

Sherlock 7 Technical Resource

Teledyne DALSA Incorporated Industrial Products

Ben Dawson Document Revision: 10 August 2012

Copyright© 2012, Teledyne DALSA, Inc. All Rights Reserved

Edge Crawler Algorithm

Edge Crawler [algorithm]

The Edge Crawler algorithm scans the image (from top, left to bottom right) until it finds an edge (change in image intensity) that is greater than the *min. edge strength* parameter. It then "crawls" along that edge contour, recording the locations and orientations of all points that it finds. The "crawling" stops when either the edge falls below *min. edge strength* or the edge contour closes back on itself.

This back-illuminated image of a hexagonal nut shows the found contours in orange. You can also turn on the Sherlock standard annotation for points (a red '+' for each point), but this annotation is often too dense and obscures the contour locations.



min. contour length sets the minimum length of an edge contour for it to be recorded. Contours shorter than *min. contour length* are discarded. *max. contour length* specifies the maximum length of an edge contour. Contours longer than *max. contour length* are discarded. The default value of *max. contour length* is large so you see most contours when setting up the Investigation. You can then adjust *max. contour length* to remove contours that are too long.

The *use* parameter selects the points to use for the edge contour. If *use* is "all", every contour edge point is used and output. If *use* is "every 2^{nd} point" every second point of the contour is used and output. If *use* is "every 4^{th} point", every fourth point of the contour is used and output. Finally, if *use* is "every 8^{th} point", then every eighth point is used and output.

The next example show how the use parameter is used. The first image on the left shows part of the text "AJ59" obscured by an overlying grid (screen). The image on the right

shows the result of processing this image with the Roughness algorithm to "remove" the overlying grid.



However, the resulting edges of the letters are quite noisy. When we run Edge Crawler with *use* = "every point", we get multiple short contours:



Setting use = "every 2nd point" gives a smoother contour, by only sampling every other point:



Setting use = "every 4th point" gives an even smoother contour by only using every fourth edge:



You can see that the contour has "jumped" across the "J" and between the "5" and "9", so this level of *use* (decimation) is too much.

The last two parameters for Edge Crawler *draw contour* and *selected contour*, control the output of contour points. As mentioned, the default annotation (red +'s) is usually too dense and so prevents seeing the used edge points. This default annotation is automatically turned off and *draw contour* is used to specify which contours are drawn using a point annotations. Your choices are

"none"No point annotations are drawn"selected"Point annotations are drawn for only the selected contour"all"Point annotations are drawn for all contours

You can turn on default annotations by double left-clicking either the "selected contour points" or "all contour points" Outputs and changing the "Display in image window" from [None] to the image used (imgA in this example):

Edit output reading 🛛 🛛
Display in image window:
imgA 💌
Store in variable:
[None]
Track SPC Reset SPC
Value limits for Cp/Cpk calculations
OK Cancel

The *selected contour* parameter selects which contour to show when *draw contour* = selected and the contour to output to Outputs that have "selected" in their name. Contours are indexed starting at 0 and in the order found when scanning the image line-by-line from the top, left to the bottom.

The Edge Crawler has the following Outputs:

- "number of contours" [Double] Number of contours found. If no contours are found, 0 is returned.
- "number of points" [Double] Total number of edge points found in all contours. This gives an approximate measure of the extent of the contours and might tell you, for example, if a hole is missing in a part or if only noise contours have been found.
- "selected contour points" [Point array] Array of contour points for the *selected contour* only. If no contour points are found or there is no contour corresponding to *selected contour*, then this array will be empty.
- "all contour points" [Point array] Array of all contour points. If no contours were found, this array will be empty.
- "angle for selected" [Double] Dominant angle for the "selected contour" only. The "dominant angle" is the most frequent angle of the edges in the selected contour. If no contours or found or there is no contour corresponding to the *selected contour*, then 0 is returned.
- "angle for all" [Double] Dominant angle integrated over all contours. The "dominant angle" is the most frequent angle of the edges summed over all contours found. If not contours are found then 0 is returned, so check "number of contours"