

EE 4065 – Embedded Digital Image Processing Final Project

Due: Final's data posted on BYS

- Each group will submit a printed formal report on the course final date.
- The report should consist of detailed explanation of the method considered, codes, results and evaluation of the results.
- There will be an oral presentation on the course final date.
- Responsible AI usage is allowed both in report preparation and code formation.

Question 1 (20 points) We will form a thresholding function specific to the following scenario. There is one bright object in the image acquired by ESP32 CAM. Background object pixels are darker compared to the object pixels. The object to be detected by thresholding has 1000 pixels. Therefore, the thresholding result should be such that the object is extracted based on its size.

- a- (5 points) Form the complete Python code to run on PC to perform this operation.
- b- (15 points) Form the complete C code to run on the ESP32 CAM to perform this operation.

Question 2 (40 points) We will perform handwritten digit detection via YOLO in this question. You will form the training and test data sets manually by writing digits (from 0 to 9) on paper. You will train the system on PC. Here, you can benefit from the STMicroelectronics web sites below.

<https://github.com/STMicroelectronics/stm32ai-modelzoo-services>
<https://github.com/STMicroelectronics/stm32ai-modelzoo>

- a- (40 points) Form the complete inference module to run on the ESP32 CAM. Test with actual images.
- b- (10 points) If you skip the ESP32 CAM implementation and go with Python implementation to run on PC, you can use this part only.
- c- (10 points) If you pick an available YOLO model running on ESP32 CAM, then you will use this part only.

Question 3 (20 points) Implement the upsampling and downsampling operations to run on the ESP32 CAM module.

- a- (10 points) perform upsampling with a given value.
- b- (10 points) perform downsampling with a given value.

Make sure that your modules can handle upsampling and downsampling by non-integer values such as 1.5 and 2/3.

Question 4 (20 points) Implement handwritten digit recognition with more than one available model such as SqueezeNet, EfficientNet, MobileNet, ResNet to run on the ESP32 CAM module.

- a- (15 points) Implement each module to run alone on the ESP32 CAM module.

- b- (5 points) Merge the results of the above modules to come up with a single merged/fused recognition result.

Question 5 BONUS (40 points) Implement handwritten digit detection on the ESP32 CAM module via

- a- (20 points) FOMO with Keras.
- b- (20 points) SSD+MobileNet.

Either answer the two parts or leave this bonus question. You can benefit from the web sites below.

<https://github.com/bhoke/FOMO>

[https://docs.edgeimpulse.com/studio/projects/learning-blocks\(blocks\)/object-detection/fomo](https://docs.edgeimpulse.com/studio/projects/learning-blocks(blocks)/object-detection/fomo)

Question 6 BONUS (60 points) Implement MobileVit to run on the ESP32 CAM module for handwritten digit detection. You can benefit from the web site below.

<https://keras.io/examples/vision/mobilevit/>