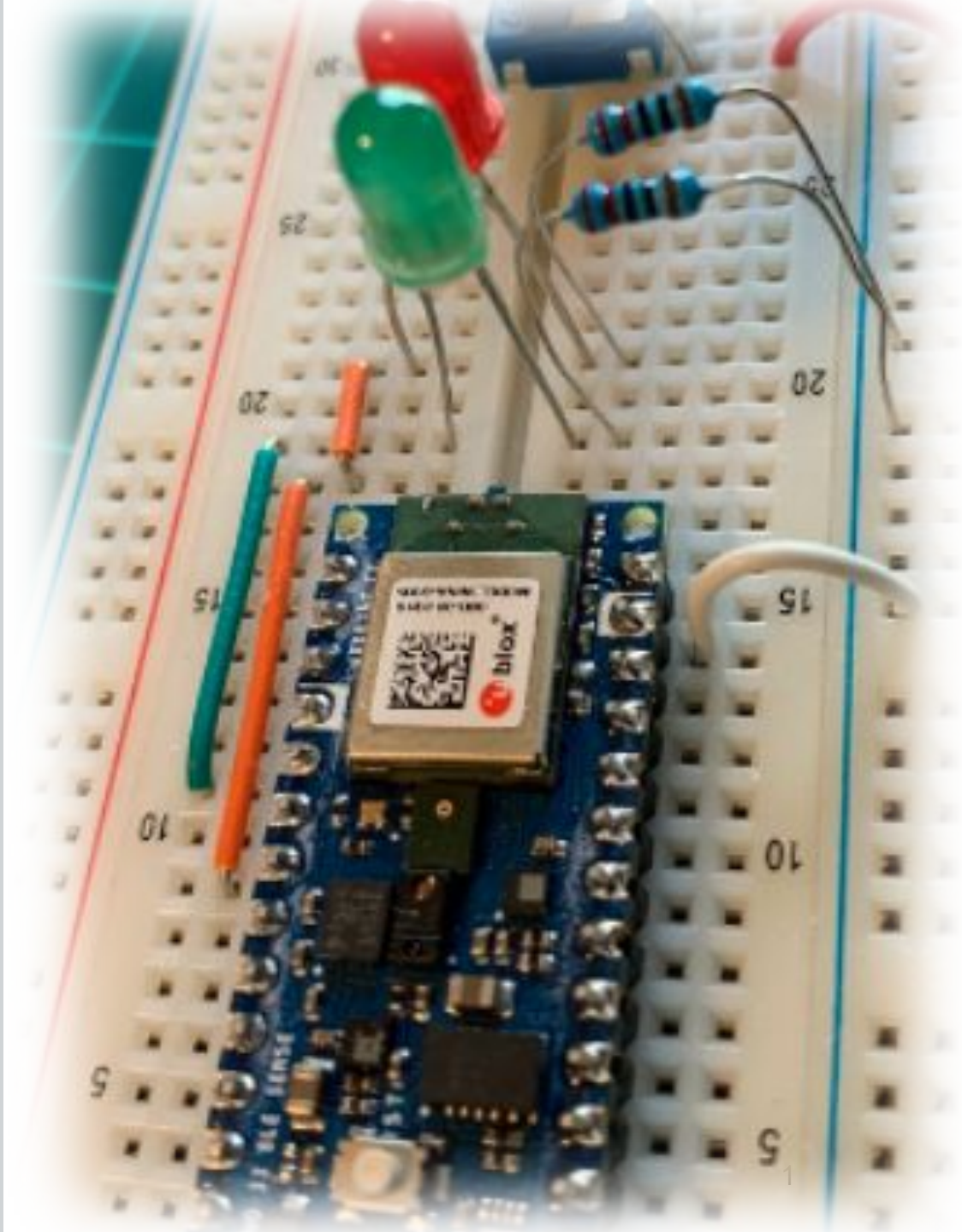
IESTI01 – TinyML

Embedded Machine Learning

13. Preventing Overfitting



Prof. Marcelo Rovai
UNIFEI

Preventing Overfitting

Preventing Overfitting

+Data

+Data

+Data

+Data

Preventing Overfitting

+Data

+Data

+Data

+Data

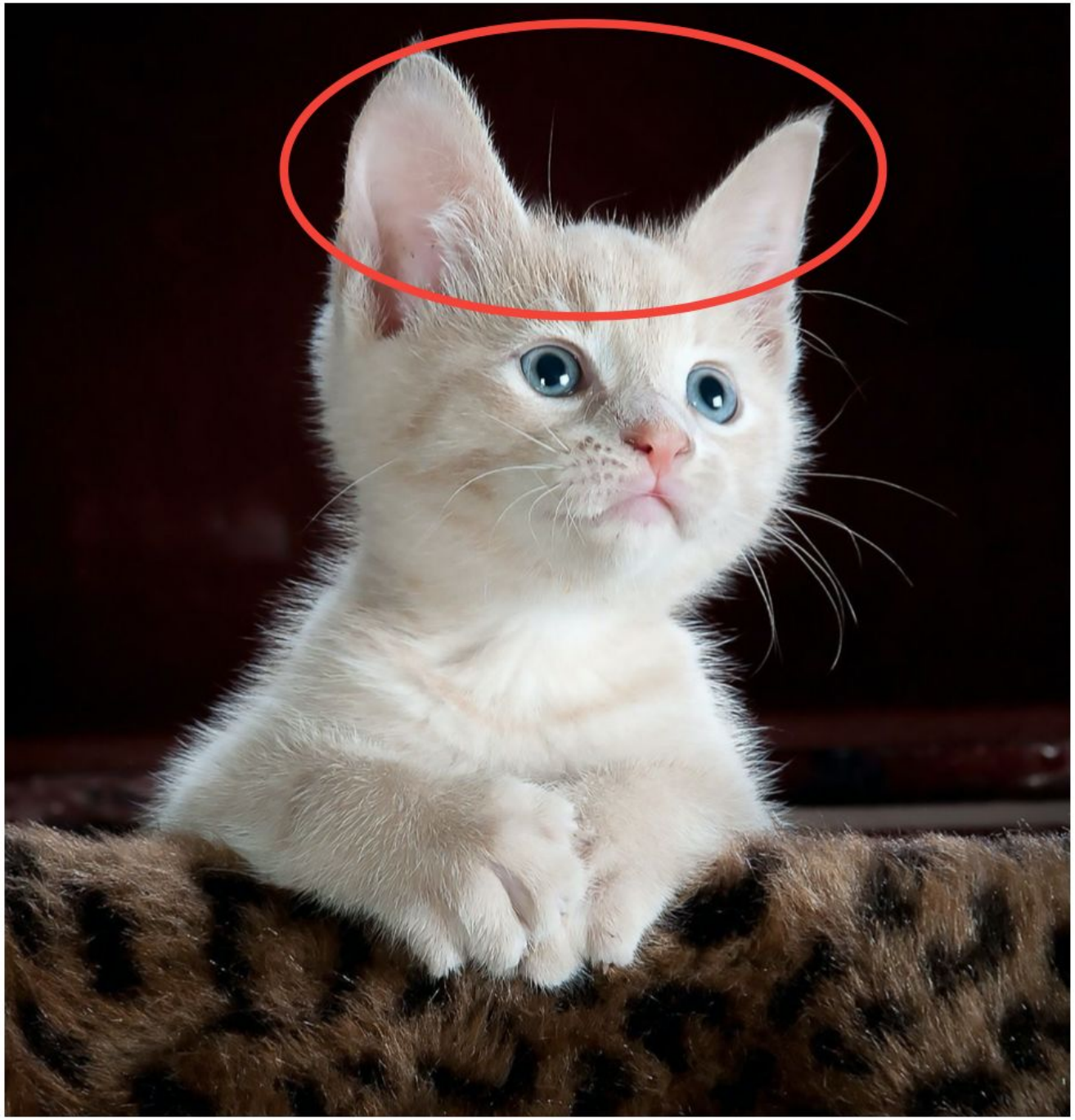
But what to do if we do not have more data?

- Data Augmentation (artificial)
- Transfer Learning
- Early Stopping
- Dropout Regularization

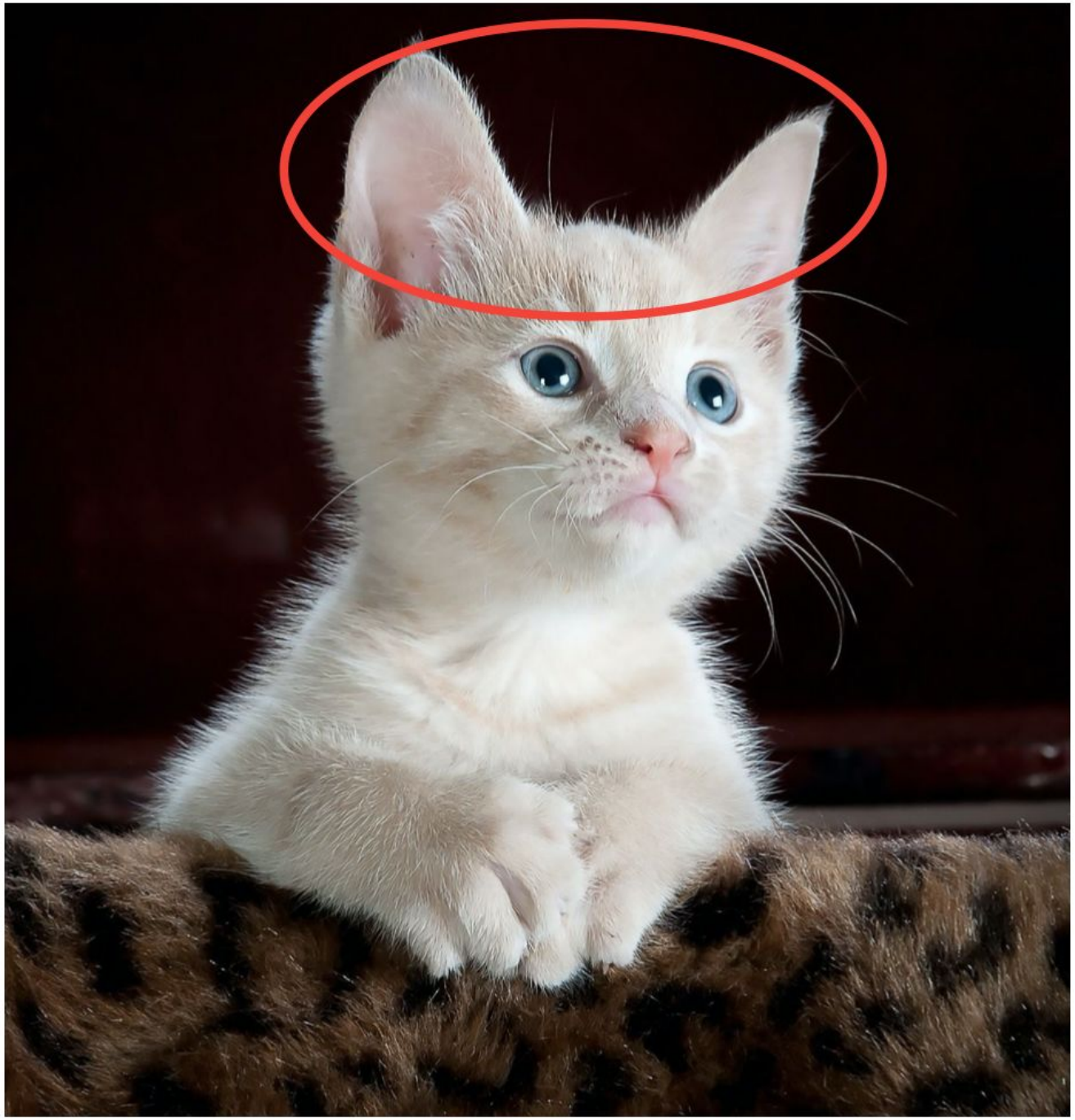
Preventing Overfitting

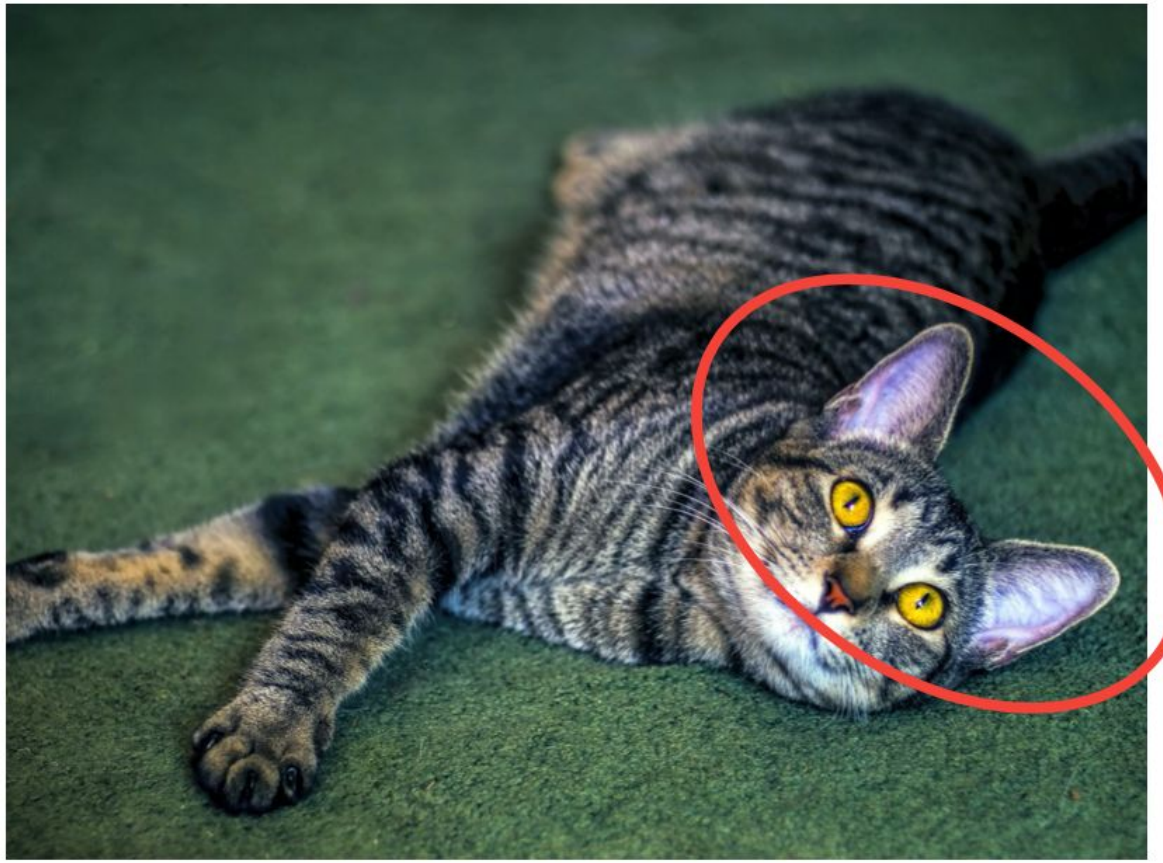
More Data, Data Augmentation (artificial)

Overfitting generally occurs when there are a small number of training examples. Data augmentation takes the approach of generating additional training data from your existing examples by augmenting them using random transformations that yield believable-looking images. This helps expose the model to more aspects of the data and generalize better.



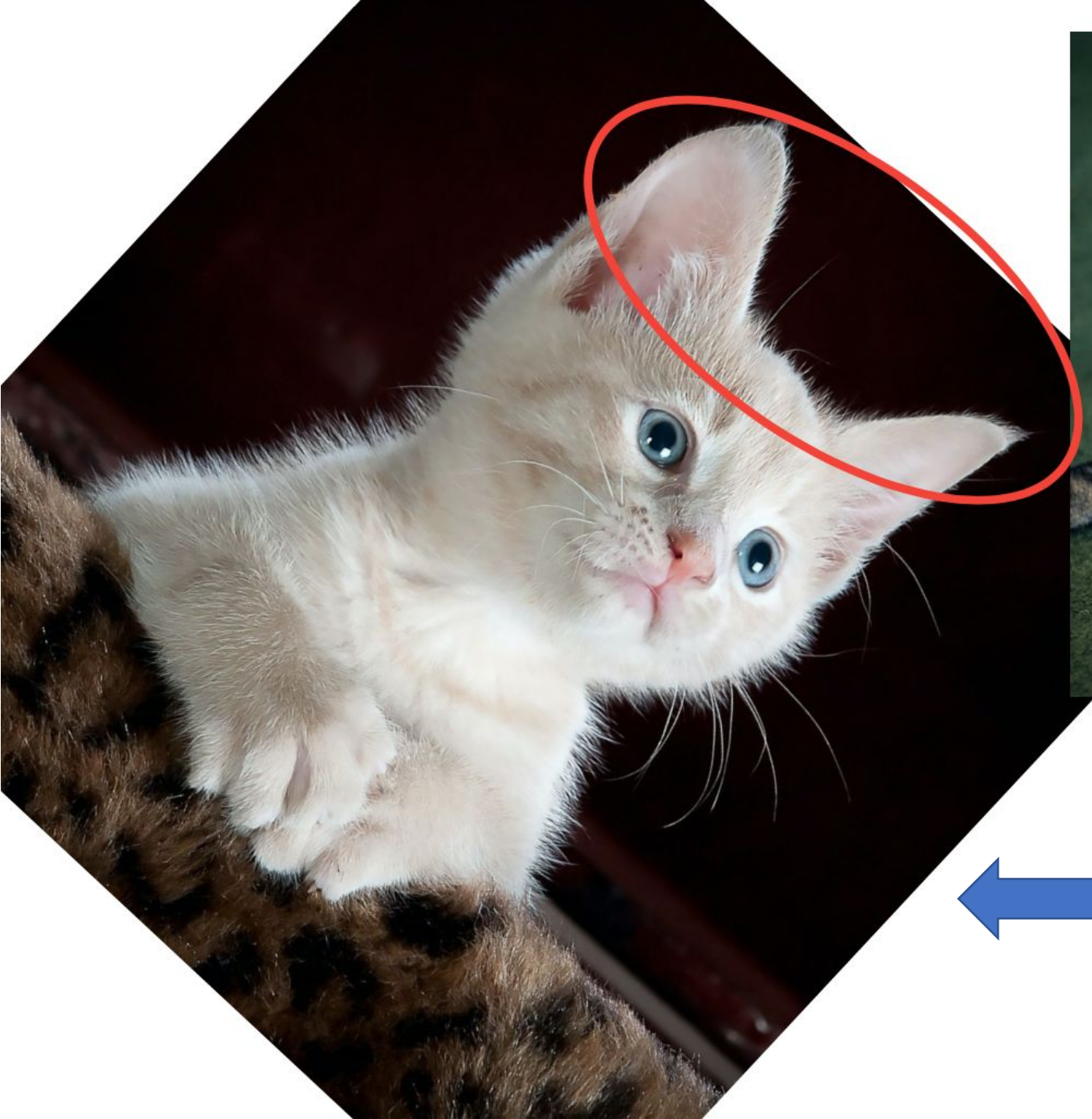
Training

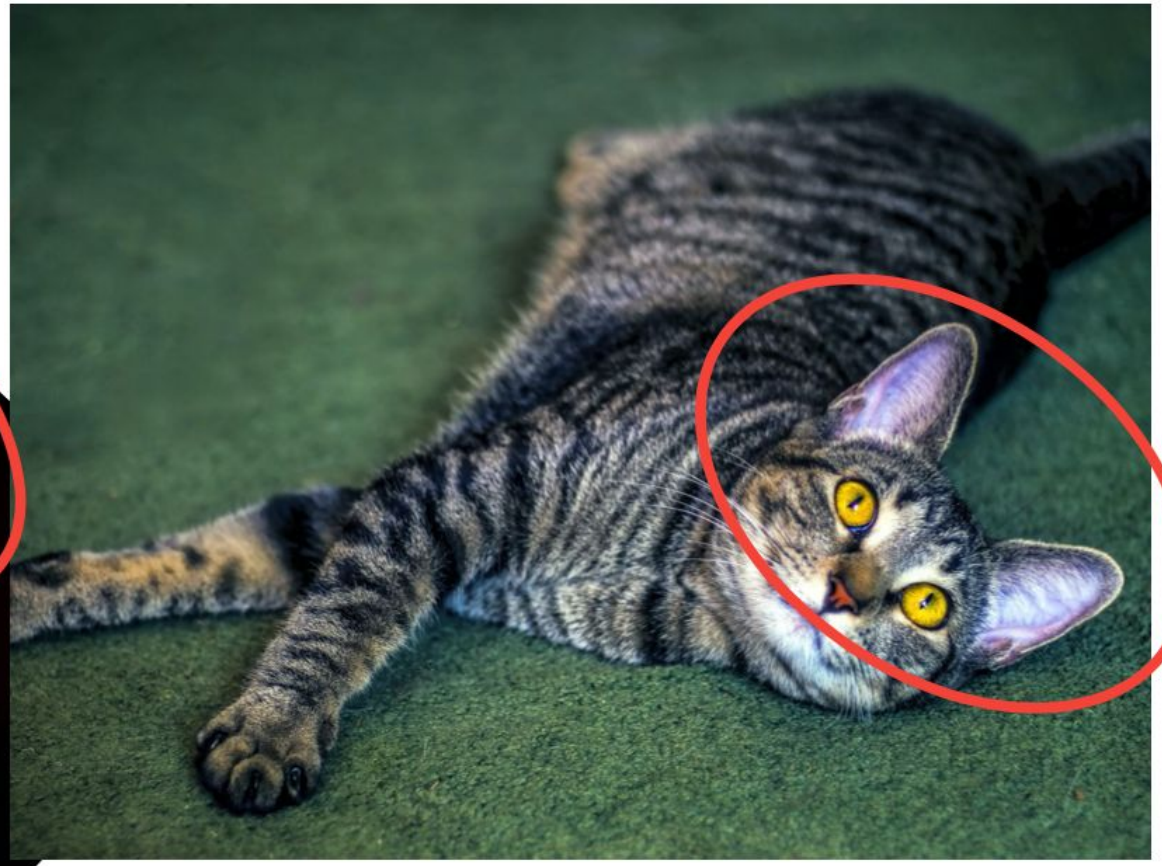




Inference

Training





Training with
Data Augmentation

Using Keras preprocessing layers

```
data_augmentation = tf.keras.Sequential([
  layers.RandomFlip("horizontal_and_vertical"),
  layers.RandomRotation(0.2),
])
```

```
plt.figure(figsize=(10, 10))
for i in range(9):
  augmented_image = data_augmentation(image)
  ax = plt.subplot(3, 3, i + 1)
  plt.imshow(augmented_image[0])
  plt.axis("off")
```

There are a variety of preprocessing layers you can use for data augmentation including:

- tf.keras.layers.RandomContrast,
- tf.keras.layers.RandomCrop,
- tf.keras.layers.RandomZoom,
- and others.

