FUTURE ELECTRONICS SYSTEM DESIGN CENTER

SENSIML - GESTURE RECOGNITION ML/AI

JANUARY 13, 2022

Revision 1.2

Table of Contents

Table of Figures

Table of Videos

Overview

In this tutorial, we are going to use the SensiML Analytics Toolkit to build a hand gesture application that will run on the STM STWIN and Future Electronics Compagno board using the Panasonic GridEYE Sensor.

Objective

- Demonstrate how to collect and annotate a high-quality dataset of hand gestures using the Panasonic GridEYE 8x8 thermal array sensor on the Future Compagno board using the SensiML Data Capture Lab
- Build a data pipeline to extract features in real-time on your target device
- Train a Classification model using SensiML AutoML
- Convert the model into a Knowledge Pack
- Perform live validation of the Knowledge Pack running in the SensiML Open Gateway with live data streaming from the boards
- Compile the SensiML knowledge pack library into the firmware and test the classifier on-device using the OpenGateway Interface

Prerequisites

- The Hardware
  - STM STWIN board (STEVAL-STWINKT1B)
  - Future Electronics Compagno board
- The Software
  - ST Micro Integrated Development Environment for STM32: STM32CubeIDE
  - Future Electronics **DC Engine**

    Documents > Future Created Material - Sync to Laptop > Tool Kit > Future Created Demo Kits > Compagno-SensiML

    - Future Electronics data collection/recognition firmware: **Future-SensiML_xx.zip**
      - Future Electronics data collection project file: **Future-SensiML_DataCollection.xx.bin**
    - Future Electronics **Compagno_GridEYE20Hz.ssf** configuration file for live data streaming over SimpleStreaming InferFace
    - Future Electronics Data Capture Lab project file: **FutureCompagnoGridEYE_xx.zip**
      - Future Electronics Recognition project file: **Future-SensiML_Reconition.xx.bin**

  - SensiML Analytics Toolkits
    - SensiML Data Capture Lab: Download, Documentation
    - SensiML Open Gateway: Download, Documentation
    - Create SensiML Community Edition account
      - **https://sensiml.com/plans/community-edition/**
  - Python 3.7 or 3.8

Capturing Sensor Data

For every machine learning project, the quality of the final product depends on the quality of your curated data set. Time series sensor data, unlike image and audio voice wake-word detection, are often unique to the application as the combination of sensor placement, sensor type, and event type greatly affects the type of data created. Because of this, you will be unlikely to have a relevant dataset already available, meaning you will need to collect and annotate your own dataset.

To help you to build a dataset for your application we have created the SensiML Data Capture Lab, which we are going to use for collecting and annotating data for different hand gestures.

Data Collection Firmware

In order to collect sensor data, you will need to flash the STWIN board with data collection firmware. For this tutorial, we have already built the data collection firmware which you can flash to the device using the STM32CubeProgrammer software. The device will reboot and load the new firmware.

Detailed instructions for building and flashing the firmware is available from Future Electronics Compagno User Guide on DC Engine.

Connecting to the Firmware

To begin data collection, we will use the SensiML Data Capture Lab. Log in to the SensiML Community Edition account you created. We have created an initial project for you to get started quickly. First:

1. Download this project and unzip it
2. Open the Data Capture Lab and select Upload Project

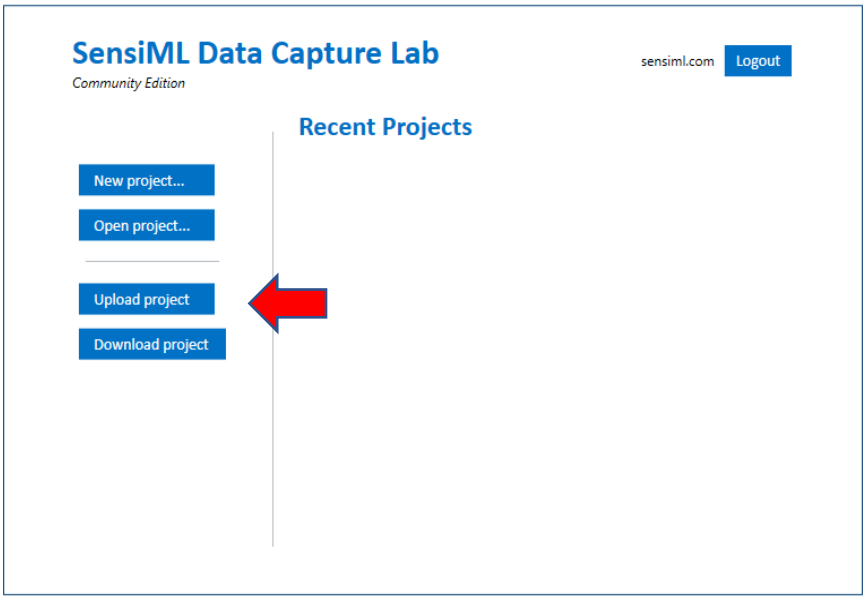


Figure 1. SensiML Data Capture Lab

3. Select browse and find the FutureCompagnoGridEYE.dclproj file inside of the FutureCompagnoGridEYE_xx.zip folder
4. Click the **Upload** button to upload the project to your account. You are now ready to start collecting your own Sensor Data

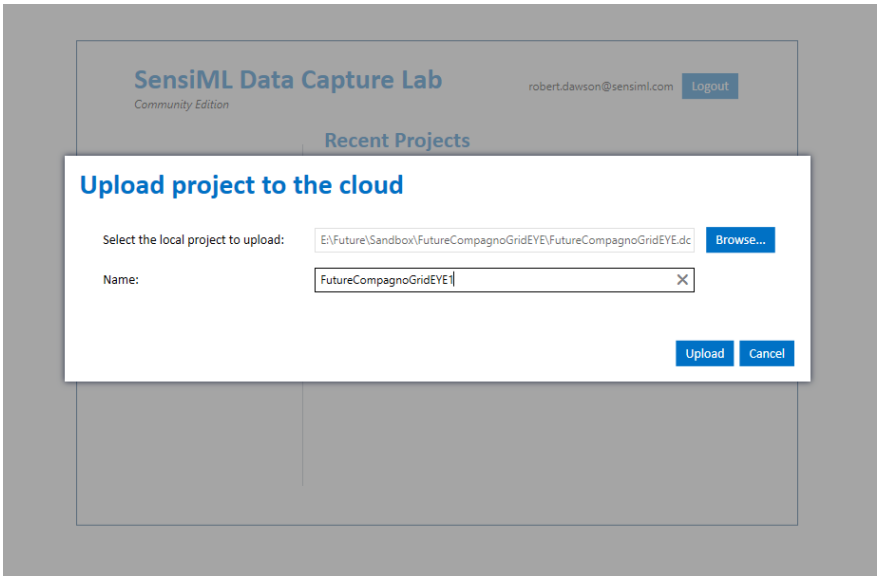


Figure 2. Upload Project

5. The Data Capture Lab opens to the Label Explore Screen, switch to the Capture Mode Screen by clicking the **Switch Mode** button and selecting **Capture** mode
6. If the Sensor Configuration is not available select EDIT->IMPORT DEVICE PLUG_IN, browse to the location of the Compagno_GridEYE20Hz.ssf
7. Select Sensor Configuration

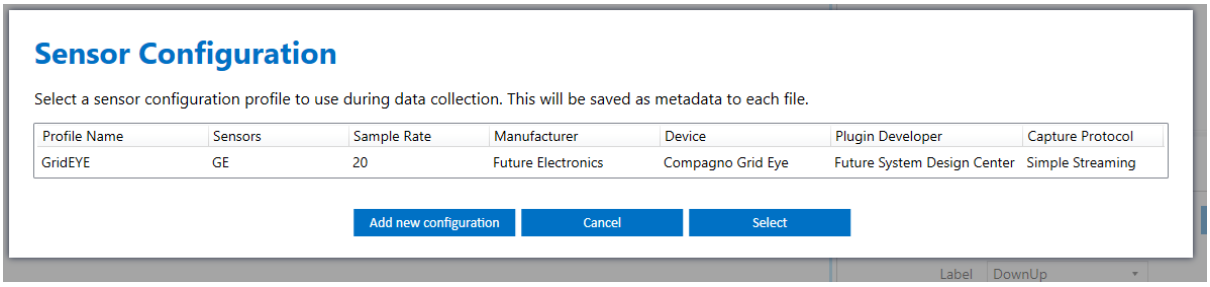


Figure 3. Sensor Configuration

8. Now that you are in Capture mode, we can connect to the sensor to record data.
9. First, switch the connection method to Serial Port

