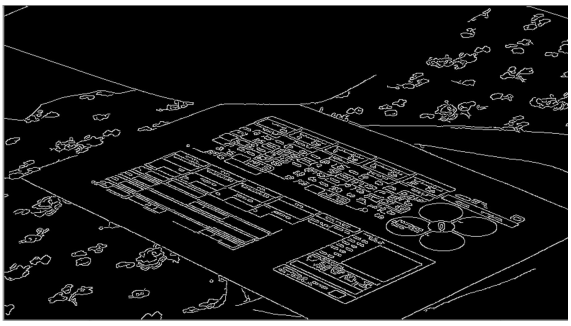
COL783 ASSIGNMENT-4
ROBUST FORMS PROCESSING

Submitted By- Vikram Kumar and Divya Garg

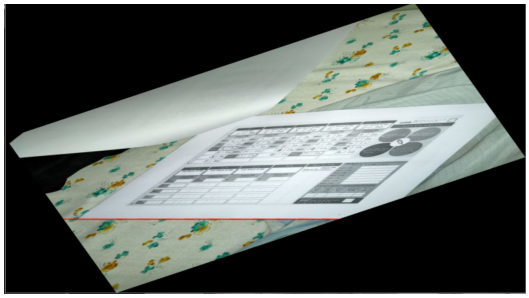
IMAGE ALIGNMENT

a. Image was resized to a standard size
b. It was converted into a binary image to detect edges using Canny Edge Detection
c. Once the edges were detected, hough transform was used to calculate the angle of rotation and then built-in function was used to rotate the image using perspective transformation


Resized


Canny Edges (params are varied for thresholding)


Hough lines

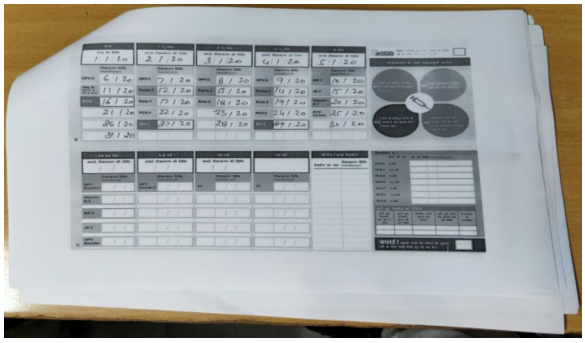
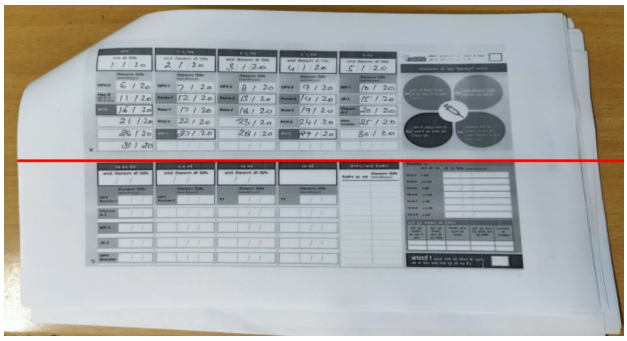

Rotated image


For perspective transformation

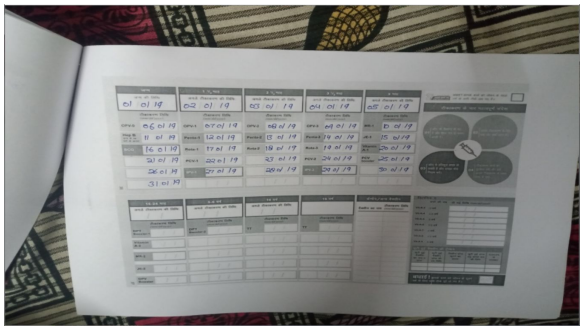
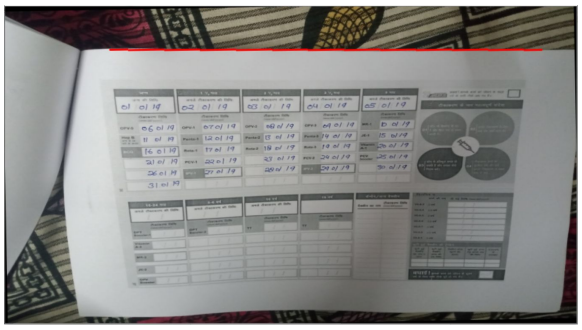
Image	Angle of Rotation for alignment	Canny Edge Parameters (min, max, aperture)	Hough Params (Rho, length accuracy)
booklets/3.jpg	0	30, 150, 3	
booklets/7.jpg	3.0349316553235823	0, 255, 3	
booklets/8.jpg	-0.926762659901782	0, 255, 3	
booklets/12.jpg	0	30, 150, 3	
printouts/2.jpg	0	0, 255, 3	
printouts/9.jpg	1.010997052330402	0, 255, 3	
printouts/18.jpg	25.9996506538092	0, 255, 3	
printouts/20.jpg	16.941542392836567	0, 255, 3	
printouts/44.jpg	3.9654874256263213	30, 150, 3	
printouts/48.jpg	6.087008331931995	(30, 150, 3)	(1, 5)
scanned/9.jpg	2.021994104660804	0, 255, 3	
scanned/10.jpg	1.031212669887953	5, 20, 3	
scanned/11.jpg	0	5, 20, 3	
scanned/12.jpg	0	5, 20, 3	

ALIGNED IMAGES:

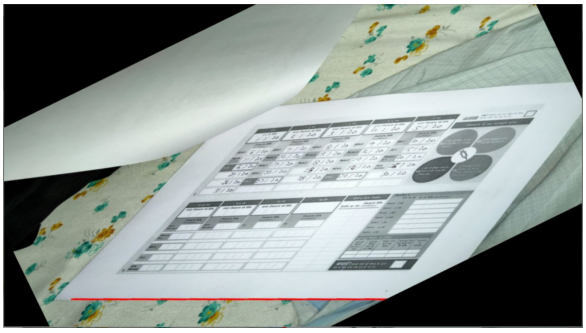
Printouts\2.jpg (Angle of rotation = 0), Canny Edge params = 0, 255

Printouts\9.jpg (Angle of rotation = 1.010997052330402) Canny Edge params = 0, 255

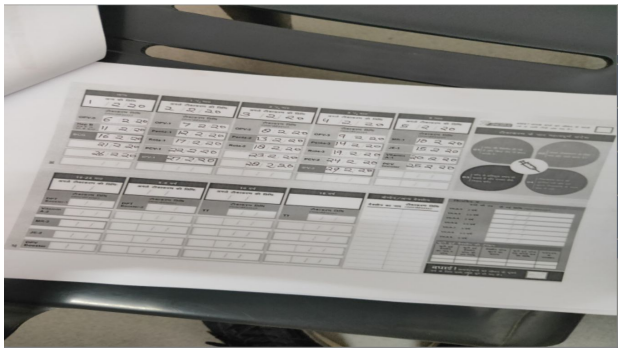
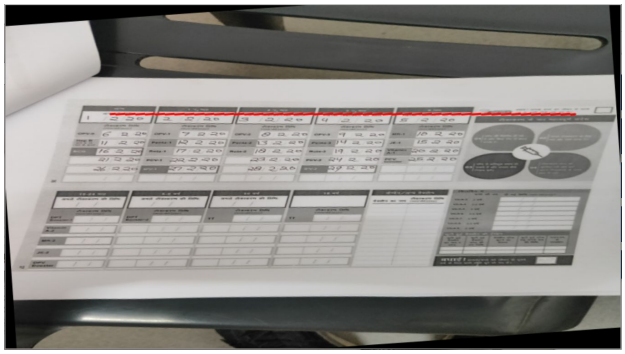



Printouts/18.jpg (Angle of rotation = 25.99) Canny Edge params = 0, 255, rho = 1, accuracy=100

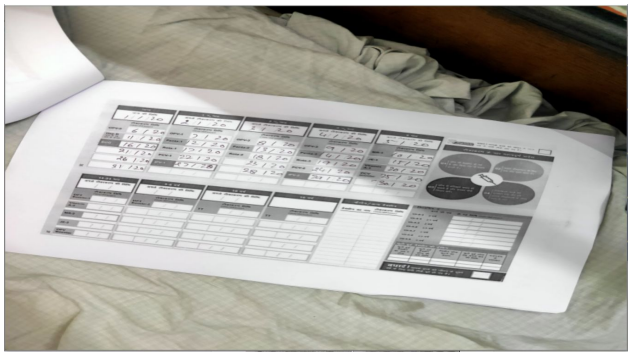
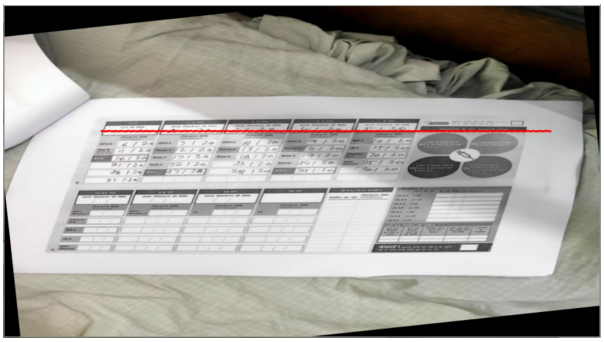