Using tf.image

```
1 flipped = tf.image.flip_left_right(image)
2 visualize(image, flipped)
```

Original image

Augmented image

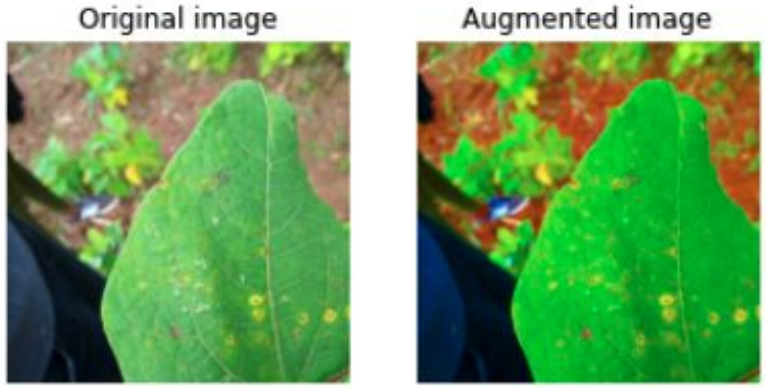
```
1 rotated = tf.image.rot90(image)
2 visualize(image, rotated)
```

Original image

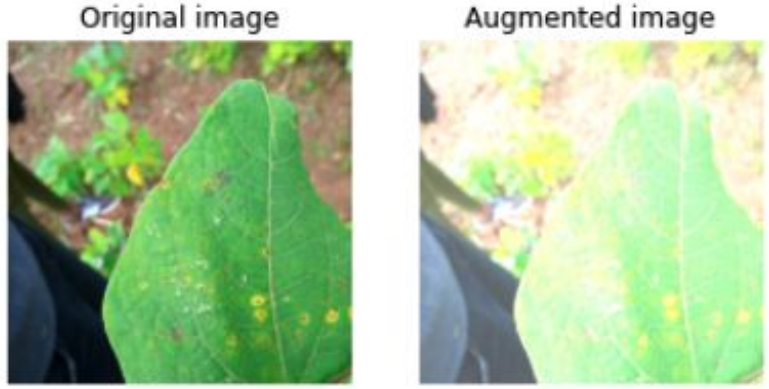
Augmented image

Using tf.image

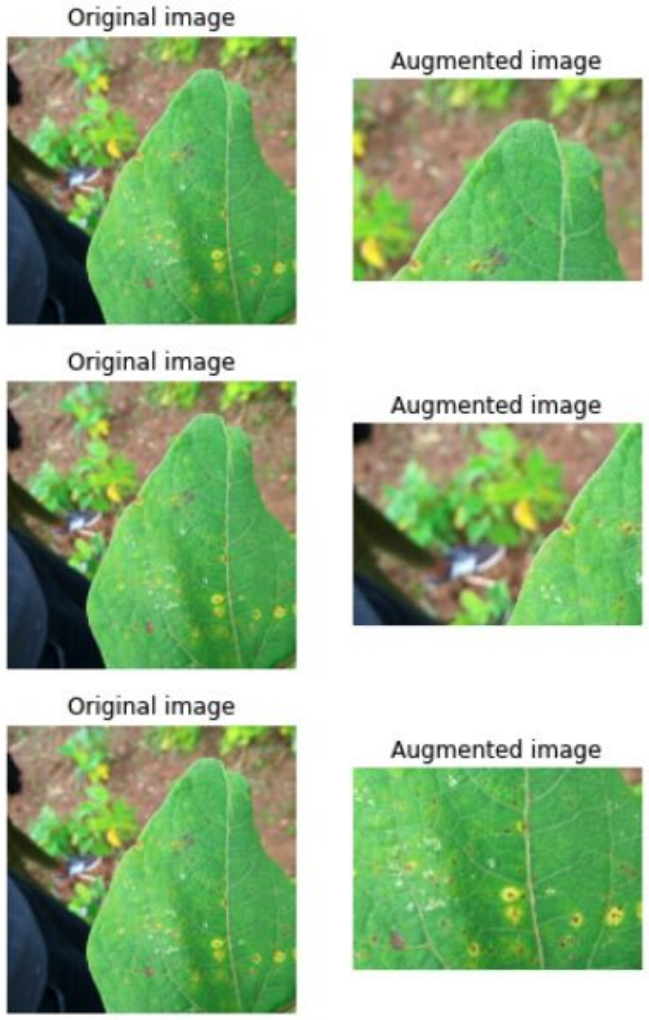
```
saturated = tf.image.adjust_saturation(image, 3)
visualize(image, saturated)
```




```
bright = tf.image.adjust_brightness(image, 0.4)
visualize(image, bright)
```



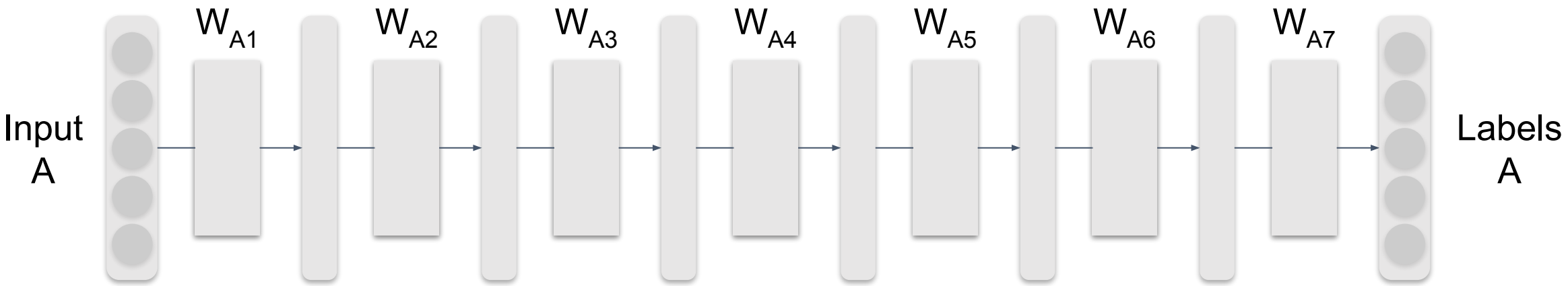

```
for i in range(3):
  seed = (i, 0)  # tuple of size (2,)
  stateless_random_crop = tf.image.stateless_random_crop(
      image, size=[210, 300, 3], seed=seed)
  visualize(image, stateless_random_crop)
```



Preventing Overfitting

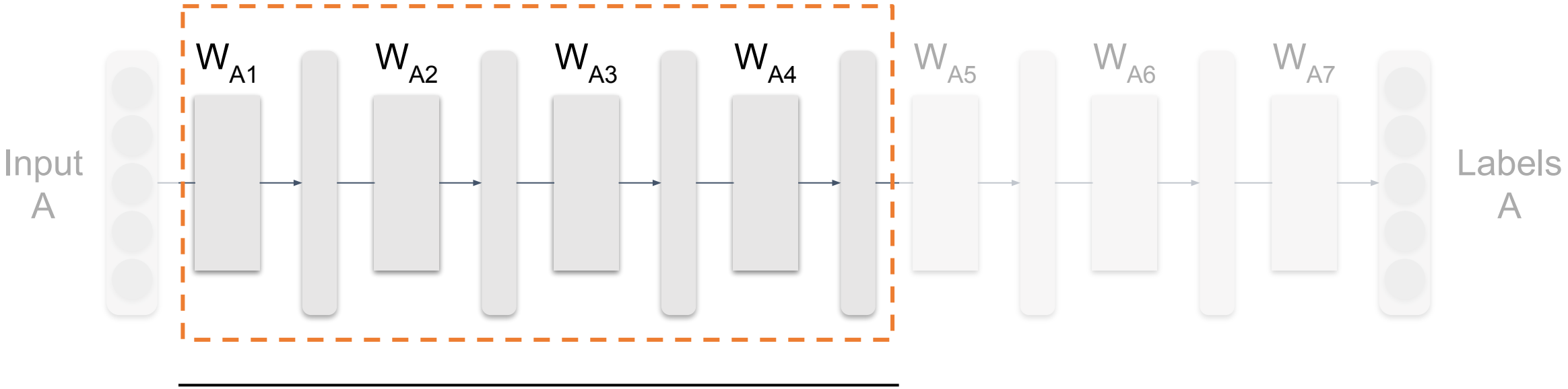
Transfer Learning

End Result of Training



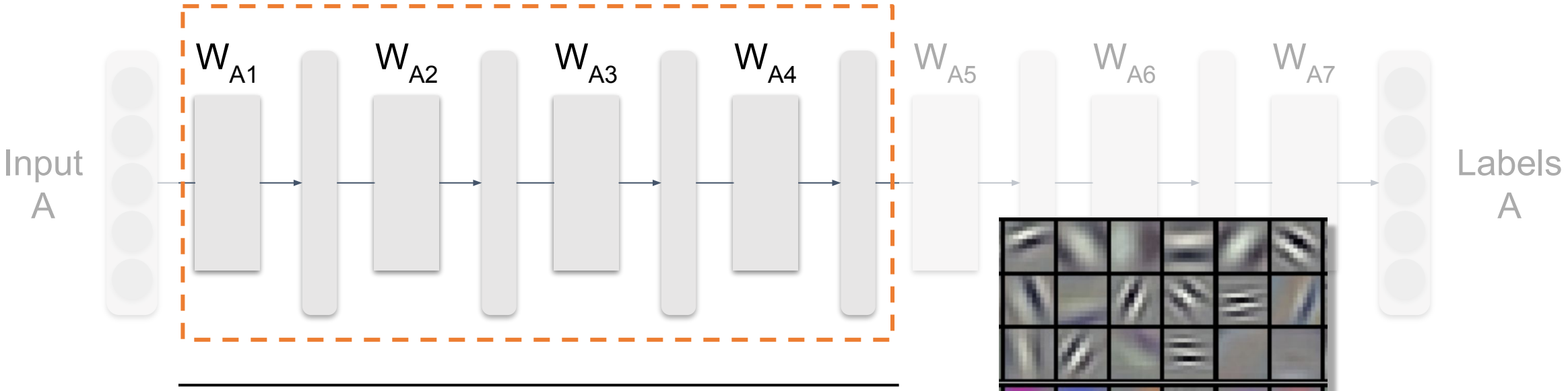
The end result of the training is to learn the weights of the neural network model.

End Result of Training

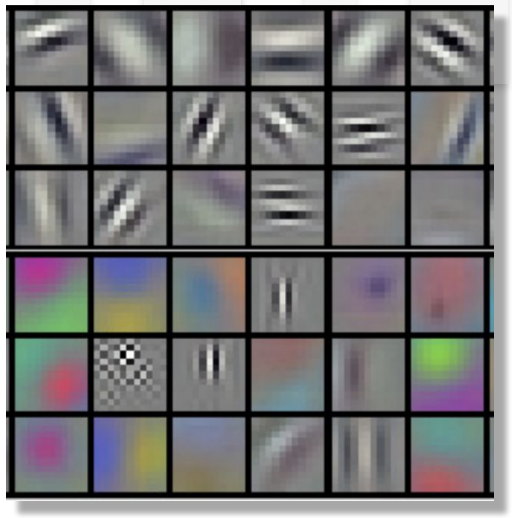


Learns ***general features*** irrespective of task

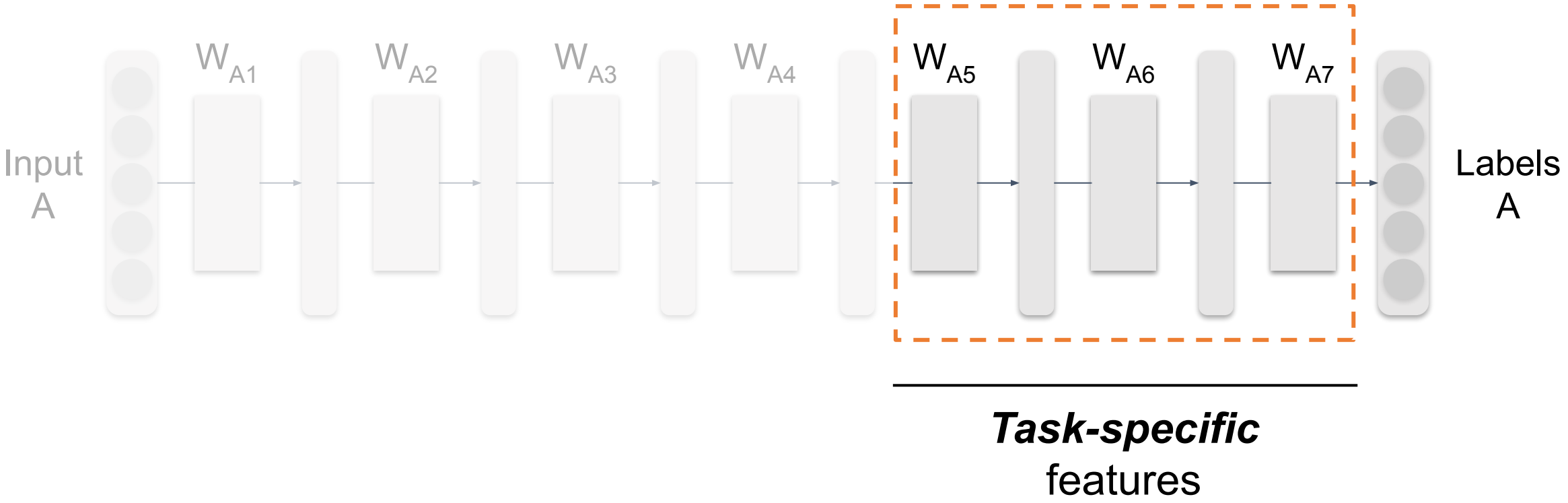
End Result of Training



Learns ***general features*** irrespective of task



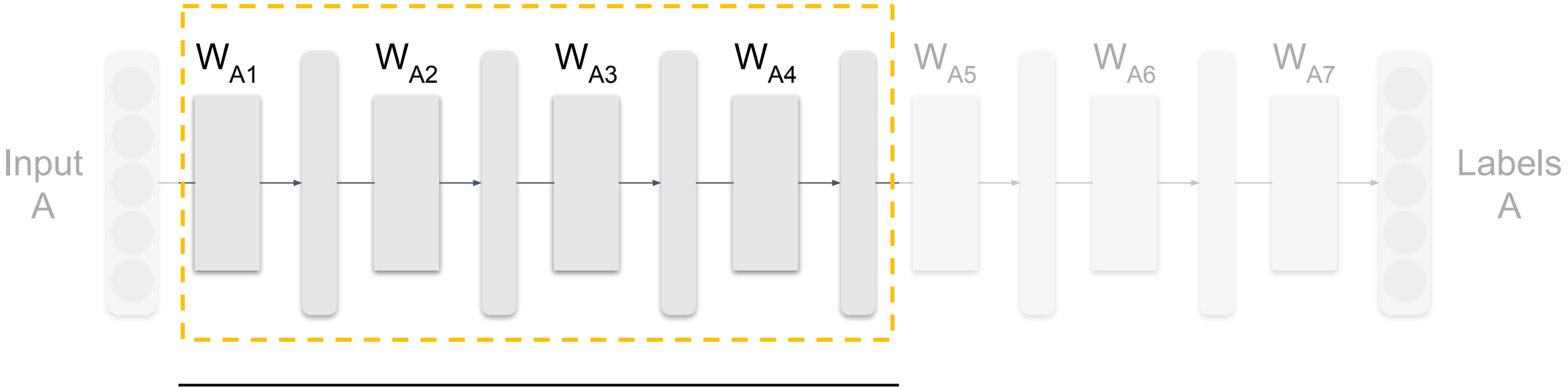
End Result of Training




Source: Google

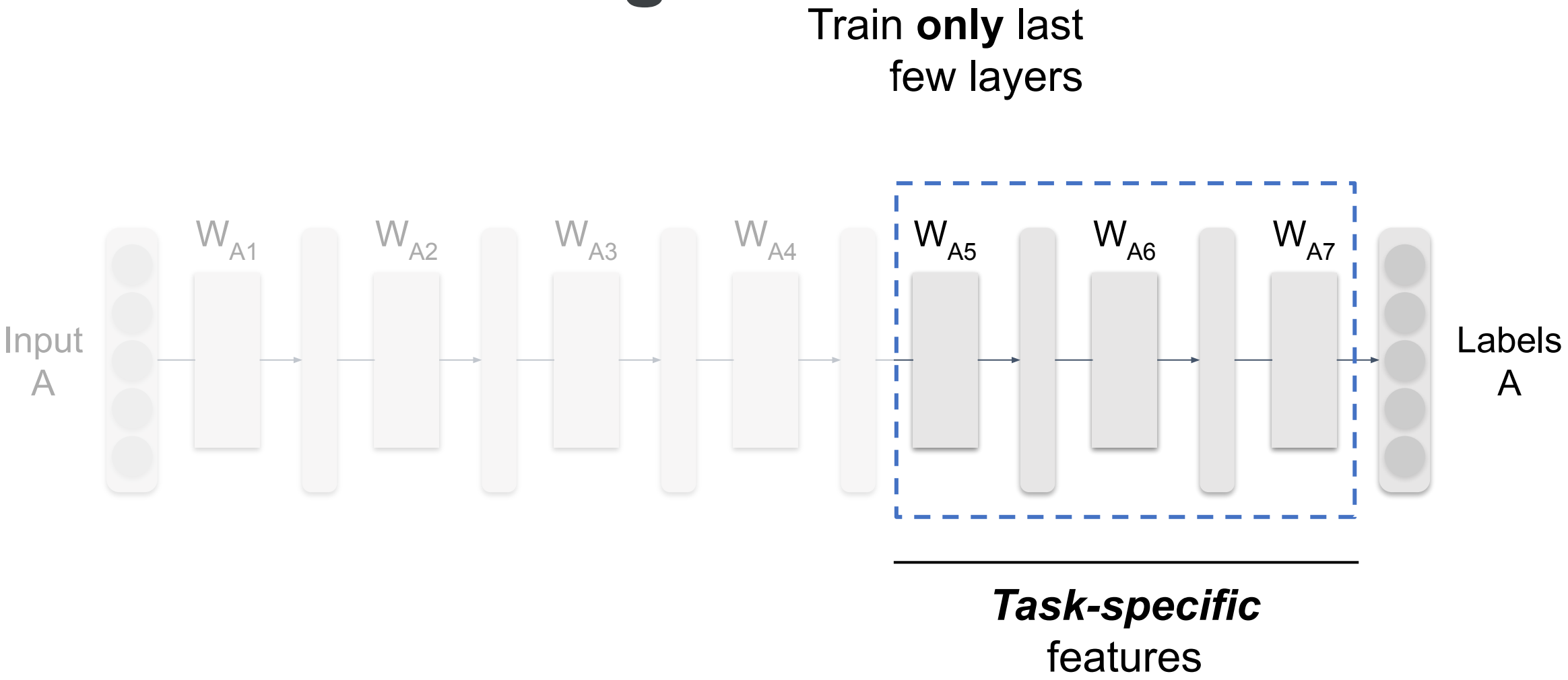
Transfer Learning

Reuse (freeze general feature extraction)

