Hand Gesture Recognition on Arduino Using Recurrent Neural Networks and Ambient Light

Matthew Lipski - m.s.lipski@student.tudelft.nl
Supervisors - Mingkun Yang & Ran Zhu
TU Delft

1. Aim

1.1 Research Question

Which recurrent neural network (RNN) architecture is most appropriate for recognizing hand gestures on an Arduino Nano 33 BLE, using 3D-formatted data from OPT101 photodiodes?

- Which recurrent neural network architectures produce the highest accuracy for hand gesture recognition?

- What is the minimum acceptable accuracy for recognizing hand gestures on an Arduino Nano 33 BLE?

- What is the maximum acceptable inference latency for recognizing hand gestures on an Arduino Nano 33 BLE?

- How can 3D-formatting data be exploited for better gesture recognition performance?

1.2 Research Overview

- Physical buttons in public areas create additional risk of spreading disease, making hand gestures are a compelling alternative.

- There are three key challenges for implementing hand gestures:

  1. Additional hardware is needed to recognize hand gestures, and is likely to be low in resolution since costs should be minimized.

  2. Gestures must be accurately recognized across a variety of users.

  3. Gestures must be recognized in real-time for a positive user experience.

- This research overcomes these issues by using data from OPT101 photodiodes, which is fed into a CNN-LSTM neural network to recognize gestures on an Arduino Nano 33 BLE microcontroller.

- Although similar systems exist [1], this research improves on them in a number of ways:

  1. The system uses fewer photodiodes, resulting in fewer neural network input features, than existing solutions.

  2. The data from photodiodes is 3D-formatted, which better preserves temporal information and improves recognition accuracy.

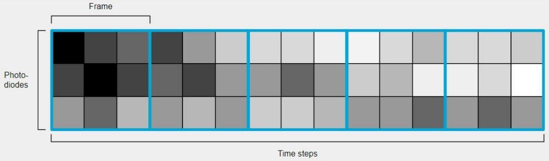
  3. The CNN-LSTM architecture used yields a higher validation accuracy than architectures used in existing solutions

2. System

2.1 3D-Formatted Data

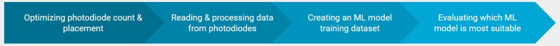
- 3D-formatting splits 2D data into *n* frames, making it more suitable for sequential data.

- 2D data can be thought of as an image, in this case with resolution *photodiodes X time steps*.



2.2 System Overview

- This research is only the final step of a larger project:



Optimizing photodiode count & placement → Reading & processing data from photodiodes → Creating an ML model training dataset → Evaluating which ML model is most suitable

- Photodiodes output a voltage which increases with the intensity of light that hits them, meaning they can track hand shadows under ambient light.

- Each gesture is made up of 100 samples/time steps from 3 photodiodes over a 5 second window.

- This data is 3D-formatted and input into a neural network, which outputs a prediction for which one of 10 gestures was performed.

2.3 Dataset

- The dataset used for training & validation was created by another member of the project group, and contains 5 repetitions of each of the 10 gestures per hand across 47 participants/ candidates.

- Some data instances were left out for various reasons, leaving 4672 total data instances in the dataset.

- Dataset is not final, i.e. is missing the preprocessing done by another project group member, due to time limitations.

3. Results

3.1 Inference Latency Testing

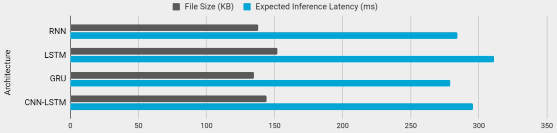
- For technical reasons, the architectures investigated in this study could not be deployed on Arduino using the TensorFlow Lite for Microcontrollers library.

- However, the inference latency in milliseconds of regular artificial neural networks (ANNs) could be predicted accurately from their file size in kilobytes, using the equation $y = 1.89x + 23.6$ with $R^2 = 0.998$.

- This equation was used to predict the inference latency of RNN architectures.

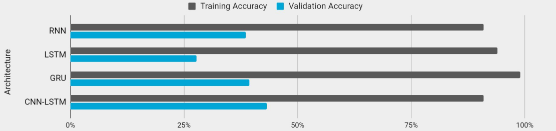
- The largest ANN that could fit in the Arduino's memory was 160KB and had an inference latency of only 333ms, which can be considered real-time as human reaction time is on average 627ms [2].

- Model size was the limiting factor, not inference latency, so all neural networks tested were designed to have a file size no larger than 160KB.



3.2 Accuracy Testing

- To ensure testing was consistent and representative, 5-fold cross-validation [3]; dropout regularization [4]; between-subjects validation; parameter tuning; and variable neural network training times were used.



- CNN-LSTM performed best, still suffered poor accuracy and had overfitting.

- When using within-subjects validation, CNN-LSTM achieved 79% accuracy, suggesting that there is much variation in the dataset, and that some candidates are not representative of the dataset as a whole.

- More work needs to be done to better integrate the system, such as applying preprocessing to the dataset, but CNN-LSTMs show promising results.

[1] H. Duan, M. Huang, Y. Yang, J. Hao, και L. Chen, 'Ambient Light Based Hand Gesture Recognition Enabled by Recurrent Neural Network', IEEE Access, τ. 8, σσ. 7303–7312, 2020.
[2] Charles Arthur Nagler and William Merle Nagler. Reaction time measurements. Forensic Science, 2:261–274, 1973.
[3] D. Berrar, 'Cross-Validation', 01 2018.
[4] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(56):1929–1958, 2014.