Chapter 3

HORIZONTAL BLACK RUNS AND THEIR PARAMETERS

§3.1 Definitions and Some Properties of Runs

The rawest form of description of a figure F is to give, for every pixel in the grid, its value 0 or 1. This is called the *bit map code.* This form of coding is expensive in memory, and does not make explicit the structure of the figure. In this section, we analyse the properties of a one-dimensional structure of a figure, namely, its *runs.*

For $0 \leq i \leq M-1$, and $0 \leq b \leq e \leq N-1$, we define

$$[b\,..\,e]_i \doteq \{p(i,j) \mid b \leq j \leq e\}. \tag{3.1}$$

Such an interval is a (horizontal) *run,* with *beginning* b and *end* e. When there is no ambiguity on the row i we write $[b\,..\,e]$ for $[b\,..\,e]_i$.

Let us recall the two types of adjacencies on a quadruled grid. Pels $p(i,j)$ and $p(i',j')$ are 4-adjacent if and only if

$$|i-i'|+|j-j'|=1.$$

They are 8-adjacent if and only if

$$max(|i-i'|,|j-j'|)=1.$$

Consider two runs $[b\,..\,e]_i$ and $[b'\,..\,e']_{i'}$ on two successive rows (i.e., $i' = i \pm 1$). For $k = 4$ or 8, we shall say that these runs are not k-adjacent if and only if no pixel of the former is k-adjacent to some pixel of the latter. This is true if and only if one of the following holds:

$$b > e' + h \tag{3.2}$$

or

$$b' > e + h, \tag{3.3}$$

where

$$\begin{aligned} h &\doteq 0 \quad \text{if} \quad k = 4 \\ &\doteq 1 \quad \text{if} \quad k = 8. \end{aligned} \tag{3.4}$$

If (3.2) holds, we shall say that $[b'\,..\,e']$ is to the left of $[b\,..\,e]$. If (3.3) does not hold, i.e., if $b' \le e + h$, we shall say that $[b'\,..\,e']$ is not to the right of $[b\,..\,e]$. (Note that, somewhat counterintuitively, this definition does not prevent $[b'\,..\,e']$ from possibly extending far to the right of $[b\,..\,e]$.) On the other hand, runs $[b\,..\,e]$ and $[b'\,..\,e']$ are k-adjacent if and only if

$$b \le e' + h \quad \text{and} \quad b' \le e + h. \tag{3.5}$$

Consider now a figure F. It is a union of *black* runs. Of course, we will consider as runs of F only *maximal* runs, i.e., runs $[b\,..\,e]_i$ such that

$$x(i, b-1) = x(i, e+1) = 0,$$

and

$$x(i,j) = 1 \quad \text{for} \quad b \le j \le e. \tag{3.6}$$

(We know that $a > 0$ and $b < N-1$ by the frame assumption.)

The k-connected components of F are unions of runs of F; the runs of a k-connected component of F forms together a k-connected component of the set $\mathcal{F}$ of runs of F, and vice versa.

Two runs of F may be adjacent only if they are on successive rows.

§3.2 The Parameters of a Run

In this section, we associate to runs several parameters which will permit to recover adjacency properties. The computation of these parameters will be the subject matter of the next section.

For $0 \leq i \leq M-1$, we let $nbrun[i]$ designate the number of runs of F on row i. Then, the runs on row i can be written

$$run[i,t] \qquad 0 \leq t \leq nbrun[i]-1, \tag{3.7}$$

with index t increasing from left to right. Given $run[i,t] = [b..e]_i$, set

$$be[i,t] \doteq b, \tag{3.8}$$

$$en[i,t] \doteq e, \tag{3.9}$$

where, clearly, "be" and "en" stand for beginning and end.

Next, we associate to $run[i,t]$ the following four parameters:

- $lefpr[i,t] \doteq$ number of runs which are to the left of $run[i,t]$ on the preceding row, $(i > 0)$.
- $lefsu[i,t] \doteq$ number of runs which are to the left of $run[i,t]$ on the succeding row, $(i < M-1)$.
- $nripr[i,t] \doteq$ number of runs which are not to the right of $run[i,t]$ on the preceding row, $(i > 0)$.
- $nrisu[i,t] \doteq$ number of runs which are not to the right of $run[i,t]$ on the succeding row, $(i < M-1)$.

