

IPTC-Amazon collaboration

Meeting
(April 13th, 2023)



POLITÉCNICA



www.iptc.upm.es

IPTC-Amazon: Index

Index:

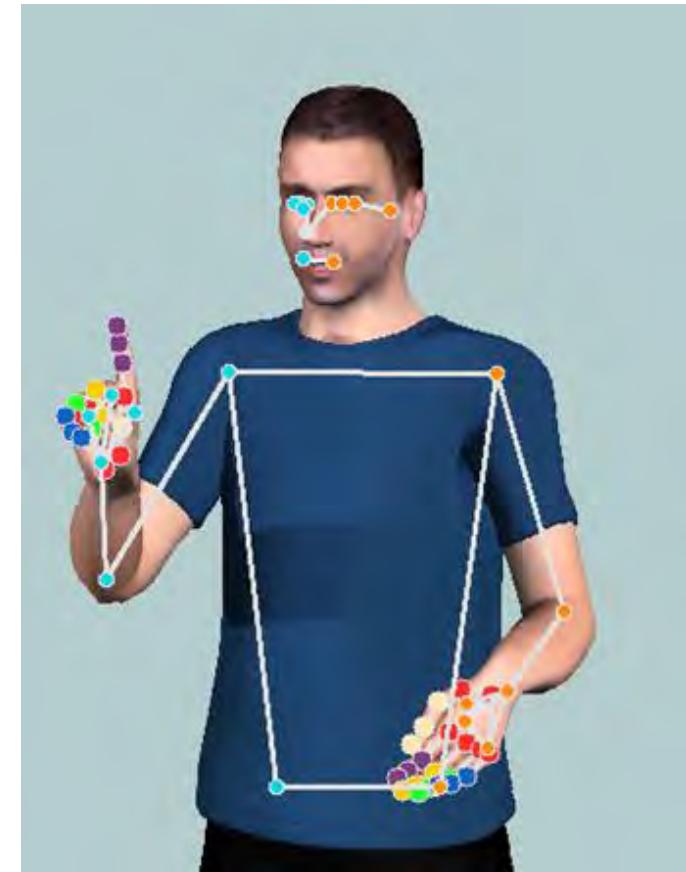
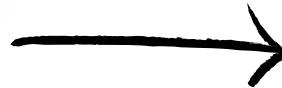
- Sign language motion generation from high level sign characteristics
- Speaker diarization with multimodal inputs
- Pose and spatial movement as input for dynamic content search & generation
- Entangling AI-audio synthesis models and multimodal representations
- Zero-shot sonorizing of video sequences

Sign language motion generation and analysis

Our Objective

```
<hamnosys_manual>
    <hamfist/>
    <hambetween/>
    <hamfist/>
    <hamthumbacrossmod/>
    <hamextfingeru/>
    <champalmd/>
    <hamshoulders/>
    <hamlrat/>
</hamnosys_manual>
```

