Automatic Handwritten Flowchart Conversion

Digital Image Processing Project

Sanidhya Singal, Aditya Adhikary

https://github.com/sayhitosandy/Flowchart_Converter

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Progress till mid evaluation

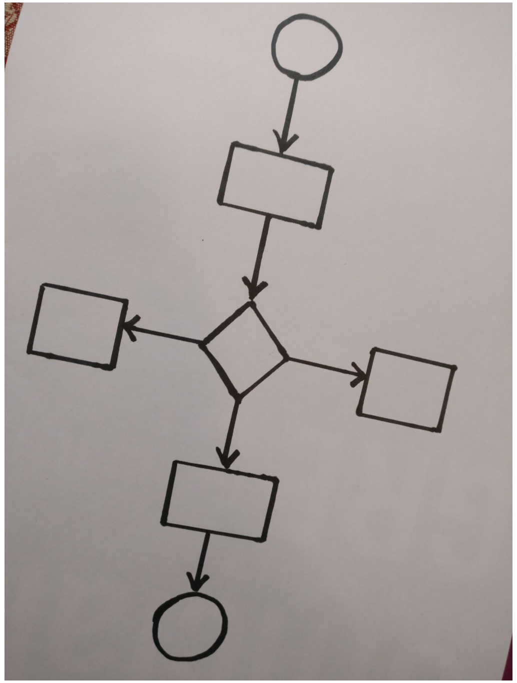
Dataset: Handwritten flowcharts consisting of rectangles, diamonds, circles. Arrows are straight. No text inside the shapes for now.

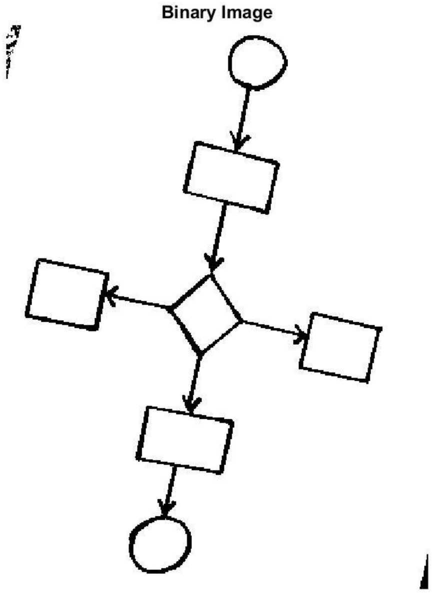
Image Processing Techniques:

1. Take the jpg image from a mobile camera, as an RGB, and convert to grayscale.
2. Resize so that the image aspect ratio is maintained.
3. Perform binarization using a locally adaptive threshold, to accommodate for different lighting conditions in different areas (The foreground is darker than the background in this case).
4. Bitwise invert the image.

Progress till mid evaluation

5. Noise removal - Small areas of color which are incorrectly created during binarization are removed by finding the connected components, calculating their areas, labeling them, and removing ones with areas less than a certain threshold.
6. Fill the closed areas in the image, and find the edges using this filled image.
7. Find the Hough transform of the edge-highlighted image, and find out the various angles in the image. The most commonly occuring angle is calculated from this, and the image rotated to restore its correct orientation.
8. The filled, rotated image is then decomposed to first remove the arrows from the image, then retrieve the arrows by subtracting this image from the filled, rotated image. Noise removal of shapes with small areas are done wherever required.




