MCQ Answer Script Marking (OCR)

What will be presented

- Project brief and objectives
- Methodology
- Results
- Conclusion

The project and its objectives

Project brief

- Python and OpenCV based project.
- Program accepts a picture of an answer script.
- Detect rectangles and then marks on the selected areas.
- Gives the marks out of 100%.
- Marks the correct and wrong answers on the image.

Objectives

- Try to implement a OpenCV based real-time MCQ marking system design
- Reduce the time takes to markup the answer scripts
- Reduce the mistakes that can happen when marking manually
- Can use in any laptop / device that can run python

METHODOLOGY

Common operators

- Grayscale conversion, resizing the image, and imshow (show output image)

Canny operator

- imgCanny = cv2.Canny(imgBlur, 10, 70)
- Canny operator used to identify the edges main parts of the image.
- Because Canny uses hysteresis thresholding and non-maximum suppression, the resulting edges are usually thinner and more linked.

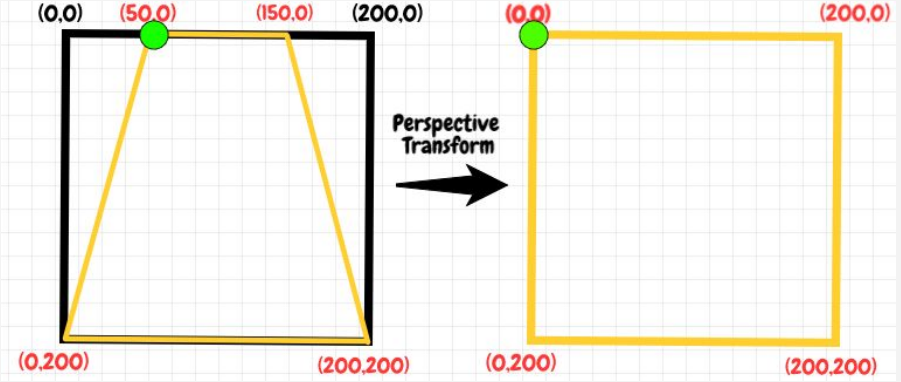
METHODOLOGY

Find contours

- contours, hierarchy = cv2.findContours()
- Find all the available contours
- rectCon = tools.rectContour(contours)
- Find and get array of all the rectangular contours

METHODOLOGY

Warp transformation



- cv2.getPerspectiveTransform()
- cv2.warpPerspective()

- cv2.getPerspectiveTransform() -> use to get transformation matrix
- cv2.warpPerspective() -> apply warp transformation
- Warp transformation use to change the image shape

METHODOLOGY

Threshold

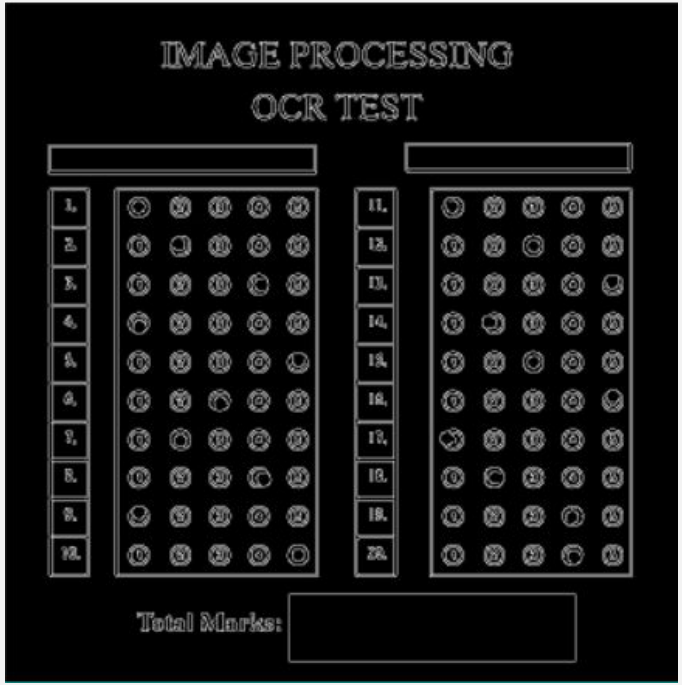
- cv2.threshold()
- Apply threshold on selected area of the images.
- Make it easy to identify the OCR marks on the image

