ELEC4342 Embedded Machine Learning
University of Hong Kong
Spring 2024

Project: Hand Gesture Recognition on Raspberry Pi

Report Deadline: Sunday, April 21st, 11:59 PM

I Objective

In this project, you will create a hand gesture recognition system deployed on a Raspberry Pi. You will collect a dataset, design a neural network, apply quantization and pruning techniques, and compare the performance of different methods. After the implementation, you will give a brief report in class (on April 22nd and April 25th).

II Requirements

II.1 Dataset Collection

II.1.1 Basic task

The basic tasks should be implemented with selected hand gesture dataset. You can get the data set from: https://www.kaggle.com/datasets/datamunge/sign-language-mnist/data

II.1.2 Bonus

For bonus, you can collect your own dataset for test and real-time demo with the requirements below:

- Collect a hand gesture dataset containing at least three classes: scissors, rock, and paper.
- The dataset should be suitable for classification tasks and have sufficient samples for training and testing.

Hint: Use the Raspberry Pi camera to capture the images of the gestures (You can record a video of gestures using Raspberry Pi and select images from the video). Make sure that each class contains over 2000 images and that your gestures are always the main components of the images. Then divide your images into train set and test set.

II.2 Neural Network Design and Optimization

- Design or select a suitable neural network architecture for hand gesture classification.
- Apply the quantization and pruning techniques learned in class to optimize the network for deployment on the Raspberry Pi.
- Document the design choices and optimization steps in your report.

II.3 Deployment

- Use mature compilation tools such as TVM, MNN, NCNN, or MQBench to deploy the original trained network on the Raspberry Pi.
- Compare the inference speed of different deployment methods in II.2 and II.3. Analyze the reasons for performance differences.
- Select the best-performing method for the final demo.

III Real-time Demo

III.1 Basic task

Show the real-time performance (including accuracy on test dataset, FPS of inference and memory consumption, etc.) in the final presentation.

III.2 Bonus

- Utilize the Raspberry Pi camera to capture real-time video feed.
- Implement a demo that performs hand gesture recognition on the captured frames.
- Display the recognized gesture class and the corresponding frames per second (FPS) on the output video stream.
- Record a video (about 1 min) to show the performance of your demo.

IV Experiment Report and Submission

- Write a concise report in **IEEE two-column conference format (maximum 6 pages)** describing the project details.
- Include information about the dataset, network design, quantization and pruning implementation, and deployment results.
- Show the details and basic tests to demonstrate the effectiveness and authenticity of your sparsity and quantization methods.
- Analyse and compare the performance of different deployment methods.
- Present the training process, the results, and any insights gained during the project.
- Show the percentage of each group member's contribution to the entire project.

Put your source code with the captured video (about 1 min) in a zip file. Submit the zip file and your report (in PDF) to the Moodle. (Note that submission is required from just one member of each group.) The template of Latex or Word can be get from:
https://template-selector.ieee.org/secure/templateSelector/publicationType

V Evaluation Criteria

The final grade of the project would be calculated as:

$$\#\text{Your\_Grade} = \alpha \times \#\text{Group\_Grade} \times \#\text{Num}_{\text{group\_member}} \times \gamma \tag{1}$$

where γ denotes the percentage of your individual contribution to the overall project, whereas α assumes a value of 1.1 for two-student group and reduces to 1 when the group comprises three students.

V.1 Basic Function Check

- The basic implementation of gesture recognition with neural network.
- The implementation of pruning.
- The implementation of quantization aware training.
- The implementation of compiler accelerated network.

V.2 Report Check

The report should show the results and analysis of tasks in the Requirements. It would also be evaluated by its:

- Organization and Clarity
- Presentation and Formatting

V.3 Real-time Demo

V.3.1 Basic check

The real-time accuracy and speed of implemented neural network on selected dataset.

V.3.2 Bonus

- The real-time demo and display with the camera captured image
- The speed of inference of optimized network
- Rough accuracy of recognition