

# Crop identification with Edge Impulse

## Lab Workbook

# Objective

“Develop a ML system which can be deployed on Sony Spresense which can identify crops”

# Table of Contents

Updated

[Slide 3](#)

Pre-Lab: Workshop Pre-requisites

[Slide 4](#)

Lab

[Slide 7](#)

Lab

Lab

# About this Workbook

The contents of this workbook are created by Adiuvo Engineering & Training, Ltd.

If you have any questions about the contents, or need assistance, please contact Adam Taylor at [adam@adiuvoengineering.com](mailto:adam@adiuvoengineering.com).

# **Pre-Lab**

## Workshop Pre-requisites

# Required Hardware

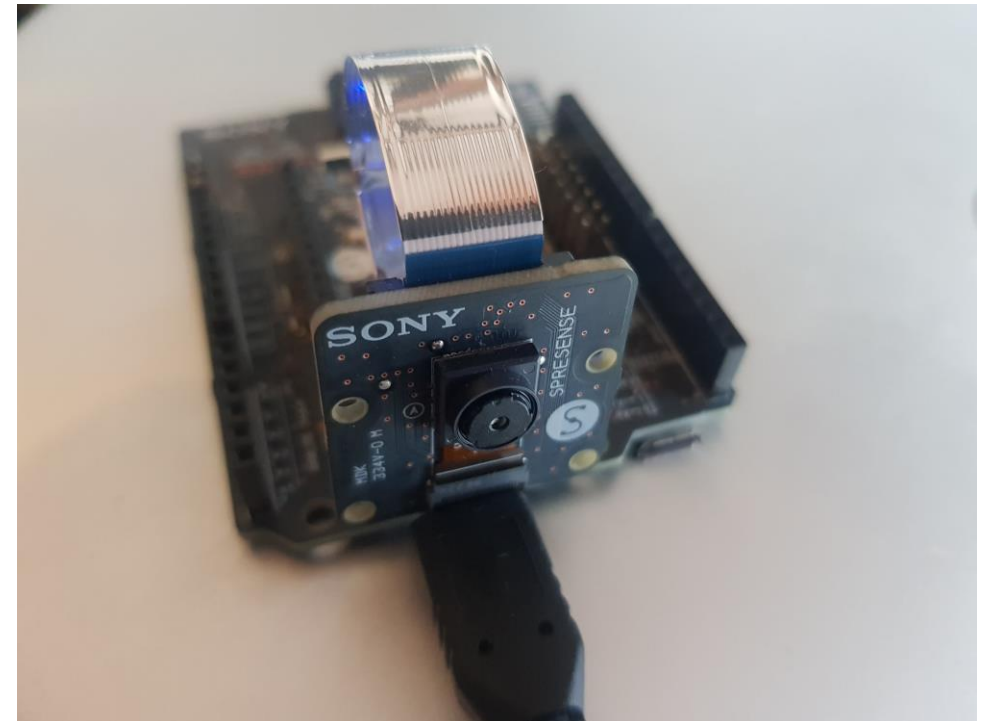
This lab requires a [Sony Spresense](#) development board with the Extension board and Camera module.

Boards can be purchased from the link below.

[Development Board](#)

[Camera Module](#)

[Extension Board](#)



# Downloads and Installations

**Step 1** – Download and install the following at least one day prior to the workshop. This may take a significant amount of time and drive space.

Edge Impulse Account	<a href="#">Register</a>
Edge Impulse Daemon	<a href="#">Install</a>
GNU Make	<a href="#">Download</a>
GNU ARM Embedded Toolchain 8-2018-q4-major	<a href="#">Download</a>
Git	<a href="#">Download</a>
Text Editor e.g Notepad++ / Vs Code etc	

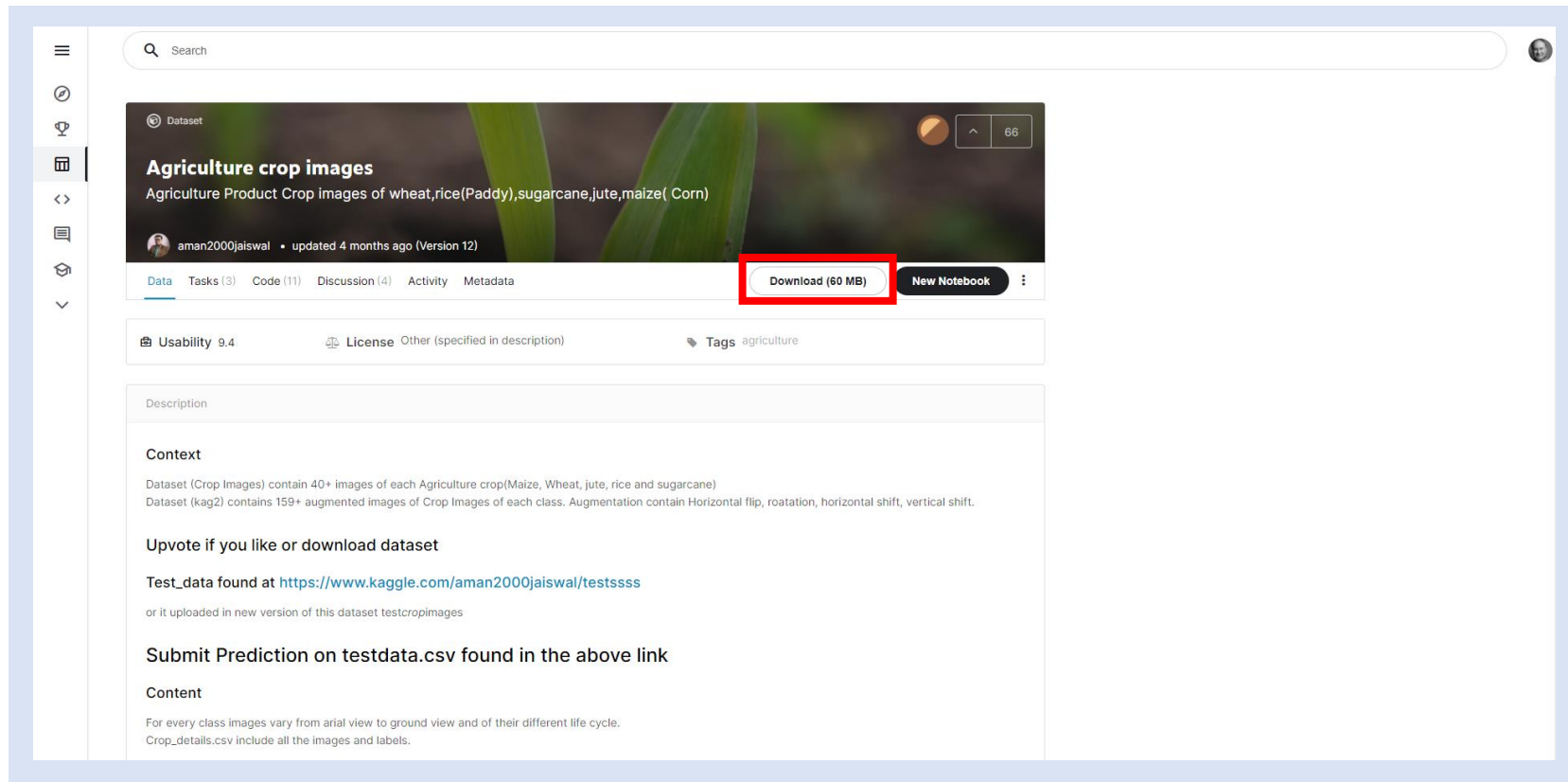
# Lab

## Get the Data Set



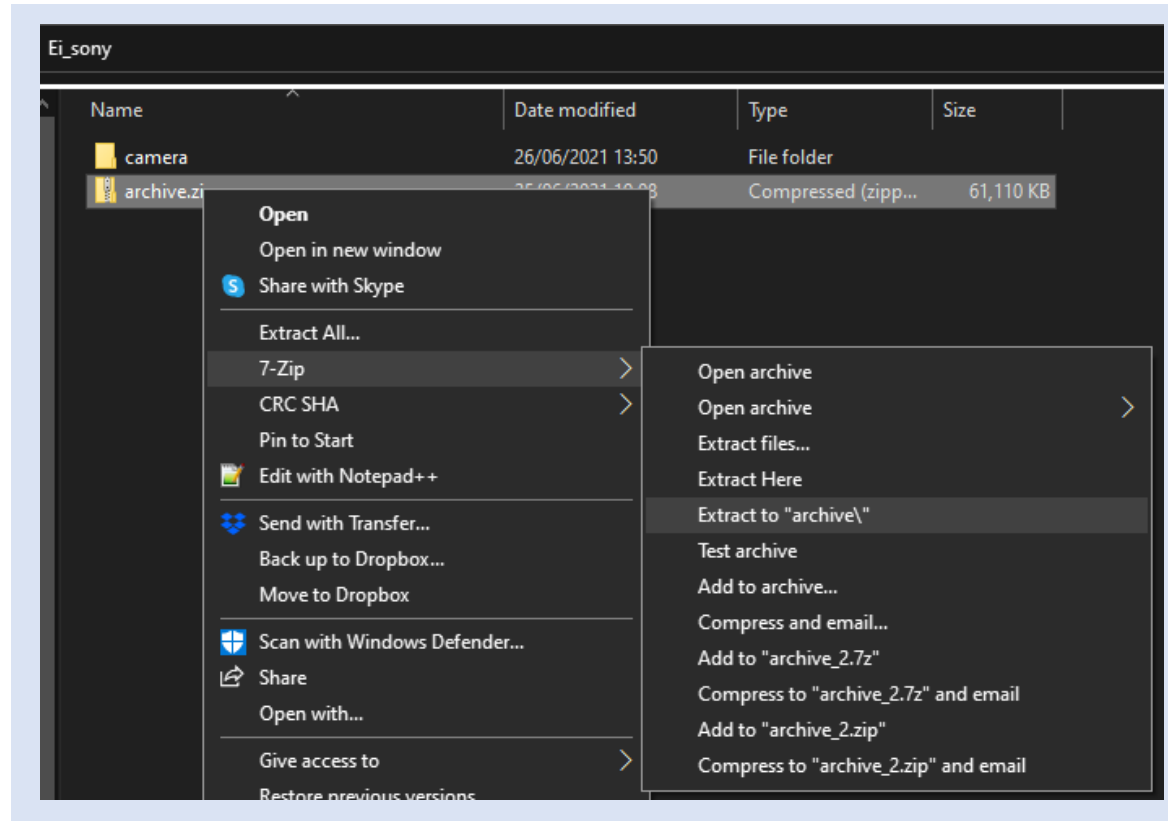
# Edge Impulse: Get the Data Set

**Step 1** – Log into Kaggle and download the data set to a location  
<https://www.kaggle.com/aman2000jaiswal/agriculture-crop-images>