Add weighted

- cv2.addWeighted()
- Use to multiply image with original image

Results

Canny Operator

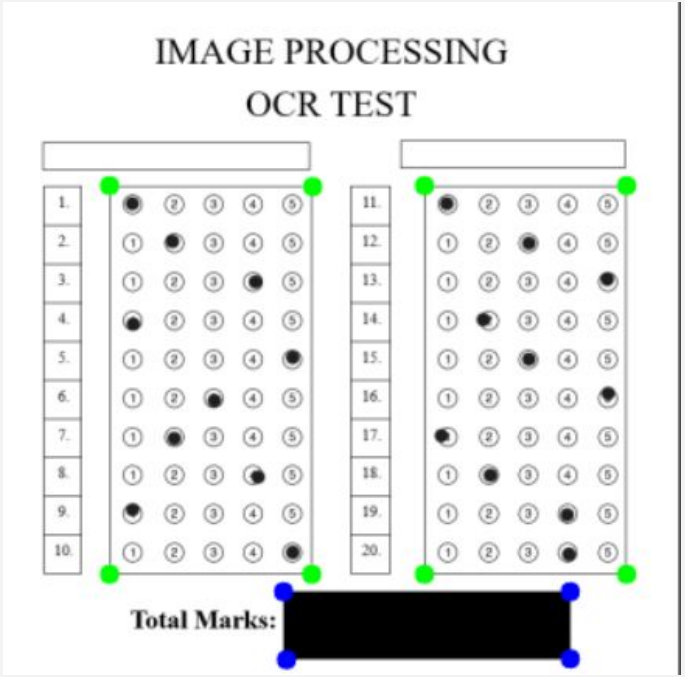


Results

Contour Detection

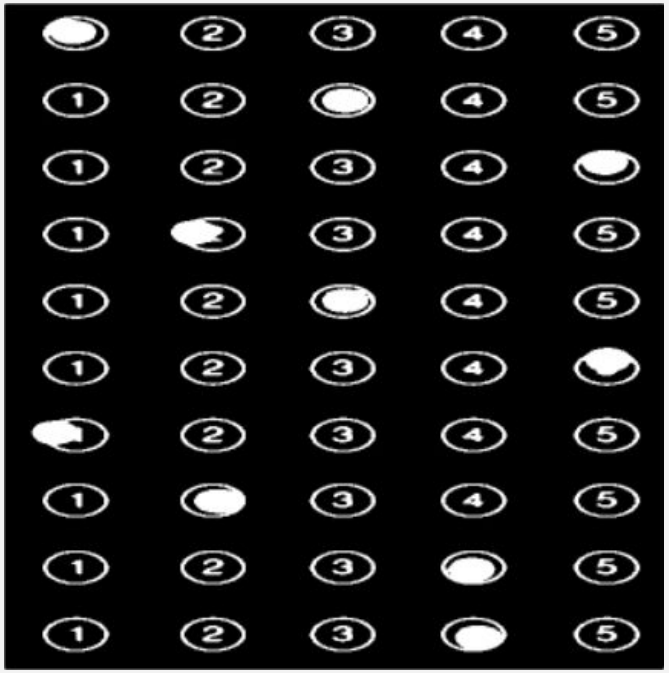
Results

