Through-Screen Finger Localization and Tracking using Reflected Light

Andrei Grigore Croitoru
a.croitoru-01@student.tudelft.nl Supervisor: Dr. Qing Wang

1) Background

Visible-light positioning (VLP) systems share a common geometry where the light sources are fixed and the receiver moves through the scene.

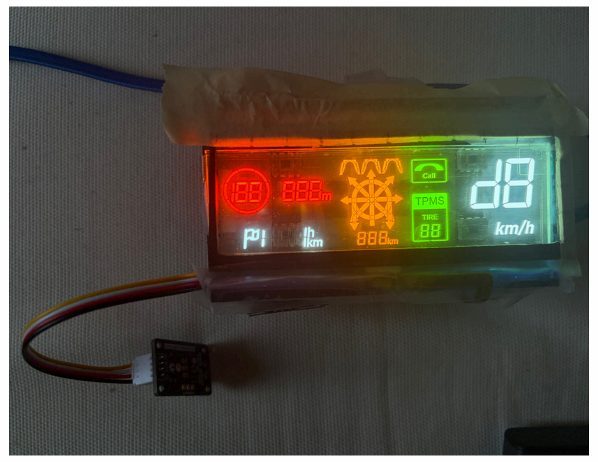
Our goal: Invert the common geometry of VLP setups and manage to track a hovering finger above an OLED display using under-screen photodiode sensors on the microcontroller that already drives the screen, without deep learning.

Prior work demonstrated that through-screen reflected-light sensed by an array of photodiodes under a transparent OLED carries sufficient spatial information to classify air-written digits with up to 91% accuracy [1], if the finger is 1 cm above the screen. Building on this idea, we ask whether the same signal can drive continuous tracking on a segmented display using lightweight classical algorithms that fit the embedded chip itself.

We see this technology being used in mobile phones to detect the user's finger and preload the applications or links the user intends to press on.

Hardware

- SparkFun transparent OLED HUD
- Four OPT101 photodiodes
- One Arduino Due



2) Research questions

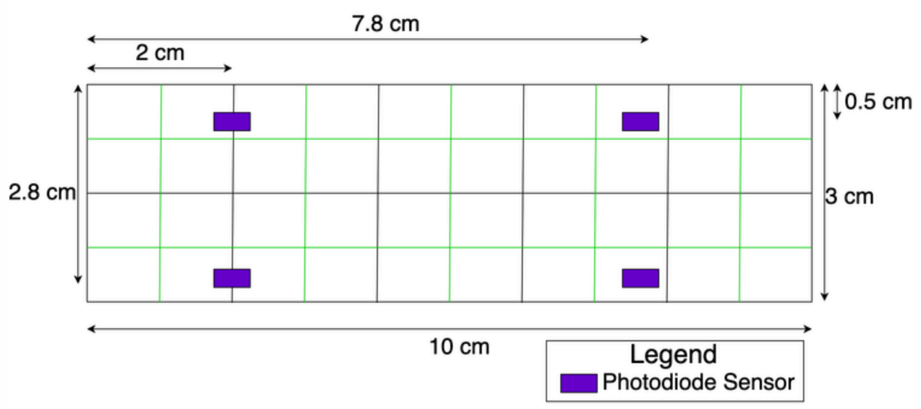
Can 4 under-screen photodiodes localise and track a hovering finger above a segmented OLED display in real time on embedded hardware?

Sub-questions

1. To what extent do preprocessing and feature engineering drive accuracy, relative to the choice of classifier?
2. How does accuracy translate from finger localization to tracking?
3. Does the model fit the SRAM, flash and real-time budget of the Arduino?

3) Methodology

We will approach this problem as device-free fingerprinting, a calibration grid maps the 4 photodiode intensity vector to known positions of the finger. This 10 x 4 data collection grid is shown below



- 199 finger captures at 1 second capture window with the finger hovering at 15-20 mm from the screen
- performance evaluated on a 5 x 2 grid

Compared Architectures

1. regression-then-snap to the evaluation grid
2. two-stage classifier with a 5-way column classifier and a 2-way row classifier, recombined as cell = row x 5 + col

Preprocessing

We subtract the per-channel no finger baseline before localisation in order to remove the screen-finger coupling.

Feature representations

We have tried 30 variants (4 to 18 dimensions). All the variants are composed of one or more of these components:

1. raw intensity values (4 dimensions)
2. squared intensity values (4 dimensions)
3. six pair-wise log ratios between all intensities (6 dimensions)
4. sum-normalised features (4 dimensions)

On top of this we have added Sigmoid Function Data Preprocessing [2]

Models compared

1. Logistic Regression
2. k-NN Classifier
3. SVC with RBF kernel
4. Random Forest Classifier
5. Extreme Learning Machine Classifier

All hyperparameters were selected by a 5 fold cross validation.