# Edge Impulse: Get the Data Set

## Step 2 – Extract the Compressed Zip file

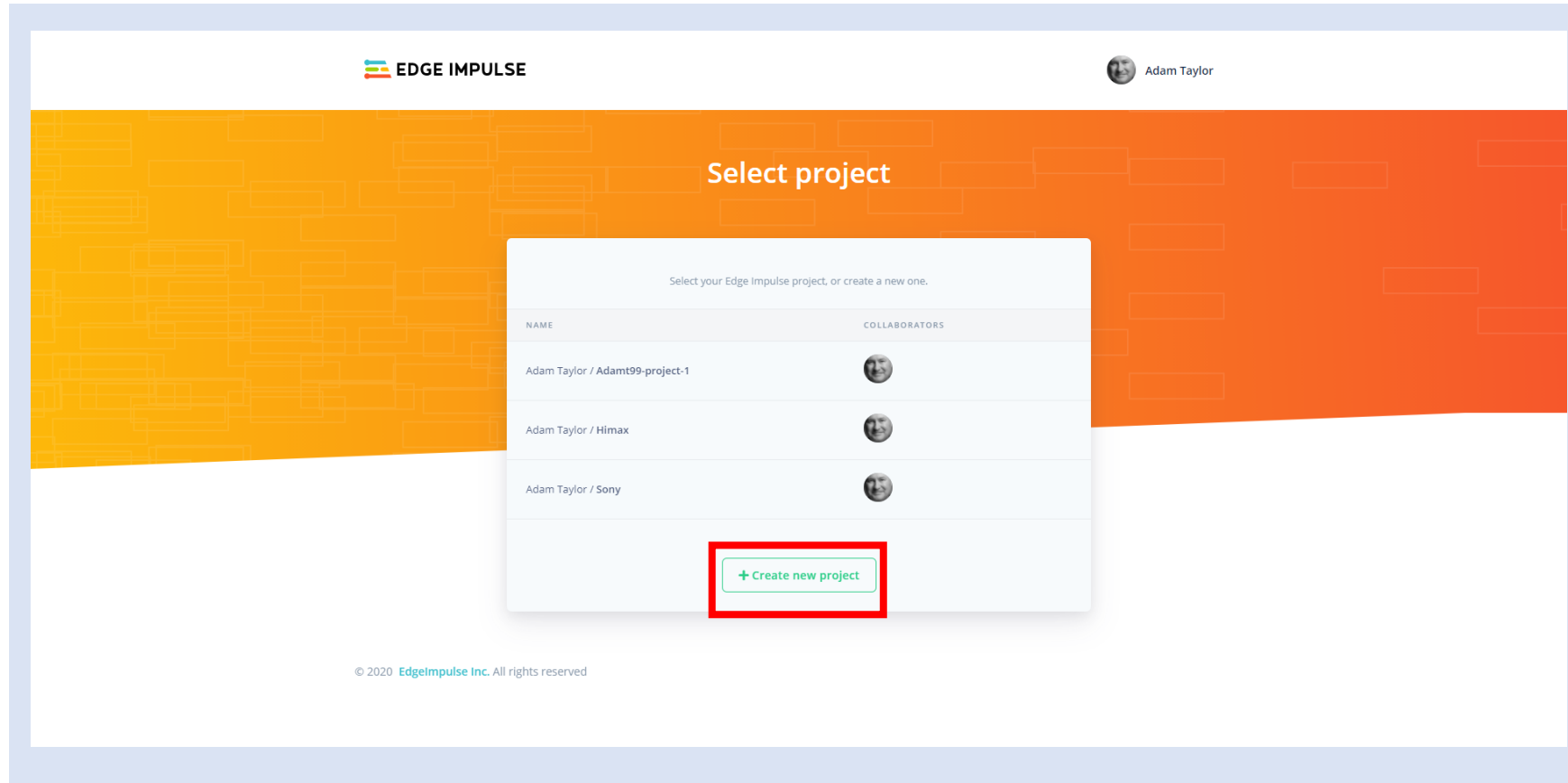


# Lab

# Creating The Impulse

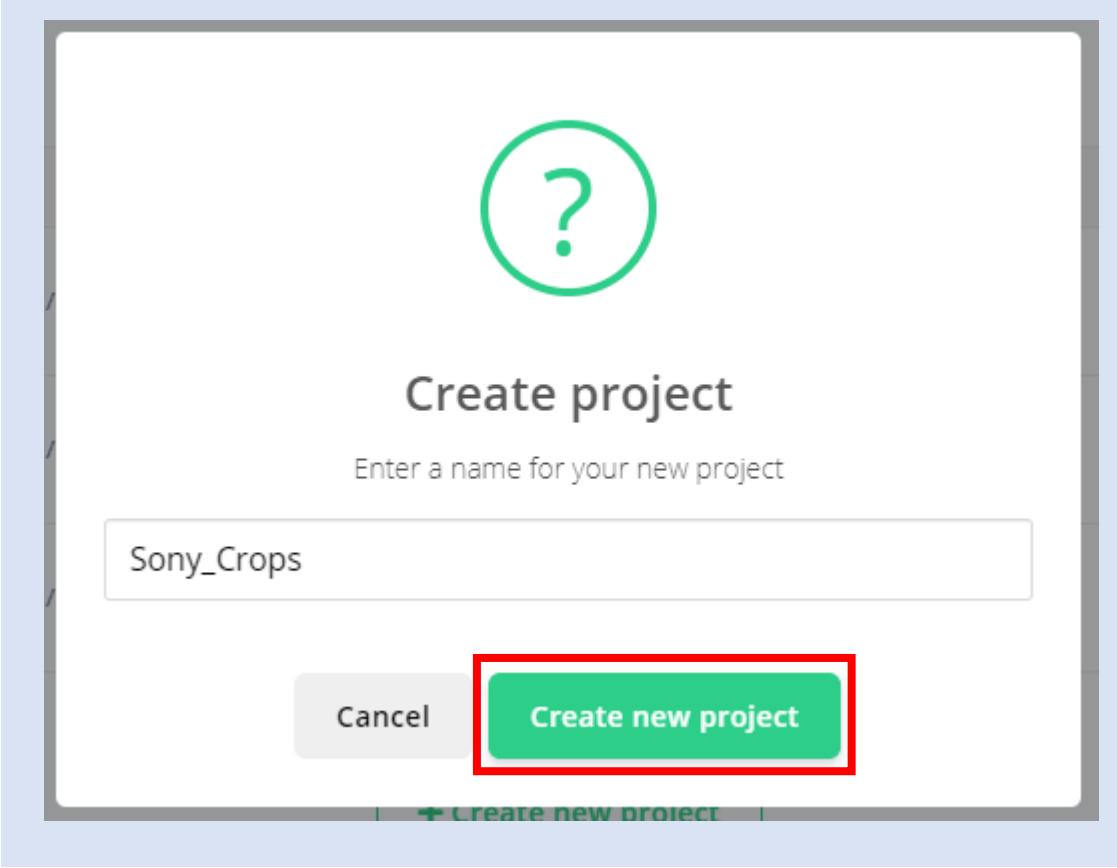
# Edge Impulse: Creating the Impulse

**Step 1** – Log Into Edge Impulse Studio via [www.edgeimpulse.com](https://www.edgeimpulse.com)  
In edge impulse studio select create project



# Edge Impulse: Creating the Impulse

**Step 2** – Name the Project Sony\_Crops – Click on Create New Project



?

**Create project**

Enter a name for your new project

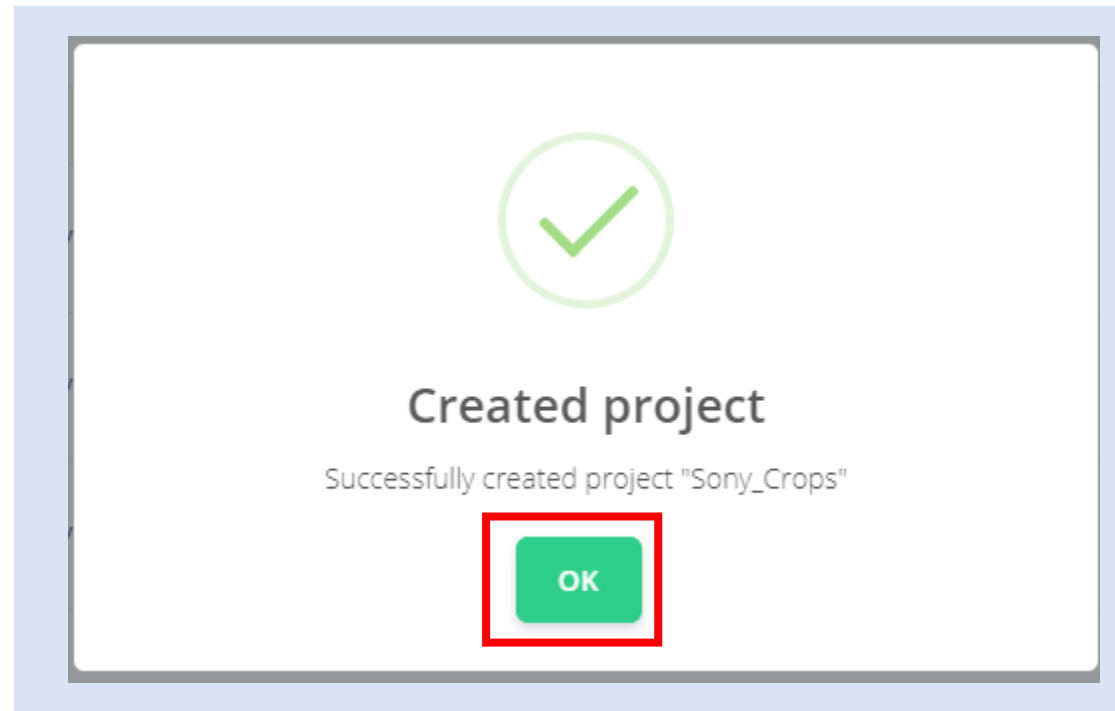
Sony\_Crops

Cancel

**Create new project**

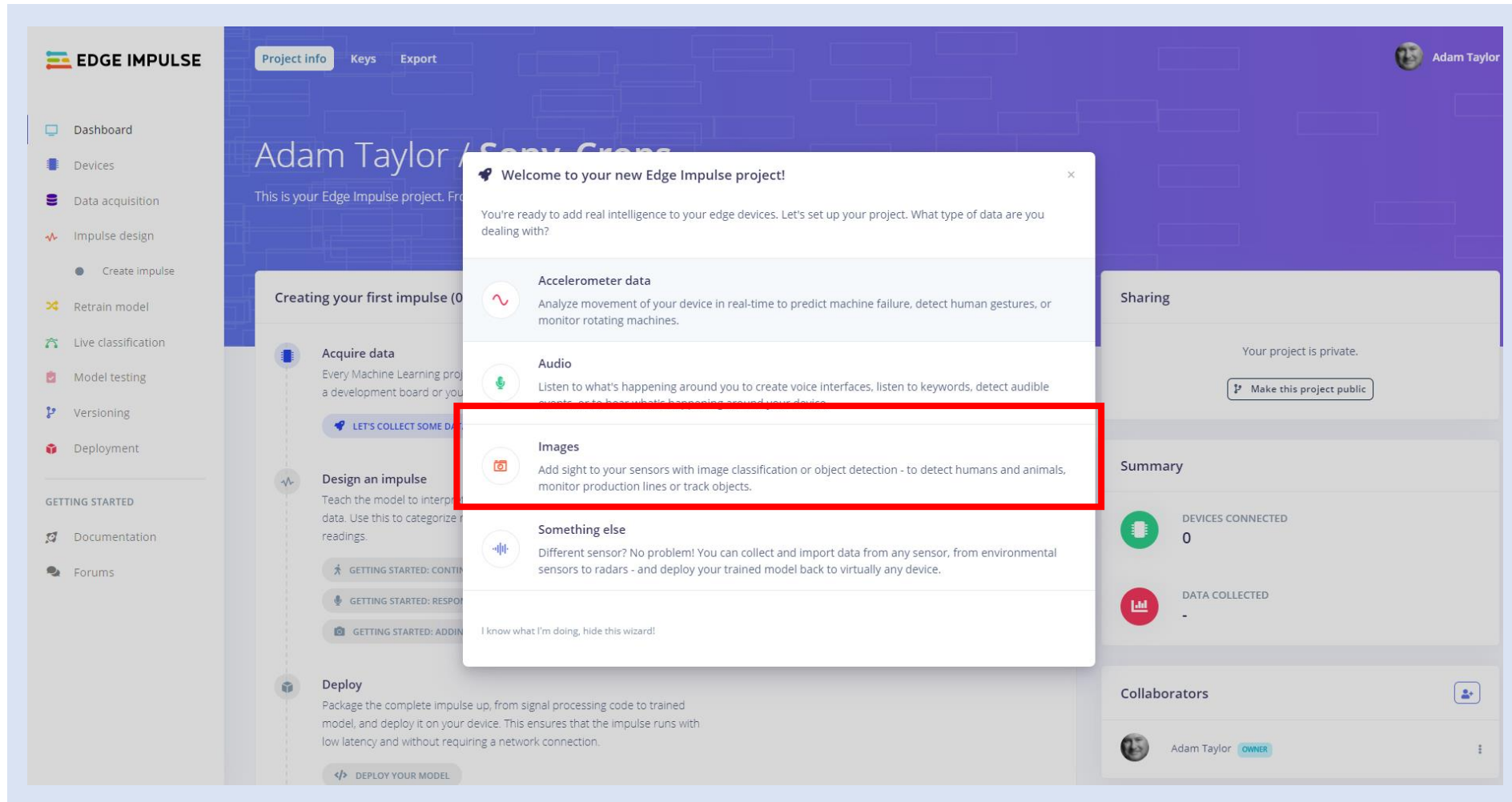
# Edge Impulse: Creating the Impulse

**Step 3 – Click OK**



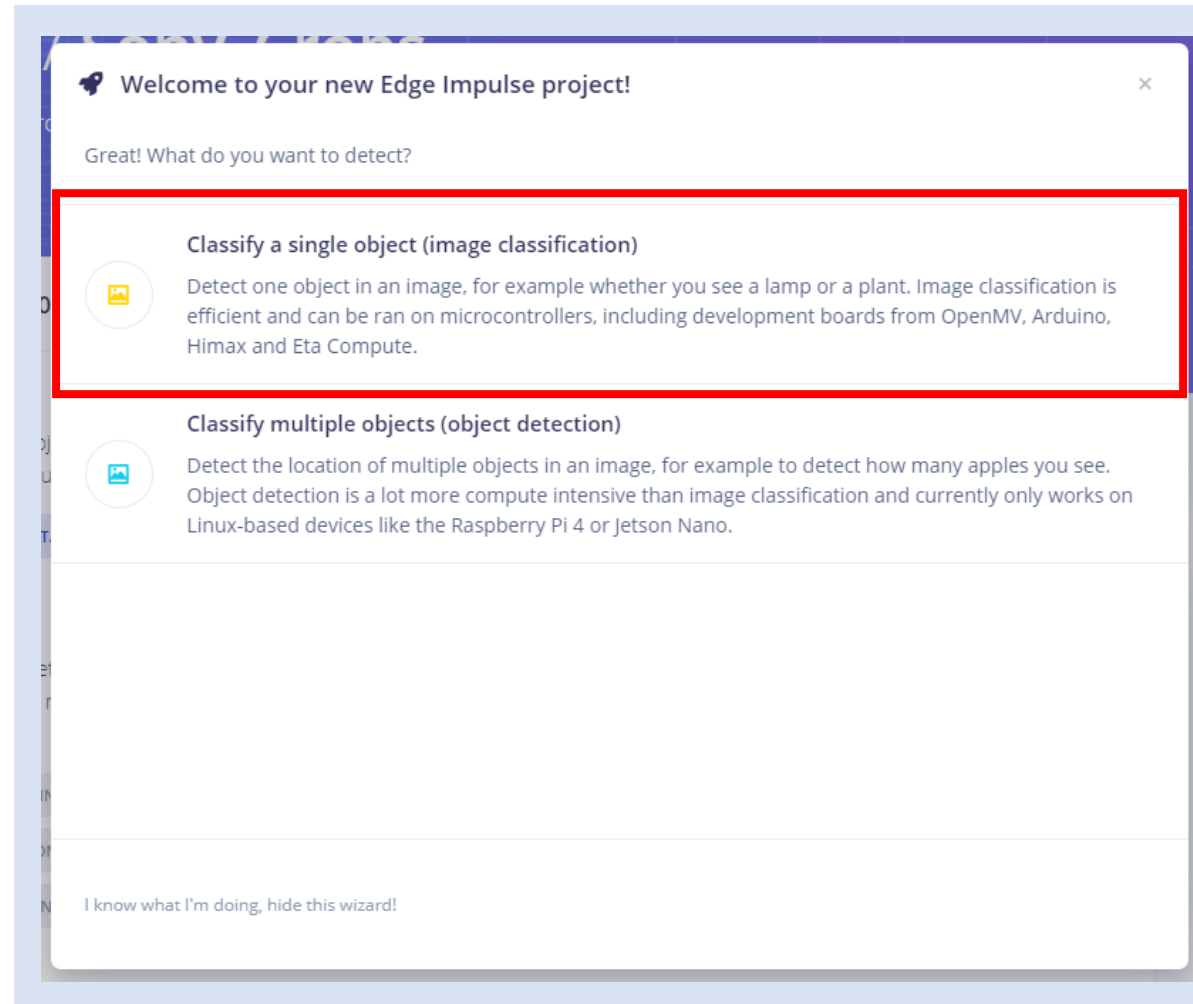
# Edge Impulse: Creating the Impulse

## Step 4 – Select the Images Option in the new project



# Edge Impulse: Creating the Impulse

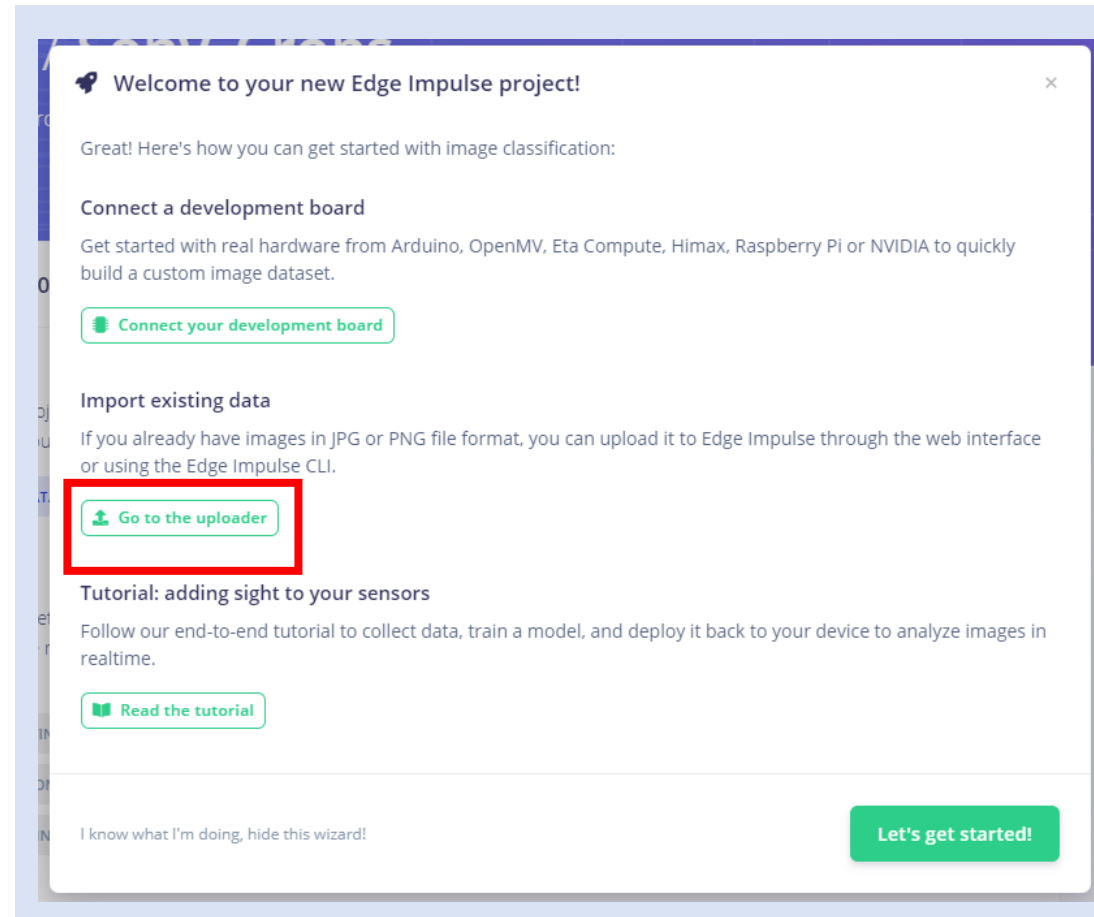
## Step 5 – Select Classify a single object (image classification)





# Edge Impulse: Creating the Impulse

## Step 6 – Select Go to the Uploader



# Edge Impulse: Creating the Impulse

**Step 7** – In the uploader, choose the all the wheat pictures from directory archive\kag2\wheat from the Kaggle data set. Leave the upload category to split between training and testing, enter the label wheat.

