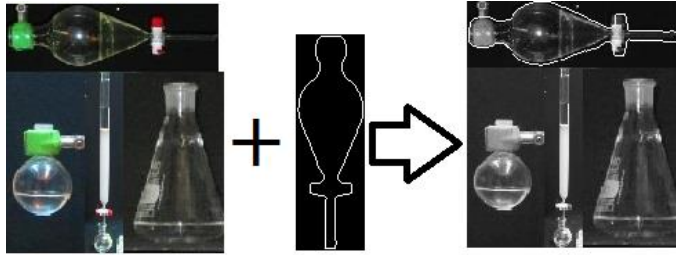


Generalized hough transform with rotation



Find an object that fit Template Itm in image Is . The orientation of the template and the object in the image does not have to be the same as that as the template. The template Itm is matched to the image Is in various of rotations and the best match is chosen. The function use Generalized Hough transforms to match template to the image.

MAIN_find_object_in_image (Itm, Is) is the main function. The output is the boundary and location and size of the template in the image with the object boundary marked on it.

Input:

Is : Color image with the object to be found.



Itm : A template of the object to be found. The template is a binary image with the boundary of the template marked 1(white) and all the rest of the pixels marked 0.



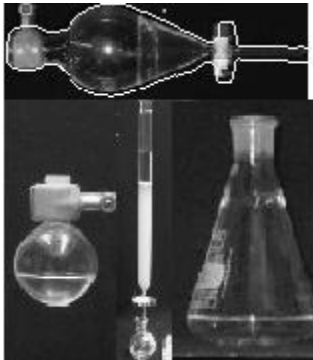
Template of object could be created by extracting the object boundary in image with uniform background, this could be done (for symmetric objects) using the code at:

<http://www.mathworks.com/matlabcentral/fileexchange/46887-find-boundary-of-symmetric-object-in-image>

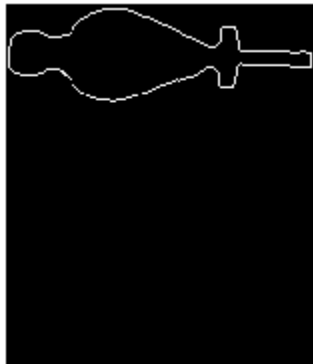
Additional optional input parameters are given in the next page.

Output

Ismarked: The image with the template marked upon it in the location of and rotation angle of the best match.



Iborders: Binary image of the borders of the template in for the best match (borders of the found object).



Xbest,Ybest: Location on the image (in pixels) where the template best match the image for the upper left corner of the template

ItmRot: The angle of rotation of the template (*Itm*) that gave best results.

BestScore: Score of the best match found in the scan (the score of the output).

How to use

Run the function: **MAIN_find_object_in_image**(*Is,Itm*) With the above parameters.

The output is the parameters

[*Ismarked,Iborders,Ybest,Xbest, ItmRot, BestScore*]

Described above.

Other functions:

Generalized_hough_transform(*Is*, *Itm*)

Find template *Itm* in greyscale image *Is* using generalize hough transform

Return the *x,y* coordinates of the best match.

Also return the *score* of the best match.

(there is no resizing or rotating of the template or image during scan).

INPUT

Is: Greyscale picture where the template *Itm* should be found

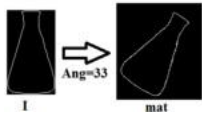
Itm: Binary edge image of the template with edges marked 1 and the rest 0

OUTPUT

x,y: Coordinates of template *Itm* in image *Is* for the best match (Location the edge (point [1,1]) of the template *Itm* in *Is* for the match with the highest score).

score: Score of the best match.

Rotate_binary_edge_image(*I*,*Ang*)



Rotate edge image (*I*) in (*Ang*) degrees

The rotated output image will also be a binary edge image.

The connectivity/topology of all edges/curves in the input image (*I*) will be maintained and the line thickness of the curves in the output image (*mat*) will remain 1 pixel.

The center of rotation is the center of the image

The dimensions of the output image (*mat*) will be different from the input image and will be set such that the rotated image is fully within the image frame.

Input

***I*:** Binary edge image (logical type) consist of lines and curves with a thickness of one pixels (such as curves, contour line, template, or edge images)

***Ang*:** Rotation angle of the image in Degrees

Output

***mat*:** Rotated version of the input image (*I*), also binary edge image, the connectivity/topology of the edges/curves in input image (*I*) is maintained and also the line thickness remain one pixel.