10. Next, click the Find Devices button and then the Scan button. After it scans it will display a list of devices. Select the one corresponding to your board and click the Connect button
11. Now you are ready to begin recording sensor data

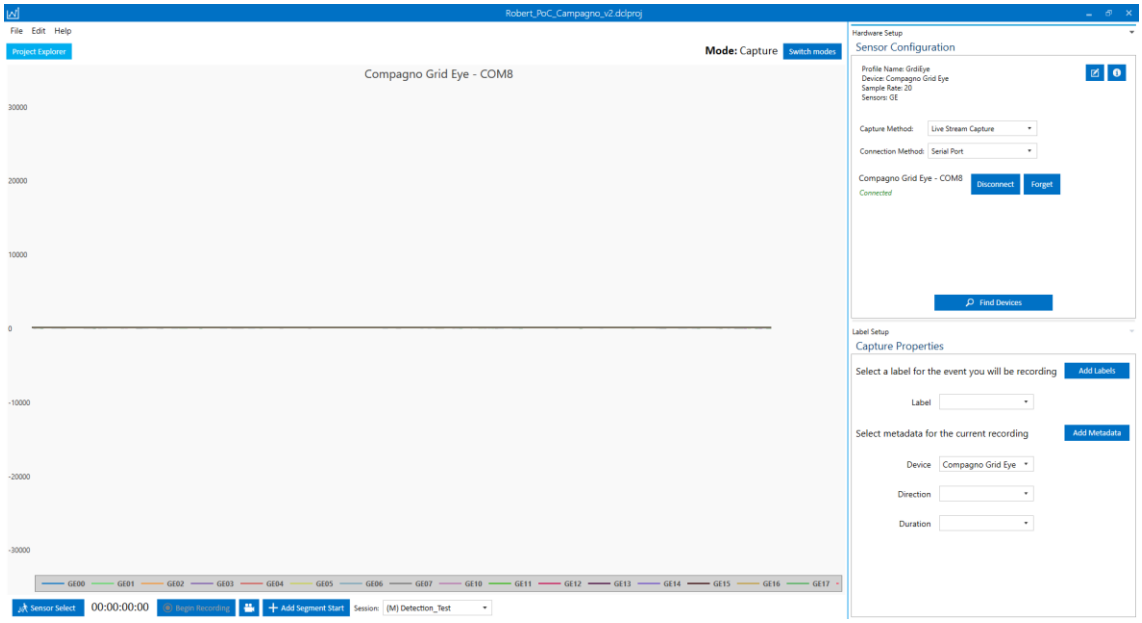
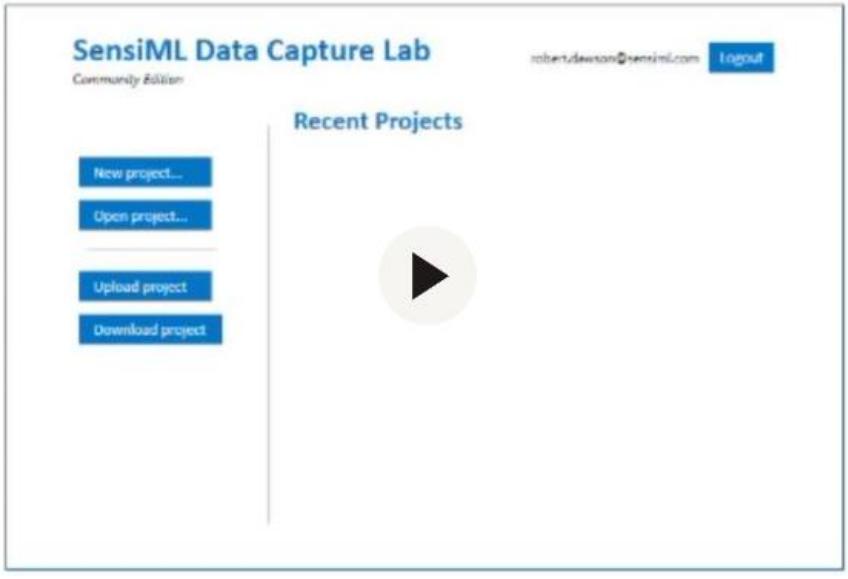


Figure 4. Capture Sensor Data

See tutorial video 1 below for details:



Video 1. **LiveStreamingGridEyeDataToSensiMLDCL.mp4**

Capturing Environmental Context

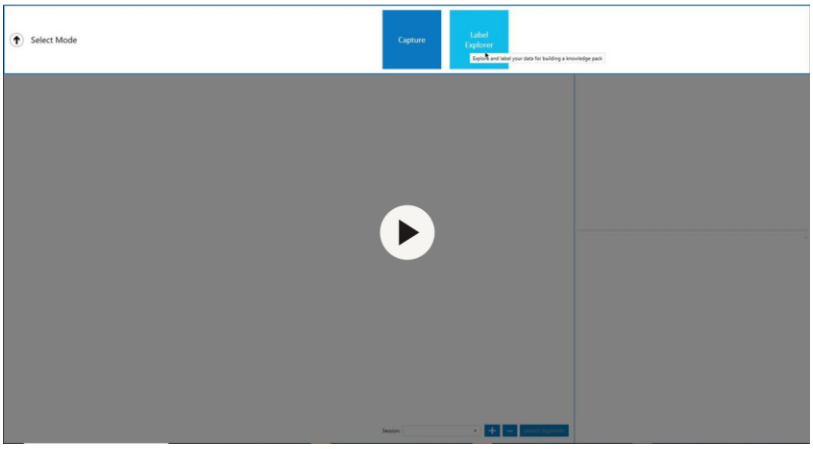
The data capture lab uses Labels to identify regions of sensor data as belonging to a specific class. For example, we would put a label around the region of sensor data that corresponds to the gesture of moving the handle over the GridEYE sensor arriving from the bottom and departing from the top and the opposite arriving from the top and departing from the bottom.

1. Identify which type of data you are going to collect
2. Select the option from the **Label** in the dropdown menu

Note: If you are going to be recording multiple types of events, I typically use the label **Unknown**. The file that is recorded will start with the name of the **Label** that you select. In addition to labels, it is also important to capture information about the environment. Capturing this contextual information enables you to build highly tailored models. Additionally, it makes it possible to diagnose why a model might be failing for subsets of your data. For example, in this gesture dataset, we captured the contextual properties **data collector** and **speed of movement**. This metadata can be useful in building models for subtypes of data as well as debugging problems with model accuracy.

You can capture the contextual information in the Data Capture Lab using the metadata properties. Metadata is applied to the entire captured file, so when you are creating your data collection strategy think carefully about what information you may need. Metadata can be created as a selectable dropdown or text field.

See tutorial video 2 below for details:

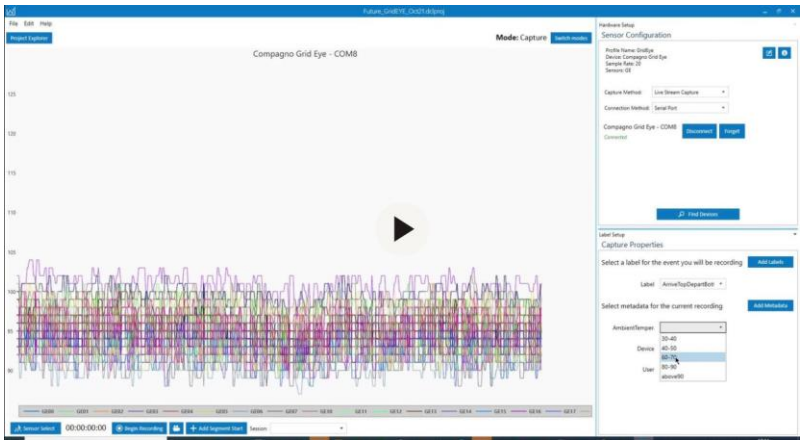


Video 2. ***SetupProject.mp4***

Recording data

The data collection firmware streams GridEYE sample data at a sample rate of 20Hz using a simple streaming interface over UART. The captured data will be saved locally to your computer as well as be synced up to the SensiML Cloud. This allows other members of your team who have permission to see and label your new captured file. Alternatively, if you already have a data collection method for your device, the Data Capture Lab can import CSV and WAV files directly.