The screenshot displays the Edge Impulse web interface for uploading data. On the left is a sidebar with navigation links: Dashboard, Devices, Data acquisition, Impulse design, Create impulse, Retrain model, Live classification, Model testing, Versioning, Deployment, GETTING STARTED, Documentation, and Forums. The main content area is titled 'UPLOAD DATA (SONY\_CROPS)'. It features a 'Choose Files' button (highlighted with a red box) next to '164 files'. Below this, the 'Upload into category' section has three radio button options: 'Automatically split between training and testing' (selected), 'Training', and 'Testing'. The 'Label' section has two radio button options: 'Infer from filename' and 'Enter label:' (selected). The 'Enter label:' option is accompanied by a text input field containing the word 'Wheat' (highlighted with a red box). At the bottom right of the main area is a green 'Begin upload' button.

# Edge Impulse: Creating the Impulse

**Step 8** – Upload the Sugar Cane images from directory archive\kag2\sugarcane from the Kaggle data set. Leave the upload category to split between training and testing, enter the label wheat.

Upload existing data

You can upload existing data to your project in the [Data Acquisition Format](#) (CBOR, JSON, CSV), or as WAV, JPG or PNG files.

Select files

Choose Files 160 files

Upload into category

☒ Automatically split between training and testing ?

☐ Training

☐ Testing

Label

☐ Infer from filename ?

☒ Enter label:

Sugar\_Cane

Begin upload

# Edge Impulse: Creating the Impulse

**Step 9** – Upload the Sugar Cane images from directory archive\kag2\maize from the Kaggle data set. Leave the upload category to split between training and testing, enter the label wheat.

Upload existing data

You can upload existing data to your project in the [Data Acquisition Format](#) (CBOR, JSON, CSV), or as WAV, JPG or PNG files.

Select files

[Choose Files](#) 160 files

Upload into category

☒ Automatically split between training and testing ?

☐ Training

☐ Testing

Label

☐ Infer from filename ?

☒ Enter label:

Maize

[Begin upload](#)

# Edge Impulse: Creating the Impulse

**Step 10** – Click on Data acquisition and note the number of data elements, you should have three labels and there should be images under both training and test data

**EDGE IMPULSE**

DATA ACQUISITION (SONY\_CROPS)

Training data Test data

Did you know? You can capture data from any device or development board, or upload your existing datasets - [Show options](#)

DATA COLLECTED  
379 items

LABELS  
3

Record new data [Connect using WebUSB](#)

No devices connected to the remote management API.

RAW DATA  
Click on a sample to load...

SAMPLE NAME	LABEL	ADDED	LENGTH
maize040ahf.jpeg.2919a4g5	maize	Today, 18:07:57	-
maize040a.jpeg.2919a4fc	maize	Today, 18:07:57	-
maize039ahf.jpeg.2919a4c5	maize	Today, 18:07:57	-
maize038arot.jpeg.2919a49k	maize	Today, 18:07:56	-
maize038ahs.jpeg.2919a48c	maize	Today, 18:07:56	-
maize038ahf.jpeg.2919a472	maize	Today, 18:07:56	-
maize038a.jpeg.2919a45o	maize	Today, 18:07:56	-
maize037arot.jpeg.2919a44e	maize	Today, 18:07:56	-

# Edge Impulse: Creating the Impulse

**Step 11** – Click on Impulse Design – Notice the Image data processing is already added into the

The screenshot shows the Edge Impulse web interface for creating a new impulse. On the left sidebar, the 'Impulse design' option is highlighted with a red rectangle. The main area is titled 'CREATE IMPULSE (SONY\_CROPS)' and features a user profile for 'Adam Taylor'. Below the title bar, there's a description: 'An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.' The workspace contains several blocks: a red 'Image data' block with input fields for 'Image width' and 'Image height' (both set to 96), a dropdown for 'Fit to shortest axis', and a note about optimal image sizes for transfer learning. There are also two dashed boxes for 'Add a processing block' and 'Add a learning block', and a green 'Output features' block with a checkmark. A 'Save impulse' button is located at the bottom right.

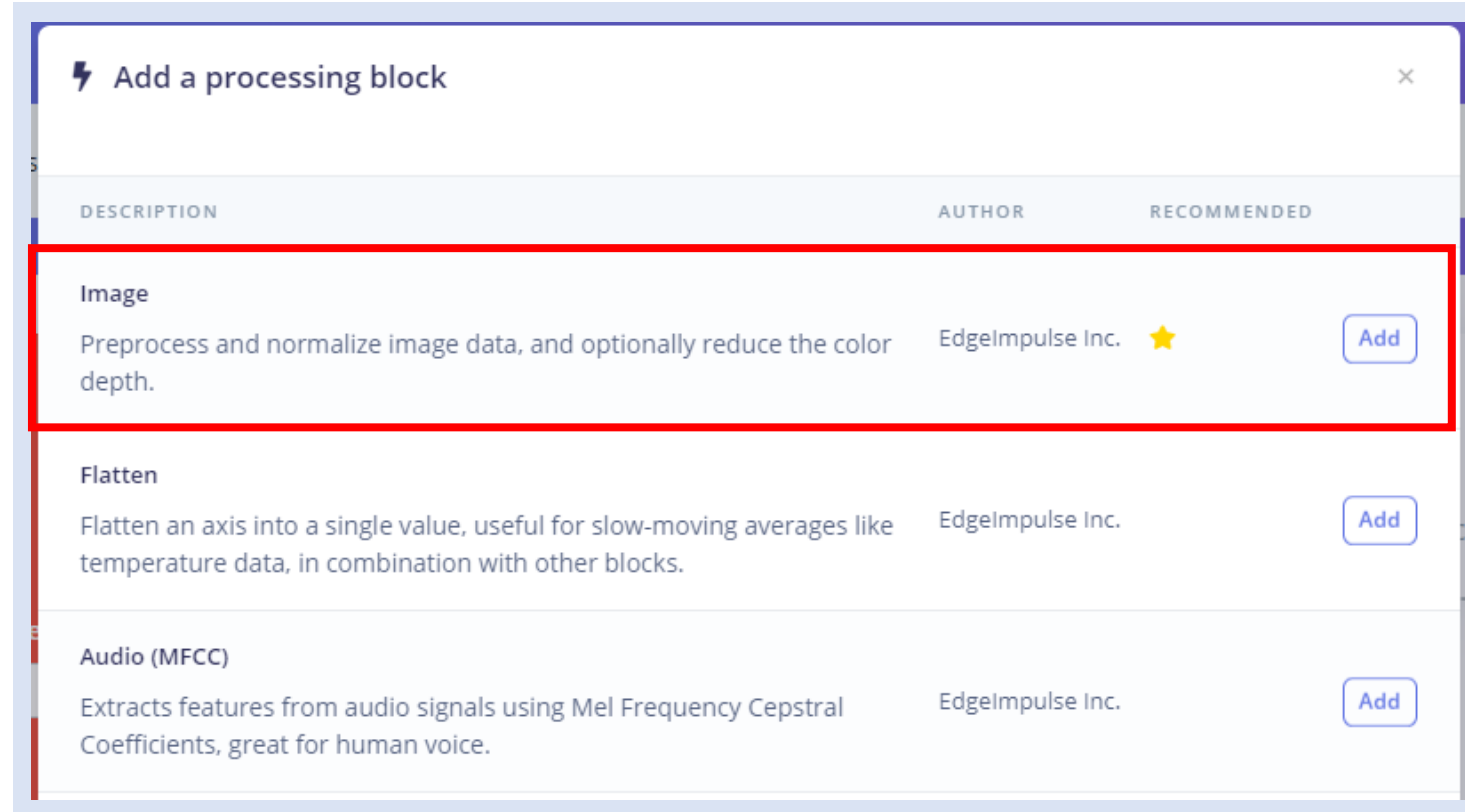
# Edge Impulse: Creating the Impulse

## Step 12 – Click on Add a processing block

The screenshot shows the Edge Impulse web interface for creating a new impulse. The left sidebar contains navigation links: Dashboard, Devices, Data acquisition, Impulse design, Create impulse (selected), Retrain model, Live classification, Model testing, Versioning, and Deployment. Below these are 'GETTING STARTED' links: Documentation and Forums. The main area is titled 'CREATE IMPULSE (SONY\_CROPS)' and includes a user profile for Adam Taylor. A descriptive text box states: 'An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.' The workflow consists of three blocks: 'Image data' (red), 'Add a processing block' (light blue, highlighted with a red rectangle), 'Add a learning block' (light blue), and 'Output features' (teal). The 'Image data' block shows 'Axes' set to 'image', 'Image width' and 'Image height' both set to 96, and a 'Shortest axis' dropdown. A note at the bottom of the 'Image data' block states: 'For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.' A 'Save Impulse' button is located at the bottom right of the workflow area.

# Edge Impulse: Creating the Impulse

## Step 13 – Select Image as the processing block





# Edge Impulse: Creating the Impulse

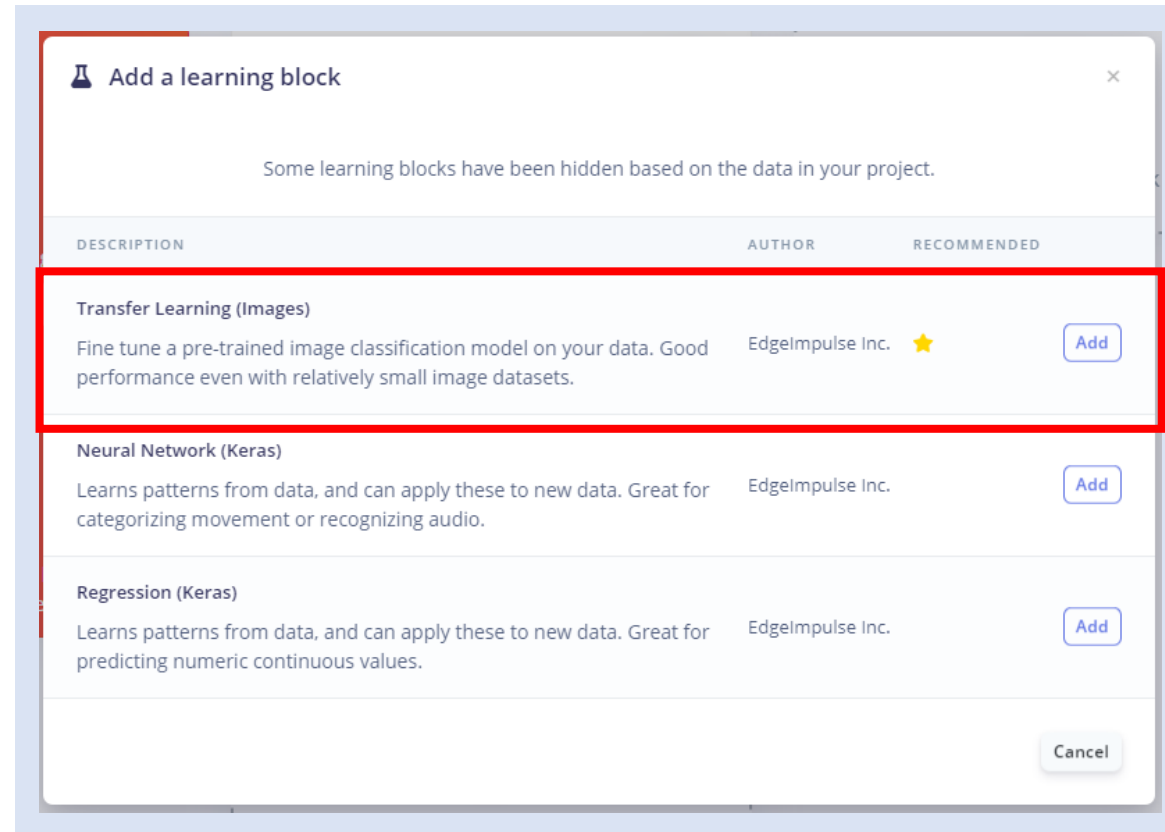
## Step 14 – Click on Add a learning block

The screenshot shows the Edge Impulse web interface for creating a new impulse named 'SONY\_CROPS'. The interface is divided into several sections:

- Left Sidebar:** Contains navigation links for Dashboard, Devices, Data acquisition, Impulse design, Create impulse (selected), Retrain model, Live classification, Model testing, Versioning, and Deployment. Below these are 'GETTING STARTED' and 'Documentation' links.
- Header:** Displays 'EDGE IMPULSE' and the user's name 'Adam Taylor'.
- Impulse Definition:** A text box explains: 'An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.'
- Main Workspace:**
  - Image data (Red Panel):** Shows 'Axes' set to 'image'. 'Image width' and 'Image height' are both set to '96'. A dropdown menu is set to 'Shortest axis' with a visual representation of a crop. A note at the bottom states: 'For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.'
  - Image (White Panel):** Shows 'Name' as 'Image' and 'Input axes' with a checked checkbox for 'image'.
  - Add a learning block (Red Box):** A dashed box containing a flask icon and the text 'Add a learning block', which is the target of the instruction.
  - Output features (Teal Panel):** Shows a checkmark icon, indicating that output features are defined.
- Bottom Right:** A green 'Save Impulse' button.

