Day2 - 23/05/22.

I. Document Image

OCR - optical character recognition.

① Skew correction → align it to be straight,

Compute required angle ??

↓

Rotation matrix.

Hough transforms, ... many methods exist

Edge detection canny → sq. filters.

↳ gaussian filter. ↳ general edge detection. σ is varying deviation of the

Hough transform → detects geometric shapes.

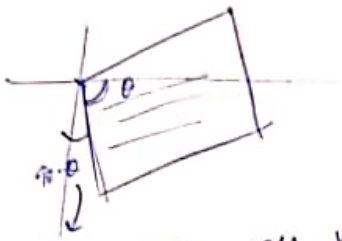
Hough space → point intersection } many lines

↓

(r, θ).

probabilistic houghline → faster

(Hough space is 4th quadrant)



how much you need to rotate.

if image has a lot } noise the algo fails

↳ soln:- → do denoising techniques.

② LINE WORD DETECTION.

→ perform adv. denoising cause we will recognize words based on white spaces ⇒ so remove noise.

→ Dilation helps differentiate between required spaces. (in between words gets filled)

↳ easy to recognize.

(slightly small kernel)

* Structuring Element → can be used for shape detection.

↳ determines filter shape.

(2) Reading a cheque number.

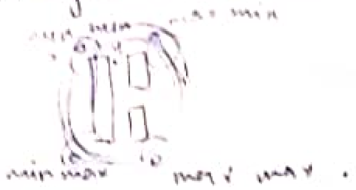
(1) extract symbols / map { ^reference img } — (1a) map bounding boxes to numbers/chars

(2) identify characters in text

(3) map them to ref dictionary.

template matching.

recognize MICR characters.



x y

Thresholding is basically binarizing an image.

closing op helps in identifying contours which helps in locating the target text.

- the closing can be ignored } as the recognized characters will only match in the cheque no.
- But more time consuming less efficient.

for comparing images => seeing if pixels are equal isn't always right => becoz noise need to account for lighting & angles.

IX. Ravi kiran Sir pres.

→ india.iiit.ac.in

→ Layout Parsing,

→ HInDoLA => analysis & annotation ecosystem.

HInDoLA.

→ object segmentation.
(but lines don't have fixed physical boundaries).
→ measure progress
→ annotate images → 
→ aspect ratio
problem 
↳ need to find a method independent of resize.

scale resize → blur

clusters → model training is biased.

→ scribble data medial axis.
(used patching to determine scribbles)
→ (then used scribbles to determine polygons)] → joining patches again problem is bypassed.

→ scribble to polygon
↳ by seam generation

! step by step & divide & conquer.