Hand configurations

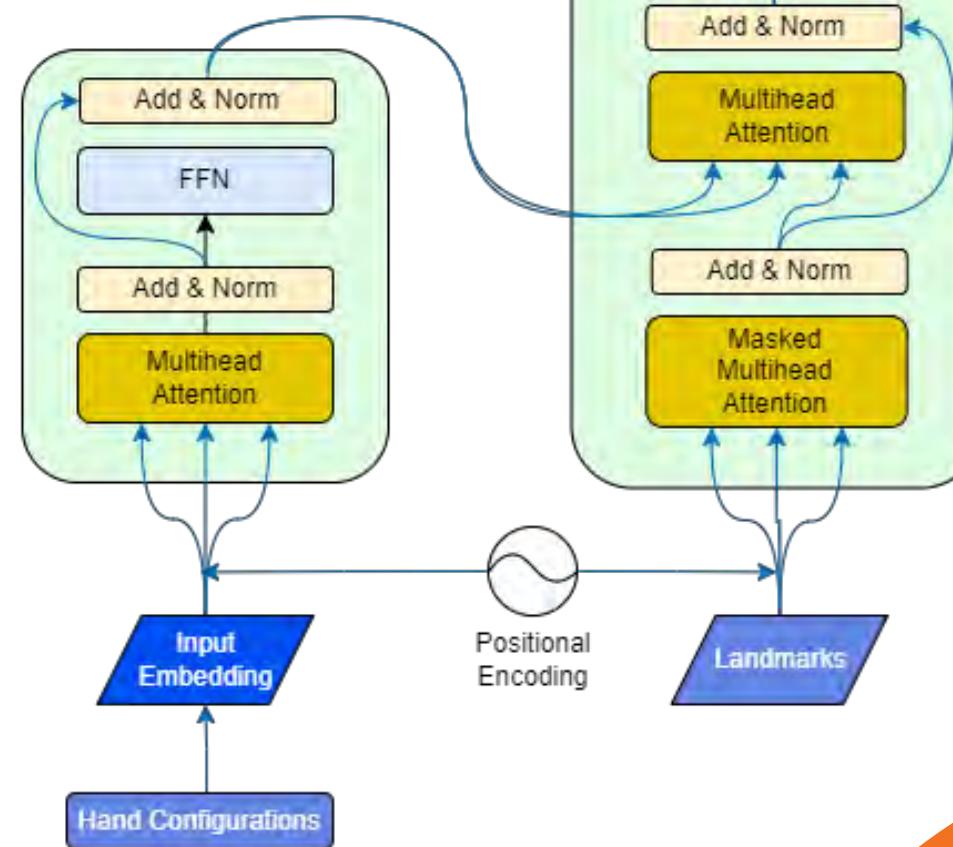


Landmarks

What have we done?

Transformer

- Based on a spanish to english translator transformer

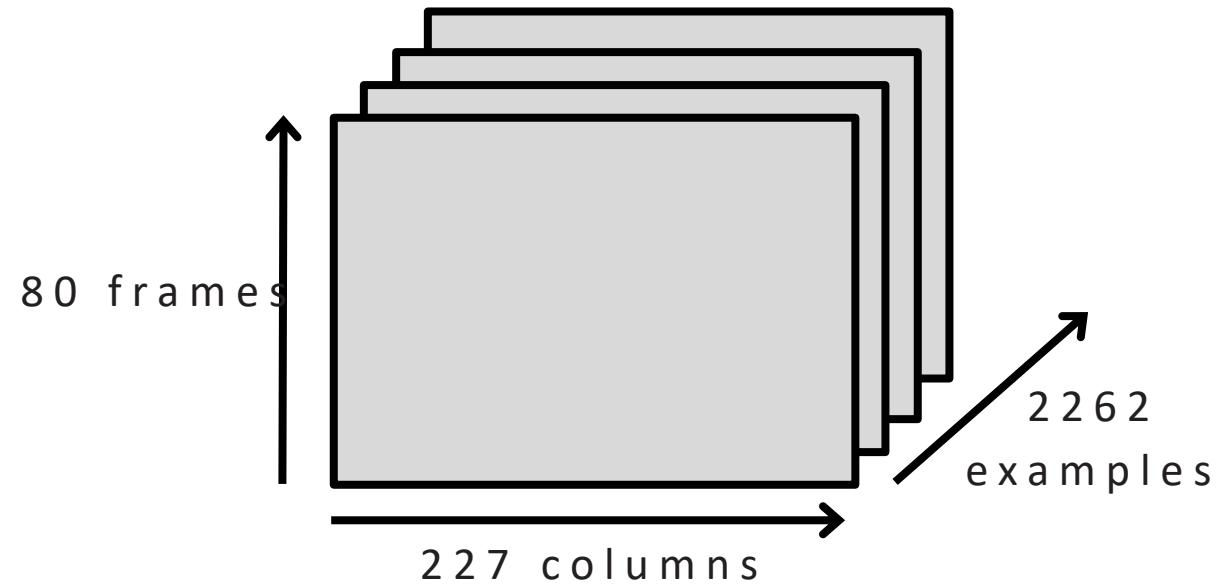


How have we done it?

- Data Preparation
- Training, Validation and Test
- Tokenizing and formatting the data
- Building the model
- Training the model
- Testing

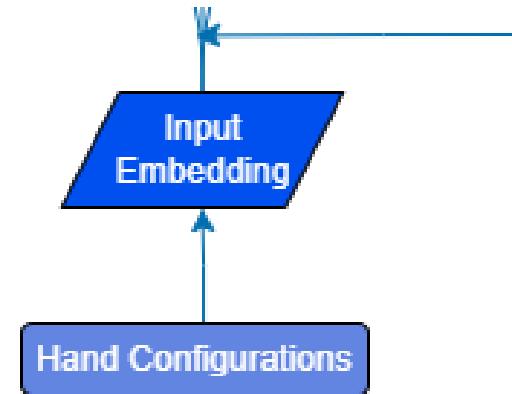
Data Preparation & Train, Test and Validation Set

