

# tinyML<sup>®</sup> Research Symposium

*Enabling Ultra-low Power Machine Learning at the Edge*

April 22, 2024



[www.tinyML.org](http://www.tinyML.org)



# Comparing classic ML techniques with DL for tiny ML human activity recognition

Bruno Montanari

Field Application Engineer

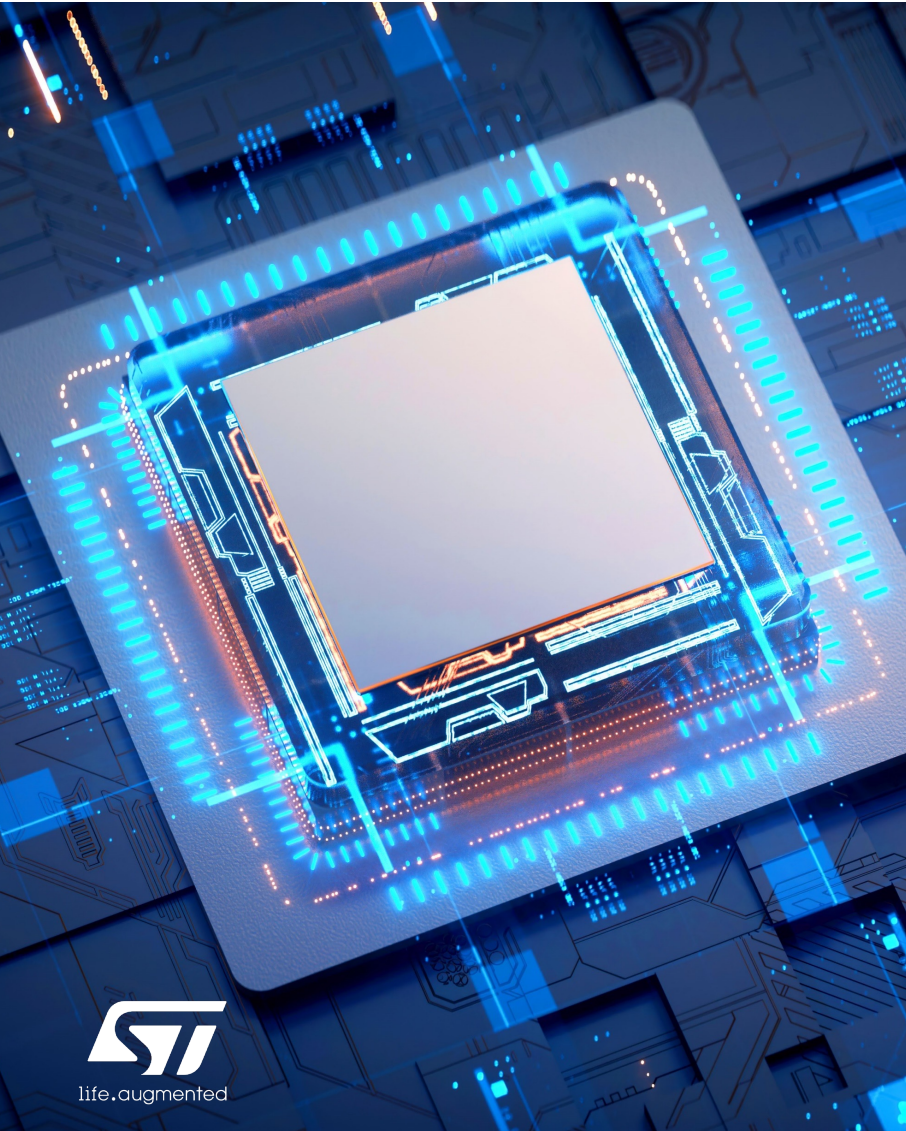
STMicroelectronics

Centro Universitario FEI - Brazil





# Introduction

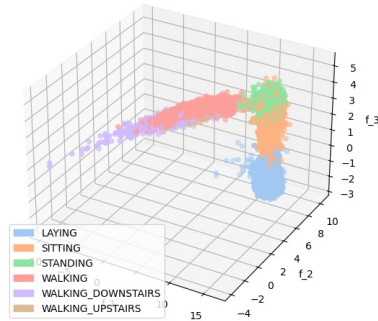


- Analysis and classification of Human Activity Recognition (HAR) on MCUs using inertial sensors
  - running, walking, climbing up and down stairs
- Compare the inference time, memory density and power consumption when using:
  - Classic ML Algorithms
  - DL
  - Add PCA before the ML and DL
- Expected Benefits:
  - Lower complexity
  - Lower memory cost
  - Faster inference

