

case horizontally to the left) completely (that is, until it reaches the edge of the region) and only then examines the remaining directions. In contrast the *breadth-first* method markings proceed outward, step by step, equally in all directions.

Due to the way exploration takes place, the memory requirement of the *breadth-first* variant of the *flood-fill* version is generally much lower than that of the *depth-first* variant. For example, when flood filling the region in Fig. 11.2 using the implementation given (Prog. 11.1), the stack in the *depth-first* variant grows to a maximum of 28,822 elements, while the queue used by the *breadth-first* variant never exceeds a maximum of 438 nodes.

### 11.1.2 Sequential Region Labeling

Sequential region marking is a classical, nonrecursive technique that is known in the literature as “region labeling”. The algorithm consists in essence of two steps: (1) a preliminary labeling of the image regions and (2) resolving cases where more than one label occurs (i. e., has been assigned in the previous step) in the same region. Even though this algorithm is relatively complex, especially the second stage, its moderate memory requirements have made it the method of choice in practice over other simpler methods. The entire process is summarized in Alg. 11.2.

#### Step 1: Preliminary labeling

In the first stage of region labeling, the image is traversed from top left to bottom right sequentially to assign a preliminary label to every foreground pixel. Depending on the definition of neighborhood (either 4- or 8-connected) used, the vicinity of each pixel must be examined ( $\times$  marks the actual pixel at the position  $(u, v)$ ):

$$\mathcal{N}_4(u, v) = \begin{array}{ccc} & & N_2 \\ & \times & \\ N_1 & & \end{array} \quad \text{or} \quad \mathcal{N}_8(u, v) = \begin{array}{ccc} N_2 & N_3 & N_4 \\ N_1 & \times & \\ & & \end{array}$$

When using the 4-connected neighborhood  $\mathcal{N}_4$ , only the two neighbors  $N_1 = I(u-1, v)$  and  $N_2 = I(u, v-1)$  need to be considered, but when using the 8-connected neighborhood  $\mathcal{N}_8$ , all four neighbors  $N_1 \dots N_4$  must be examined. In the following example, we will use an 8-connected neighborhood and the image from Fig. 11.3 (a).

#### Propagating labels

Again we assume that, in the image, the value  $I(u, v) = 0$  represents **background** pixels and the value  $I(u, v) = 1$  represents **foreground** pixels. We will also consider neighboring pixels that lie outside of the image matrix (e. g., on the array borders) to be part of the background. The