Blob detection (Segmentation of Regions)

Suppose we want to find regions of pixels that share some common attribute.

Problem is transformed into a graph:

- Noise
- Nodes are pixels
- Edges connect "similar" pixels
- 4 or 8 connectivity

Algorithm: Union Find

- Each pixel assigned an id.
- A table allows each id to have a "parent".
- Long chains of parents possible.
- At end: each root parent for every pixel that is connected will be the same.
```python
resolve(id)
    while (id.parent != id)
        id = id.parent;
    return id;

connect(a, b)
    ar = resolve(a)
    br = resolve(b)
    ar.parent = br;
```

General Blob Structure

1. Iterate over pixels, connect()ing.
2. Iterate over pixels (again):
   - get parent of each pixel
   - accumulate pixel stats
     (use hash table, parent id => stats object)
3. Do something with stats object.