Master's degree ICT Internet Multimedia Engineering

MACHINE LEARNING FOR HUMAN DATA – FINAL EXAMINATION

Instructor

Michele Rossi - michele.rossi@unipd.it

Lab. classes

Eleonora Cicciarella - eleonora.cicciarella@phd.unipd.it

Cesare Bidini - cesare.bidini@phd.unipd.it

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

General guidelines

The final exam is project-based

- This does not mean that you can avoid understanding the theory…see the next slides

1. Pick a project among the 13 options that we provide to you: each consists of a challenge and an associated dataset
2. Design one/more original solution(s) to the problem based on neural networks, implement it/them in TensorFlow and evaluate and compare the performance
3. Prepare a **report** and a **presentation** describing your work

- You can work in a group with another student
  - max 2 people per group
  - you are free to arrange the groups
  - both members have to contribute to the work

Exam dates and submission deadlines

- Exam: January 28-30, 2026
  - report + code submission deadline: **Jan. 25, 2026**
- Exam: February 17-19, 2026
  - report + code submission deadline: **Feb. 14, 2026**
- Exam: June 15-16, 2026
  - report + code submission deadline: **June 12, 2026**
- Exam: July 6-7, 2026
  - report + code submission deadline: **July 3, 2026**
- Exam: September 15-16, 2026
  - report + code submission deadline: **Sept. 12, 2026**

IMPORTANT NOTES on next page!

Exam dates: important notes

- The exam will be held in presence
  - online exams are no longer allowed by UNIPD

- Depending on the number of students we may need to split you into groups that will take the exam on different days
  - the exam dates in the previous slide indicate the first days of the session
  - please, be prepared to be scheduled for a different day than the one indicated (try to be available for the day of the exam and the following ones; in case you have unmovable appointments, inform us **as soon as you enroll**)
  - we will send you the schedule some days before the exam when the UNIWEB enrollment will close

For the final examination you must

1. Fill out the group selection form in Moodle indicating the students (1 or 2) in your group (we will send the instructions through the Moodle's news channel) remember to do that!

Group and project self-selection

2. Upload in Moodle (following the instructions about naming etc.)

A. a report (use the LaTex template available on Moodle)
B. the code of the implementation in **TensorFlow**

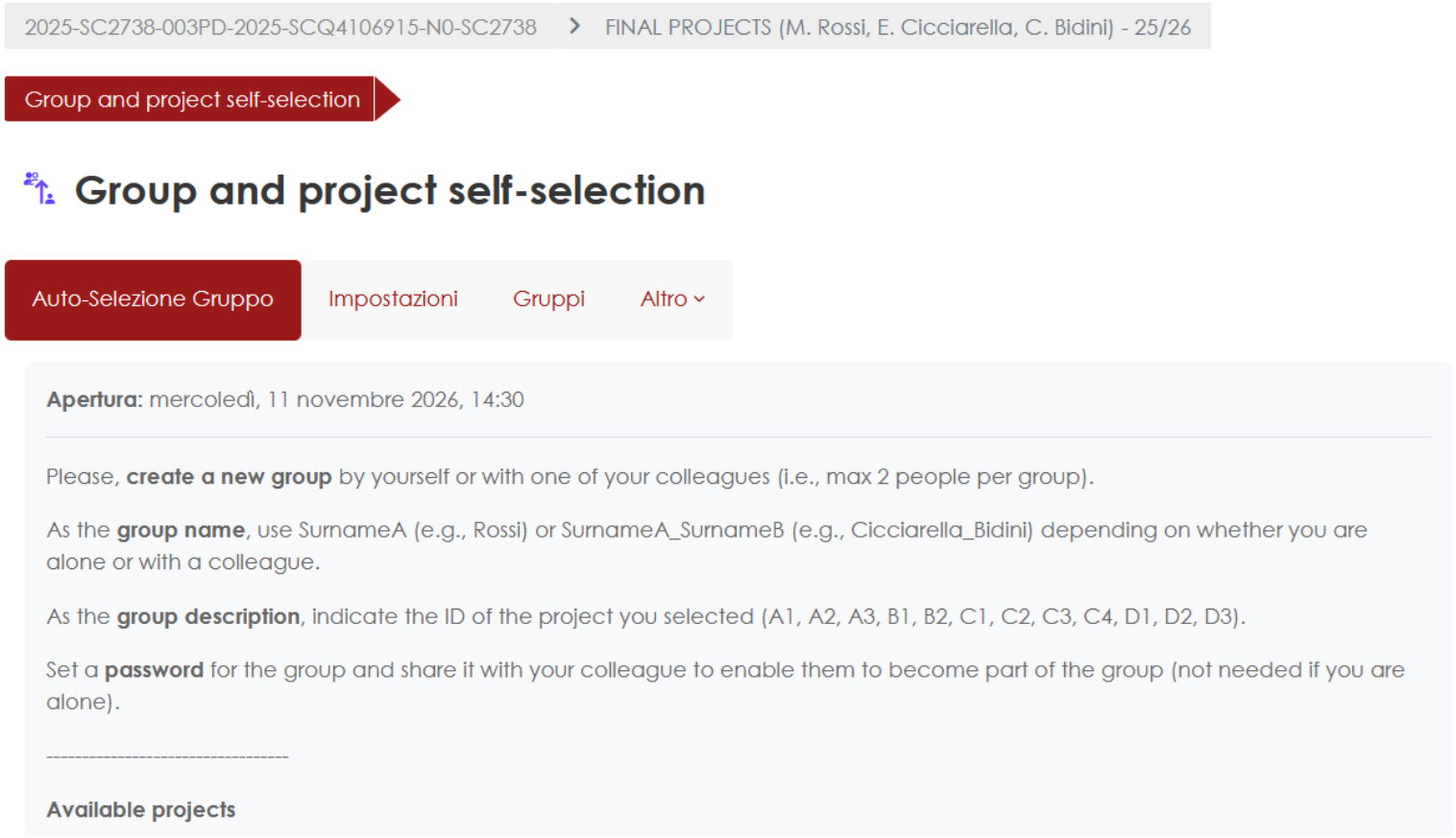
COMPITO
Project report and code upload - January 28-30, 2026

3. Prepare a presentation through slides (**20 minutes strict**, possibly including a demo) for the day of the exam

Group self-selection in Moodle

Deadline: when you enroll in UNIWEB for the exam

You can fill it out also before (recommended)

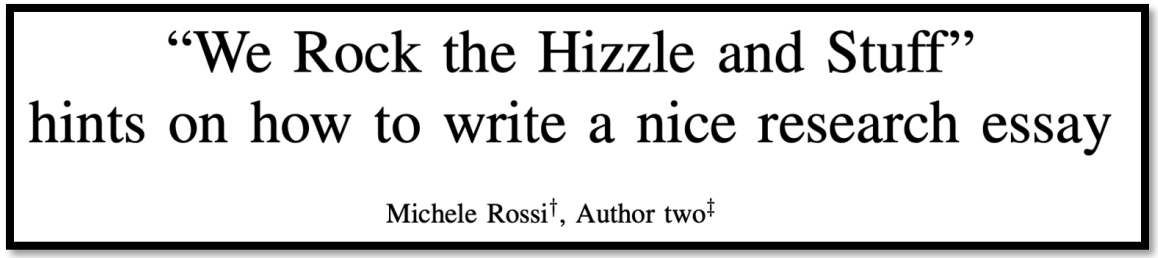


The report

1. **Should be done in LaTex** following the template available on Moodle.
2. Should be written in a **clear and organized** manner.
3. Should include **graphical presentations** of your approaches.
4. Should clearly **show and discuss the results**.

FILE

Project report - Latex template



"We Rock the Hizzle and Stuff"
hints on how to write a nice research essay

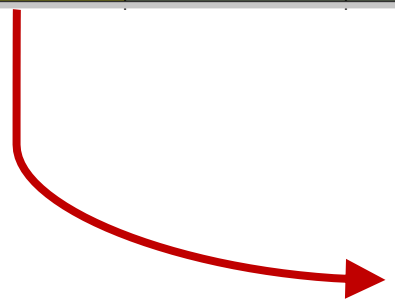
Michele Rossi†, Author two‡

Detailed information will be provided soon on the Moodle page of the course

Evaluation

- The evaluation will consider different aspects
  - about the report, the presentation and the project itself

project					written report			oral	
originality (10)	preprocessing (10)	learning architecture (15)	comparison / performance analysis (10)	live demo (10)	clarity of exposition (10)	completeness of results (accuracy, complexity, time) (10)	technical soundness (15)	duration (10)	clarity (10)



Show how the approach works on some examples (using pre-trained networks) or a walkthrough

Guidelines

- Prepare the project and the report considering the grid we use for the evaluation (see previous slide)
  - pay attention to the pre-processing phase (normalize the data…).
  - create original neural network architectures.
  - compare the performance of different approaches (use the correct metrics...check about data balancing).
  - evaluate the performance of the algorithms in terms of running time and complexity (memory occupation)

Guidelines

- Be creative!
  - We provided you with some ideas for possible project developments, but original works are always welcome!
  - You can use the neural network architectures seen during the labs and/or experiment with new approaches!
  - We provide you with some references but try to explore a bit other contributions in the literature that may be helpful (search for them in https://scholar.google.it/)
  - Pre-processing techniques may be useful
  - Implement your own neural network architecture…DO NOT use pre-trained models from Keras: the objective of the project is that you put into practice the things you learned during the theoretical lessons, not to improve your skills about reusing networks/code developed by others :)

Guidelines

- The use of TensorFlow is mandatory
  - Pytorch is only allowed for Spiking Neural Networks
- The use of pretrained networks is **not allowed**
  - You can use them for comparison but cannot be the main architectures
- During the exam we will ask you the reasoning behind using the specific architectures (e.g., CNN/RNN/GNN/SNN/attention…)
  - **Do not use the NN functions as black boxes**: you need to understand why you are using the specific architecture
  - **REMEMBER**: Python is not intelligent, it takes something as input and provides an output, it only checks the shape of the data pay attention and use your theoretical knowledge

Common mistakes to avoid

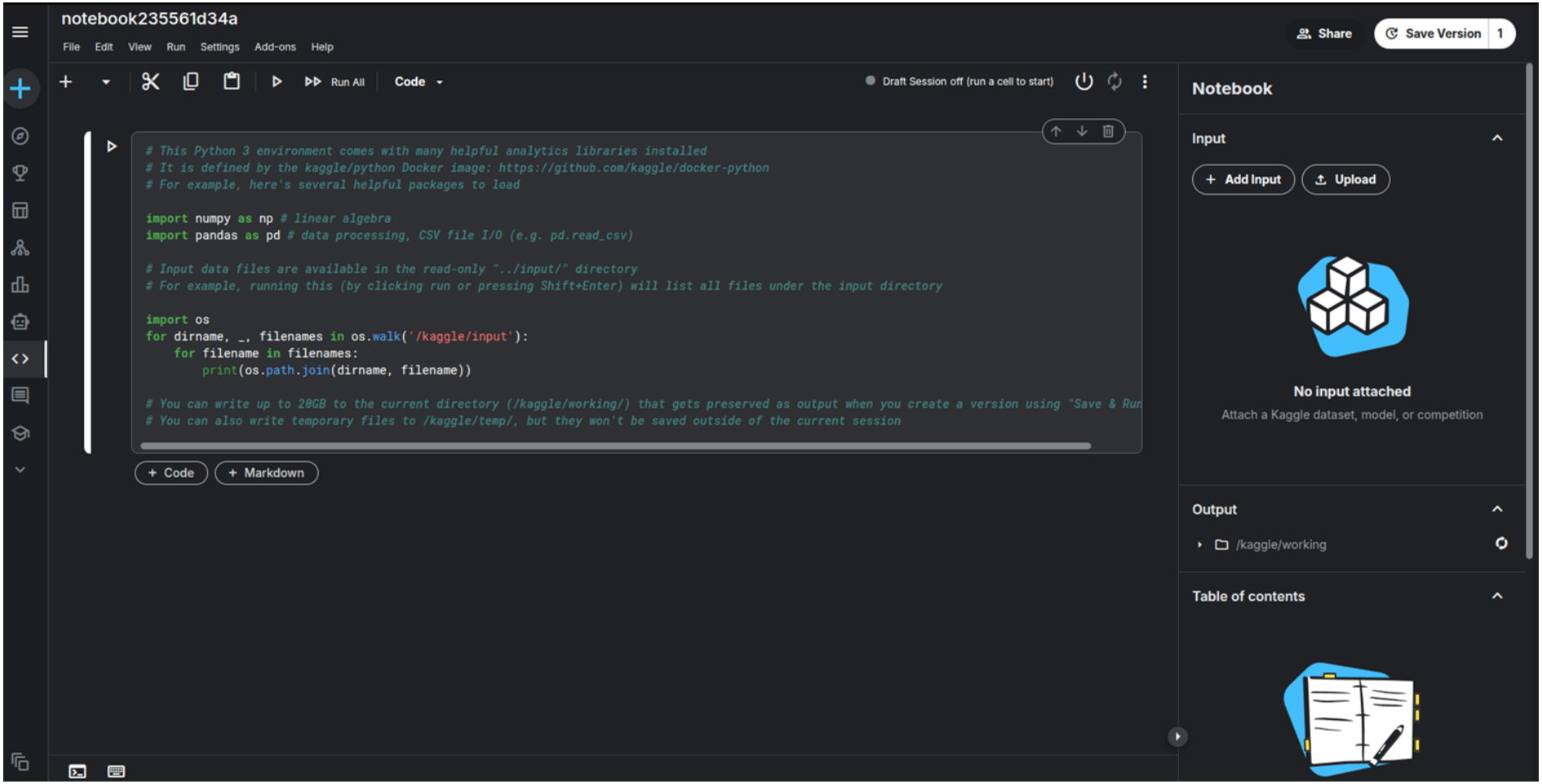
- Data not correctly normalized
  - This is an important step for ML algorithms to not have biases in the algorithm.
- Preprocessing not considered
  - In addition to ML you may need to apply some signal processing algorithms to clean the data before NN.
- Train/validation/test sets not correctly split
  - The three sets do not have to overlap: no data from training should be used during validation or test.
- Validation performed on a small number of samples that is not statistically significant
  - e.g., evaluation performed on 1 or 2 samples…
- Complexity of the algorithms in terms of time and memory not analyzed
- Use wrong input data
  - e.g., for IMU datasets, obtain the activity prediction by using single IMU measurements and not a sequence of measurements.

Computing resources

kaggle

- You can use your own GPU to carry out the training, or any of the available cloud computing services. If you do not have one, we suggest to take a look at Kaggle Notebooks
- Kaggle provides 30 hours/week of free GPUs usage
- Warning: the maximum time allowed for a single run is 9 consecutive hours for CPUs and 12 for GPUs… so remember to save intermediate results (if needed)!

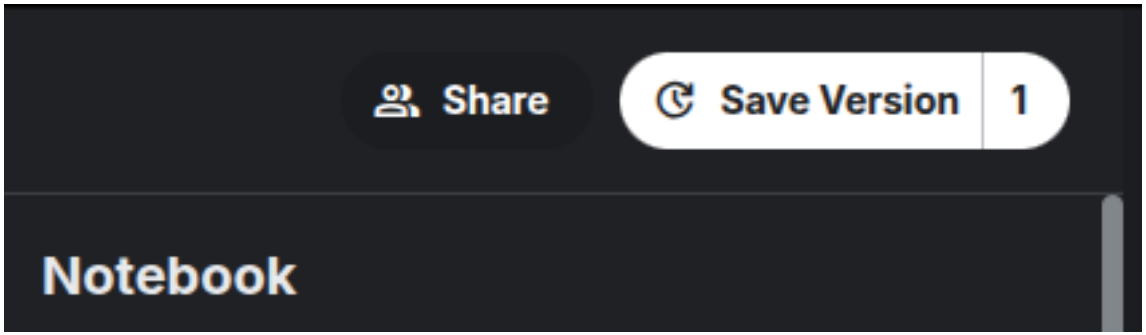
Kaggle Notebooks



Temporary vs Permanent outputs

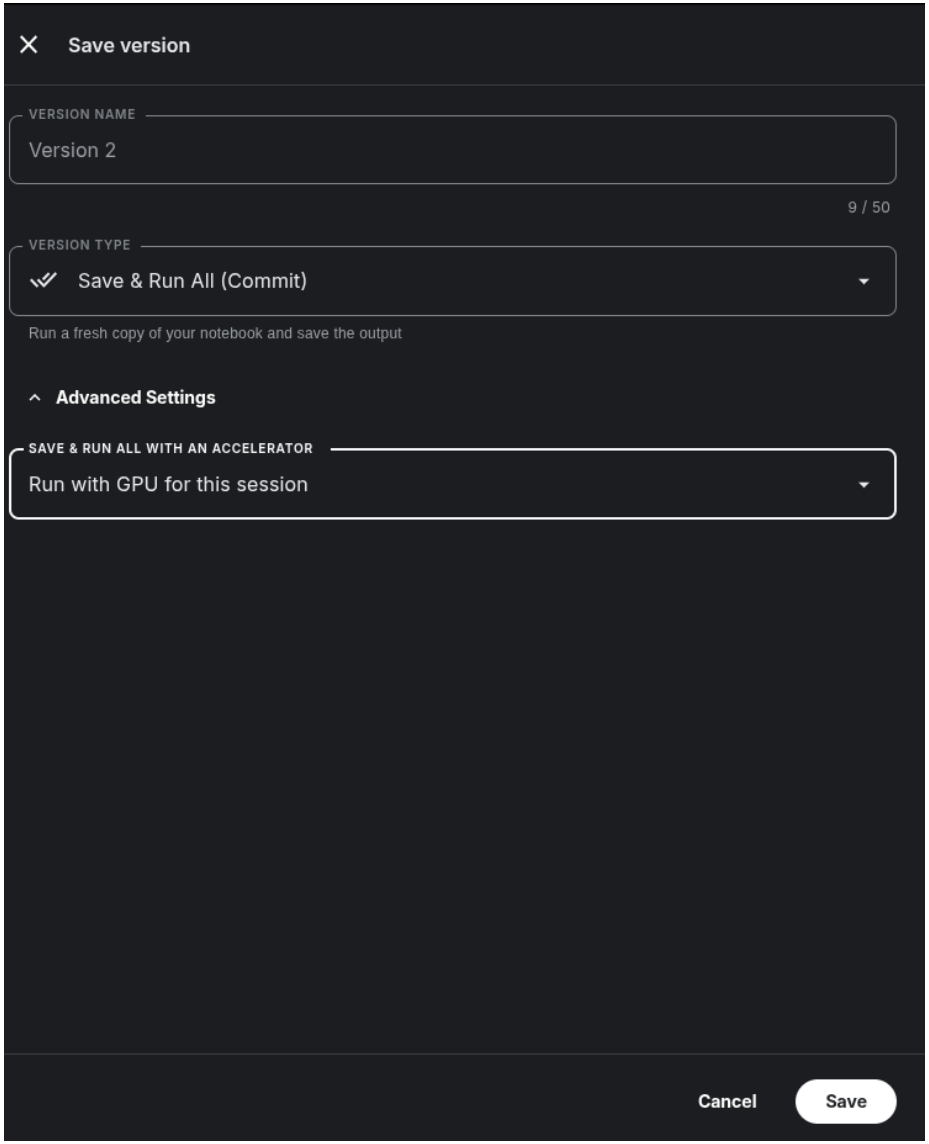
- You can run notebooks by clicking on the symbol  in the notebook, however the output of the session will be temporary.

- To save permanent versions of the notebook with their associated outputs, you should use the Save Version button on the top part of the notebook. You can have multiple saved versions of a notebook.



Temporary vs Permanent outputs

- After clicking the Save Version button, you can choose whether to use a GPU.
- Select “Save and Run All” to run the code and save the outputs.
- You can choose the GPU type from the options in the notebook main interface (on the right panel).