These parameters allow us to determine the labels of the runs of F, $run[i \pm 1, u]$, $u \geq 0$, which are k-adjacent to $run[i, t]$. If such runs exist, they are

$$(a) \quad run[i-1, u] \quad \text{where} \quad lefpr[i,t] \leq u \leq nripr[i,t] - 1; \tag{3.10}$$

$$(b) \quad run[i+1, u] \quad \text{where} \quad lefsu[i,t] \leq u \leq nrisu[i,t] - 1. \tag{3.11}$$

The numbers of such runs are

$$\text{in } (a): \qquad conpr[i,t] \doteq nripr[i,t] - lefpr[i,t], \tag{3.12}$$

$$\text{in } (b): \qquad consu[i,t] \doteq nrisu[i,t] - lefsu[i,t], \tag{3.13}$$

where the prefix "con" stands for "connected to".

Equations (3.10) and (3.11) are illustrated by Figure 3.1 under both 4- and 8-adjacencies.

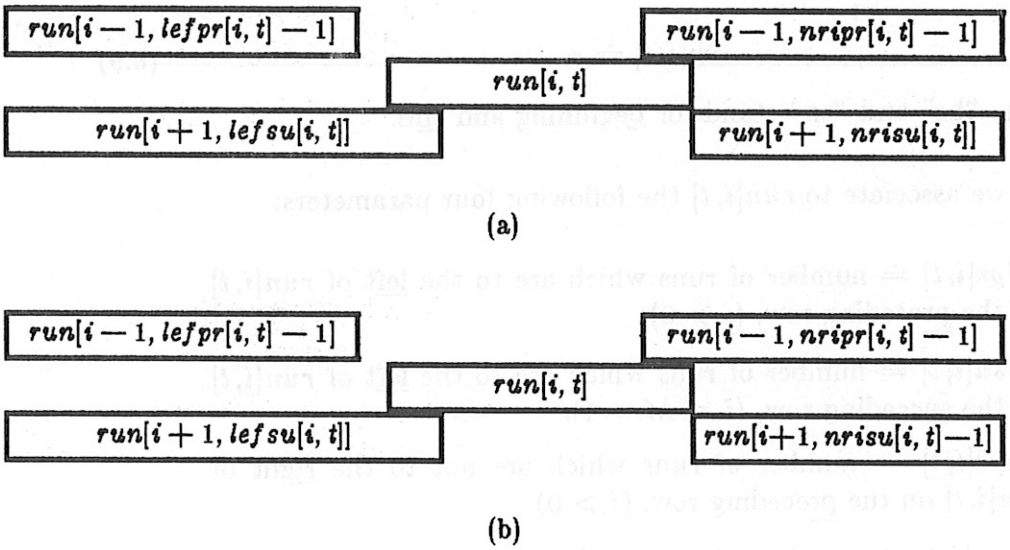

Figure 3.1. Run parameters under 4–(a) and 8–(b) adjacencies.

For the sake of clarity of the presentation, it will be convenient, in the sequel, to have available two "dummy" parameters, namely *rigpr* and *rigsu*, which are defined as follows:

$$rigpr[i,t] \doteq nripr[i,t] - 1,$$

$$rigsu[i,t] \doteq nrisu[i,t] - 1.$$

§3.3 The Computation of Run Parameters

To evaluate the parameters *lefpr*, *lefsu*, *nripr*, and *nrisu* we shall make use of the dummy functions $lef(i',j')$, and $nri(i',j')$ which merely count the numbers of black-to-white and white-to-black transitions along row i'.

For $i' = 0,\ldots,M-1$, and $j' = 0,\ldots,N-2$, set

$$\begin{aligned} lef(i',j') &\doteq 0 \quad \text{if } 0 \le j' \le h \ \ (i.e., k = 8 \text{ and } j' = 0) \\ &\doteq \sum_{\ell=h}^{j'} \big(1 - x(i',\ell+g)\big) \times x(i',\ell-h) \quad \text{otherwise}; \qquad (3.14) \\ nri(i',j') &\doteq 0 \quad \text{if } 0 \le j' \le g \ \ (i.e., k = 4 \text{ and } j' = 0) \\ &\doteq \sum_{\ell=g}^{j'} \big(1 - x(i',\ell-g)\big) \times x(i',\ell+h) \quad \text{otherwise}; \qquad (3.15) \end{aligned}$$

where

$$\begin{aligned} g \doteq 1 - h &= 1 \quad \text{if } k = 4 \\ &= 0 \quad \text{if } k = 8. \qquad (3.16) \end{aligned}$$

