BOUNDARY CONDITIONS IN IMAGE MORPHOLOGY

Why have an entire section on boundary conditions? Why not simply adopt a mathematically elegant definition, and use it. For natural images, where there is no *a priori* bias toward any particular grayscale value for pixels beyond the image proper, this is a perfectly reasonable approach. For document images, however, this can give unexpected results.

What is the mathematically elegant definition? It is the one that treats erosion and dilation as duals, even in the presence of boundaries. We will call it the *symmetric boundary condition* (SBC): extend an image by the min value (0 for binary images) for dilation and by the max value (1 for binary images) for erosion. This gives strict duality. To avoid confusion, we call the extra pixels that are used to extend the image beyond its actual boundaries the *frame pixels*.

However, for document images, there is a natural bias toward extending the image with background (0, OFF) pixels, for both dilation and erosion, because we typically have background pixels at the boundaries. So we might want to handle the boundary conditions by requiring that we get the same result as if the image were actually extended sufficiently by 0 pixels, for all morphological operations. This does not give strict duality between dilation and erosion, but I find it preferable for document imaging applications. We will call this the *asymmetric boundary condition* (ABC).

Erosion is thus different for SBC and ABC. Suppose you have a binary image with all pixels ON. An erosion with a 3x3 *Sel* flips a 1 pixel wide border around the image to OFF using ABC, and has no effect on the image using SBC.

The most problematic operation on document images is closing. Consider a document imaging application with binary images, and let's take a specific closing example. Suppose you have an image with one ON pixel, located near the left hand edge at (x=5,y=50), in an image of size 100 x 100, and you dilate with a horizontal structuring element of width 21 pixels with the origin at its center. Here are the three cases:

1. *Closing using asymmetric boundary condition convention with no added border pixels.* The dilation stops at the image boundary, and you get a line only 16 ON pixels long (5 to the boundary, the original pixel, and 10 to the right of the original pixel). The subsequent erosion with a 21 pixel wide *Sel* removes it entirely, because there is no location in which all 21 pixels of the *Sel* are covered by ON pixels. Removal of the original pixel is not typically what you want for a closing operation.

2. *Closing using asymmetric b.c. with added OFF border pixels.* The dilation expands the orignal pixel into a line of length 21 that includes 5 frame pixels, and the subsequent erosion leaves a single pixel, the same one that you started with.

3. *Closing using a symmetric b.c.*. As we will see, this can be implemented without any actual frame pixels being added. But for visualization purposes, imagine there is a frame of OFF pixels for the dilation, which expands the original pixel into a line 21 pixels long, including 5 in the frame. Then change the frame pixels to ON and perform the erosion. Because there are now an infinite number of ON pixels in the frame, the erosion only removes the 10 pixels to the right of the original pixel. The original pixel and the five pixels to its left remain ON. (Remember, the erosion leaves the *Sel* origin ON if all hits in the *Sel* are covered by ON pixels.)

So we get three different results! The third result, where pixels can be connected with the boundary by the closing, is the mathematically correct one in terms of strict duality (SBC), but for document images, does one really want the closing operation to close gaps from image pixels to the boundary? The first result loses the original pixel, which is also likely to be undesirable. Only the second result gives us an image that seems intuitively correct for a situation where the image seems naturally to be extended by OFF pixels because the foreground pixels are assumed to be localized to the actual image (without a frame). Note that the second result is *independent of the location of the actual image boundary*. This is the reason that we're considering boundary conditions in some detail: I prefer ABC for document images.

In Leptonica, you get to choose which convention -- ABC or SBC -- you want to use. There is a function, `resetMorphBoundaryCondition()`, that takes one of two values, *SYMMETRIC_MORPH_BC* and*ASYMMETRIC_MORPH_BC*. We initialize it to ABC, so that by default *all pixels outside the image are assumed to be 0 (OFF) for both dilation and erosion*. The disadvantages of this convention are that we lose duality near the boundary, and have to be careful with closing. However, we gain the condition that an erosion cannot bring ON pixels in from the boundary onto an image that is mostly OFF near the boundary, as most binary document image pages are. If you do a translation (which could be implemented as either a dilation or an erosion by a *Sel* with one hit that is not at the *Sel* center), and you don't want to introduce a rectangle of black pixels, you must use the convention that the image is surrounded by OFF pixels. If you use SBC with such a translation, you will introduce a rectangle of ON pixels in an erosion but not in a dilation.

As you can see, the type of boundary conditions you want to use may depend on your application. Most mathematicians would likely argue for using SBC. However, these issues are introduced here because SBC is not the only possible choice, and with Leptonica you can decide which boundary conditions you want to use.

We discuss below exactly how we handle the boundary conditions. It turns out that there is essentially no difference in implementation complexity or efficiency between the two choices, for either of the two implementations we give here. Those implementations are discussed below: *full-page rasterops* and *destination word accumulation*.

Source : http://www.leptonica.com/binary-morphology.html