Kaggle tutorials and guides

Kaggle allows you to share, upload and download models, data and notebooks.
For further information, you can check the following urls for useful guides and for the broader kaggle documentation:

https://www.kaggle.com/docs

https://www.kaggle.com/code/nicholasdunham/saving-and-persisting-data-in-kaggle

https://www.datacamp.com/tutorial/tutorial-kaggle-datasets-tutorials-kaggle-notebooks

Proposed Projects

PART A – ON BODY AND ENVIRONMENTAL SENSORS

1) A1: Activity recognition with four accelerometers
2) A2: Pathological gait recognition
3) A3: Motor imagery classification from EEG for brain–computer interface
4) A4: Arrhythmias detection from ECG recording

PART B – AUDIO SIGNALS

1) B1: Speech command recognition (keyword spotting)
2) B2: Environmental sound classification
3) B3: Speaker identification/verification
4) B4: Digit recognition (Spiking Heidelberg Digit)

PART C – IMAGES

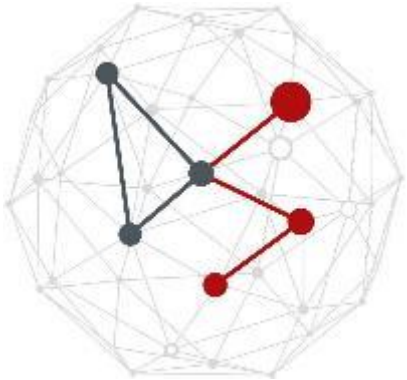
1) C1: Diabetic retinopathy detection
2) C2: Bone age prediction from hand radiographs
3) C3: Lung disease prediction from X-ray images
4) C4: Blood cell type prediction

PART D – RADIO SIGNALS

1) D1: Gesture recognition through radars

MIME
Master's degree ICT Internet Multimedia Engineering

PART A: ON BODY AND ENVIRONMENTAL SENSORS



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
DIPARTIMENTO MATEMATICA
800 1222·2022 ANNI
UNIVERSITÀ DEGLI STUDI DI PADOVA

Proposed Projects

PART A – ON BODY AND ENVIRONMENTAL SENSORS

1) A1: Activity recognition with four accelerometers
2) A2: Pathological gait recognition
3) A3: Motor imagery classification from EEG for brain–computer interface
4) A4: Arrhythmias detection from ECG recording

PART B – AUDIO SIGNALS

1) B1: Speech command recognition (keyword spotting)
2) B2: Environmental sound classification
3) B3: Speaker identification/verification
4) B4: Digit recognition (Spiking Heidelberg Digit)

PART C – IMAGES

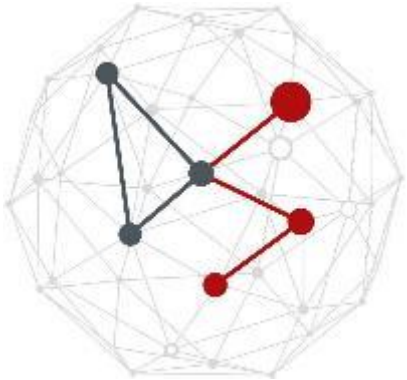
1) C1: Diabetic retinopathy detection
2) C2: Bone age prediction from hand radiographs
3) C3: Lung disease prediction from X-ray images
4) C4: Blood cell type prediction

PART D – RADIO SIGNALS

1) D1: Gesture recognition through radars

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT A1



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project A1 “Activity recognition with four accelerometers”

Reference papers

[Fadel19] Fadel, W. F., Urbanek, J. K., Albertson, S. R., Li, X., Chomistek, A. K., & Harezlak, J., Differentiating Between Walking and Stair Climbing Using Raw Accelerometry Data, in Statistics in Biosciences, 11(2), 334–354, 2019.

[Karas19] Karas, M., Bai, J., Strączkiewicz, M., Harezlak, J., Glynn, N. W., Harris, T., … Urbanek, J. K., Accelerometry Data in Health Research: Challenges and Opportunities. Review and Examples, in Statistics in Biosciences, 11, 210–23, 2019.

Dataset (760.8 MB uncompressed)

https://physionet.org/content/accelerometry-walk-climb-drive/1.0.0/

Why is activity recognition important?

- Navigation systems
  - adapt to user movement
  - e.g., predict direction and only use that portion of the map(s)
  - put the system into power saving mode when there is no mobility
- First responders
  - security personnel, firefighters
  - e.g., who has to be assisted first
- Assisted living
  - react to reduced activity levels
  - unusual mobility patterns
  - user motion-aware services and/or environments
- Rehabilitation
  - measure recovery of motor functions
  - measure effectiveness of rehabilitation
- In most of these cases the use of cameras is not possible
- The system has to be unobtrusive, lightweight, portable, … □ use IMU or radio waves

Dataset description

- Four 3-axial ActiGraph GT3X+ wearable accelerometers at
  1. left ankle
  2. right ankle
  3. left hip
  4. left wrist
- Dataset: collected from 13 males & 19 females aged between 23 and 52
- Six activities considered
  - 1=walking
  - 2=descending stairs
  - 3=ascending stairs
  - 4=driving
  - 77=clapping
  - 99=non-study activity

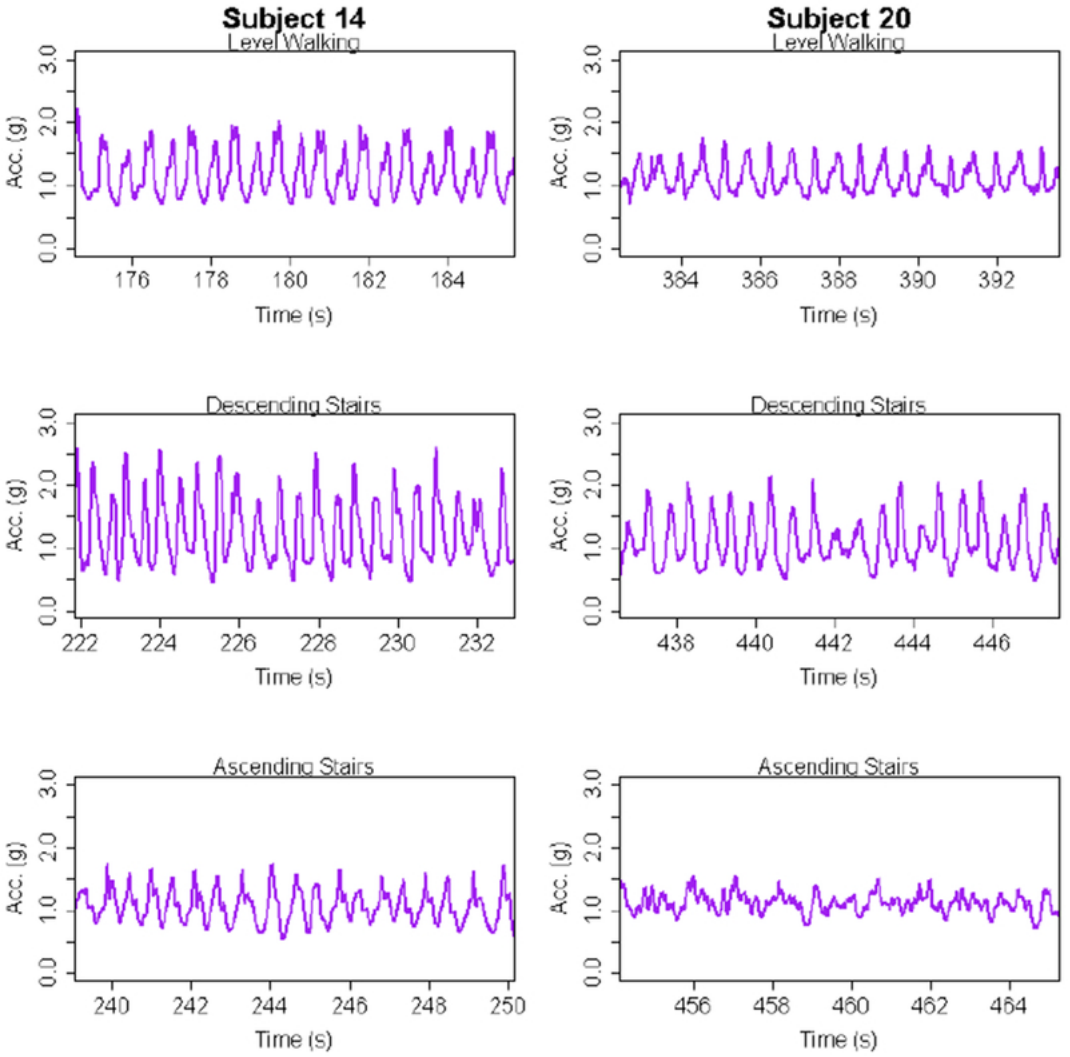
Dataset description

- Dataset format □ Text file: 14 variables
  - activity: Type of activity
  - time_s: Time from device initiation (seconds [s])
  - lw_x: Left wrist x-axis measurement (gravitation acceleration [g])
  - lw_y: Left wrist y-axis measurement (gravitation acceleration [g])
  - lw_z: Left wrist z-axis measurement (gravitation acceleration [g])
  - lh_x: Left hip x-axis measurement (gravitation acceleration [g])
  - lh_y: Left hip y-axis measurement (gravitation acceleration [g])
  - lh_z: Left hip z-axis measurement (gravitation acceleration [g])
  - la_x: Left ankle x-axis measurement (gravitation acceleration [g])
  - la_y: Left ankle y-axis measurement (gravitation acceleration [g])
  - la_z: Left ankle z-axis measurement (gravitation acceleration [g])
  - ra_x: Right ankle x-axis measurement (gravitation acceleration [g])
  - ra_y: Right ankle y-axis measurement (gravitation acceleration [g])
  - ra_z: Right ankle z-axis measurement (gravitation acceleration [g])

Dataset description

- No information is provided to convert the data into the global frame
- In [Fadel19] authors consider the vector magnitude to remove the effects of the sensor orientation

$$vm(t) = \sqrt{x(t)^2 + y(t)^2 + z(t)^2}.$$

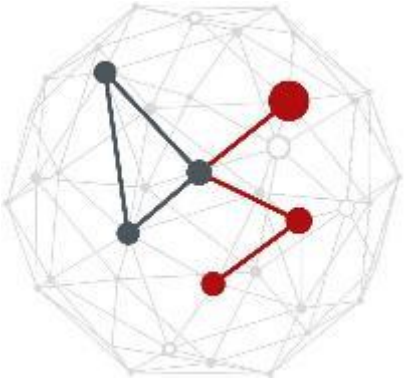


Possible project developments

- Walking VS stairs climbing
  - 3-class classification problem
  - consider the first three activities as in [Fadel19]
- Features
  - use the magnitude or try to use the information on the 3 axes
  - use the features defined in [Fadel19], or
  - compute features with FFT/DCT in an audio-like fashion, or
  - use raw signals with automatic feature extraction
- Classification architecture
  - CNN or RNN…SNN
- REMEMBER: this is about sequential data…you cannot perform the classification by considering a single sample (a common mistake)

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT A2



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project A2 “Pathological gait recognition”

Reference papers

[Jun20_1] K. Jun, Y. Lee, S. Lee, D.-W. Lee, and M. S. Kim, Pathological Gait Classification Using Kinect v2 and Gated Recurrent Neural Networks, IEEE Access, vol. 8, pp. 139881-139891, 2020.

[Jun20_2] K. Jun, D. W. Lee, K. Lee, S. Lee, and M. S. Kim, Feature Extraction Using an RNN Autoencoder for Skeleton-based Abnormal Gait Recognition, IEEE Access, vol. 8, pp. 19196-19207, 2020.

[Lee19] D. W. Lee, K. Jun, S. Lee, J. K. Ko, and M. S. Kim, Abnormal gait recognition using 3D joint information of multiple Kinects system and RNN-LSTM, in Proceedings of the 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2019.

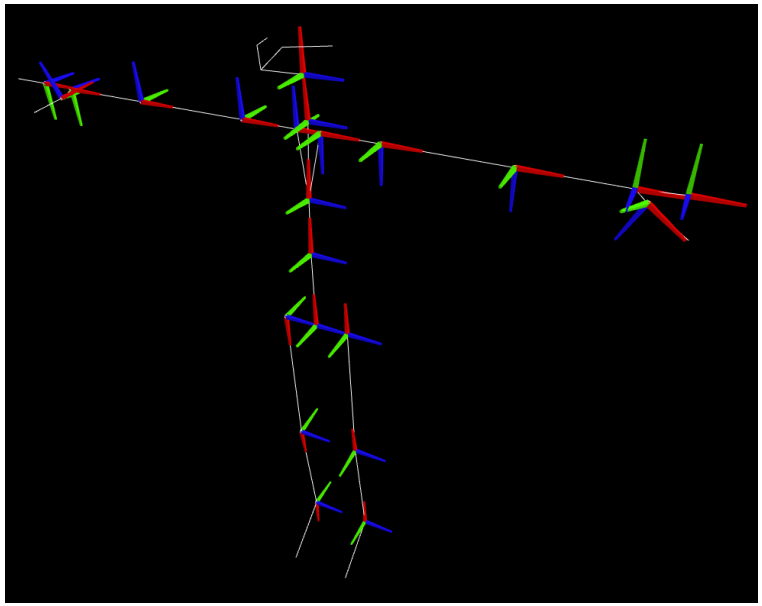
Dataset (111.22 MB uncompressed)

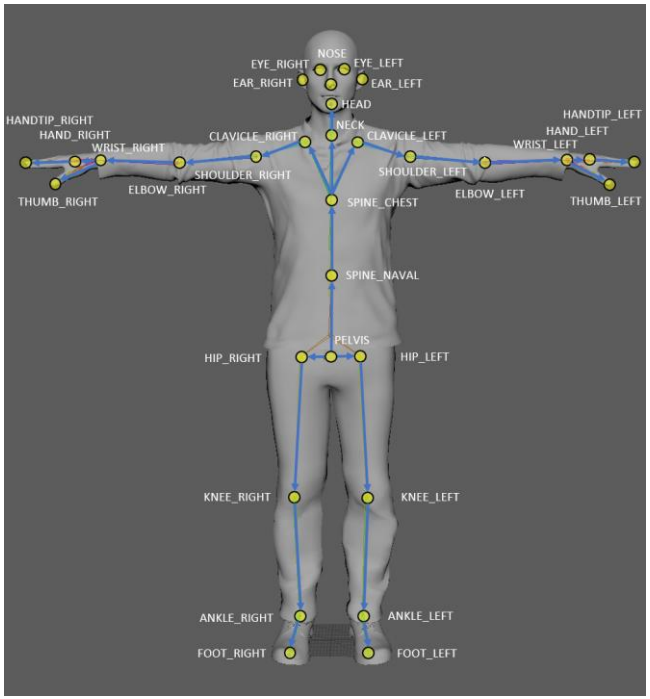
https://drive.google.com/file/d/1BLUXyk_59agThysXwamfVyZahSI-sScI/view?usp=sharing

https://ieee-dataport.org/documents/azure-kinect-3d-skeleton-and-foot-pressure-data-pathological-gaits

Dataset description

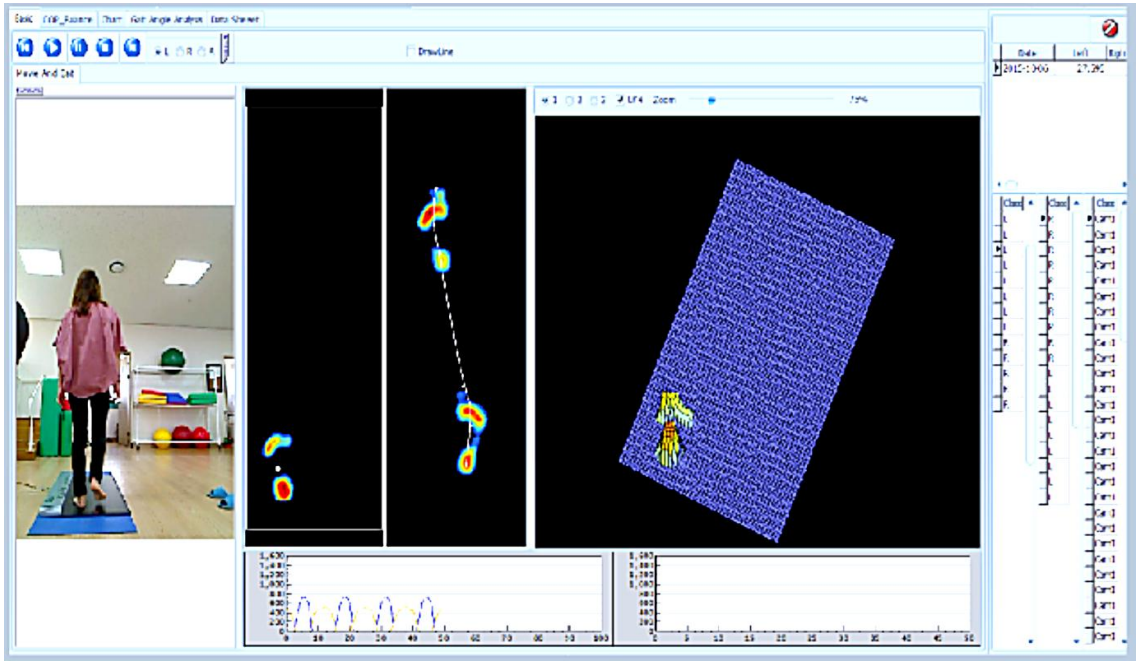
- Sequential skeleton data
  - Azure Kinect (Microsoft Corp. Redmond, WA, USA)
  - https://docs.microsoft.com/en-us/azure/kinect-dk/body-joints





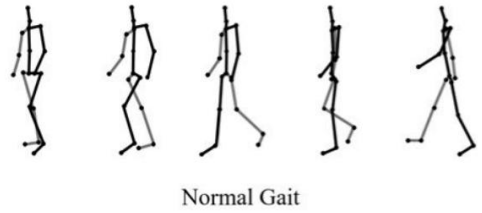
Dataset description

- Average foot pressure data
  - GW1100 (GHIWell, Korea)
  - size = 48 x 128
  - GW1100 is a 1080mm x 480mm sized pressure plate and contains 6,144 high-voltage matrix sensors with maximum pressure 100 N/cm^2

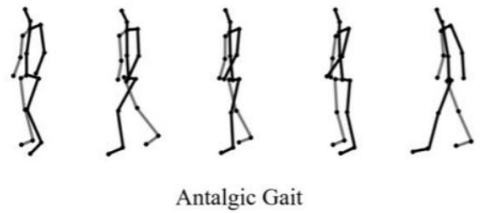


Dataset description

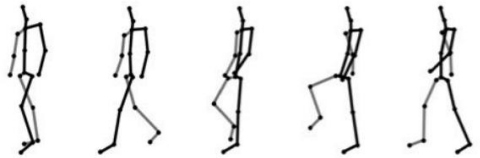
- Simultaneously collected data from the two sensors for normal and five pathological gaits
  1. antalgic
  2. lurching
  3. steppage
  4. stiff-legged
  5. Trendelenburg



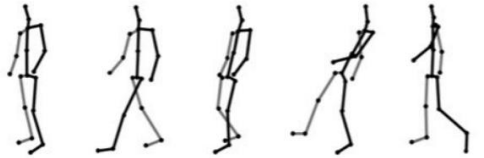
Normal Gait



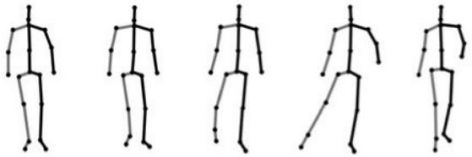
Antalgic Gait



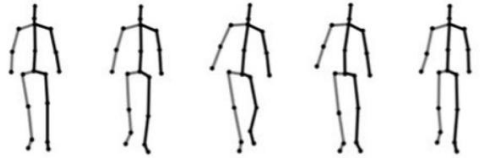
Steppage Gait



Lurching Gait



Stiff-legged Gait

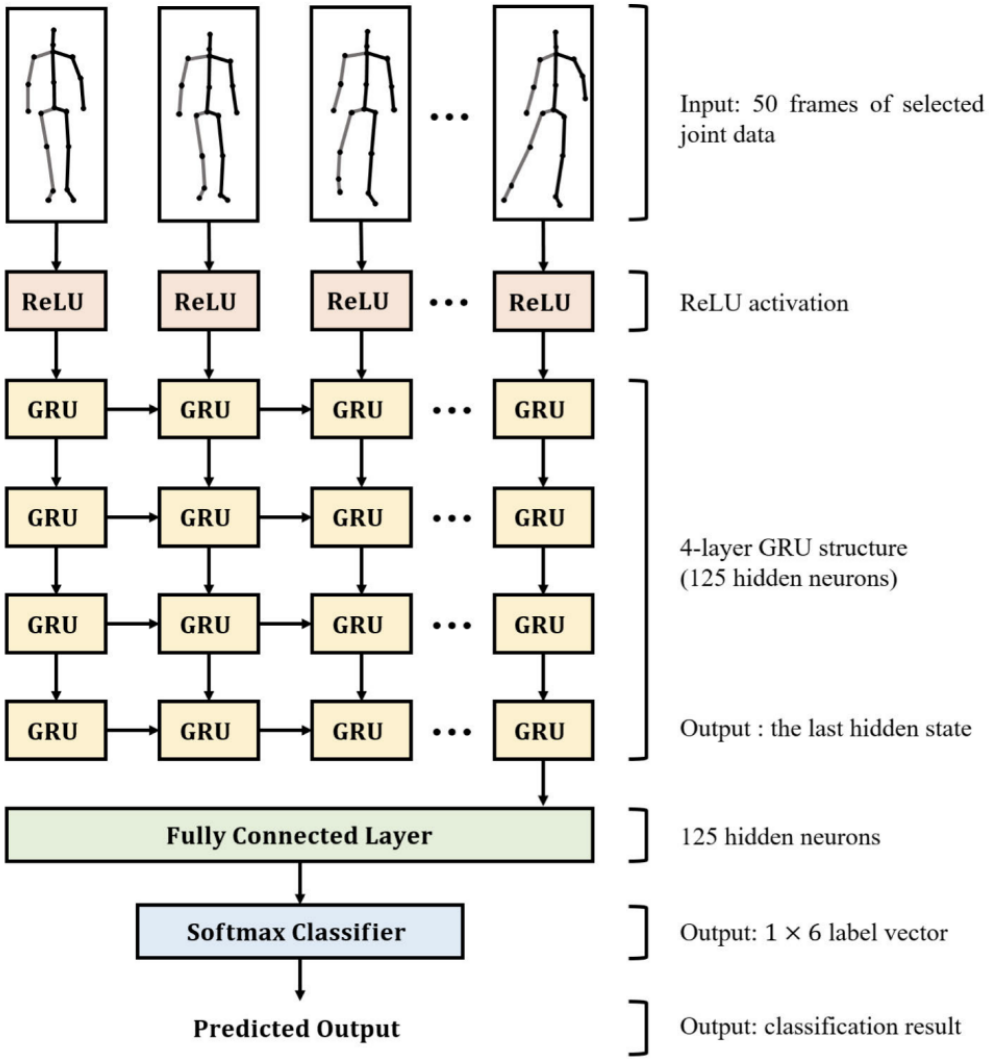


Trendelenburg Gait

- 1,440 data instances (12 people x 6 gait types x 20 walkings)