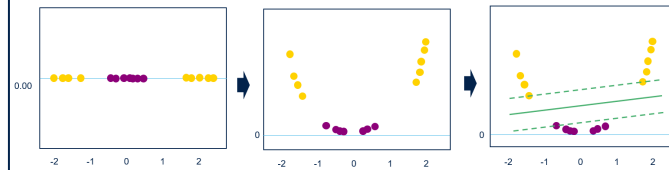


# Main ML and DL algorithms

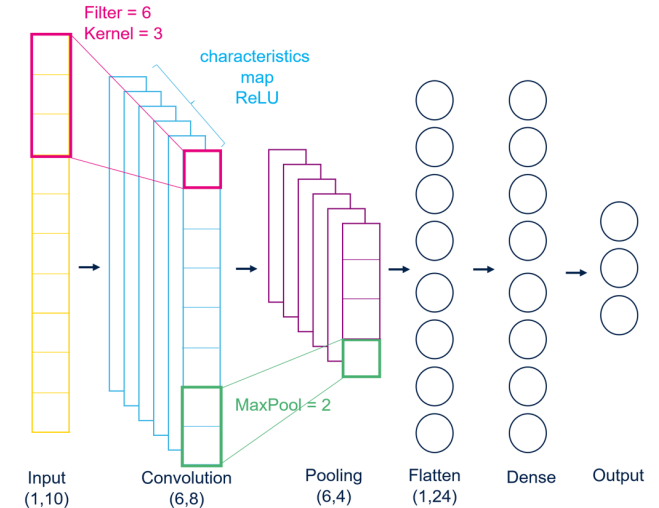
## Principal Component Analysis



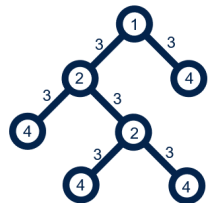
## Support Vector Machine



## 1 Dimensional Convolution Neural Network

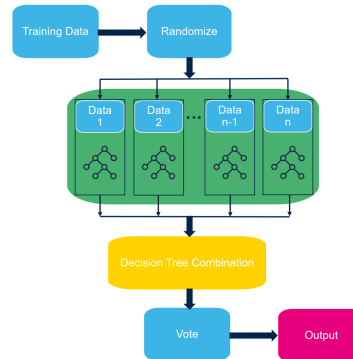


## Decision Tree Regressor



1. Root Node
2. Chance Node
3. Branch
4. Leaf Node

## Random Forest

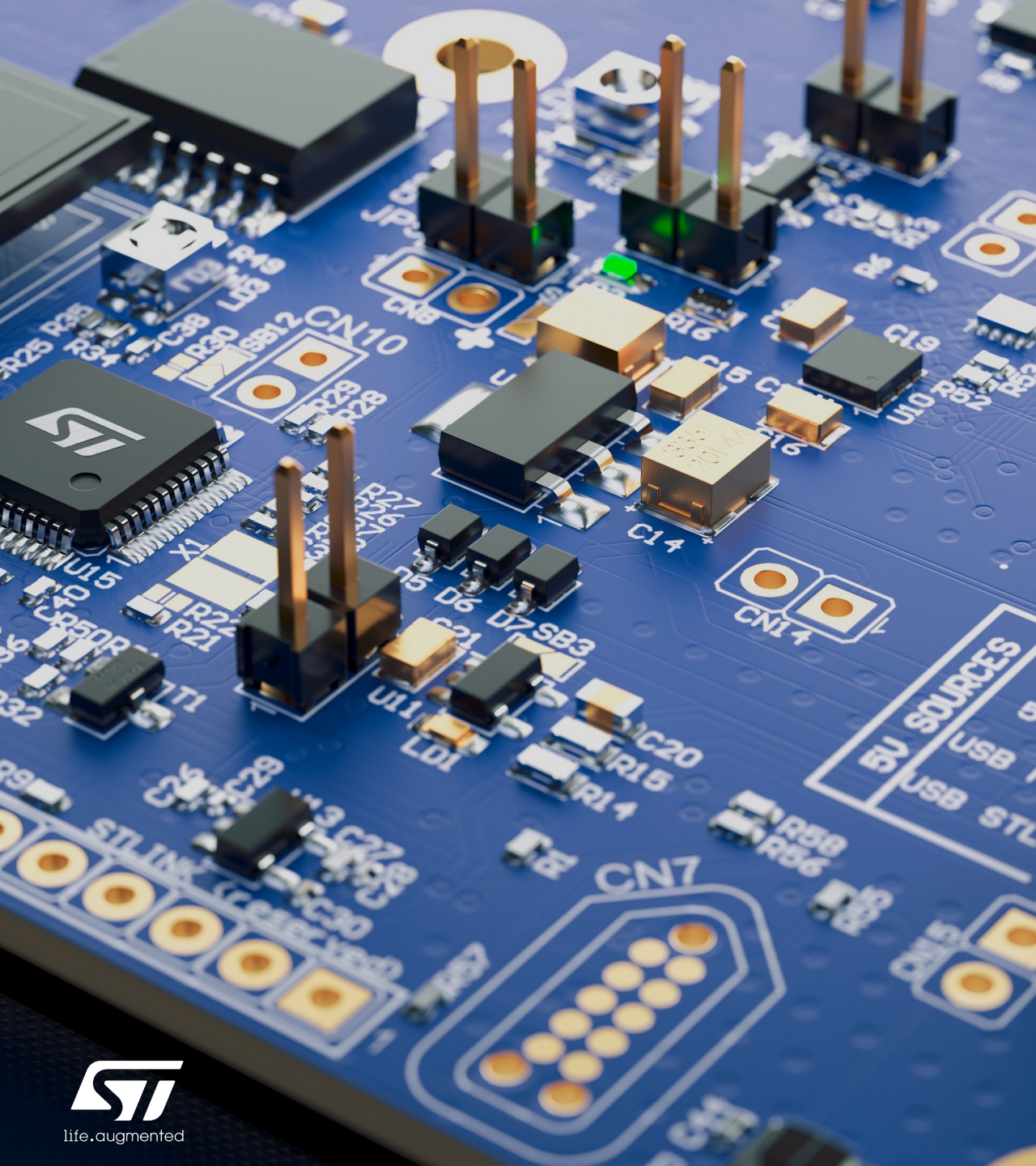


All images created by the author

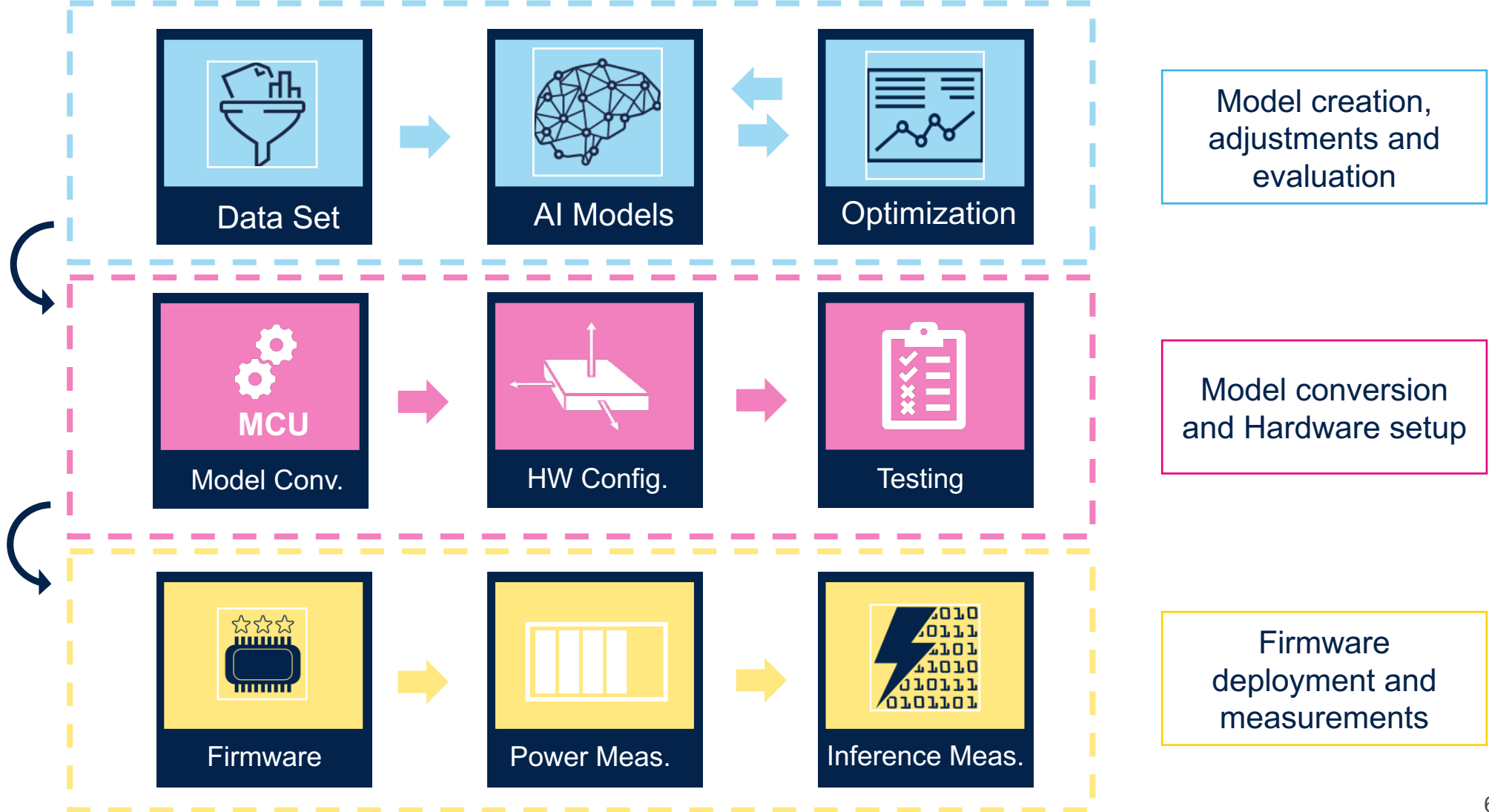


# Resources

- Open data sets used:
  - [UCI HAPT](#)
  - [STMicroelectronics HAR](#)
- Software and framework:
  - Python:
    - Scikit-Learn / Keras / TensorFlowLite
  - Model export
    - TFLite / ONNX / MicroML (raw C)
  - Embedded C
    - STM32CubeIDE + X-CUBE-AI
- Hardware:
  - B-U585I-IOT02A
  - X-NUCLEO-LPM01A