See tutorial video 3 below for details:

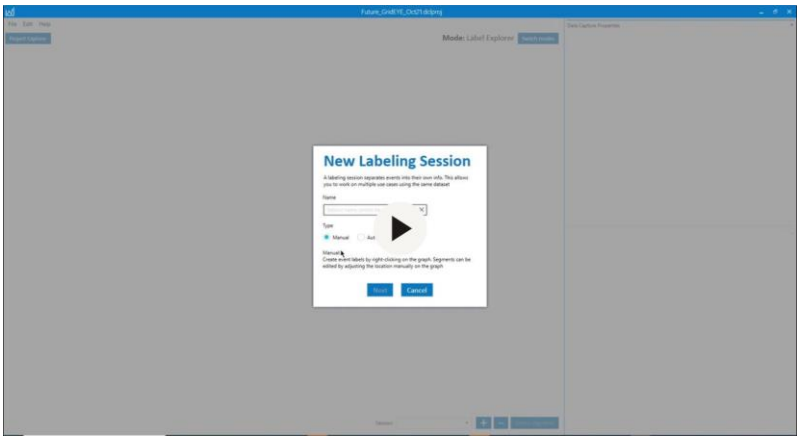


Video 3. ***DataCollection.mp4***

Labelling Sensor Data

Now that we have collected example sensor data, it is time to annotate the dataset by applying labels.

See tutorial video 4 below for details:



Video 4. ***LabelDataAutoSegment.mp4***

Building a Model

The SensiML Analytic Studio is where you can create a query to pull data into your model, build models using AutoML, validate model accuracy against held out test data and finally download your model as firmware code for the target device. For the next part of the tutorial, you will need to log into Analytic Studio (https://app.sensiml.cloud) .

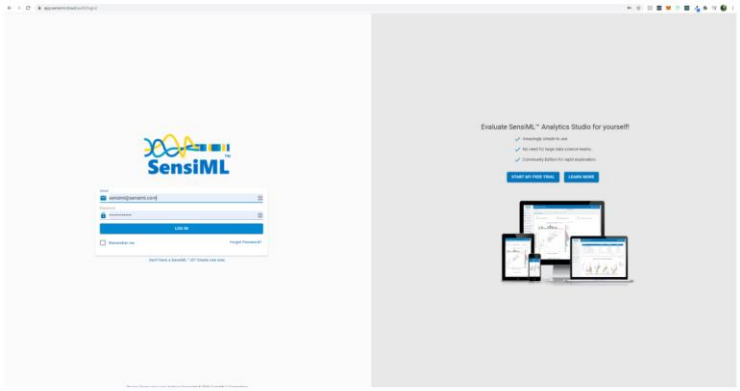


Figure 5. Analytic Studio

After logging in, open the FutureCompagnoGridEYE Project we created in the Data Capture Lab by clicking the open icon. This will set the FutureCompagnoGridEYE as the active project in the Analytics Studio.

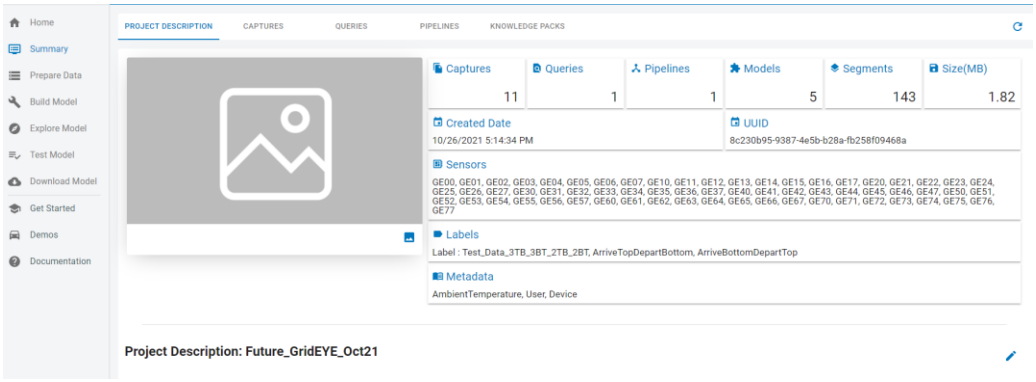


Figure 6. Analytic Studio Active Project

Prepare Data

Next, we can create a query that will generate the input data for our modelling. Click on the **Prepare Data** tab on the left and fill in the fields as shown in the image below.

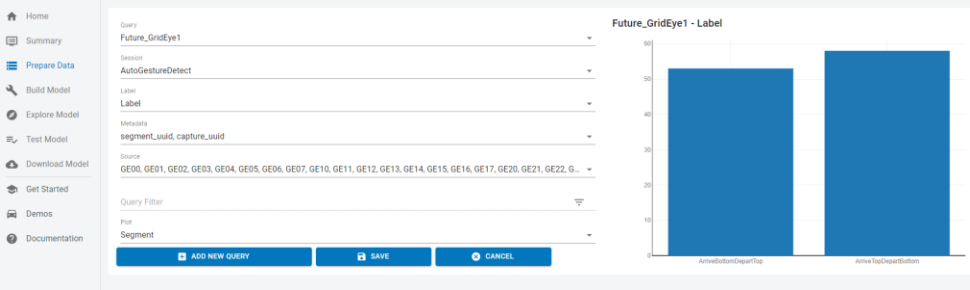
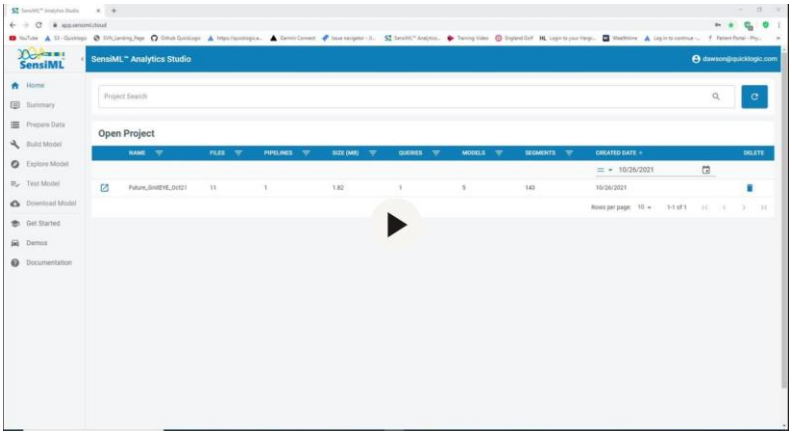


Figure 7. Preparing Data

Once you have filled in the fields, click save which will create the query that can then be used to build a model. You will select his query on the next screen. After you save the query, you will also see a graph that contains statistics about the size of the dataset that is going to be used.

See tutorial video 5 below for details:



Video 5. ***AnalyticStudioPrepareData.mp4***

Setting up a pipeline

Now that we have created a query, we can go to the build model screen. Click on the **Build Model** tab in the navbar on the left side. This will bring you to the model building screen. At this point, you have not yet created a pipeline. Click on the **+ Add New Pipeline** button and enter the name of your pipeline.

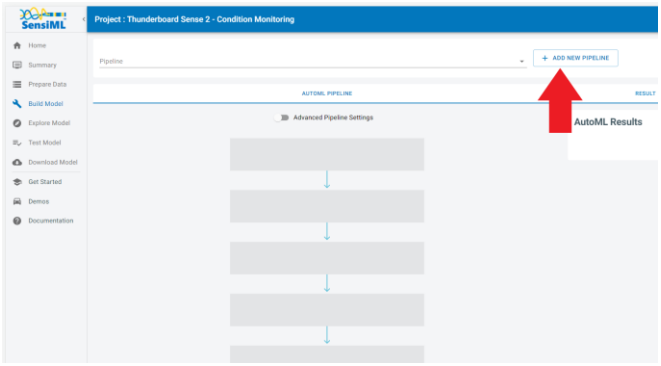


Figure 8. Adding Pipeline

Ensure that the Advanced Pipeline Settings slider is enabled.