- 754 x 3 examples (2626 dataframes)
 - Array of dataframes
 - Fill with zeros if the dataframe is less than 80 frames
 - Each dataframe will contain the landmarks + a stop token + hand configurations
-
- 1696 dataframes for training
 - 227 dataframes for validation
 - 339 dataframes for testing



Tokenizing & Formatting the data

- We are only going to tokenize the hand configurations.
- We train the tokenizers in our dataset.
- The WordPiece Tokenization Algorithm
- reserved_tokens =["[PAD]","[UNK]","[START]","[END]"]

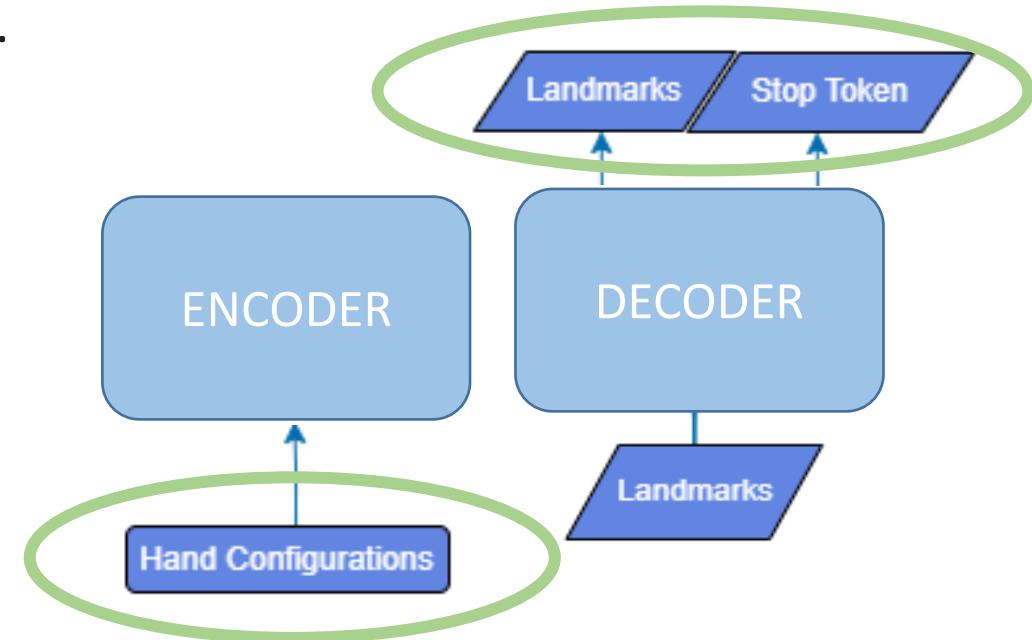


EXAMPLE

- **Hand Configurations:** hamflathand hamthumboutmod hamextfingeru hampalmu hamshoulders hamlrat hamparbegin hammovei hamarcu hamreplace hamextfingerui hampalmd hamparend hamrepeatfromstart next next
- **Tokens:** tf.Tensor([49 43 38 47 41 40 34 86 90 46 131 44 35 55 36 36], shape=(16,), dtype=int32)
- **Recovered text after detokenizing:** hamflathand hamthumboutmod hamextfingeru hampalmu hamshoulders hamlrat hamparbegin hammovei hamarcu hamreplace hamextfingerui hampalmd hamparend hamrepeatfromstart next next

Tokenizing & Formatting the data

- The model will seek to predict the target landmarks N+1.
- The training dataset will be divided (in, out).
 - `in_train_1 = hand_config_start_end_packer(x_train)`
 - `in_train_2 = y_train[:, :-1, :]` # The landmarks
 - `out_train_1 = y_train[:, 1:, :-1]` # The landmarks
 - `out_train_2 = y_train[:, 1:, -1]` # Stop Token