# Methodology – from design to test



# Experiments and results

- Inference time and memory for all evaluated models running on the ARM Cortex M33
  - Balanced memory optimization
  - 8bit Quantization (PTQ)
- Model evaluation
  - Accuracy and confusion matrix
- Power consumption
  - MCU was placed in STOP2 mode when not executing the model with full RAM retention

Basic Information and Characteristics		
Microcontroller	STM32U575 @ 160Mhz	
Run Mode	Typ. 7.15 mA @ 44.7uA/MHz	
Low Power Mode	106uA for the entire board / MCU 19.5uA	
External Memories	512-Mbit NOR / 64-Mbit PSRAM	
Internal Memories	2MB FLASH / 786 KB SRAM	
MEMS Settings	ODR = 50Hz	ODR = 26Hz
Datasets	UCI HAPT	STMicroelectronics
Input Size	(561,1)	(26,3)
PCA Cost (FLASH/RAM)	217.51 KB 4.51 KB	23.42 KB 0.17 KB
AI Models Explored	DTR, RF, SVM and 1DCNN	

# Avg Results

## UCI HAPT

UCI HAPT			
Model	FLASH (KB)	RAM (KB)	Inference (ms)
DTR	13.34	3.72	0.023
PCA+DTR	217.51+23.19	4.51+3.72	15.4+0.054
