | 0 | — | 0 | -1 | 0 | -1 | 0 | — | 0 | 1 | 0 |

We can notice that four key-values define the function $S''_{t/t'}$. Indeed, $S''_{t/t'}(h) \neq 0$ for $h \in \{B - N, B, E, E + N\}$. So, the function $S''_{t/t'}$ can be coded by these four values, that are directly deductible of maximal superpositions triplets. Then, we can compute the function $S_{t/t'}$ from $S''_{t/t'}$ with the adequate replacements:

$$S''_{t/t'}(h) = S_{t/t'}(h + 1) - 2S_{t/t'}(h) + S_{t/t'}(h - 1) \tag{6}$$

$$S_{t/t'}(h) = S''_{t/t'}(h - 1) + 2S_{t/t'}(h - 1) - S_{t/t'}(h - 2) \tag{7}$$

The overlapping surface for every superposed lines pairs may be decomposed in a sum of overlapping surface of superposed segments. So, we get:

$$S_{l/l'}(h) = \sum_{(t,t')\in l \times l'} S_{t/t'}(h)$$

## 3.3 Discrete Function of two Grids Superposition

For every vertical translation $v$, we determine the overlapping surface of the two superposed grids by computing the sum of the function of superposed lines overlapping surfaces, so:

$$M_{P/P'}[h,v] = \sum_{(l,l')\in L_v} S_{l/l'}(h) = \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} S_{t/t'}(h)$$

To avoid the calculation of the function $S_{t/t'}$ for every horizontal translation $h$, we use the function $S''_{t/t'}$ that is coded on four key-values. By means of the following equations, all the overlapping surfaces may be iteratively computed in a single scan.

$$M_{P/P'}[h,v] = \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} \left(S''_{t/t'}(h-1) + 2S_{t/t'}(h-1) - S_{t/t'}(h-2)\right)$$

$$M_{P/P'}[h,v] = \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} \left(S''_{t/t'}(h-1)\right) + \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} \left(2S_{t/t'}(h-1)\right)$$

$$- \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} \left(S_{t/t'}(h-2))\right)$$

$$M_{P/P'}[h,v] = \sum_{(l,l')\in L_v} \sum_{(t,t')\in l\times l'} \left(S''_{t/t'}(h-1)\right) + 2M_{P/P'}[h-1,v] - M_{P/P'}[h-2,v]$$

## 3.4 The Algorithm

The algorithm computes a matrix $M''_{P/P'}$, sum of functions $S''_{t/t'}$ for every pair of segments superposed. Then, the superpositions matrix $M_{P/P'}$ is computed with $M''_{P/P'}$ and the equation previously given.

## 3.5 Analysis of Complexity

Given $P$ and $P'$ two grids of $n$ (resp. $n'$) lines and $m$ (resp. $m'$) columns and given $q$ (resp. $q'$) the maximal number of segments in a line of $P$ (resp. $P'$). The grids $P$ and $P'$ may be extended with empty lines and/or empty columns to $N$ lines and $M$ columns such that $M = max(m, m')$ and $N = max(n, n')$ without modification on $q$ and $q'$.

The Algorithm 1 we can distinguish two parts: the building of the matrix $M''_{P/P'}$ and the calculation of $M_{P/P'}$ from $M''_{P/P'}$. The building of

---

**Algorithm 1:** Computing of the overlapping surfaces matrix $M_{P/P'}$.

---

> **Data** : Two Grids $P$ and $P'$ of respectively $n \times m$ and $n' \times m'$ cells
>
> **Result** : $M_{P/P'}$ translations matrix of $P$ on $P'$ of $(n+n'-1) \times (m+m'-1)$ elements

1   *Initialization to 0 of the $M''_{P/P'}$ elements*

2   **foreach** *vertical translation $v$* **do**

3      **foreach** *pair $(l, l')$ of superposed lines* **do**

4         **foreach** *segment $(x, y)$ of line $l$* **do**

5            **foreach** *segment $(x', y')$ of line $l'$* **do**

6               $B \leftarrow min(x' - x, y' - y)$

7               $E \leftarrow max(x' - x, y' - y)$

8               $N \leftarrow min(y - x + 1, y' - x' + 1)$

9               $M''_{P/P'}[B - N, v] + +$

10               $M''_{P/P'}[B, v] - -$

11               $M''_{P/P'}[E, v] - -$

12               $M''_{P/P'}[E + N, v] + +$

13   **for** $v \leftarrow -n + 1$ *to* $n' - 1$ **do**

14      $M_{P/P'}[-m + 1, v] \leftarrow M''[-m, v]$

15      $M_{P/P'}[-m + 2, v] \leftarrow M''[-m + 1, v] + 2M_{P/P'}[-m + 1, v]$

16      **for** $h \leftarrow -m + 3$ *to* $m' - 1$ **do**

17        $M_{P/P'}[h, v] \leftarrow M''[h - 1, v] + 2M_{P/P'}[h - 1, v] - M_{P/P'}[h - 2, v]$