Binary Image

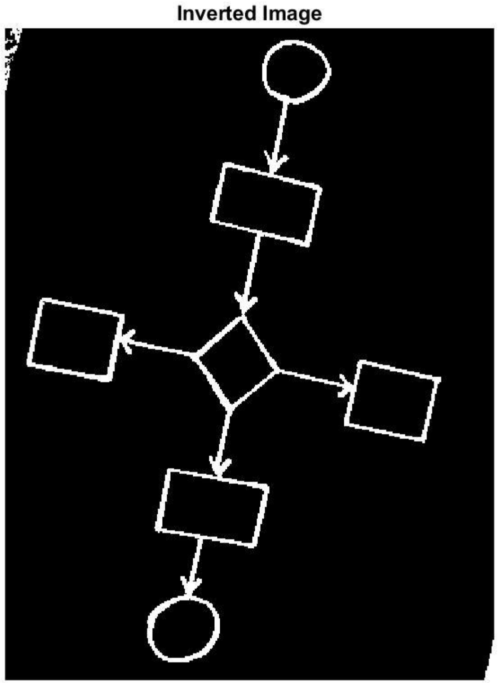
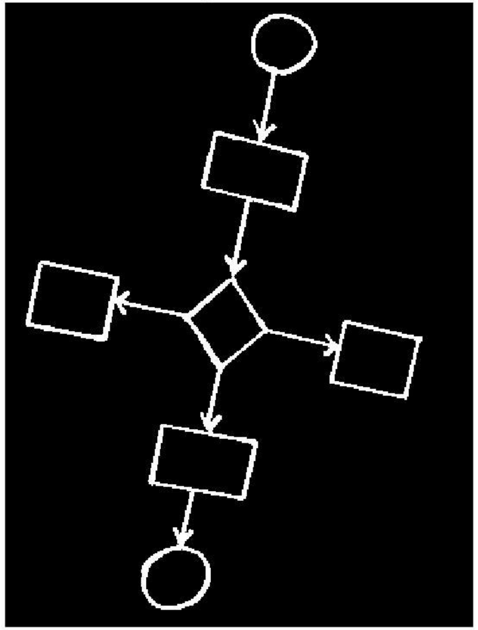
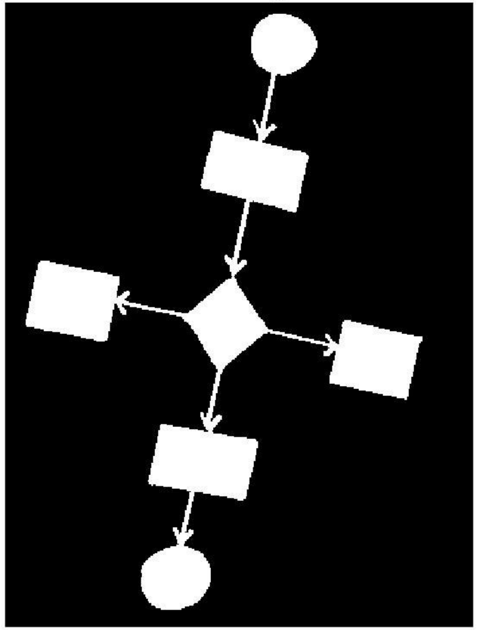