Selecting the main rectangles



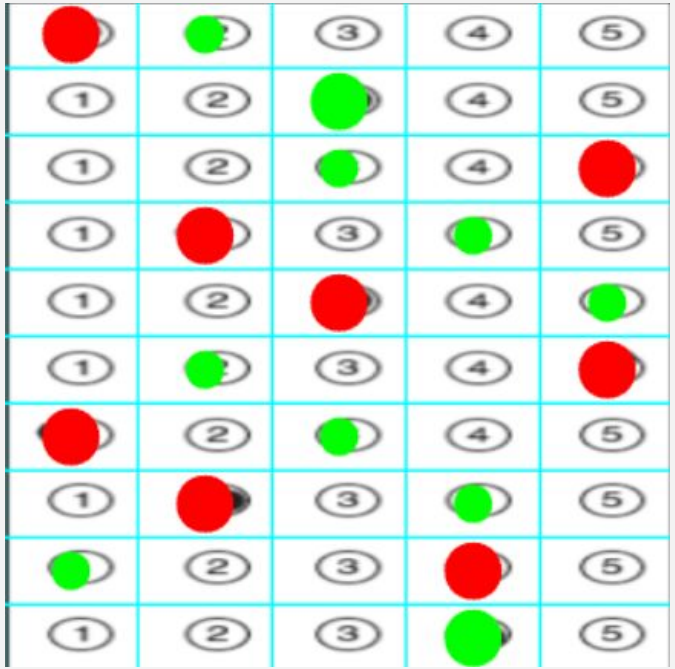
Results

Warp and threshold



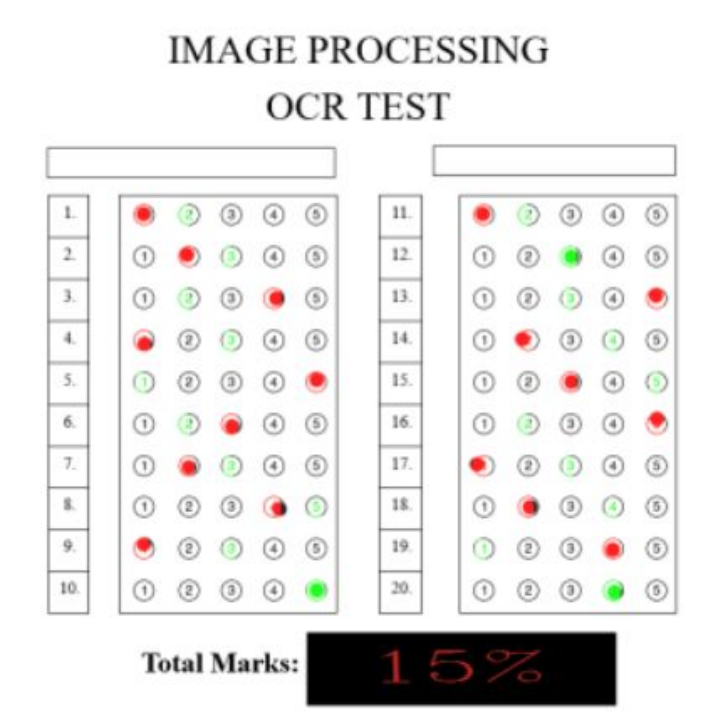
Results

Grid and mark answers



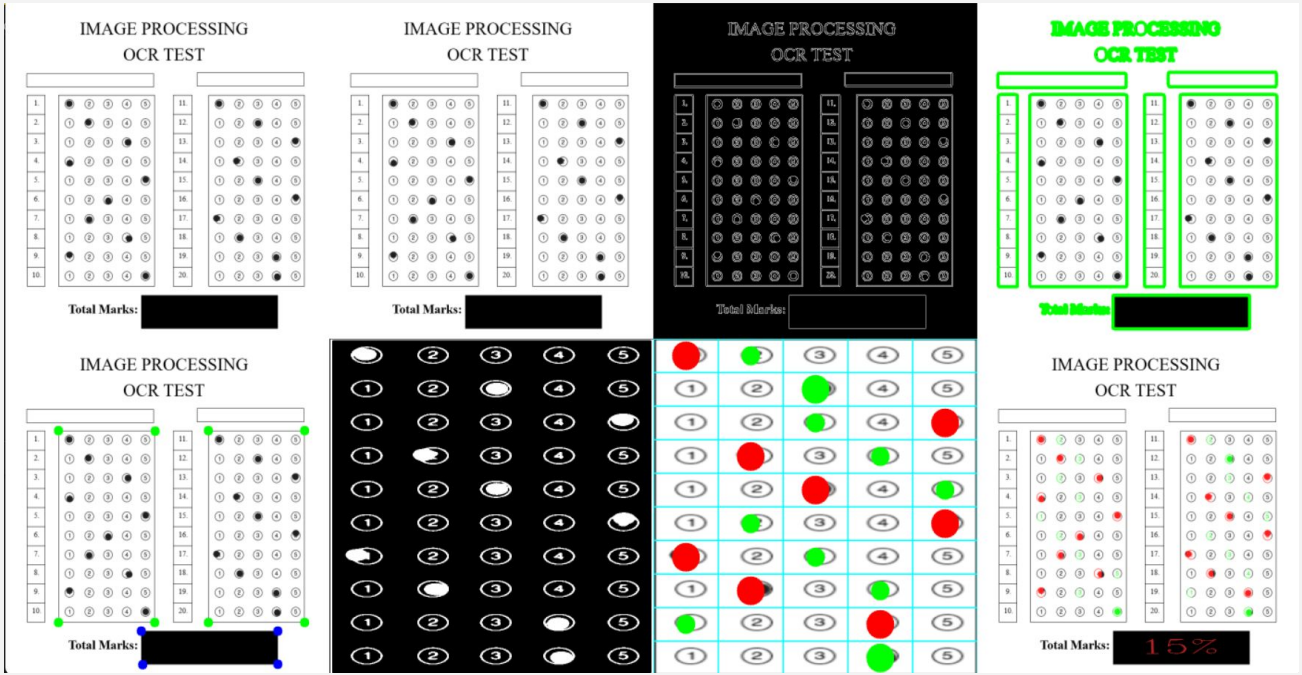
Results

Final result



