# Topology of Digital Images

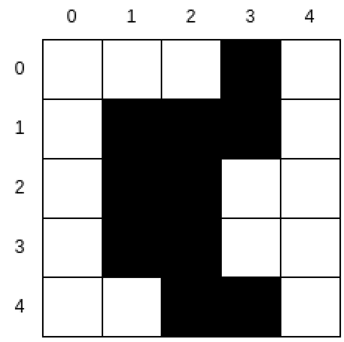
Ali Fele Paranj

July 3, 2025

**Abstract**

In this note I will give a crash course on the topology of digital images. I will define the concepts of local neighborhood, and the topology that it generates. I will discuss the open and closed sets in a digital image, as well as the closure (dilation) and the interior (erosion) operator.

The backbone of a digital image, i.e. its canvas $D$, can be though of as a subset of $\mathbb{Z}^2$, and any particular digital image is simple a function from this domain to $\{0, \cdots, 255\}^3$ where we have assumed that the image is a three channel 8-bit RGB image. However for our purpose here, we will follow a different point of view. We fix a particular binary image (hence a particular function $f : D \to \{0, 1\}$), and then each binary image can be thought of as a set of black and white (or alternatively 0,1) pixels. Using our first point of view, we are in fact working with the product space $D \times \{0, 1\}$, and the image will be the graph of the function $f$. For instance, consider the following digital image.




With our point of view, this image can be though of as the set

$$I = \{(0, 0, 0), (0, 1, 0), (0, 2, 0), (0, 3, 1), (0, 4, 0), (1, 0, 0), (1, 1, 1), (1, 2, 1), \cdots, (4, 3, 1), (4, 4, 0)\}.$$

Since the white pixels and the black pixels carry the dual information (knowing the set of white pixels we can determine the black pixels and vice versa), it is more convenient to assume the white pixels to be the “background” and the black pixels to be the “foreground” and express the e image $I$ by only specifying the foreground pixels. So with this point of view we have

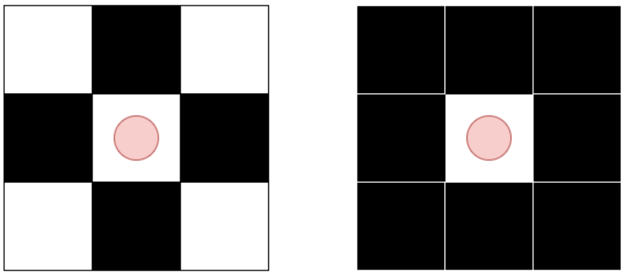
$$I = \{(0, 3), (1, 1), (1, 2), (1, 3), (2, 1), \cdots, (4, 2), (4, 3)\}, \tag{1}$$

where we have also dropped the third component because it is a redundant information (as we know from the context that we are representing the foreground pixels).

