TinyML for Efficient Detection, Classification, and Control

Michael Ibrahim, Vicente Arroyos, Kyle Johnson, Emmanuel Azuh

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

AVELA

Supported by NASA Grant 80NSSC20M0104

WASHINGTON SPACE GRANT

Introduction

> MilliMobile[1], a battery-free autonomous robot, operates solely on harvested solar or RF energy. This design can integrate various sensors, making it adaptable for multiple applications like:

Space **Search & Rescue** **Agriculture**

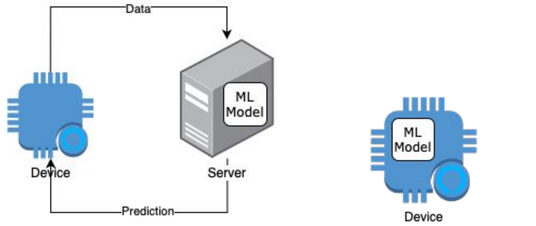
Fig. 1: Visualized applications of the MilliMobile

> With near-gram mass and penny-sized design, MilliMobile can closely interact with a variety of insects enabling detailed observation in their natural environments. By integrating a machine learning solution, MilliMobile can be used for insect detection, classification, and active seeking control, offering a novel approach to researching a myriad of insect species.

Fig. 2a) MilliMobile on the tip of a finger
Fig. 2b) MilliMobile moving across a flat plane
Fig. 2c) MilliMobile moving across a leaf
Fig. 2d) Millimobile moving across soil

TinyML

> TinyML (Tiny Machine Learning) will enable the MilliMobile to perform real-time insect detection, classification, and active insect seeking control behaviors directly on-device.



Traditional Approach **TinyML Approach**

> Using TinyML, MilliMobile can utilize a model trained on an insect database, paired with the OV7675 camera, to identify and track insects autonomously.

Fig. 3) Evaluation Board[2] with Arduino Nano 33 BLE (blue) and OV7675 attached on top (green)

> Due to the 1 MB flash and 256 KB RAM constraints of MilliMobile's nRF52840 MCU, we are limited in the number and quality of photos that can be used for training the model (128x128 pixels).

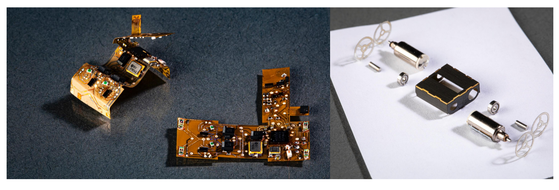


Fig. 4) Compact PCB limits MCU compute, impacting image quality and model training

Training the Model

> We use the *Zenodo Insect Classification Dataset V2* to train our model
> The model was first trained on Google Colab, then quantized and converted to C using Unix's hexdump command.

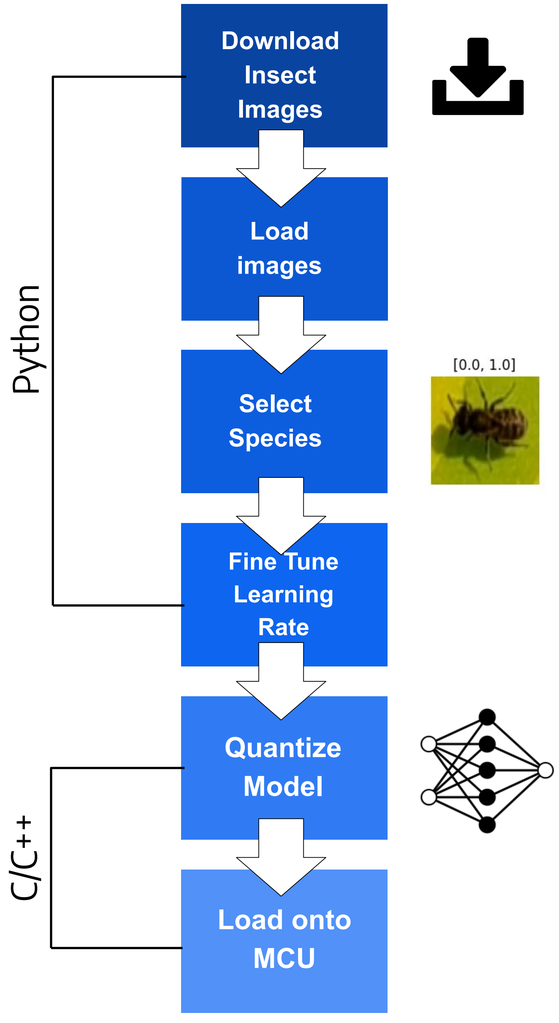


Fig. 5) TinyML training process for nRF52840 MCU

```
Model: "sequential_1"
```

Layer (type)	Output Shape	Param #
functional (Functional)	(None, 1, 1, 256)	218,544
reshape_1 (Reshape)	(None, 256)	0
dropout (Dropout)	(None, 256)	0
flatten (Flatten)	(None, 256)	0
dense_2 (Dense)	(None, 2)	514

```
Total params: 219,058 (855.70 KB)
Trainable params: 213,586 (834.32 KB)
Non-trainable params: 5,472 (21.38 KB)
```

Fig. 6) TinyML output after training

Future Works

> The current dataset is very large. We plan to finetune the model by:

- Reducing the number of insect species being trained (model classes)
- Changing input from the camera from RGB to grayscale
- Reduce image size to 96x96 or 64x64

> Introduce control to be able to track and follow an insect once detected.

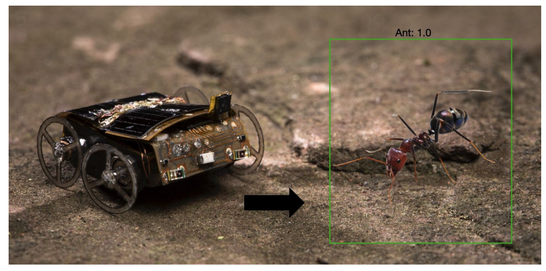


Fig 7) Mock-up of the MilliMobile identifying and then actively seeking and following an ant.

References

> [1]Kyle Johnson, Zachary Englhardt, Vicente Arroyos, Dennis Yin, Shwetak Patel, and Vikram Iyer. 2023. MilliMobile: An Autonomous Battery-free Wireless Microrobot.

> [2]Botland. 2024. Arduino Tiny Machine Learning Kit with Arduino Nano 33 BLE Sense Lite.

Learn More About MilliMobile