# Edge Impulse: Creating the Impulse

## Step 15 – Select Transfer Learning (Images)



# Edge Impulse: Creating the Impulse

## Step 16 – Click on Save Impulse

The screenshot displays the Edge Impulse configuration interface with four main panels:

- Image data** (Red panel):
  - Icon: Database symbol
  - Axes: image
  - Image width: 96
  - Image height: 96
  - Shortest axis: dropdown menu with a diagram showing two images and a red arrow pointing from the shorter one to the longer one.
  - Footer: *For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.*
- Image** (White panel):
  - Icon: Lightning bolt
  - Name: Image
  - Input axes: ☒ image
- Transfer Learning (Images)** (Purple panel):
  - Icon: Flask
  - Name: Transfer learning
  - Input features: ☒ Image
  - Output features: 3 (Sugar\_Cane, Wheat, maize)
- Output features** (Teal panel):
  - Icon: Checkmark
  - Output features: 3 (Sugar\_Cane, Wheat, maize)
  - Save Impulse** button (highlighted with a red rectangle)

# Edge Impulse: Creating the Impulse

## Step 17 – Click on Save Impulse

**EDGE IMPULSE**

CREATE IMPULSE (SONY\_CROPS) Adam Taylor

Successfully stored impulse. Configure the signal processing and learning blocks in the navigation bar.

**Image data**

Axes  
image

Image width: 96 Image height: 96

Shortest axis

For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.

**Image**

Name: Image

Input axes: ☒ image

**Transfer Learning (Images)**

Name: Transfer learning

Input features: ☒ Image

Output features: 3 (Sugar\_Cane, Wheat, maize)

**Output features**

3 (Sugar\_Cane, Wheat, maize)

Save Impulse

Navigation bar (left sidebar):

- Dashboard
- Devices
- Data acquisition
- Impulse design
- Create impulse
  - Image**
  - Transfer learning
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

# Edge Impulse: Creating the Impulse

## Step 18 – Select Dashboard and then change the latency calculation to Sony Spresense

The screenshot shows the Edge Impulse web interface. The left sidebar contains a navigation menu with the following items: Dashboard (highlighted with a red box), Devices, Data acquisition, Impulse design (with sub-items: Create impulse, Image, Transfer learning), Retrain model, Live classification, Model testing, Versioning, and Deployment. Below this is a 'GETTING STARTED' section with links to Documentation and Forums.

The main content area is titled 'Creating your first impulse (67% complete)'. It features a vertical progress bar with three stages: 'Acquire data' (completed), 'Design an impulse' (in progress), and 'Deploy' (not started). The 'Design an impulse' stage has three sub-tasks: 'GETTING STARTED: CONTINUOUS MOTION RECOGNITION', 'GETTING STARTED: RESPONDING TO YOUR VOICE', and 'GETTING STARTED: ADDING SIGHT TO YOUR SENSORS'. A 'DEPLOY YOUR MODEL' button is at the bottom of this section.

On the right side, there is a 'Sharing' section with a 'Make this project public' button. Below it is a 'Summary' section showing 'DEVICES CONNECTED: 0' and 'DATA COLLECTED: 474 items'. The 'Collaborators' section lists 'Adam Taylor' as the 'OWNER'. The 'Project info' section shows 'Project ID: 37550', 'Labeling method: One label per data item', and 'Latency calculations: Sony Spresense' (highlighted with a red box).

At the bottom, there is a 'Download block output' table:

TITLE	TYPE	SIZE
Image training data	NPY file	379 windows
Image training labels	NPY file	379 windows

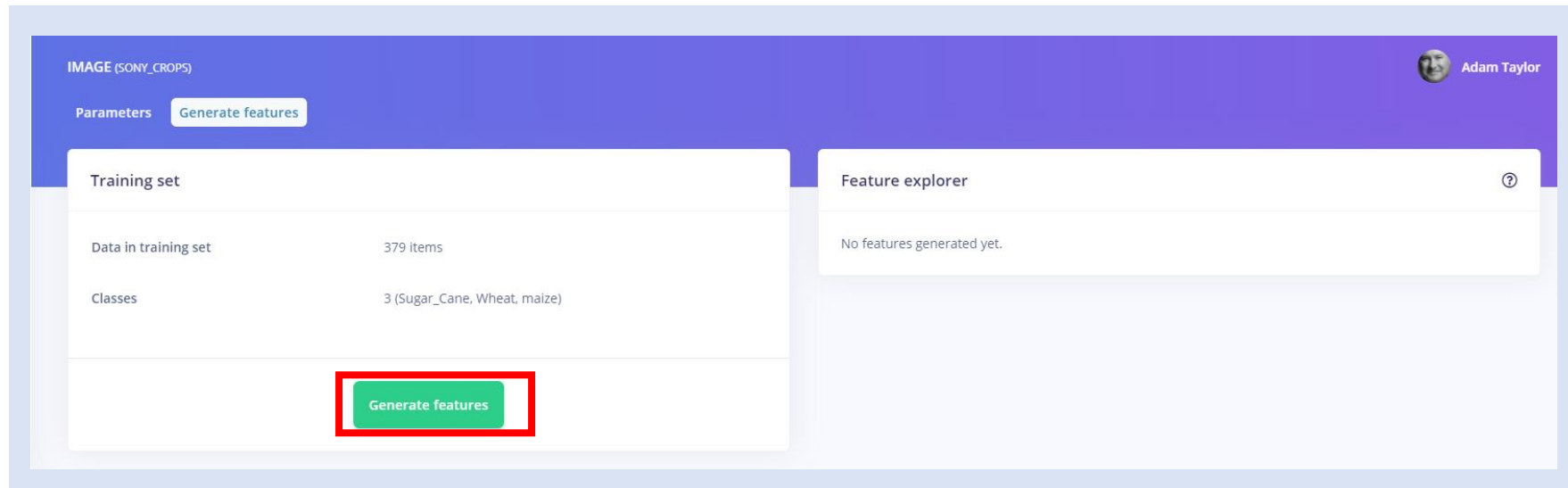
# Edge Impulse: Creating the Impulse

**Step 19** – Click save parameters and then select Generate features

The screenshot displays the Edge Impulse web interface. At the top, there's a purple header with the text 'IMAGE (SONY\_CROPS)' and a user profile 'Adam Taylor'. Below the header, there are two tabs: 'Parameters' and 'Generate features', with the latter highlighted by a red rectangle. The main content area is divided into several sections. On the left, there's a 'Raw data' section showing a cornfield image. Below it, the 'Raw features' section displays a long string of hexadecimal values. The 'Parameters' section on the left includes a 'Color depth' dropdown menu set to 'RGB' and a 'Save parameters' button, which is also highlighted by a red rectangle. On the right, the 'DSP result' section shows the same cornfield image. Below it, the 'Processed features' section displays a long string of numerical values. At the bottom right, the 'On-device performance' section shows 'PROCESSING TIME 4 ms.' and 'PEAK RAM USAGE 4 KB'.

# Edge Impulse: Creating the Impulse

## Step 20 – Click Generate features



# Edge Impulse: Creating the Impulse

## Step 21 – Click on Transfer learning and then Start training

The screenshot displays the Edge Impulse web interface for the 'TRANSFER LEARNING (SONY\_CROPS)' project. The left sidebar contains navigation options: Dashboard, Devices, Data acquisition, Impulse design, Create impulse, Image, **Transfer learning** (highlighted with a red box), Retrain model, Live classification, Model testing, Versioning, and Deployment. The main content area is titled 'TRANSFER LEARNING (SONY\_CROPS)' and includes a version selector '#1' and a link to 'Click to set a description for this version'. The 'Neural Network settings' section contains 'Training settings' with input fields for 'Number of training cycles' (20), 'Learning rate' (0.0005), 'Data augmentation' (unchecked), and 'Minimum confidence rating' (0.60). Below this is the 'Neural network architecture' section, showing an 'Input layer (27,648 features)' and an 'Output layer (3 features)'. A central box displays 'MobileNetV2 0.35 (final layer: 16 neurons, 0.1 dropout)' with a 'Choose a different model' link. At the bottom, the 'Start training' button is highlighted with a red box. The right sidebar shows the 'Training output' section.



# Edge Impulse: Creating the Impulse

**Step 22** – Note the 72% accuracy – This is not acceptable so now we will need to optimize the solution

TRANSFER LEARNING (SONY\_CROPS) #1 [Click to set a description for this version](#)

Adam Taylor

### Neural Network settings

#### Training settings

Number of training cycles ②


Learning rate ②

Data augmentation ② ☐

Minimum confidence rating ②

#### Neural network architecture

Input layer (27,648 features)



MobileNetV2 0.35 (final layer: 16 neurons, 0.1 dropout)

[Choose a different model](#)

Output layer (3 features)

[Start training](#)

### Training output

```

Converting TensorFlow Lite float32 model...
Converting TensorFlow Lite int8 quantized model with float32 input and output...
Converting TensorFlow Lite int8 quantized model with int8 input and output...
Calculating performance metrics...
Profiling float32 model...
Profiling float32 model (tfLite)...
Profiling float32 model (EON)...
Profiling int8 model...
Profiling 78% done
Profiling int8 model (tfLite)...
Profiling int8 model (EON)...

Model training complete


Job completed

```


### Model

Model version: ② [Quantized \(int8\)](#)

Last training performance (validation set)



ACCURACY  
**72.4%**



LOSS  
**0.67**

Confusion matrix (validation set)

	SUGAR_CANE	WHEAT	MAIZE
SUGAR_CANE	76.2%	0%	23.8%
WHEAT	10%	66.7%	23.3%
MAIZE	20%	4%	76%
F1 SCORE	0.71	0.78	0.68

[Feature explorer \(full training set\)](#)

# Edge Impulse: Creating the Impulse

**Step 23** – Click on Create impulse and then change the image data from fit shortest axis to squash, save the impulse

The screenshot displays the Edge Impulse web interface for creating a new impulse. The left sidebar contains a navigation menu with the following items: Dashboard, Devices, Data acquisition, Impulse design, **Create impulse** (highlighted with a red box), Image, Transfer learning, Retrain model, Live classification, Model testing, Versioning, and Deployment. The main workspace is titled 'CREATE IMPULSE (SONY\_CROPS)' and includes a user profile for Adam Taylor. A descriptive text box states: 'An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.' The workspace is divided into four colored panels: 1. **Image data** (red): Shows 'Axes' set to 'image', 'Image width' and 'Image height' both set to 96, and a dropdown menu for 'Image data' currently set to 'fit shortest axis' (highlighted with a blue box). A note at the bottom states: 'For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.' 2. **Image** (white): Shows 'Name' as 'Image' and 'Input axes' with a checked checkbox for 'Image'. 3. **Transfer Learning (Images)** (purple): Shows 'Name' as 'Transfer learning', 'Input features' with a checked checkbox for 'Image', and 'Output features' as '3 (Sugar\_Cane, Wheat, maize)'. 4. **Output features** (teal): Shows '3 (Sugar\_Cane, Wheat, maize)' and a green 'Save Impulse' button (highlighted with a red box).

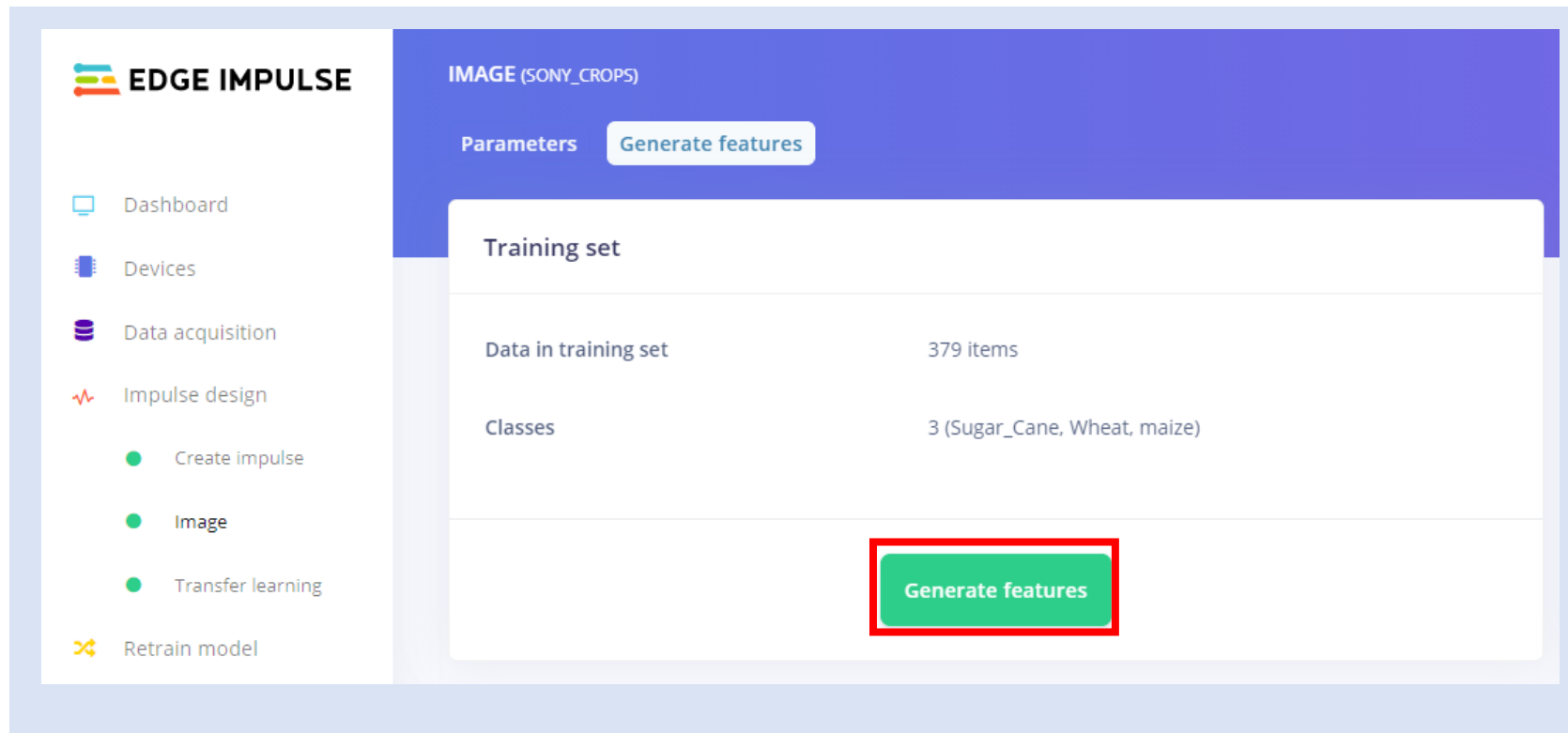
# Edge Impulse: Creating the Impulse

**Step 24** – Click on Image and change the color depth from RGB to GreyScale, save parameters

The screenshot displays the Edge Impulse web interface. On the left is a sidebar with navigation links: Dashboard, Devices, Data acquisition, Impulse design, Create impulse, Image (highlighted with a red box), Transfer learning, Retrain model, Live classification, Model testing, Versioning, Deployment, GETTING STARTED, Documentation, and Forums. The main area is titled 'IMAGE (SONY\_CROPS)' and includes tabs for 'Parameters' and 'Generate features'. The 'Raw data' section shows a sample image of corn plants. Below this, the 'Raw features' section displays a hexagram of feature values. The 'Parameters' section shows the 'Image' impulse with the 'Color depth' dropdown menu set to 'Grayscale' (highlighted with a red box). A 'Save parameters' button is located below the dropdown. The 'DSP result' section shows the processed image in grayscale. The 'Processed features' section displays a hexagram of feature values. The 'On-device performance' section shows 'PROCESSING TIME' as 5 ms and 'PEAK RAM USAGE' as 4 KB.