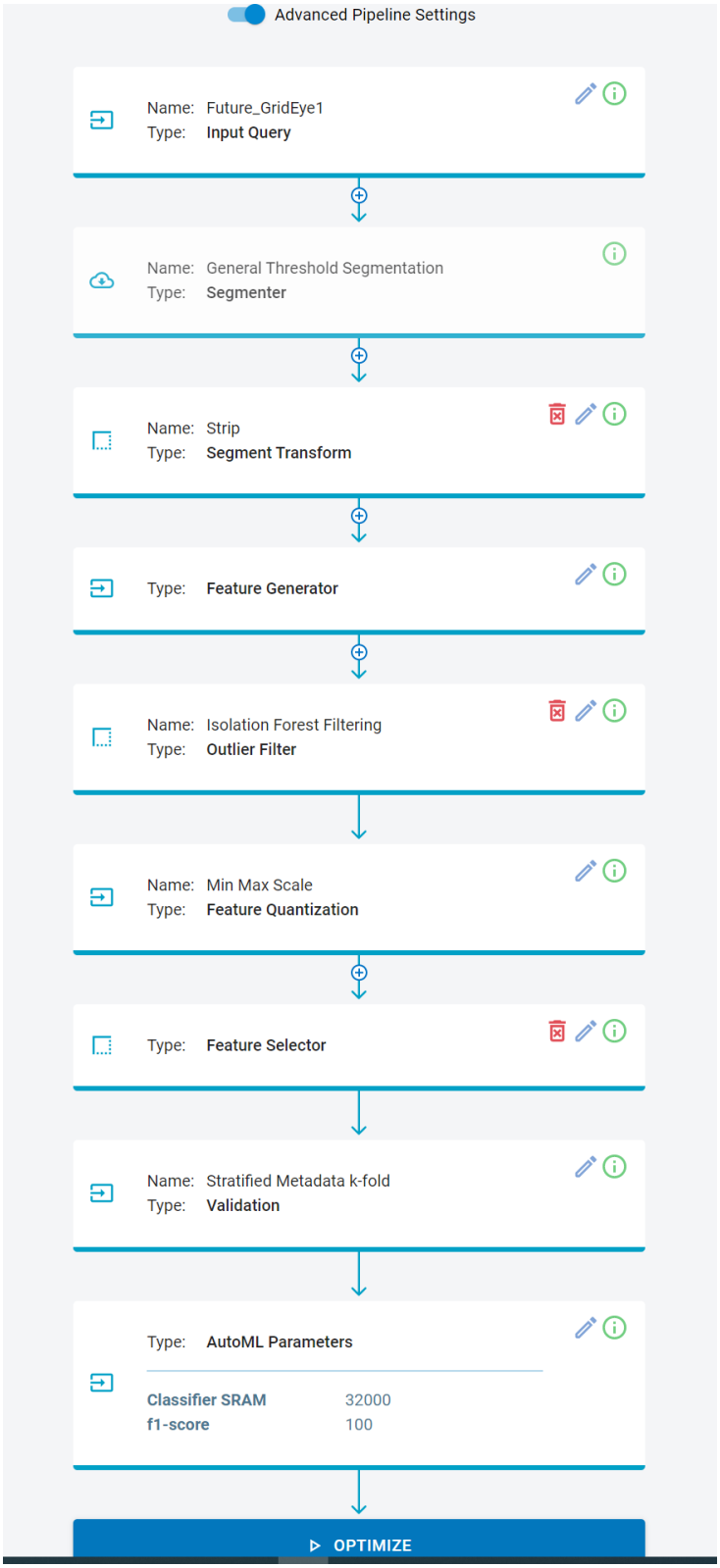


Figure 9. Advanced Pipeline Settings

Validation Settings

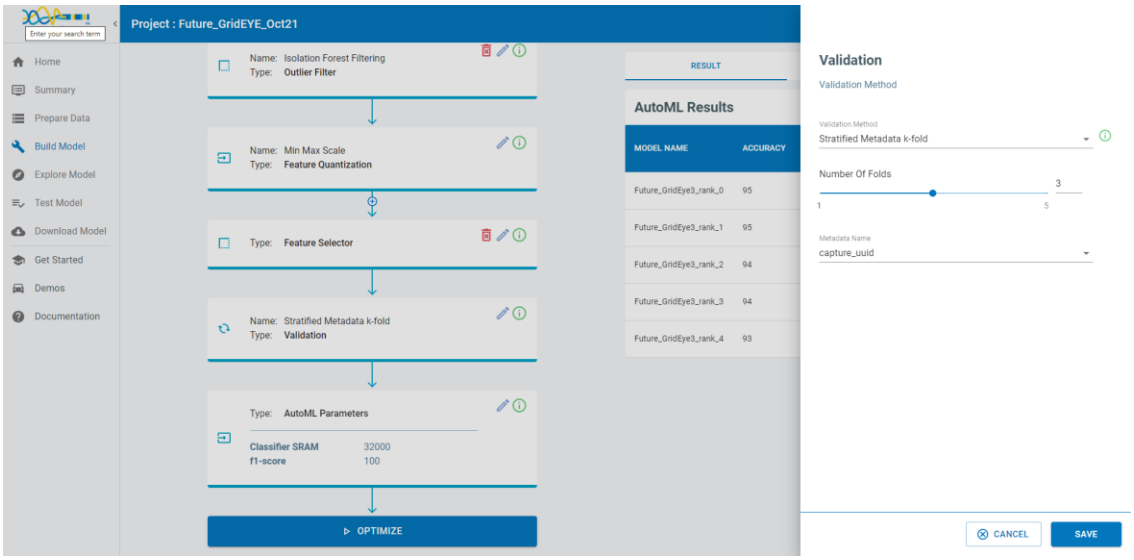
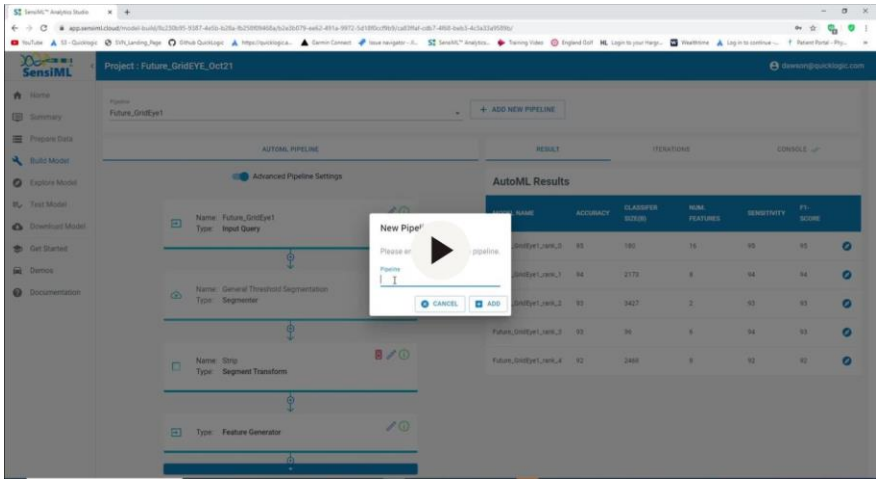


Figure 10. Validation Settings

At the moment we need to select Advanced Pipeline Settings, the plan is to add a Future GridEYE pipeline drop-down option, if this is not available follow the following steps to manually set up the pipeline.

See tutorial video 6 below for details:



Video 6. ***BuildModel.mp4***

Training the model with AutoML

AutoML is used to create a set of models within the desired statistical (accuracy, f1-score, sensitivity, etc.) and classifier size. As the algorithm iterates each optimization step, it narrows down the searching space to find a desired number of models. The optimization terminates when the desired model is found, or the number of iterations reaches the max number of iterations.

We take advantage of dynamic programming and optimizations for training algorithms to speed up the computation. This makes it possible to search for large parameter spaces quickly and efficiently. The results are ranked by the fitness score which considers the model's statistical and hardware parameters.

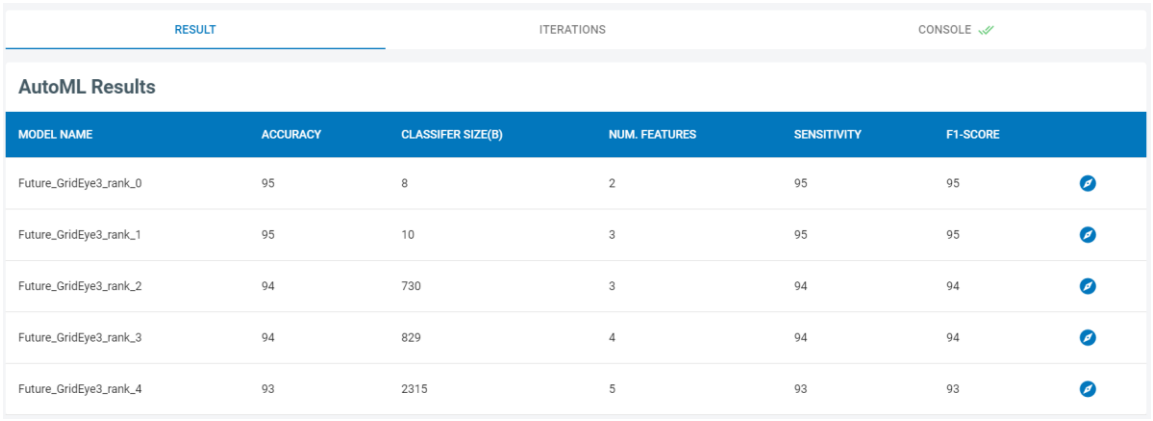


Figure 11. AutoML

Once the models have been generated, you can explore the details of the top five candidate models in the explore models tab. In this tab, there are visualizations and information about the models including, features, confusion matrix, model hyperparameters, and the Knowledge Pack training and inference pipeline.

