WEyeDS: A Desktop Webcam Dataset for Gaze Estimation

Anatolii Evdokimov[1], Catherine Finegan-Dollak[1], Arryn Robbins[1]

[1]University of Richmond, VA, USA

ETRA 2024 LBW

Background

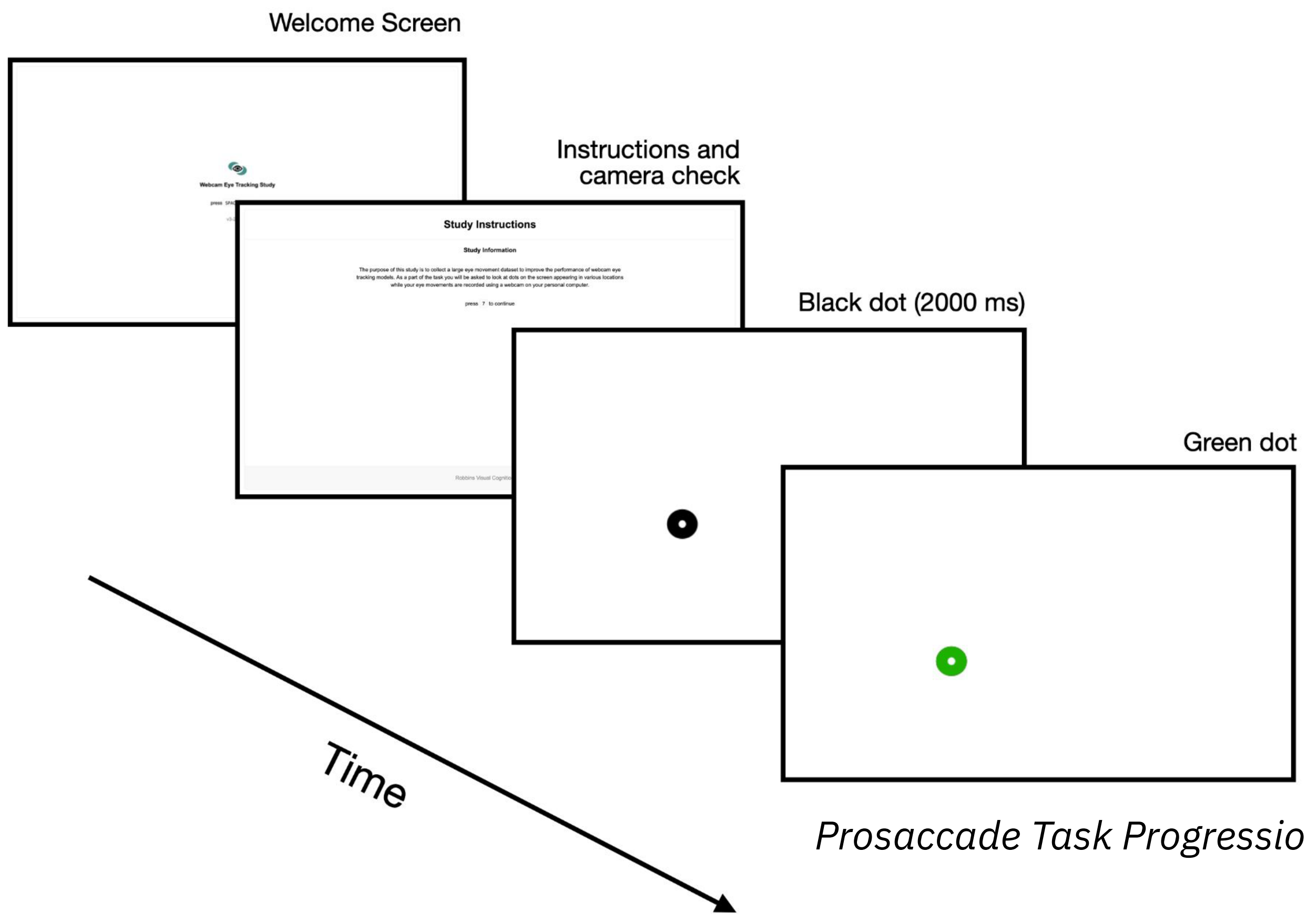
- Significant advances were made in appearance-based 2D gaze estimation with some models achieving <2cm estimation error [1].
- Those models require large open datasets like GazeCapture [2].
- There is a need to have a dataset with variability as high as GazeCapture for desktop and laptop devices.