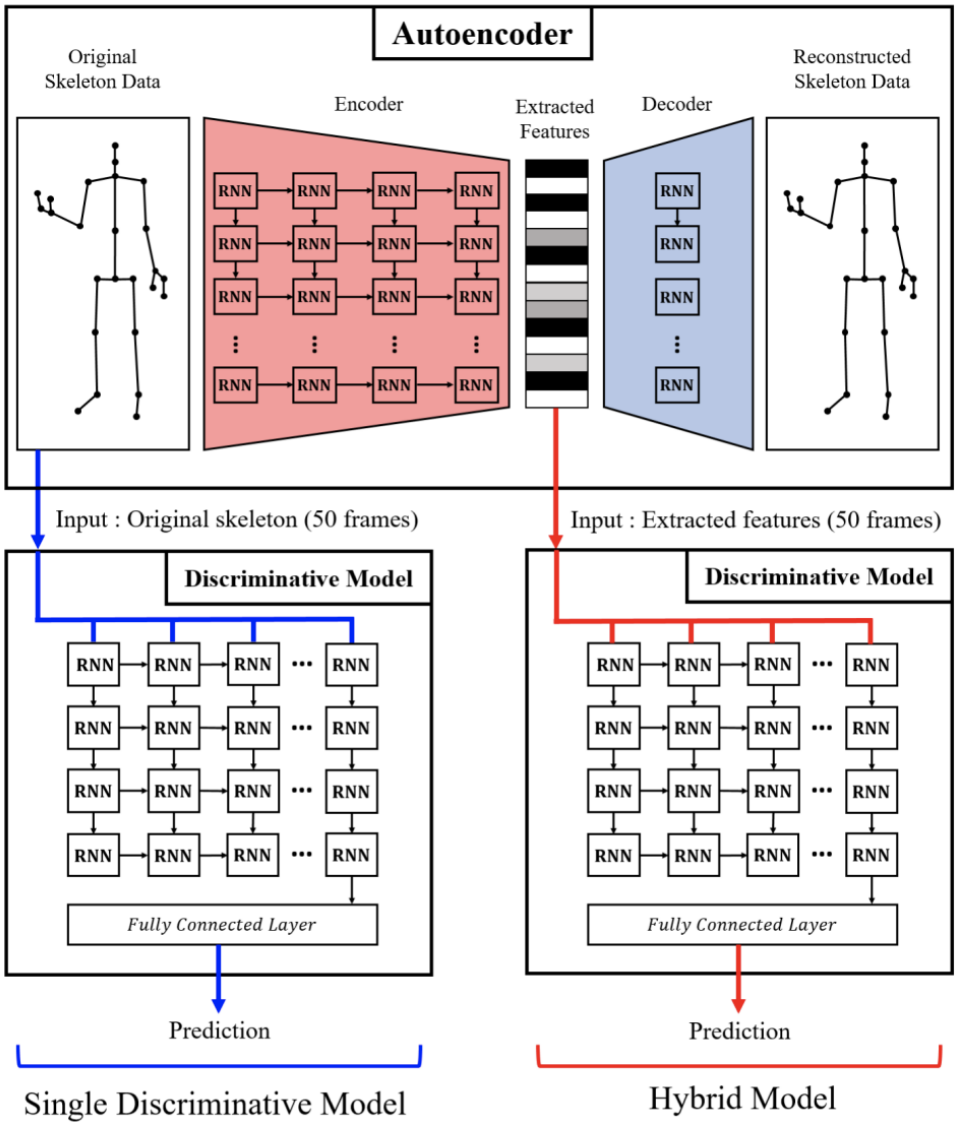
Approach in [Jun20_1]

- RNN with GRU or LSTM cells
- About 90% of accuracy
- Performance are evaluated considering sub-groups of joints



Approach in [Jun20_2]

- RNN with GRU or LSTM cells combined with an RNN autoencoder for feature extraction
- Improvement of about 5% with respect to the previous approach

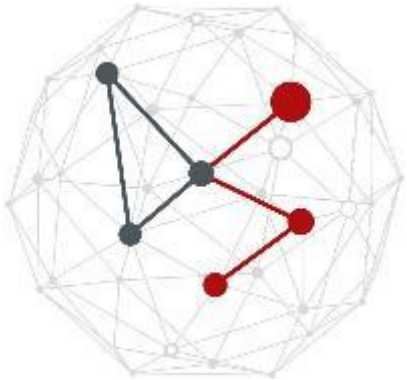


Possible project developments

- Consider the Sequential skeleton data
- Learning architectures – CNN, RNN…SNN
  - use raw signals with automatic feature extraction or
  - manually extract features or
  - combine the classification network with a preliminary automatic feature extraction network (e.g., autoencoder)
- Average foot pressure data never used by the authors…maybe they can be useful as a side information?
- REMEMBER: this is about sequential data…you cannot perform the classification by considering a single sample (common mistake)

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT A3



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project A3 “Motor imagery classification from EEG for brain computer interface”

Reference papers

[Kaya18] Kaya, M., et al. A large electroencephalographic motor imagery dataset for electroencephalographic brain computer interfaces, in Scientific data, vol. 5, no. 1, pp. 1–16, 2018.

[Altaheri23] H. Altaheri et al., Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: A review, Neural Computing and Applications, vol. 35, no. 20, pp. 14681–14722, 2023.

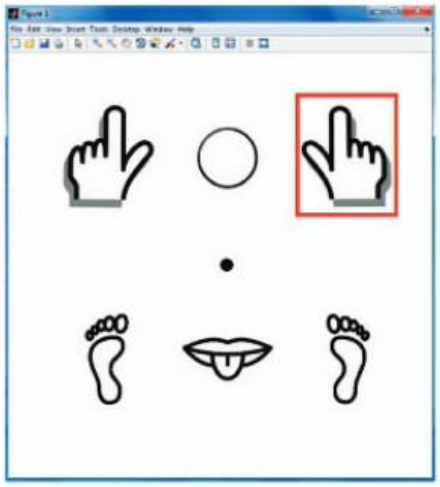
MI-BCI Dataset (5.3 GB uncompressed)

https://drive.google.com/file/d/1nUTwg3d8_IZKdkwSF5Bb3iYOt7DGbFj4/view?usp=sharing

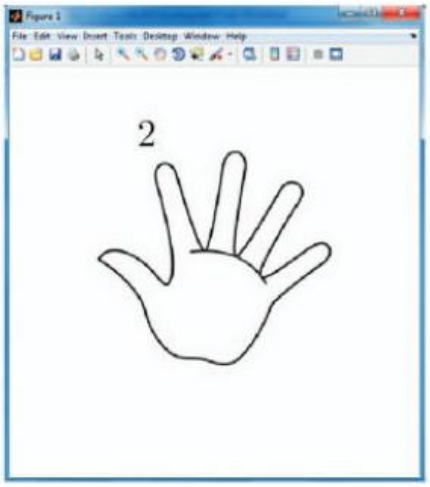
Dataset description

- Largest EEG BCI dataset publicly available
- 60h of EEG recordings from human subjects performing Motor Imagery (MI) tasks
- Sampling rate: 200 Hz and 1000Hz
- 21 active EEG leads
- 13 participants
- 3 different interaction types (paradigms)

a



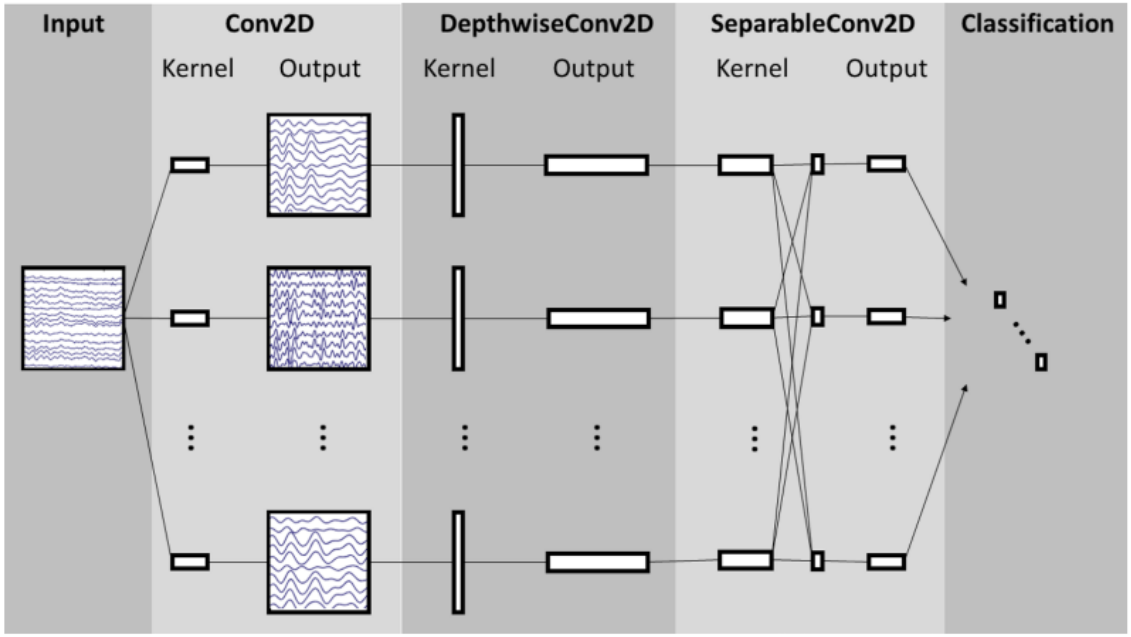
b



1. Left-Right Hand movement
2. Left-Right Hand+Leg+Tongue
3. Individual finger movement (1000Hz)

Approach in [Lawhern18]

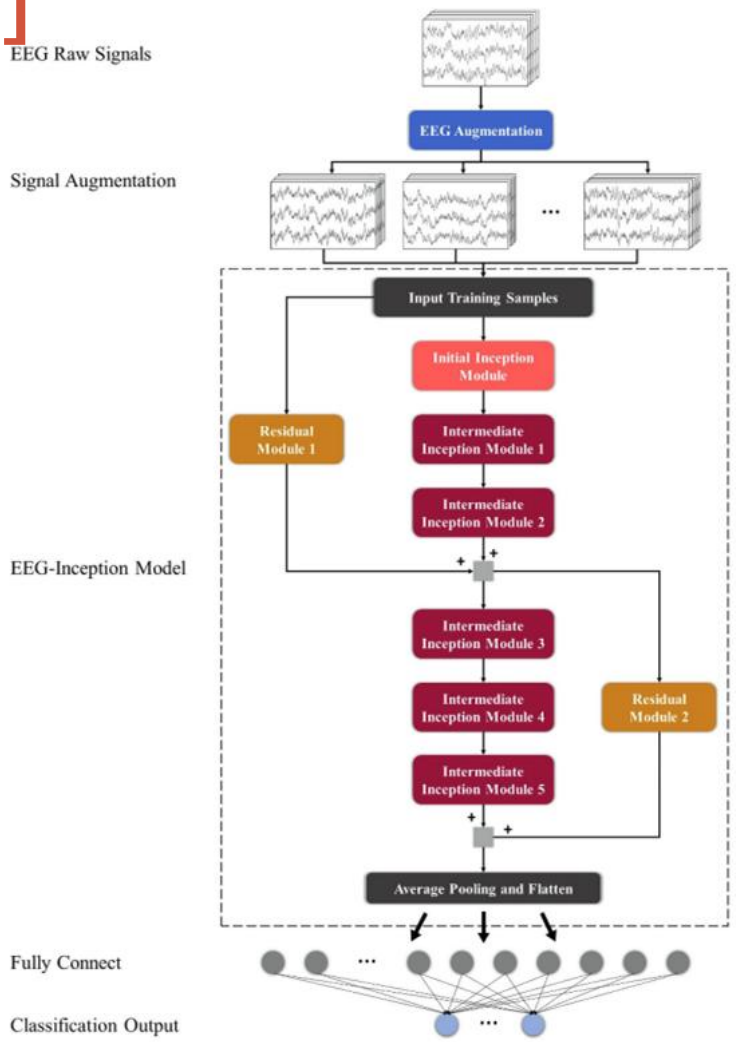
- Compact CNN architecture, one of the most established baseline models for EEG signals
- Exploits both temporal and spatial convolutions to learn frequency filters, and frequency-specific spatial filters



[Lawhern18] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces, Journal of neural engineering, vol. 15, no. 5, p. 056013, 2018.

Approach in [Zhang21]

- CNN architecture based on ensemble of multiple inception modules and residual modules
- Data augmentation is used to reduce risk of overfitting
- NB: Study is conducted on a different dataset, so the results are not directly comparable



[Zhang21] C. Zhang, Y.-K. Kim, and A. Eskandarian, EEG-inception: an accurate and robust end-to-end neural network for EEG-based motor imagery classification, Journal of Neural Engineering, vol. 18, no. 4, p. 046014, 2021.

Project proposal

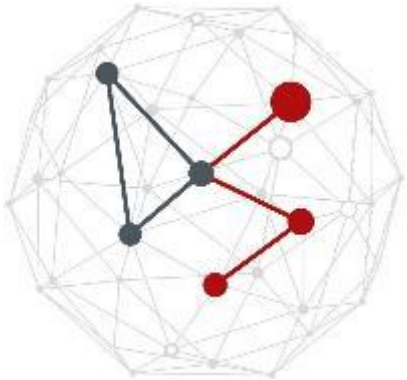
- Classify Motor Imageries (ordered in level of difficulty)
  - Left and Right hand movement, or passive (3 classes)
  - Left and Right hand/leg, tongue and passive (6 classes)
  - Five fingers movement (more challenging) (5 classes)

- Experiment with
  - Different pre-processing techniques (crucial for EEG data)
  - Different data representations (classic feature extraction, spectral representation, etc …)
  - Different architectures (novel, or combinations of existing ones)

MIME

Master's degree ICT Internet Multimedia Engineering

PROJECT A4



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project A4 “Arrhythmias detection from ECG recording”

Reference papers

[Acharya18] Acharya, U. R., Fujita, H., Oh, S. L., Raghavendra, U., Tan, J. H., Adam, M., Gertych, A., Hagiwara, Y., Automated identification of shockable and non-shockable life-threatening ventricular arrhythmias using convolutional neural network. Future Generation Computer Systems, Volume 79, Part 3, Pages 952-959 (2018)

[Ruiz25] Ruiz-Barroso, P., Castro, F. M., Miranda, J., Constantinescu, D., Atienza, D., Guil, N., FADE: Forecasting for anomaly detection on ECG. Computer Methods and Programs in Biomedicine, Volume 267 (2025)

Dataset (940 MB uncompressed)

https://www.kaggle.com/datasets/mondejar/mitbih-database/data

Description of the dataset

- The **MIT-BIH Arrhythmia** dataset includes 48 half-hour ECG recordings from 47 subjects (records 201 and 202 are from the same subject), sampled at 360 Hz
- Each ECG has 2 channels, obtained by placing electrodes on different parts of the chest
- 23 recordings (the “100 series”) are chosen at random from a collection of ECG recordings
- 25 recordings (the “200 series”) are selected to include examples of uncommon but clinically important arrhythmias
- Each ECG is stored as a **csv file** with 3 columns: the number of sample, and the raw value for the first and second channel

See https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=932724 for more information on the dataset

Description of the dataset

- For each ECG is also present an annotation file in .txt, which stores the time, number of sample, **beat annotations** and **rhythm annotations,** given by two cardiologists independently
- A description of the annotated symbols can be find at: https://physionet.org/physiobank/database/html/mitdbdir/intro.htm
- According to the AAMI (Association for the Advancement of Medical Instrumentation) standards, the **beat** annotations in the MIT-BIH dataset can be grouped into 5 classes:
    - **Normal beat (N)**
    - **Supraventricular ectopic beat (S)**
    - **Ventricular ectopic beat (V)**
    - **Fusion beat (F)**
    - **Unknown beat (Q)**

(see next slide)

Description of the dataset

Beat annotations

Table 1: ECG class description using AAMI standard.

AAMI class	MIT-BIH heart beat types				
Normal beat (N)	Normal beat (N)	Left bundle branch block beat (L)	Right bundle branch block beat (R)	Atrial escape beat (e)	Nodal (junctional) escape beat (j)
Supraventricular ectopic beat (S)	Atrial premature beat (A)	Aberrated atrial premature beat (a)	Nodal (junctional) premature beat (J)	Supraventricular premature beat (S)	
Ventricular ectopic beat (V)	Premature ventricular contraction (V)	Ventricular escape beat (E)			
Fusion beat (F)	Fusion of ventricular and normal beat (F)				
Unknown beat (Q)	Paced beat (/)	Fusion of paced and normal beat (f)	Unclassified beat (Q)		

Approach in [Acharya18]

- Task: automated differentiation of shockable and non-shockable ventricular arrhythmias

Table 2
ECG rhythm used.

Shockable VA rhythm	Non-shockable VA rhythm
Ventricular fibrillation	Normal sinus rhythm
Ventricular flutter	Ventricular bigeminy
Rapid ventricular tachycardia	Ventricular ectopic beats
	Ventricular escape rhythm

- ECG signals are pre-processed to **remove noise and baseline wander**, then are **standardized** using Z-scoring
- Synthetic data are generated to balance the dataset
- **Architecture: 4 convolutional layers**, each of them followed by **max-pooling** + **three fully connected layers**
- 10-fold cross validation is used

Approach in [Acharya18]

- **NOTE**: the dataset used here is a fusion of three different datasets, included the MIT-BIH Arrhythmia
- From the original ECG recordings, 2-second ECG fragments are extracted
- Rhythm annotations are considered for this type of task
- See table below for the conversion of MITDB (=MIT-BIH) rhythm annotations into shockable and non-shockable class

Table 3
ECG annotations used and the total number of segments obtained from each database.

Type	Used physionet annotations		Database
Non-shockable	B	Ventricular bigeminy	MITDB, VFDB
	N	Normal sinus rhythm	MITDB, VFDB, CUDB
	HGEA	High grade ventricular ectopic activity	VFDB
	NSR	Normal sinus rhythm	VFDB
	VER	Ventricular escape rhythm	VFDB
Shockable	VFL	Ventricular flutter	MITDB, VFDB
	VT	Ventricular tachycardia	MITDB, VFDB
	VF	Ventricular fibrillation	VFDB
	VFIB	Ventricular fibrillation	VFDB, CUDB

Approach in [Ruiz25]

- Task: normal ECG forecasting and anomaly detection

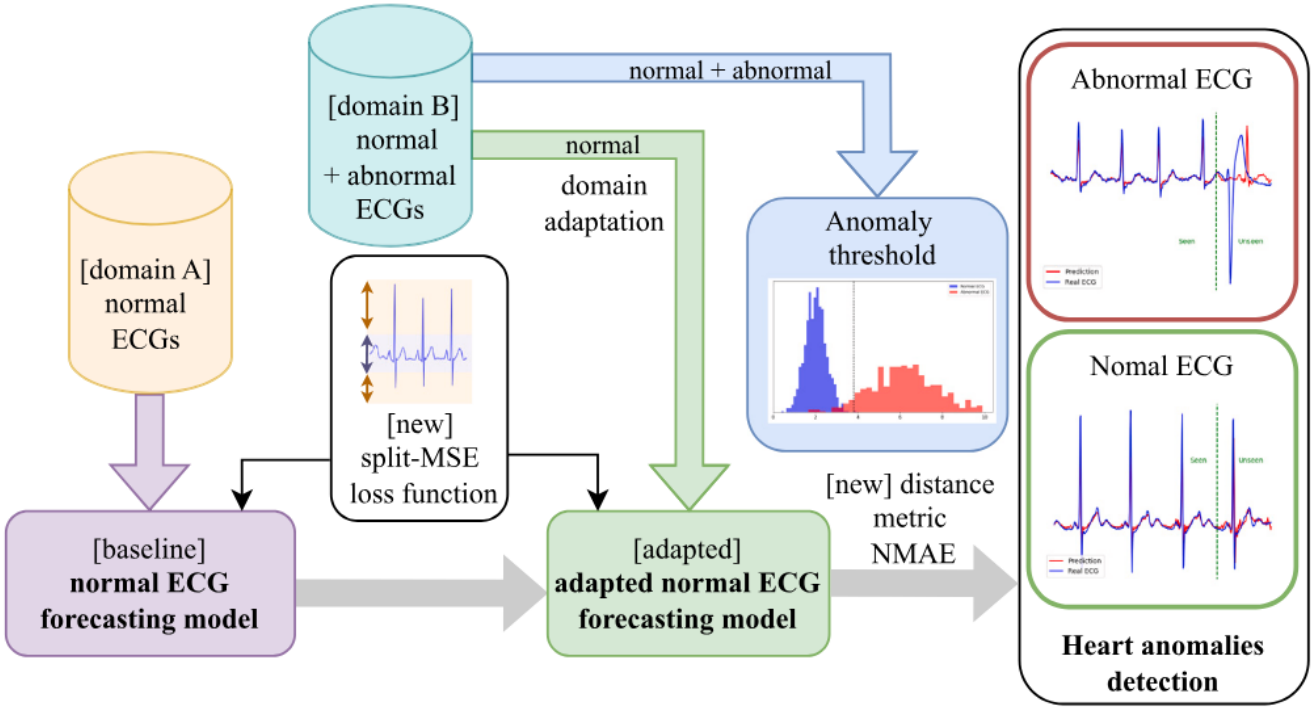


Fig. 1. Proposed System. General functionality of our proposed system for ECG forecasting with domain adaptation and heart anomaly detection.