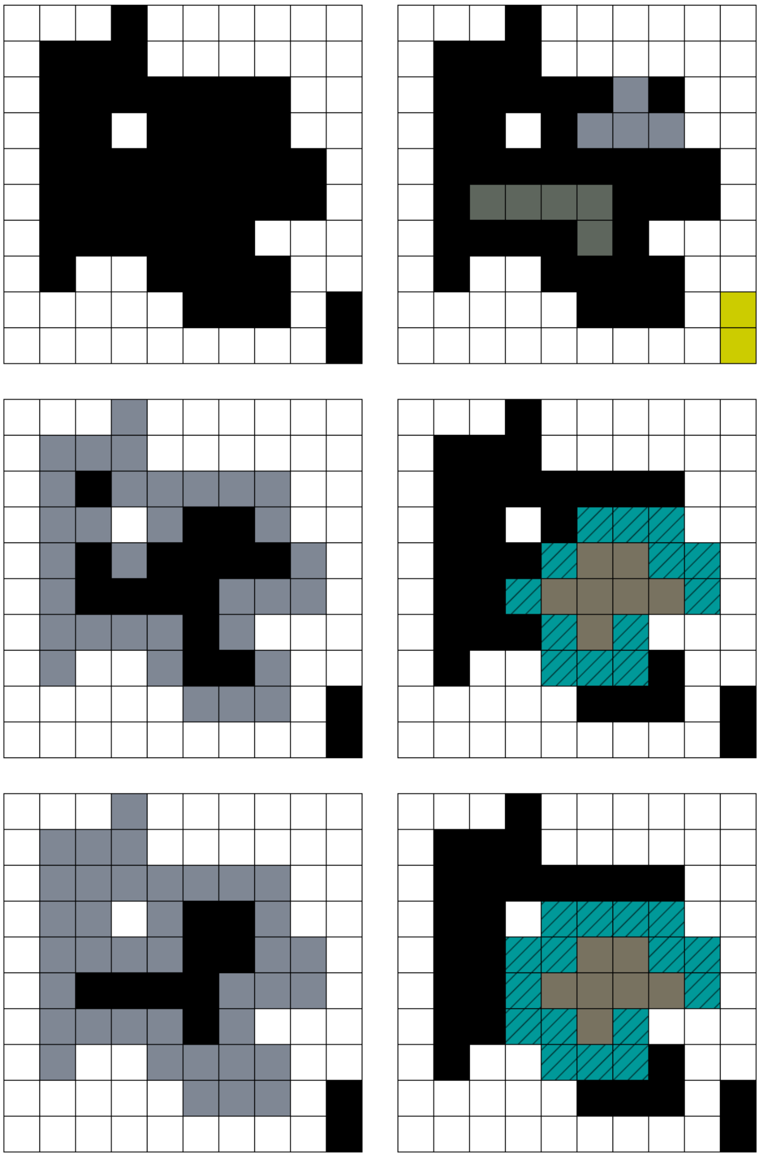
We can now equip this set with a topology, and there are multiple ways to do some. One simple way is by defining a metric on the space. Two metrics are very popular: taxicab metric ($d_1$), and max-metric ($d_\infty$) given as below.

$$d_1((n_1, n_2), (m_1, m_2)) = |m_1 - n_1| + |m_2 - n_2|, \qquad d_\infty((n_1, n_2), (m_1, m_2)) = \max\{|m_1 - n_1|, |m_2 - n_2|\}.$$

These metrics define the neighborhood pixels of a particular pixel, using which we can define a topology on the set. For instance, the following figure demonstrates the local neighborhood of a pixel marked with red dot with the taxicab metric (left figure) and the max-metric (right figure). We can also say that these are the unit balls of radius 1 with the corresponding metric used. These are also sometime called 4-neighborhood, or 8-neighborhood cells.



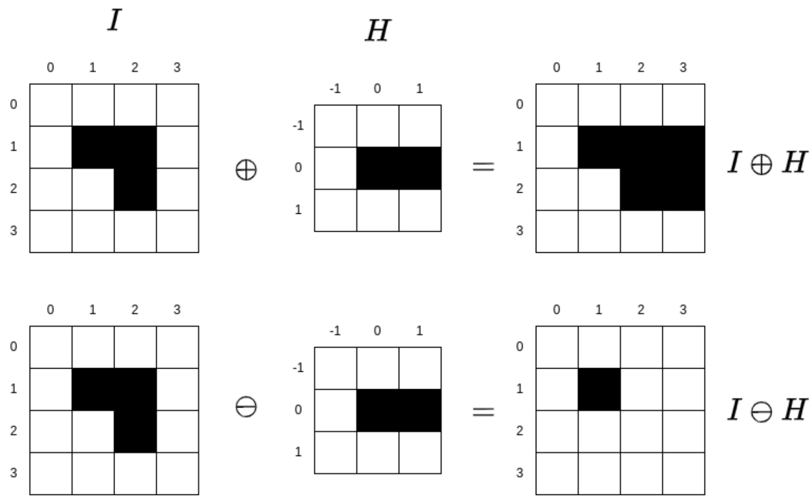
These local neighborhoods is a neighborhood base for a topology, and we can determine this topology by determining the open sets, which are the sets where every point in the set has a local neighborhood that is contained in the set. Once we have the notion of topology, then we can talk about the boundary, interior, and the closure of a set. These concepts are depicted in the figure below.