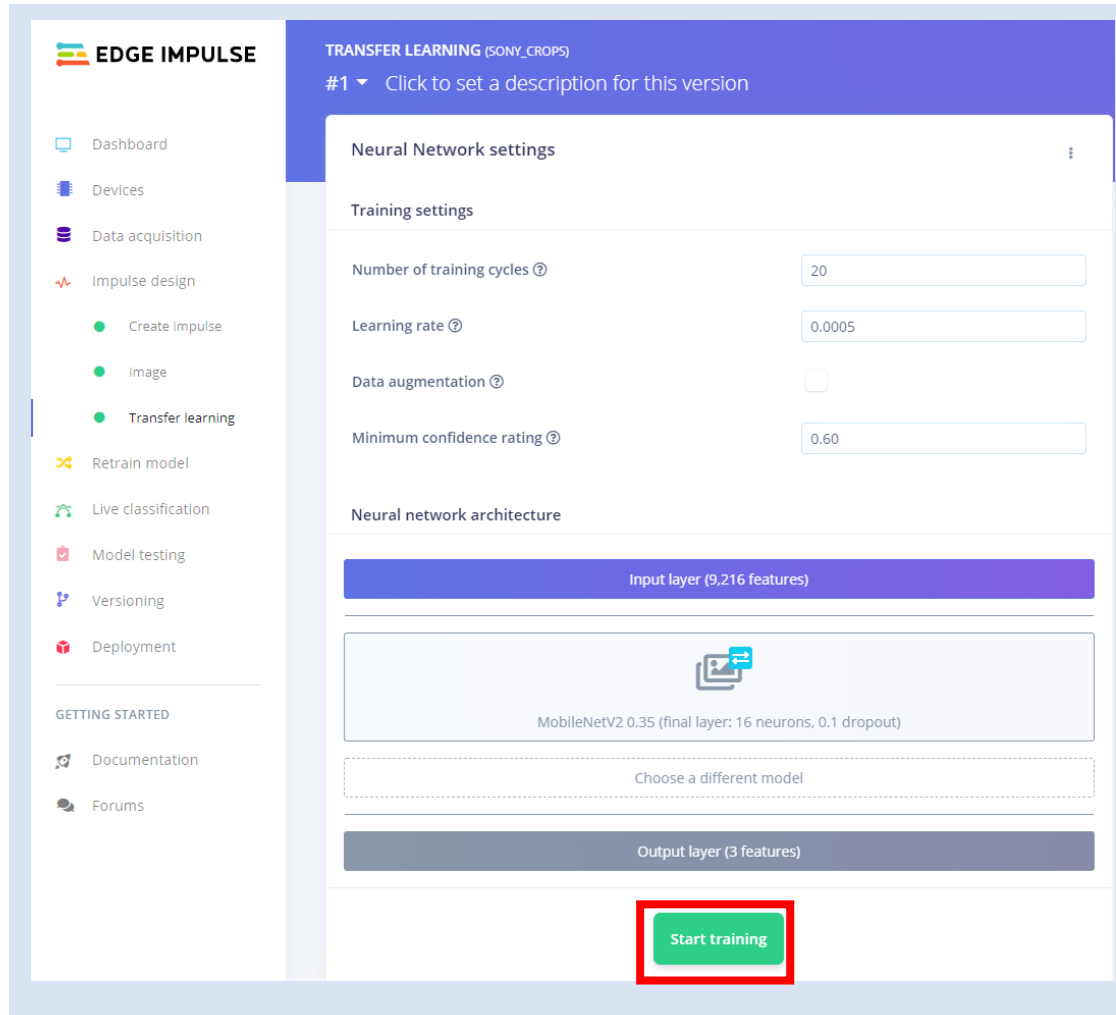
# Edge Impulse: Creating the Impulse

Step 25 – Click on Generate features



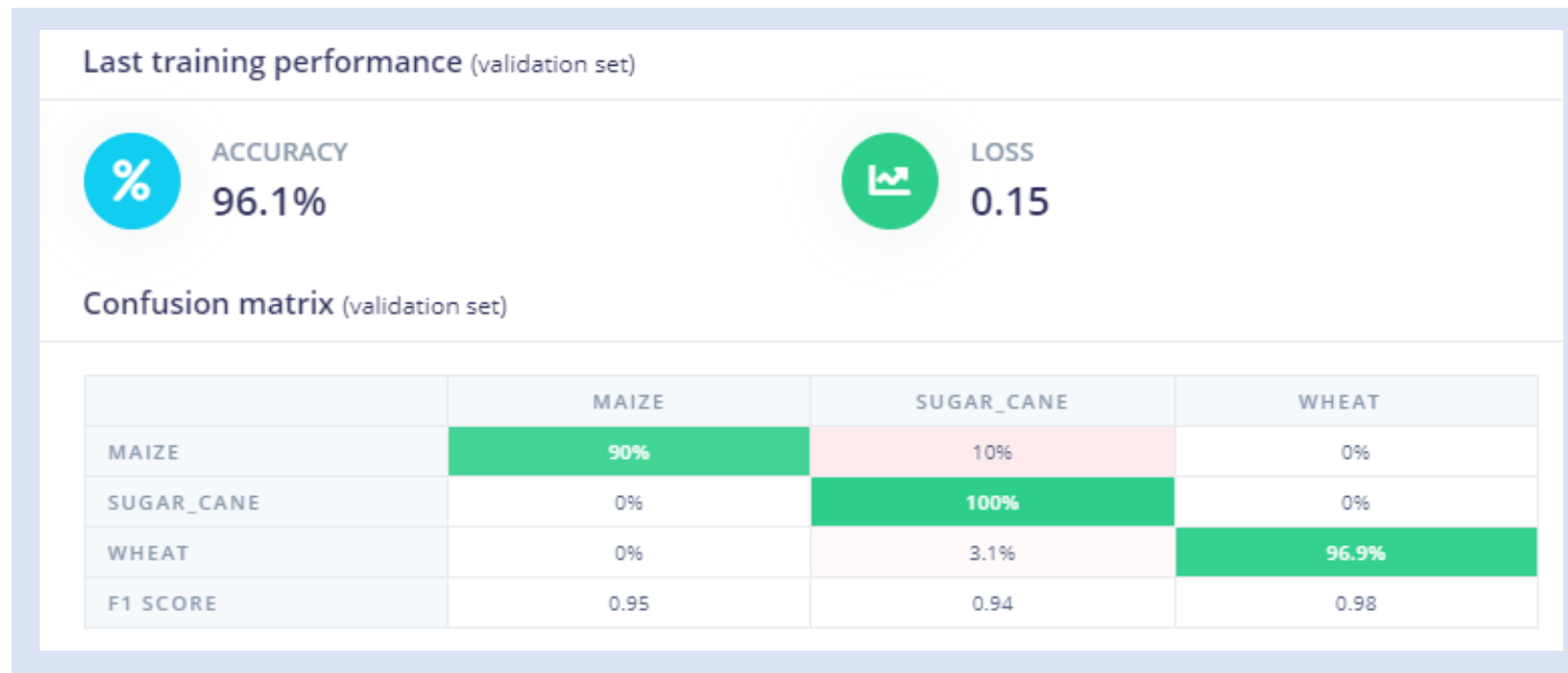
# Edge Impulse: Creating the Impulse

**Step 26** – Rerun the training of the Transfer Learning, click on Start training



# Edge Impulse: Creating the Impulse

**Step 27** – Rerun the training of the Transfer Learning, click on Start training



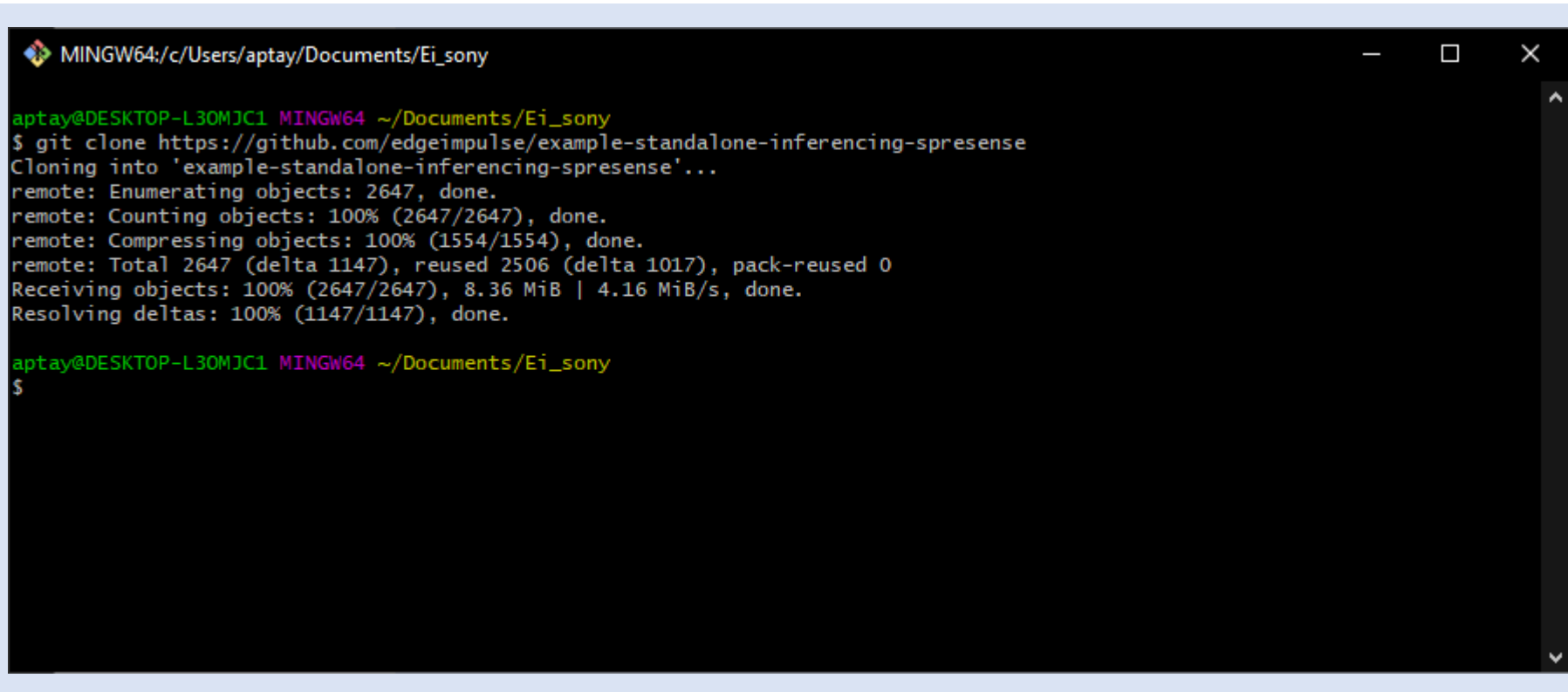
# Lab

## Testing on the board

# Edge Impulse: Creating the Impulse

## Step 1 – Clone the repository at

<https://github.com/edgeimpulse/example-standalone-inferencing-spresense>

A screenshot of a Windows terminal window titled 'MINGW64:/c/Users/aptay/Documents/Ei\_sony'. The terminal shows the command to clone a GitHub repository and its output. The output indicates that the repository was successfully cloned, with details on object enumeration, counting, and compression. The terminal window has a standard Windows title bar with minimize, maximize, and close buttons.

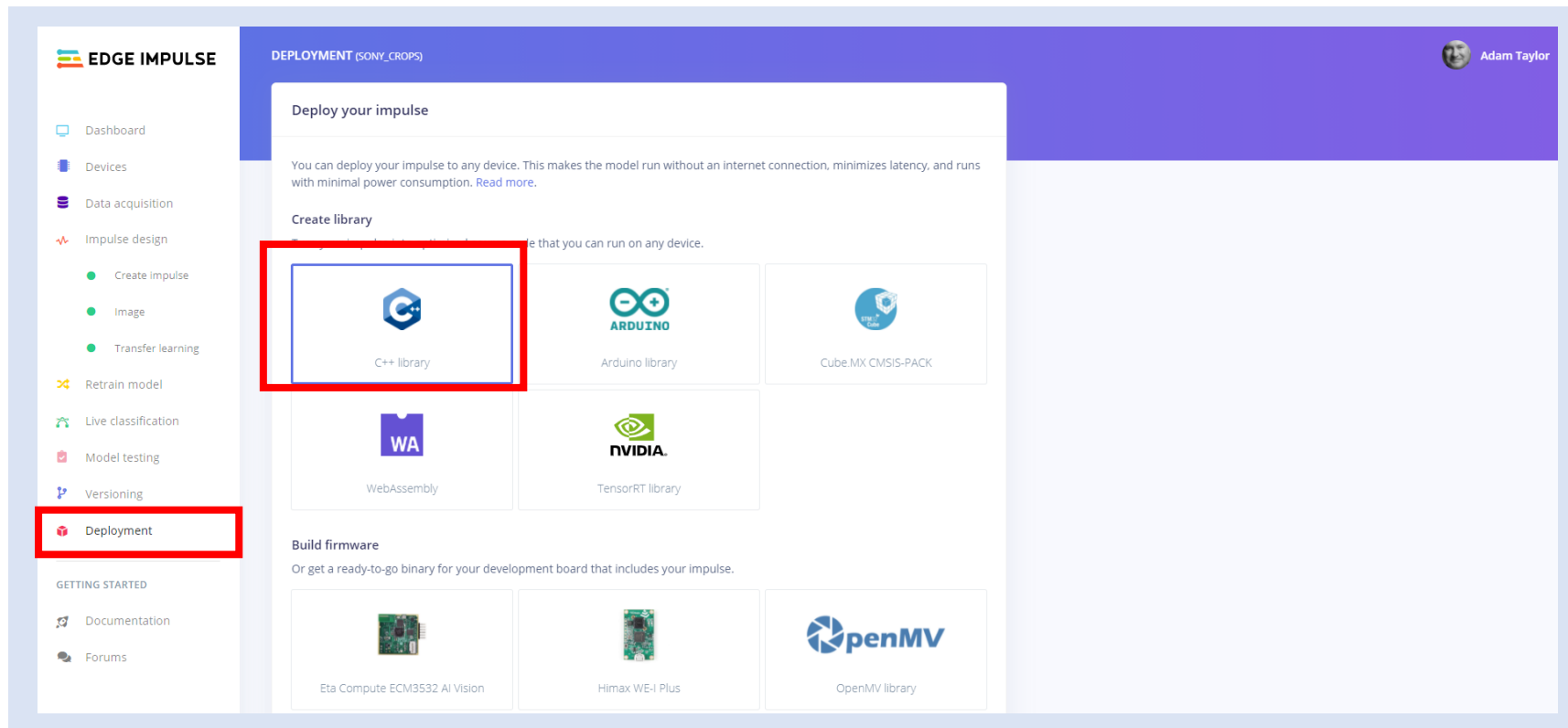
```
MINGW64:/c/Users/aptay/Documents/Ei_sony
aptay@DESKTOP-L30MJC1 MINGW64 ~/Documents/Ei_sony
$ git clone https://github.com/edgeimpulse/example-standalone-inferencing-spresense
Cloning into 'example-standalone-inferencing-spresense'...
remote: Enumerating objects: 2647, done.
remote: Counting objects: 100% (2647/2647), done.
remote: Compressing objects: 100% (1554/1554), done.
remote: Total 2647 (delta 1147), reused 2506 (delta 1017), pack-reused 0
Receiving objects: 100% (2647/2647), 8.36 MiB | 4.16 MiB/s, done.
Resolving deltas: 100% (1147/1147), done.

aptay@DESKTOP-L30MJC1 MINGW64 ~/Documents/Ei_sony
$
```