If, in the sum (3.14), we set $u = \ell - h$, $lef(i',j')$ is the number of integers u, $0 \le u \le j' - h$, such that $x(i',u) = 1$ and $x(i',u+1) = 0$, i.e., a transition from black to white, or the right end of a black run. Now, we can take $i' = i \pm 1$ and $j' = be[i,t] - 1$. Then, since $u \le be[i,t] - h - 1$, any such pixel $p(i',u)$ is the end of a run to the left of $run[i,t]$. Thus,

$$lefpr[i,t] = lef(i-1, be[i,t]-1),$$

and

$$lefsu[i,t] = lef(i+1, be[i,t]-1). \tag{3.17}$$

By the same reasoning it readily follows that $nri(i',j')$ is counting a number of white to black transitions, and

$$nripr[i,t] = nri(i-1, en[i,t]),$$

$$nrisu[i,t] = nri(i+1, en[i,t]). \tag{3.18}$$

The computation of functions lef and nri can be performed iteratively within a window of size 3 moving along row i', and centered at pel $p(i',j')$. Specifically, we have the following iteration:

$$\begin{aligned}
j' = 0 \quad & lef(i',j') := 0. \\
& nri(i',j') := 0 \quad \text{if } k=4, \\
& \qquad := \big(1 - x(i',0)\big)\times x(i',1) \quad \text{if } k=8. \qquad (3.19) \\
j' > 0 \quad & lef(i',j') := lef(i',j'-1) \\
& \qquad + \big(1 - x(i',j'+g)\big)\times x(i',j'-h). \\
& nri(i',j') := nri(i',j'-1) \\
& \qquad + \big(1 - x(i',j'-g)\big)\times x(i',j'+h). \qquad (3.20)
\end{aligned}$$

With all these ancillary details settled, we wish to examine briefly the processing of runs along row i. This processing is described by the procedure *processonrow*, Section 1, Appendix B. All operations are performed within a 3×3 window centered at pel $p(i,j)$. Prior to starting the processing of row i, the current run index u, and lef and nri are set to initial values (3.19).

For any position (i,j) of the window, in fact for $j = 0, \ldots, N-2$, the following steps are carried out:

First, $lef(i \pm 1, j)$, and $nri(i \pm 1, j)$ are computed using (3.19) and (3.20). Using an obvious notation, the code reads:

```
lp:=lp+(1-frow[j+g])*frow[j-h];
ls:=ls+(1-trow[j+g])*trow[j-h];
np:=np+(1-frow[j-g])*frow[j+h];
ns:=ns+(1-trow[j-g])*trow[j+h]
```

Second, a test is performed to detect the presence of the pattern $x(i,j) = 0$ and $x(i,j+1) = 1$.

```
IF (srow[j]=0) AND (srow[j+1]=1) THEN
```

When the test is successful, the beginning of a new run is registered, (see (3.7) and (3.17)):

```
be:=j+1;
lefpr:=lp;                     (3.21)
lefsu:=ls
```

Third, a test is performed to detect the presence of the pattern $x(i,j) = 1$ and $x(i,j+1) = 0$.

```
IF (srow[j]=1) AND (srow[j+1]=0) THEN
```

When the test is successful, the end of the current run is registered, (see (3.8) and (3.18)):

```
en:=j;
nripr:=np;
nrisu:=ns;                     (3.22)
conpr:=nripr-lefpr;
consu:=nrisu-lefsu;
```

Also included in this third step is the incrementation of the run index. Finally, upon completing the processing of row i, we have $nbrun[i] = u$.

Clearly, the processing which we just described has a time cost linear in the number of pixels processed, as we required in Section 2.1.3. As far as storage space is concerned, the requirements essentially consist in the $3N$ words necessary to store the three rows. Additional requirements are negligible. In fact, we will see in the following chapter that the information conveyed by the eight parameters in (3.21) and (3.22) is exploited, at the next higher level of our hierarchical system, as soon as these parameters have been acquired (see the call of procedure *allocate* in *processonrow*), and it should be evident that the acquisition itself can be performed at the cost of very little extra-storage. Incidentally, one should note that this observation clarifies the reason why run parameters appear in the program without the "parameterization" which was used for the sole purpose of the theoretical exposition.