In the figure above, the subplot (Left, Up) corner represents a binary digital image where the back pixels are the foreground and the white pixels are the background. We can express this space as a set similar to (1). In the (Right, Up) subplot, three subsets are shown: gray, green and yellow. The green set is an open set, because for every point in it, there is a local neighborhood (both 4-neighborhood and 8-neighborhood) that is contained in the digital image (black region). This means that this set is open in both topologies induced by $d_1$ as well as $d_\infty$. The gray set is neither open, nor closed. And the yellow set is closed, because it contains the neighborhoods (both 4-neighborhood, and 8-neighborhood) of all of its elements. So the yellow set is open in both topologies induced by $d_1$ and $d_\infty$. In the middle row, the left subplot shows the boundary of the black region in the topology induced by the 4-neighborhood. In a topological space a point is a boundary point every neighborhood if that point has non-empty intersection with the image and its complement complement. The plot in the right, shows a gray set, that when considered along with the dashed regions, is the closure of the gray set. The plots on the third row are the same as the ones on the second row, with the only difference that we have used the topology induced by the 8-neighborhood.

Note that to construct a topological space there is not need for any metric function (although a metric always induces a topology). This means that we can define neighborhoods as arbitrary as we want, without worrying about the metric functions that can generate those neighborhoods.

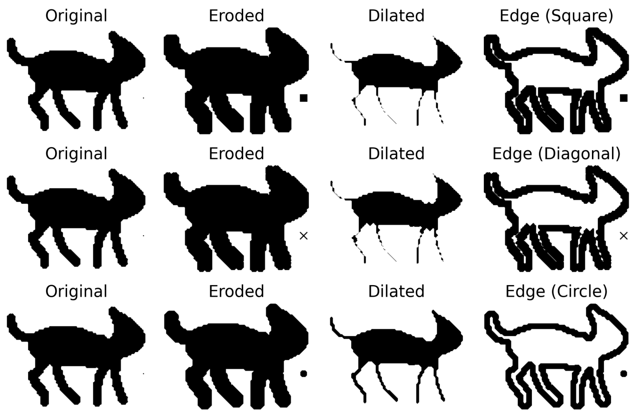
Erosion and dilation of a binary image are intuitively pealing off/adding layers from/to the boundary of regions. For any erosion and dilation operator an structural element is needed, that is often denoted as $H$. The following figure demonstrates erosion and dilation operations with a particular structure element.




Erosion acts like the $\cdot^\circ$ operator (interior) in topology literature in analysis, where as Dilation acts like the $\bar{\cdot}$ operator (closure) in the topology that its local neighborhood basis is successive dilation of the structure element with itself. With this point of view, the duality between erosion and dilation, as expressed in (9.17) in [THE TEXT BOOK], is nothing but a well know fact in analysis

$$(A^\circ)^c = (\overline{A})^\circ$$

where the structural element $H$ should be a hermitian (symmetric in this case) matrix. The following figure shows some common structuring elements and the corresponding edge in that topological space.




A graph representation of a skeletonized binary image can be constructed by first identifying all critical pixels, namely the endpoints (with one neighbor) and junctions (with three or more neighbors), and treating them as graph nodes. The skeleton is then traversed by initiating a search from each unvisited node and exploring connected pixels along the skeleton path using a depth-first approach. These paths typically traverse pixels with exactly two neighbors, corresponding to linear segments. The traversal terminates upon reaching another node (endpoint or junction), at which point the sequence of intermediate pixels is stored as an edge in the graph. This method results in a compact graph where nodes represent structural features and edges represent skeleton paths between them.

**Algorithm 1** Construct Graph from Skeleton Image

**Require:** Skeleton image $S$, set of nodes $N$ (endpoints and junctions)
**Ensure:** Graph $G$ with nodes and edges

```
Initialize empty graph G
Initialize empty set visited
for each node v ∈ N do
    if v ∈ visited then
        continue
    end if
    Add v to visited
    for each neighbor u of v in S do
        path ← [v]
        prev ← v
        curr ← u
        while curr ∉ N do
            Append curr to path
            neighbors ← neighbors of curr in S excluding prev
            if length of neighbors ≠ 1 then
                break
            end if
            prev ← curr
            curr ← neighbors[0]
        end while
        Append curr to path
        Add edge from path[0] to path[-1] to G with attribute pixels=path
    end for
end for
```