Printouts/20.jpg (Angle of rotation = 16.941542392836567) Canny Edge params = 0, 255

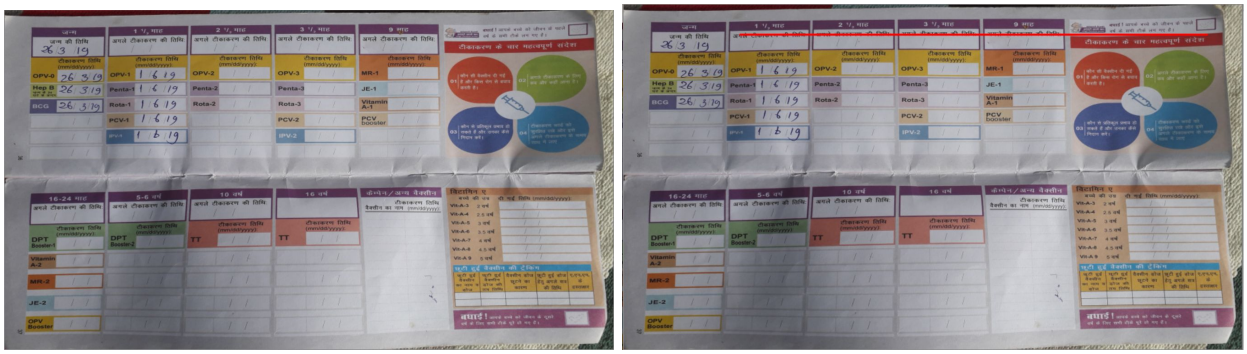

Printouts/44.jpg (Angle of rotation = 3.9596658293138676) Canny Edge params = 30,150

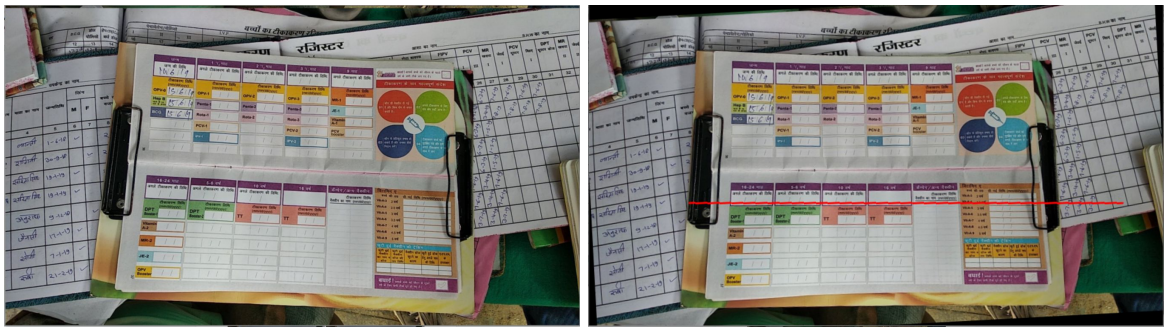
Printouts/48.jpg (Angle of rotation = 6.079588957838487) Canny Edge params = 30,150

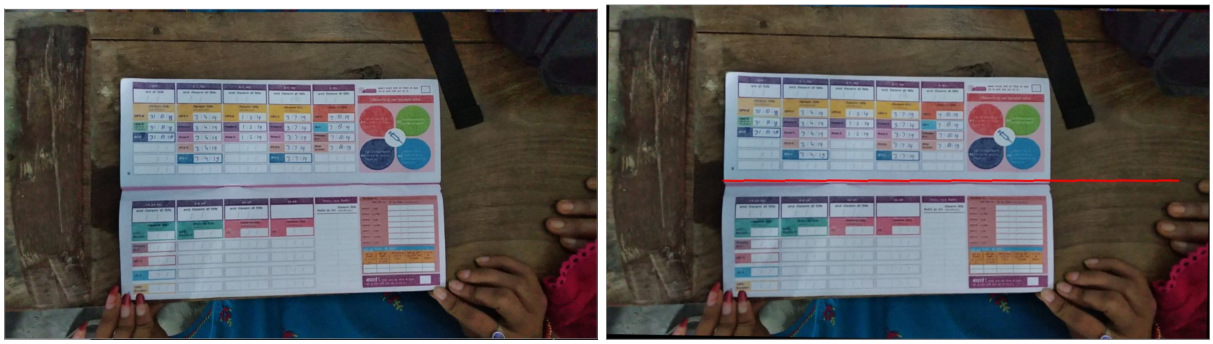
Booklets/3.jpg (Angle of rotation = 0) Canny Edge params = 30,150



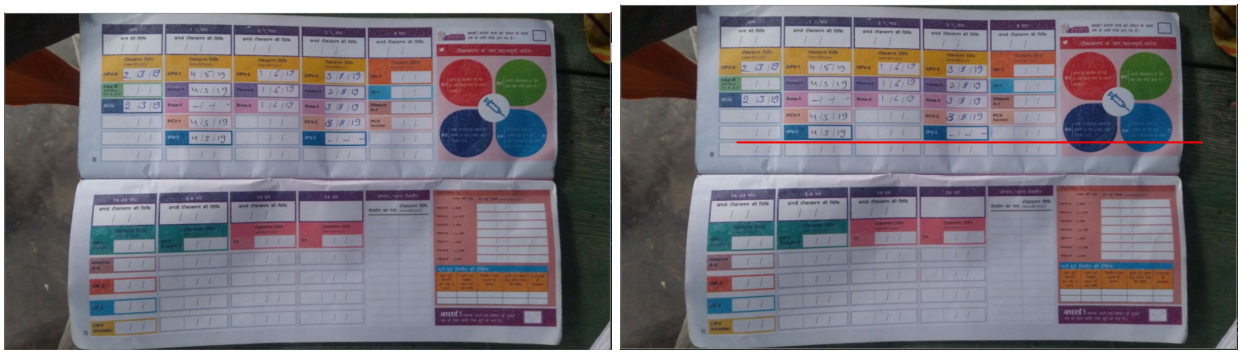
Booklets/7.jpg (Angle of rotation = 3.0349316553235823) Canny Edge params = 0, 255



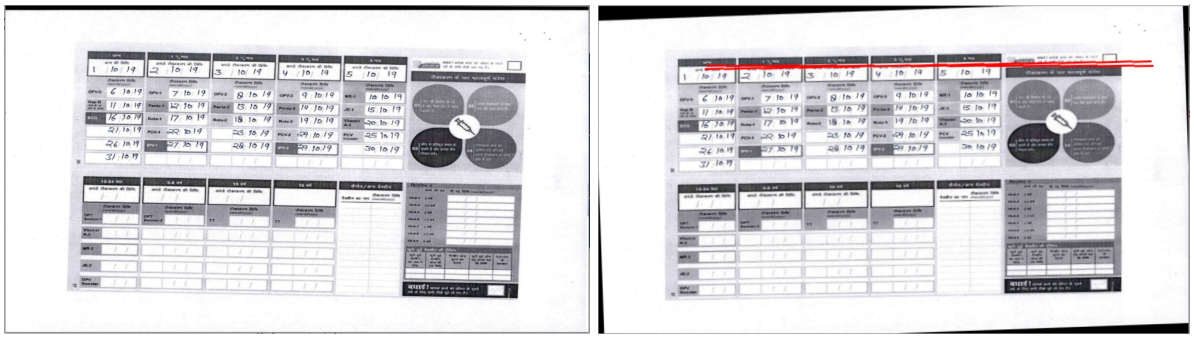
Booklets/8.jpg (Angle of rotation = -0.926762659901782) Canny Edge params = 0, 255



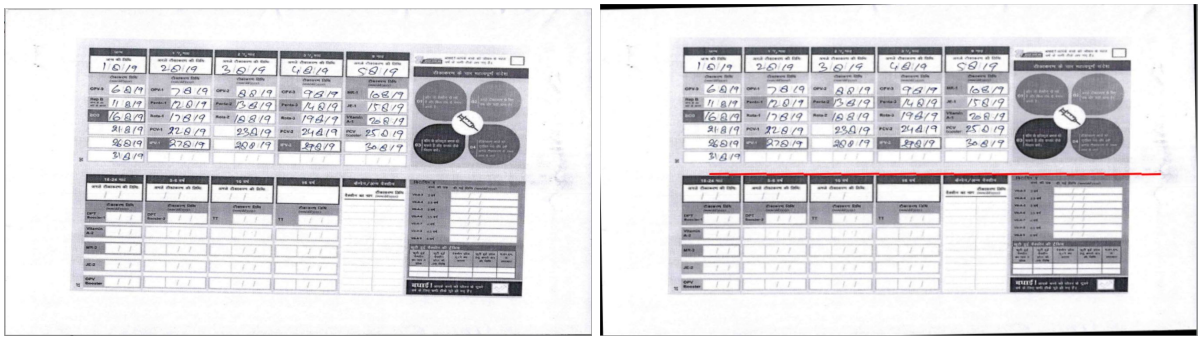
Booklets/12.jpg (Angle of rotation = 0) Canny Edge params = 30,150