Evaluation Methods

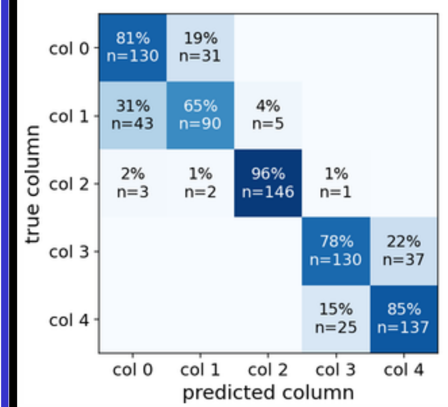
The primary metric for success is **cell accuracy for both evaluation methods:**

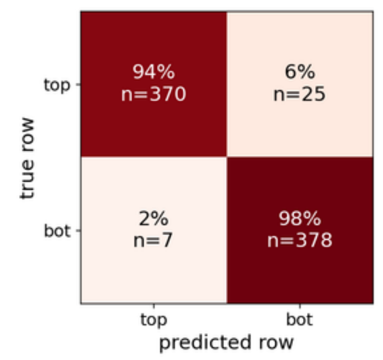
1. 20 iterations on a 80/20 training testing split
2. 40-fold leave-one-calibration-dot-out (LODO), each fold withholds all visits of one dot

4) Results

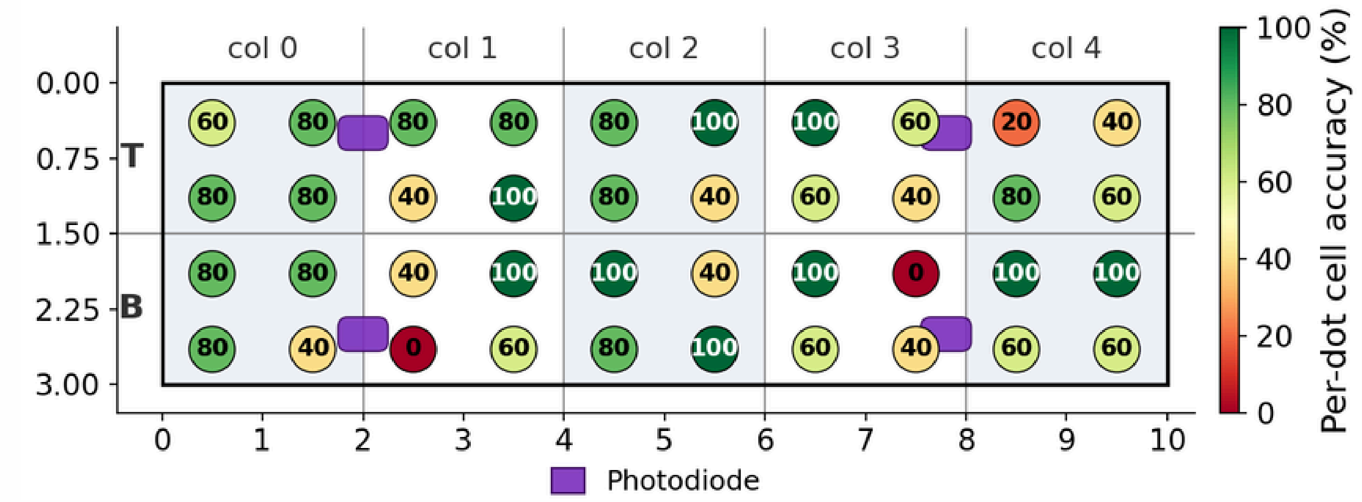
The two-stage classifier beats the regression-then-snap architecture by **17 percentage points**.
Best (model, feature) pair is (**logreg, log+raw+sq+pn)** with

- **77.2%** random-split cell accuracy
- **66.8%** LODO cell accuracy

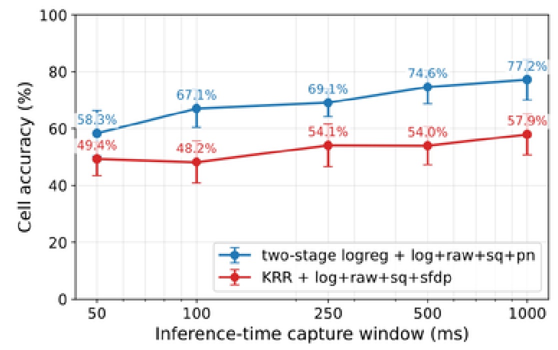




The X-axis is the bottleneck
Row accuracy: 95.9%
column accuracy: 81.2%

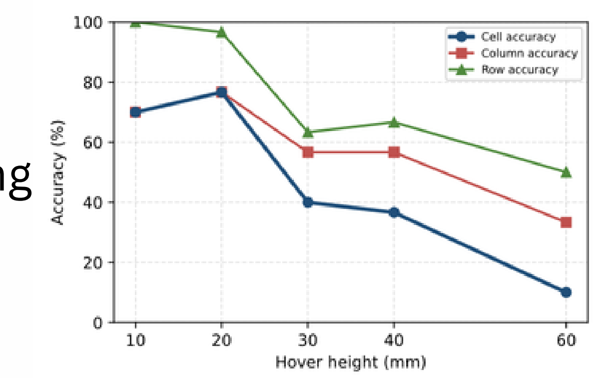


Dots nearest the photodiodes are hardest to identify since they sit on the steepest part of the received-power curve.



Tracking degrades gracefully as the capture window shrinks

- 77.2% at 1000 ms to 58.3% at 50 ms



The best accuracy is around the training hover height.
Accuracy decreases as hover height deviates from the training range.

5) Conclusion

Four under-screen photodiodes localize a hovering finger to the correct cell with 77.2% accuracy (66.8% LODO) without using deep learning, in under 1 ms of inference, entirely on the Arduino Due that drives the screen.

[1 Hao Liu, Hanting Ye, Jie Yang, and Qing Wang. 2021. Through-Screen Visible Light Sensing Empowered by Embedded Deep Learning. :
[2]: Y. -C. Wu et al., "Received-Signal-Strength (RSS) Based 3D Visible-Light-Positioning (VLP) System Using Kernel Ridge Regression Machine Learning Algorithm With Sigmoid Function Data Preprocessing Method"