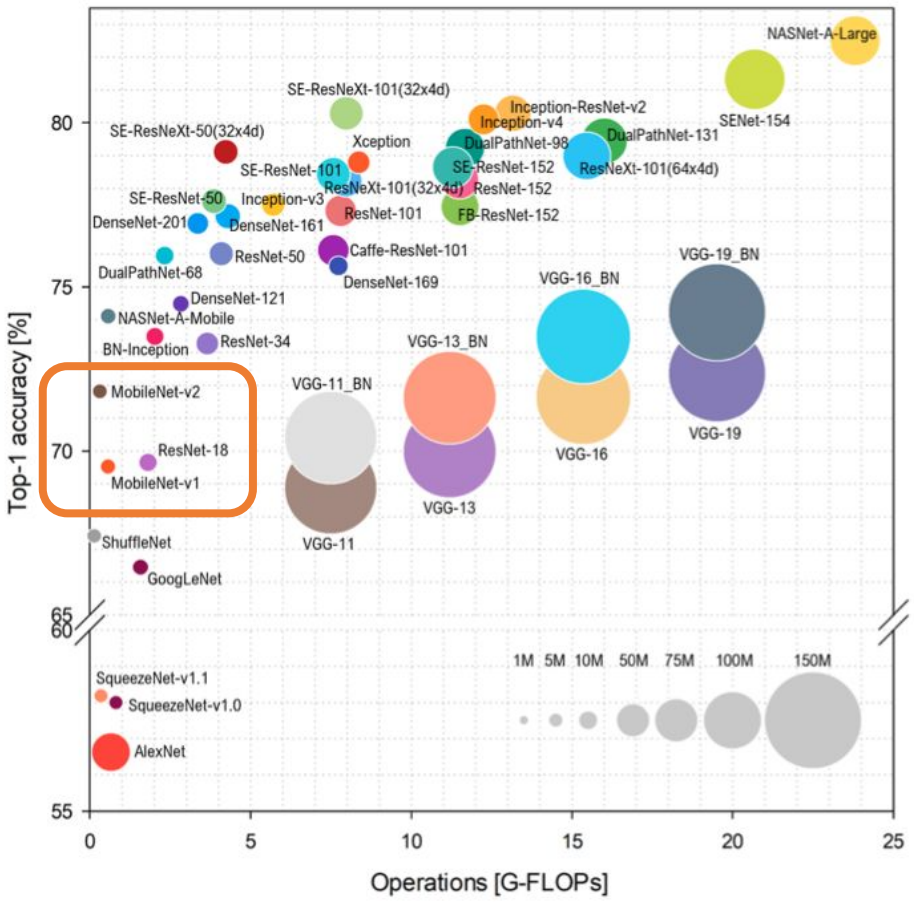


Learns ***general features*** irrespective of task

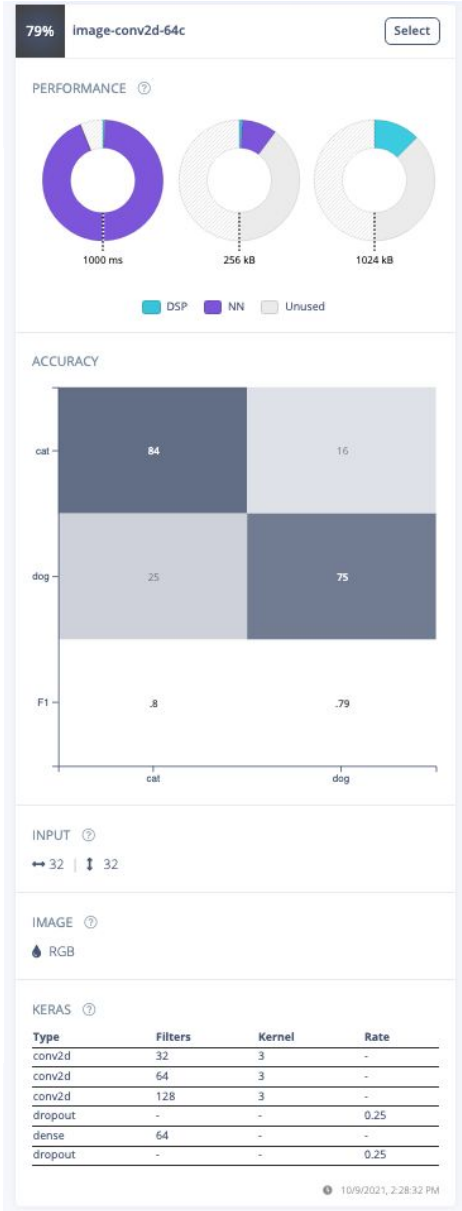
Transfer Learning

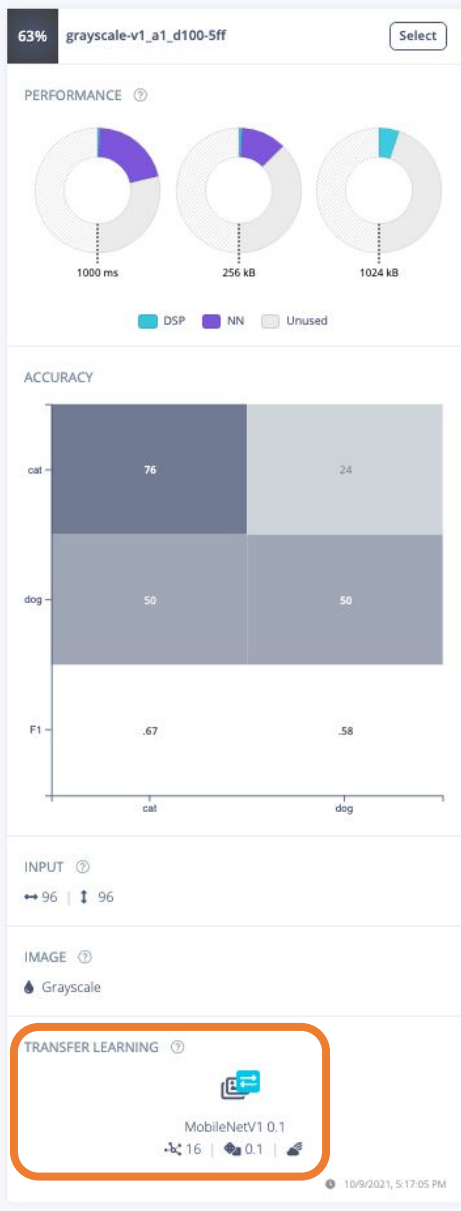


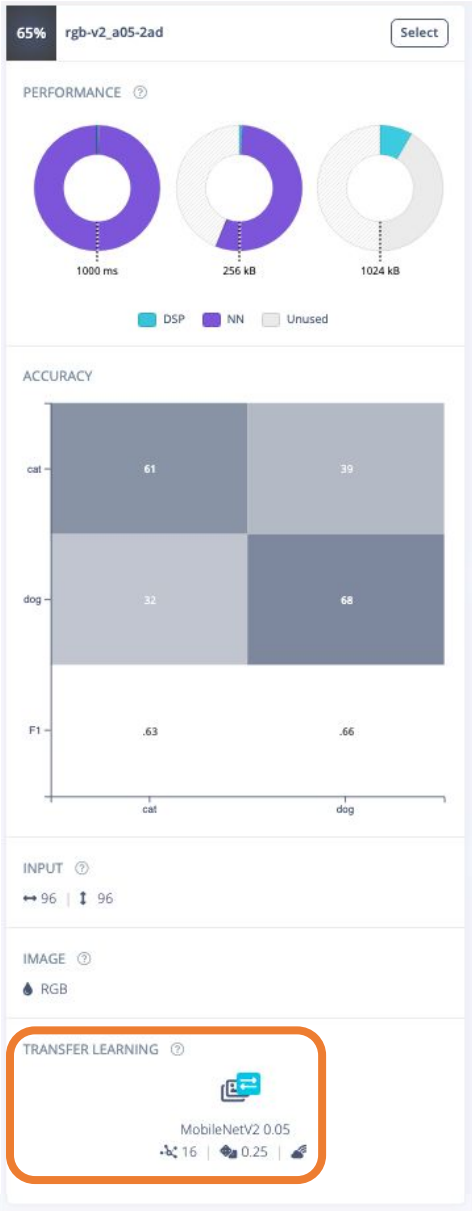
EON Tuner



https://arxiv.org/pdf/1810.00736.pdf



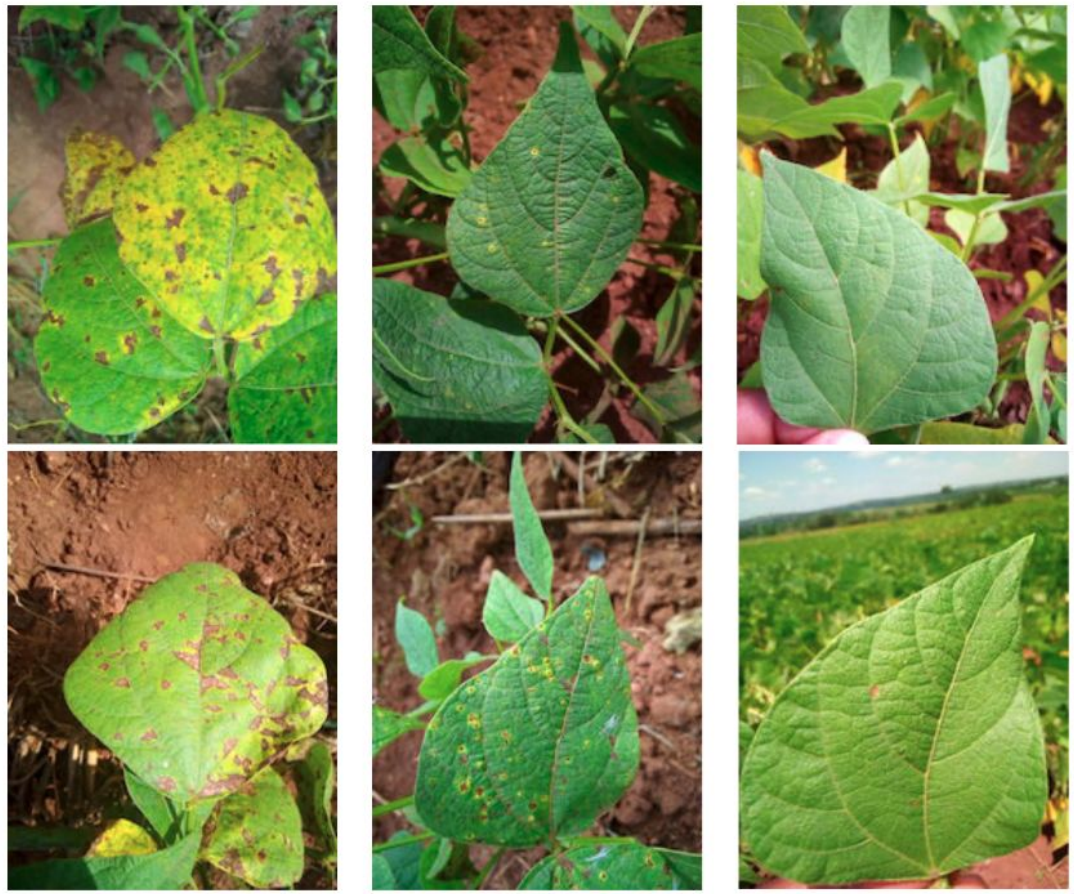




iBean

Detecting Diseases in the Bean plants

iBean Dataset



Angular Leaf Spot | Bean Rust | Healthy

This dataset is of leaf images taken in the field in different districts in Uganda by the Makerere AI lab in collaboration with the National Crops Resources Research Institute (NaCRRI), the national body in charge of research in agriculture in Uganda.

Goal:

To build a neural network that can tell the difference between the healthy and diseased leaves.

Dataset:

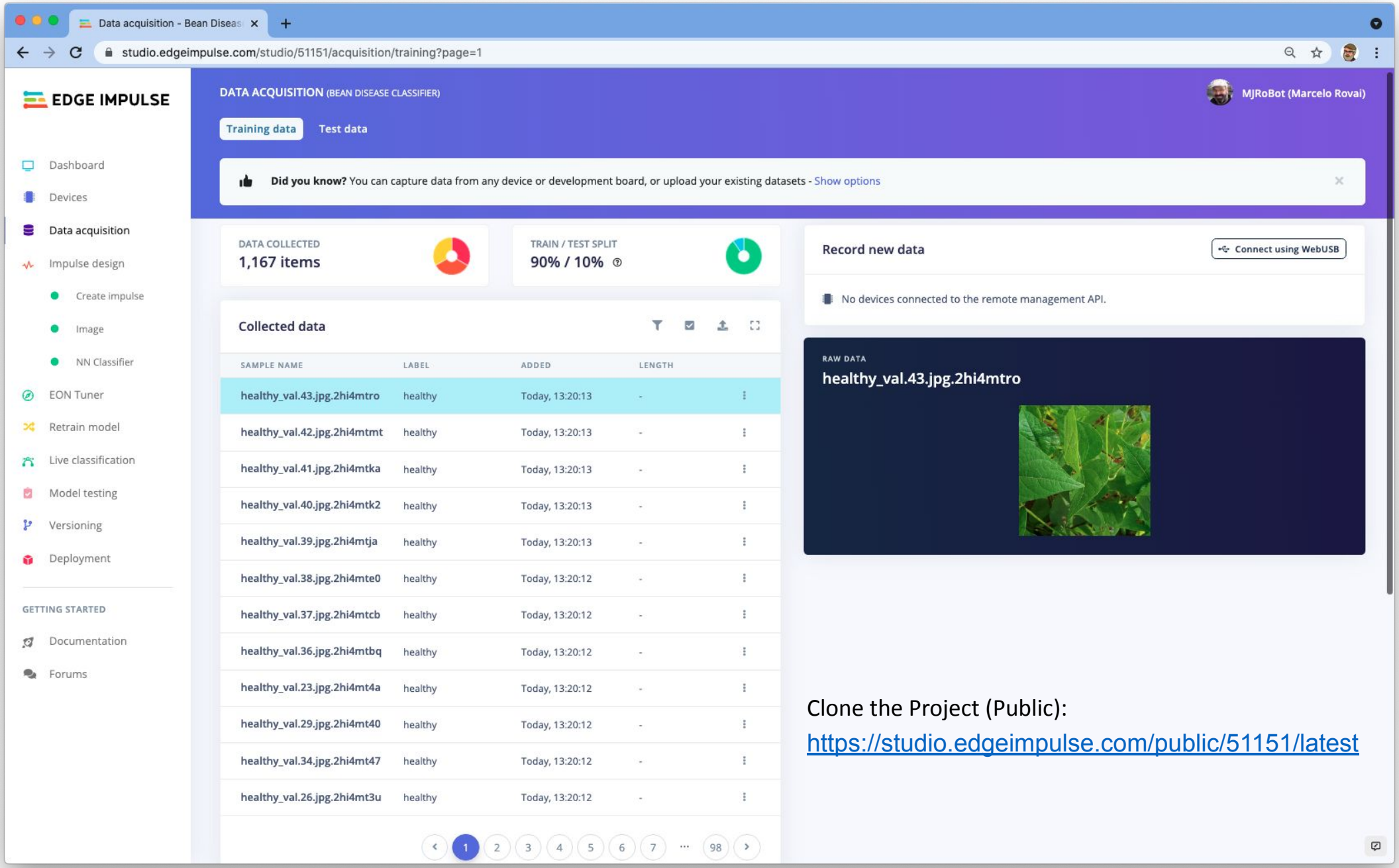
Training, Test and Validation data based on 224x224 pixel color images taken of bean plants in Uganda.

Class	Examples
Healthy class	428
Angular Leaf Spot	432
Bean Rust	436
Total:	1,296

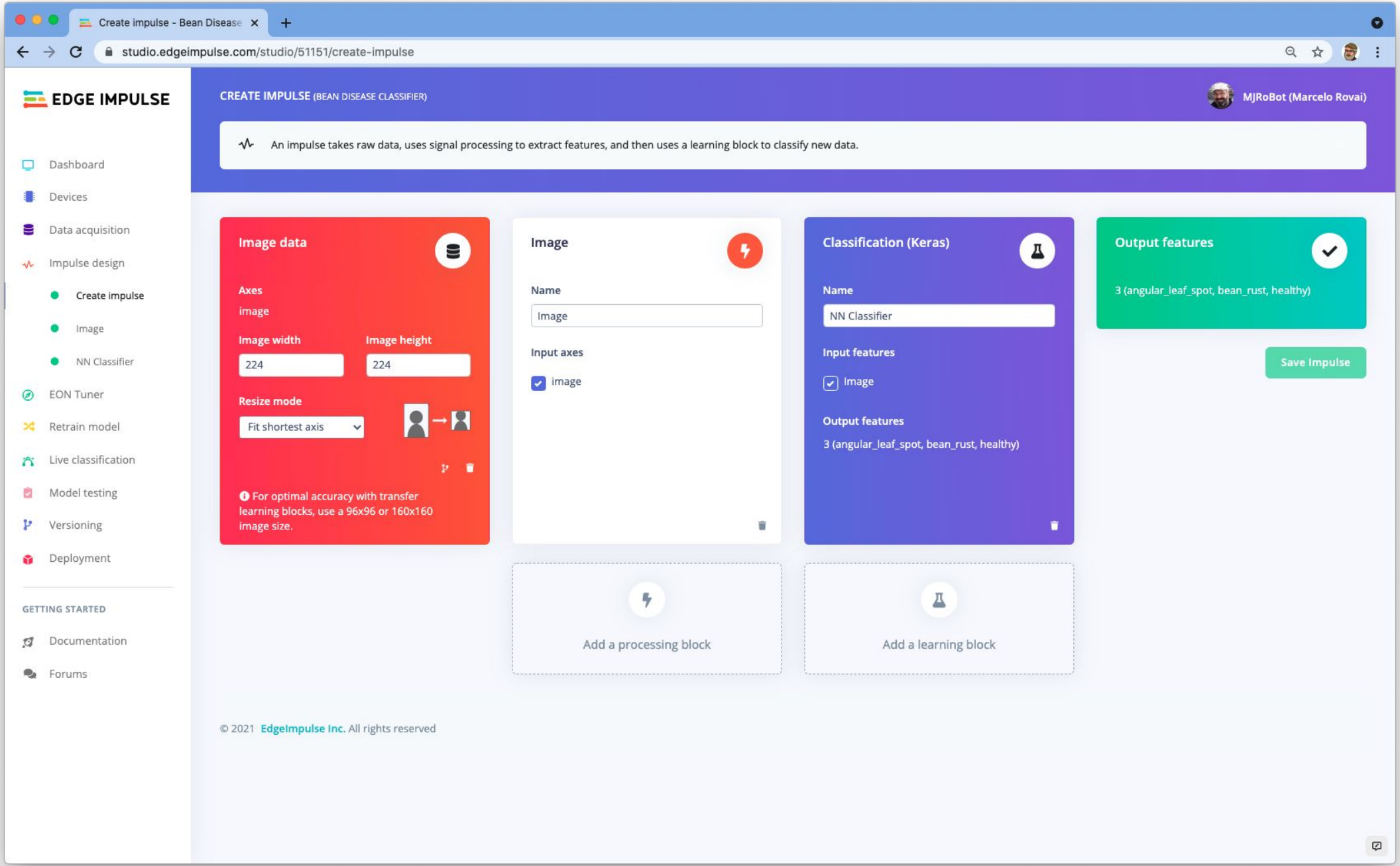
Problem!