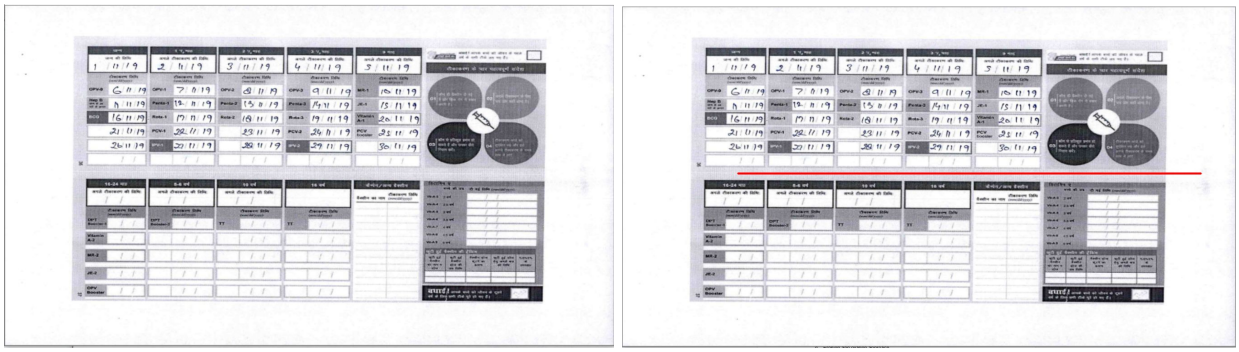
Scanned/9.jpg (Angle of rotation = 2.021994104660804) Canny Edge params = 0, 255



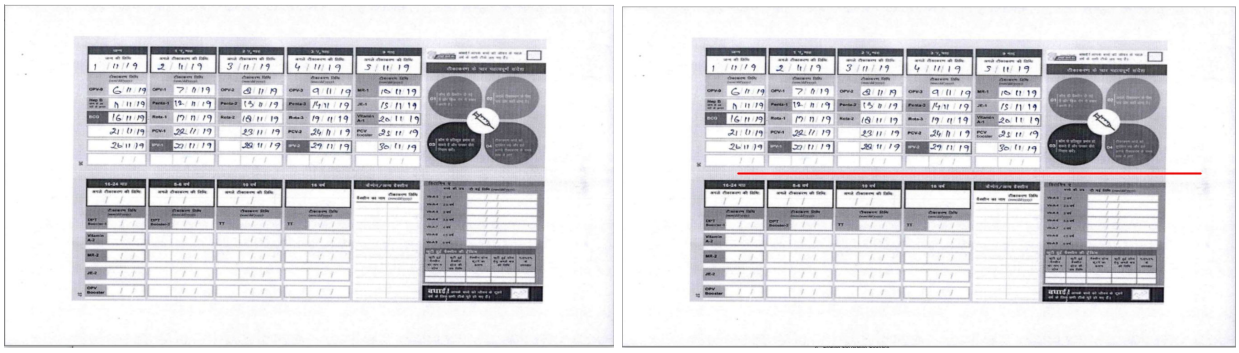
Scanned/10.jpg (Angle of rotation = 1.010997052330402) Canny Edge params = 5, 20



Scanned/11.jpg (Angle of rotation = 0) Canny Edge params = 5, 20

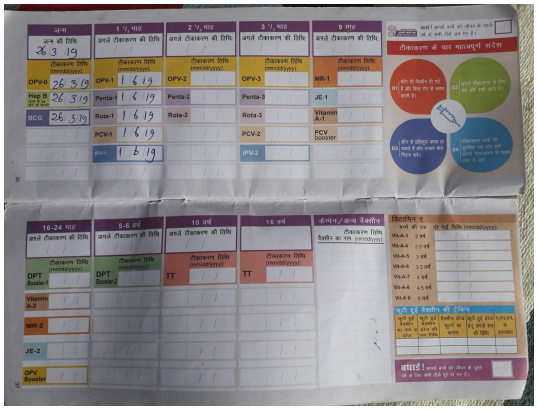
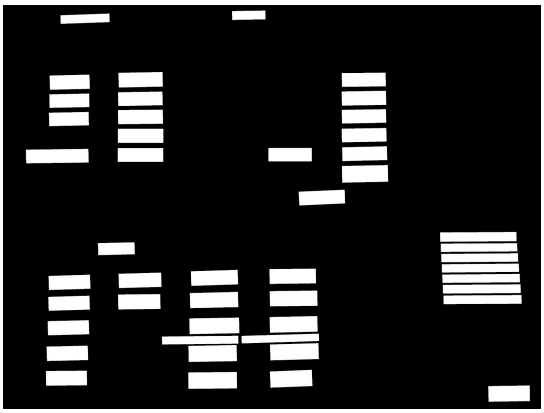


Scanned/12.jpg (Angle of rotation = 0) Canny Edge params = 5, 20

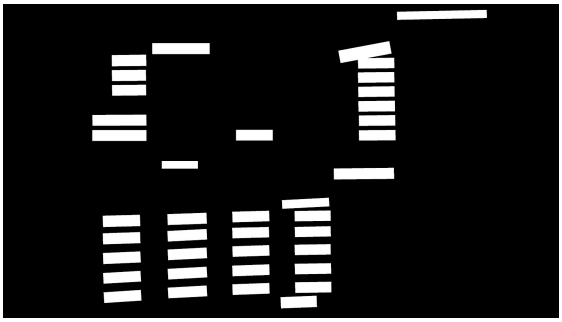


Failed approaches included edge detection using normal blur and Sobel Operators

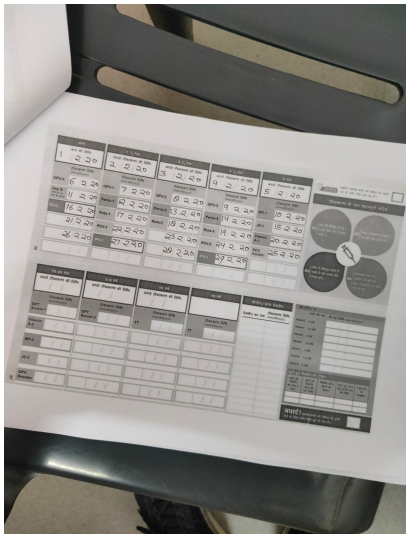
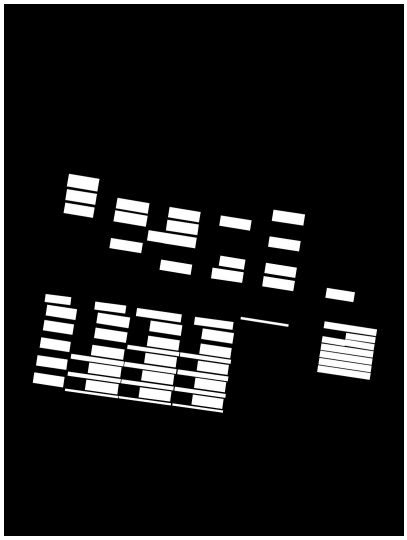
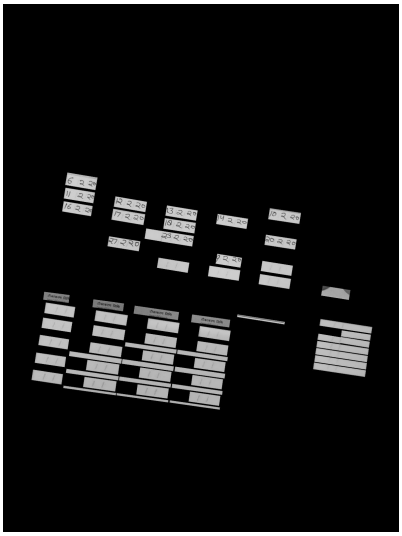
Form Field Segmentation

