Neural Networks on the STM32 with STM32Cube.AI
The Key Steps Behind Neural Networks

1. Capture data
2. Train NN Model
3. Clean, label Data
   Build NN topology
4. Convert NN into
   optimized code for MCU
5. Process & analyze
   new data using trained NN
Form Factor Hardware to Capture and Process Data

Capture data

Process & analyze new data using trained NN

www.st.com/SensorTile
www.st.com/SensorTile-edu
Form Factor Hardware
AI IoT Node for More Connectivity (Q1 2019)

Capture data

More debug capabilities
• Integrated ST-Link/V2.1
• PMOD extension connector
• Arduino Uno extension connectors

Sub-1GHz + NFC Dynamic NFC Tag

Wi-Fi

Process & analyze new data using trained NN

https://www.st.com/iotnode
Collecting Data & Architecting a NN Topology

Services provided by Partners

Capture data

Clean, label Data
Build NN topology

ST tools to support

ST BLE Sensor mobile phone application
Collect and label data from the SensorTile.

Selected partners
Neural Networks engineering services support. Data scientists and Neural network architects.
Input your framework-dependent, pre-trained Neural Network into the STM32Cube.AI conversion tool

Automatic and fast generation of an STM32-optimized library

STM32Cube.AI offers interoperability with state-of-the-art Deep Learning design frameworks

Train NN Model

Process & analyze new data using trained NN

Convert NN into optimized code for MCU
STM32CubeMX Extension

STM32Cube.AI Interoperability

- Easy add of user code via public API interfaces
- Sensor agnostic
- RTOS agnostic or bare metal
- Select your IDE:

ONNX

- Caffe2
- Chainer
- mxnet
- PyTorch
- Microsoft CNTK

Supported soon

Train NN Model

Convert NN into optimized code for MCU
ST Toolbox for Neural Networks

More Than Just a Conversion Tool

- Function packs for **quick prototyping**
- **Audio** and **motion** examples
- **STM32 Community** for support and **idea** exchange
- **Dedicated** topic for Neural Networks

Process & analyze new data using trained NN

Convert NN into optimized code for MCU
Audio Scene Classification (ASC)
Audio Example in FP-AI-SENSING1 Package

Embedded audio

Labelling controlled by smartphone application

Data stored on the device SD card for future learning

Indoor, Outdoor, In vehicle

Embedded audio

Inferences running on the microcontroller

NN & example dataset provided

Inference result displayed on mobile app
Human Activity Recognition (HAR)
Motion Example in FP-AI-SENSING1 Package

- Embedded motion
- Labelling controlled by smartphone application
- Data stored on the device SD card for future learning

Stationary, walking, running, biking, driving

- Embedded motion
- Inferences running on the microcontroller
- Inference result displayed on mobile app

5 classes

NN & example dataset provided
STM32 Solutions for AI
More Than Just the STM32Cube.AI

An extensive toolbox to support easy creation of your AI application

- **AI extension for STM32CubeMX**
  To map pre-trained Neural Networks onto the STM32

- **SensorTile reference hardware**
  To run inferences or data collection

- **Mobile phone application**
  To collect and label data
  To display the result of inference processing on the STM32

- **Function packs for Quick prototyping**
  Audio and motion examples

- **STM32 Community**
  With dedicated Neural Networks topic

- **ST Partner Program**
  With a dedicated group of Partners providing Neural Networks engineering services
  Data scientists and Neural network architects

- ... And more coming!

https://www.st.com/STM32CubeAI
www.st.com/STM32CubeAI