Report 5

Jiachen Tian

Objectives achieved this week

- Exploring and trying different algoritms for precision improvement and aut
omation.
- Added a few error checks for video constructions.

Objectives for next week

- Keep finging suitable algorithms for precise pupil tracking.
- Start working on glint tracking.
- Match video tracking to data to shorten runtime.

Results Demo

Here comes the harddest part of the whole research -- algorithms that runs fast, robust, precise, and automated(Testing for different options).

name/evaluate	Hough transform	RANSAC	Harmony Search
Fast?	**Mediocre(O(n^4))** In order to detect the circle, Hough transform algorithm would be performed in a 3-D fashion — Cast every potential centers(x, y) to a new dimension as a point, then for each point, iterate through every possible radius and Angels. Finally, perform a vote to determine the center of the circle(most voted), and its corresponding radius. Could be improved by parallelism.	**Slow** Finding the "trend" that corresponds to most data in the image by first selecting a few candidates, create a circle using the candidates, create a donut shape region that includes the user defined error margin. Keep on looping the same algorithm and at the same time updating the candidates chosen, util find the candidates collection that concludes the most data in the error margin.	**Slow** Prepare a Harmony Memory. Improvise a new "Harmony" with experience or Randomness. If the new harmony is better, include it in Harmony Memory. Repeat the above steps until a perfect(Harmony/ circle) is located.
Robust?	**Yes** Hough transform itself is robust enough. However, it needs tons of preprocessing to generate perfect result	**No** RANSAC itself is not robust. Would be easily affected by outliers. Not the mention the equal amounts of preprocessing compared with Hough Transform.	**No** Every determination generated is based on the previous data and previous generated data. Some outliers would mess up everything

name/evaluate	Hough transform	RANSAC	Harmony Search
Automated?	**No** Preprocessing needed including blurring, threshold, and canny image which requires tons of user inputs. image 1 shows my attempt to automize canny image.	**No** Same reason as Hough transform, too many preprocessing needed that requires user inputs.	**No** Same reasons as Hough transform. In order to get the perfect “harmony”, outliers must be eliminated.
Precise?	**Yes** Very precise even with outliers	**No** Not robust even to outliers	**No**
Conclusion	Best algorithm so far	Abandoned	Scam

Attempts to automize preprocessing for Hough Transform using machine learning

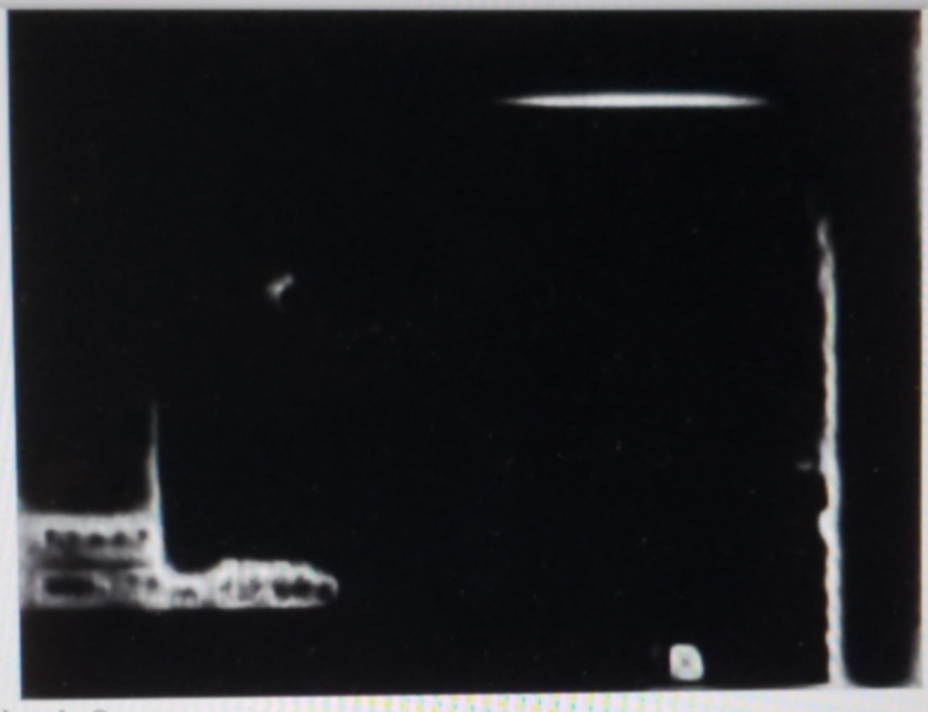
- Looks like machine learning favors the frame more than the pupil.
- Alternative option is to create one's own machine learning model.

File Transfer

— Python detect_edges_image.py --edge-detector hed_model --image ../input/

edge detector...
ng Canny edge detection...
ng holistically-nested edge detection...
_tracking wangaiqing$ python3 detect_edges_i
image ../input/chosen_pic.png

HED



(x=11, y=1) ~ L:0

edge detector...
ing Canny edge detection...
ing holistically-nested edge detection...

反馈


MacBook Pro
F4
F5
F6
F7
F8

Conclusion

-This week, I've explored three different algorithms for circle finding, and among them all, Hough transform is the best. However, the issue remains rega rding how to automate the program, otherwise, the user has to filter out the best image outcome from blurring(two variables), threshold(2 variables), and canny(2 variables).