RF	49.36	3.72	0.149
PCA+RF	217.51+112.18	4.51+3.72	15.4+0.328
SVM	13575.70	52.23	178
PCA+SVM	217.51+296.27	4.51+28.62	15.4+2.66
CNN	1131.22	55.00	28.4
PCA+CNN	217.51+41.22	4.51+3.91	15.4+0.7

## ST

ST			
Model	FLASH (KB)	RAM (KB)	Inference (ms)
DTR	12.09	1.63	0.023
PCA+DTR	23.42+12.93	0.17+1.59	2.08+0.027
RF	47.53	1.63	0.148
PCA+RF	23.42+38.06	0.17+1.59	2.08+0.138
SVM	658.57	30.52	15.48
PCA+SVM	23.42+200.79	0.17+16.24	2.08+10.19
CNN	60.84	4.84	1.34
PCA+CNN	23.42+40.46	0.17+3.91	2.08+0.7

## Power consumption

Model	STMICROELECTRONICS	UCI HAPT
	Avg. Current (uA)	Avg. Current (uA)
PCA + DTR	122	214
PCA + RF	123	218
PCA + SVM	191	239
1DCNN	122	352
PCA + 1DCNN	164	226

## Accuracy

Model	STMICROELECTRONICS	UCI HAPT
	Accuracy %	Accuracy %
DTR	94.61	98.32
PCA + DTR	98.32	97.63
RF	96.38	95.44
PCA + RF	97.28	94.19
SVM	94.78	94.80
PCA + SVM	94.55	94.37
1DCNN	99.19	99.38
PCA + 1DCNN	97.80	91.65