Not a lot of data

Dataset: https://github.com/AI-Lab-Makerere/ibean/



Clone the Project (Public):
https://studio.edgeimpulse.com/public/51151/latest


Create impulse - Bean Disease
studio.edgeimpulse.com/studio/51151/create-impulse
EDGE IMPULSE
CREATE IMPULSE (BEAN DISEASE CLASSIFIER)
MJRoBot (Marcelo Rovai)
An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Image data
Axes
image
Image width
224
Image height
224
Resize mode
Fit shortest axis
For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.
Image
Name
Image
Input axes
image
Classification (Keras)
Name
NN Classifier
Input features
Image
Output features
3 (angular_leaf_spot, bean_rust, healthy)
Output features
3 (angular_leaf_spot, bean_rust, healthy)
Save Impulse
Add a processing block
Add a learning block
© 2021 EdgeImpulse Inc. All rights reserved

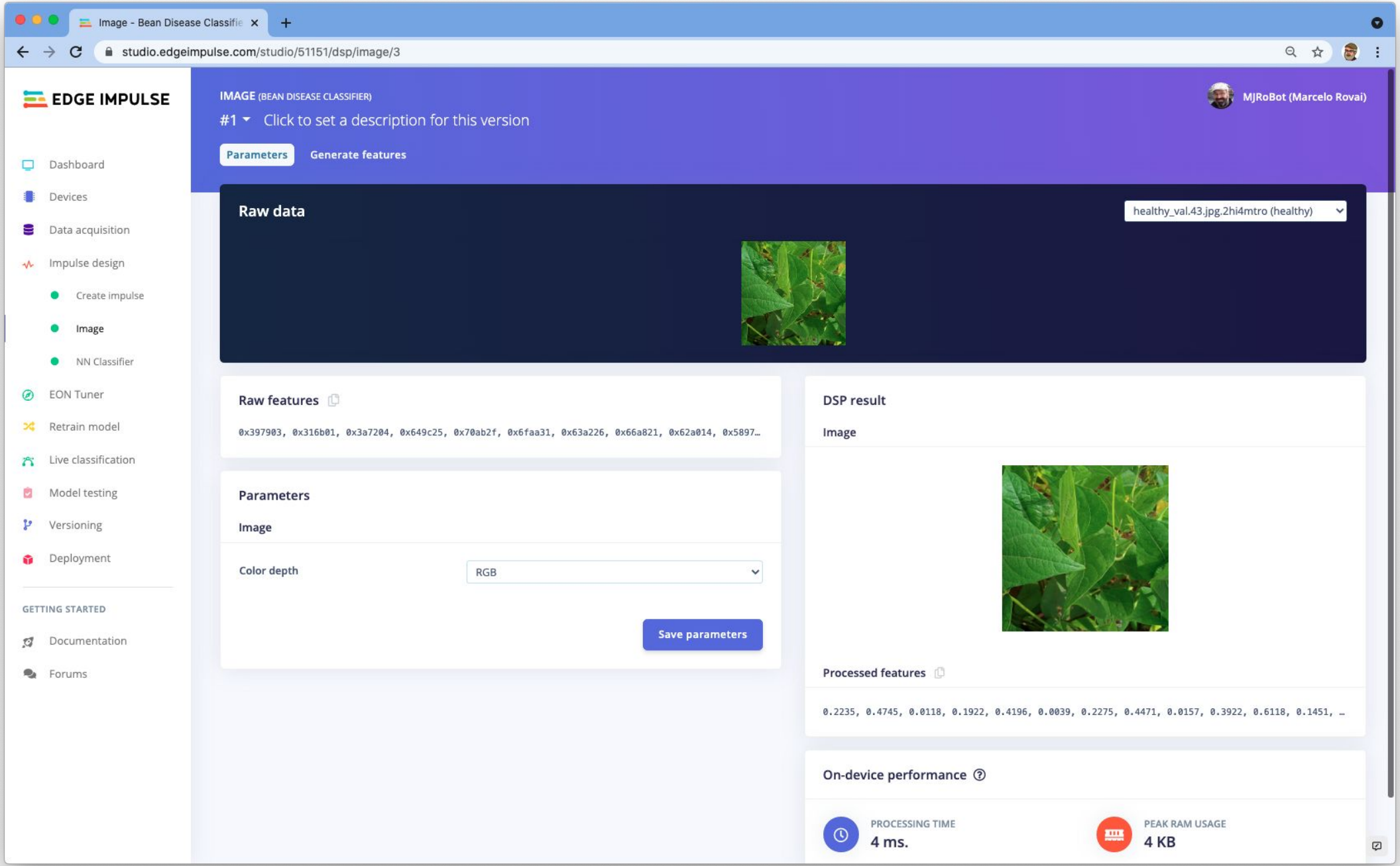


Image - Bean Disease Classifie
studio.edgeimpulse.com/studio/51151/dsp/image/3
EDGE IMPULSE
IMAGE (BEAN DISEASE CLASSIFIER)
#1 Click to set a description for this version
MJRoBot (Marcelo Rovai)
Parameters
Generate features
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Raw data
healthy_val.43.jpg.2hi4mtro (healthy)
Raw features
0x397903, 0x316b01, 0x3a7204, 0x649c25, 0x70ab2f, 0x6faa31, 0x63a226, 0x66a821, 0x62a014, 0x5897…
Parameters
Image
Color depth
RGB
Save parameters
DSP result
Image
Processed features
0.2235, 0.4745, 0.0118, 0.1922, 0.4196, 0.0039, 0.2275, 0.4471, 0.0157, 0.3922, 0.6118, 0.1451, …
On-device performance
PROCESSING TIME
4 ms.
PEAK RAM USAGE
4 KB

Image - Bean Disease Classifie

studio.edgeimpulse.com/studio/51151/dsp/image/3/generate-features#!

EDGE IMPULSE

#1 Click to set a description for this version

Parameters | Generate features

- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
  - Image
  - NN Classifier
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums

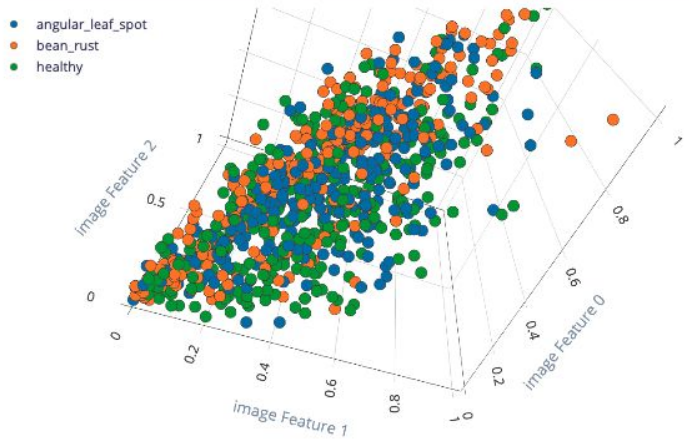
Training set

Data in training set	1,167 items
Classes	3 (angular_leaf_spot, bean_rust, healthy)

Generate features

Feature explorer (1,167 samples)

X Axis: image Feature 0 | Y Axis: image Feature 1 | Z Axis: image Feature 2

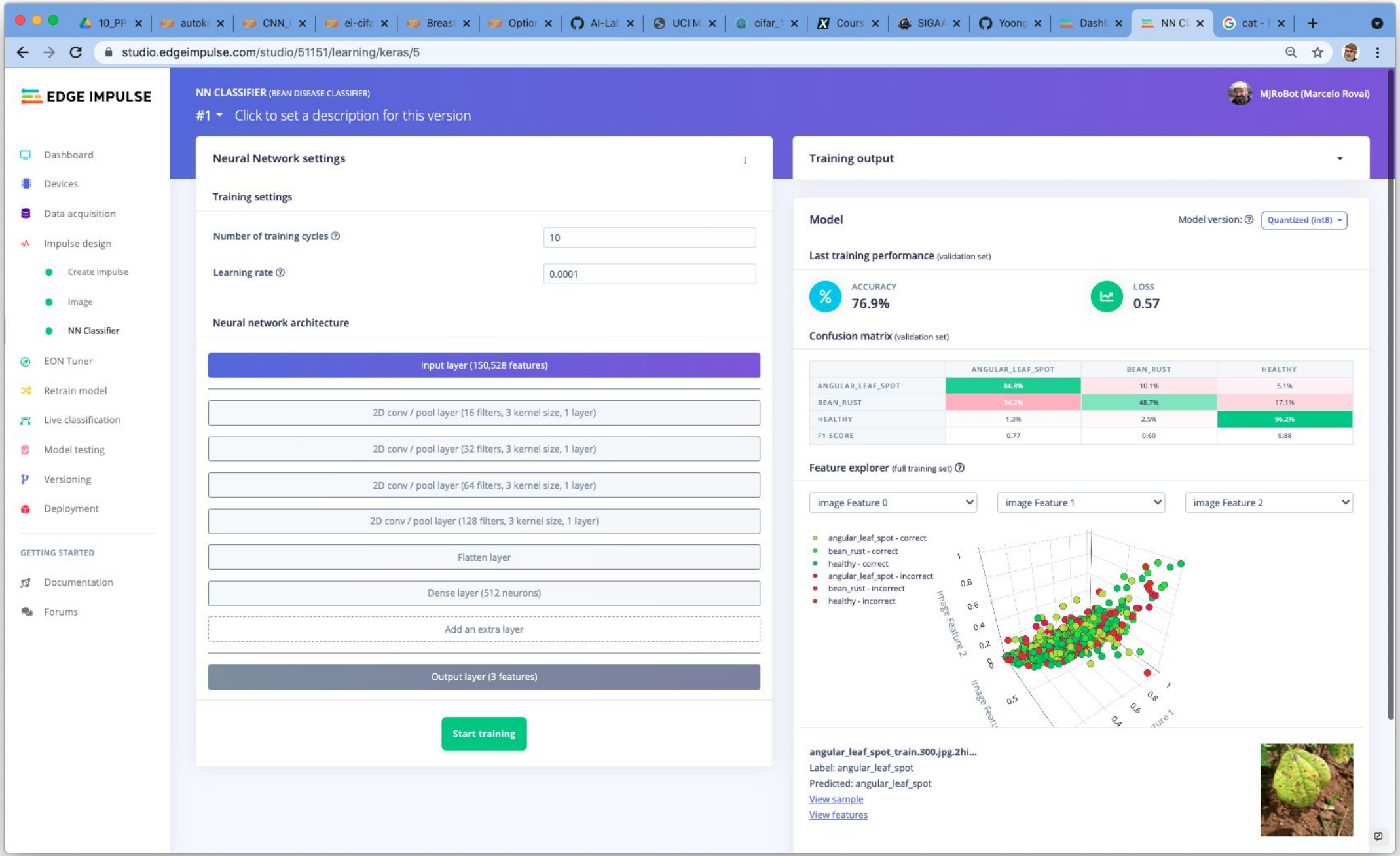


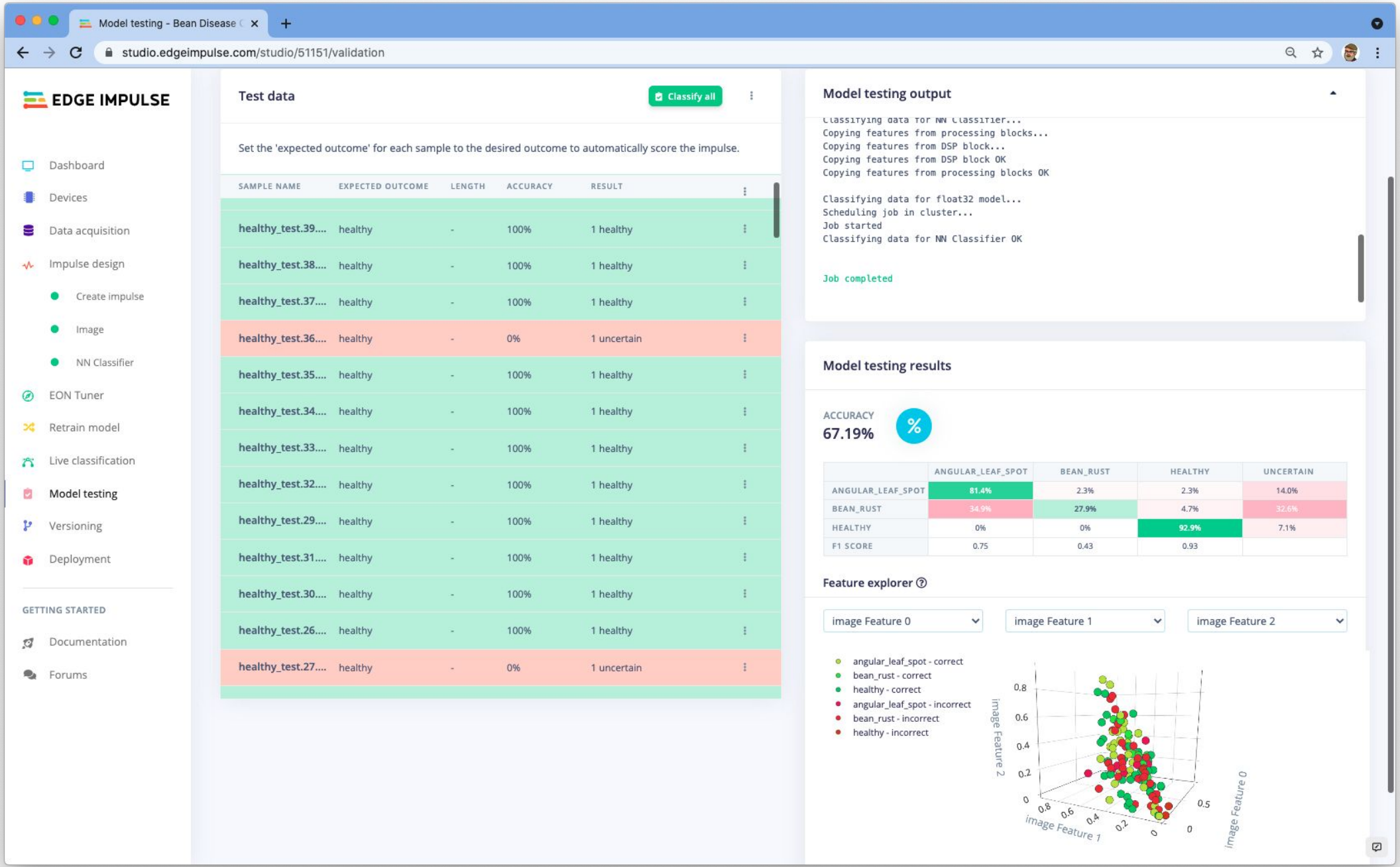
bean_rust_val.20.jpg.2hi4ikh7
Label: bean_rust
View sample
View features