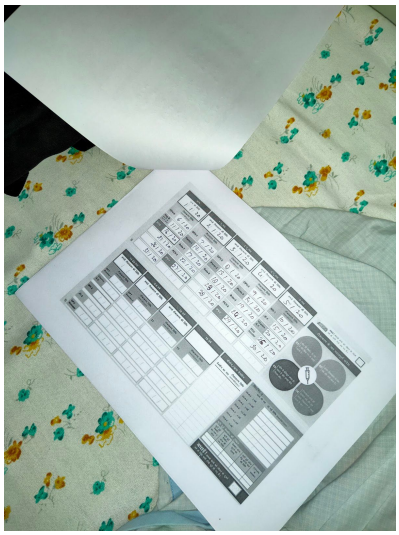
booklets/3.jpg output field

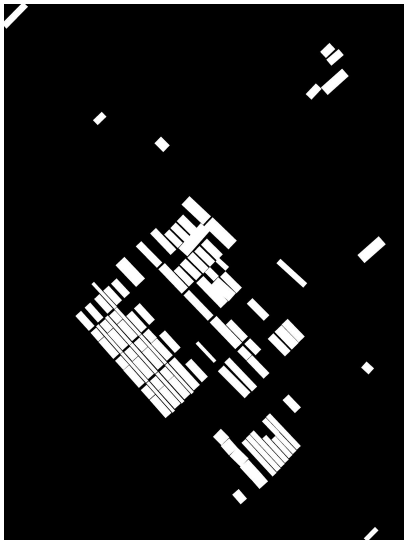
booklets/12.jpg output

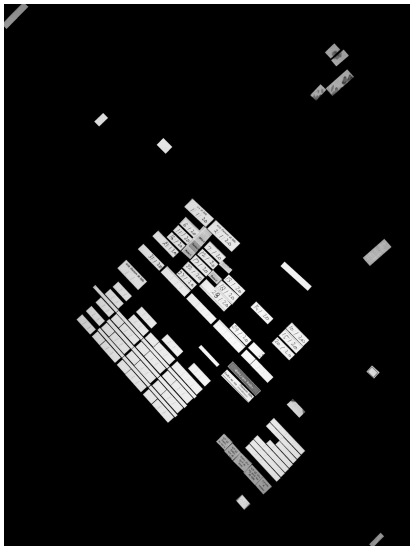
printout/44.png output form field character extraction



printout/18.jpg

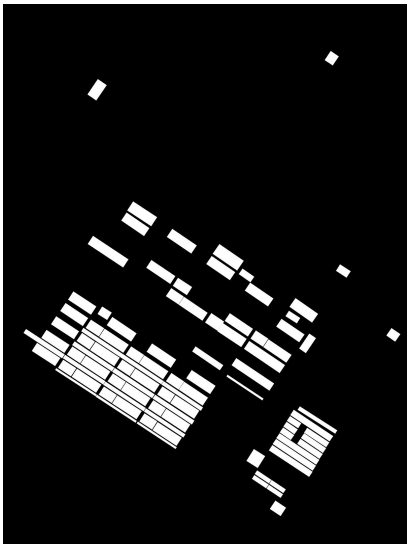


output(-85,-75)

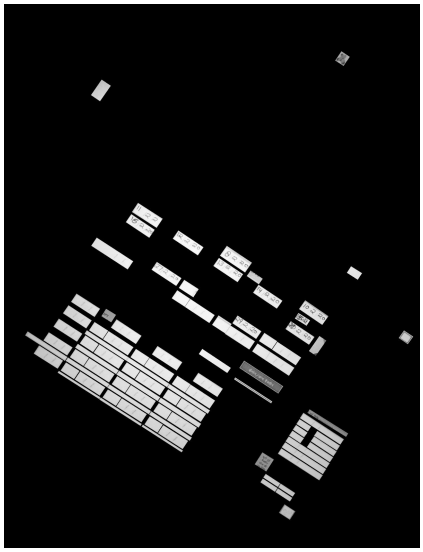


character extraction

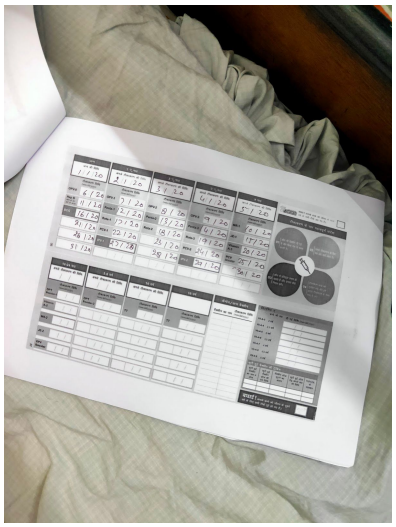
printout/20.jpg



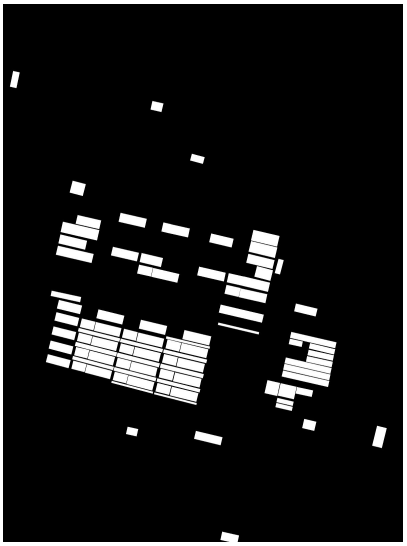
output(-58,-54)



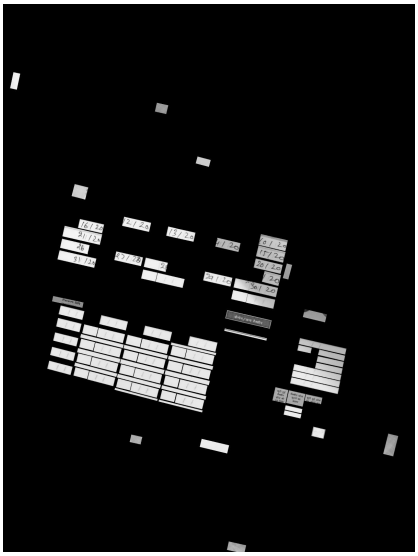
character extraction



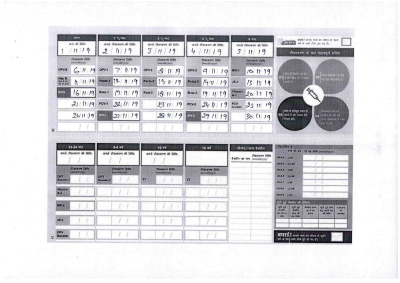
printout/48.jpg



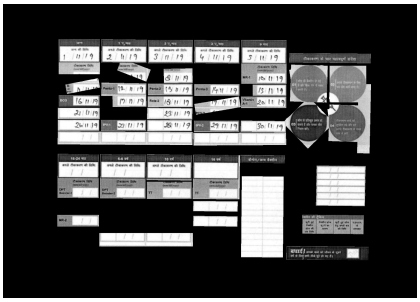
output(-78,-75)