Approach in [Ruiz25]

- Use an encoder-decoder model
- **Encoder**: codifies the ECG signal into a higher and richer dimensionality.
- It uses two paths of 4 Res-Net blocks to operate at both low (*slow* path) and high frequency (*fast* path)
- The output of the encoder is the concatenation of the outputs of both paths
- The **decoder** uses the encoded information to forecast the future ECG signal
- It follows the design of U-Net
- Use a **tailored loss function** called *Split-MSE* to train the model

Approach in [Ruiz25]

- **NOTE**: the dataset used here is a fusion of the MIT-BIH *Normal Sinus Rhythm* and the MIT-BIH *Arrhythmia* datasets

- The ECG recordings are split into windows of 4 s

- Forecast 1 s window of ECG

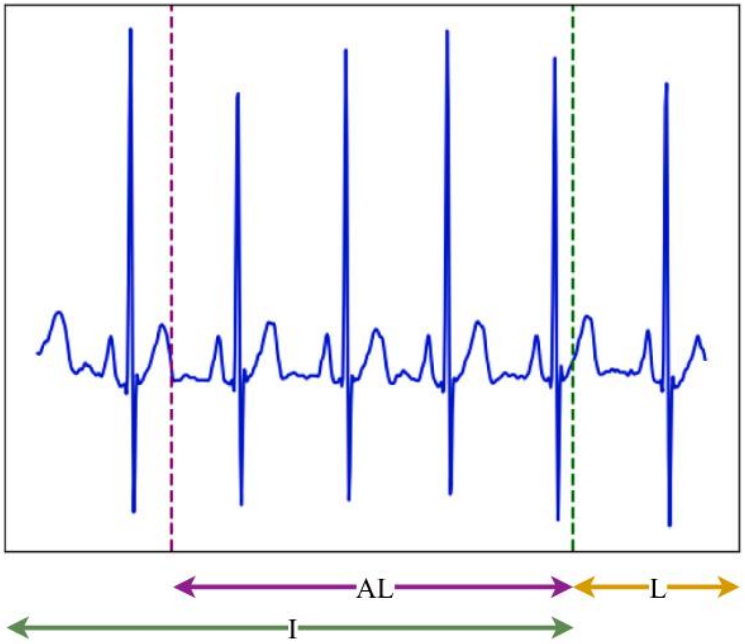


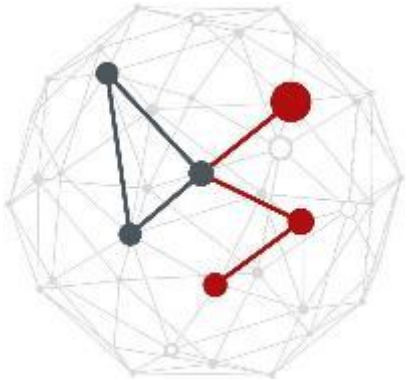
Fig. 2. ECG signal slicing. ECG slices for training the forecasting model: I represents the input data, L is the label data and AL shows the auxiliary label data.

Possible project developments

- Try different ECG pre-processing techniques
- Explore dimensionality reduction techniques
- Use spectral features
- Use one or both leads
- Solve different tasks:
  - Supervised: Arrhythmia classification (5 classes)
  - Supervised: Fatal vs. non-fatal arrhythmia (2 classes)
  - Unsupervised: Detect unusual or abnormal ECG segments (possible arrhythmias) without knowing labels (i.e., perform anomaly detection)
- Split the ECG taking into account the annotaitonsa and according to the task you choose
- Implement different architectures: Autoencoders, Attention modules, SNNs…

MIME
Master's degree ICT Internet Multimedia Engineering

PART B
AUDIO SIGNALS



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 1222·2022 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Proposed Projects

PART A – ON BODY AND ENVIRONMENTAL SENSORS

1) A1: Activity recognition with four accelerometers
2) A2: Pathological gait recognition
3) A3: Motor imagery classification from EEG for brain–computer interface
4) A4: Arrhythmias detection from ECG recording

PART B – AUDIO SIGNALS

1) B1: Speech command recognition (keyword spotting)
2) B2: Environmental sound classification
3) B3: Speaker identification/verification
4) B4: Digit recognition (Spiking Heidelberg Digit)

PART C – IMAGES

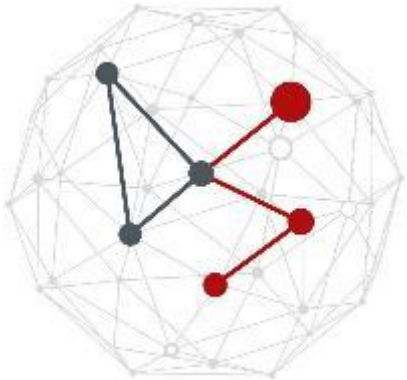
1) C1: Diabetic retinopathy detection
2) C2: Bone age prediction from hand radiographs
3) C3: Lung disease prediction from X-ray images
4) C4: Blood cell type prediction

PART D – RADIO SIGNALS

1) D1: Gesture recognition through radars

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT B1



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project B1 “Speech recognition”

Reference papers

[Sainath15] Tara N. Sainath, Carolina Parada, Convolutional Neural Networks for Small-footprint Keyword Spotting, INTERSPEECH, Dresden, Germany, September 2015.

[Warden18] Pete Warden, Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition, arXiv:1804.03209, April 2018.

https://arxiv.org/abs/1804.03209

- The authors are from Google Inc.
- Reference dataset released by Google [Warden18]

Dataset description

- Reference dataset for small-footprint keyword spotting (KWS)
  - Released in August 2017
  - 65,000 one-second-long utterances of 30 words
  - by thousands of different people
  - released under creative commons 4.0 license
  - collected by AIY (https://aiyprojects.withgoogle.com/)

Google blog

https://ai.googleblog.com/2017/08/launching-speech-commands-dataset.html

Speech dataset (2.11 GB uncompressed)

http://download.tensorflow.org/data/speech_commands_v0.02.tar.gz

Approaches for implementing a KWS engine

- **LVCSR based KWS** - This approach uses a two-stage process. In the first stage, the transcription of the speech into words is done using a Large Vocabulary Continuous Speech Recognition (LVCSR) engine, outputting formatted text. In the second stage, a textual search for the key-words within the text is performed. Using this approach, results from LVCSR and the text search are combined to spot the key-words
- **Phoneme Recognition based KWS** - This approach also uses a two-stage process. In the first stage, the speech is transformed to a sequence of phonemes. In the second stage, the application searches for phonetically transcribed key-words in the phoneme sequence obtained from the first stage
- **Word Recognition based KWS** [Sainath15] - This approach searches for the key-words in a one stage operation. The recognition is phoneme-based and the KWS engine looks for the keyword in the speech stream based on a target sequence of phonemes representing the key-word

CNN model from [Sainath15]

- Features are obtained from raw audio data
- 40-dimensional log Mel filterbanks coefficients
  - audio frame length 25 ms
  - with a 10 ms time shift
- At every new audio frame
  - Feature vector is obtained
  - And stacked with 23 frames to the left and 8 to the right (32 frames total)
  - This returns 32 frames at a time, spanning over 31 x 10 ms + 25 ms = 0.335 s
- A Convolutional Neural Network (CNN) is used to detect words
- Input to the CNN is a matrix of size t x n = 32 x 40 = 1,280 elements
  - t represents the number of elements in time (number of audio frames)
  - n represents the number of elements in the frequency domain (Mel features)

CNN model from [Sainath15]

- 27-44% improvement for KWS with respect to traditional neural networks

- The paper focus is on
  - Devising CNN architectures with small memory footprint
  - Playing with CNN parameters (number of kernels, strides, pooling, etc.)

Possible project developments

- Experiment with different audio features
  - Type of coefficients (e.g., discrete Wavelet transform)
  - Design of Mel filterbanks
- Play with a standard/deep CNN using
  - dropout, regularization
- Investigate recent/new ANN architectures
  - Autoencoder-based (CNN/RNN autoencoder + following SVM)
  - Attention mechanism and/or inception-based CNN networks
  - Comparison of different architectures: memory *vs* accuracy

Useful resources

Recent developments

[Chorowski15] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio, Attention-Based Models for Speech Recognition, Conference on Neural Information and Processing Systems (NIPS), Montréal, Canada, 2015.

[Tang18] R. Tang and J. Lin, Deep residual learning for small-footprint keyword spotting, in IEEE ICASSP, Calgary, Alberta, Canada, 2018.

[Andrade18] D. C. de Andrade, S. Leo, M. L. D. S. Viana, and C. Bernkopf, A neural attention model for speech command recognition, arXiv:1808.08929, 2018. https://arxiv.org/pdf/1808.08929.pdf

White Paper: “Key-Word Spotting - The Base Technology for Speech Analytics”

https://pdfs.semanticscholar.org/e736/bc0a0cf1f2d867283343faf63211aef8a10c.pdf

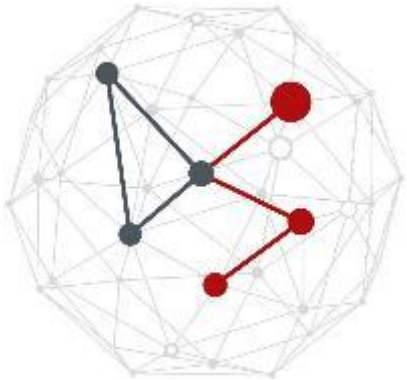
Example code:

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/speech_commands/

MIME

Master's degree ICT Internet Multimedia Engineering

PROJECT B2



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project B2 “Environmental sound classification”

Reference papers

[Piczak15] K.J. Piczak, ESC: Dataset for Environmental Sound Classification, in Proceedings of the 23rd ACM International Conference on Multimedia, Brisbane, Australia, 2015.

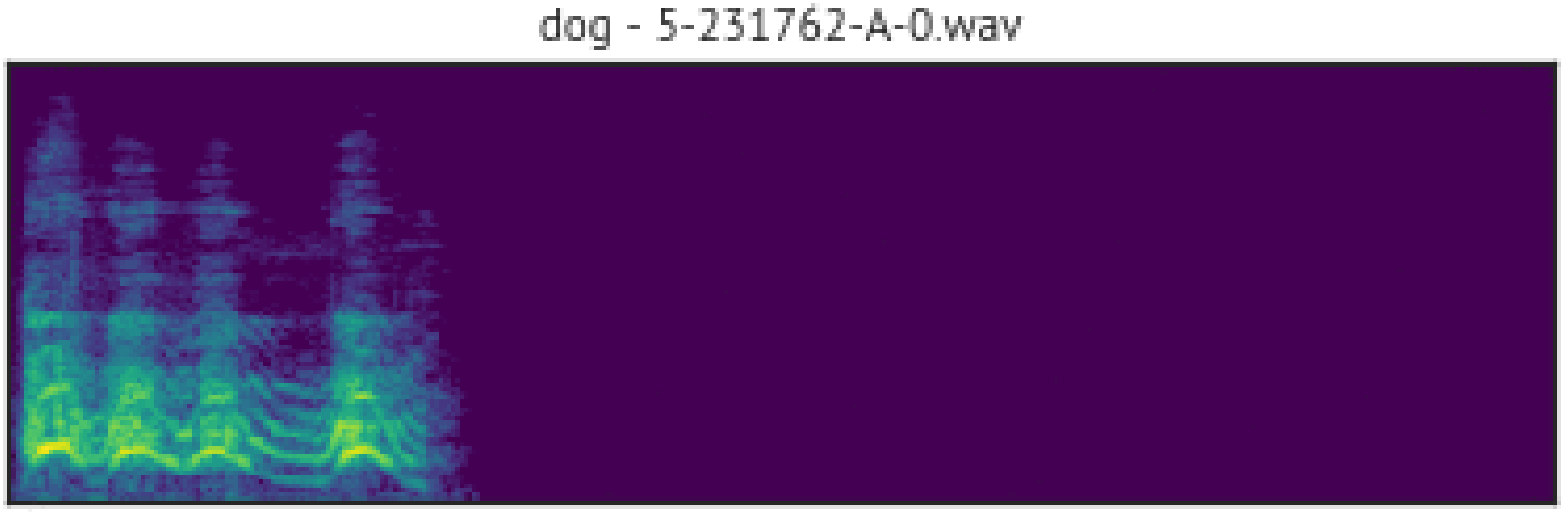
[Piczak15-1] K. J. Piczak, Environmental sound classification with convolutional neural networks, in Proceedings of the IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), Boston, MA, 2015.

ECS-50 dataset (884 MB uncompressed)

- https://github.com/karolpiczak/ESC-50
- Annotated collection of 2000 short clips comprising 50 classes of various common sound events

High level description of the dataset

- 5-second-long clips, 44.1 kHz, single channel
- Arranged into 5 uniformly sized cross-validation folds, ensuring that clips originating from the same initial source file are always contained in a single fold



High level description of the dataset

- 50 classes in the dataset

Animals	Natural soundscapes & water sounds	Human, non-speech sounds	Interior/domestic sounds	Exterior/urban noises
Dog	Rain	Crying baby	Door knock	Helicopter
Rooster	Sea waves	Sneezing	Mouse click	Chainsaw
Pig	Crackling fire	Clapping	Keyboard typing	Siren
Cow	Crickets	Breathing	Door, wood creaks	Car horn
Frog	Chirping birds	Coughing	Can opening	Engine
Cat	Water drops	Footsteps	Washing machine	Train
Hen	Wind	Laughing	Vacuum cleaner	Church bells
Insects (flying)	Pouring water	Brushing teeth	Clock alarm	Airplane
Sheep	Toilet flush	Snoring	Clock tick	Fireworks
Crow	Thunderstorm	Drinking, sipping	Glass breaking	Hand saw

High level description of the dataset

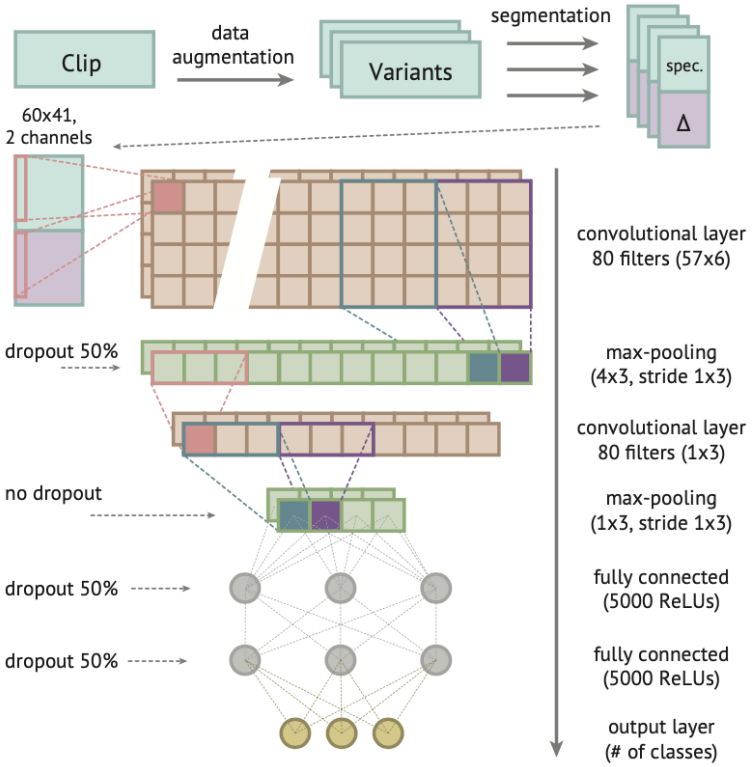
- ESC-10: selection of 10 classes from the bigger dataset
  - The differences between classes are much more pronounced, with limited ambiguity
  - Classes: *sneezing, dog barking, clock ticking, crying baby, crowing rooster, rain, sea waves, fire crackling, helicopter, chainsaw*

- meta/esc50.csv data description, the “esc10” column indicates if a given file belongs to the *ESC-10* subset

- meta/esc50-human.xlsx contains the human classification accuracy

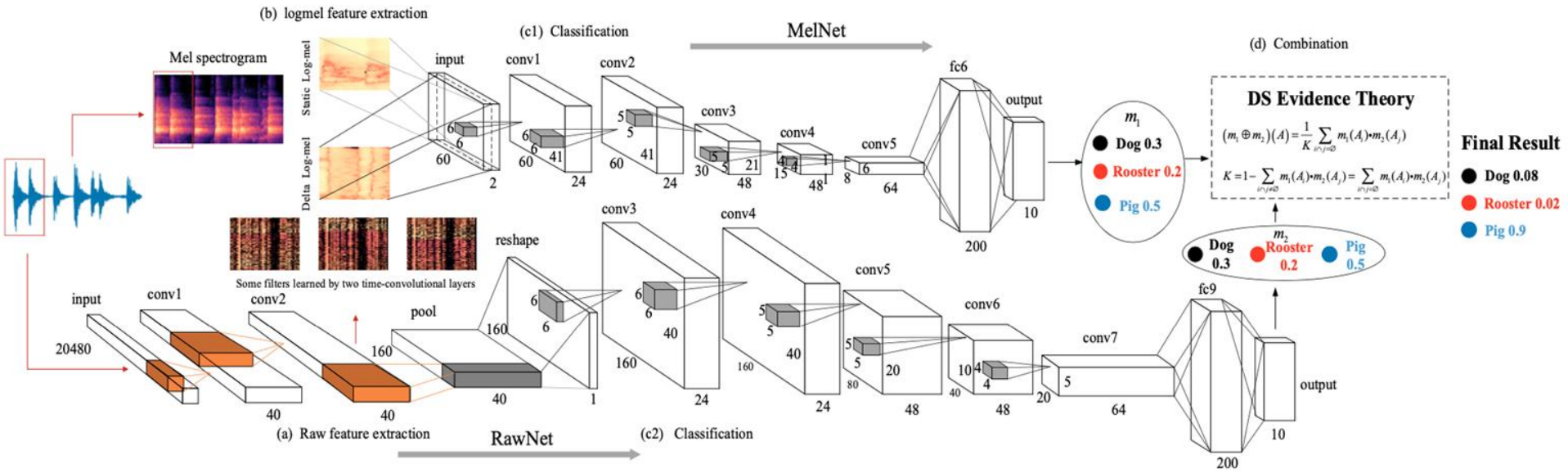
Approach in [Piczak15-1]

- Data augmentation: apply random time delays to the original recordings
- Feature extraction: log-scaled mel-spectrograms with 60 mel-bands
  - resampled to 22,050 Hz
  - windows size 1024
  - hop length 512
- Learning architecture: CNN



Other reference [Li18]

- Combines mel-spectrogram features and raw audio waveform



[Li18] S. Li, Y. Yao, J. Hu, G. Liu, X. Yao and J. Hu, An Ensemble Stacked Convolutional Neural Network Model for Environmental Event Sound Recognition, Applied Science, vol. 8, no. 1152, July 2018.