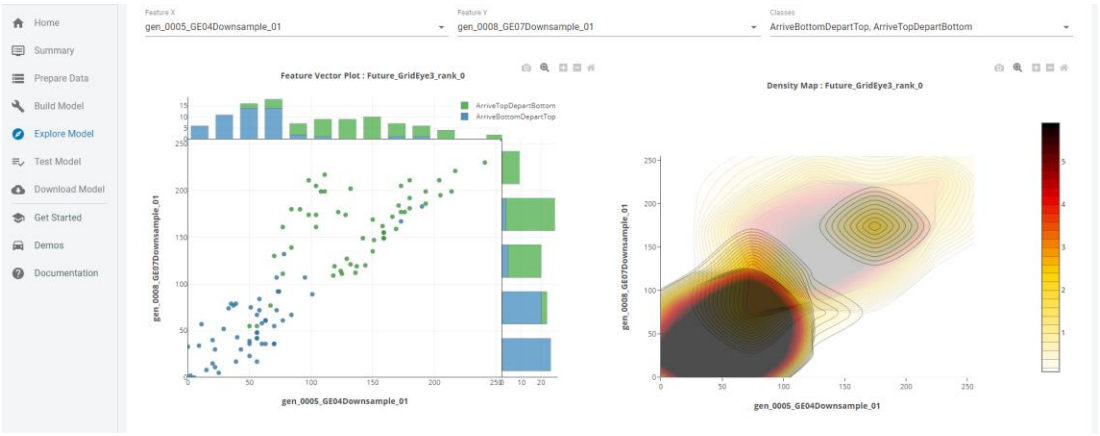


Figure 12. AutoML models

Validating the model against the Test Set

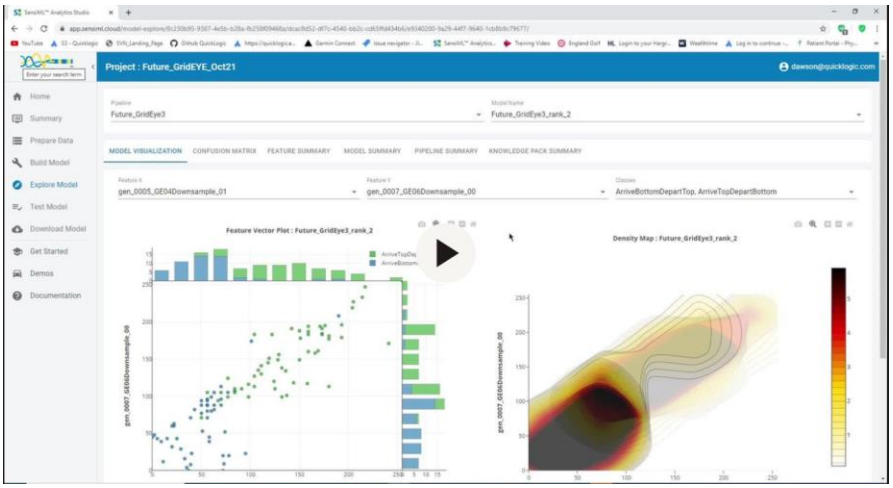
Before you flash the model to the device, you can test the model using the Test Modell tab. You can test against any of the captured data files. To do this:

1. Go to the Explore Model tab of the Analytic Studio
2. Select the pipeline you built the model with
3. Select the model you want to test
4. Add a filter to the Set column to only include Test data

5. Click on the three dots menu and the top left->Select All
6. Click the Compute Accuracy button to classify the selected capture files

The model will be compiled in the SensiML Cloud and the output of the model will be returned.

See tutorial video 7 below for details:



Video 7. ***FirstTestOfModel.mp4***

The Figure 13 shows the segment start and segment classified for all the detected events.
You can click on the Results button in the table to see more detailed information for how each of the captures were classified by the selected model.

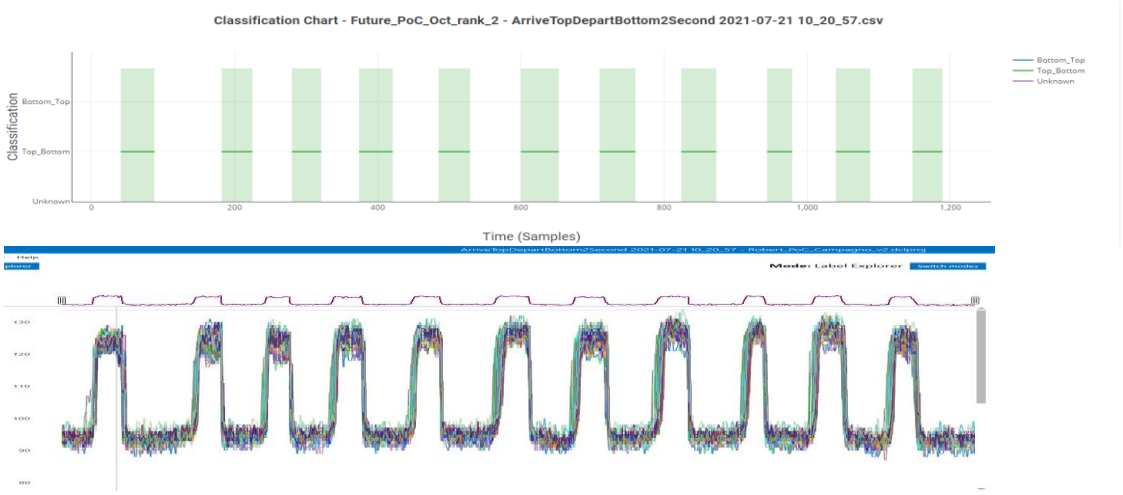


Figure 13. Validating the Model

Deploying Knowledge Pack Firmware in Open Gateway

Open Gateway

The Open Gateway implements the Simple Streaming Service protocol to enable forwarding data to the SensiML Data Capture Lab for recording and annotation. The Gateway supports connecting to sensor sources over a Serial, BLE, and TCP/IP connections. It also supports recording video and sensor data locally to the gateway.

Clone the github from https://github.com/sensiml/open-gateway
In the directory for the open-gateway open a command box

a. pip install -r requireds.txt
b. pip install appdirs

Note: On windows use python 3.7 or 3.8. You can install python from Microsoft Store for Windows 10