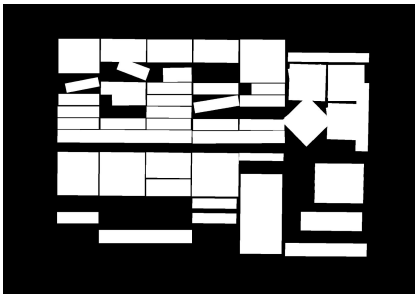
character extraction



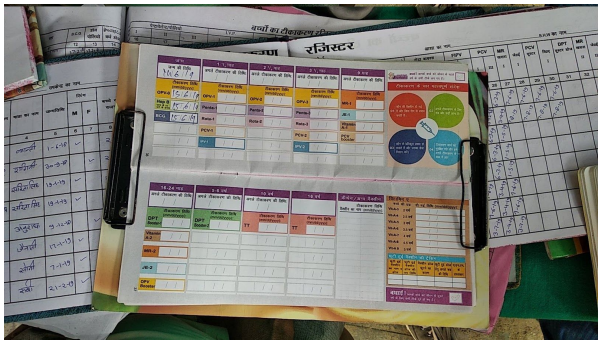
scanned/11.jpg



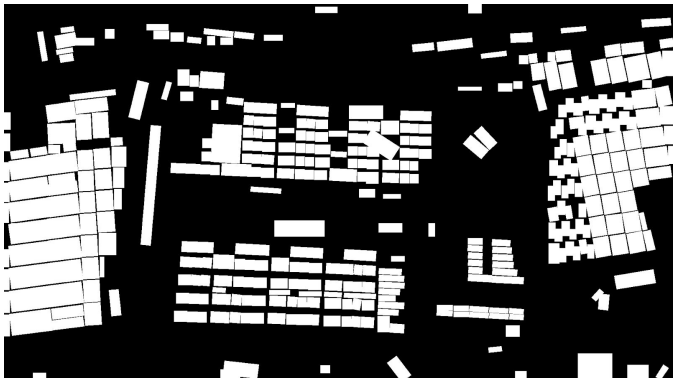
character extraction



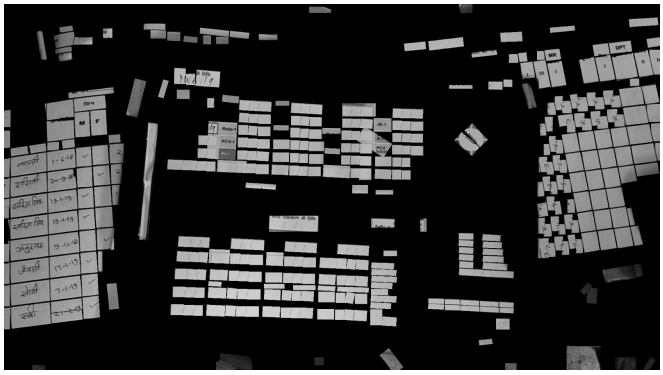
output



booklets/7.png

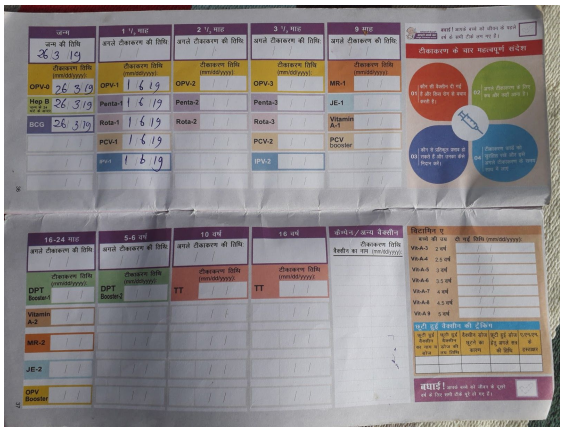


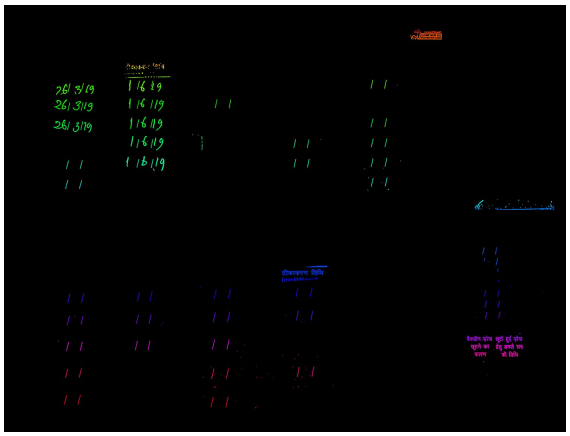
output

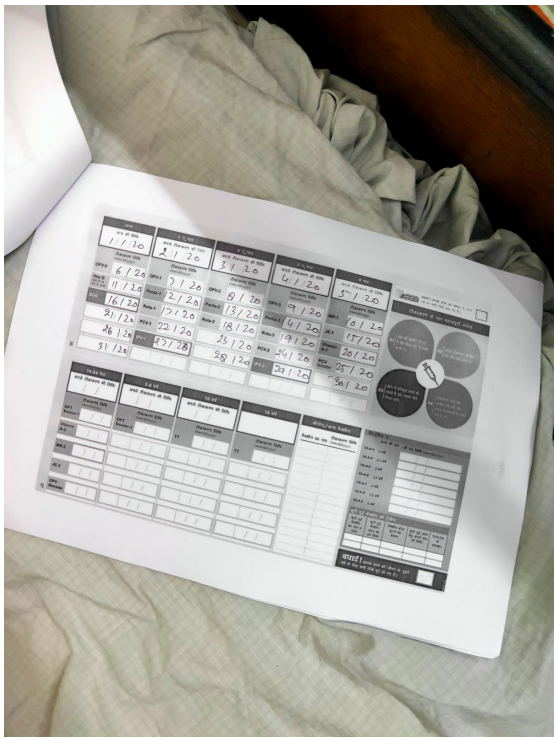


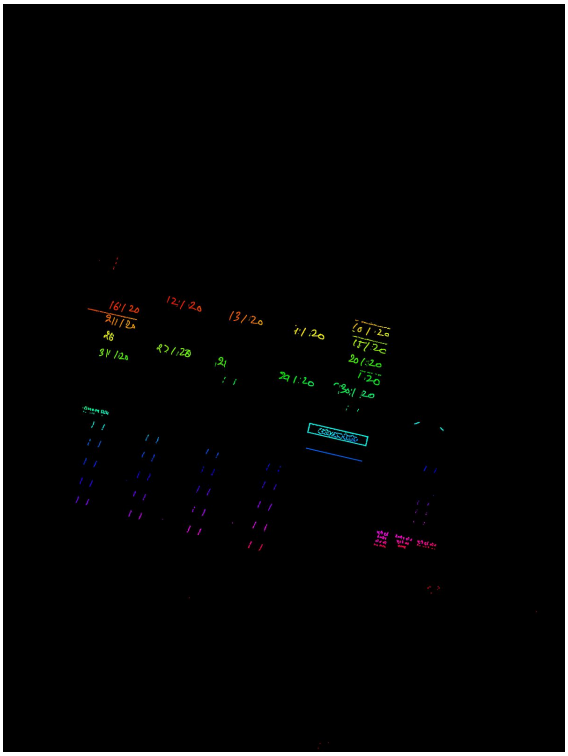
Character extraction

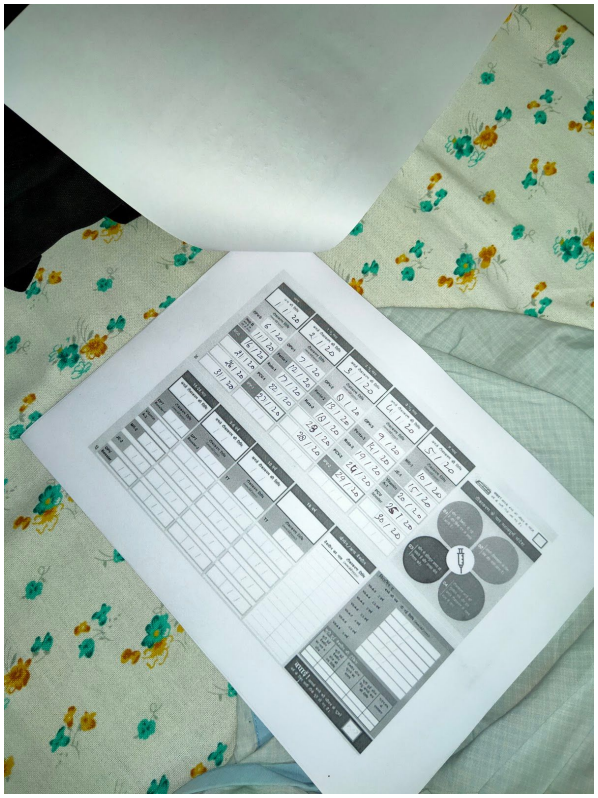
Character Detection

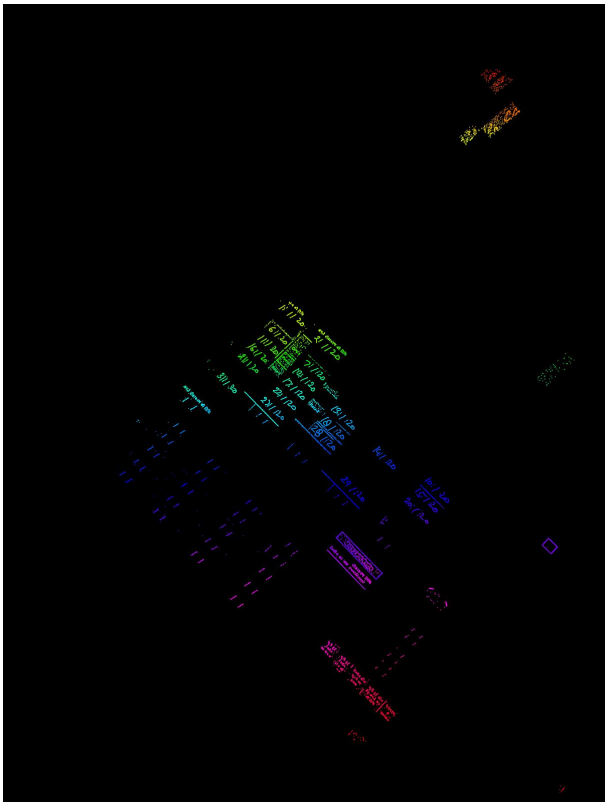


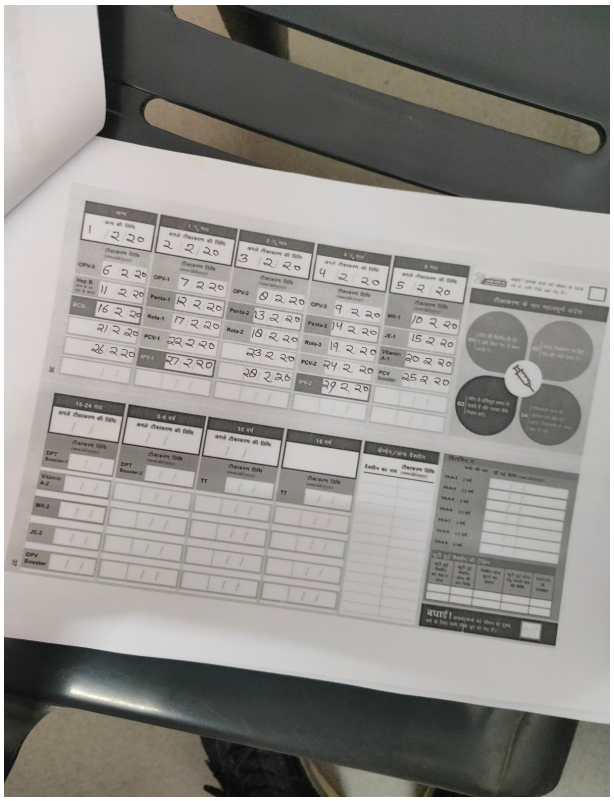


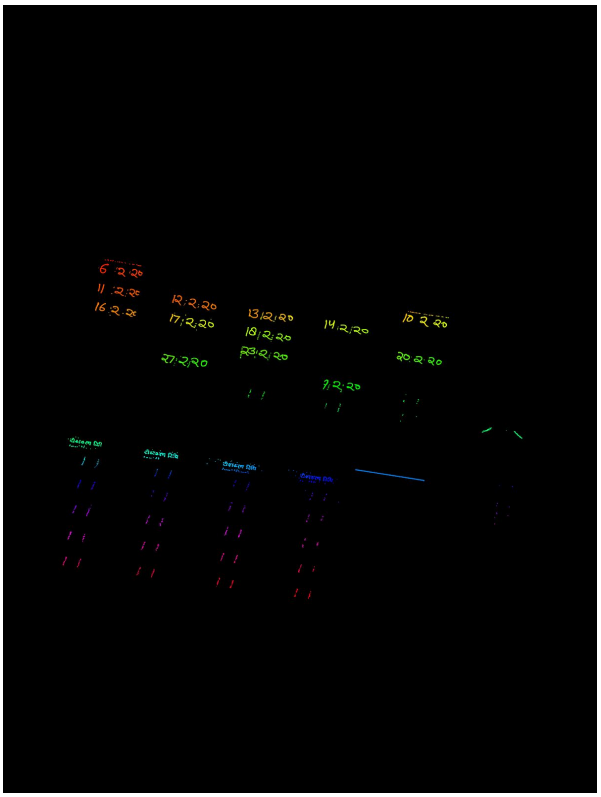


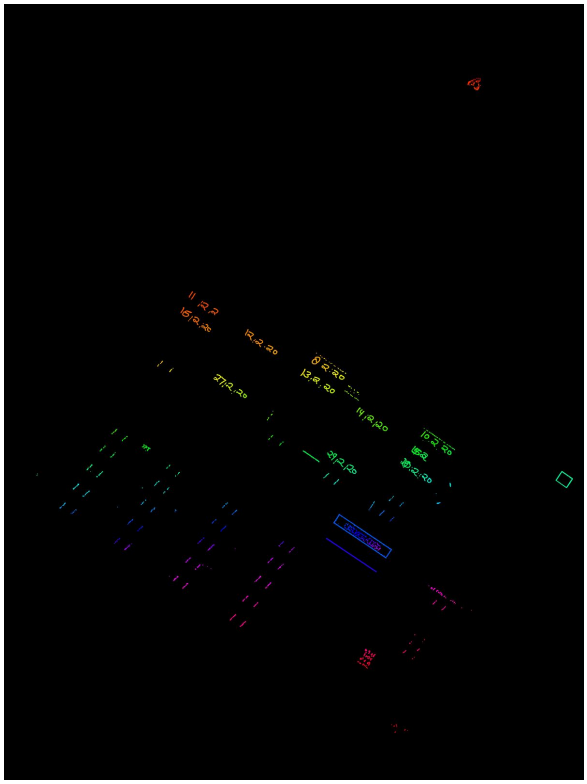












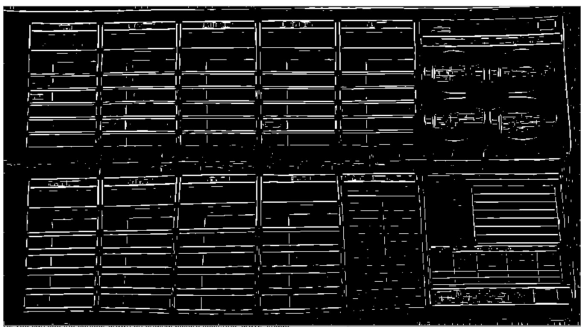
Extra Credit: Removing the slashes from the images:

The image is first binarized using adaptive thresholding and then hit and miss algorithm was used to remove all the slashes. It is then dilated for clear image

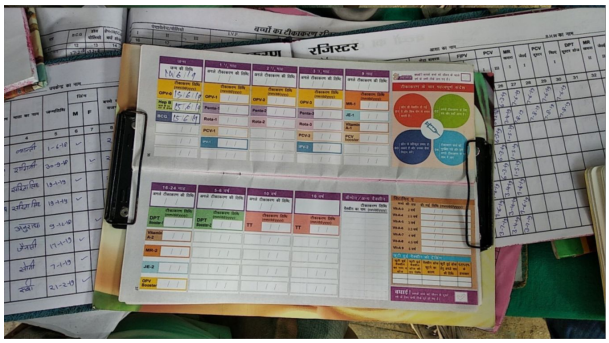
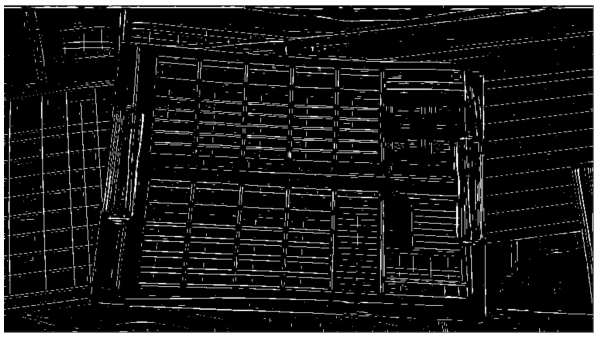
Image	Vertical kernel	Horizontal kernel	Adaptive Thresholding	Dilation
booklets/3.jpg	9	9	5, 1	rect(1,1)
booklets/7.jpg	9	9	5, 1	rect(1,1)
booklets/8.jpg	9	9	5, 1	rect(1,1)
booklets/12.jpg	9	9	5, 1	rect(1,1)
printouts/2.jpg	9	9	5, 1	rect(1,1)
printouts/9.jpg	9	9	5, 1	rect(1,1)
printouts/18.jpg	4	9	7, 1	rect(2,2)
printouts/20.jpg	diagonal(7,7)	13	9, 3	rect(2,2)
printouts/44.jpg	5	11	7, 1	rect(1,1)

printouts/48.jpg	5	11	7, 1	rect(1,1)
scanned/9.jpg	9	9	7,1	rect(2,2)
scanned/10.jpg	9	9	7,1	rect(2,2)
scanned/11.jpg	9	9	7,1	rect(2,2)
scanned/12.jpg	9	9	7,1	rect(2,2)