---

the matrix $M''_{P/P'}$ (line 1 to 12) is in $O(N^2 \times (q \times q'))$. The computing of the matrix $M_{P/P'}$ (line 13 to 17) requires $O(M \times N)$ operations. So, the global complexity of the algorithm is in $O(N \times (M + N \times (q \times q'))$

In considering that the maximal number of segments in a line is bounded, the complexity becomes $O(N \times (M + N)$. Such hypothesis is reasonable in a concrete application, like the research of maximal superpositions of two characters. In the worst case where the number of segments is in $O(M)$ (a checkerboard grid for example), we achieve an equivalent complexity of the naive algorithm discussed in section 2.

## 4   A Concrete Application

In the food or pharmaceutical industry, the impression of expiration dates or informations on products must remain perfectly readable. Printing processes often generate defaults. The goal of our application, is to verify if the printed text is sufficient in quality and without ambiguity.

In our application, every character is coded by a matrix such that for every dashed cell, an ink-bubble will be projected for the printing. The assessment of the printing quality consists therefore to compare the *characters to print* to *printed characters.*

## 4.1 Characters Segmentations and Deformations Analysis



**Fig. 3.** Characters to Print



**Fig. 4.** Printed Characters

The printing matrix of every character is coded by a digital set where each projected ink bubble corresponds to a dashed cell (see figure 3). The printed characters undergo the distortions owed to the printing process (trajectories of inks bubbles are not perfect) and the picture aquisition process (the picture resolution is such that every ink bubble covers a set of dashed cells, see figure 4).



**Fig. 5.** Segmented Printed Characters

The printed characters undergo distortions of scaling, rotations and slanting that are estimated by the analysis of the parallelogram including the printed text. A space analysis between characters allows to separate every character (see [CL96] and Figure 5). Then, every character is coded as a digital set (grid).

A scaling, rotation, and slant similar to the printed characters is applied to the character to print in order to get two sets of characters theoretically identical, in size, rotations and slants. Differences between grids of the printed characters and grids of the characters to print after distortions are generated by the imprecisions of inks bubbles trajectories. The analysis of these differences is the basis to decide if a character still remains readable without ambiguity.

## 4.2   Characters Superpositions

For every character, we are looking for one maximal superposition of the grid to print on the printed polyomino to determine the differences. In the characters case, the maximal number of segments in a horizontal line is bounded to four with the character "M" and "W", so the algorithm presented here, is very convenient.

A first method to compare the two superposed grids consists in counting all dashed cells which are exclusively in one set. This brutal method has for inconvenience to not hold on with the characters geometry. Indeed, deviations of projected ink bubbles, or their missing can create or break connexity between the different parts of characters. A "O" can easily become a "C", and vice versa.

The characters printing quality depends on all of these parameters: distortions owed to rotations, slants and scales, as well as the add or loss of connexity between the different parts of the character. In quantifying these parameters and in weighting them, we get a quality criterion for every printed character on which is based our application.

## 5   Conclusion

We presented an algorithm computing the overlapping surface for the set of superpositions of two difital sets coded in grids. This algorithm is a generalization of the algorithm presented in [dF97]. It allows to easily determine the maximal superpositions of two grids. This algorithm is used for a practical application of printing quality control where we must determine if characters have been correctly printed, i.e. every printed character can be read without ambiguity.

This algorithm is linear in the size of the superpositions matrix if the number of segments composing a line of the grids remains limited. In the general case, the complexity is equivalent to the naive algorithm presented in section 2. But in our application, where the grids to be superposed are characters, thus the number of segments of a line is bounded and we can benefit the improvement brought by our algorithm.

# References

[CL96]  R. G. Casey and E. Lecolinet. A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):690–706, July 1996.

[dF97]  G. d'Andréa and C. Fiorio.  Superpositions maximales de polyominos horizontalement convexes.  Rapport de Recherche LIRMM, Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, 161 rue Ada – 34392 Montpellier Cedex 5 – France, July 1997.

[Gol54]  S.W. Golomb.  Checkerboards and polyominoes. *American Mathematical Monthly, LXI*, 10:672–682, December 1954.

[Gol65]  S.W. Golomb. *Polyominoes*. Charles Scribner's Sons, New-York, 1965.