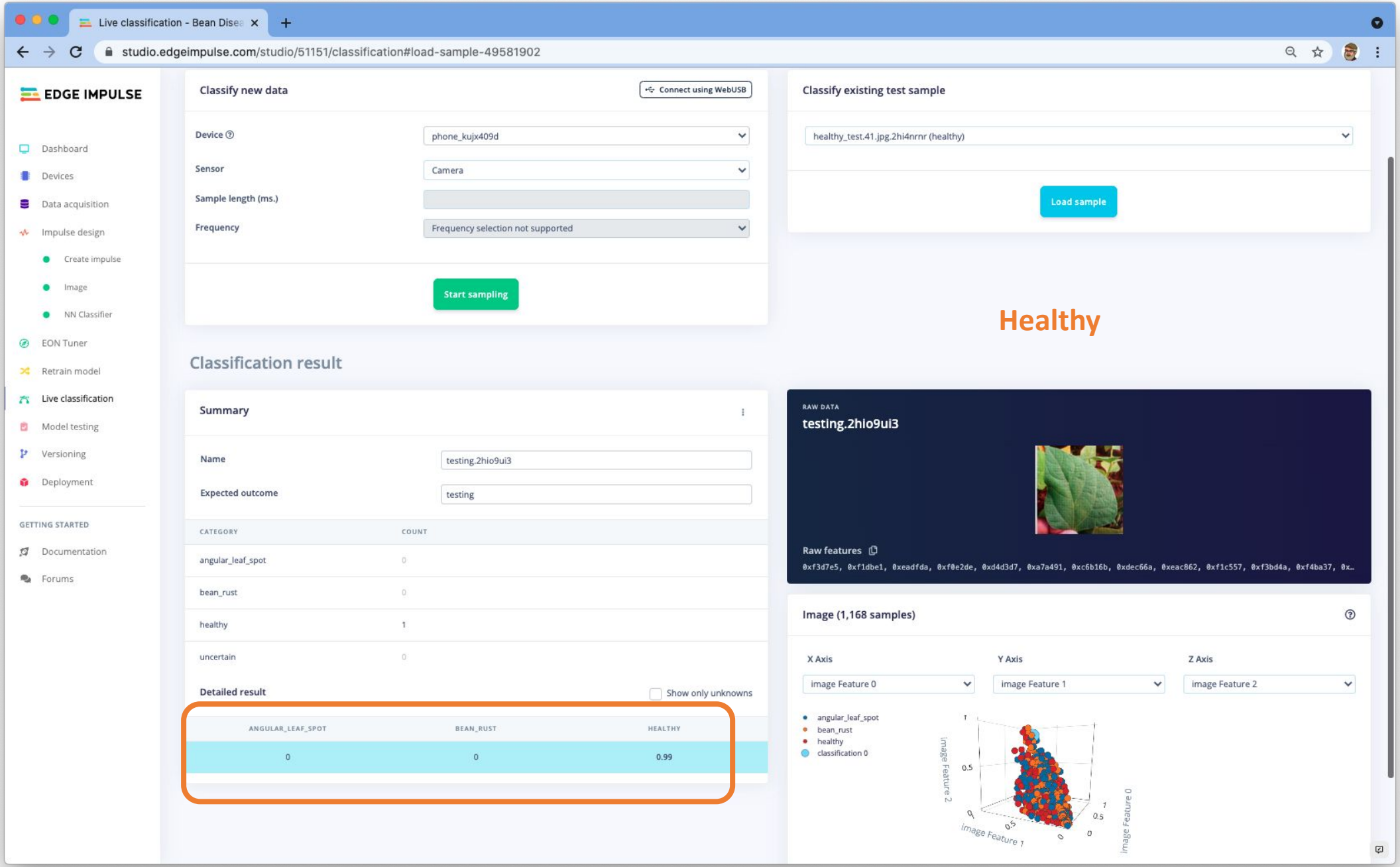
On-device performance

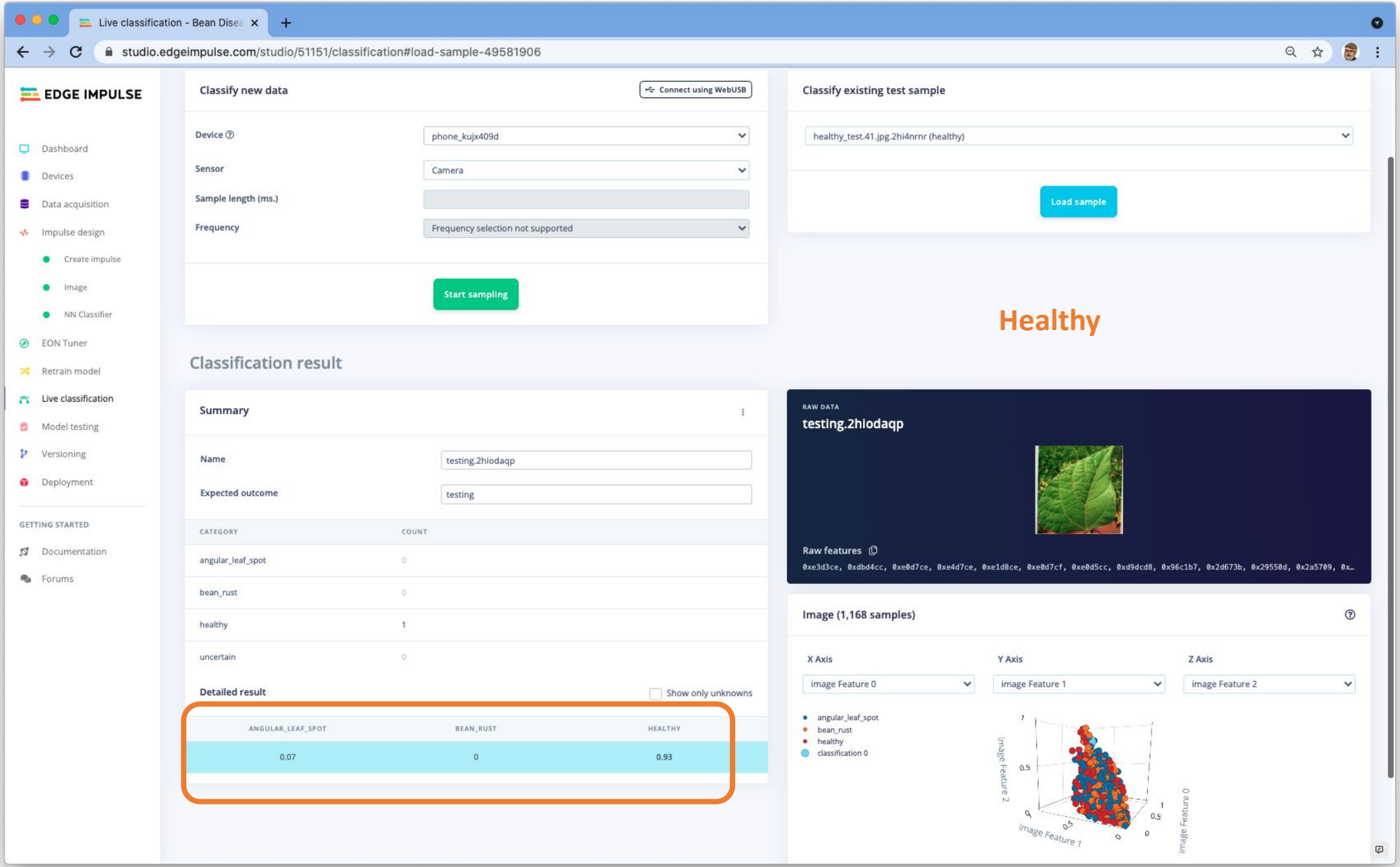
PROCESSING TIME
4 ms.

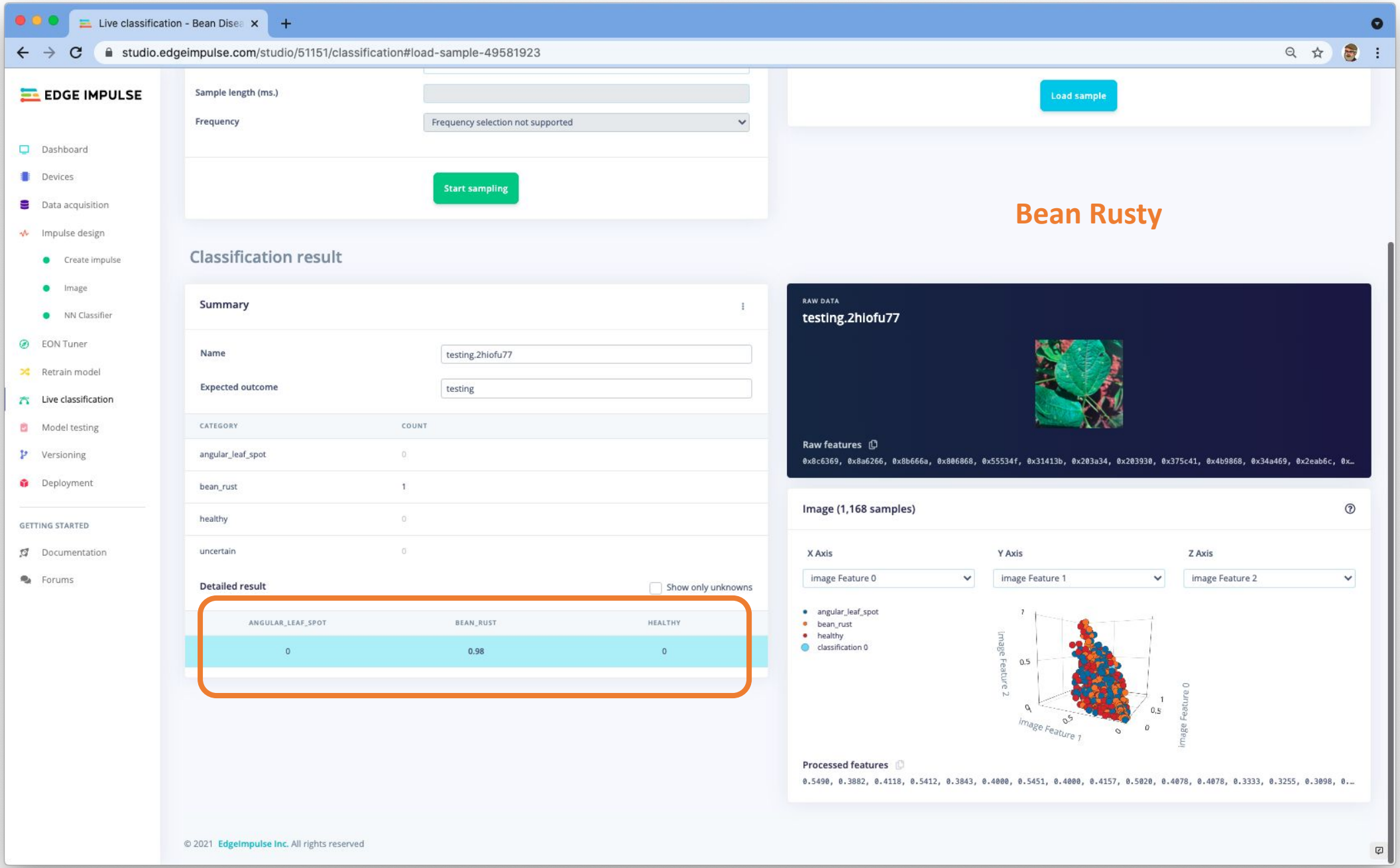
PEAK RAM USAGE
4 KB

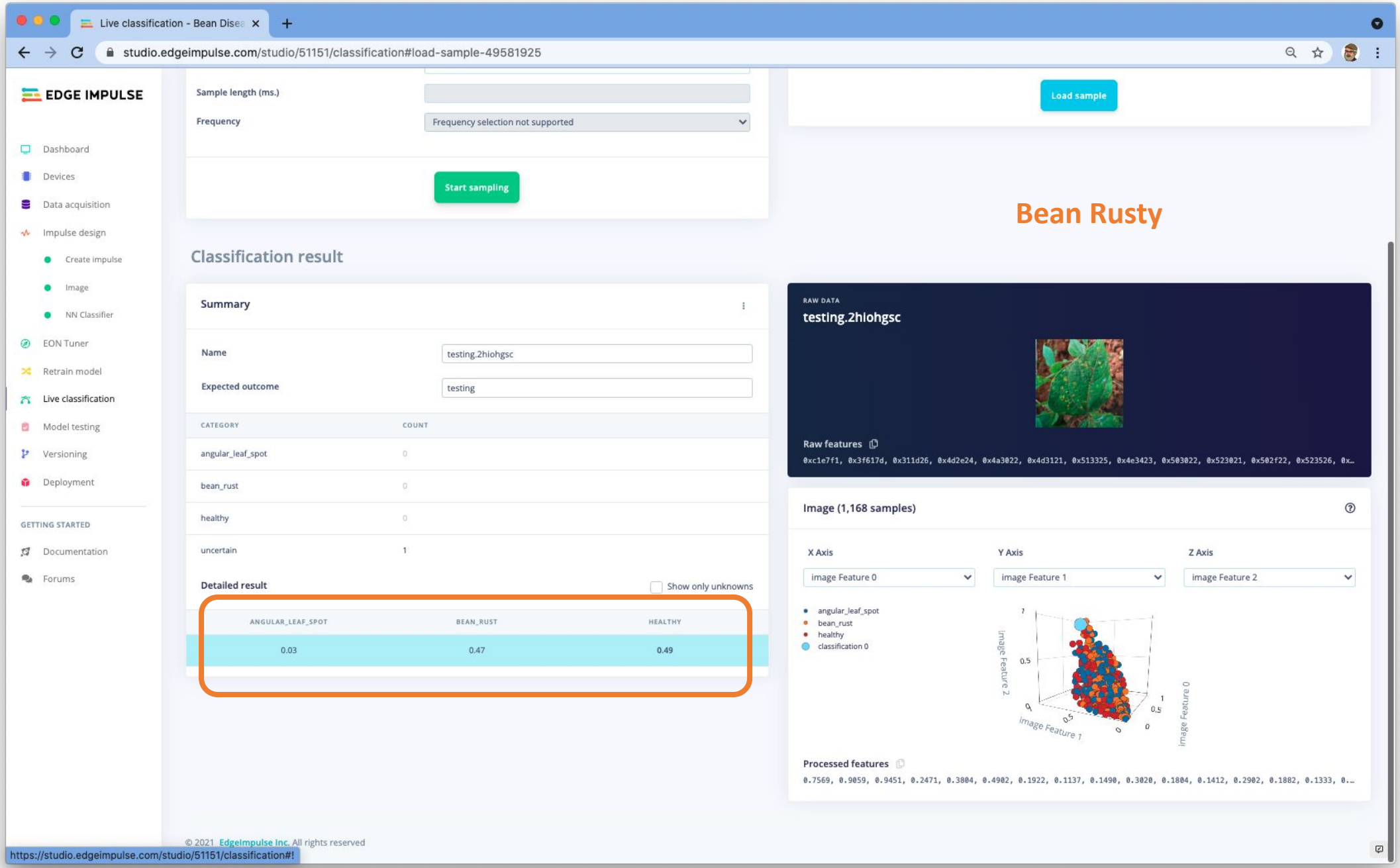

studio.edgeimpulse.com/studio/51151/learning/keras/5
EDGE IMPULSE
NN CLASSIFIER (BEAN DISEASE CLASSIFIER)
#1 Click to set a description for this version
MJRoBot (Marcelo Rovai)
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Neural Network settings
Training settings
Number of training cycles
10
Learning rate
0.0001
Neural network architecture
Input layer (150,528 features)
2D conv / pool layer (16 filters, 3 kernel size, 1 layer)
2D conv / pool layer (32 filters, 3 kernel size, 1 layer)
2D conv / pool layer (64 filters, 3 kernel size, 1 layer)
2D conv / pool layer (128 filters, 3 kernel size, 1 layer)
Flatten layer
Dense layer (512 neurons)
Add an extra layer
Output layer (3 features)
Start training
Training output
Model
Model version:
Quantized (int8)
Last training performance (validation set)
ACCURACY
76.9%
LOSS
0.57
Confusion matrix (validation set)
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
ANGULAR_LEAF_SPOT 84.8% 10.1% 5.1%
BEAN_RUST 34.2% 48.7% 17.1%
HEALTHY 1.3% 2.5% 96.2%
F1 SCORE 0.77 0.60 0.88
Feature explorer (full training set)
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot - correct
bean_rust - correct
healthy - correct
angular_leaf_spot - incorrect
bean_rust - incorrect
healthy - incorrect
angular_leaf_spot_train.300.jpg.2hi...
Label: angular_leaf_spot
Predicted: angular_leaf_spot
View sample
View features

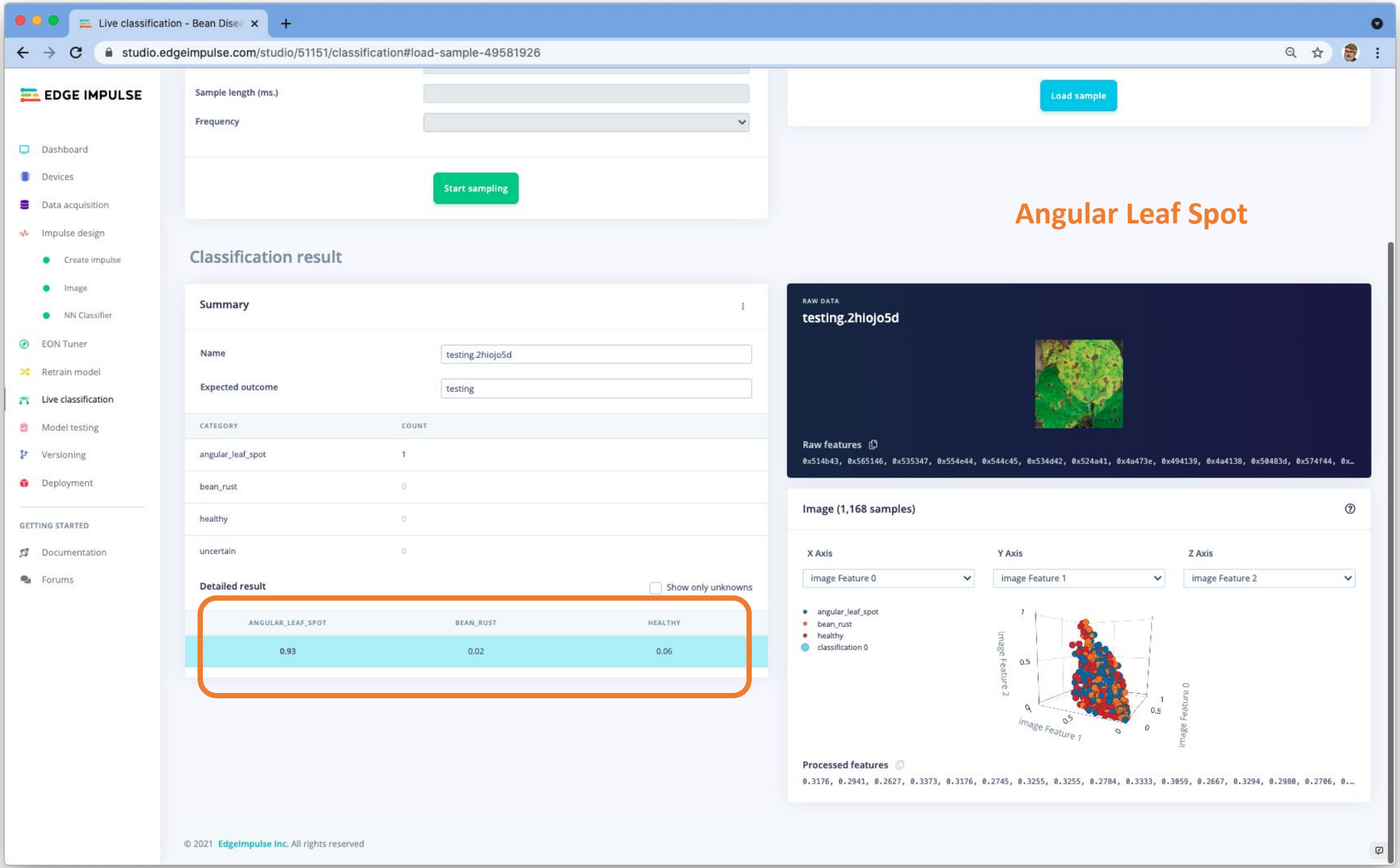

Model testing - Bean Disease
studio.edgeimpulse.com/studio/51151/validation
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Test data
Classify all
Set the 'expected outcome' for each sample to the desired outcome to automatically score the impulse.
SAMPLE NAME
EXPECTED OUTCOME
LENGTH
ACCURACY
RESULT
healthy_test.39.... healthy - 100% 1 healthy
healthy_test.38.... healthy - 100% 1 healthy
healthy_test.37.... healthy - 100% 1 healthy
healthy_test.36.... healthy - 0% 1 uncertain
healthy_test.35.... healthy - 100% 1 healthy
healthy_test.34.... healthy - 100% 1 healthy
healthy_test.33.... healthy - 100% 1 healthy
healthy_test.32.... healthy - 100% 1 healthy
healthy_test.29.... healthy - 100% 1 healthy
healthy_test.31.... healthy - 100% 1 healthy
healthy_test.30.... healthy - 100% 1 healthy
healthy_test.26.... healthy - 100% 1 healthy
healthy_test.27.... healthy - 0% 1 uncertain
Model testing output
Classifying data for NN Classifier...
Copying features from processing blocks...
Copying features from DSP block...
Copying features from DSP block OK
Copying features from processing blocks OK
Classifying data for float32 model...
Scheduling job in cluster...
Job started
Classifying data for NN Classifier OK
Job completed
Model testing results
ACCURACY
67.19%
ANGULAR_LEAF_SPOT BEAN_RUST HEALTHY UNCERTAIN
ANGULAR_LEAF_SPOT 81.4% 2.3% 2.3% 14.0%
BEAN_RUST 34.9% 27.9% 4.7% 32.6%
HEALTHY 0% 0% 92.9% 7.1%
F1 SCORE 0.75 0.43 0.93
Feature explorer
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot - correct
bean_rust - correct
healthy - correct
angular_leaf_spot - incorrect
bean_rust - incorrect
healthy - incorrect
image Feature 2
image Feature 1
image Feature 0