Inverted Image

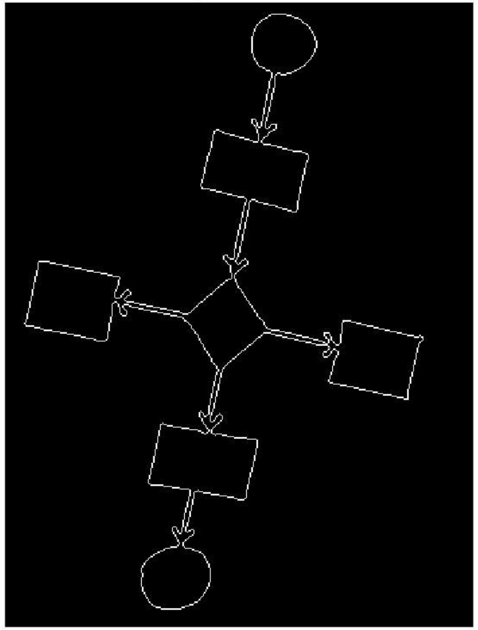
Image Cleaned



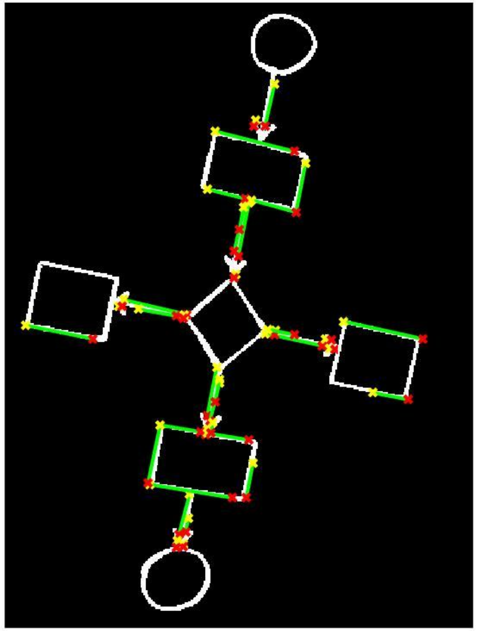
Filled Image



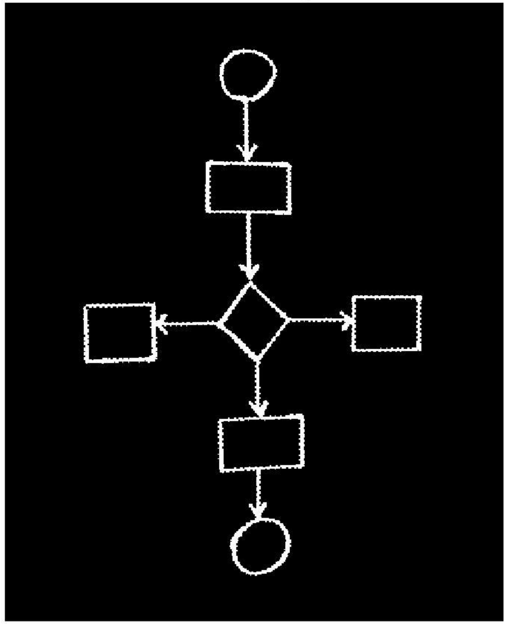
Edge Detection



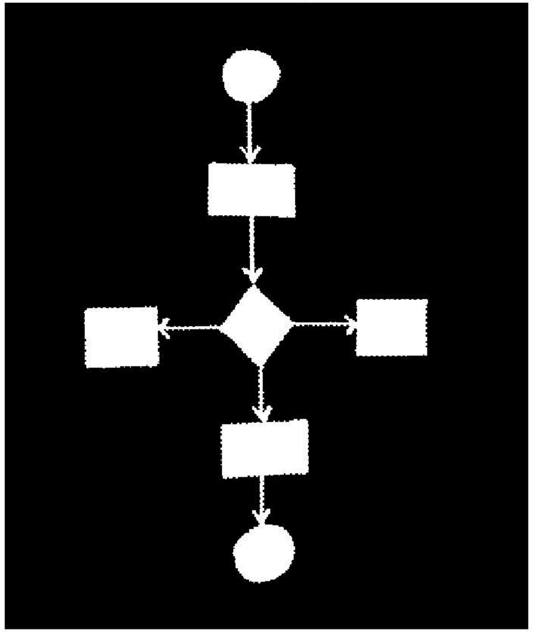
Detected Lines



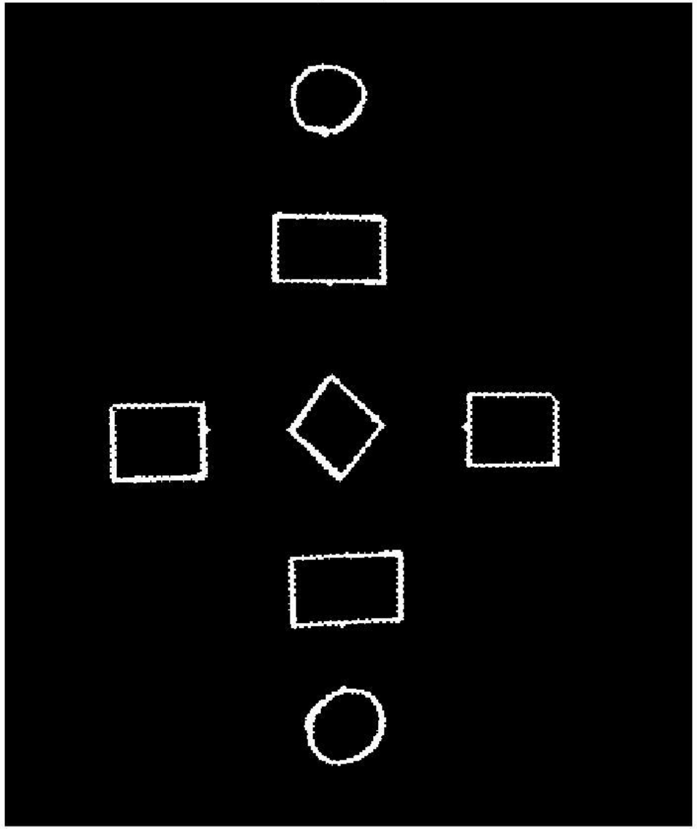
Rotated Image



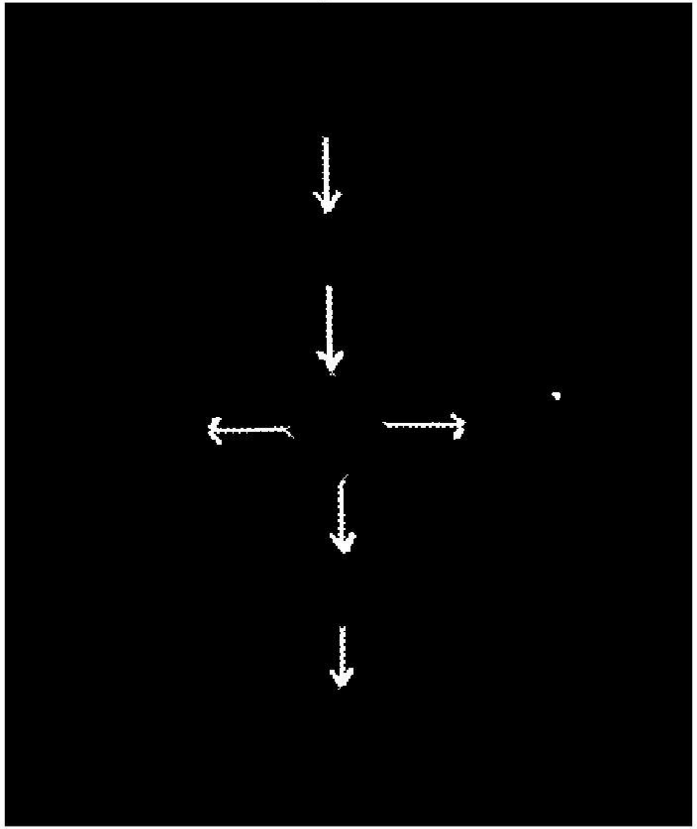
Filled Image



Only Shapes



Only Arrows

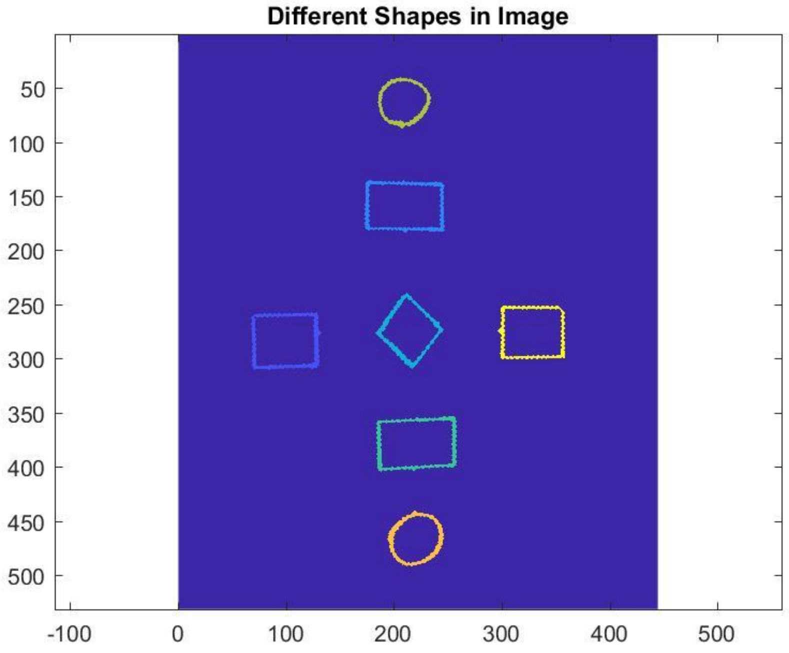


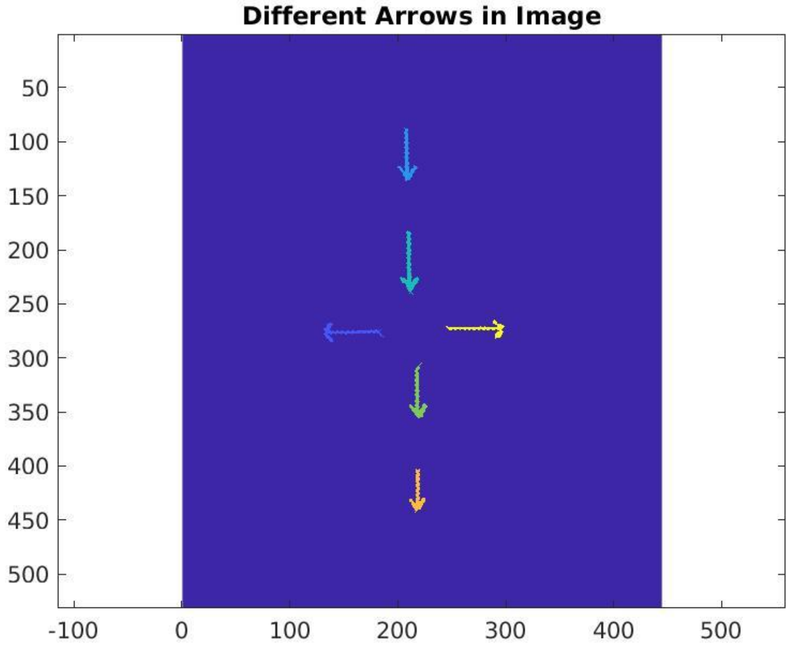
Progress

9. Shape Recognition: For the 3 different types of shapes, Circles, Rectangles, Diamonds, we used the properties