Describing 2D Shapes for Geometric Matching

Robert B. Fisher
School of Informatics
University of Edinburgh

Data Description

Goal: describe parts in same vocabulary of boundary shapes as model

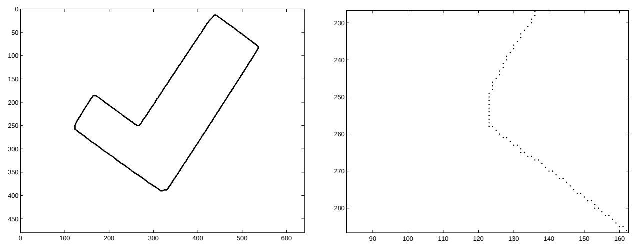
Assume a binary image of the part

Need to find pixels that lie on the boundary

Boundary Finding

1) Get points that lie on boundary:

```
[r,c] = find( bwperim(Image,4) == 1 )
```

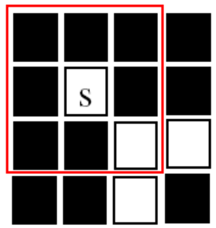


2) Remove any spurs on boundary, track and segment

```
[sr,sc] = removespurs(r,c,H,W);
[tr,tc] = boundarytrack(sr,sc);
[cr,cc] = findcorners(tr,tc);
```


Removing Dangling Spurs

Spur: any boundary pixel with only 1 neighbor inside a 3x3 neighborhood



```
changed=1;
while changed==1
  changed = 0;
  [sr,sc] = find(work==1);     % work: boundary pixels
  for i = 1 : length(sr)       % check each boundary point
    neigh = work(sr(i)-1:sr(i)+1,sc(i)-1:sc(i)+1);
    count=sum(sum(neigh));
    if count < 3               % only point and at most
      work(sr(i),sc(i)) = 0;   % 1 neighbor so remove it
      changed=1;
```

Trailing `ends` omitted.

Removing Unnecessary Boundary Pixels

Find unnecessary corners:

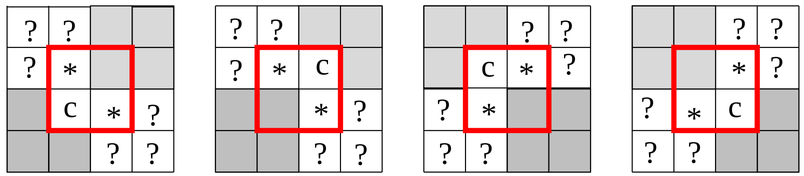
* - boundary point to keep

c - boundary point to remove

? - boundary point thru here somehow

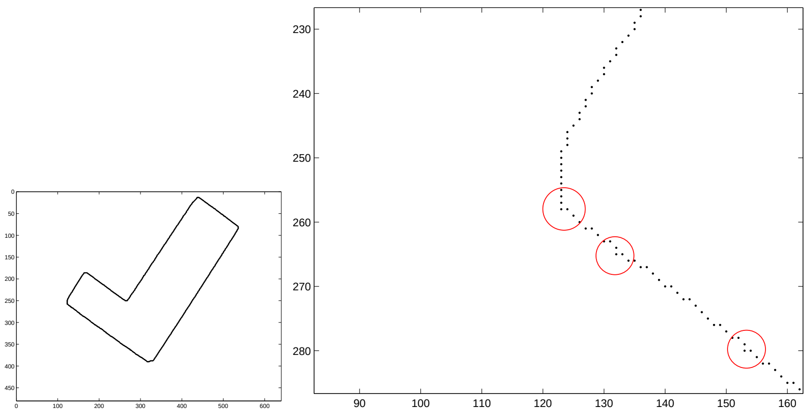
shaded box - interior or exterior pixel

thick red box - pixel neighbourhood inspected

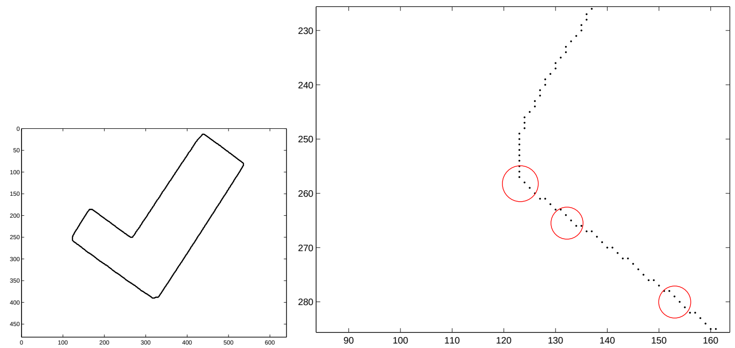


Boundary Cleaning Results

Raw boundary:



Cleaned boundary:



Getting a Consecutive Boundary Track

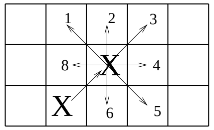
TRACK TO FIRST
UNTRACKED BOUNDARY
PIXEL ENCOUNTERED
AS i GOES 1...7

NEXT DIRECTIONS

1	2	3
8	x	4
7	6	5

NEXT = (LAST + 3 + i) MOD 8 + 1

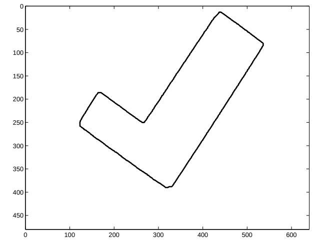
EXAMPLE TRACKING



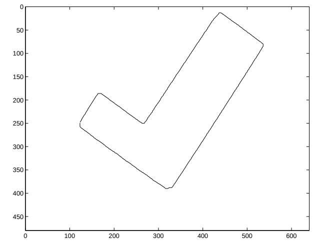
LAST MOVE = 3
NEXT MOVE = 8,1,2,3,4,5,6

Tracking Results

Despurred boundary (unorganized point set):



Tracked boundary (consecutive point set):



What Have We Learned?

Introduction to

- Getting a boundary from a binary image
- Cleaning that boundary up using morpholgical operations
- Making a consecutive list of points