



### Developing Machine-Learning Applications on the Raspberry Pi Pico

## DAY 4 : Designing and Testing a Machine-Learning Model

Sponsored by



1111111









### Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click "Help" or submit a question asking for assistance.
- Participate in 'Group Chat' by maximizing the chat widget in your dock.
- Submit questions for the lecturer using the Q&A widget. They will follow-up after the lecture portion concludes.



### THE SPEAKER



Jacob Beningo

Visit 'Lecturer Profile'

### Beningo Embedded Group - President

Focus: Embedded Software Consulting

An independent consultant who specializes in the design of real-time, microcontroller based embedded software. He has published two books:

- <u>Reusable Firmware Development</u>
- MicroPython Projects
- Embedded Software Design (https://bit.ly/3PZCtNO)

Writes a weekly blog for DesignNews.com focused on embedded system design techniques and challenges.

Visit <u>www.beningo.com</u> to learn more ...

Visit 'Lecturer Profile' in your console for more details.





### **Course Sessions**

- Getting Started with the Raspberry Pi Pico and Machine Learning
- Machine-Learning Tools and Process Flow
- Collecting Sensor Data Using Edge Impulse
- Designing and Testing a Machine-Learning Model
- Deploying Machine-Learning Models and Next Steps











### Designing the ML Model – Impulse Designer

🔁 EDGE IMPULSE	Jacob Beningo / Beningo-LEL-PicoML	US .
	Project info	
🛄 Dashboard		
Devices		
🔀 Data sources	Jacob Beningo / Beningo-CEC-PicoML	
Data acquisition	This is your Edge Impulse project. From here you acquire new training data, design impulses and train models.	
↓ Impulse design		
Create impulse		
<ul> <li>Spectral features</li> </ul>	About this project	Sharing
<ul> <li>NN Classifier</li> </ul>		
🧭 EON Tuner	Creating your first impulse (100% complete)	Your project is private.
🔀 Retrain model		Image: The second sec
🎢 Live classification	Acquire data	
Model testing	Every Machine Learning project starts with data. You can capture data from a development board or your phone, or import data you already collected.	Summary
Performance calibration	✓ LET'S COLLECT SOME DATA	
Versioning		1
🎁 Deployment	W         Design an impulse           Teach the model to interpret previously unseen data, based on historical	
	data. Use this to categorize new data, or to find anomalies in sensor readings.	DATA COLLECTED 17m 2s
GETTING STARTED	* GETTING STARTED: CONTINUOUS MOTION RECOGNITION	-





### Designing the ML Model – Impulse Designer

Time series data	
Input axes (6)	
accX, accY, accZ, gryX, gryY, gryZ	
Window size	0
	2000 ms
Window increase	2000 113.
	80 ms.
Frequency (Hz)	0
62.5 C	
Zero-pad data	0
<b>×</b>	

- Window Size the size of the data that we will slide across to analyze data features
- Window Increase the size of the window steps
- **Zero-pad** If the sample data set is smaller than the window, pad remaining space with 0's.





### Designing the ML Model – Impulse Designer

Add a processing block		×
Did you know? You can bring your ov	vn DSP code.	
DESCRIPTION	AUTHOR RECOMMEND	ED
Spectral Analysis Great for analyzing repetitive motion, such as data from accelerometers. Extracts the frequency and power characteristics of a signal over time.	Edgelmpulse Inc. 🄺	Add
IMU (Syntiant) Syntiant only. Great for analyzing repetitive motion, such as data from accelerometers. Extracts the frequency and power characteristics of a signal over time.	EdgeImpulse Inc. 🛛 🚖	Add
Flatten Flatten an axis into a single value, useful for slow-moving averages like temperature data, in combination with other blocks.	EdgeImpulse Inc.	Add
Spectrogram Extracts a spectrogram from audio or sensor data, great for non-voice audio or data with continuous frequencies.	EdgeImpulse Inc.	Add
Raw Data Use data without pre-processing, Useful if you want to use deep learning to learn features.	EdgeImpulse Inc.	Add
Some processing blocks have been hidden based on the data in y	our project. Show all blocks anyway	
Add custom block		Cancel





### Designing the ML Model – Impulse Designer

Add a learning block			×
Did you know? You can bring your own model in PyTe	orch, Keras or scik	kit-learn.	
DESCRIPTION	AUTHOR	RECOMMEN	NDED
<b>Classification (Keras)</b> Learns patterns from data, and can apply these to new data. Great for categorizing movement or recognizing audio.	Edgelmpulse Inc	. 🔺	Add
Anomaly Detection (K-means) Find outliers in new data. Good for recognizing unknown states, and to complement classifiers. Works best with low dimensionality features like the output of the spectral features block.	EdgeImpulse Inc	. ★	Add
<b>Regression (Keras)</b> Learns patterns from data, and can apply these to new data. Great for predicting numeric continuous values.	Edgelmpulse Inc		Add
<b>Custom classification</b> Use a custom machine learning classification model (with type "other") from your organization.	Edgelmpulse Inc		Add
Some learning blocks have been hidden based on the data in you	ır project. Show all blo	ocks anyway	
			Cancel



### Designing the ML Model – Impulse Designer

Time series data	Spectral Analysis	Classification (Keras)	Output features
Input axes (6) accX, accY, accZ, gryX, gryY, gryZ Window size	Name Spectral features Input axes (6) accX accY accZ gryX gryY	Name NN Classifier Input features Output features 4 (Circle, Idle, UpDown, Wave)	4 (Circle, Idle, UpDown, Wave) Save Impulse
Zero-pad data ⑦ ✔	gryZ F Add a processing block	T Add a learning block	