# UCI HAPT (PCA+1DCNN)

True label	LAYING	1936	8	0	0	0	0
	SITTING	3	1349	425	0	0	0
	STANDING	0	141	1765	0	0	0
	WALKING	0	0	0	1663	8	51
	WALKING_DOWNSTAIRS	0	0	0	47	1282	77
	WALKING_UPSTAIRS	0	0	0	3	19	1522
			LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS

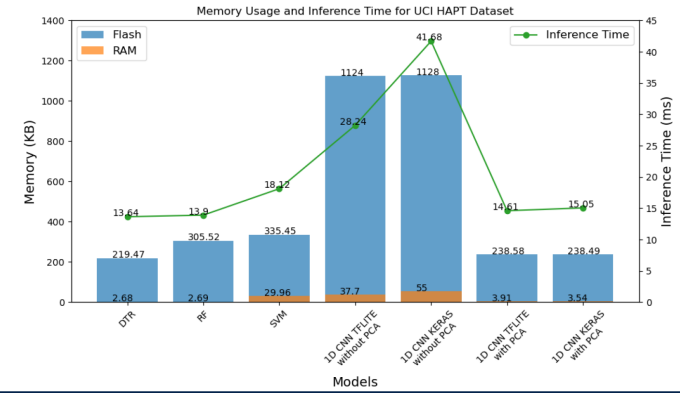
# ST (PCA+1DCNN)

True label	STANDING	2587	113	0
	WALKING	25	2899	1
	RUNNING	2	1	1272
		STANDING	WALKING	RUNNING

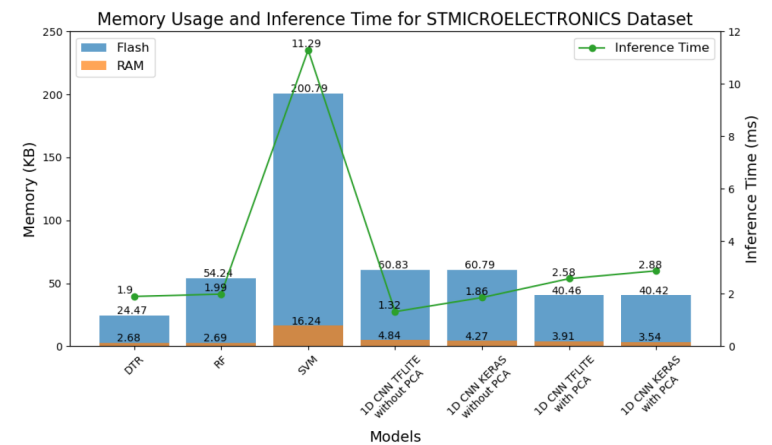
Predicted label

# Best Results

## UCI HAPT



## ST



# Key takeaways

- DTR proved to be an interesting alternative to 1DCNN for the data sets used
- Selective Advantage of PCA: Incorporating it can lead performance improvements
- Benefits of PCA are not uniform across all algorithms and must be weighed against the increased computational cost for others
- The combination of PCA + 1DCNN provided an equivalent memory and inference time, but with lower complexity
- <https://github.com/brunomontanari/TinyMLSymposium2024>



# Future work

- Evaluate more datasets
  - WISDM
  - PAMP2
  - UniMib
  - SHL
- Exploring various optimization and quantization methods available
  - TFLITE
  - X-CUBE-AI
- Adapting tinyML frameworks to compare classic ML and DL for FL scenarios
  - Benefits of lowering the AI model complexity



**Questions?**



# Our technology starts with You

© STMicroelectronics - All rights reserved.

ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.

For additional information about ST trademarks, please refer to [www.st.com/trademarks](http://www.st.com/trademarks).

All other product or service names are the property of their respective owners.



life.augmented

# Copyright Notice

This presentation in this publication was presented at the tinyML<sup>®</sup> Research Symposium 2024. The content reflects the opinion of the author(s) and their respective companies. The inclusion of presentations in this publication does not constitute an endorsement by tinyML Foundation or the sponsors.

There is no copyright protection claimed by this publication. However, each presentation is the work of the authors and their respective companies and may contain copyrighted material. As such, it is strongly encouraged that any use reflect proper acknowledgement to the appropriate source. Any questions regarding the use of any materials presented should be directed to the author(s) or their companies.

tinyML is a registered trademark of the tinyML Foundation.

**[www.tinyml.org](http://www.tinyml.org)**