Embedded AI Enabled Air-Writing for a Post-COVID World
Extended Abstract

Goedemondt, K.S.; Yang, J.; Wang, Q.

Publication date
2022

Document Version
Final published version

Published in
42nd WIC Symposium on Information Theory and Signal Processing in the Benelux (SITB 2022)

Citation (APA)
Goedemondt, K. S., Yang, J., & Wang, Q. (2022). Embedded AI Enabled Air-Writing for a Post-COVID World: Extended Abstract. In J. Louveaux, & F. Quitin (Eds.), *42nd WIC Symposium on Information Theory and Signal Processing in the Benelux (SITB 2022)* (pp. 67-68)

Extended Abstract: Embedded AI Enabled Air-Writing for a Post-COVID World

Koen Goedemondt, Jie Yang, and Qing Wang
Delft University of Technology, The Netherlands
Emails: {k.s.goedemondt@student.tudelft.nl, j.yang-3@tudelft.nl, qing.wang@tudelft.nl}

I. Introduction

Touchscreens and buttons had became a medium for virus transmission during the COVID-19 pandemic. We have seen in our daily life that people use tissues and keys to press buttons inside elevators, on public screens, etc. In the post-COVID world, *touch-free* interaction with public touchscreens and buttons may become more popular.

Motivated by the rise of visible light communication and sensing, we design a real-time embedded system to enable touch-free fingertip writing of the digits 0–9 with only *ambient light* and *simple photodiodes*. We propose an embedded deep learning model to learn the spatial and temporal patterns in the dynamic shadow for air-writing digits recognition. The model is devised with a lightweight convolutional architecture such that it can run on a resource-limited device. We evaluate our model using the LightDigit dataset [1] and report the results in terms of accuracy and inference time.

LightDigit dataset. It is a new air-writing digits dataset collected by a researcher going through 70000 images in the MNIST dataset [2] and replicating them with air-writing and ambient light to obtain time-series information. The dataset contains 20880 instances of air-writing digits 0–9. Each instance has $500 \times 9 = 4500$ samples (i.e., samples per photodiode $\times$ number of photodiodes). For more details about the LightDigit dataset please refer to [1].

II. Embedded AI Algorithm

Data processing. The classification principle of our proposed algorithm is image processing using a convolutional neural network. Each instance in the LightDigit dataset is compressed into a 50×9 image (see Figure 1 for illustrations). Irrelevant samples in each instance are stripped from the beginning and the end. A sample is considered relevant if the variation of light across channels lies above a predetermined threshold, i.e., a sample with all channels (almost) equally lit will be removed. This is done to correct for different writing speeds, as a user will generally not be writing for the entire sampling time. Then, the samples are downsampled either by averaging samples into one or by simply keeping equally spaced samples, and removing the rest. In both cases, 50 samples are retained to form a 50×9 image. Finally, the image is globally normalized, instead of each channel independently. This is essential to compensate for continuously lit or dark channel, which would otherwise significantly distort the image.

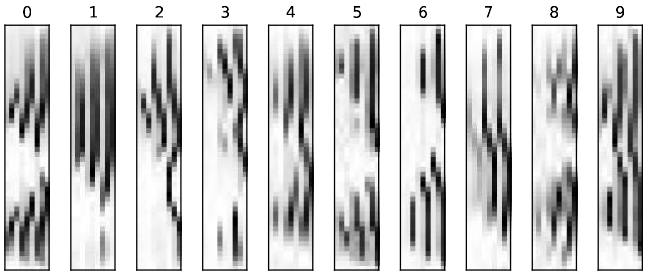

Fig. 1: Converting each air-writing digit to a 50×9 image.

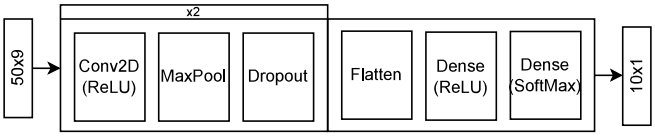

Fig. 2: Proposed model architecture which is optimized for various sizes. Note that dense softmax is the output layer.

Deep learning model. Our deep learning model is shown in Figure 2. It is based on the widely-used LeNet-5 architecture [3]. The goal is to keep the model as compact as possible while maintaining high accuracy. The final model contains two convolutional layers with ReLU activation followed by a max-pooling and dropout layer. The output of these layers is flattened and fed as input to a dense layer which then connects to the final output layer.

III. Implementation and Evaluation

We implement and run our embedded deep learning model on the NUCLEO-H743ZI2 STM32 board. This MCU board has an ARM-Cortex M7 CPU running at a maximum of 480 MHz, 1 MB SRAM and 2 MB flash. For detecting light, the system uses a 4×4 grid of OPT101 photodiodes, which are spaced 5 mm apart. They are sampled by the MCU through two MCP3008 ADCs at 100 Hz. We create the model in TensorFlow 2.0 and use TFLM [4] to port it the MCU. The model parameters are automatically quantized to 8-bit integer values, which decreases memory footprint as well as execution time. The model hyperparameters are optimized using the Hyperband algorithm [5] in keras tuner [6]. The amount of filters, kernel size and number of dense nodes were especially relevant. Rectangular kernels are found to perform best on this

TABLE I: Evaluation results of the within-subjects scenario.

