Subject: Re: Thinning algorithm without for loops
Posted by JD Smith on Thu, 09 Aug 2007 16:52:09 GMT
View Forum Message <> Reply to Message

On Thu, 09 Aug 2007 01:46:54 +0000, nathan12343 wrote:

> On Aug 8, 3:33 pm, JD Smith <jdsm...@as.arizona.edu> wrote:
>> [quoted text muted]
>
> JD, really elegant way of doing that. I'm still trying to figure out
> what you did with those two rebin commands to resize the pixels that
> passed step 1,

Just some simple rearrangement. I take the list of kept indices, del=[d1,d2,d3,...], and turn it into an 8xn copy of itself, on its side:

```
 d1 d1 d1 d1 d1 d1 d1 d1 d1
 d2 d2 d2 d2 d2 d2 d2 d2 d2
 d3 d3 d3 d3 d3 d3 d3 d3 d3
 . . . . . . . . .
 . . . . . . . . .
```

To each row of this, I add a custom offset to index the upper right, right, lower right, lower, lower left, left, upper left, upper neighbors. The offset in terms of the single "running" index into the array (of the type returned by WHERE) is -xs for the row above, +xs for the row below, +1 to the right, -1 to the left, etc. I take those eight offsets, call them o1...o8, and simply add them to each row:

```
 o1 o2 o3 o4 o5 o6 o7 o8
 o1 o2 o3 o4 o5 o6 o7 o8
 o1 o2 o3 o4 o5 o6 o7 o8
 . . . . . . . . .
 . . . . . . . . .
```

Add these two together, and you have all 8 neighbors of each of the d's together in a row.

JD