Useful links

- Some useful functions https://nbviewer.jupyter.org/github/karoldvl/paper-2015-esc-dataset/blob/master/Notebook/ESC-Dataset-for-Environmental-Sound-Classification.ipynb

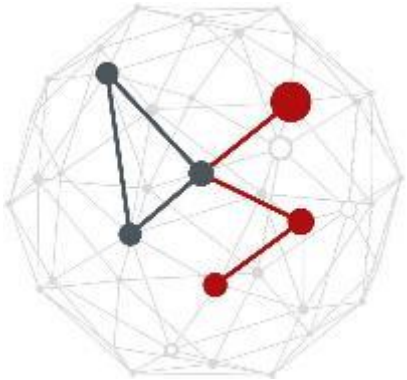
Possible project developments

- Classification tasks
  - on the entire ESC-50 dataset
  - on the restricted ESC-10 dataset
  - on each of the 5 groups of sounds:
    - animals
    - natural soundscapes & water sounds
    - human, non-speech sounds
    - interior/domestic sounds
    - exterior/urban noises
- Features: try with different approaches: mel-spectrogram, other manual-extracted features, raw data, combinations
- Architectures
  - different possibilities: CNN, RNN, …

MIME

Master's degree ICT Internet Multimedia Engineering

PROJECT B3



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project B3 “Speaker identification”

Reference papers

[Nagrani18] Nagrani, A., Chung, J. S., Zisserman, A., Voxceleb: a large-scale speaker identification dataset. arXiv: 1706.08612 (2018)

[Ollerenshaw23] Ollerenshaw, A., Jalal, M. A., Hain, T., Dynamic Kernels and Channel Attention for Low Resource Speaker Verification. arXviv: 2211.02000 (2023)

Voxceleb1 dataset (32.6 GB compressed)

https://huggingface.co/datasets/ProgramComputer/voxceleb/tree/main/vox1

Download the following folders:

- vox1_dev_wav.zip
- vox1_test_wav.zip
- vox1_meta.csv

Speaker identification vs. verification

- Speaker identification and speaker verification are both techniques used in the realm of voice and speech analysis, but they serve distinct purposes in the context of recognizing and authenticating individuals based on their voices.
- Speaker identification involves determining the identity of an unknown speaker by comparing its voice to a set of known speakers.
- Speaker verification focuses on confirming or denying the claimed identity of a speaker. It is a one-to-one comparison between the voice of an individual claiming a particular identity and a stored reference voice sample of that same individual.

Dataset split for speaker identification and verification tasks

[1] Useful link: https://www.robots.ox.ac.uk/~vgg/data/voxceleb/vox1.html

- Verification split: this is the default split when you download the data. For this task, *dev* and *test* set are **disjoint** (i.e., the two sets contain different celebrities). To test your architecture, you can use the "List of trial pairs - VoxCeleb1 (cleaned)" txt file available in [1].
- Identification split: for this task, you can merge the original *dev* and *test* sets, and then **perform a custom split**. Note that for speaker identification, the *dev* and *test* sets **must not be disjoint** (i.e., the two sets should contain different audio extracts from **the same** speakers). Alternatively, you can use the "Dataset split for identification" txt file available in [1].

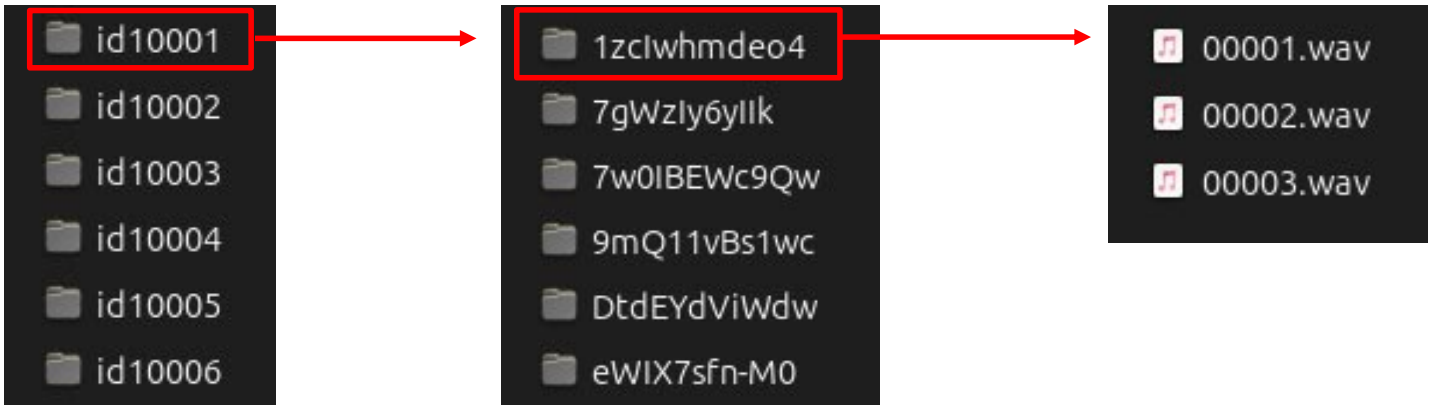
  Note: In this file, the **first column** specifies the dataset split: 1 for training set, 2 for validation set, and 3 for test set.

Description of the dataset

- VoxCeleb1 contains over **150,000 utterances** from over **1251 celebrities**, with an average utterance length of 8.2 s
- It is **fairly gender balanced** (55% male)
- The speakers span a wide range of different ethnicities, accents, professions, and ages
- It is built for speaker recognition under **noisy** and **unconstrained conditions**
- The speech segments are corrupted with **real-world noise**, including laughter, cross-talk, channel effects, music,…
- The dataset is divided into **development** and **test** set

Description of the dataset

- Structure of the vox1_dev_wav folder (similar to vox1_test_wav):
  1. There is a folder for each speaker, named with a unique speaker ID
  2. Inside each speaker folder there are several subfolders, each corresponding to a different Youtube video source
  3. Each subfolder contains different .wav files (individual speech segments extracted from that video)



- The vox1_meta.csv file contains the mapping speaker_id ⟷ celebrity, plus other useful information

VoxCeleb1 ID	VGGFace1 ID	Gender	Nationality	Set
id10001	A.J._Buckley	m	Ireland	dev
id10002	A.R._Rahman	m	India	dev
id10003	Aamir_Khan	m	India	dev
id10004	Aaron_Tveit	m	USA	dev
id10005	Aaron_Yoo	m	USA	dev
id10006	Abbie_Cornish	f	Australia	dev

Approach in [Nagrani18]

- All audio are first converted to single-channel, **16-bit streams** at a **16 kHZ sampling rate**
- Spectrograms are generated using a **Hamming window** of width 25 ms and step 10 ms
- Mean and variance normalization is applied on every frequency bin of the spectrum
- To evaluate the **identification performance** the authors created a held-out validation test consisting of all the speech segments from a single video for each identity

Approach in [Nagrani18]

- Use a modified version of the VGG-M CNN architecture for both speaker identification and verification
- Use a variable average pooling layer to accomodate to variable length input at test time
- Identification results: **80.5% top-1** and 92.1% top-5 accuracy
- Verification resutls: **7.8% EER** (equal error rate)

Layer	Support	Filt dim.	# filts.	Stride	Data size
conv1	7×7	1	96	2×2	254×148
mpool1	3×3	-	-	2×2	126×73
conv2	5×5	96	256	2×2	62×36
mpool2	3×3	-	-	2×2	30×17
conv3	3×3	256	384	1×1	30×17
conv4	3×3	384	256	1×1	30×17
conv5	3×3	256	256	1×1	30×17
mpool5	5×3	-	-	3×2	9×8
fc6	9×1	256	4096	1×1	1×8
apool6	1×n	-	-	1×1	1×1
fc7	1×1	4096	1024	1×1	1×1
fc8	1×1	1024	1251	1×1	1×1

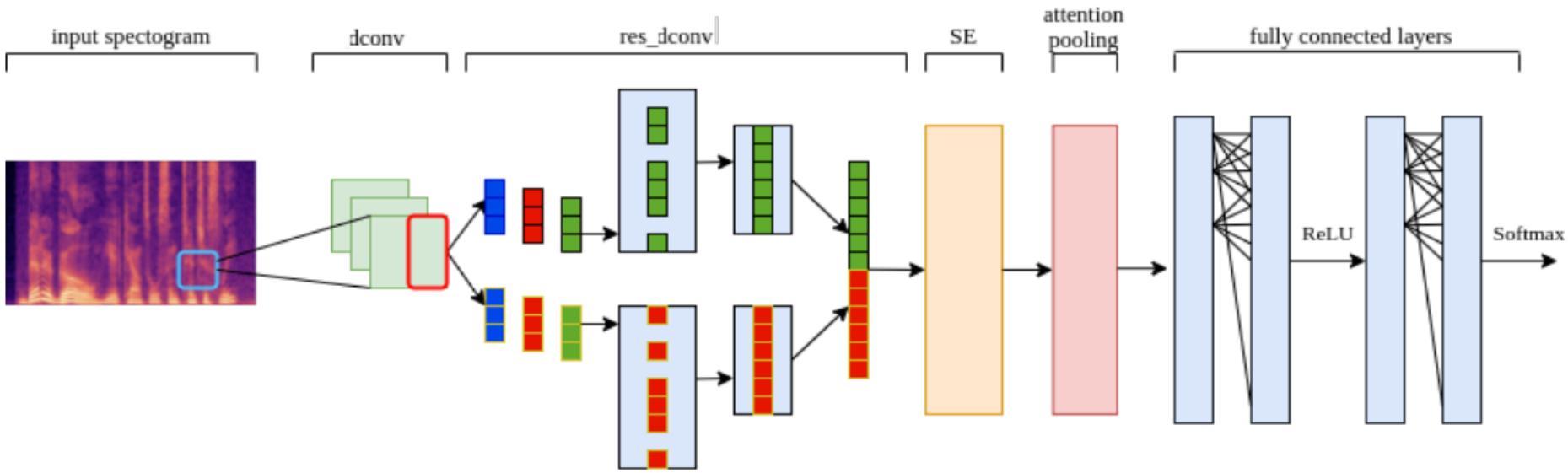
Table 4: *CNN architecture. The data size up to fc6 is for a 3-second input, but the network is able to accept inputs of variable lengths.*

Approach in [Ollerenshaw23]

- Frequency and Time domain augmentation using “Specaugment”, a data technique specifically used for speech.
- Random augmentation chosen among reverberation, babble, music and noise was applied directly to the input audio.
- Reverberation was convoluted with the original speech audio using RIRs (Room Impulse Responses) from a RIR generator.

Approach in [Ollerenshaw23]

- The paper uses dynamic kernel convolutions instead of static one.
- The input features are chunked and fed into hierarchical layers that use skip connections.
- The representation are then concatenated and passed through a channel attention based on the Squeeze-and-Excitation mechanism.

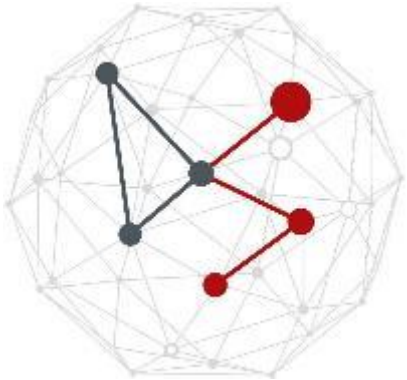


Possible project developments

- Task: identification and/or verification
- Try different features: mel-spectrogram, other manual-extracted features, raw data, combinations
- Different architectures: RNN, GNN,…

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT B4



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project B4 “Digit recognition (SHD)”

Reference papers

[Hammouamri23] Hammouamri, I., Khalfaoui-Hassani, I., & Masquelier, T. Learning delays in spiking neural networks using dilated convolutions with learnable spacings. arXiv preprint arXiv:2306.17670 (2023)

[Yao21] Yao, M., Gao, H., Zhao, G., Wang, D., Lin, Y., Yang, Z., Li, G., Temporal-wise Attention Spiking Neural Networks for Event Streams Classification. arXviv preprint arXiv:2107.11711 (2021)

SHD dataset (331 MB uncompressed)

https://zenkelab.org/datasets/

Description of the dataset

- **Purpose:** benchmark classification dataset for spiking neural networks (SNNs) research
- Speech recordings of spoken digits (0–9), derived from the **Heidelberg Digits** dataset
  - ~**10,000** high-quality aligned studio recordings of spoken digits in both **German** and **English** language ⟶ **20 classes**
  - **12 distinct speakers** (two of which are only present in the test set)
  - Train / test split already provided
- Each audio file is **converted into spike trains** using a **cochlear (auditory) model**, simulating biological hearing
- The dataset is released under Creative Commons Attribution 4.0 International License

Approach in [Hammouamri23]

- The delay of a connection is the delay between spike emission and reception
- **Idea:** learn synaptic weights and delays jointly in a **deep SNN** with LIF neurons
- **How?** Model the synaptic connection between neuron *j* in layer *L-1* and neuron *i* in layer *L* with a one dimensional temporal convolution with kernel k_{ij}^{L}
- The kernels contain only a few non-zero weights whose positions correspond to the delays
- Learn the kernel element positions (i.e., delays) with 1D dilated convolutions with learnable spacings (DCLS) with a Gaussian kernel

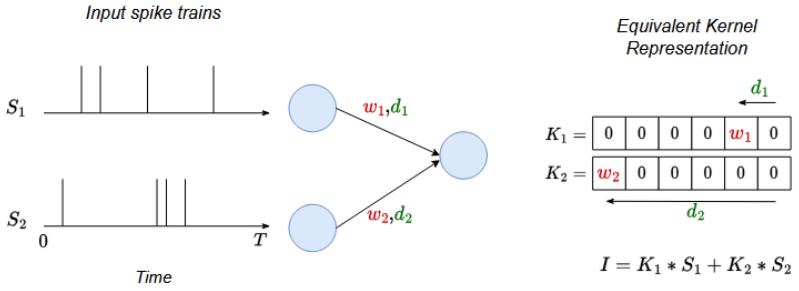
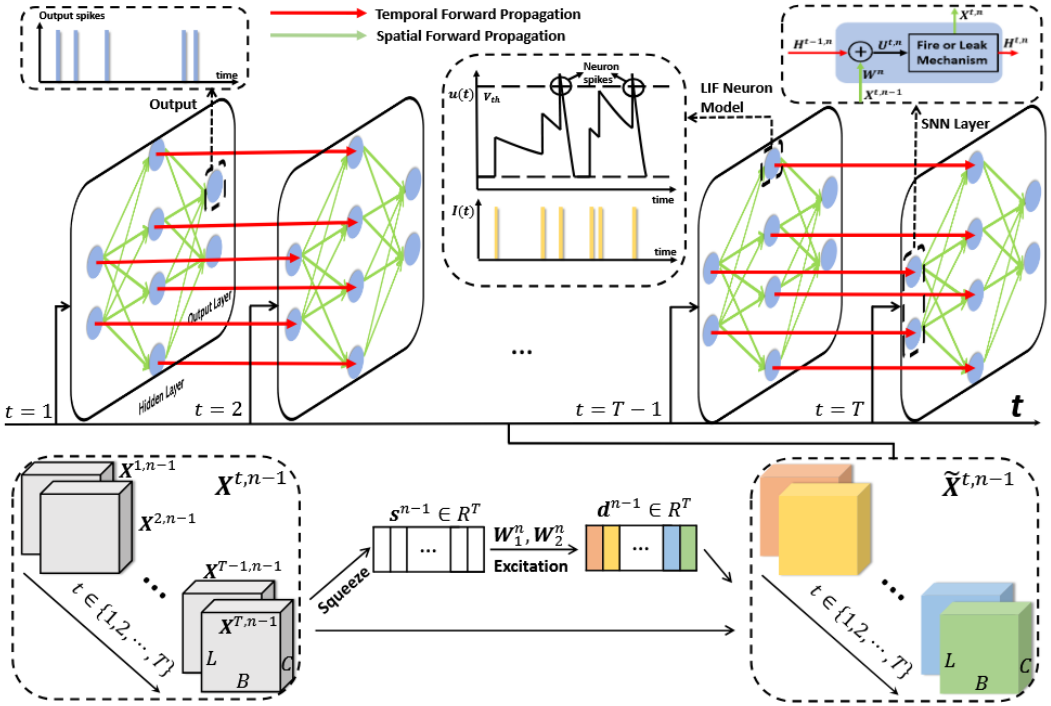


Figure 2: Example of one neuron with 2 afferent synaptic connections, convolving $K1$ and $K2$ with the zero left-padded S_1 and S_2 is equivalent to following Equation 6

Approach in [Yao21]

- Temporal-wise attention SNN (TA-SNN) model to learn frame-based representation
- Use squeeze and excitation steps
- Investigate different positions where to insert the TA module (no TA, only at input layer, all depth layers but the input…)

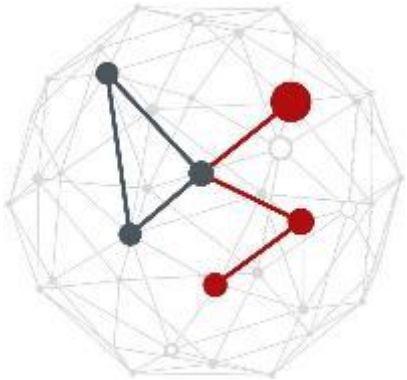


Possible project developments

- Try different spiking neuron models
- Experiment with different surrogate gradient functions
- Different architectures: recurrent SNNs, recurrent + convolutional SNNs, attention module, spiking autoencoders..
- Experiment with energy efficient network (i.e., add a regularization term to the loss…)

MIME
Master's degree ICT Internet Multimedia Engineering

PART C
VISUAL DATASETS



DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO
MATEMATICA

800 1222·2022 ANNI

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Proposed Projects

PART A – ON BODY AND ENVIRONMENTAL SENSORS

1) A1: Activity recognition with four accelerometers
2) A2: Pathological gait recognition
3) A3: Motor imagery classification from EEG for brain–computer interface
4) A4: Arrhythmias detection from ECG recording

PART B – AUDIO SIGNALS

1) B1: Speech command recognition (keyword spotting)
2) B2: Environmental sound classification
3) B3: Speaker identification/verification
4) B4: Digit recognition (Spiking Heidelberg Digit)

PART C – IMAGES

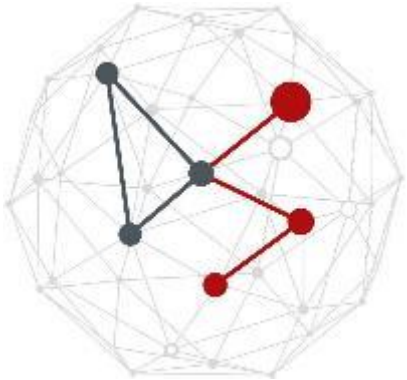
1) C1: Diabetic retinopathy detection
2) C2: Bone age prediction from hand radiographs
3) C3: Lung disease prediction from X-ray images
4) C4: Blood cell type prediction

PART D – RADIO SIGNALS

1) D1: Gesture recognition through radars

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT C1



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 1222·2022 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project C1 “Diabetic retinopathy detection”

Reference papers

[Mary24] Mary, A., Kavitha, P., Diabetic retinopathy disease detection using shapley additive ensembled densenet-121 resnet-50 model. *Multimed Tools Appl* 83, 69797–69824 (2024).

[Bala24] Bala, R., Sharma, A. & Goel, N. CTNet: convolutional transformer network for diabetic retinopathy classification. Neural Comput & Applic 36, 4787–4809 (2024).

Dataset (9.5 GB compressed)

https://www.kaggle.com/competitions/aptos2019-blindness-detection/data

IMPORTANT NOTE ON THE NEXT SLIDE

APTOS Dataset

- Note 1: the test set does not include the labels, as the dataset has been used for a challenge. For this reason, download just the train set and use it as if it was the whole dataset.

- Note 2: you are required to **accept the competion rule**, in order to download the dataset.

High level description of the dataset

- *Diabetic retinopathy* (DR) is an eye condition that can cause vision loss and blindness in people who have diabetes.
- The APTOS dataset comprises 3,662 **retina images** taken using fundus photography, with variable resolution.
- A clinician has rated each image for the **severity of DR on a scale from 0 to 4**:
  - 0 – No DR
  - 1 – Mild
  - 2 – Moderate
  - 3 – Severe
  - 4 – Proliferative DR (PDR)

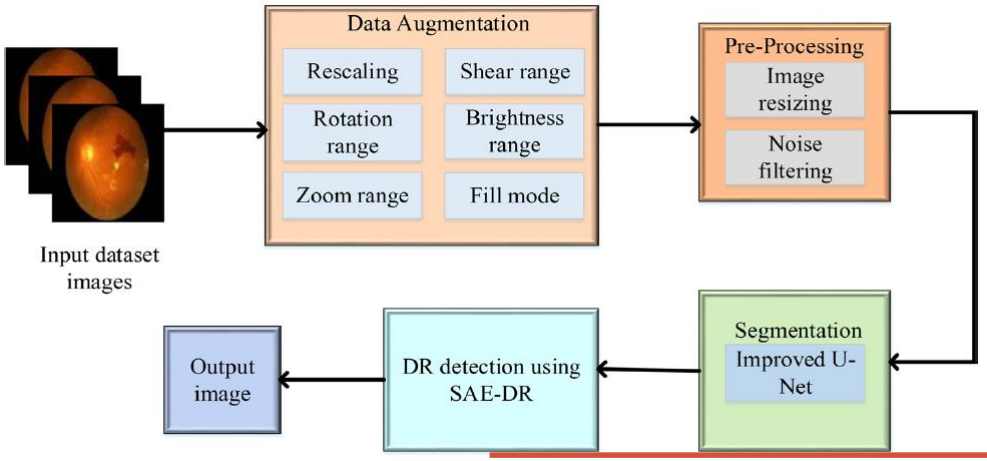
High level description of the dataset

- The dataset folder contains a subfolder which stores the images and a csv file containing the images IDs and labels

Fig.: Example retina images with associated DR severity.

