Sherlock 7 Technical Resource

Teledyne DALSA Incorporated
Industrial Products

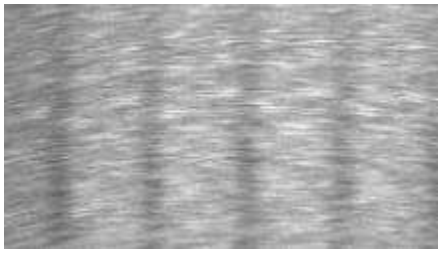
Ben Dawson
Document Revision: 10 August 2012

Chatter Edges Preprocessor

The left image shows an edge with a very slow and diffuse transition. The right image shows the result of applying Chatter Edges with a ***filter size*** parameter of 10.

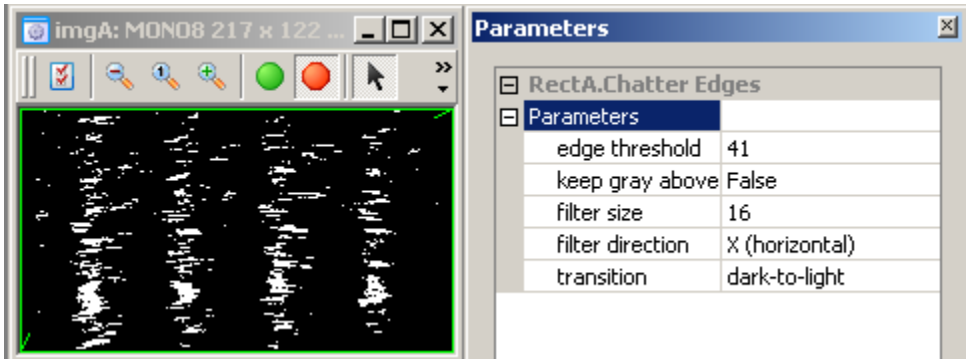
Chatter Edges [Preprocessor]

Chatter Edges was developed to find the diffuse and noisy edges created when a mechanical bearing "chatters". Here is an image chatter marks caused on metal:



Chatter Edges uses a differentiator filter with an adjustable filter kernel size. For example, a ***filter size*** of 4 has a kernel of {-1,0,0,1} and one of size 6 has {-1,0,0,0,0,1}. This filter is scanned across the input image, horizontally, vertically or both (***filter direction*** parameter) to build up the output image.

Applying a ***filter size*** of 16 with a ***filter direction*** of "X (horizontal)" amplifies the low-contrast edges in the above image:



Any output value below ***edge threshold*** is set to 0. Values at and above ***edge threshold*** are either output as full white (pixel value = 255) or, if ***keep gray above*** is True then filter output value above edge threshold are passed unchanged. In the above example, ***keep gray*** above is False so we get a binary (0 and 255) output image.

Filter size (that is, convolution kernel size) values 2 to 255 are possible. Because convolution is a linear process, we can decompose this differentiating filter into two rectangle ("boxcar") filters, with one filter shifted one pixel position with respect to the other. A rectangle filter is a low-pass filter, which acts to remove noise. The -1,….,+1 elements are a differentiator to amplify edges.

Filter direction sets the direction of the filter. An "X (horizontal") ***filter direction*** means that the filter is horizontal. A "Y (vertical") ***filter direction*** uses a vertical filter. In both cases the filter is scanned across the entire input image to build up the output image. If ***filter direction*** is "either", then both X and Y filter directions are applied. The outputs

from the X and Y filters are combined by logical OR if ***keep gray above*** is False, or by the maximum of the X and Y filters (if above ***edge threshold***) if ***keep gray above*** is True.

Transition type sets the edge contrast change to amplify. "Dark-to-light" amplifies edges that transition from dark (low value) pixels to bright (high value) pixels and "light-to-dark" amplifies edges that transition from bright to dark. "either" will amplify both kinds of contrast changes.

This is a preprocessor so it has no outputs.

Sherlock's edge detectors could also be used on these edges, but they output edge positions on a line, rather than scanning the entire image and amplifying low-contrast, high-noise edge transitions.