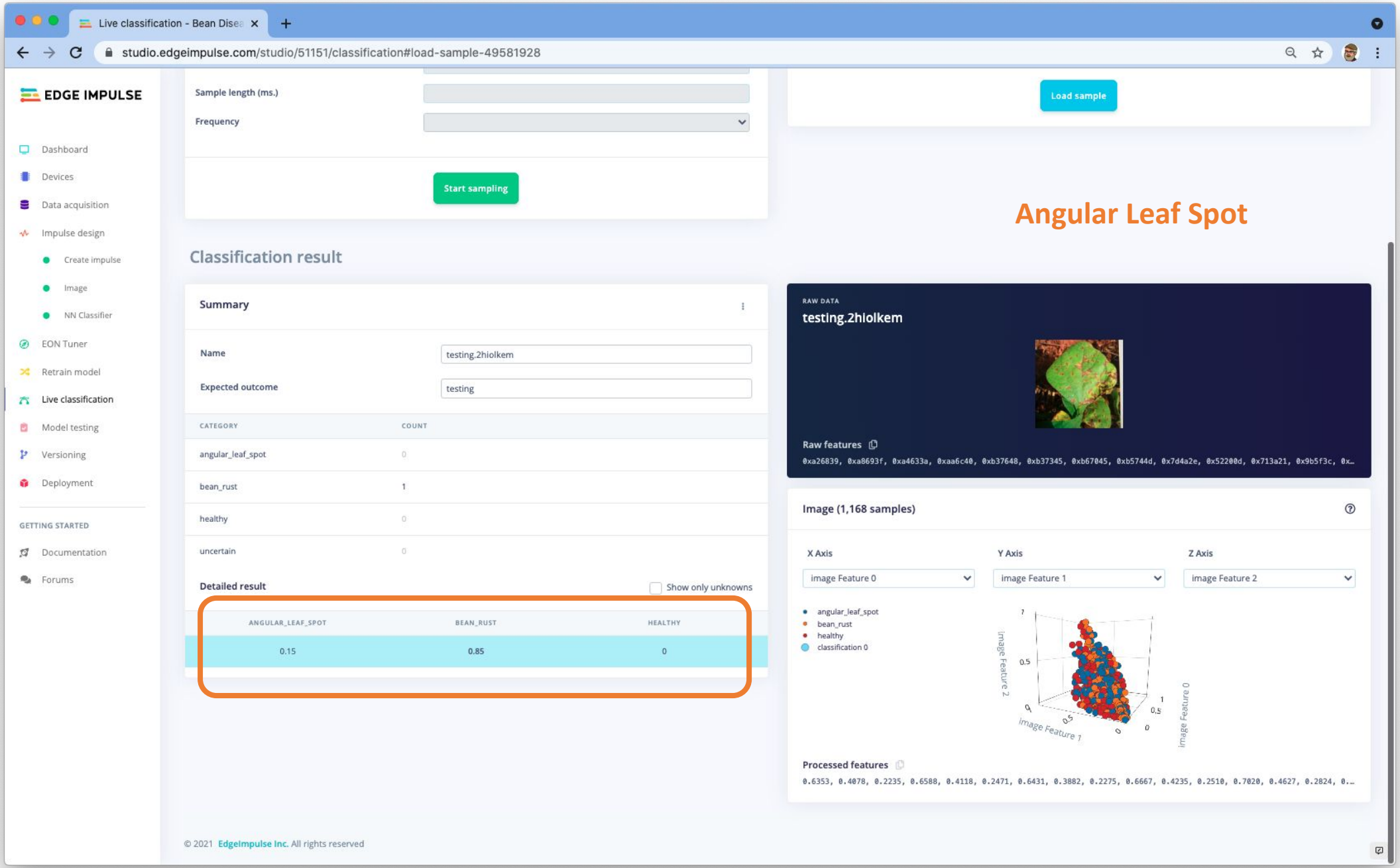

Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581902
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Classify new data
Connect using WebUSB
Device
phone_kujx409d
Sensor
Camera
Sample length (ms.)
Frequency
Frequency selection not supported
Start sampling
Classify existing test sample
healthy_test.41.jpg.2hi4nrnr (healthy)
Load sample
Healthy
Classification result
Summary
Name
testing.2hio9ui3
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot
0
bean_rust
0
healthy
1
uncertain
0
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0
0
0.99
RAW DATA
testing.2hio9ui3
Raw features
0xf3d7e5, 0xf1dbe1, 0xeadfda, 0xf0e2de, 0xd4d3d7, 0xa7a491, 0xc6b16b, 0xdec66a, 0xeac862, 0xf1c557, 0xf3bd4a, 0xf4ba37, 0x…
Image (1,168 samples)
X Axis
image Feature 0
Y Axis
image Feature 1
Z Axis
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
image Feature 2
image Feature 1
image Feature 0


Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581906
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Classify new data
Connect using WebUSB
Device
phone_kujx409d
Sensor
Camera
Sample length (ms.)
Frequency
Frequency selection not supported
Start sampling
Classify existing test sample
healthy_test.41.jpg.2hi4nrnr (healthy)
Load sample
Healthy
Classification result
Summary
Name
testing.2hiodaqp
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot 0
bean_rust 0
healthy 1
uncertain 0
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0.07
0
0.93
RAW DATA
testing.2hiodaqp
Raw features
0xe3d3ce, 0xdbd4cc, 0xe0d7ce, 0xe4d7ce, 0xe1d8ce, 0xe0d7cf, 0xe0d5cc, 0xd9dcd8, 0x96c1b7, 0x2d673b, 0x29550d, 0x2a5709, 0x…
Image (1,168 samples)
X Axis
image Feature 0
Y Axis
image Feature 1
Z Axis
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
image Feature 2
image Feature 1
image Feature 0

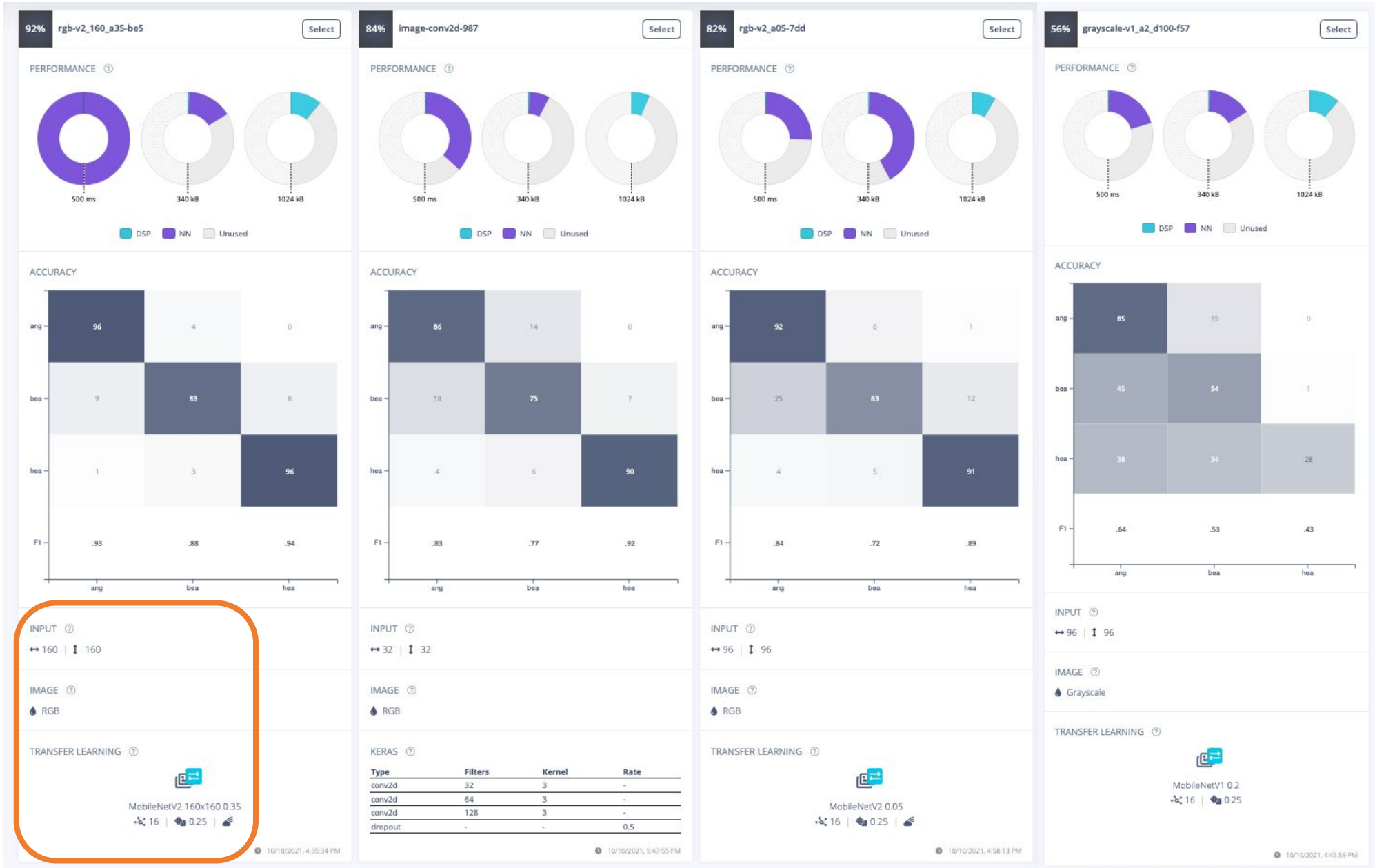

Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581923
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Sample length (ms.)
Frequency
Frequency selection not supported
Start sampling
Load sample
Bean Rusty
Classification result
Summary
Name
testing.2hiofu77
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot 0
bean_rust 1
healthy 0
uncertain 0
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0
0.98
0
RAW DATA
testing.2hiofu77
Raw features
0x8c6369, 0x8a6266, 0x8b666a, 0x806868, 0x55534f, 0x31413b, 0x203a34, 0x203930, 0x375c41, 0x4b9868, 0x34a469, 0x2eab6c, 0x…
Image (1,168 samples)
X Axis
Y Axis
Z Axis
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
Processed features
0.5490, 0.3882, 0.4118, 0.5412, 0.3843, 0.4000, 0.5451, 0.4000, 0.4157, 0.5020, 0.4078, 0.4078, 0.3333, 0.3255, 0.3098, 0.…
© 2021 EdgeImpulse Inc. All rights reserved

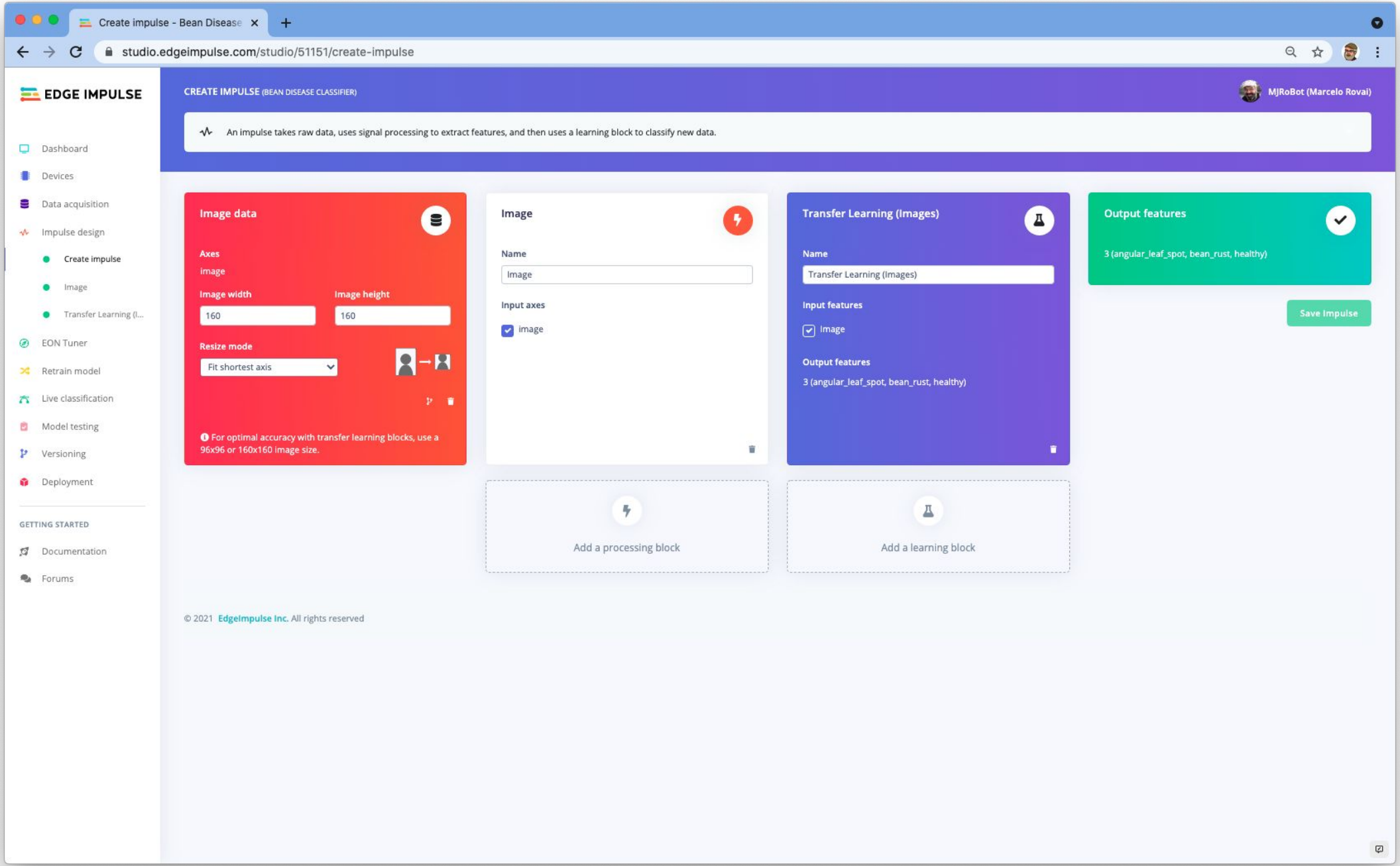

Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581925
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Sample length (ms.)
Frequency
Frequency selection not supported
Start sampling
Load sample
Bean Rusty
Classification result
Summary
Name
testing.2hiohgsc
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot
bean_rust
healthy
uncertain
1
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0.03
0.47
0.49
RAW DATA
testing.2hiohgsc
Raw features
0xc1e7f1, 0x3f617d, 0x311d26, 0x4d2e24, 0x4a3022, 0x4d3121, 0x513325, 0x4e3423, 0x503022, 0x523021, 0x502f22, 0x523526, 0x…
Image (1,168 samples)
X Axis
Y Axis
Z Axis
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
Processed features
0.7569, 0.9059, 0.9451, 0.2471, 0.3804, 0.4902, 0.1922, 0.1137, 0.1490, 0.3020, 0.1804, 0.1412, 0.2902, 0.1882, 0.1333, 0.…
© 2021 EdgeImpulse Inc. All rights reserved
https://studio.edgeimpulse.com/studio/51151/classification#!

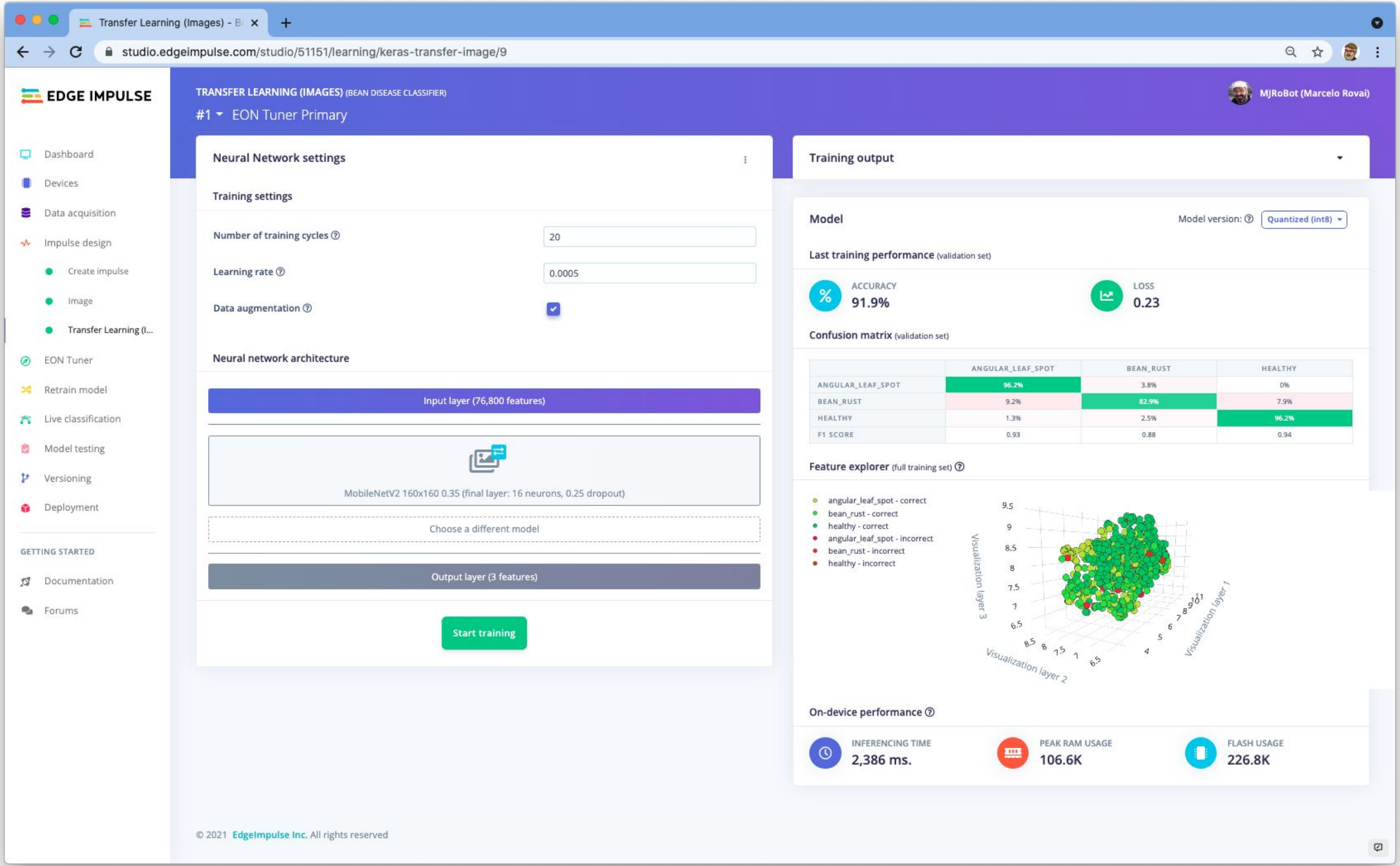

Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581926
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Sample length (ms.)
Frequency
Start sampling
Load sample
Angular Leaf Spot
Classification result
Summary
Name
testing.2hiojo5d
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot 1
bean_rust 0
healthy 0
uncertain 0
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0.93
0.02
0.06
RAW DATA
testing.2hiojo5d
Raw features
0x514b43, 0x565146, 0x535347, 0x554e44, 0x544c45, 0x534d42, 0x524a41, 0x4a473e, 0x494139, 0x4a4138, 0x50483d, 0x574f44, 0x…
Image (1,168 samples)
X Axis
Y Axis
Z Axis
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
Processed features
0.3176, 0.2941, 0.2627, 0.3373, 0.3176, 0.2745, 0.3255, 0.3255, 0.2784, 0.3333, 0.3059, 0.2667, 0.3294, 0.2980, 0.2706, 0.…
© 2021 EdgeImpulse Inc. All rights reserved


Live classification - Bean Dise
studio.edgeimpulse.com/studio/51151/classification#load-sample-49581928
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
NN Classifier
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Sample length (ms.)
Frequency
Start sampling
Load sample
Angular Leaf Spot
Classification result
Summary
Name
testing.2hiolkem
Expected outcome
testing
CATEGORY
COUNT
angular_leaf_spot
0
bean_rust
1
healthy
0
uncertain
0
Detailed result
Show only unknowns
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
0.15
0.85
0
RAW DATA
testing.2hiolkem
Raw features
0xa26839, 0xa8693f, 0xa4633a, 0xaa6c40, 0xb37648, 0xb37345, 0xb67045, 0xb5744d, 0x7d4a2e, 0x52200d, 0x713a21, 0x9b5f3c, 0x…
Image (1,168 samples)
X Axis
Y Axis
Z Axis
image Feature 0
image Feature 1
image Feature 2
angular_leaf_spot
bean_rust
healthy
classification 0
Processed features
0.6353, 0.4078, 0.2235, 0.6588, 0.4118, 0.2471, 0.6431, 0.3882, 0.2275, 0.6667, 0.4235, 0.2510, 0.7020, 0.4627, 0.2824, 0.…
© 2021 EdgeImpulse Inc. All rights reserved

EON Tuner


92% rgb-v2_160_a35-be5 Select
PERFORMANCE
500 ms 340 kB 1024 kB
DSP NN Unused
ACCURACY
ang 96 4 0
bea 9 83 8
hea 1 3 96
F1 .93 .88 .94
ang bea hea
INPUT
160 | 160
IMAGE
RGB
TRANSFER LEARNING
MobileNetV2 160x160 0.35
16 | 0.25
10/10/2021, 4:35:34 PM
84% image-conv2d-987 Select
PERFORMANCE
500 ms 340 kB 1024 kB
DSP NN Unused
ACCURACY
ang 86 14 0
bea 18 75 7
hea 4 6 90
F1 .83 .77 .92
ang bea hea
INPUT
32 | 32
IMAGE
RGB
KERAS
Type Filters Kernel Rate
conv2d 32 3 -
conv2d 64 3 -
conv2d 128 3 -
dropout - - 0.5
10/10/2021, 5:47:55 PM
82% rgb-v2_a05-7dd Select
PERFORMANCE
500 ms 340 kB 1024 kB
DSP NN Unused
ACCURACY
ang 92 6 1
bea 25 63 12
hea 4 5 91
F1 .84 .72 .89
ang bea hea
INPUT
96 | 96
IMAGE
RGB
TRANSFER LEARNING
MobileNetV2 0.05
16 | 0.25
10/10/2021, 4:58:13 PM
56% grayscale-v1_a2_d100-f57 Select
PERFORMANCE
500 ms 340 kB 1024 kB
DSP NN Unused
ACCURACY
ang 85 15 0
bea 45 54 1
hea 38 34 28
F1 .64 .53 .43
ang bea hea
INPUT
96 | 96
IMAGE
Grayscale
TRANSFER LEARNING
MobileNetV1 0.2
16 | 0.25
10/10/2021, 4:45:59 PM