Approach in [Mary24]

- Task: DR severity classification
- Data augmentation
- Pre-processing: image resizing and noise filtering
- Image segmentation using an updated **U-Net** model
- Use a weighted averaged based ensemble model to combine **DenseNet-121** and **ResNet-50** models and produce the final output



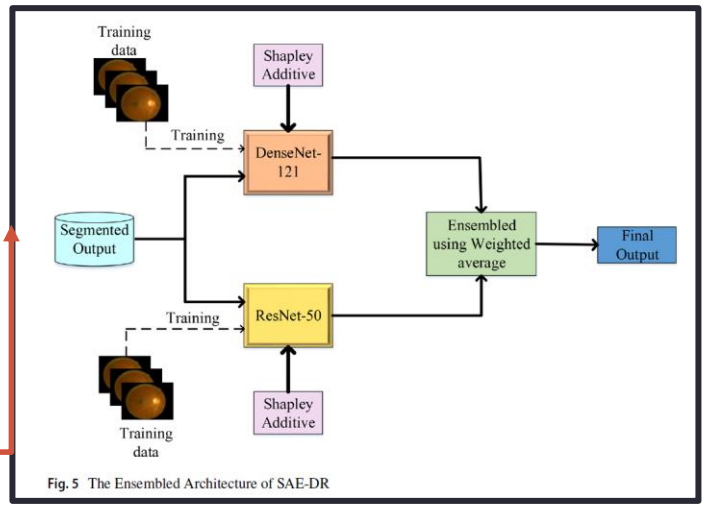


Fig. 5 The Ensembled Architecture of SAE-DR

Approach in [Bala24]

- Lightweight DR classification model
- Proper data augmentation
- Pre-processing: image cropping and resizing, gaussian blur
- Learning architecture: **CTNet** – combines CNN and Transformers

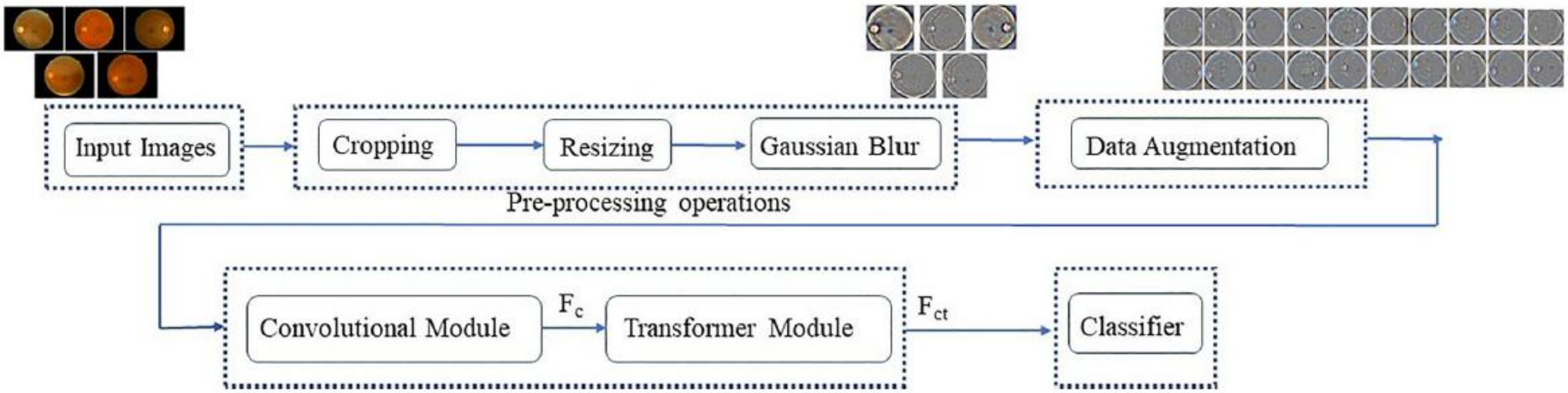


Fig. 3 Block diagram of the proposed model

Approach in [Bala24]

- The convolutional module (CM) extracts local spatial features which are patchified into small patches

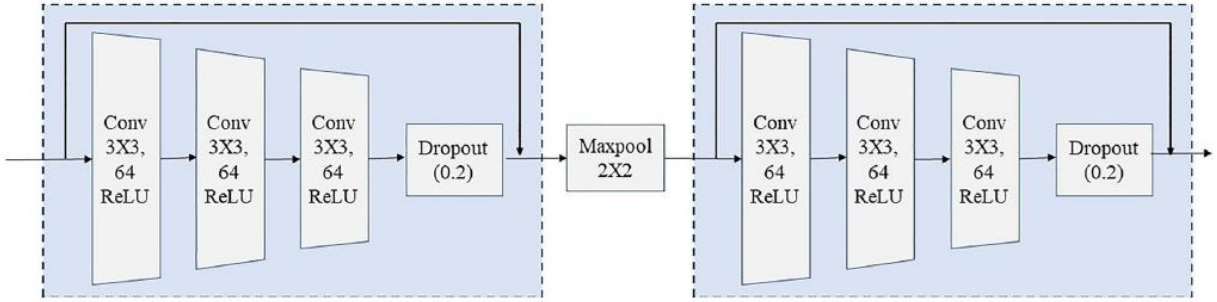


Fig. 4 Convolution Module (CM)

- The transformer module (TM) analyzes these patches extracting global contextual information

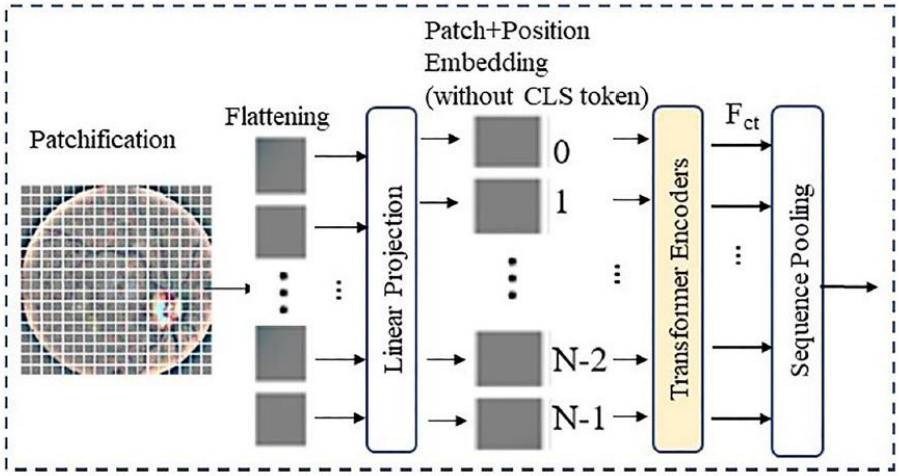


Fig. 5 Transformer Module (TM)

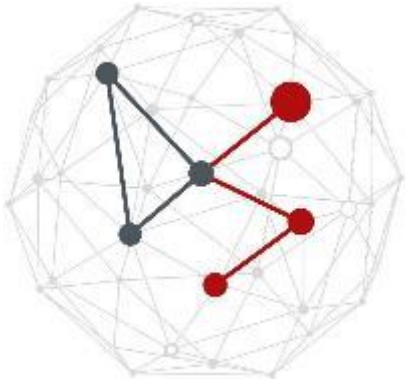
Possible project developments

- Explore different data augmentation and pre-processing techniques
- Combine neural networks for automatic feature extraction and standard ML classifiers
- Learning Architecture: GNNs (or GCNs), convolutional SNNs, autoencoders, attention module…

MIME

Master's degree ICT Internet Multimedia Engineering

PROJECT C2



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project C2 “Bone age prediction from hand radiographs”

Reference papers

[Larson18] D. B. Larson, M. C. Chen, M. P. Lungren, S. S. Halabi, N. V. Stence, C. P. Langlotz, Performance of a Deep-learning neural network Model in assessing skeletal Maturity on Pediatric hand radiographs, Radiology, vol. 287, no. 1, pp. 313-322, April 2018.

[Halabi19] S. S. Halabi *et al.*, The RSNA Pediatric Bone Age Machine Learning Challenge, Radiology, vol. 290, pp. 498-503, 2019.

https://www.rsna.org/rsnai/ai-image-challenge/rsna-pediatric-bone-age-challenge-2017

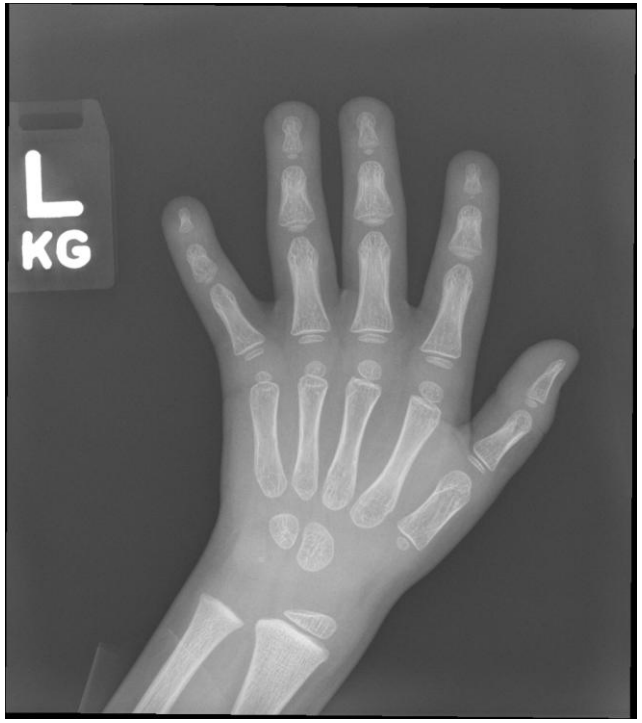
Papers available at: https://drive.google.com/drive/folders/1E3-qofGlzTAe8oN7gUrmsqteObNBd6HT?usp=sharing

Dataset (10.3 GB uncompressed)

https://stanfordmedicine.app.box.com/s/4r1zwio6z6lrzk7zw3fro7ql5mnoupcv/folder/42459416739

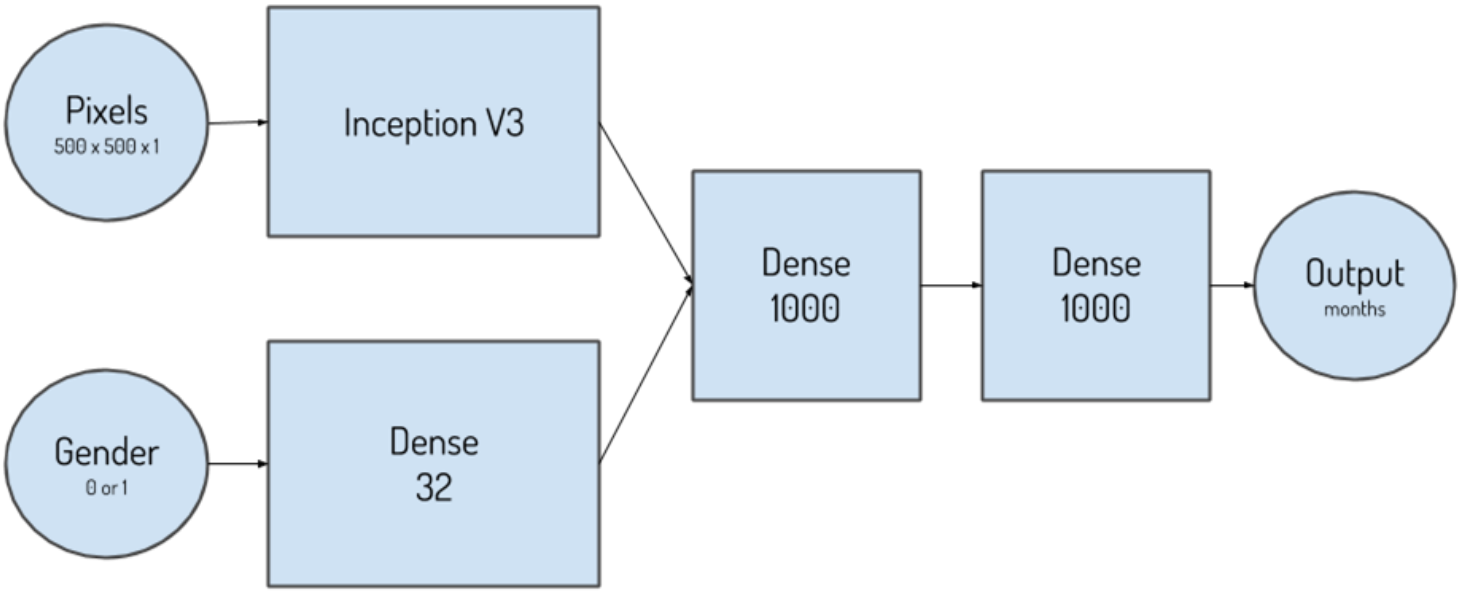
Dataset description

- 12,612 training hands' X-ray images (digital and scanned) from two U.S. hospitals
- CSV file containing the age (to be predicted) and the gender (useful side information)



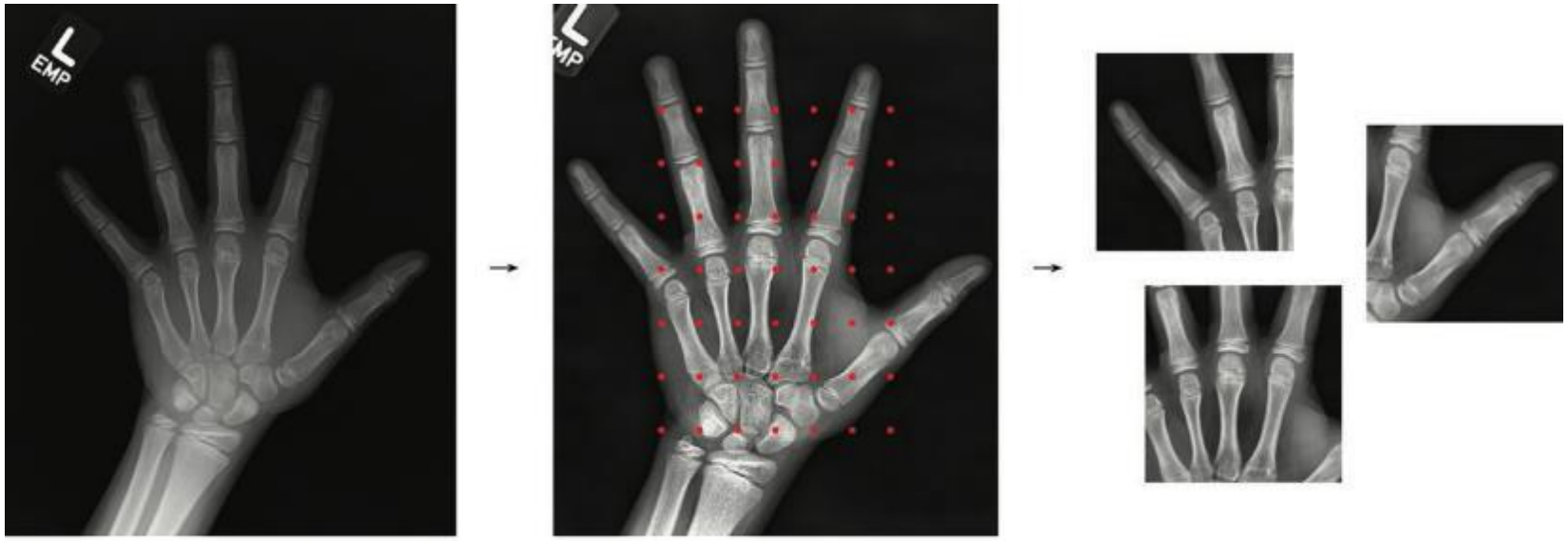
Winner model from [Halabi19]

- The age is predicted with an accuracy of 4 months



Second-place model from [Halabi19]

- Gender-specific models
- Each image was divided into 49 overlapping patches
- Use ResNet-50



Original Raw Image

Cropped + Resized + CLAHE

Each red point represents the center of an extracted patch.

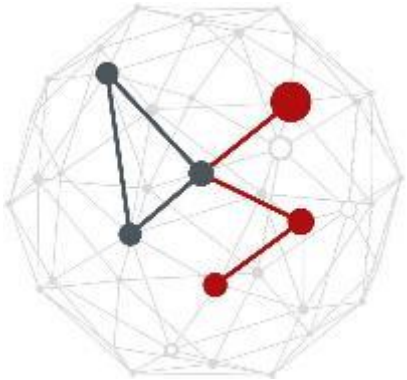
Each patch is used as a training example for the CNN.

Possible project developments

- Solve the bone age prediction as a regression task
  - as input: use row images or extract features
  - as output: use the age value (for regression)
  - assess the importance of the gender information into the regression
  - possible idea: use the entire image or use subpatches and then apply a decision fusion mechanism
- Architectures
  - different possibilities: CNN, RNN, attention, … SNN

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT C3



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project C3 “Lung disease prediction from X-ray images”

Reference papers

[Jiancheng2023] Yang, Jiancheng, et al. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification, Scientific Data 10.1 (2023): 41.

[Xiaosong2017] Wang, Xiaosong, et al. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Dataset (2 GB uncompressed)

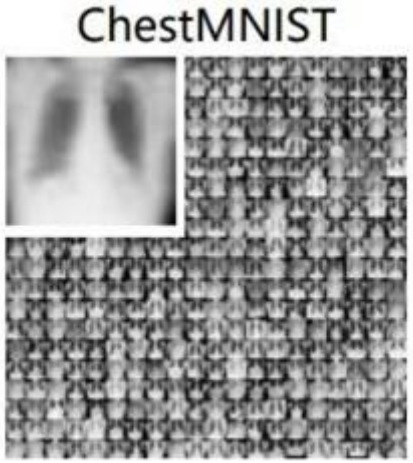
https://medmnist.com/ dataset+code

https://zenodo.org/records/10519652 download the chestMNIST dataset

Three versions: image sizes 1x64x64, 1x128x128, and 1x224x224

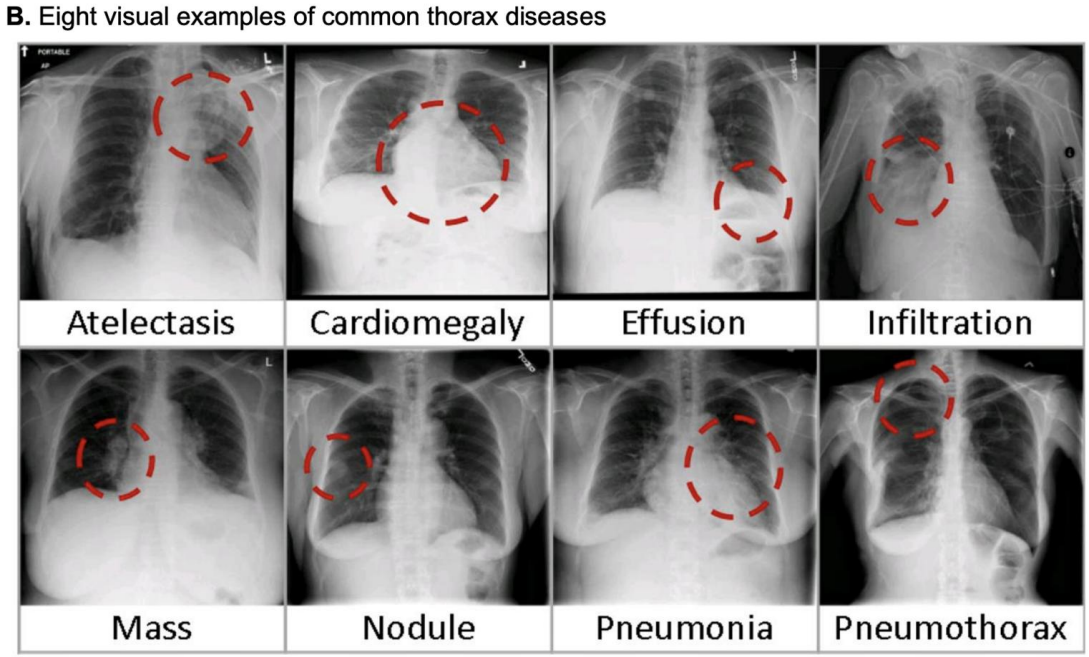
Dataset description

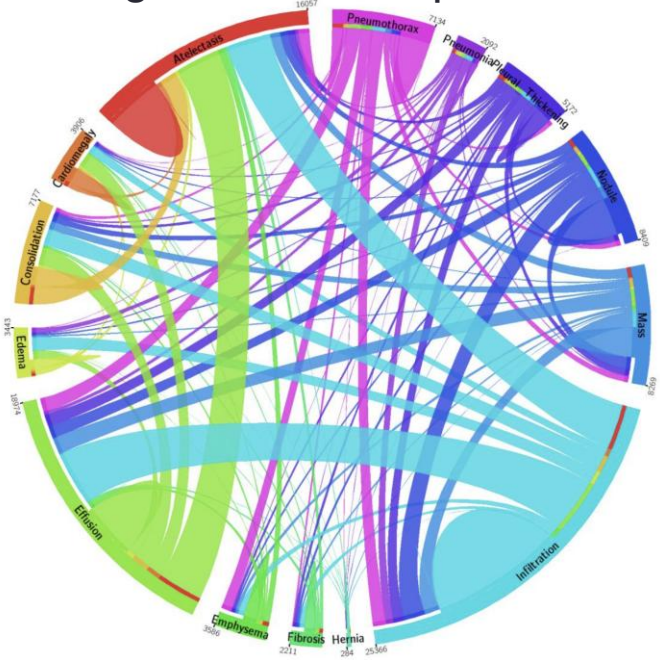
- 112,120 frontal X-ray images
- 30,805 unique patients
- 14 different classes of diseases