Current Study

- The goal of this project is to **create a new eye movement dataset** collected using laptop/desktop webcams.
- **Evaluate the dataset** using an existing gaze prediction model.
- **Discuss limitations and improvements** to the current data collection procedure and experiment design.

Method

- Dataset was **collected online** from a pool of participants at the University of Richmond using webcams.
- Participants completed **a prosaccade task** (following a dot on the screen).
- Participants first saw a **black dot** appear on the screen for 2000 ms.
- Once the dot **changed color** to green, participants pressed a space bar to see the next dot.



Prosaccade Task Progression

Data

- The dataset consists of full-face images of **38** (54) **participants** completing a prosaccade task along with various metadata.
- Each participant looked at **dots in 100 locations** across a predefined grid of 50 cells. The cells were used to generate dot locations that evenly cover the prediction space.
- Each *x* and *y* dot coordinate corresponds to **a proportion** of the participants' vertical and horizontal viewport.
- With 6 frames per each location, each participant has **600 image examples** in their trial.
- Helpful metadata about participants' screens and browsers was also collected:
  - **Browser data:** user agent, platform.
  - **Screen data:** screen width and height, scroll width, inner window width and height, device pixel ratio.

Data Processing & Benchmark Model

Data Processing

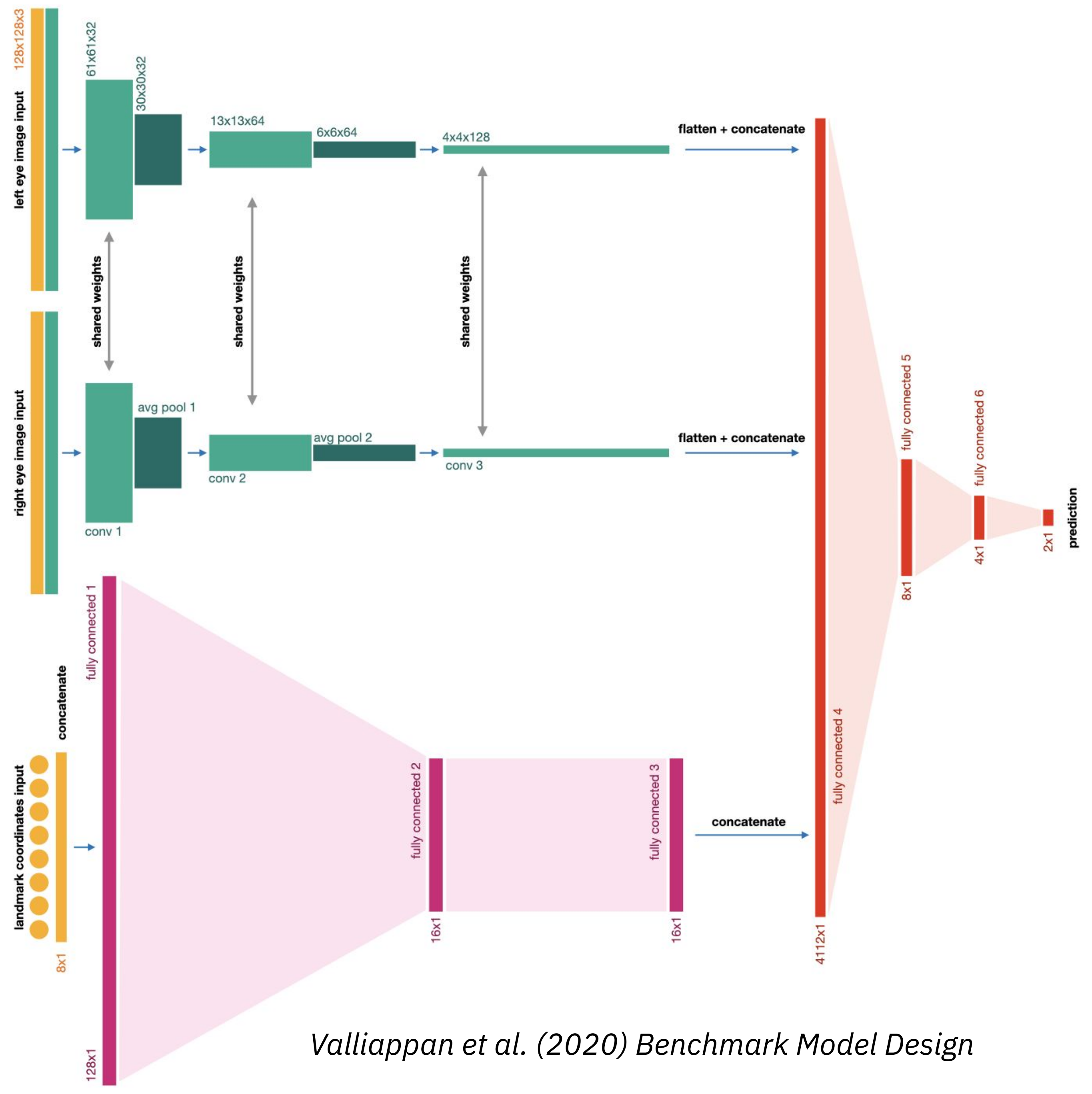
- To generate image inputs for the model, we used Google's MediaPipe face mesh [3] to obtain **eye crops and eye corner landmarks** to add to our training examples.
- Additionally, **we rescaled** the *y* coordinate of each dot location to be on the same scale as the *x* coordinate. This was done **to obtain a meaningful loss value** at the end of training since a proportion of the width is on a different scale compared to the proportion of the height.
- Before training, the **eye crops were normalized**, and the **left eye image was flipped** to match the right eye image to ensure that weights can be shared between the convolutional components of the model.