Create impulse - Bean Disease
studio.edgeimpulse.com/studio/51151/create-impulse
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer Learning (I...
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
CREATE IMPULSE (BEAN DISEASE CLASSIFIER)
MJRoBot (Marcelo Rovai)
An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.
Image data
Axes
image
Image width
160
Image height
160
Resize mode
Fit 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 (Images)
Input features
Image
Output features
3 (angular_leaf_spot, bean_rust, healthy)
Output features
3 (angular_leaf_spot, bean_rust, healthy)
Save Impulse
Add a processing block
Add a learning block
© 2021 EdgeImpulse Inc. All rights reserved


Transfer Learning (Images) - B
studio.edgeimpulse.com/studio/51151/learning/keras-transfer-image/9
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer Learning (I...
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
TRANSFER LEARNING (IMAGES) (BEAN DISEASE CLASSIFIER)
#1 EON Tuner Primary
MJRoBot (Marcelo Rovai)
Neural Network settings
Training settings
Number of training cycles
20
Learning rate
0.0005
Data augmentation
Neural network architecture
Input layer (76,800 features)
MobileNetV2 160x160 0.35 (final layer: 16 neurons, 0.25 dropout)
Choose a different model
Output layer (3 features)
Start training
Training output
Model
Model version:
Quantized (int8)
Last training performance (validation set)
ACCURACY
91.9%
LOSS
0.23
Confusion matrix (validation set)
ANGULAR_LEAF_SPOT
BEAN_RUST
HEALTHY
ANGULAR_LEAF_SPOT 96.2% 3.8% 0%
BEAN_RUST 9.2% 82.9% 7.9%
HEALTHY 1.3% 2.5% 96.2%
F1 SCORE 0.93 0.88 0.94
Feature explorer (full training set)
angular_leaf_spot - correct
bean_rust - correct
healthy - correct
angular_leaf_spot - incorrect
bean_rust - incorrect
healthy - incorrect
Visualization layer 1
Visualization layer 2
Visualization layer 3
On-device performance
INFERENCING TIME
2,386 ms.
PEAK RAM USAGE
106.6K
FLASH USAGE
226.8K
© 2021 EdgeImpulse Inc. All rights reserved

Model testing - Bean Disease

studio.edgeimpulse.com/studio/51151/validation

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
- Impulse design
  - Create impulse
  - Image
  - Transfer Learning (I...
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums

MODEL TESTING (BEAN DISEASE CLASSIFIER)

MJRoBot (Marcelo Rovai)

This lists all test data. You can manage this data through Data acquisition.

Test data

Classify all

Set the 'expected outcome' for each sample to the desired outcome to automatically score the impulse.

SAMPLE NAME	EXPECTED OUTCOME	LENGTH	ACCURACY	RESULT
testing.2hiolkem	testing	-		
testing.2hiojo5d	testing	-		
testing.2hiohgsc	testing	-		
testing.2hiofu77	testing	-		
testing.2hiodaqp	testing	-		
testing.2hio9ui3	testing	-		
healthy_test.41.jpg...	healthy	-	100%	1 healthy
healthy_test.40.jpg...	healthy	-	100%	1 healthy
healthy_test.39.jpg...	healthy	-	100%	1 healthy
healthy_test.38.jpg...	healthy	-	100%	1 healthy
healthy_test.37.jpg...	healthy	-	100%	1 healthy
healthy_test.36.jpg...	healthy	-	100%	1 healthy
healthy_test.35.jpg...	healthy	-	100%	1 healthy
healthy_test.34.jpg...	healthy	-	100%	1 healthy
healthy_test.33.jpg...	healthy	-	100%	1 healthy
healthy_test.32.jpg...	healthy	-	100%	1 healthy

Model testing output

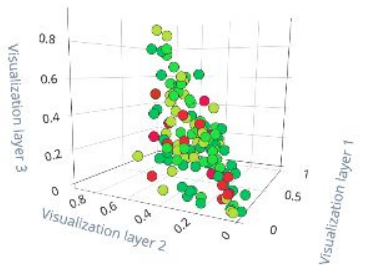
Model testing results

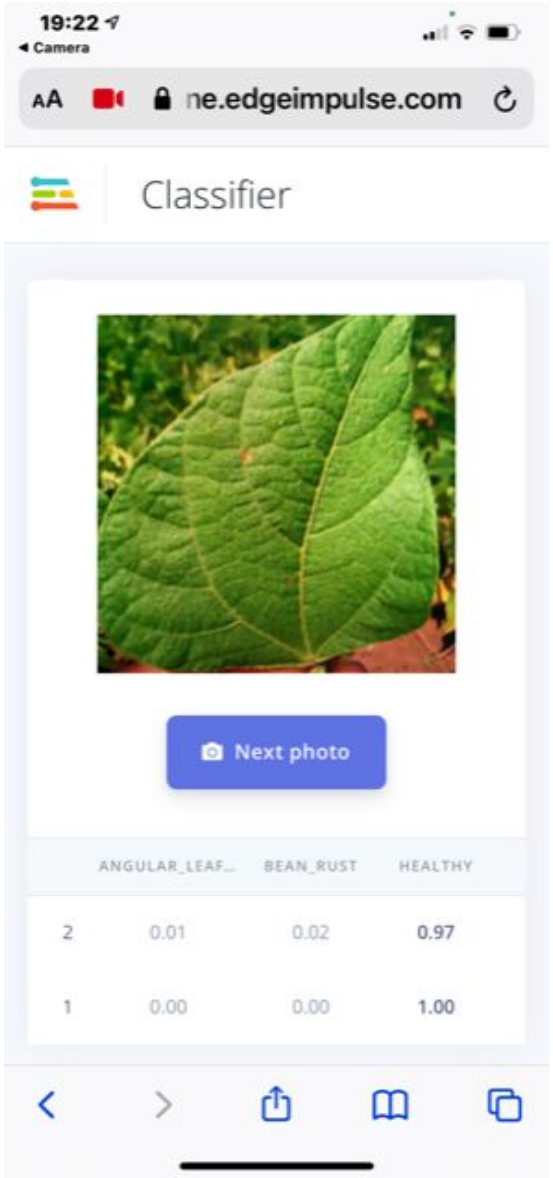
ACCURACY
84.38%

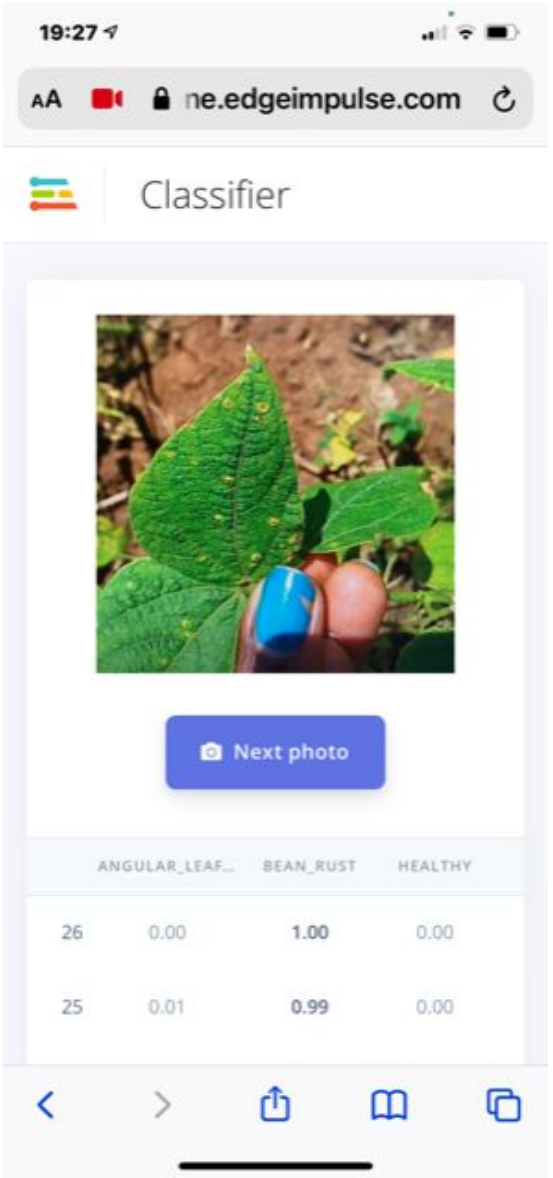
	ANGULAR_LEAF_SPOT	BEAN_RUST	HEALTHY	UNCERTAIN
ANGULAR_LEAF_SPOT	79.1%	14.0%	0%	7.0%
BEAN_RUST	7.0%	76.7%	2.3%	14.0%
HEALTHY	0%	2.4%	97.6%	0%
F1 SCORE	0.85	0.80	0.98	

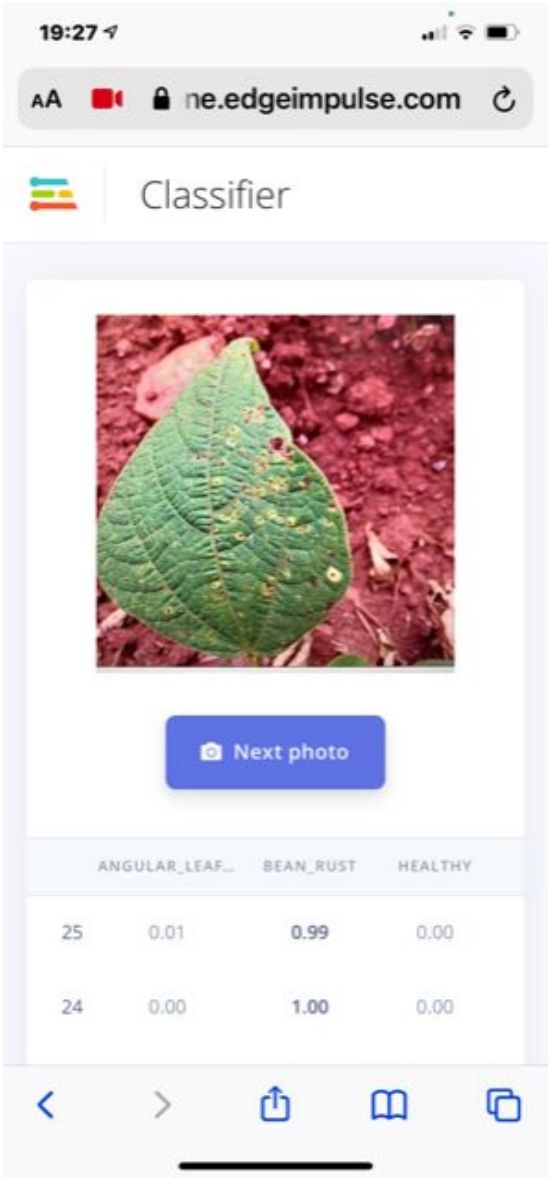
Feature explorer

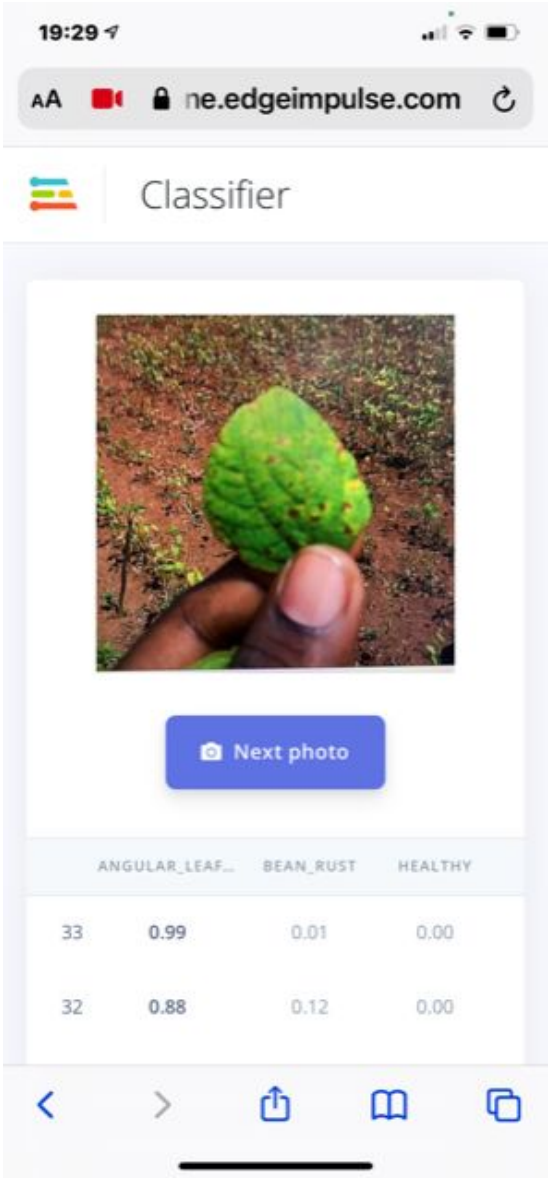
- angular_leaf_spot - correct
- bean_rust - correct
- healthy - correct
- angular_leaf_spot - incorrect
- bean_rust - incorrect
- healthy - incorrect




19:22
Camera
AA
ne.edgeimpulse.com
Classifier
Next photo
ANGULAR_LEAF...
BEAN_RUST
HEALTHY
2
0.01
0.02
0.97
1
0.00
0.00
1.00


19:27
AA
ne.edgeimpulse.com
Classifier
Next photo
ANGULAR_LEAF...
BEAN_RUST
HEALTHY
26
0.00
1.00
0.00
25
0.01
0.99
0.00

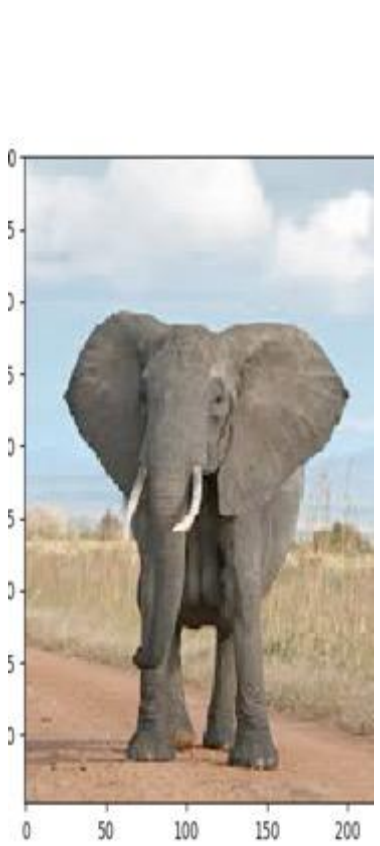

19:27
AA
ne.edgeimpulse.com
Classifier
Next photo
ANGULAR_LEAF...
BEAN_RUST
HEALTHY
25
0.01
0.99
0.00
24
0.00
1.00
0.00

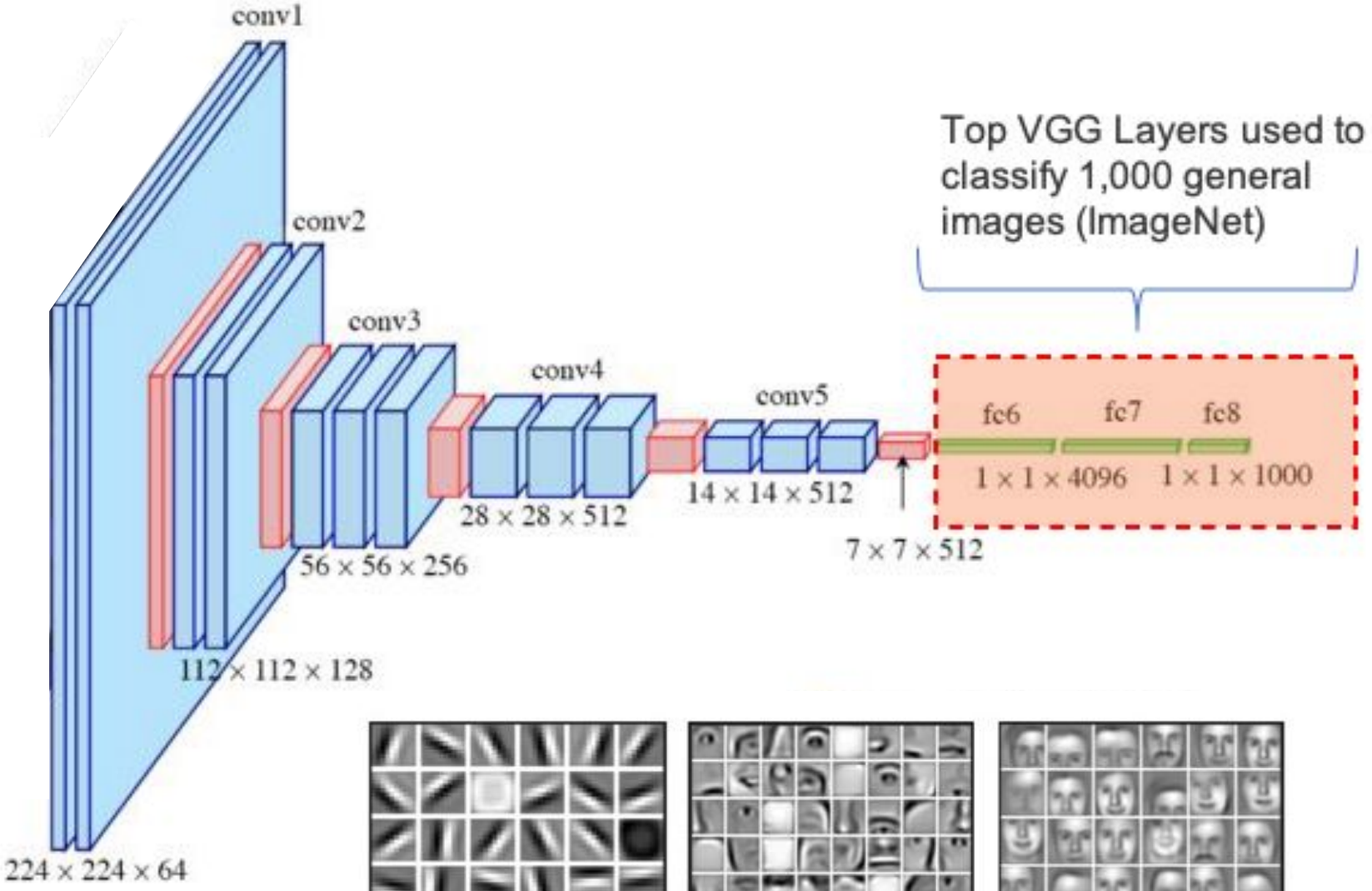

19:29
AA
ne.edgeimpulse.com
Classifier
Next photo
ANGULAR_LEAF...
BEAN_RUST
HEALTHY
33
0.99
0.01
0.00
32
0.88
0.12
0.00

covidXray

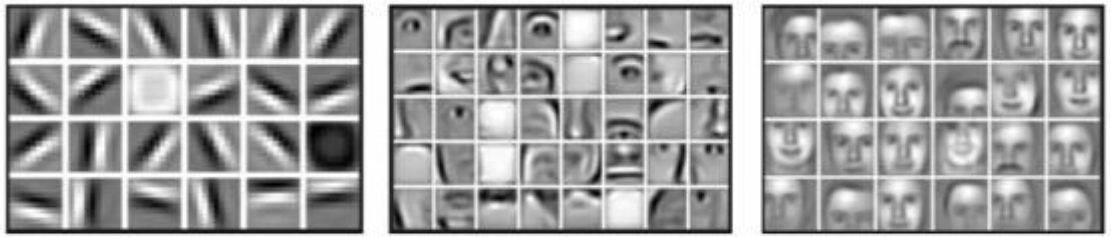
Detecting Covid-19 in Chest X-Ray images