# Edge Impulse: Creating the Impulse

**Step 2** – Click on the Deployment menu and select C++ library



# Edge Impulse: Creating the Impulse

**Step 3** – Click on build and save the zip file in the same directory as the recently cloned repository

Available optimizations for Transfer learning

<b>Quantized (int8)</b> <a href="#">Currently selected</a>	RAM USAGE 296.8K ROM USAGE 585.2K	LATENCY 3,351 ms ACCURACY -
<b>Unoptimized (float32)</b> <a href="#">Click to select</a>	RAM USAGE 957.0K ROM USAGE 1.6M	LATENCY 12,430 ms ACCURACY -

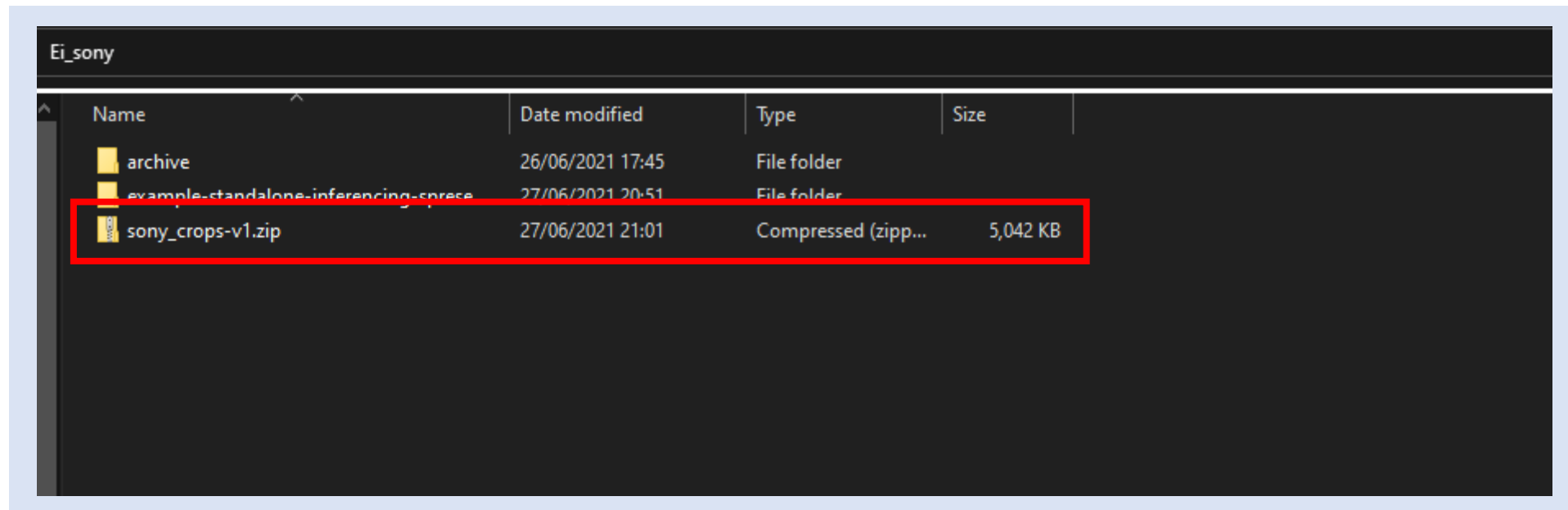
[Analyze optimizations](#)

Estimate for Sony Spresense

[Build](#)

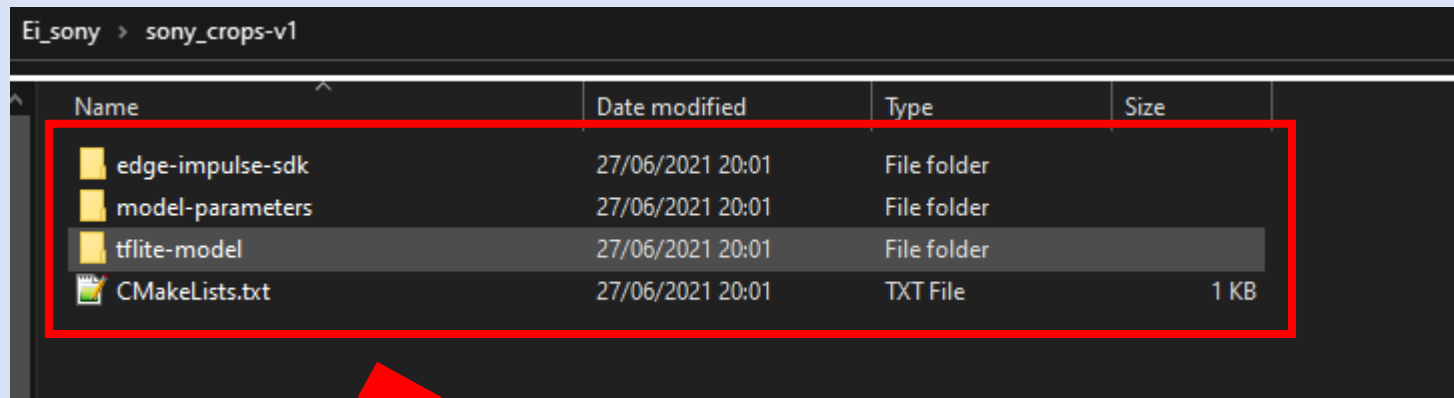
# Edge Impulse: Creating the Impulse

**Step 4** – Unzip the compressed file just downloaded



# Edge Impulse: Creating the Impulse

**Step 5** – Copy the extracted files from sony\_crops-v1 to example-standalone-inferencing-spresense\edge\_impulse



Ei\_sony > sony\_crops-v1

Name	Date modified	Type	Size
edge-impulse-sdk	27/06/2021 20:01	File folder	
model-parameters	27/06/2021 20:01	File folder	
tflite-model	27/06/2021 20:01	File folder	
CMakeLists.txt	27/06/2021 20:01	TXT File	1 KB

Ei\_sony > example-standalone-inferencing-spresense > edge\_impulse

Name	Date modified	Type	Size
README.md	27/06/2021 20:51	MD File	1 KB

# Edge Impulse: Creating the Impulse

**Step 6** – Click on live classification and select the sample you wish to classify – select load sample

The screenshot displays the Edge Impulse web interface. On the left sidebar, the 'Live classification' option is highlighted with a red box. The main content area is titled 'LIVE CLASSIFICATION (SONY\_CROPS)' and features a user profile for Adam Taylor. A notification bar at the top states: 'Did you know? Capture data from any device or development board into the testing category to live classify data - Show options'. The interface is divided into two main sections: 'Classify new data' and 'Classify existing test sample'. The 'Classify new data' section includes fields for 'Device' (No devices connected), 'Sensor', 'Sample length (ms.)' (5000), and 'Frequency', with a 'Start sampling' button. The 'Classify existing test sample' section, which is highlighted with a red box, contains a dropdown menu showing 'maize040arot.jpeg.2919a4hn (maize)' and a 'Load sample' button.

# Edge Impulse: Creating the Impulse

**Step 7 – Click to copy the raw features to the clipboard**

**Classification result**

**Summary**

Name: maize040arot.jpeg.2919a4hn


Expected outcome: maize

CATEGORY	COUNT
Sugar_Cane	0
Wheat	0
maize	1
uncertain	0

**Detailed result** ☐ Show only unknowns

SUGAR_CANE	WHEAT	MAIZE
0	0.06	0.94

**RAW DATA**  
maize040arot.jpeg.2919a4hn

Raw features:  0x8ba2c4, 0x8...x90a7c9, 0x91a8cb, 0x92a9cc, 0x93aacd, 0x96acce, 0x9aafcf, 0x9fb2d0, 0xa2b5d0, 0xa9bad4...

**Image (380 samples)**

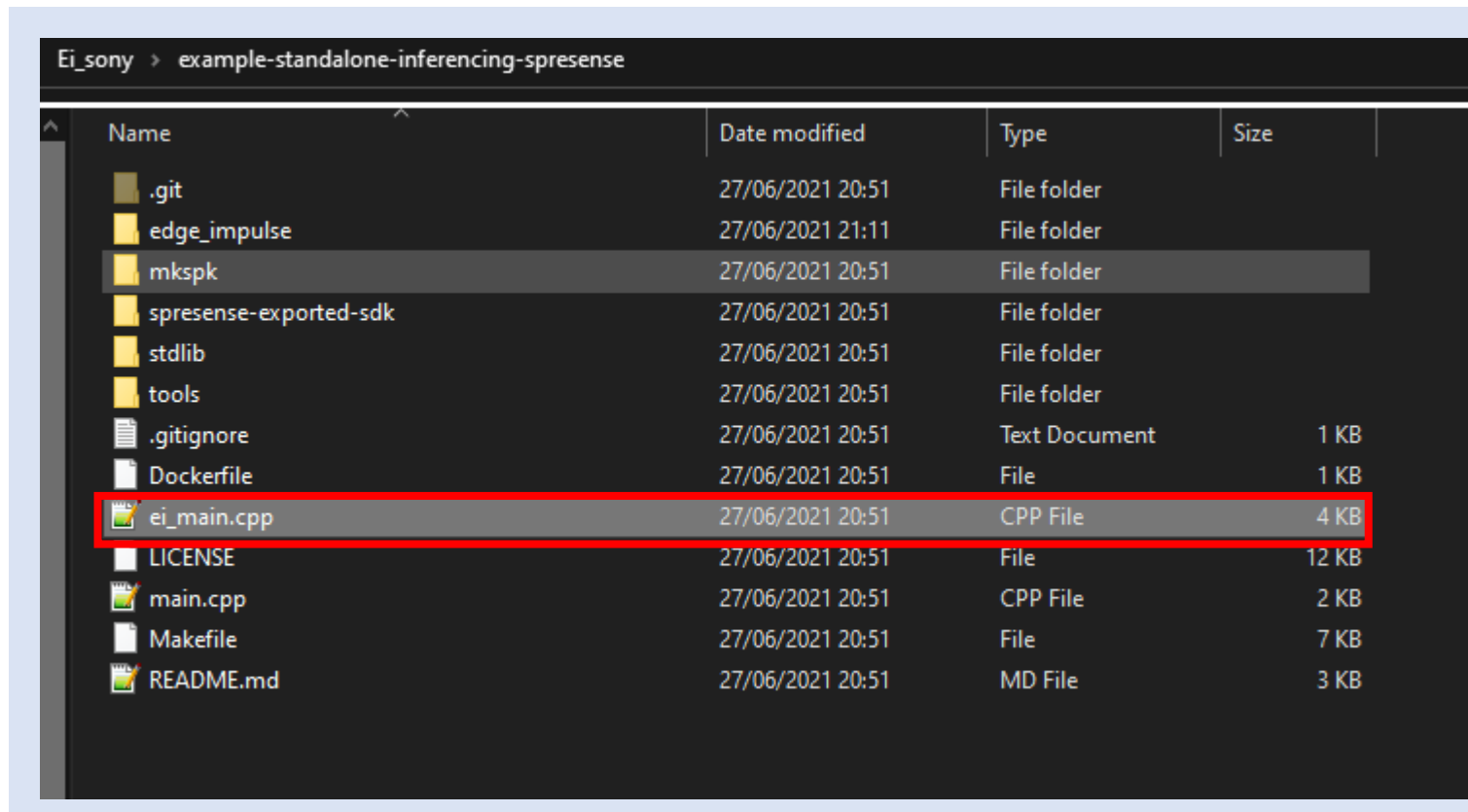
X Axis: Visualization layer 1  
Y Axis: Visualization layer 2  
Z Axis: Visualization layer 3

Legend: Sugar\_Cane (blue), Wheat (orange), maize (red), classification 0 (light blue)

Processed features: 0.6235, 0.6314, 0.6431, 0.6475, 0.6514, 0.6553, 0.6639, 0.6760, 0.6892, 0.6996, 0.7211, 0.7327, 0.7436, 0.74...