MedMNIST2D	Data Modality	Tasks (# Classes/Labels)	# Samples	# Training / Validation / Test
ChestMNIST	Chest X-Ray	Multi-Label (14) Binary-Class (2)	112,120	78,468 / 11,219 / 22,433

some images have multiple labels





Approach in [Xiaosong2017]

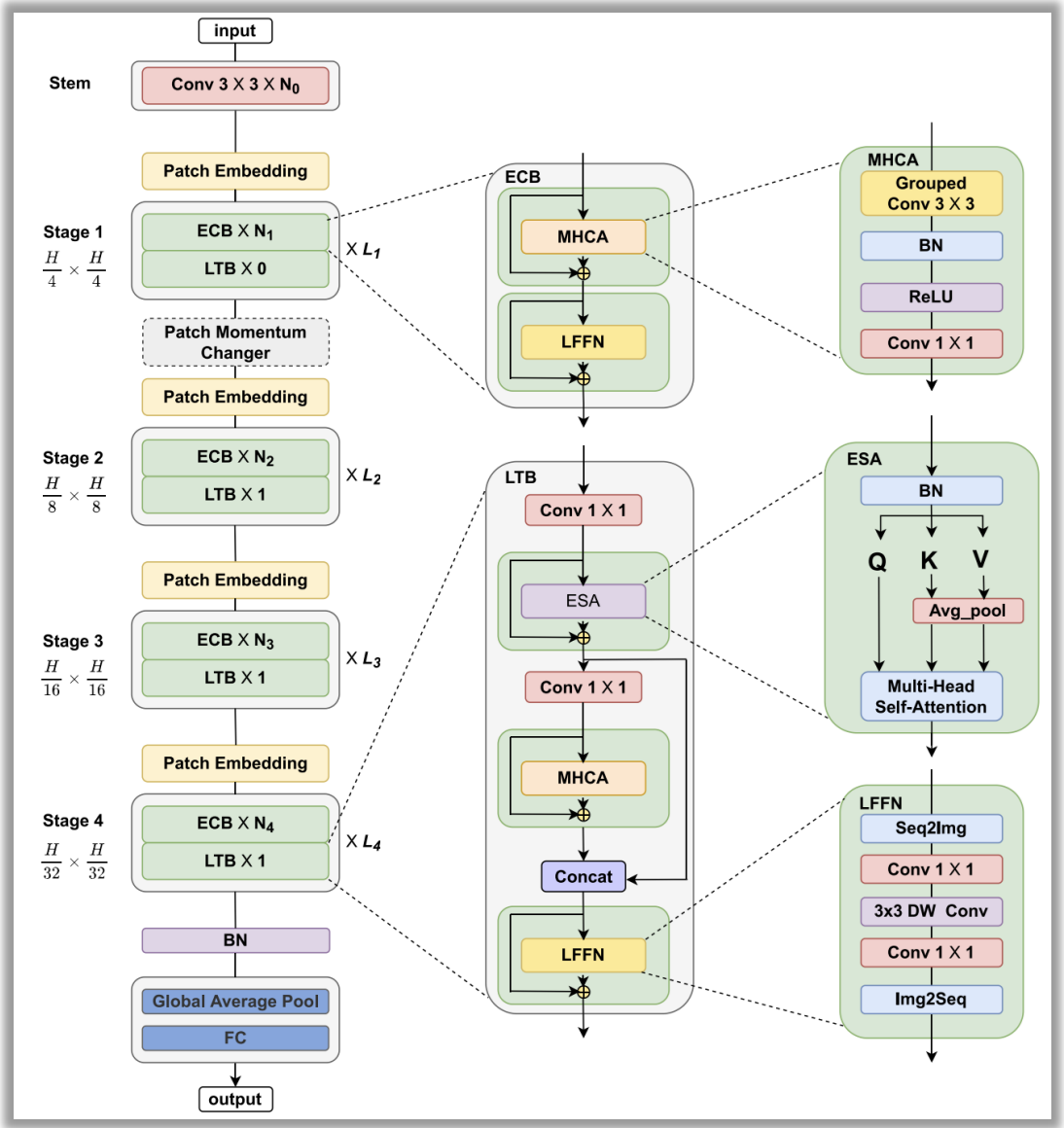
- Classification considering 8 classes
- 4 different pre-trained models: AlexNet, GoogLeNet, VGG and ResNet-50

Setting	Atelectasis	Cardiomegaly	Effusion	Infiltration	Mass	Nodule	Pneumonia	Pneumothorax
Initialization with different pre-trained models								
AlexNet	0.6458	0.6925	0.6642	0.6041	**0.5644**	0.6487	0.5493	0.7425
GoogLeNet	0.6307	0.7056	0.6876	0.6088	0.5363	0.5579	0.5990	0.7824
VGGNet-16	0.6281	0.7084	0.6502	0.5896	0.5103	0.6556	0.5100	0.7516
ResNet-50	**0.7069**	**0.8141**	**0.7362**	**0.6128**	0.5609	**0.7164**	**0.6333**	**0.7891**
Different multi-label loss functions								
CEL	0.7064	0.7262	0.7351	0.6084	0.5530	0.6545	0.5164	0.7665
W-CEL	0.7069	0.8141	0.7362	0.6128	0.5609	0.7164	0.6333	0.7891

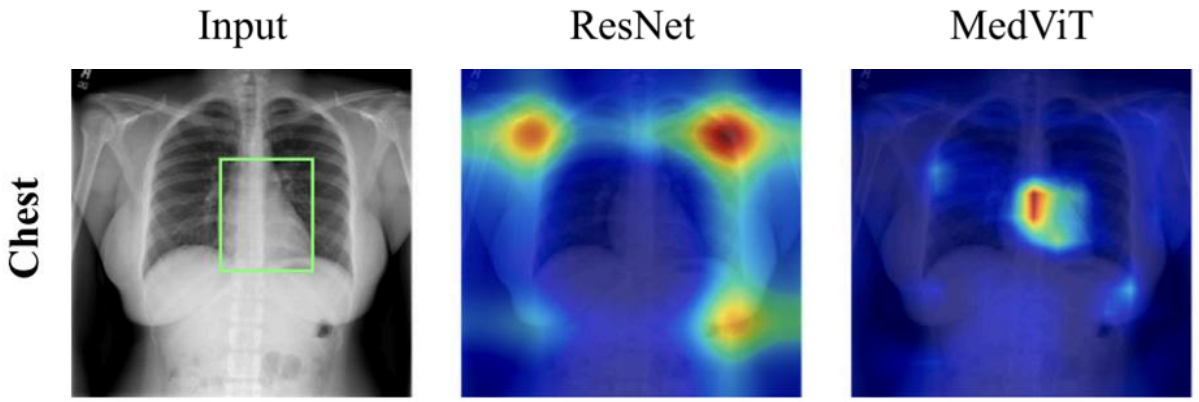
Table 3. *AUCs of ROC curves for multi-label classification in different DCNN model setting.*

Approach in [Jiancheng2023]

- MedViT: composed of a patch embedding layer, transformer blocks and a series of stacked convolution in each stage



Approach in [Jiancheng2023]



Methods	ChestMNIST	
	AUC	ACC
ResNet-18 (28) [20]	0.768	0.947
ResNet-18 (224) [20]	0.773	0.947
ResNet-50 (28) [20]	0.769	0.947
ResNet-50 (224) [20]	0.773	0.948
auto-sklearn [75]	0.649	0.779
AutoKeras [76]	0.742	0.937
Google AutoML [77]	0.778	0.948
MedVIT-T (224)	0.786	0.956
MedVIT-S (224)	0.791	0.954
MedVIT-L (224)	0.805	0.959

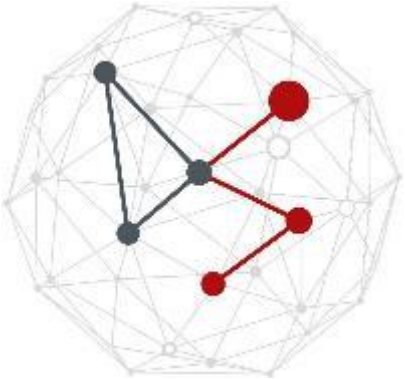
Possible project developments

- Classification task
  - use row images or extract features
  - use the 8 classes considered in [Xiaosong2017] or 14 classes as done in [Jiancheng2023]
  - try with different image sizes (64x64, 128x128, 224x224)
  - possible approaches: classify the entire image or use subpatches and then apply a decision fusion mechanism
  - use attention mechanisms
  - use spiking neural networks

- Architectures
  - CNN, RNN, attention, …

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT C4

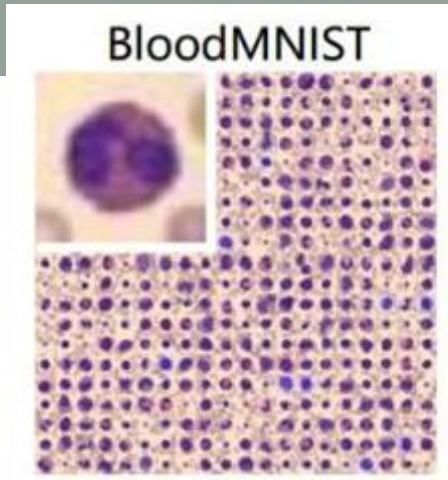


DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

800 ANNI 1222 · 2022

UNIVERSITÀ DEGLI STUDI DI PADOVA



Project C4 “blood cell type prediction”

Reference papers

[Jiancheng2023] Yang, Jiancheng, et al. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification, Scientific Data 10.1 (2023): 41.

[Acevedo2020] Acevedo, A. et al. A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. Data Brief 30, 105474, (2020).

Dataset (2 GB uncompressed)

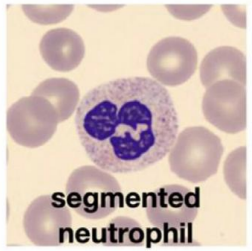
https://medmnist.com/ dataset+code

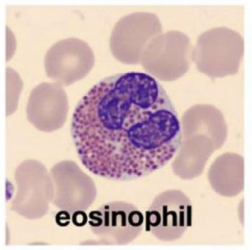
https://zenodo.org/records/10519652 download the bloodMNIST dataset

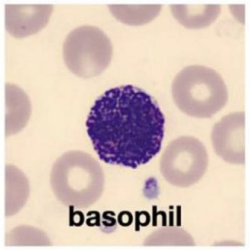
Three versions: image sizes 3x64x64, 3x128x128, and3x 224x224

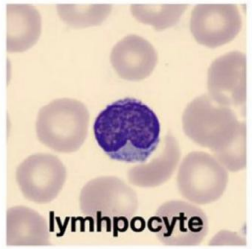
Dataset description

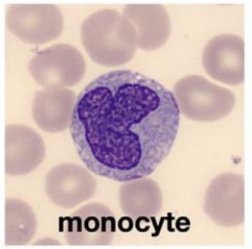
- 17,092 images of individual normal cells
- 8 different classes

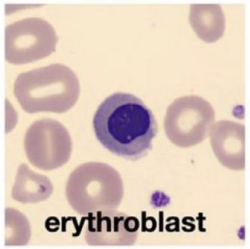








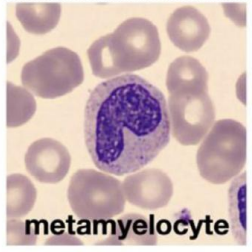


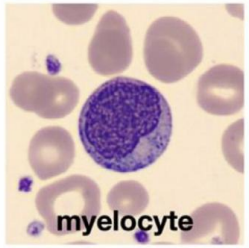


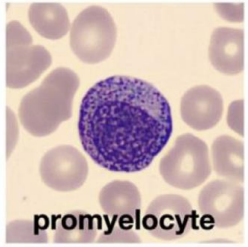
MedMNIST2D	Data Modality	Tasks (# Classes/Labels)	# Samples	# Training / Validation / Test
BloodMNIST	Blood Cell Microscope	Multi-Class (8)	17,092	11,959 / 1,712 / 3,421

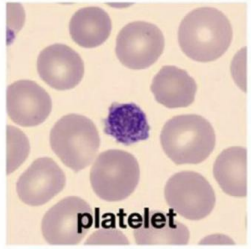
Types and number of cells in each group.

CELL TYPE	TOTAL OF IMAGES BY TYPE	%
neutrophils	3329	19.48
eosinophils	3117	18.24
basophils	1218	7.13
lymphocytes	1214	7.10
monocytes	1420	8.31
immature granulocytes (metamyelocytes, myelocytes and promyelocytes)	2895	16.94
erythroblasts	1551	9.07
platelets (thrombocytes)	2348	13.74
Total	17,092	100



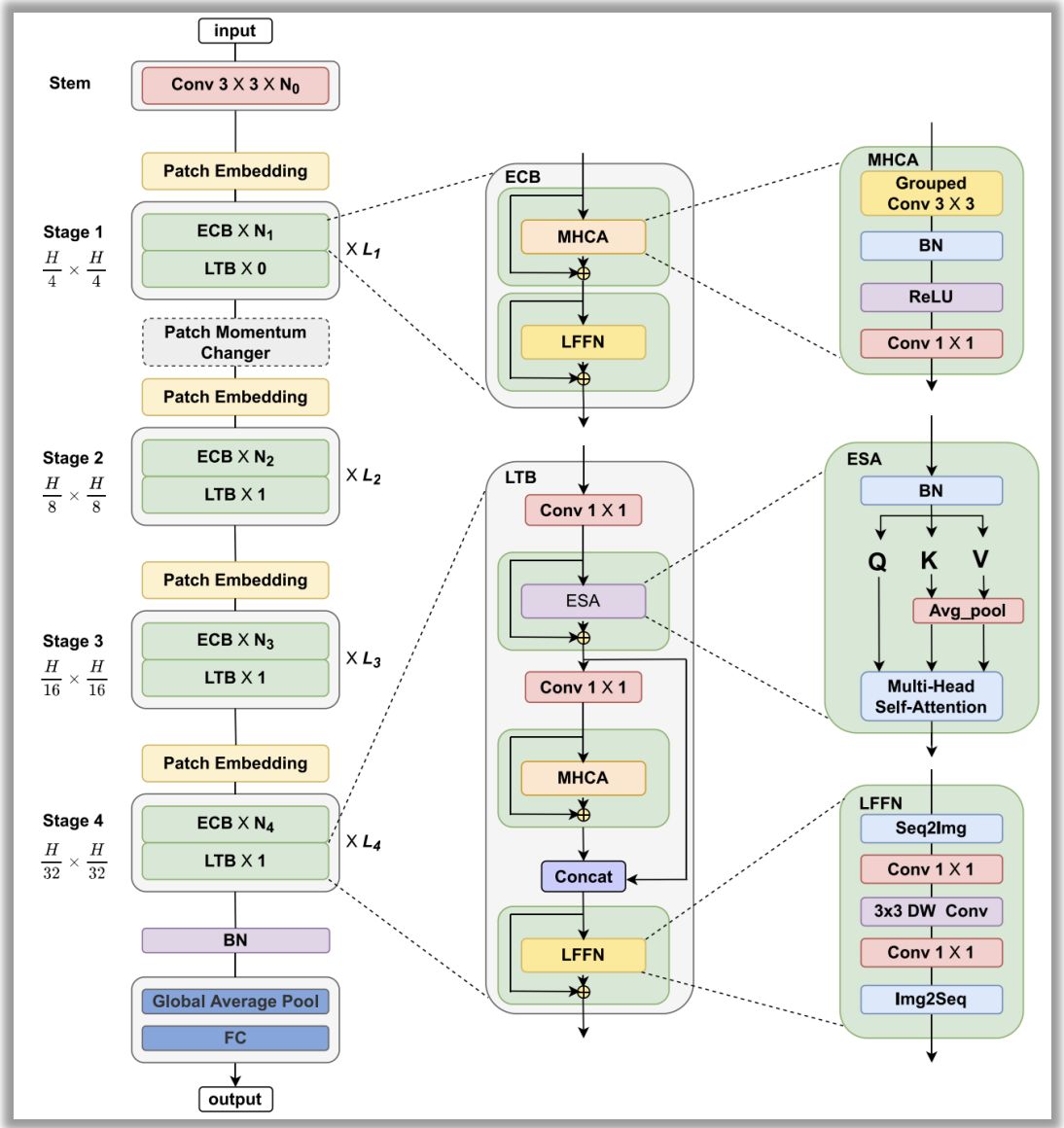






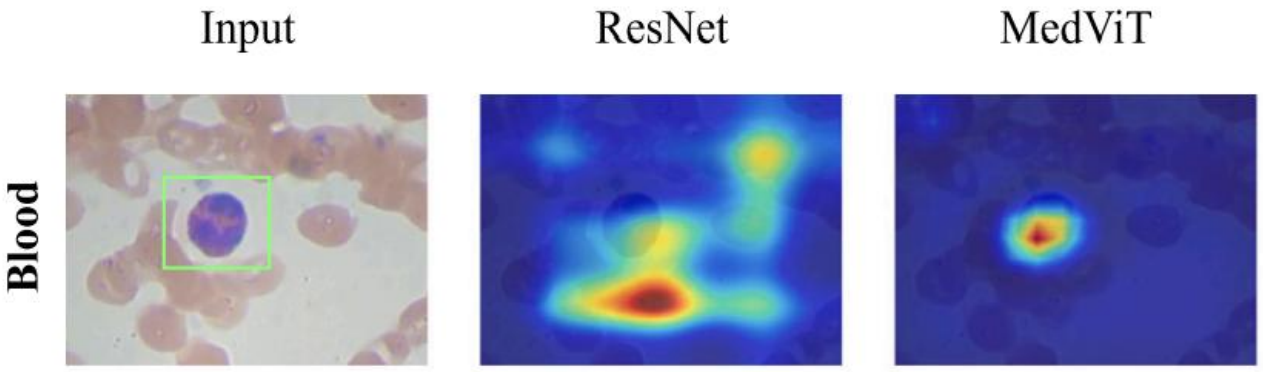
Approach in [Jiancheng2023]

- MedViT: composed of a patch embedding layer, transformer blocks and a series of stacked convolution in each stage

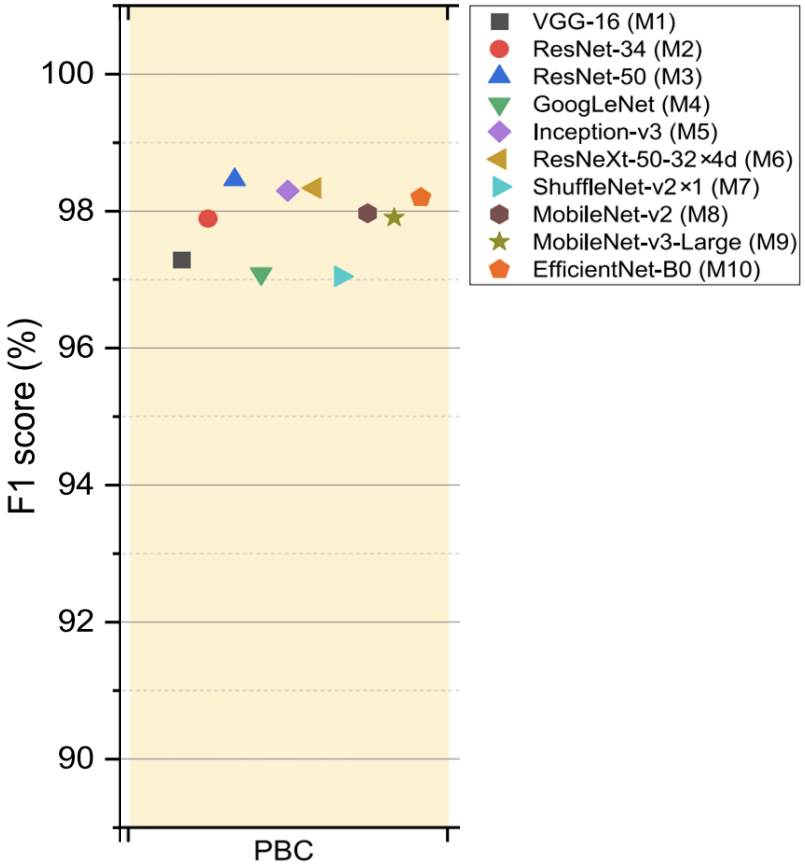


Approach in [Jiancheng2023]

Methods	BloodMNIST	
	AUC	ACC
ResNet-18 (28) [20]	0.998	0.958
ResNet-18 (224) [20]	0.998	0.963
ResNet-50 (28) [20]	0.997	0.956
ResNet-50 (224) [20]	0.997	0.950
auto-sklearn [75]	0.984	0.878
AutoKeras [76]	0.998	0.961
Google AutoML [77]	0.998	0.966
MedVIT-T (224)	0.996	0.950
MedVIT-S (224)	0.997	0.951
MedVIT-L (224)	0.996	0.954



Approach in [Rui2022]



PBC	
Methods	**Accuracy**
Acevedo et al. [35]	96.2%
Bagido et al. [36]	98.4%
Long et al. [67]	**99.3%**
Proposed	98.46%

Check the referenced papers for their approaches

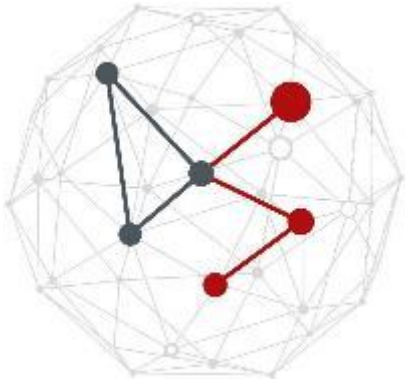
[Rui2022] Liu, Rui, et al. AIMIC: Deep learning for microscopic image classification. *Computer Methods and Programs in Biomedicine* 226 (2022): 107162.