VGG-16 Convolutional Neural Network Model



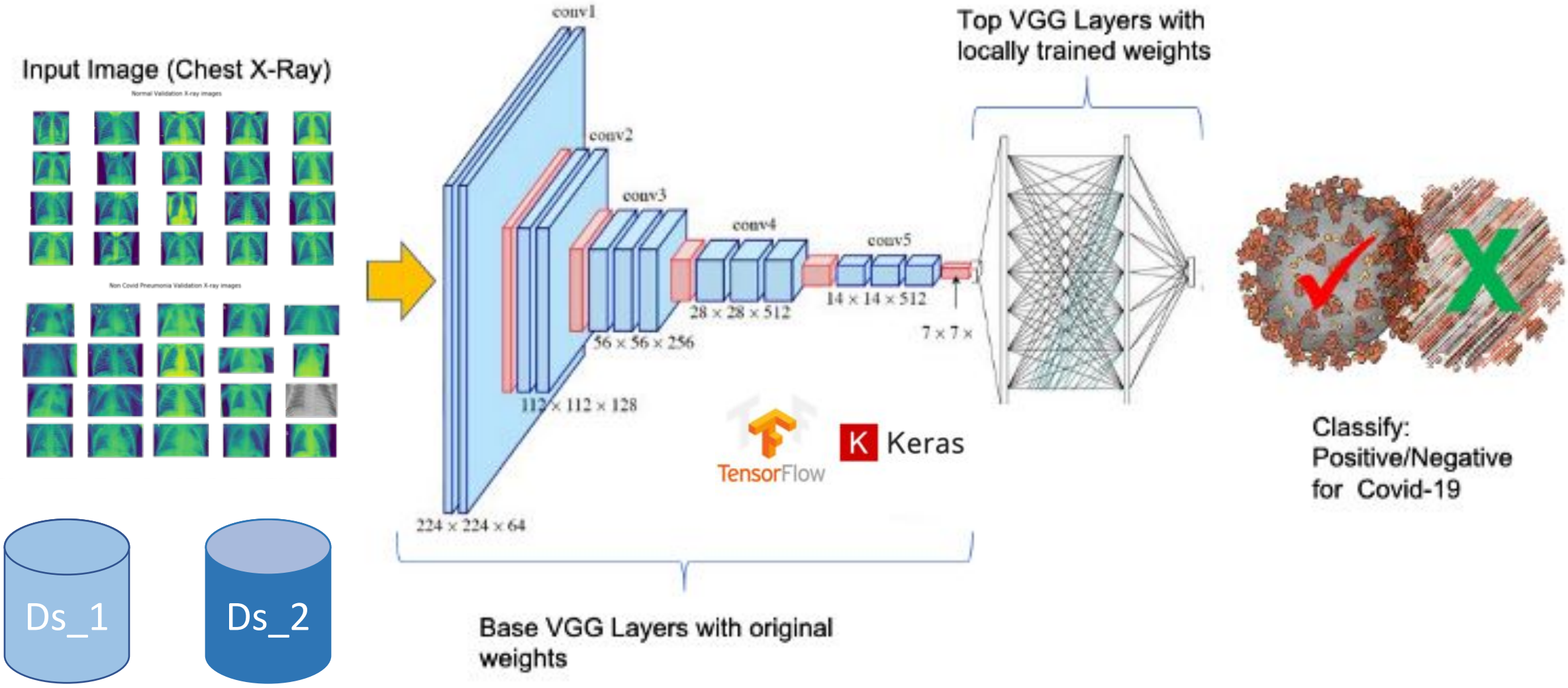


Elephant
93.0%

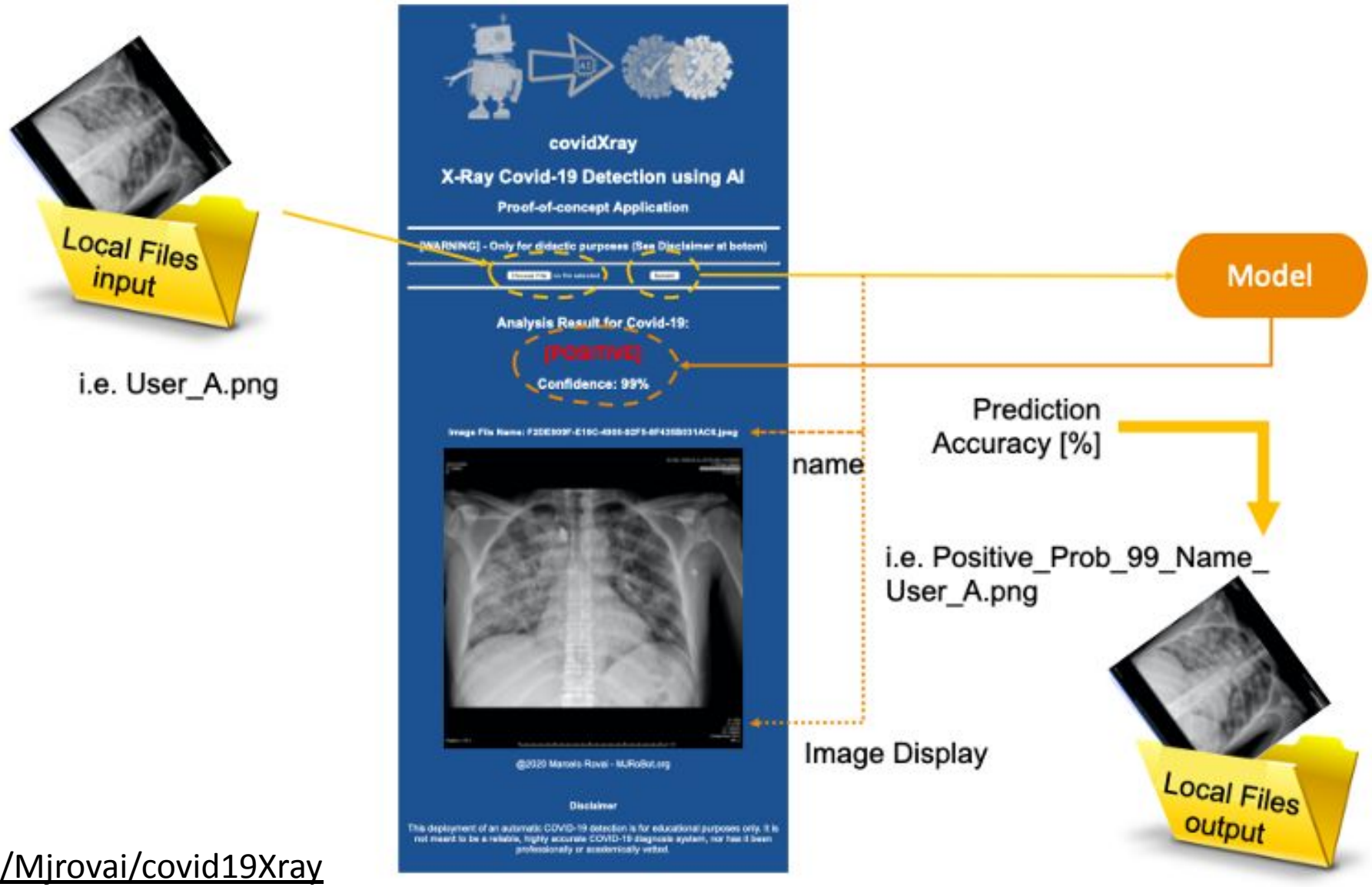


First Layer Representation | Second Layer Representation | Third Layer Representation

Training the model (Transfer Learning)



Inference



https://github.com/Mjrovai/covid19Xray

Preventing Overfitting

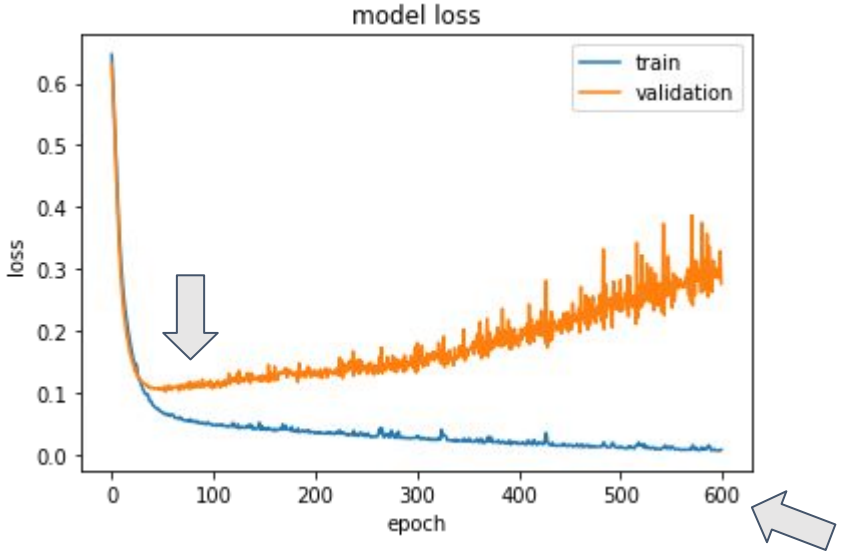
Early Stopping & Dropout Regularization

Early Stopping

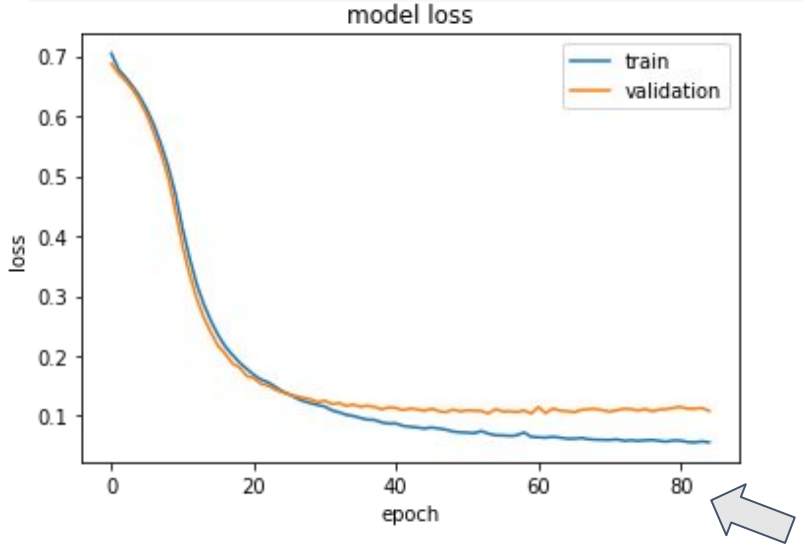
```
from tensorflow.keras.callbacks import EarlyStopping

early_stop = EarlyStopping(monitor='val_loss',
                           mode='min',
                           verbose=1,
                           patience=25)
```

```
history = model.fit(X_train,
                    y_train,
                    epochs=600,
                    validation_data=(X_test, y_test),
                    verbose=1
                    )
```

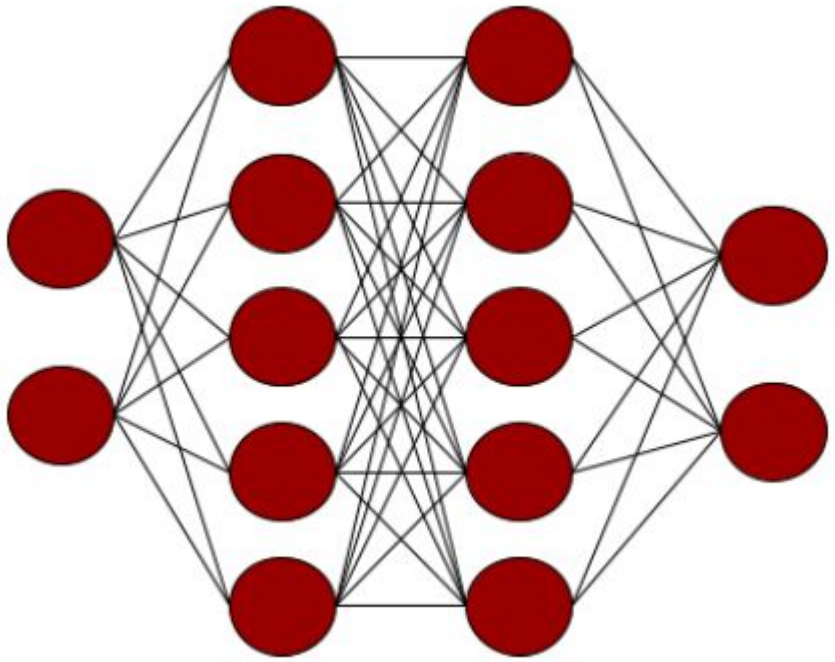



```
history = model.fit(x=X_train,
                    y=y_train,
                    epochs=600,
                    validation_data=(X_test, y_test),
                    verbose=1,
                    callbacks=[early_stop]
                    )
```



Dropout Regularization

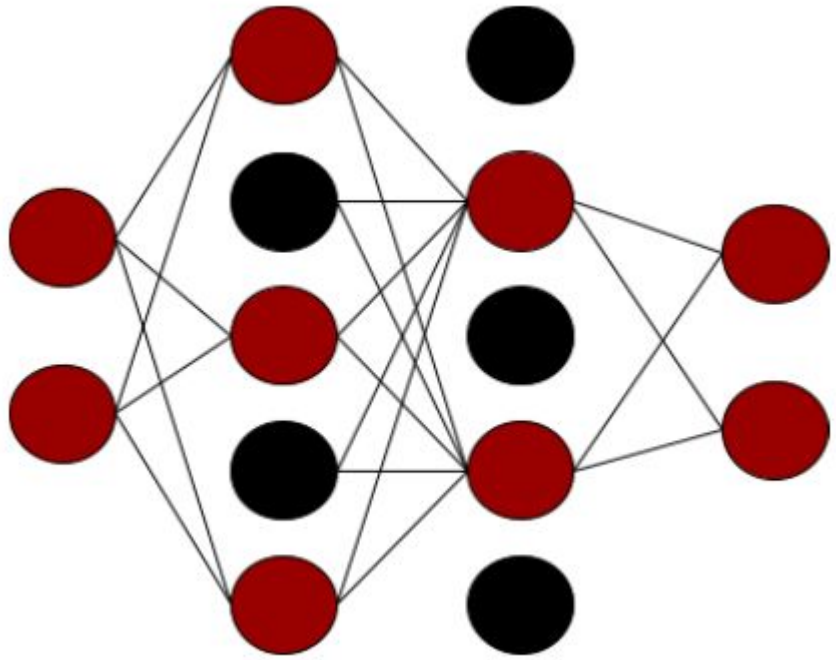
```
model = tf.keras.models.Sequential([
                    tf.keras.layers.Flatten(input_shape=(28,28)),
                    tf.keras.layers.Dense(256, activation=tf.nn.relu),
                    tf.keras.layers.Dense(128, activation=tf.nn.relu),
                    tf.keras.layers.Dense(64, activation=tf.nn.relu),
                    tf.keras.layers.Dense(10, activation=tf.nn.softmax)])
```



Fashion MNIST Dataset

- 20 Epochs
- 94.0% Accuracy on Train Data
- 88.5% Accuracy on Validation Data

Dropout Regularization



```
model = tf.keras.models.Sequential([
                    tf.keras.layers.Flatten(input_shape=(28,28)),
                    tf.keras.layers.Dense(256, activation=tf.nn.relu),
                    tf.keras.layers.Dropout(0.2),
                    tf.keras.layers.Dense(128, activation=tf.nn.relu),
                    tf.keras.layers.Dropout(0.2),
                    tf.keras.layers.Dense(64, activation=tf.nn.relu),
                    tf.keras.layers.Dropout(0.2),
                    tf.keras.layers.Dense(10, activation=tf.nn.softmax)])
```

Fashion MNIST Dataset

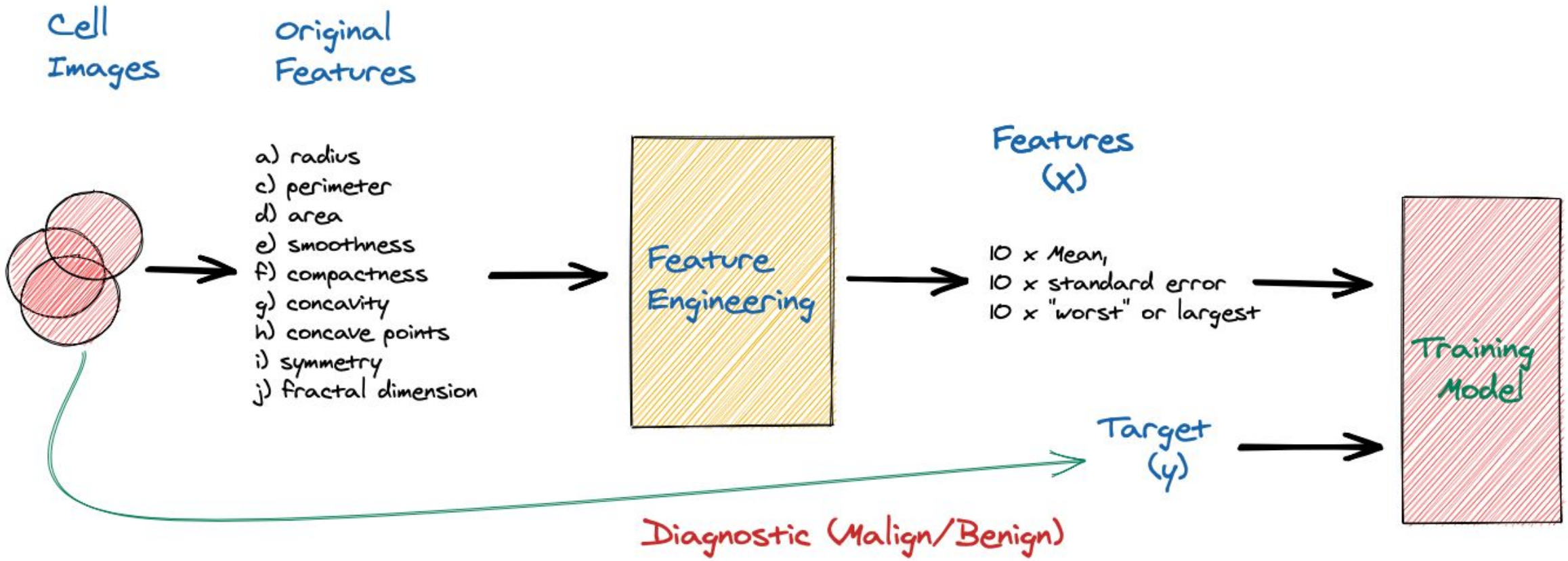
- 20 Epochs
- 89.5% Accuracy on Train Data
- 88.3% Accuracy on Validation Data

Removing a random number of neurons and connections (in this example, 20%), reduces the chances of the neurons becoming overspecialized and the model will generalize better, reducing the overfit.

Wisconsin Diagnostic Breast Cancer (WDBC)

Code Time!

Breast_Cancer_Classification.ipynb



UCI ML Breast Cancer Wisconsin (Diagnostic) datasets. https://goo.gl/U2Uwz2

Reading Material

Main references

- Harvard School of Engineering and Applied Sciences - CS249r: Tiny Machine Learning
- Professional Certificate in Tiny Machine Learning (TinyML) – edX/Harvard
- Introduction to Embedded Machine Learning - Coursera/Edge Impulse
- Computer Vision with Embedded Machine Learning - Coursera/Edge Impulse
- Fundamentals textbook: “Deep Learning with Python” by François Chollet
- Applications & Deploy textbook: “TinyML” by Pete Warden, Daniel Situnayake
- Deploy textbook “TinyML Cookbook” by Gian Marco Iodice

I want to thank Shawn Hymel and Edge Impulse, Pete Warden and Laurence Moroney from Google, Professor Vijay Janapa Reddi and Brian Plancher from Harvard, and the rest of the TinyMLedu team for preparing the excellent material on TinyML that is the basis of this course at UNIFEI.

The IESTI01 course is part of the **TinyML4D**, an initiative to make TinyML education available to everyone globally.

Thanks

UNIFEI