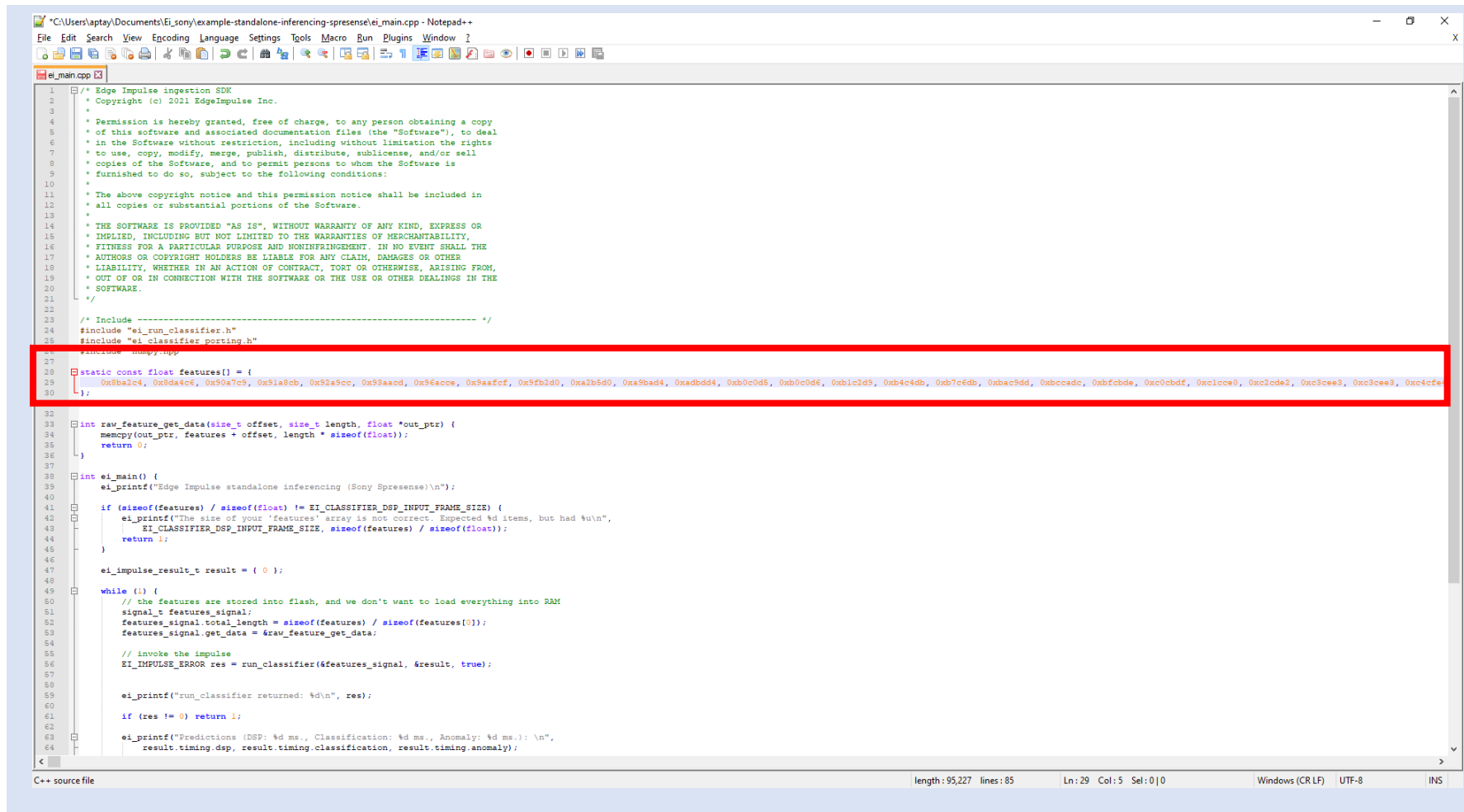
# Edge Impulse: Creating the Impulse

**Step 8** – Open the file ei\_main.cpp which is in the top level of the cloned repository



# Edge Impulse: Creating the Impulse

Step 9 – Copy the data in on line 21 and click save, and close the file

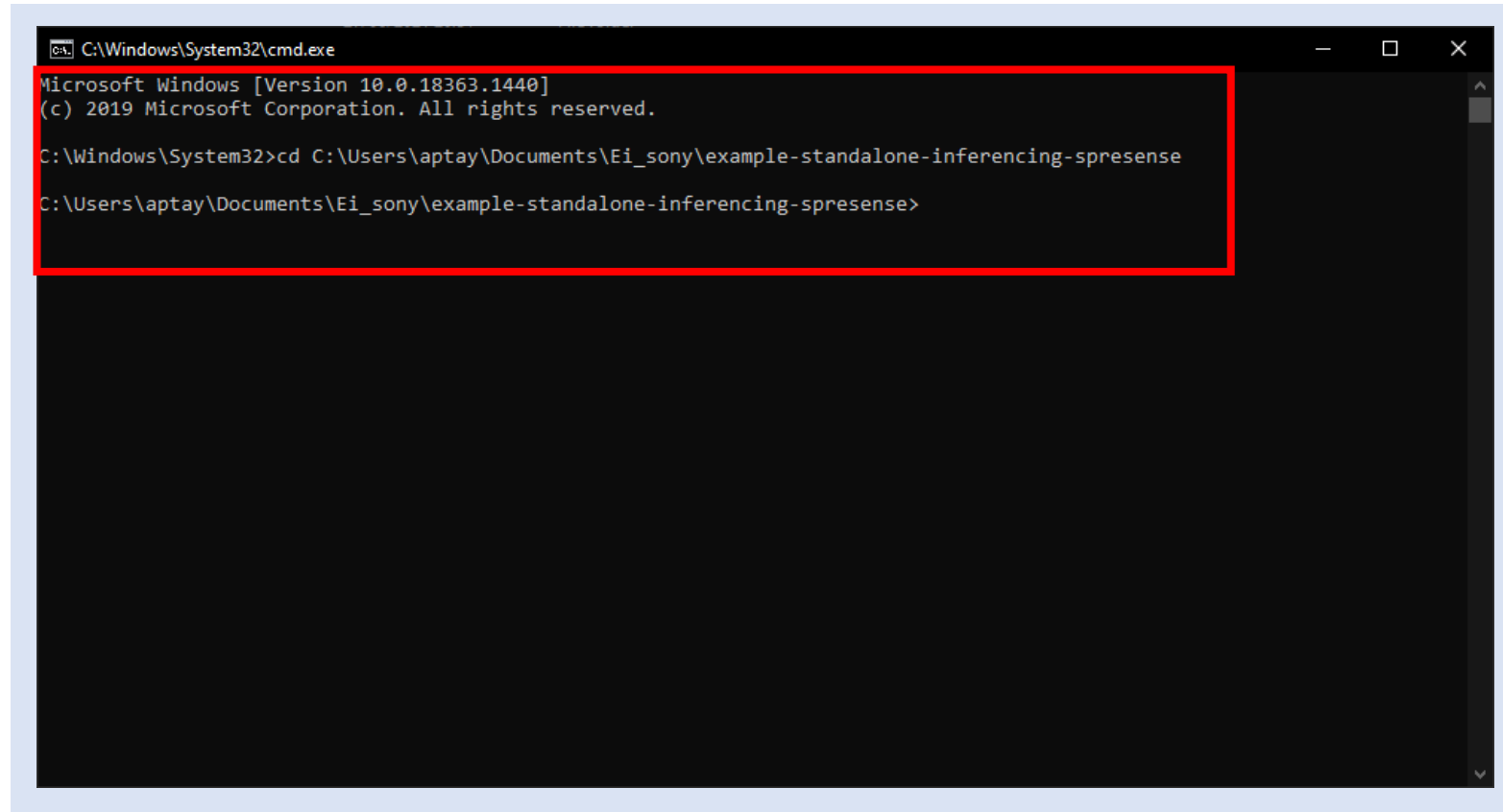


```
1  /* Edge Impulse ingestion SDK
2  * Copyright (c) 2021 EdgeImpulse Inc.
3  *
4  * Permission is hereby granted, free of charge, to any person obtaining a copy
5  * of this software and associated documentation files (the "Software"), to deal
6  * in the Software without restriction, including without limitation the rights
7  * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
8  * copies of the Software, and to permit persons to whom the Software is
9  * furnished to do so, subject to the following conditions:
10 *
11 * The above copyright notice and this permission notice shall be included in
12 * all copies or substantial portions of the Software.
13 *
14 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
15 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
16 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
17 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
18 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
19 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
20 * SOFTWARE.
21 */
22
23 /* Include ----- */
24 #include "ei_run_classifier.h"
25 #include "ei_classifier_porting.h"
26 #include "memory.h"
27
28 static const float features[] = {
29     0x8ba2c4, 0x8da4c6, 0x90a7c5, 0x91a8cb, 0x92a9cc, 0x93aacd, 0x96accc, 0x99aacf, 0x9fb2d0, 0xa2b5d0, 0xa9bad4, 0xabdd4, 0xb0c0d5, 0xb0c0d6, 0xb1c2d9, 0xb4c4db, 0xb7c6db, 0xbac5dd, 0xbccadc, 0xbfcdbd, 0xc0cbdf, 0xc1cce0, 0xc1cde2, 0xc3cee3, 0xc3cee3, 0xc4cfe
30 };
31
32 int raw_feature_get_data(size_t offset, size_t length, float *out_ptr) {
33     memcpy(out_ptr, features + offset, length * sizeof(float));
34     return 0;
35 }
36
37 int ei_main() {
38     ei_printf("Edge Impulse standalone inferencing (Sony Spresense)\n");
39     ei_printf("\n");
40
41     if (sizeof(features) / sizeof(float) != EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE) {
42         ei_printf("The size of your 'features' array is not correct. Expected %d items, but had %u\n",
43             EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE, sizeof(features) / sizeof(float));
44         return 1;
45     }
46
47     ei_impulse_result_t result = { 0 };
48
49     while (1) {
50         // the features are stored into flash, and we don't want to load everything into RAM
51         signal_t features_signal;
52         features_signal.total_length = sizeof(features) / sizeof(features[0]);
53         features_signal.get_data = &raw_feature_get_data;
54
55         // invoke the impulse
56         EI_IMPULSE_ERROR res = run_classifier(&features_signal, &result, true);
57
58         ei_printf("run_classifier returned: %d\n", res);
59
60         if (res != 0) return 1;
61
62         ei_printf("Predictions (DSP: %d ms., Classification: %d ms., Anomaly: %d ms.): \n",
63             result.timing.dsp, result.timing.classification, result.timing.anomaly);
64     }
65 }
```



# Edge Impulse: Creating the Impulse

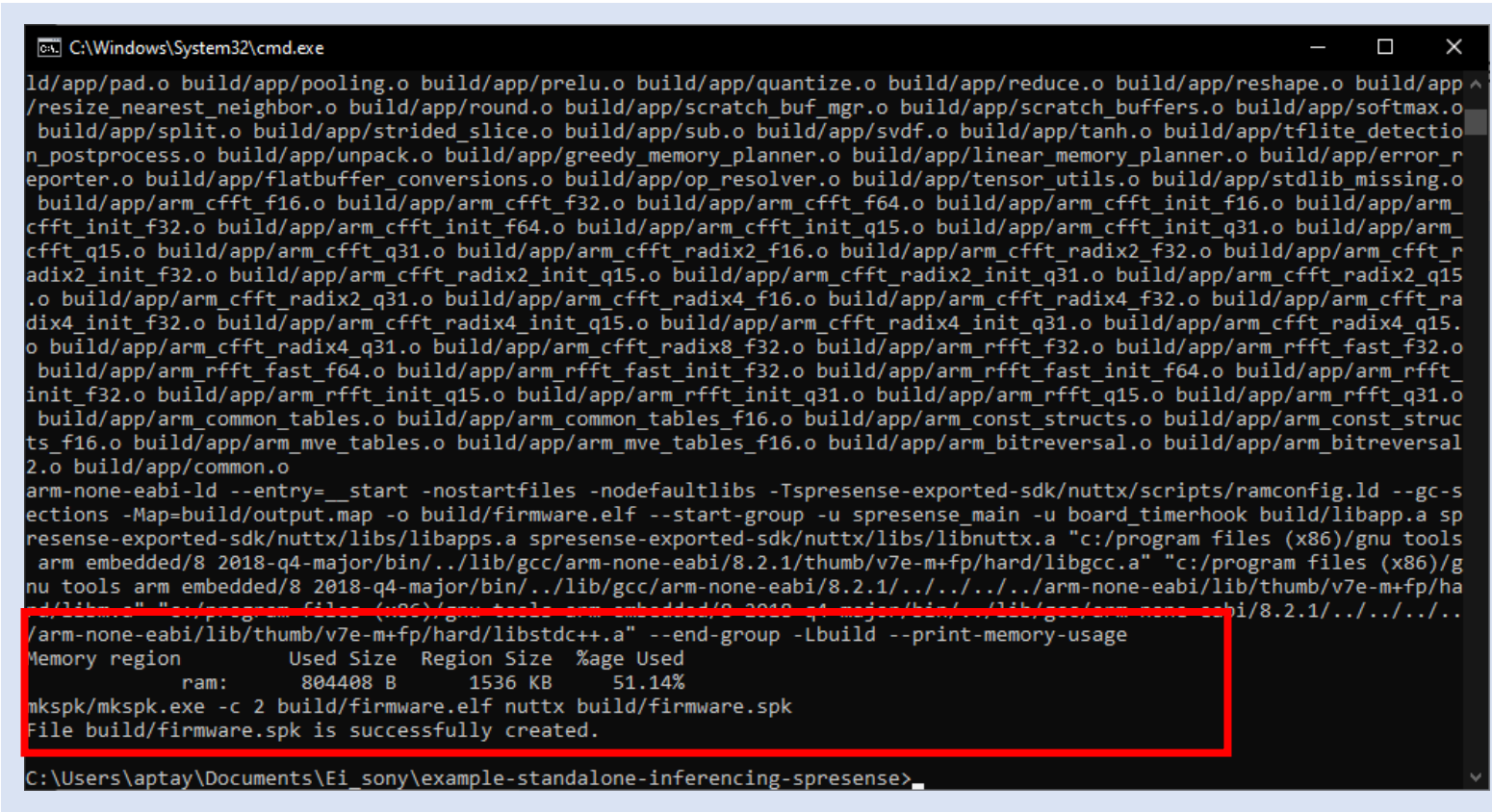
**Step 10** – Open a command window and change directory into the top level of the cloned repository

A screenshot of a Windows Command Prompt window. The title bar reads 'C:\Windows\System32\cmd.exe'. The window contains the following text: 'Microsoft Windows [Version 10.0.18363.1440]', '(c) 2019 Microsoft Corporation. All rights reserved.', 'C:\Windows\System32>cd C:\Users\aptay\Documents\Ei\_sony\example-standalone-inferencing-spresense', and 'C:\Users\aptay\Documents\Ei\_sony\example-standalone-inferencing-spresense>'. A red rectangular box highlights the first three lines of text.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Windows\System32>cd C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense
C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense>
```

# Edge Impulse: Creating the Impulse

**Step 11** – Enter the command `make -j` if you so not see the output below run the command again



```

C:\Windows\System32\cmd.exe

ld/app/pad.o build/app/pooling.o build/app/prelu.o build/app/quantize.o build/app/reduce.o build/app/reshape.o build/app/resize_nearest_neighbor.o build/app/round.o build/app/scratch_buf_mgr.o build/app/scratch_buffers.o build/app/softmax.o build/app/split.o build/app/strided_slice.o build/app/sub.o build/app/svdf.o build/app/tanh.o build/app/tflite_detection_postprocess.o build/app/unpack.o build/app/greedy_memory_planner.o build/app/linear_memory_planner.o build/app/error_reporter.o build/app/flatbuffer_conversions.o build/app/op_resolver.o build/app/tensor_utils.o build/app/stdlib_missing.o build/app/arm_cfft_f16.o build/app/arm_cfft_f32.o build/app/arm_cfft_f64.o build/app/arm_cfft_init_f16.o build/app/arm_cfft_init_f32.o build/app/arm_cfft_init_f64.o build/app/arm_cfft_init_q15.o build/app/arm_cfft_init_q31.o build/app/arm_cfft_q15.o build/app/arm_cfft_q31.o build/app/arm_cfft_radix2_f16.o build/app/arm_cfft_radix2_f32.o build/app/arm_cfft_radix2_init_f32.o build/app/arm_cfft_radix2_init_q15.o build/app/arm_cfft_radix2_init_q31.o build/app/arm_cfft_radix2_q15.o build/app/arm_cfft_radix2_q31.o build/app/arm_cfft_radix4_f16.o build/app/arm_cfft_radix4_f32.o build/app/arm_cfft_radix4_init_f32.o build/app/arm_cfft_radix4_init_q15.o build/app/arm_cfft_radix4_init_q31.o build/app/arm_cfft_radix4_q15.o build/app/arm_cfft_radix4_q31.o build/app/arm_cfft_radix8_f32.o build/app/arm_rfft_f32.o build/app/arm_rfft_fast_f32.o build/app/arm_rfft_fast_f64.o build/app/arm_rfft_fast_init_f32.o build/app/arm_rfft_fast_init_f64.o build/app/arm_rfft_init_f32.o build/app/arm_rfft_init_q15.o build/app/arm_rfft_init_q31.o build/app/arm_rfft_q15.o build/app/arm_rfft_q31.o build/app/arm_common_tables.o build/app/arm_common_tables_f16.o build/app/arm_const_structs.o build/app/arm_const_structs_f16.o build/app/arm_mve_tables.o build/app/arm_mve_tables_f16.o build/app/arm_bitreversal.o build/app/arm_bitreversal2.o build/app/common.o
arm-none-eabi-ld --entry=__start -nostartfiles -nodefaultlibs -Tspresense-exported-sdk/nuttx/scripts/ramconfig.ld --gc-sections -Map=build/output.map -o build/firmware.elf --start-group -u spresense_main -u board_timerhook build/libapp.a spresense-exported-sdk/nuttx/libs/libapps.a spresense-exported-sdk/nuttx/libs/libnuttx.a "c:/program files (x86)/gnu tools arm embedded/8 2018-q4-major/bin/../lib/gcc/arm-none-eabi/8.2.1/thumb/v7e-m+fp/hard/libgcc.a" "c:/program files (x86)/gnu tools arm embedded/8 2018-q4-major/bin/../lib/gcc/arm-none-eabi/8.2.1/../../../../arm-none-eabi/lib/thumb/v7e-m+fp/hard/libm.a" "c:/program files (x86)/gnu tools arm embedded/8 2018-q4-major/bin/../lib/gcc/arm-none-eabi/8.2.1/../../../../arm-none-eabi/lib/thumb/v7e-m+fp/hard/libstdc++.a" --end-group -lbuild --print-memory-usage
Memory region      Used Size  Region Size  %age Used
      ram:         804408 B       1536 KB       51.14%
mkspk/mkspk.exe -c 2 build/firmware.elf nuttx build/firmware.spk
File build/firmware.spk is successfully created.

