Connected Components

The connected-components algorithm is able to identify regions in an image. This might be useful when finding significant edges or blobs of similar appearance in an image.

Fast c++ implementation at\(^1\) http://xenia.media.mit.edu/~rahimi/connected/

The algorithm

From: http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm

Connected component labeling works by scanning an image, pixel-by-pixel (from top to bottom and left to right) in order to identify connected pixel regions, i.e. regions of adjacent pixels which share the same set of intensity values \(V\). (For a binary image \(V = 1\); however, in a graylevel image \(V\) will take on a range of values, for example: \(V = 51, 52, 53, \ldots, 77, 78, 79, 80\).)

Connected component labeling works on binary or graylevel images and different measures of connectivity are possible. However, for the following we assume binary input images and 8-connectivity. The connected components labeling operator scans the image by moving along a row until it comes to a point \(p\) (where \(p\) denotes the pixel to be labeled at any stage in the scanning process) for which \(V = 1\). When this is true, it examines the four neighbors of \(p\) which have already been encountered in the scan (i.e. the neighbors (i) to the left of \(p\), (ii) above it, and (iii and iv) the two upper diagonal terms). Based on this information, the labeling of \(p\) occurs as follows:

- If all four neighbors are 0, assign a new label to \(p\), else
- if only one neighbor has \(V = 1\), assign its label to \(p\), else
- if one or more of the neighbors have \(V = 1\), assign one of the labels to \(p\) and make a note of the equivalences.

After completing the scan, the equivalent label pairs are sorted into equivalence classes and a unique label is assigned to each class. As a final step, a second scan is made through the image, during which each label is replaced by the label assigned to its equivalence classes. For display, the labels might be different graylevels or colors.

\(^1\)If you use the connected components algorithm for the coursework - you must implement your own (slow) version.
Example of use

From: http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm

Count the number of turkeys in this image:

Threshold:

Connected components labelling:

Next: Size and shape analysis?