Data Collection Mode

1. Open a command window in the open-gateway-main directory, at the prompt type:

>python app.py

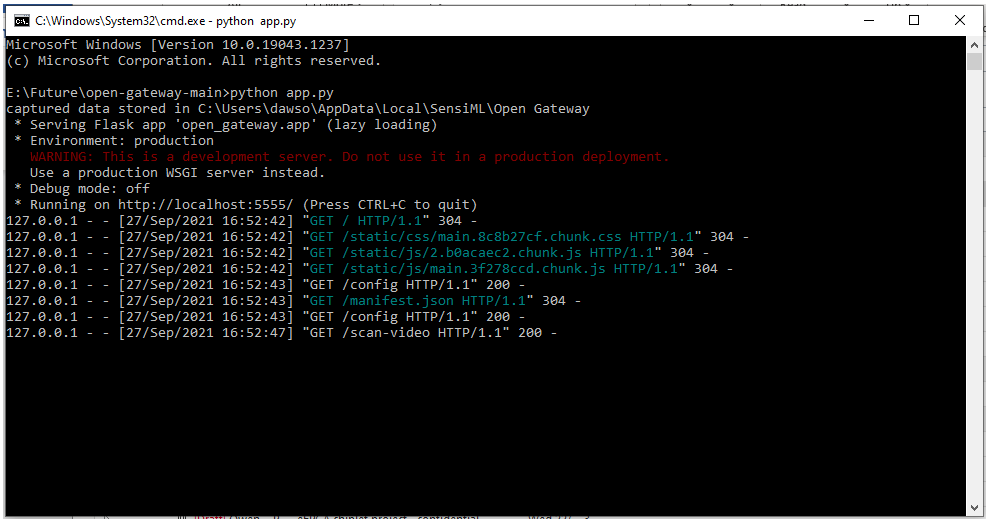


Figure 14. Commend Mode

This will open a browser

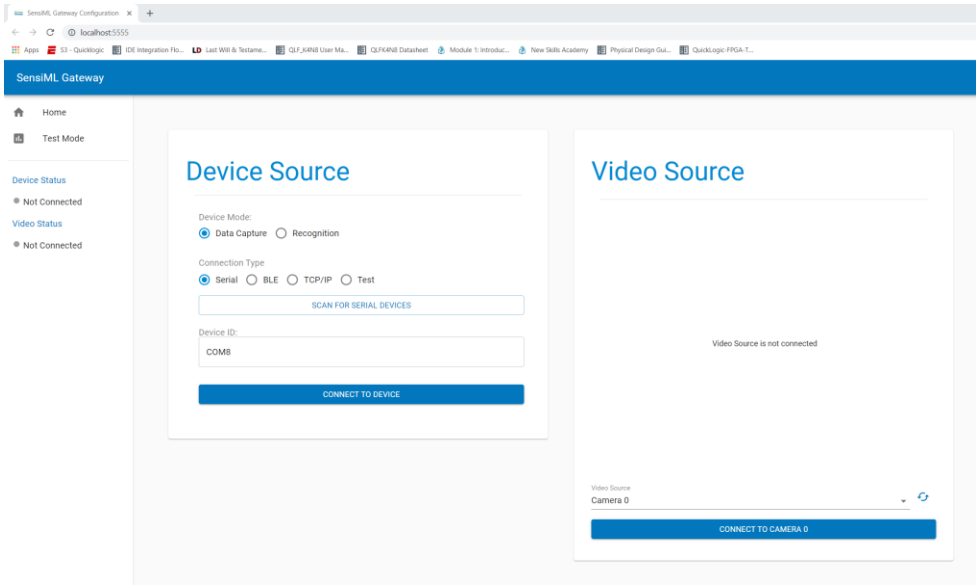


Figure 15. Open a Browser

2. Ensure that you have the following settings

   Device Mode -> Select Data Capture

   Connection Type -> Serial

3. Make sure that the STWIN and Compagno board is powered with the data collection binary
4. Scan for Serial Devices, select the USB Serial Device for the STWIN board, COM8 in example above then press SELECT in the bottom right-hand corner
5. Press CONNECT TO DEVICE
6. TEST MODE
7. To start streaming data select VIEW

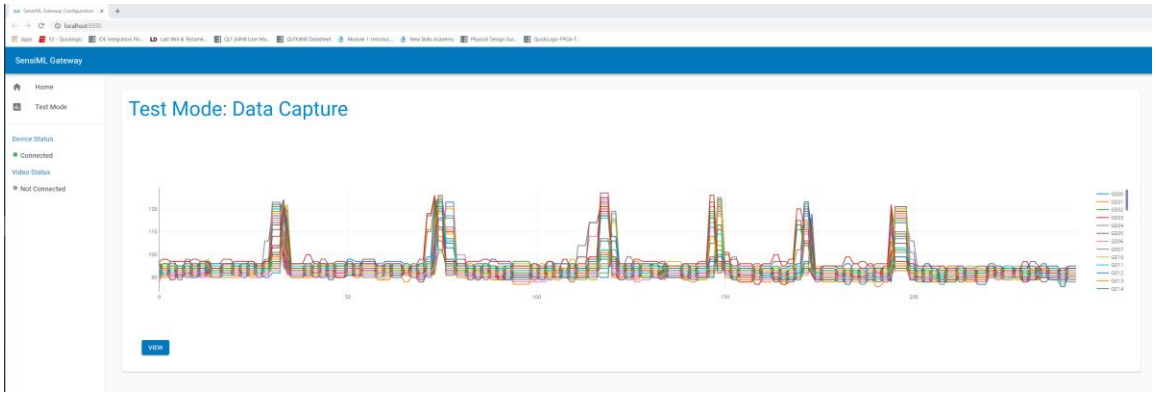


Figure 16. Test Mode

Running the New Knowledge Pack in the Gateway using Data Collection Firmware

The Open Gateway implements the Simple Streaming Service protocol to enable forwarding data to the SensiML Data Capture Lab for recording and annotation. The Gateway supports connecting to sensor sources over a Serial, BLE, and TCP/IP connections. It also supports recording video and sensor data locally to the gateway.

1. In the Analytic Studio -> Download Model, select HW Platform -> Windows x86_64

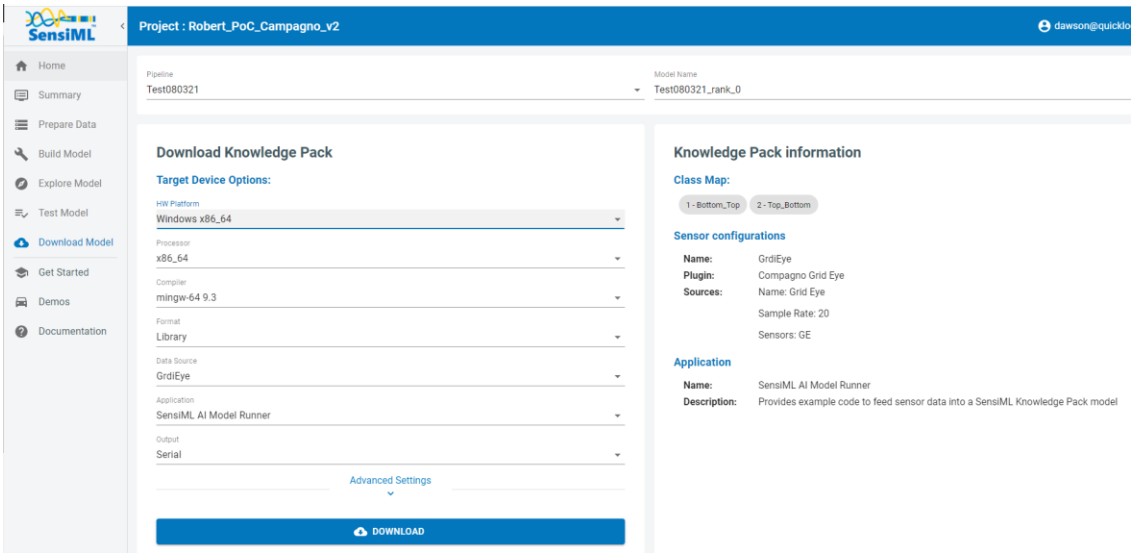


Figure 17. Download Model

2. Then DOWNLOAD, browse to the download directory you will find a zip file similar to:

   kp_e4e28e94-c7ba-4346-97c8-aec5a8ede494_Windows-x86_64_lib_9.3_p.zip

   (red text will be unique)

3. Unzip the knowledge pack referred to in step 2 and browse to the knowledgepack\libsensiml directory, copy the libsensiml.dll file into the \open-gateway-main\knowledgepack\libsensiml directory

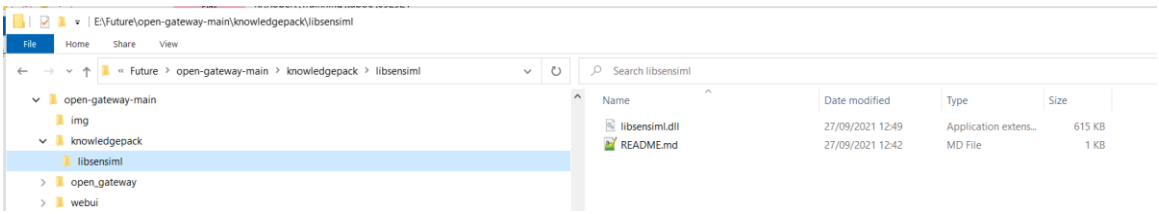


Figure 18. Unzip the Knowledge Pack

4. In the Windows Explorer browse to the open-gateway-main directory type cmd in the address window, which will open a command box in the open-gateway-main directory