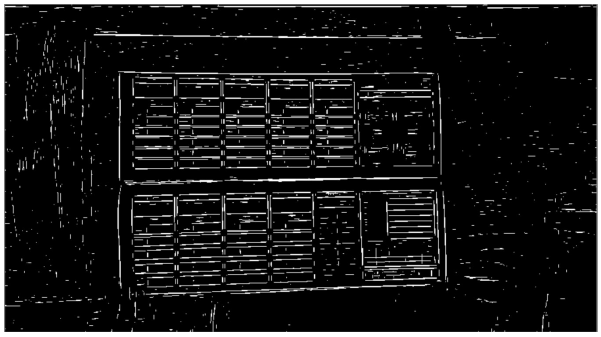
booklets/3.jpg

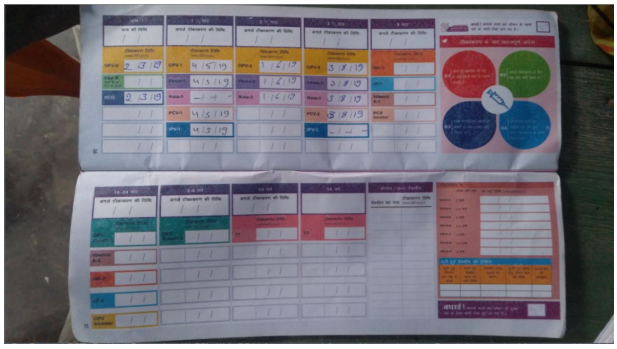
booklets/7.jpg

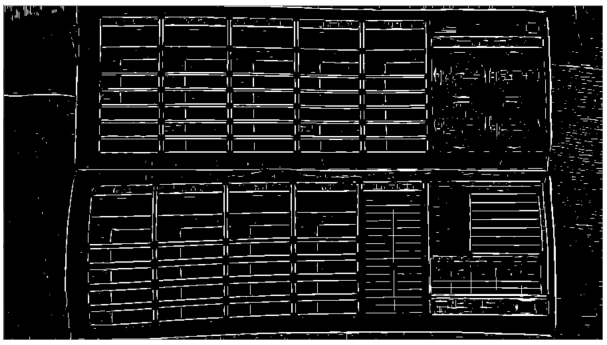



booklets/8.jpg

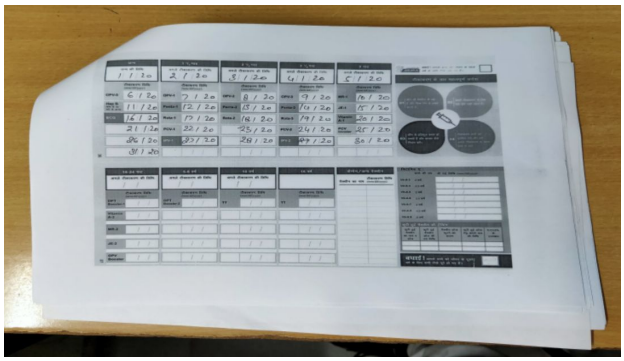



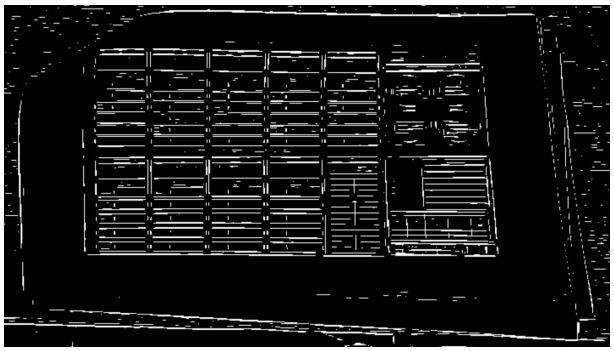
booklets/12.jpg



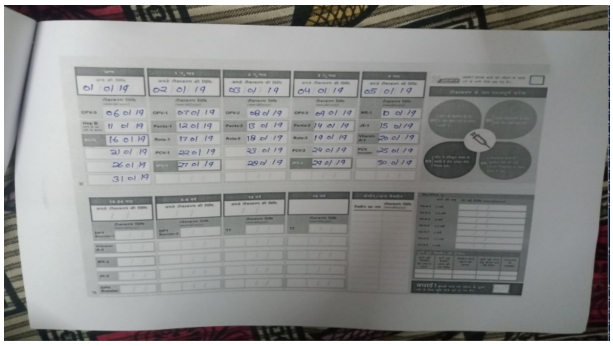


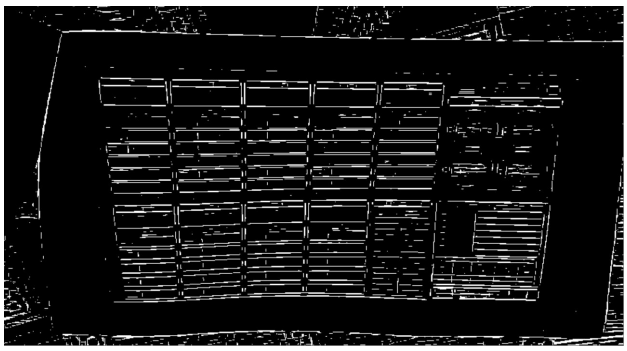
printouts/2.jpg



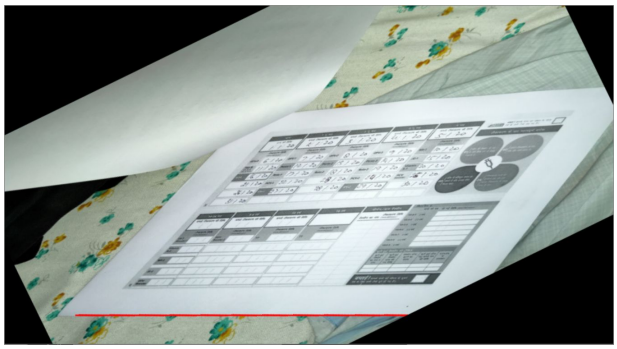


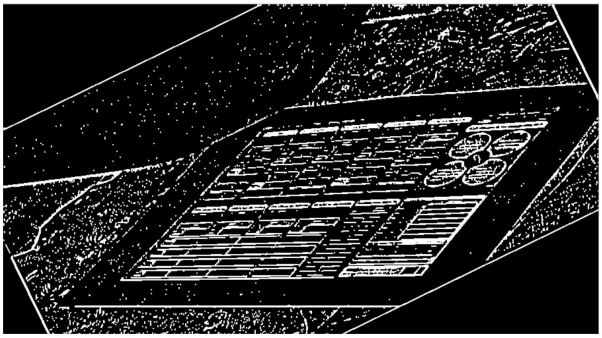
printouts/9.jpg



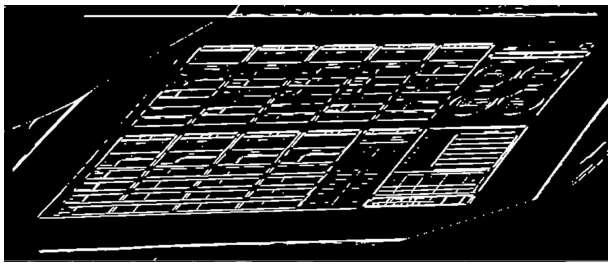


printouts/18.jpg

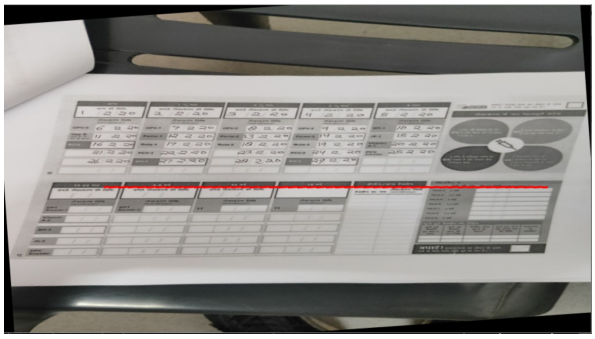
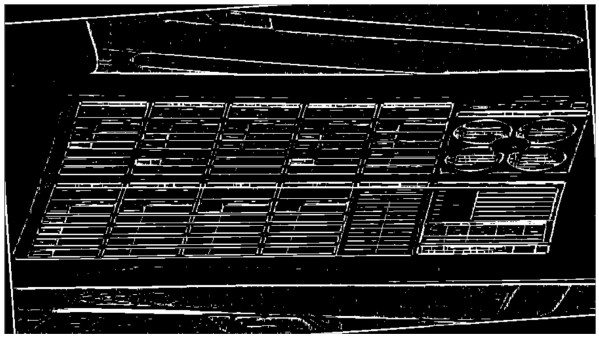




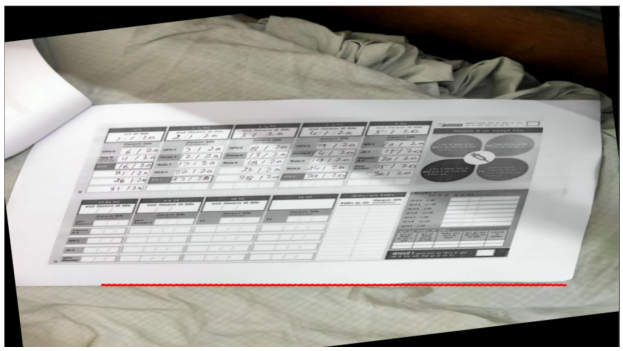
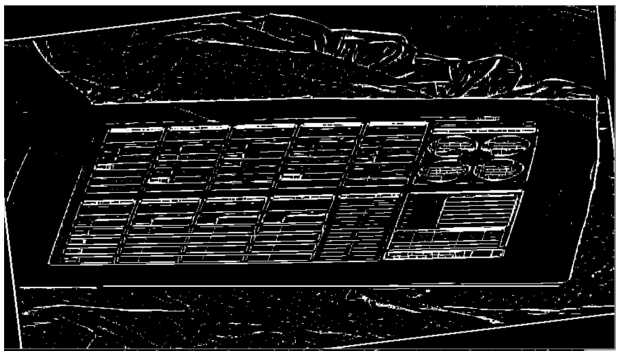
printouts/20.jpg