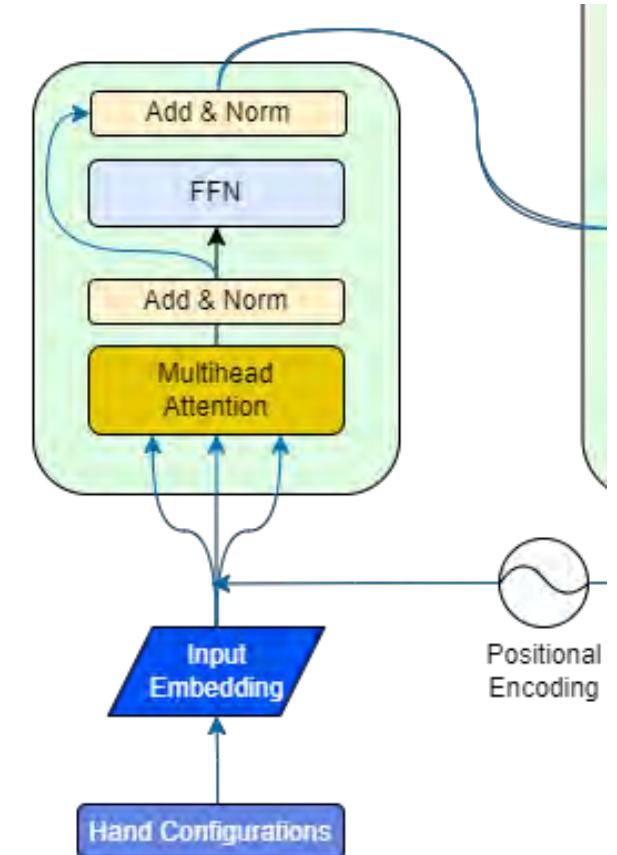


Building the model

Encoder

`keras_nlp.layers.TransformerEncoder`

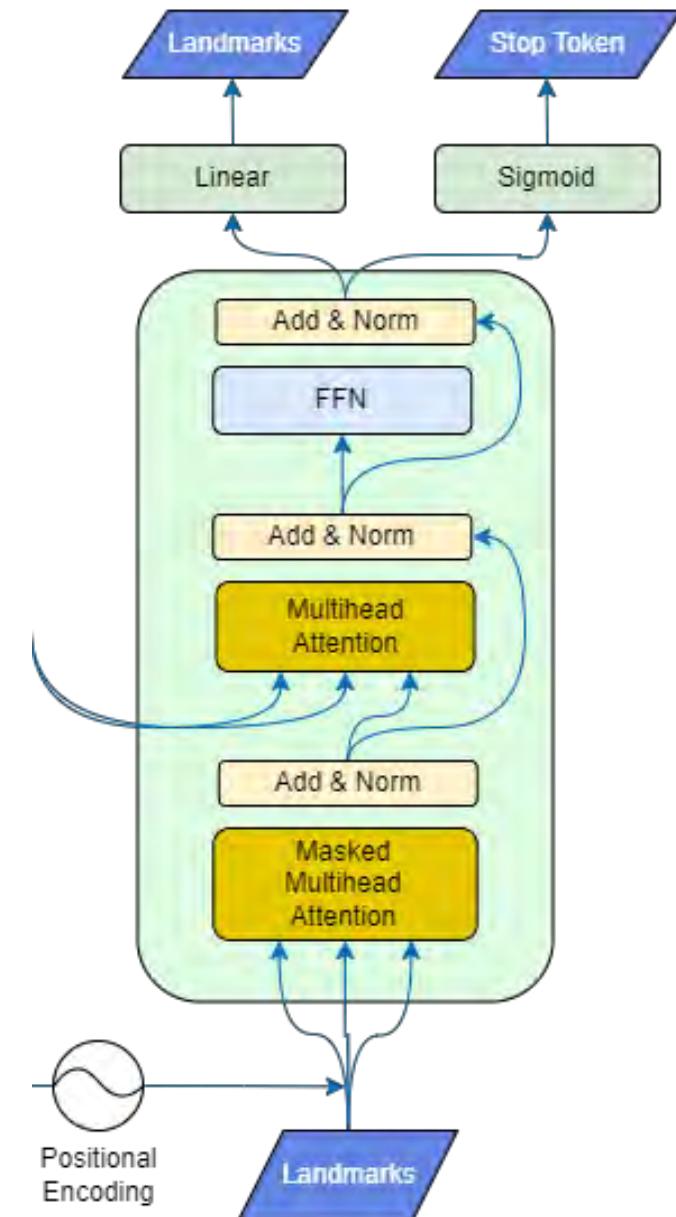
- Embedding layer + PositionalEmbedding Layer.
- `keras_nlp.layers.TokenAndPositionEmbedding` layer.