Possible project developments

- Classification task
  - use row images or extract features
  - try with different image sizes (3x64x64, 3x128x128, 3x224x224)
  - possible approaches: classify the entire image or use subpatches and then apply a decision fusion mechanism
  - use attention mechanisms
  - use spiking neural networks
- Architectures
  - CNN, RNN, attention, …

MIME
Master's degree ICT Internet Multimedia Engineering

PART D
WIRELESS SIGNALS



DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO
MATEMATICA

800 ANNI 1222·2022

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Proposed Projects

PART A – ON BODY AND ENVIRONMENTAL SENSORS

1) A1: Activity recognition with four accelerometers
2) A2: Pathological gait recognition
3) A3: Motor imagery classification from EEG for brain–computer interface
4) A4: Arrhythmias detection from ECG recording

PART B – AUDIO SIGNALS

1) B1: Speech command recognition (keyword spotting)
2) B2: Environmental sound classification
3) B3: Speaker identification/verification
4) B4: Digit recognition (Spiking Heidelberg Digit)

PART C – IMAGES

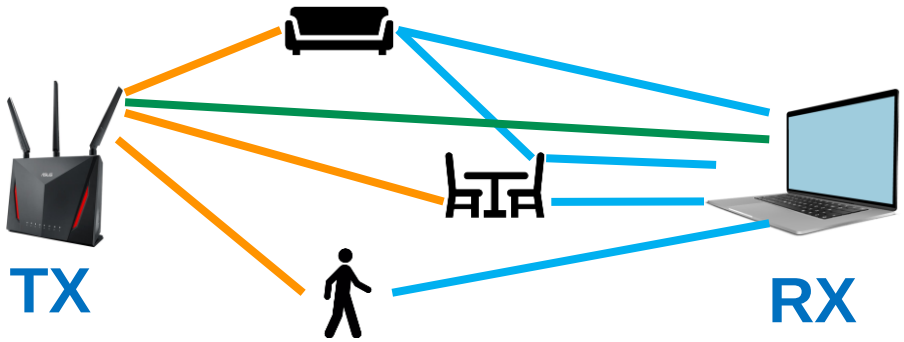
1) C1: Diabetic retinopathy detection
2) C2: Bone age prediction from hand radiographs
3) C3: Lung disease prediction from X-ray images
4) C4: Blood cell type prediction

PART D – RADIO SIGNALS

1) D1: Gesture recognition through radars

General idea

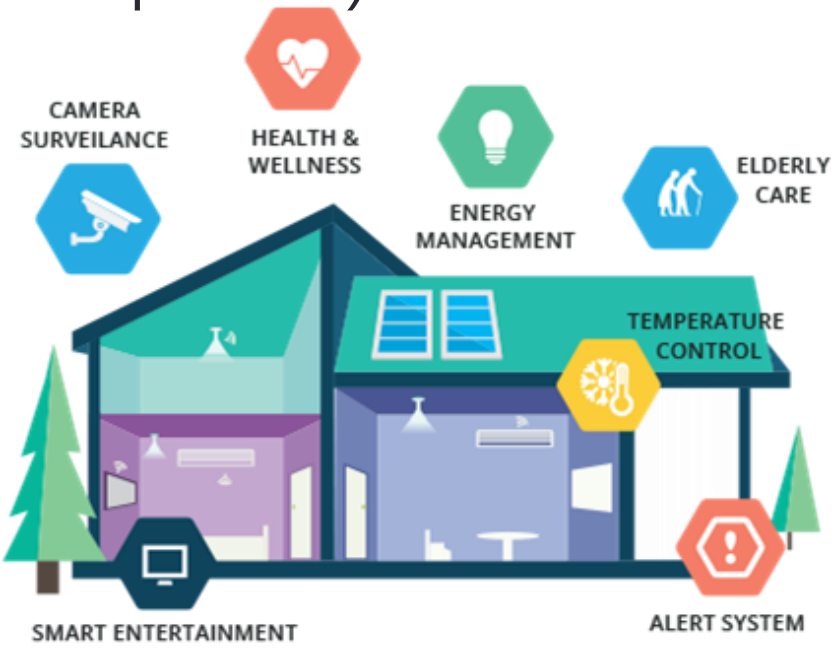
- **Main idea.** The presence and the movement of objects in the environment affect the Wi-Fi signal (multi-path) propagation
- These modifications can be
  - estimated via dedicated signal processing on the Wi-Fi channel frequency response (CFR) and
  - used as a proxy for human sensing applications



General idea

- Why radio waves for sensing?
  - more user friendly than approaches based on wearable devices: the user is not required to wear anything
  - privacy preserving: no images of the subjects are captured
  - insensitive to light conditions and the presence of dust, smoke
  - see through walls/obstacles (low frequencies)

- Applications
  - Human detection
  - Fall detection
  - Human activity recognition
  - Human vital signs monitoring
  - People tracking
  - Gesture recognition



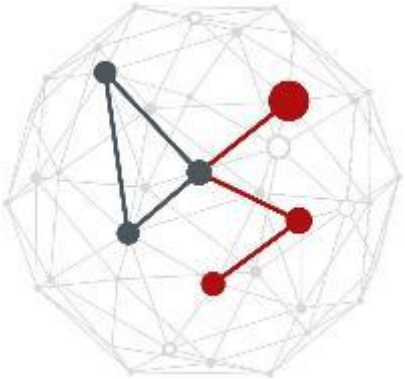
General idea

- In recent years, spurred by the pervasiveness of Wi-Fi-enabled devices, Wi-Fi sensing has been widely investigated
- September 2020: the IEEE 802.11bf working group was established to empower Wi-Fi devices with sensing capabilities [Meneghello2023]
- The goal is to allow Wi-Fi routers to perform a dual role
  - communication access points (AP)
  - monitoring devices, leveraging ongoing Wi-Fi traffic as well as ad-hoc packets to deliver the sensing service

[Meneghello2023] F Meneghello, C Chen, C Cordeiro, F Restuccia, Toward Integrated Sensing and Communications in IEEE 802.11 bf Wi-Fi Networks. *IEEE Communications Magazine*, 2023.

MIME
Master's degree ICT Internet Multimedia Engineering

PROJECT D1



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA

Project D1 “Gesture recognition through radars”

Reference papers

[Wang2016] S. Wang, J. Song, J. Lien, I. Poupyrev, O. Hilliges, Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. Proceedings of the 29th Annual Symposium on User Interface Software and Technology, 851-860, 2016.

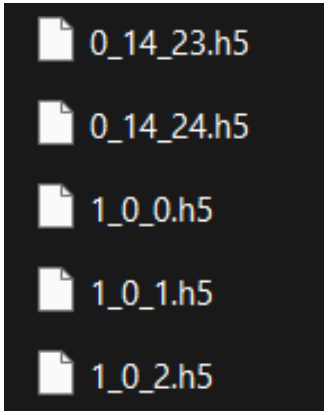
[Tsang2021] I.J. Tsang, F. Corradi, M. Sifalakis, W. Van Leekwijck, S. Latré, Radar-Based Hand Gesture Recognition Using Spiking Neural Networks. Electronics 2021, 10, 1405.

Dataset (8.56 GB uncompressed)

Soli dataset

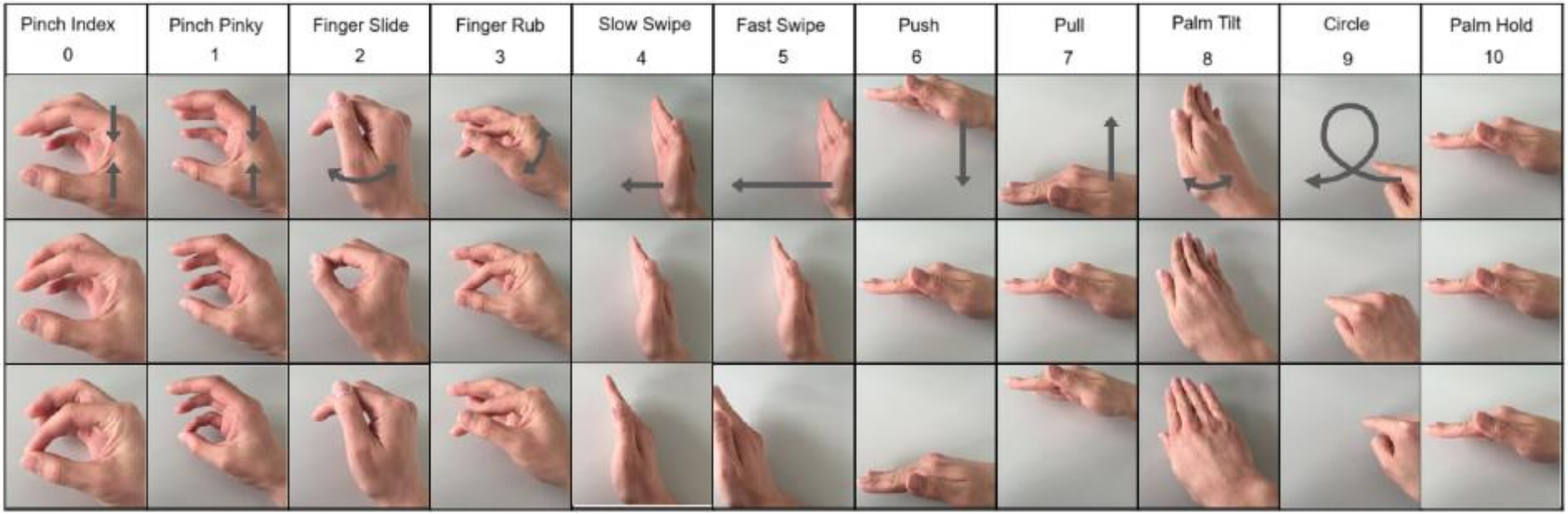
Dataset description

- The Soli dataset is based on the dynamic gesture signals collected by a millimeter-wave radar
- Each data array has shape [number of frames] x 1024 (can be reshaped back to 2D Range-Doppler images with shape 32 x 32)
- 11 gestures
- 10 users
- File names in the dataset folder are defined as [gesture ID]_[session ID]_[instance ID].h5
- Sequences with gesture ID 11 are background signals with no presence of hand



Dataset description

- The 11 gestures are listed in the image below
- Each column represents one gesture, and for each of them, three important steps are reported



Dataset description

- Dataset session arrangement for evaluation:
  - 11 (gestures) * 25 (instances) * 10 (users) = 2750 sequences

    for cross-user evaluation

    [session 2 (25), 3 (25), 5 (25), 6 (25), 8 (25), 9 (25), 10 (25), 11 (25), 12 (25), 13 (25)]

  - 11 (gestures) * (50 (instances) * 4 (sessions) + 25 (instances) * 2 (sessions)) = 2750 sequences

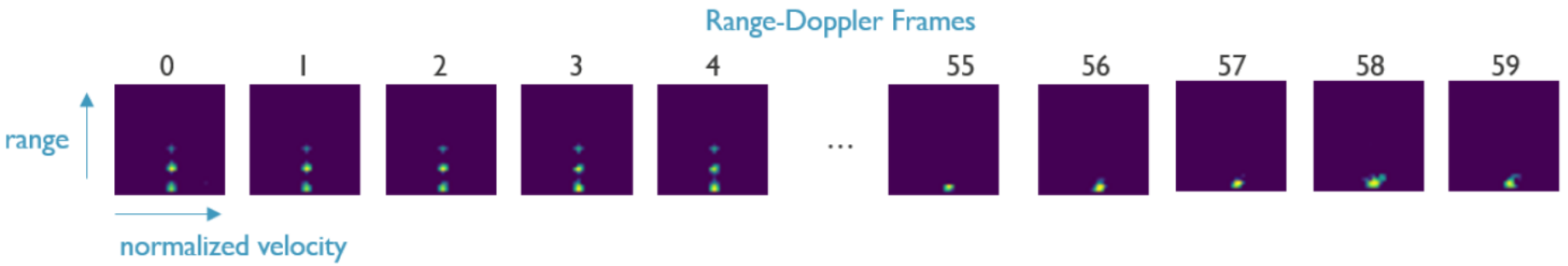
    to evaluate cross-session performance and to explore personalized gesture recognition

    [session 0 (50), 1 (50), 4 (50), 7 (50), 13 (25), 14 (25)]

- Total of 2750 + 2750 = 5500 sequences

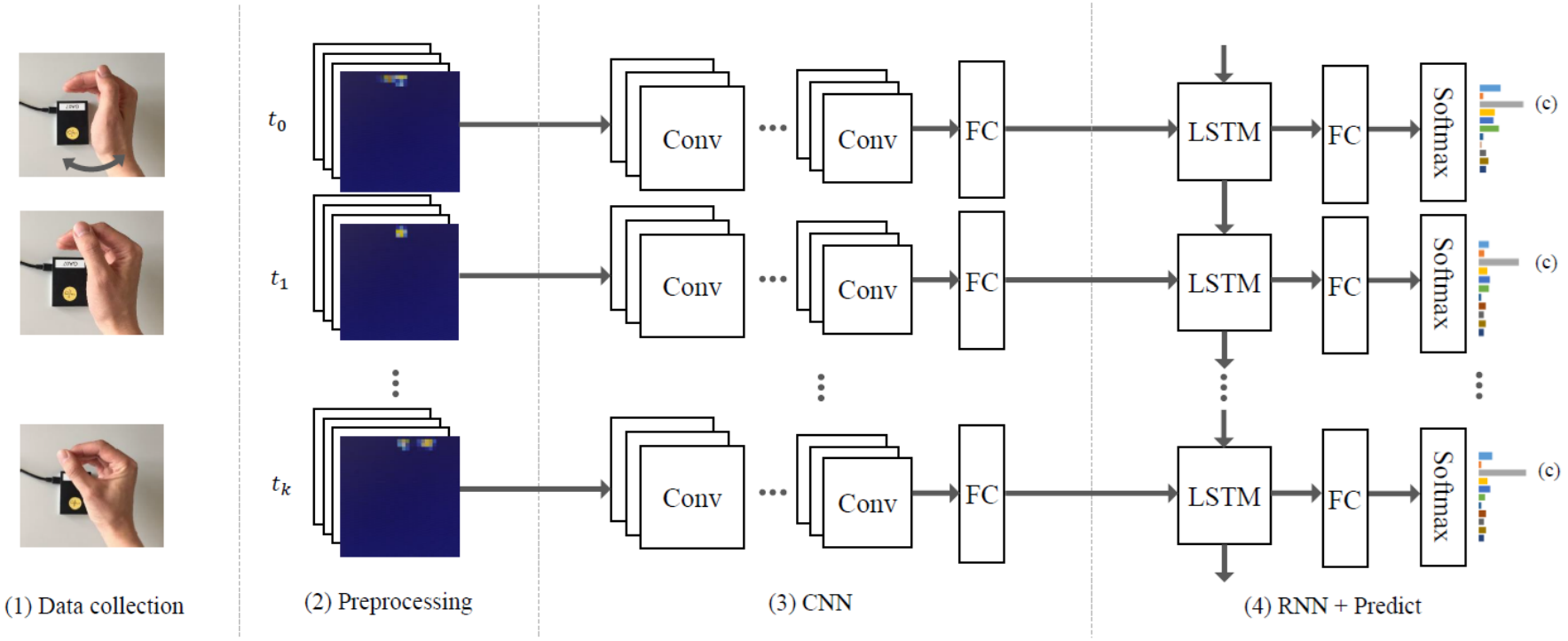
Approach in [Wang2016]

- Dataset pre-processing: background removal (using a per-pixel Gaussian model) + signal normalization
- Input to the model: set of consecutive Range-Doppler frames (stacked)



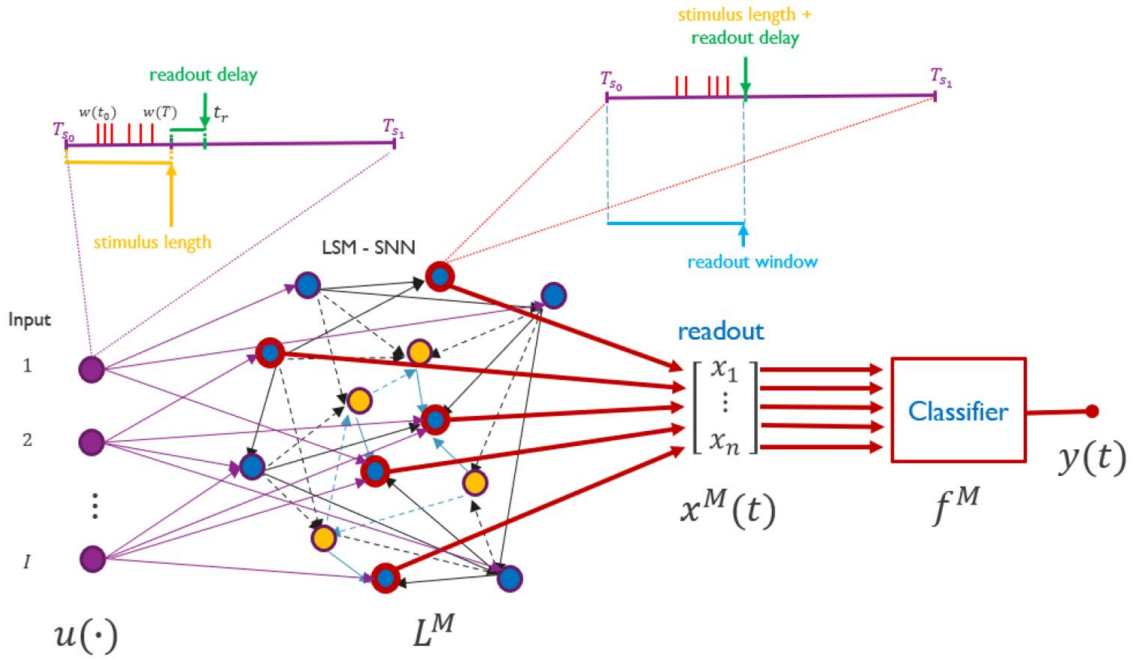
Approach in [Wang2016]

- Architecture composed of a CNN and an RNN stacked and jointly trained
- The final gesture recognition accuracy is used as optimization criterion
- Average accuracy of 87.17%



Approach in [Tsang21]

- Use of a Liquid State Machine (LSM), which is a type of reservoir computing (see [Maass2004]) that uses spiking neurons, followed by a readout map to perform the classification
- The authors used 3 different classifiers for comparison: logistic regression, random forest, and support vector machine



[Maass2004] W. Maass, H. Markram, On the computational power of circuits of spiking neurons, Journal of Computer and System Sciences, Volume 69, Issue 4, 2004.

Approach in [Tsang21]

- The liquid is composed of excitatory (E) and inhibitory (I) neurons, with 20% configured as inhibitory
- The synaptic connections are randomly assigned, creating a sparse network with the following ratios: EE=2, EI=2, IE=1, and II=1
- The Range-Doppler frames are encoded into spike trains, prior to feeding them to the LSM

Project proposal

- Classification of the 11 gestures
- Architectures
  - Non-spiking
    - Experiment with different hyper-parameters for both CNN and RNN (num. of layers, kernel size, type of recurrent units…)
    - Explore different features learning techniques (e.g., an CNN-based Autoencoder)
  - Spiking
    - Explore different configurations of the LSM (proportion of inhibitory neurons, synaptic connections…)
    - Explore different spike encoding techniques for the input data
    - Explore different architectures (e.g., fully SNN architectures with LIF neurons, using surrogate gradient approach)

Master's degree ICT Internet Multimedia Engineering

MACHINE LEARNING FOR HUMAN DATA – FINAL EXAMINATION

Instructor

Michele Rossi - michele.rossi@unipd.it

Lab. classes

Eleonora Cicciarella - eleonora.cicciarella@phd.unipd.it

Cesare Bidini - cesare.bidini@phd.unipd.it

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

DIPARTIMENTO MATEMATICA

1222 · 2022 800 ANNI

UNIVERSITÀ DEGLI STUDI DI PADOVA