

LENGTH/WIDTH RATIO TEST FOR FISH CLASSIFICATION*

Kyle Clarkson
Jason Sedano
Ian Clark

This work is produced by The Connexions Project and licensed under the
Creative Commons Attribution License †

Abstract

This test determines whether a fish is a salmon or a trout, or neither based on the length and width of the fish

One simple way to tell different fish apart is due to their ratio of length to width. For every species of fish, that ratio is essentially the same. One reason salmon are unique in that regard is because during spawning season, they grow wider and thus their length to width ratio becomes much smaller (approximately 2:1 or 3:1). Other fish, such as steelhead trout, which maintain a much more sleek, approximately 3:1 to 4:1 length to width ratio even during spawning season.

This test is done using primarily the Canny Edge Detection method which is performed by Matlab using the command `edge`. It takes the image of the fish and finds an outline of it.

*Version 1.1: Dec 18, 2003 1:52 pm US/Central

†<http://creativecommons.org/licenses/by/1.0>

Canny edge detection Image

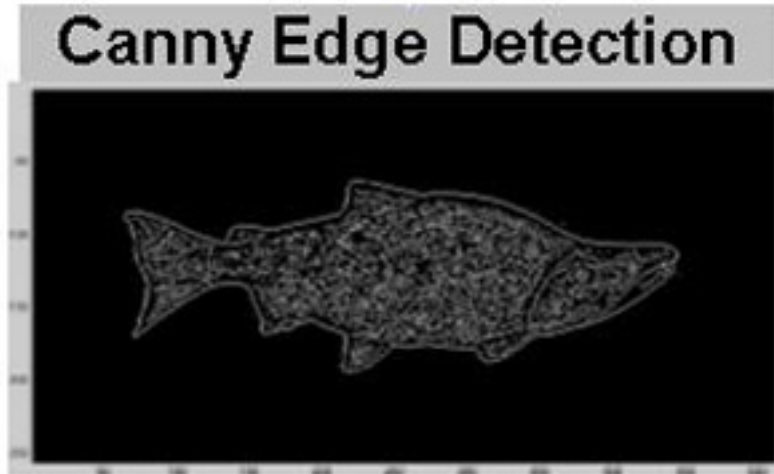


Figure 1: The Fish image after the canny edge detection is run on it.

1 Canny Edge Detection

Image edges contain strong contrasts in intensity. The Canny Edge Detector tries to find only true edges by only selecting localized edge points and only responding once to a single real edge. The Canny method also filters noise with a convolution mask built through Canny's algorithm. The edge detector then uses the gradient of the image and summing along the x direction and y direction to obtain the edge value. Direction of the edge is easily determined by looking at each directions value and seeing which one is larger. However, it is important to test for errors when one direction is zero, so Canny looks at the angle between the x-direction and y-direction. Canny also uses suppression to connect values along the direction of the edge.

NOTE: Canny Method Edge Detection¹

After the edge detection has been run, the image is binary and all the edges are represented by ones. The rows and columns are then each summed up. The algorithm then starts at each edge of the image and moves toward the center until it finds a row with a significant amount of values in it. This signals the edge of the fish closest to that side. Then, through simple calculations, the length, width, and ration can be calculated.

Example 1: Code for Length/Width Test

```
% This function uses canny edge detection to try to find an outline of the
% fish in the grayscale image and then it calculates the length to width
% ratio of the fish.
function [length,width,lwratio] = fishedge(image)

imagegray = rgb2gray(image);
```

¹http://www.pages.drexel.edu/~weg22/can_tut.html

```
imageedge = edge(imagegray,'canny');

l4rowedge = sum(imageedge);
l4coledge = sum(imageedge)';
l4size = size(imageedge);

j = 1;
% Find the top edge of the fish
while l4coledge(j)< 4
    top = j;
    j = j+1;
end
j = 1;
% Find the bottom edge of the fish
while l4coledge(l4size(1)-j+1)< 4
    bottom = l4size(1)-j+1;
    j = j+1;
end

width = bottom - top;

j = 1;
% Find the left edge of the fish
while l4rowedge(j)< 4
    left = j;
    j = j+1;
end
j = 1;
% Find the right edge of the fish
while l4rowedge(l4size(2)-j+1)%lt; 4
    right = l4size(2)-j+1;
    j = j+1;
end

length = right - left;

lwratio = length./width;
```