Decoder

`keras_nlp.layers.TransformerDecoder`

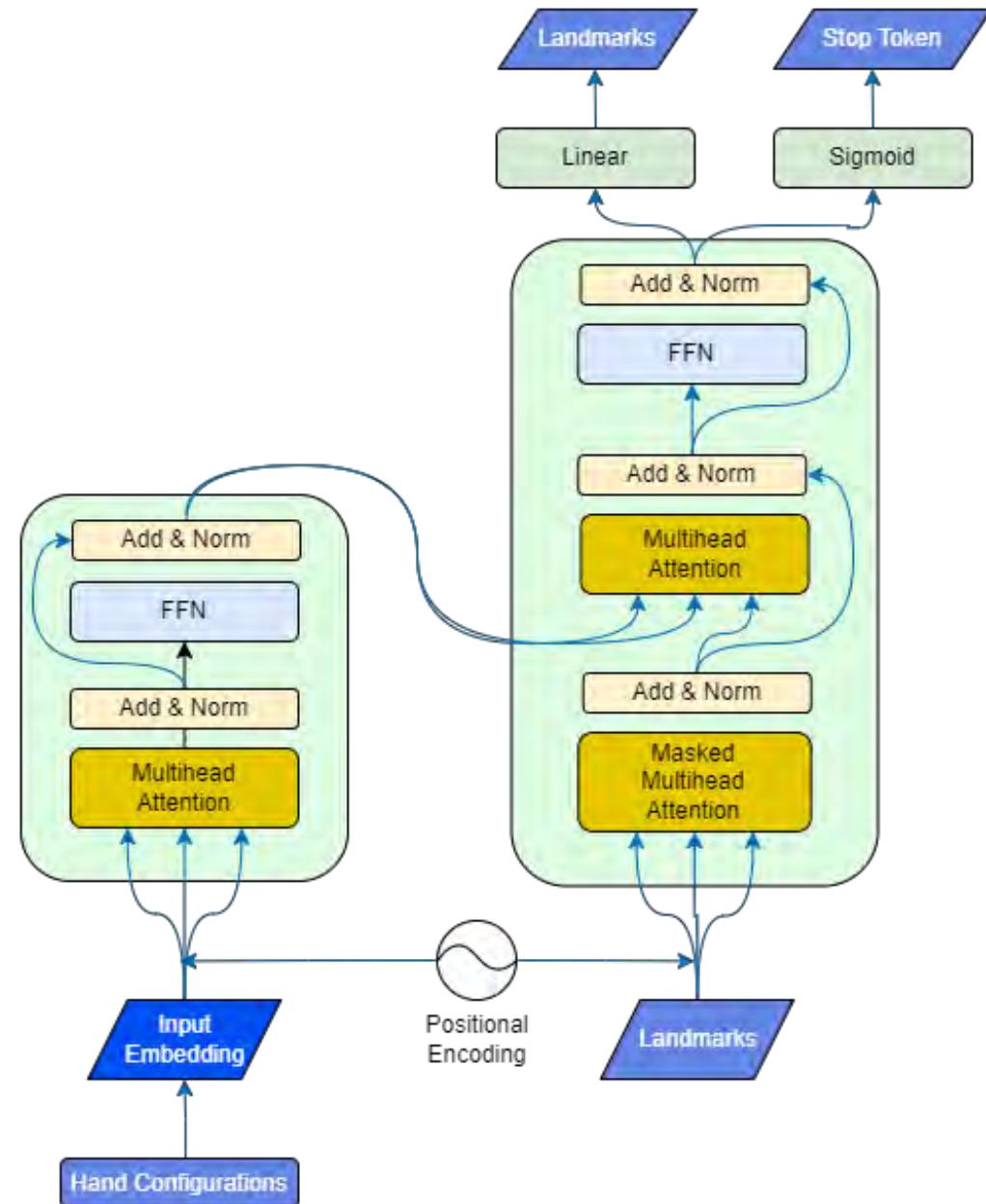
- PositionalEmbedding Layer.
- `keras_nlp.layers.PositionEmbedding` layer.
- Encoder Output + Target Sequence = Next Landmarks
- Two Outputs = Predicted Landmarks + Stop Token



Building the model

Transformer

```
transformer = keras.Model(  
    [encoder_inputs, decoder_inputs],  
    [decoder_output_1_1, decoder_output_2_2],  
    name="transformer",  
)
```

