1. Understanding the datatypes:
   1. Input data:
      1. Input data is supplied as a stack of laser confocal scans, stored as a series of images, normally in .tif. Using a .tif allow multiple channels (each for a type of biological element)
      2. These images are considered to stack along the z axis, and so each have their own ‘height’
   2. Internal data-structures
      1. Pixels – The simplest data-type, which is used to hold the intensity of a single point within a single image.
         1. Each pixel remembers it’s x,y,z coordinate, as well as the intensity at that point, and which blob2d contains it
      2. Slide – An internal representation of an input image, converts the image to a set of pixels and then blob2ds
         1. Contains information about the input image – Name, sequence number (height), dimensions
      3. Blob2d – Set of connected pixels, which make up a cross section of a 3D object at a specific height
         1. Each blob2d contains some basic information about its pixels including min/max/avg x/y values
         2. Which pixels are ‘edge’ pixels, meaning that they comprise the outside of the cross section
         3. A reference to its ‘parent’ blob2d (if it exists), from which the blob2d was derived
         4. A reference to all children directly derived from this blob2d (these would all reference this blob2d as their parent)
         5. Information about its shape, stored in a set of context bins. Note that this is only used if the blob2d is ‘stitched’
      4. Blob3d – A set of blob2ds at different heights, which make up a 3D model of an isolated object from within the slide stack
         1. Each blob3d is created by combining together all blob2ds that can be reached through their pairings from a seed blob2d
         2. Each blob3d keeps track of all blob2ds which belong to it, as well as calculating new min/max/avg x/y values for its blob2ds as a group
         3. For now each blob3d maintains some redundant information including a reference to all pixels in its blob2ds, as well as their pairings