of the shapes. We calculate the boundary boxes (smallest rectangular region enclosing each shape), and using the ratio of areas of the shape inside to the boundary box area, we distinguish rectangles from diamonds. Circles are recognized using a ratio of the perimeter and areas.
10. Arrow Direction: For each arrow, we use the information of the centroid of its boundary box, and the centre. The head of the arrow is oriented towards the centre of the edge which is closer to the centroid than the centre.
11. The arrow heads are then joined (made equal) to the closest anchor points for the different shapes: For rectangles, the four centres of the edges; and for diamonds, the four corner points.

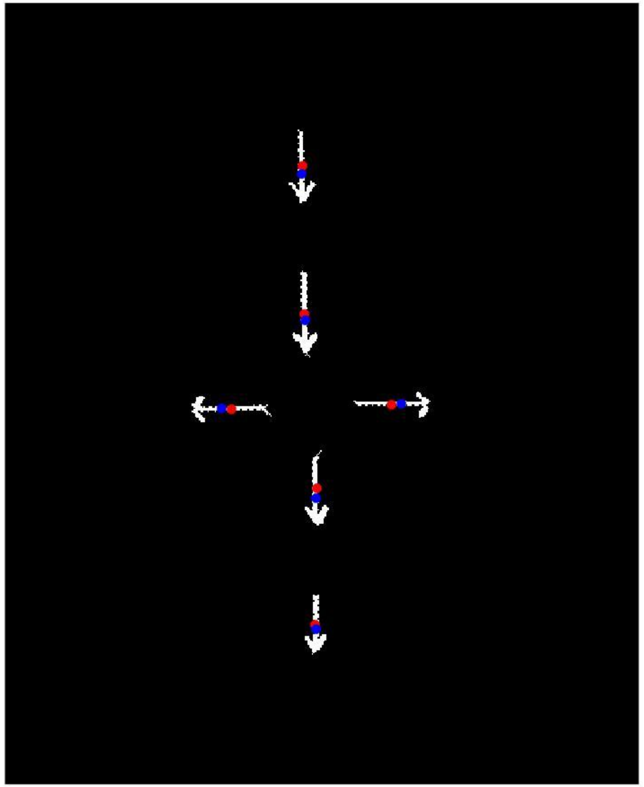
Progress

12. Reconstruction of arrows and shapes:
    - Using the information about head and tail coordinates of the arrows, we used this to reconstruct arrows in their proper positions
    - Using different tools of MATLAB, the different shapes are reconstructed.
    - The final flowchart can now be used in place of the handwritten version. It is comparatively much neater, easily portable, and can be converted into objects for extending usage.