Parameters	Dense nodes	Size (kB)	Accuracy	Inference time (s)	Inference time CMSIS-NN (s)
5k	96	9.9	0.885	0.72	0.08
8k	112	13.3	0.913	1.60	0.17
11k	80	16.2	0.936	1.48	0.16
14.7k	128	20.3	0.926	2.20	0.22

dataset. Both reference kernels[1] and CMSIS-NN [7] kernels were used when for determining inference time. CMSIS-NN is a deep learning library created by ARM consisting of highly optimized kernels specifically for Cortex M processors.

The evaluation results are presented in Table I. We consider a *within-subjects* scenario where we shuffle the data collected from 24 participants and split into training (80%) and test (20%). In addition, the training set is augmented with the simulated data from LightDigit. We observe that by converting the original time-series data from the LightDigit dataset to images and using a convolutional neural network with optimized hyperparameters, the amount of parameters could be reduced to 11k. After model quantization, this results in an embedded deep learning model of only 16 kB. The achieved accuracy is about 93.6% and the inference time using CMSIS-NN is only 0.16 seconds, running on the resource-limited ARM Cortex M-7.

IV. Challenges and future work

Through experimenting in various different light conditions, several major challenges were found which are listed below.

Clipping photodiodes. The OPT101 photodiodes are too sensitive when connected with the standard 1 MΩ feedback resisitor and start clipping around 600 lux, depending on the spectrum of the captured light as the photodiodes respond different to red, green and blue light. In bright conditions this may cause the shadow area not to be dark enough, resulting in loss of information. The sensitivity of the photodiode can be reduced by using a smaller value feedback resistor.

Distorted shadows. Multiple light sources cause the shadow cast by the users' finger to become distorted. This is especially problematic since it is impossible to train the model for every possible configuration. We attempt to tackle this problem in two different ways. 1) An algorithm is proposed to extract hand movement from relative changes in the shadow, and train a new model on this data. The purpose of this algorithm is to more accurately model hand movement over the sensing area, avoiding the problem of brightness differences and light source distribution. 2) An autoencoder based deep learning approach in order to reconstruct distorted images. Autoencoders have been used successfully in image denoising to increase model robustness [8] and image restoration [9]. We intend to experiment with modifying images from the source dataset in such a way, they can be used to train an autoencoder. The hypothesis is that this autoencoder can then be used to increase robustness in different lighting conditions.

Trigger sampling. For practical application, the system does not yet have way to start sampling automatically. This is intended to be solved by fitting the system with a APDS-9930 short range IR proximity sensor. When a user moves their hand over the sensing area, the sensor will send an interrupt to the MCU to trigger sampling.

[1]In this context, kernel refers to the operations between tensors such as convolution.

V. Conclusion

LightDigit previously used a resource-intensive LSTM model for classification, which is too heavy to run on MCUs. By converting the time-series data to an image and using an optimized CNN, the amount of parameters could be reduced to 11k, resulting in a final model of 16 kB. Running on an ARM Cortex M-7 the inference time is 0.16 seconds using CMSIS-NN, while maintaining an accuracy of 93.6%. The inference time is $9\times$ faster compared to reference kernels. Due to challenges resulting from variable light conditions, practical application of this system will require a more robust model. Since the inference time with current models is relatively fast, future work will include experimenting with larger models and more advanced preprocessing. We plan to experiment with denoising autoencoders and creating an improved algorithm to more precisely locate the finger of the user in space.

References

[1] T. Delft, "Lightdigit dataset," 2021. [Online]. Available: www.dropbox.com/s/lbllt66mnunj22g

[2] L. Deng, "The MNIST database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, 2012.

[3] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[4] A. J. et. al., "Tensorflow lite for microcontrollers," https://github.com/tensorflow/tflite-micro, 2022.

[5] L. Li and et. al., "Hyperband: A novel bandit-based approach to hyperparameter optimization," *Journal of Machine Learning Research*, 2018.

[6] T. O'Malley, etc., "Kerastuner," https://github.com/keras-team/keras-tuner, 2019.

[7] L. Lai, N. Suda, and V. Chandra, "Cmsis-nn: Efficient neural network kernels for arm cortex-m cpus," 2018. [Online]. Available: https://arxiv.org/abs/1801.06601

[8] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 1096–1103. [Online]. Available: https://doi.org/10.1145/1390156.1390294

[9] M. Suganuma, M. Özay, and T. Okatani, "Exploiting the potential of standard convolutional autoencoders for image restoration by evolutionary search," *CoRR*, vol. abs/1803.00370, 2018. [Online]. Available: http://arxiv.org/abs/1803.00370