Figure 19. Open a Command Box

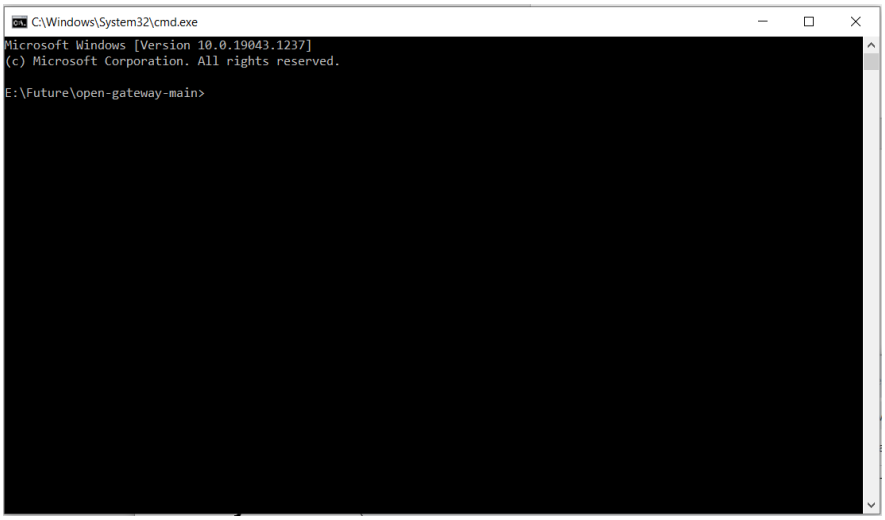


Figure 20. Command Box

5. At the prompt >python app.py -s knowledgepack/libsensiml

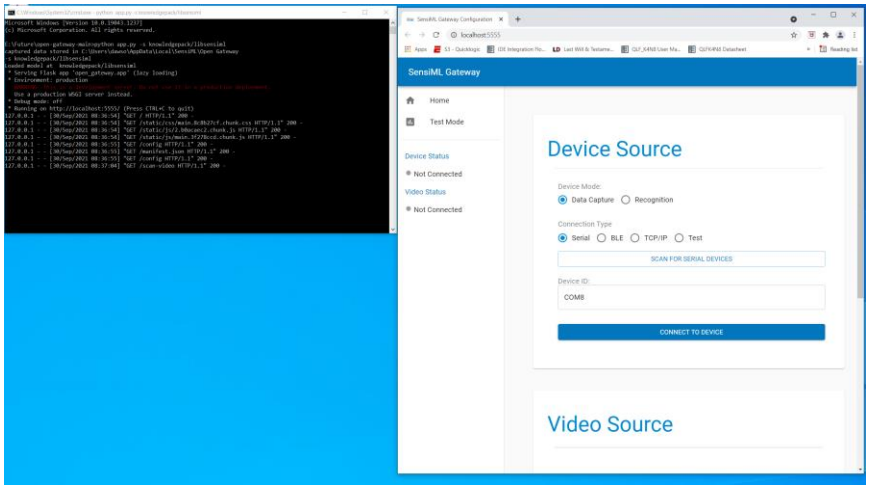


Figure 21. Running Knowledge Pack

6. SCAN FOR SERIAL DEVICES, find the Serial COM port of the STWIN board, In the example above it is COM8

7. CONNECT TO DEVICE
8. Select TEST MODE
9. Select VIEW, the command window will show the Classification detected, and the Web Page will show the data, if you see unexpected results you can record the data to debug

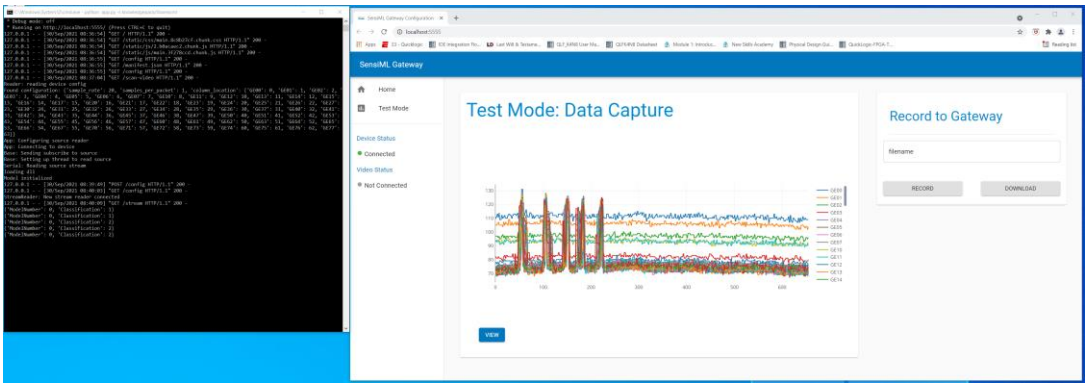
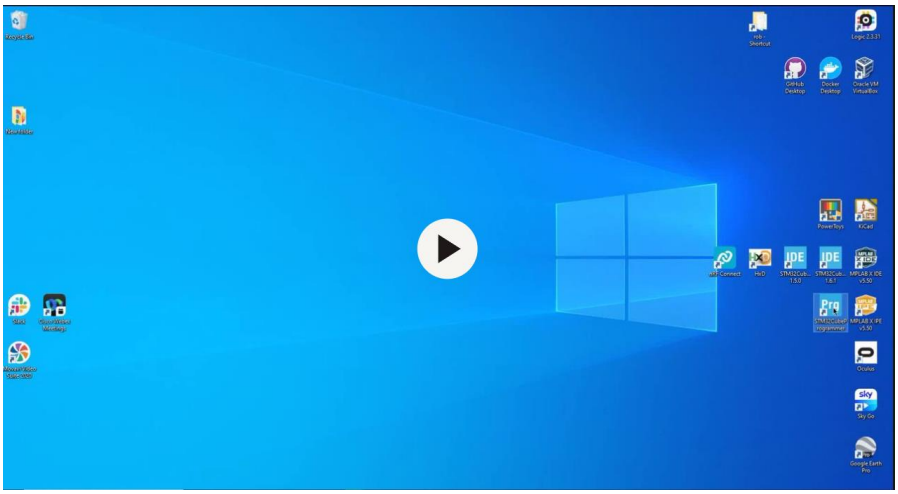


Figure 22. Test Mode Result

See tutorial video 8 below for details:



Video 8. ***testing_Knowledge_pack_in_gateway.mp4***

Deploying Knowledge Pack Firmware in Compagno Board

Downloading the Knowledge Pack

Now that we have validated our model it is time for a live test. To build the firmware for the STWIN Compagno board go to the Download Model tab of the Analytic Studio.
If you are using the community edition of SensiML you can download the firmware library for ARM GCC. For the full source code, see SensiML Plans.

1. Go to the Download Model tab of the Analytic Studio
2. Select the pipeline and model you want to download
3. Select the HW platform **ARM GCC Generic**
4. Click Download and the model will be compiled and downloaded to your computer
5. Unzip the downloaded file and copy the library folder into the Future Electronics firmware folder Core/Src/SensiML/knowledgepack/lib