## Did you expect creating the ML model to be this simple?

- 1) Yes
- 2) No













#### Spectral Feature Tuning Change Data Parameters Show: All labels Raw data Circle.3fbitlht (Circle) Raw features DSP result 3.5123, 5.5563, 19.6109, -14.1602, -54.6265, -158.0200, 5.4426, 5.4875, 19.6109, -67.2607, -51.4526, -151.9165, 7.3269, 5.6276, 19.6109, -131.53\_ After filter 250-200-150-Parameters Filter Scale axes 1 1872.00 0.00 208.00 416.00 624.00 832.00 1040.00 1248.00 1456.00 1664.00 Туре none Spectral power (log) Spectral power FFT length 16 Take log of spectrum? **Overlap FFT frames?** 0.00 7.81 11.72 15.63 19.53 23.44 27.34 31.25 Processed features 10.0814, 0.0077, -1.4047, 2.9466, 2.0966, 1.7632, 1.4024, 1.2794, 1.1973, 1.1489, 1.0798, 2.7147, 0.1431, -1.2046, 1.8892, 1.0782, 0.8276, 0.562...





### Spectral Feature - Tuning

#1  Click to set a description for this version	
Parameters Generate features	
Training set	
Data in training set	13m 1s
Classes	4 (Circle, Idle, UpDown, Wave)
Training windows	5,928
Calculate feature importance	
	Generate features





### Spectral Feature - Tuning





### Spectral Feature - Tuning

Feature importance ③	All data ~
gryZ Spectral Power 1.95 - 5.86 Hz	
gryX RMS	
gryY Spectral Power 13.67 - 17.58 Hz	
accZ Spectral Power 13.67 - 17.58 Hz	
accZ RMS	
accX RMS	
gryZ Spectral Power 5.86 - 9.77 Hz	
gryY RMS	
accZ Spectral Power 9.77 - 13.67 Hz	
gryY Spectral Power 21.48 - 25.39 Hz	





What should you look for in the feature set to determine if the model will be successful?

- Datasets that overlap
- Datasets that are clearly separated
- Datasets that intermix
- Other













### Training the Model

Neural Network settings		:
Training settings		
Number of training cycles ⑦	30	
Learning rate ⑦	0.0005	
Validation set size ⑦	20	%
Auto-balance dataset ⑦		
Neural network architecture		
Input layer (66 features)		
Dense layer (20 neurons)		
Dense layer (10 neurons)		
Add an extra layer		
Output layer (4 classes)		
Start training		





### Training the Model

Training output	Cancel
Epoch 11/30 149/149 – 1s – loss: 0.2765 – accuracy: 0.9224 – val_loss: 0.1957 – val_accuracy: 0.9098 – 776ms/epoch – 5ms/step	
Epoch 12/30 149/149 – 1s – loss: 0.1407 – accuracy: 0.9344 – val_loss: 0.1286 – val_accuracy: 0.9511 – 779ms/epoch – 5ms/step Epoch 13/30	
149/149 – 1s – loss: 0.0941 – accuracy: 0.9791 – val_loss: 0.0774 – val_accuracy: 0.9933 – 700ms/epoch – 5ms/step Epoch 14/30	
149/149 – 1s – loss: 0.0451 – accuracy: 0.9968 – val_loss: 0.0267 – val_accuracy: 1.0000 – 719ms/epoch – 5ms/step Epoch 15/30	
149/149 - 1s - loss: 0.0203 - accuracy: 0.9992 - val_loss: 0.0148 - val_accuracy: 1.0000 - 770ms/epoch - 5ms/step Epoch 16/30	
149/149 – 1s – loss: 0.0132 – accuracy: 0.9996 – val_loss: 0.0104 – val_accuracy: 1.0000 – 708ms/epoch – 5ms/step Epoch 17/30	
149/149 – 1s – loss: 0.0098 – accuracy: 1.0000 – val_loss: 0.0081 – val_accuracy: 1.0000 – /15ms/epoch – 5ms/step Epoch 18/30	





### Training the Model

st training performance (val	idation set)			
ACCURACY 100.0%		LO.	ss 00	
nfusion matrix (validation set)		1015		MANE
	CIRCLE	IDLE	OPDOWN	WAVE 00%
	0%	100%	0%	0%
PDOWN	0%	0%	100%	0%
AVE	0%	0%	0%	100%
1 SCORE	1.00	1.00	1.00	1.00
Circle - correct				•
Circle - correct Idle - correct UpDown - correct Wave - correct				•
Circle - correct Idle - correct UpDown - correct Wave - correct				
Circle - correct Idle - correct UpDown - correct Wave - correct			•	•
Circle - correct Idle - correct UpDown - correct Wave - correct		و بیمو ب		
Circle - correct Idle - correct UpDown - correct Wave - correct		م موجوع		





### Training the Model







What can we do to improve our model?

- Use the IMU analysis instead of spectral features
- Gather more data or throw away "bad" data
- Adjust the training parameters
- Change the features looked at
- All the above and more











# Thank you for attending

Please consider the resources below:

- <u>www.beningo.com</u>
  - Blog, White Papers, Courses
  - Embedded Bytes Newsletter
    - <u>http://bit.ly/1BAHYXm</u>
  - Embedded Software Design
    - <u>https://bit.ly/3PZCtNO</u>



From <u>www.beningo.com</u> under

- Blog > CEC – Developing Machine-Learning Applications on the Raspberry Pi Pico

CEC Continuing Education Center



# Thank You

Sponsored by



11111111