Benchmark Model

- **We reproduced the model from [1]** using Keras with TensorFlow backend.
- The model consists of **a convolutional component** for eye image processing and **a fully connected component** for eye corner landmark coordinate processing.
- The outputs from the two components were concatenated and processed through several additional fully connected layers to obtain the final gaze location prediction.

Training

- The model was compiled with the Adam optimizer and the Euclidean distance loss function.
- The final training set contained 18,000 examples and the validation set contained 2,400 examples.
- The model was trained with an initial learning rate of 0.016, decay rate of 0.64, and a batch size of 256 for 40 epochs.
- No hyperparameter tuning was done. The values were obtained from [1].

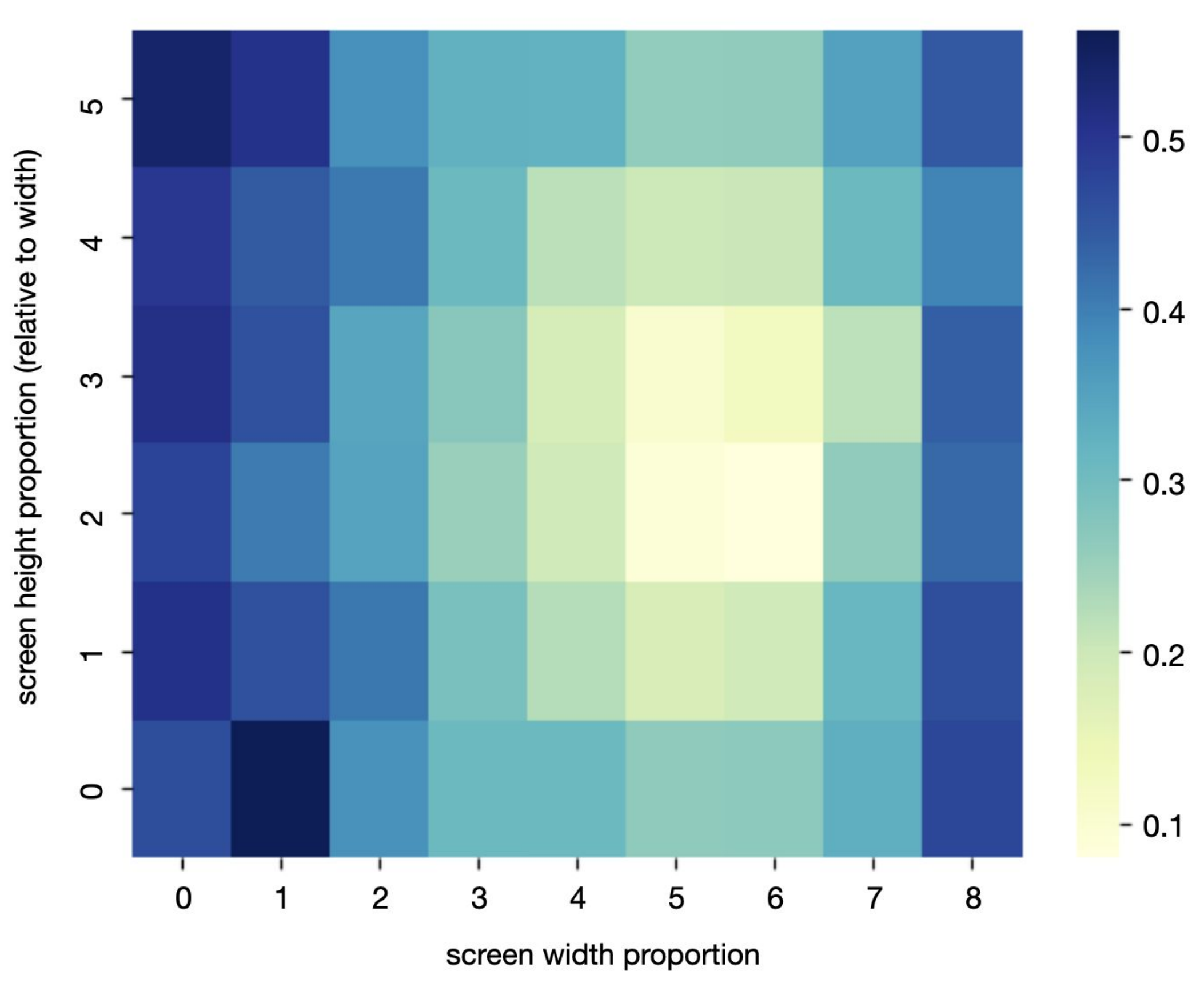


Valliappan et al. (2020) Benchmark Model Design

Preliminary Results

- The model achieved the final validation loss value of **0.293**.
  - This means that the model was on average 29.3% of the horizontal viewport away from ground truth.
- **An additional training run** on a larger dataset of 54 participants with a 80-10-10 split did not show a significant difference in the final test loss value.
- We also conducted an additional **loss analysis** with loss values averaged across screen grid cells.
  - The analysis showed better loss values towards the center of the screen

Heat Map of the Euclidean distance loss by screen region



Conclusion & Limitations: Since screen parameters are highly varied across participants, more data needs to be collected to account for variability in the population. More checks can be put in place to ensure that participants are looking at the dots

References

[1] Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff, and Vidhya Navalpakkam. 2020. Accelerating eye movement research via accurate and affordable smartphone eye tracking. Nature Communications 11, 1 (2020), 1–12. https://doi.org/10.1038/s41467-020-18360-5

[2] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. 2016. Eye Tracking for Everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[3] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A Framework for Building Perception Pipelines. CoRR abs/1906.08172 (2019). arXiv:1906.08172 http://arxiv.org/abs/1906.08172

Dataset & Code