Conclusion

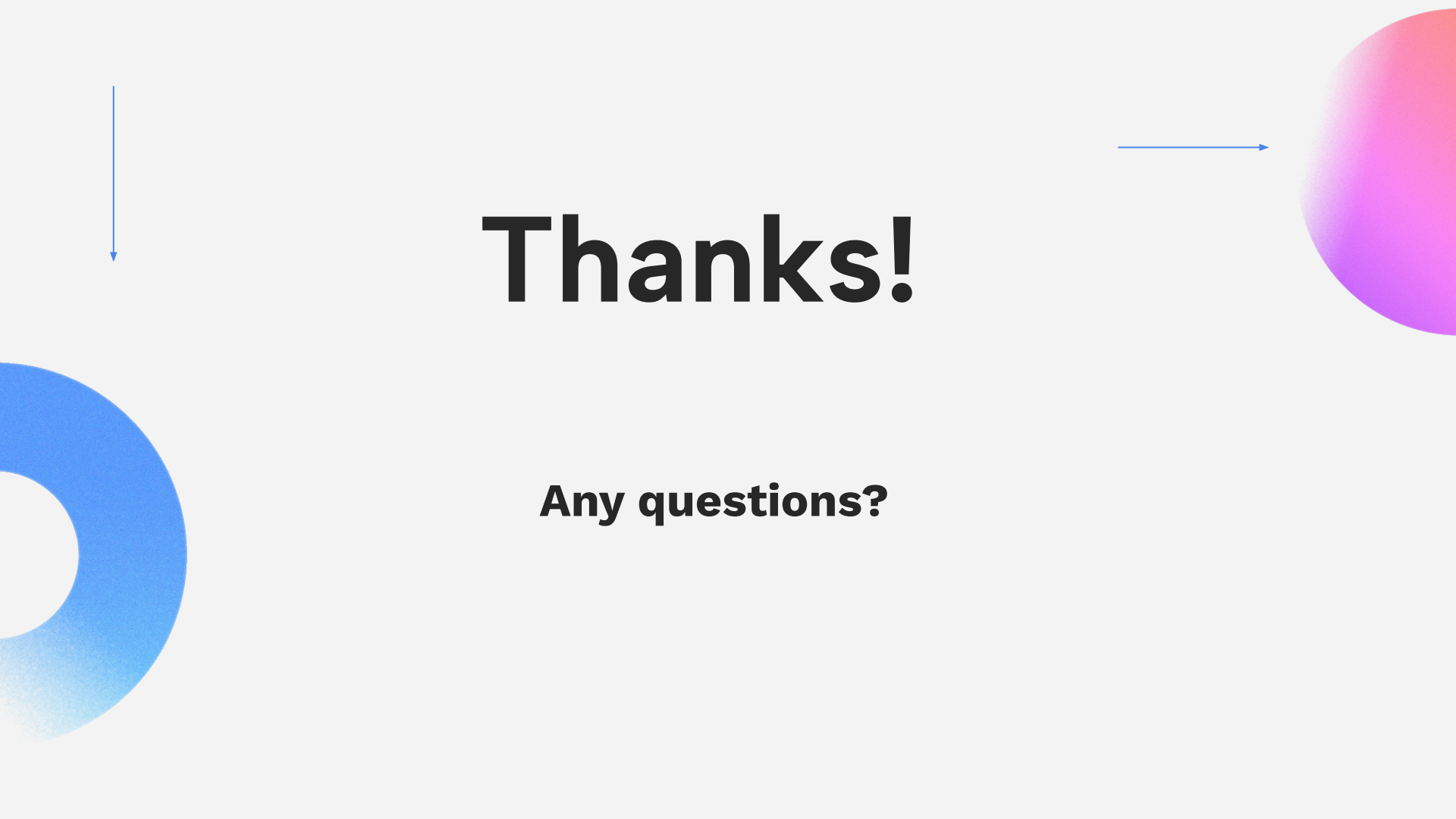
Final result



Conclusion

Challenges

- When the image not properly scanned, it’s difficult to identify contours
- Moving from MATLAB to python with openCV.


Thanks!
Any questions?