**Test 1**

Base starting point

**A screenshot of a computer screen

Description automatically generated with low confidence**

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.1

**Global Acceptance threshold:** 0.7

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

No dilation in preprocessing

ONLY USING IoU (No color and no text comparison)

**Test 2**

Increased iou\_thresh in img\_getSimilarity from 0.1 to 0.15

A screenshot of a computer screen

Description automatically generated with low confidence

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.7

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

No dilation in preprocessing

ONLY USING IoU (No color and no text comparison)

**Test 3**

Lowered global Acceptance Threshold from 0.7 to 0.6A screenshot of a computer screen

Description automatically generated with low confidence

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

No dilation in preprocessing

ONLY USING IoU (No color and no text comparison)

**Test 3**

Added text comparison into the mix.

This matching algorithm, in which if there is at least one more word detected in the sample, that resulting offset makes it so that nothing else matches.

A picture containing text, screenshot, font, line

Description automatically generated

For now, just returning 0 if text comparison is unsuccessful, and the pure iou value otherwise

A picture containing text, font, screenshot, number

Description automatically generated

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Iou binarizing thresh:** 140

**Text bin threshold:** 140

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

**Command running time:** about 5 seconds

No dilation in preprocessing

Not using Color comparison

A screenshot of a computer

Description automatically generated with medium confidence

**Test 4**

Changed text comparison algorithm

A screen shot of a computer code

Description automatically generated with medium confidence

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Iou binarizing thresh:** 140

**Text bin threshold:** 140

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

**Command running time:** about 5 seconds

No dilation in preprocessing

Not using Color comparison

A screenshot of a computer

Description automatically generated with medium confidence

Next step is to do perspective transformation to straighten sample images

This is showing very promising results.

<https://colab.research.google.com/drive/10IWD8MtCkOb_zBD9ylKinD62SnxXDdl9?authuser=1#scrollTo=4z_XtTUOvzFu>

In the above link I’m currently working on automatic corner detection for perspective transformation, but my goal is to make a GUI for calibration in which the user can select the four corners of the screen as points to use for straightening (just like in scanner apps).

**Test 5**

I changed the binarizing threshold for text preprocessing from 140 to 127.

Also, applied perspective transformation to straighten images using cv2.minAreaRect. It is expected that using other method that is not strictly a rectangle but rather any quadrilateral will yield better results.

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Iou binarizing thresh:** 140

**Text bin threshold:** 127

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

**Command running time:** about 5 seconds

No dilation in preprocessing

Not using Color comparison

Also, right now, findAllWords sometimes returns

Strings with two words. It will be better to make

Sure it always returns individual words.

A screenshot of a computer

Description automatically generated with medium confidence

Although perspective transformation helps significantly in elevating the match confidence of similar images through IoU (Structural Similarity), it is highly suspected that these poor results are due to the decrease in the binarizing threshold from 140 to 127 in the image preprocessing for text. It is probably better to use 140. Another thing that may be tried is to use Adaptive thresholding for this preprocessing step.

**Test 6**

Changing the binarizing threshold back to 140 in the text preprocessing yields these results. Precision is much better, but still, the text comparison hinders the recall with many false negatives.

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Iou binarizing thresh:** 140

**Text bin threshold:** 140

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

**Command running time:** about 5 seconds

No dilation in preprocessing

Not using Color comparison

A screenshot of a computer

Description automatically generated with medium confidence

**Test 7**

Changed find\_all\_words function to ensure it always returns pure words (not phrases separated by space).

A picture containing text, screenshot, font

Description automatically generated

**Algorithm:** Intersection over Union

**IoU thresh:** 0.1

**Iou binarizing thresh:** 140

**Text bin threshold:** 140

**Contour Finding:** RETR\_TREE

**Threshold to trigger match (iou thresh):** 0.15

**Global Acceptance threshold:** 0.6

**Cam to screen distance:** 18cm

**Binarize threshold:** 140

**Cut margin:** 0

**Command running time:** about 5 seconds

No dilation in preprocessing

Not using Color comparison

A screenshot of a computer

Description automatically generated with medium confidence

Next I want to try adaptive threshold on image preprocessing for text.