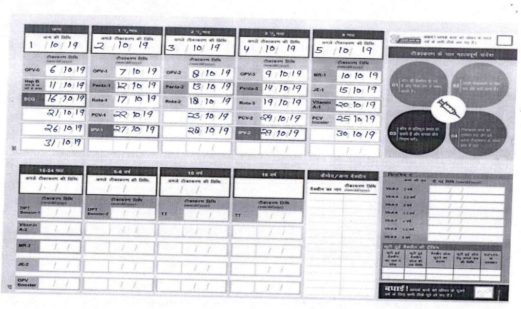
printouts/44.jpg

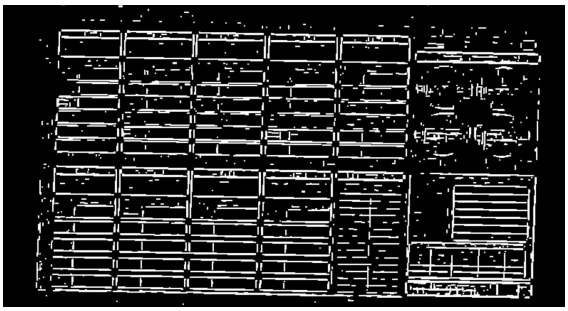



printouts/48.jpg

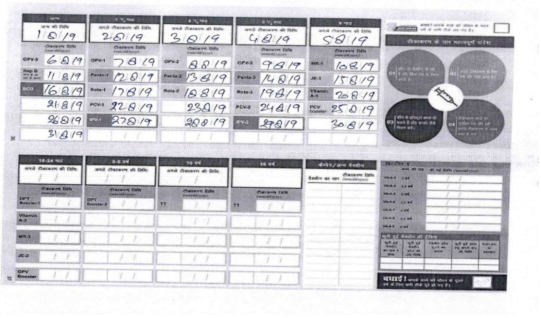



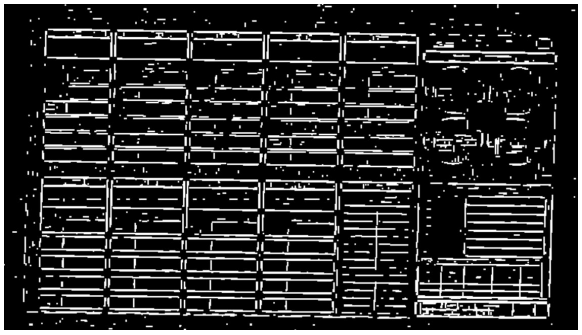
scanned/9.jpg



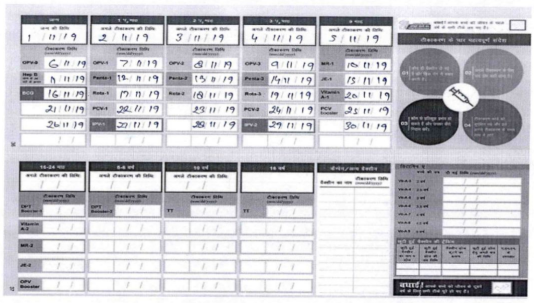


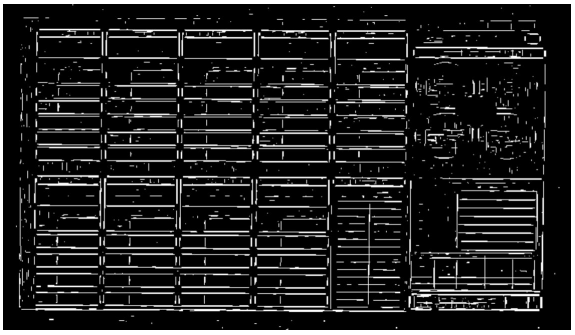
scanned/10.jpg



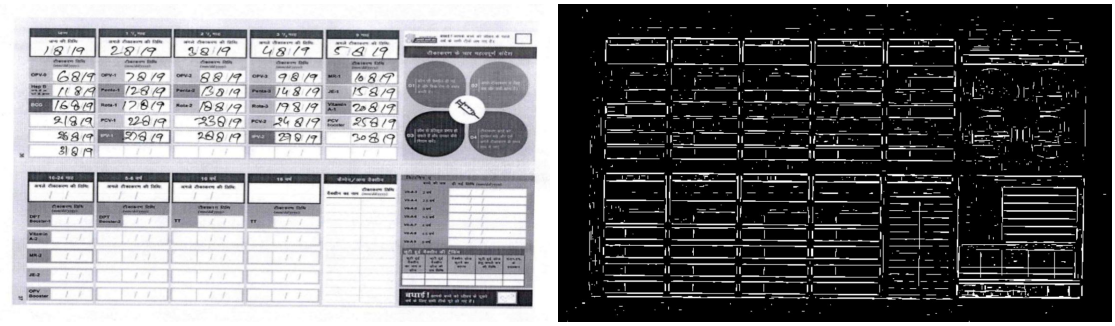


scanned/11.jpg

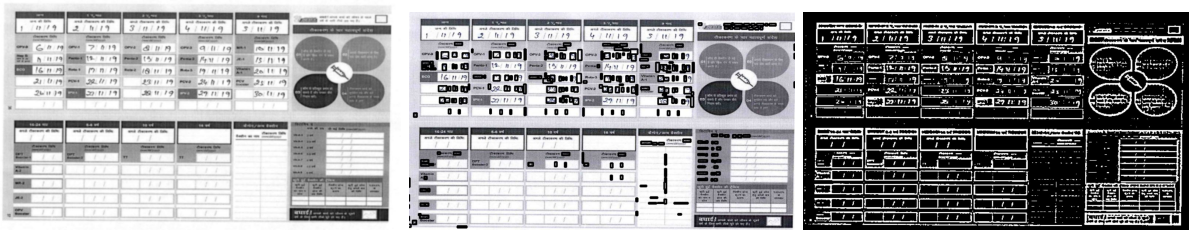




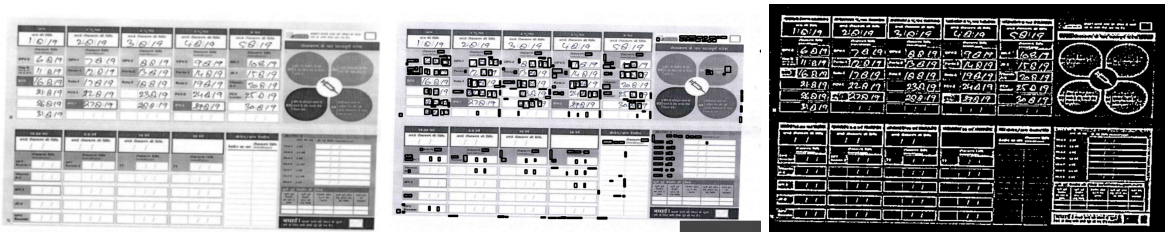
scanned/12.jpg



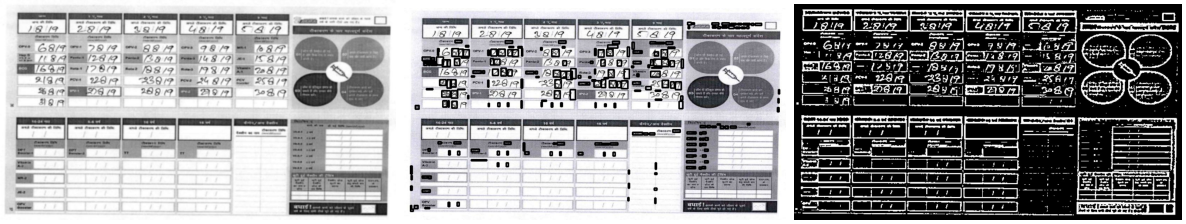
Scanned/11.jpg, (13,8) -> Adaptive thresholding: (13,8)



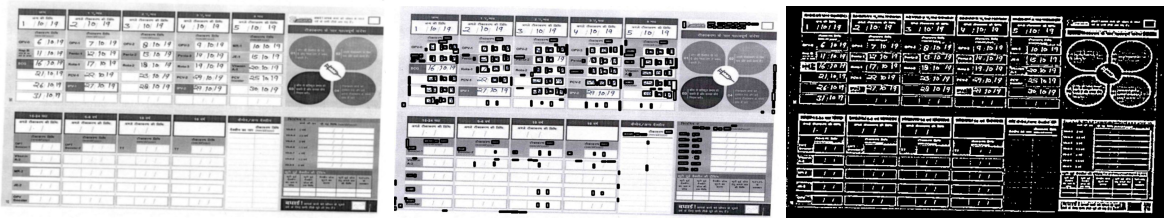
scanned/10.jpg



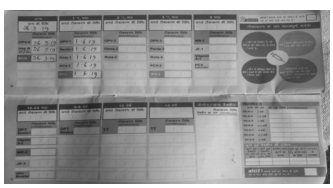
scanned/12.jpg



scanned/9.jpg



booklets/3.jpg




printouts/18.jpg (13,8)