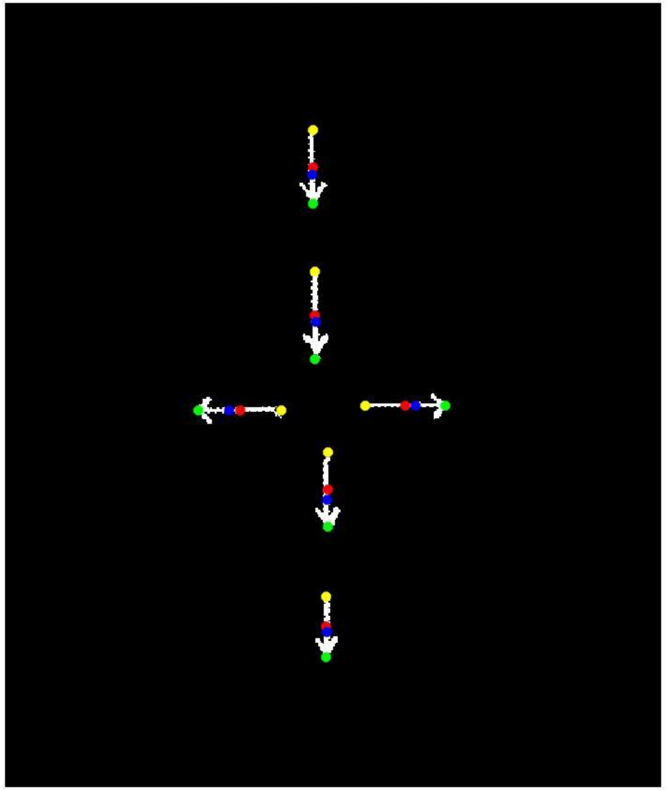

Different Shapes in Image
50
100
150
200
250
300
350
400
450
500
-100
0
100
200
300
400
500


Different Arrows in Image
50
100
150
200
250
300
350
400
450
500
-100
0
100
200
300
400
500

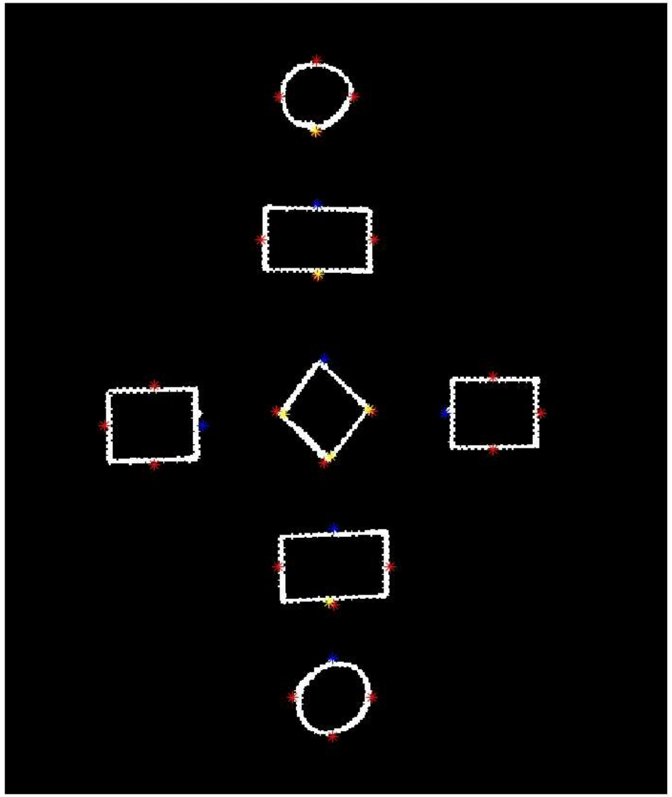
Arrow Centres and Centroids



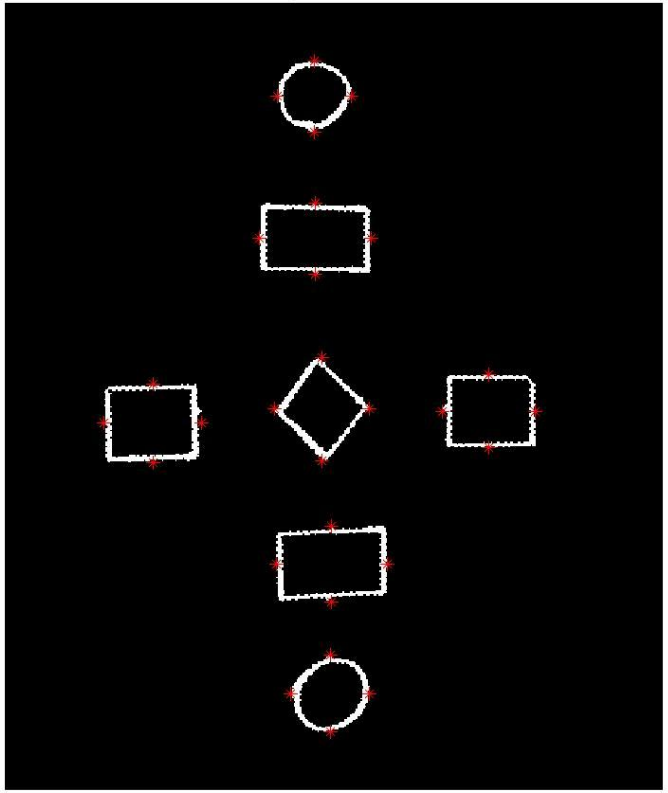
Arrows Heads and Tails



Final Arrows Heads and Tails



Shapes Anchors



Output Flowchart