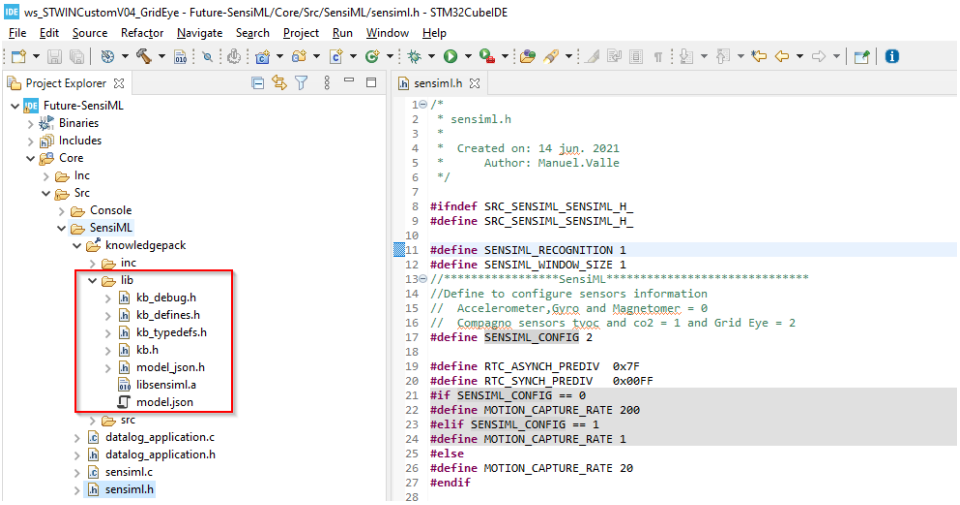


Figure 23. Firmware Folder

6. You need to make sure the define SENSIML_RECOGNITION is one in the file Core/Src/SensiML/sensiml.h

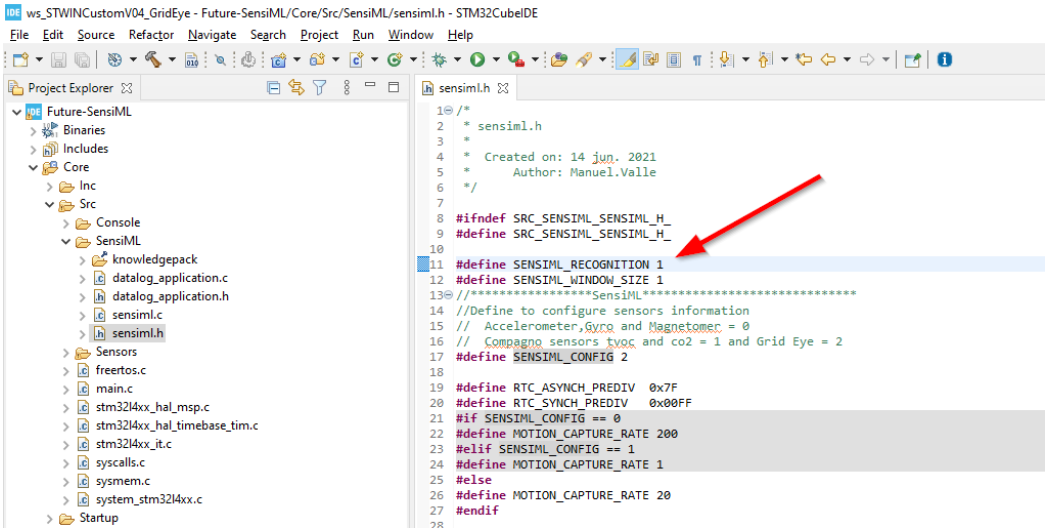
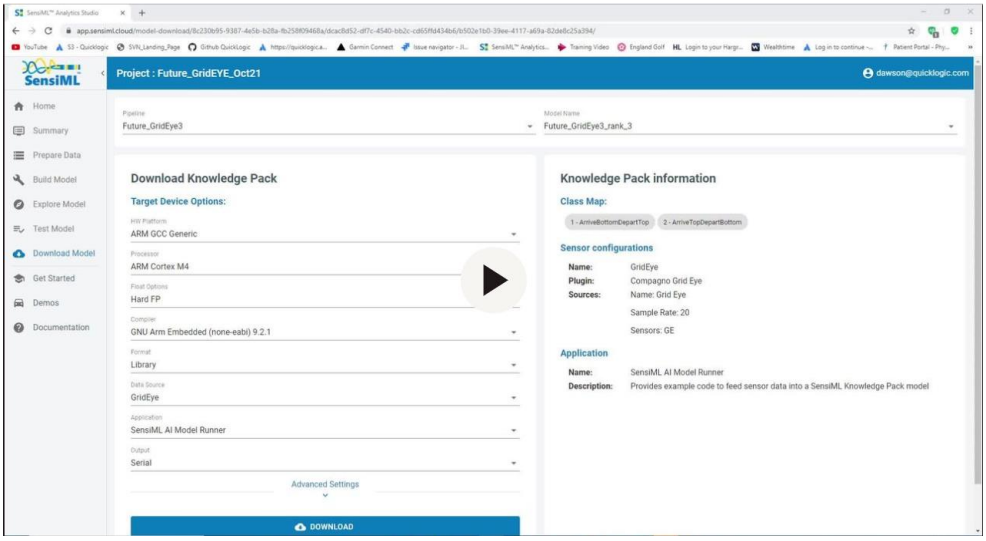


Figure 24. Firmware Folder

7. Reflash and running the new model firmware

See tutorial video 9 below for details:



Video 9. ***DownloadKnowledgePack.mp4***

Running Knowledge Pack (Classifier) on Target GridEYE

1. Integrate the knowledge pack library into your firmware and flash the binary to the STWIN

2. Open a command window in the open-gateway-main directory, at the prompt type:

   >python app.py

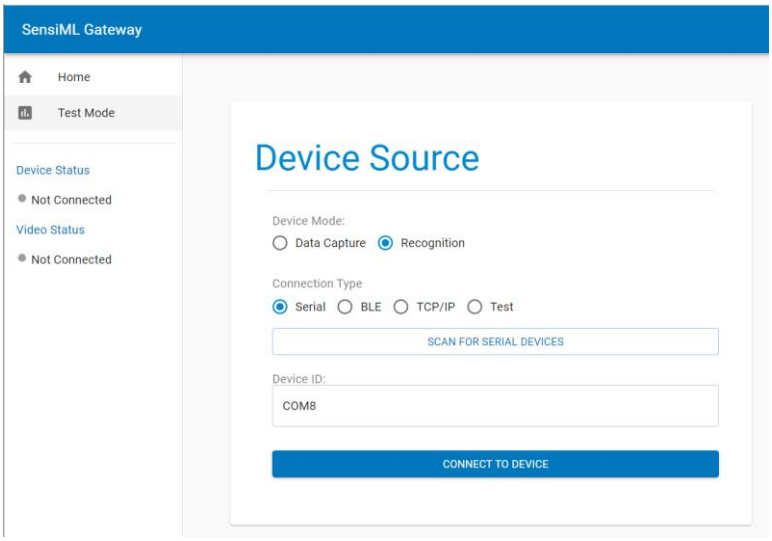


Figure 25. Open Gateway

3. SCAN FOR SERIAL DEVICES, find the Serial COM port of the STWIN board, In the example above it is COM8
4. CONNECT TO DEVICE (If the board doesn't connect, close down the browser and cmd windows, power cycle the target and then reconnect)
5. Select TEST MODE
6. START STREAM

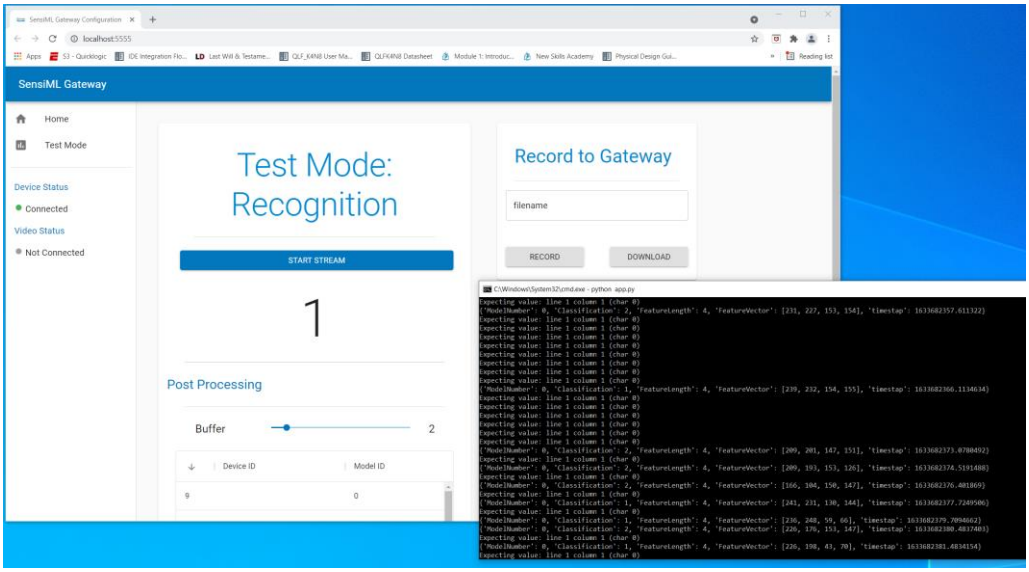
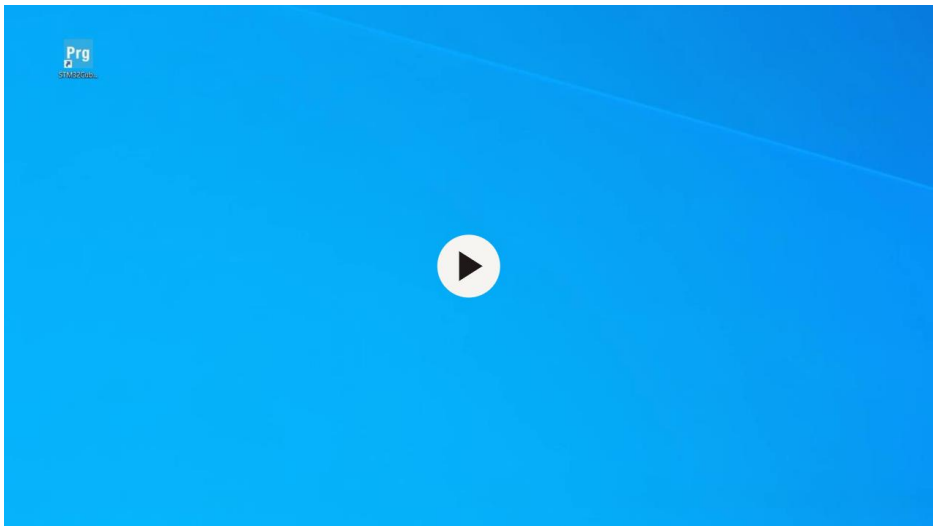


Figure 26. Running Knowledge Pack on Target GridEYE

See tutorial video 10 below for details:



Video 10. ***Recognition_in_gateway.mp4***

Summary

We hope you enjoyed this tutorial using the SensiML Analytics Toolkit. In this tutorial we have covered how to:

1. Collect and annotate a high-quality data set using the STWIN and Future Compagno Board
2. Build a query as input to your model
3. Use SensiML AutoML to search for an edge-optimized model
4. Use the SensiML Analytic Studio to test the model offline

For more information about SensiML, STM and the Compagno board contact our local Future Electronics Sales Office.