C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense>

```

# Edge Impulse: Creating the Impulse

**Step 12** – Connect the Sony Spresense to your development machine



# Edge Impulse: Creating the Impulse

**Step 13** – Enter the command make flash this will flash the application to the Spresense

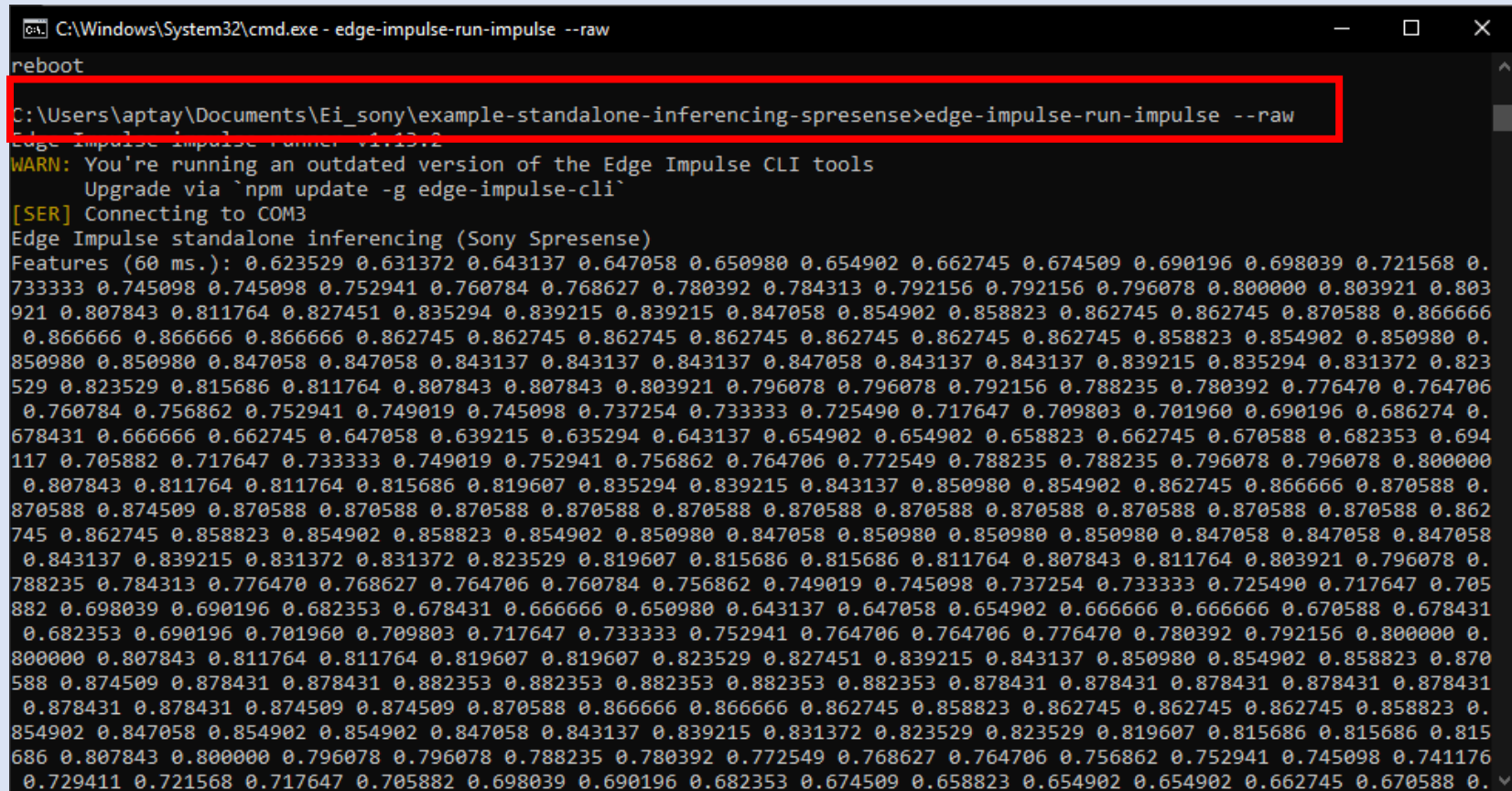
```
C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense>make flash
tools/flash_writer.py -s -d -b 115200 -n build/firmware.spk
[119;1H[JB
[A[K(B[m[33m?(B[m](B[m Select your device: COM3 - Silicon Labs CP210x USB to UART Bridge (COM3)
[34m> COM3 - Silicon Labs CP210x USB to UART Bridge (COM3)(B[m[K

Using device COM3
>>> Install files ...
install -b 115200
Install build/firmware.spk
|0%-----50%-----100|
#####
776368 bytes loaded.
Package validation is OK.
Saving package to "nuttx"
updater# sync
updater# Restarting the board ...
reboot

C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense>
```

# Edge Impulse: Creating the Impulse

**Step 14** – In the command window enter the command `edge-impulse-run-impulse --raw`



```
C:\Windows\System32\cmd.exe - edge-impulse-run-impulse --raw
reboot
C:\Users\aptay\Documents\Ei_sony\example-standalone-inferencing-spresense>edge-impulse-run-impulse --raw
Edge Impulse Impulse Runner v1.13.2
WARN: You're running an outdated version of the Edge Impulse CLI tools
       Upgrade via `npm update -g edge-impulse-cli`
[SER] Connecting to COM3
Edge Impulse standalone inferencing (Sony Spresense)
Features (60 ms.): 0.623529 0.631372 0.643137 0.647058 0.650980 0.654902 0.662745 0.674509 0.690196 0.698039 0.721568 0.
733333 0.745098 0.745098 0.752941 0.760784 0.768627 0.780392 0.784313 0.792156 0.792156 0.796078 0.800000 0.803921 0.803
921 0.807843 0.811764 0.827451 0.835294 0.839215 0.839215 0.847058 0.854902 0.858823 0.862745 0.862745 0.870588 0.866666
0.866666 0.866666 0.866666 0.862745 0.862745 0.862745 0.862745 0.862745 0.862745 0.862745 0.858823 0.854902 0.850980 0.
850980 0.850980 0.847058 0.847058 0.843137 0.843137 0.843137 0.847058 0.843137 0.843137 0.839215 0.835294 0.831372 0.823
529 0.823529 0.815686 0.811764 0.807843 0.807843 0.803921 0.796078 0.796078 0.792156 0.788235 0.780392 0.776470 0.764706
0.760784 0.756862 0.752941 0.749019 0.745098 0.737254 0.733333 0.725490 0.717647 0.709803 0.701960 0.690196 0.686274 0.
678431 0.666666 0.662745 0.647058 0.639215 0.635294 0.643137 0.654902 0.654902 0.658823 0.662745 0.670588 0.682353 0.694
117 0.705882 0.717647 0.733333 0.749019 0.752941 0.756862 0.764706 0.772549 0.788235 0.788235 0.796078 0.796078 0.800000
0.807843 0.811764 0.811764 0.815686 0.819607 0.835294 0.839215 0.843137 0.850980 0.854902 0.862745 0.866666 0.870588 0.
870588 0.874509 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.862
745 0.862745 0.858823 0.854902 0.858823 0.854902 0.850980 0.847058 0.850980 0.850980 0.850980 0.847058 0.847058 0.847058
0.843137 0.839215 0.831372 0.831372 0.823529 0.819607 0.815686 0.815686 0.811764 0.807843 0.811764 0.803921 0.796078 0.
788235 0.784313 0.776470 0.768627 0.764706 0.760784 0.756862 0.749019 0.745098 0.737254 0.733333 0.725490 0.717647 0.705
882 0.698039 0.690196 0.682353 0.678431 0.666666 0.650980 0.643137 0.647058 0.654902 0.666666 0.666666 0.670588 0.678431
0.682353 0.690196 0.701960 0.709803 0.717647 0.733333 0.752941 0.764706 0.764706 0.776470 0.780392 0.792156 0.800000 0.
800000 0.807843 0.811764 0.811764 0.819607 0.819607 0.823529 0.827451 0.839215 0.843137 0.850980 0.854902 0.858823 0.870
588 0.874509 0.878431 0.878431 0.882353 0.882353 0.882353 0.882353 0.882353 0.878431 0.878431 0.878431 0.878431 0.878431
0.878431 0.878431 0.874509 0.874509 0.870588 0.866666 0.866666 0.862745 0.858823 0.862745 0.862745 0.862745 0.858823 0.
854902 0.847058 0.854902 0.854902 0.847058 0.843137 0.839215 0.831372 0.823529 0.823529 0.819607 0.815686 0.815686 0.815
686 0.807843 0.800000 0.796078 0.796078 0.788235 0.780392 0.772549 0.768627 0.764706 0.756862 0.752941 0.745098 0.741176
0.729411 0.721568 0.717647 0.705882 0.698039 0.690196 0.682353 0.674509 0.658823 0.654902 0.654902 0.662745 0.670588 0.
```



# Edge Impulse: Creating the Impulse

**Step 15** – Scroll down the processing results and you will see the classification, press cntrl-c to stop the application – compare this with the predicted accuracy of the model

```
C:\Windows\System32\cmd.exe - edge-impulse-run-impulse --raw
035294 0.078431 0.035294 0.031372 0.070588
Predictions (time: 3844 ms.):
Sugar_Cane: 0.003906
Wheat: 0.039062
maize: 0.960937
run_classifier returned: 0
Predictions (DSP: 60 ms., Classification: 3844 ms., Anomaly: 0 ms.):
[0.003906, 0.039062, 0.960937]
Features (60 ms.): 0.623529 0.631372 0.643137 0.647058 0.650980 0.654902 0.662745 0.674509 0.690196 0.698039 0.721568 0.
733333 0.745098 0.745098 0.752941 0.760784 0.768627 0.780392 0.784313 0.792156 0.792156 0.796078 0.800000 0.803921 0.803
921 0.807843 0.811764 0.827451 0.835294 0.839215 0.839215 0.847058 0.854902 0.858823 0.862745 0.862745 0.870588 0.866666
0.866666 0.866666 0.866666 0.862745 0.862745 0.862745 0.862745 0.862745 0.862745 0.858823 0.854902 0.850980 0.
850980 0.850980 0.847058 0.847058 0.843137 0.843137 0.843137 0.847058 0.843137 0.843137 0.839215 0.835294 0.831372 0.823
529 0.823529 0.815686 0.811764 0.807843 0.807843 0.803921 0.796078 0.796078 0.792156 0.788235 0.780392 0.776470 0.764706
0.760784 0.756862 0.752941 0.749019 0.745098 0.737254 0.733333 0.725490 0.717647 0.709803 0.701960 0.690196 0.686274 0.
678431 0.666666 0.662745 0.647058 0.639215 0.635294 0.643137 0.654902 0.654902 0.658823 0.662745 0.670588 0.682353 0.694
117 0.705882 0.717647 0.733333 0.749019 0.752941 0.756862 0.764706 0.772549 0.788235 0.788235 0.796078 0.796078 0.800000
0.807843 0.811764 0.811764 0.815686 0.819607 0.835294 0.839215 0.843137 0.850980 0.854902 0.862745 0.866666 0.870588 0.
870588 0.874509 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.870588 0.862
745 0.862745 0.858823 0.854902 0.858823 0.854902 0.850980 0.847058 0.850980 0.850980 0.850980 0.847058 0.847058 0.847058
0.843137 0.839215 0.831372 0.831372 0.823529 0.819607 0.815686 0.815686 0.811764 0.807843 0.811764 0.803921 0.796078 0.
788235 0.784313 0.776470 0.768627 0.764706 0.760784 0.756862 0.749019 0.745098 0.737254 0.733333 0.725490 0.717647 0.705
882 0.698039 0.690196 0.682353 0.678431 0.666666 0.650980 0.643137 0.647058 0.654902 0.666666 0.666666 0.670588 0.678431
0.682353 0.690196 0.701960 0.709803 0.717647 0.733333 0.752941 0.764706 0.764706 0.776470 0.780392 0.792156 0.800000 0.
800000 0.807843 0.811764 0.811764 0.819607 0.819607 0.823529 0.827451 0.839215 0.843137 0.850980 0.854902 0.858823 0.870
588 0.874509 0.878431 0.878431 0.882353 0.882353 0.882353 0.882353 0.882353 0.878431 0.878431 0.878431 0.878431 0.878431
0.878431 0.878431 0.874509 0.874509 0.870588 0.866666 0.866666 0.862745 0.858823 0.862745 0.862745 0.862745 0.858823 0.
854902 0.847058 0.854902 0.854902 0.847058 0.843137 0.839215 0.831372 0.823529 0.823529 0.819607 0.815686 0.815686 0.815
686 0.807843 0.800000 0.796078 0.796078 0.788235 0.780392 0.772549 0.768627 0.764706 0.756862 0.752941 0.745098 0.741176
0.729411 0.721568 0.717647 0.705882 0.698039 0.690196 0.682353 0.674509 0.658823 0.654902 0.654902 0.662745 0.670588 0.
```