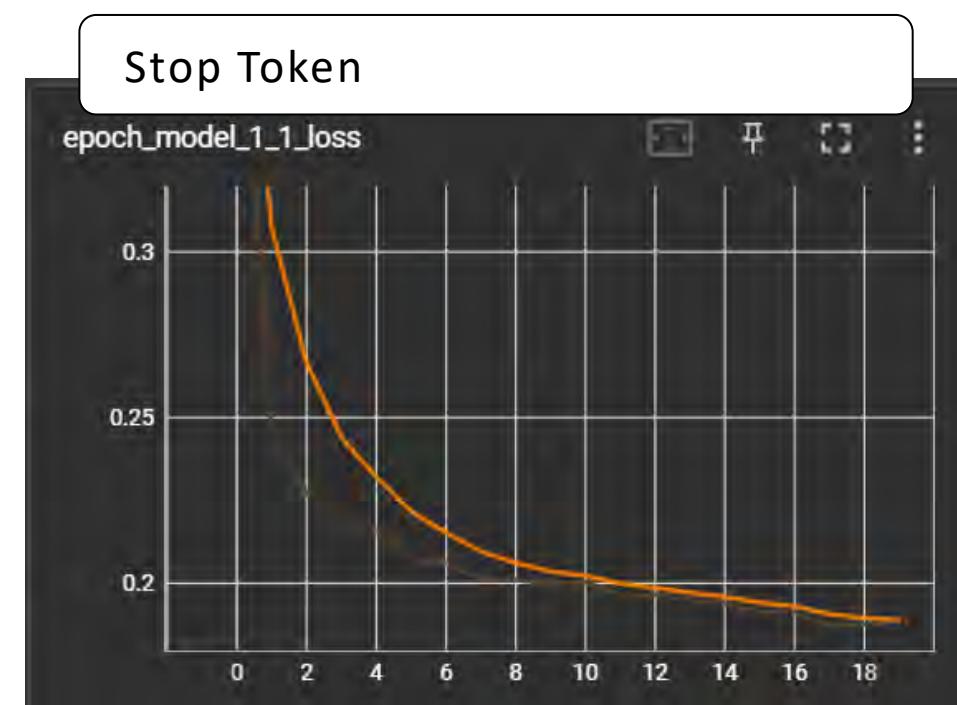
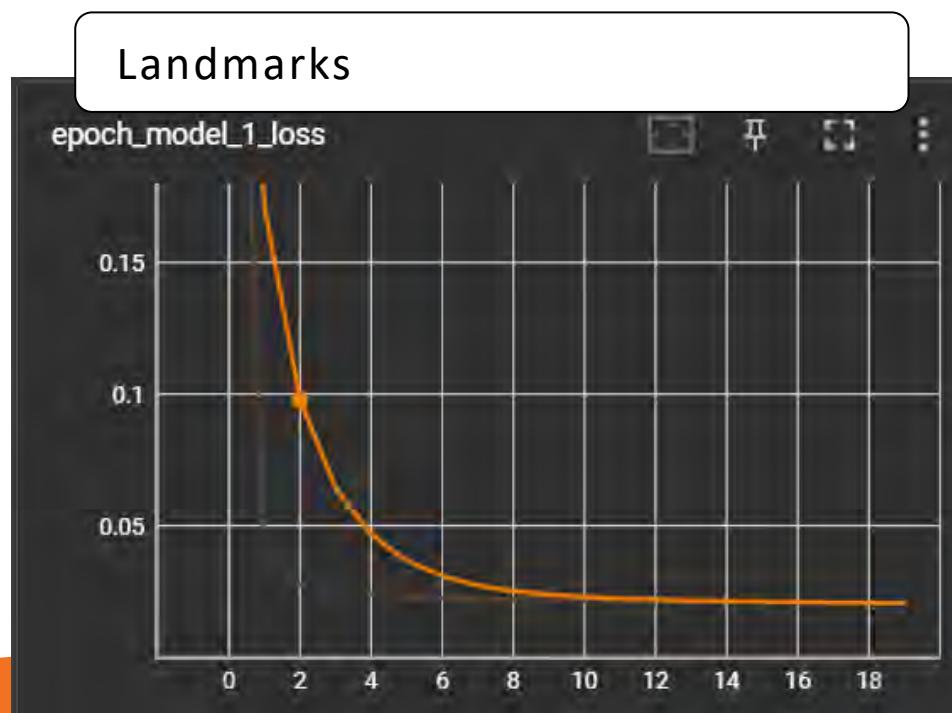


Training the model

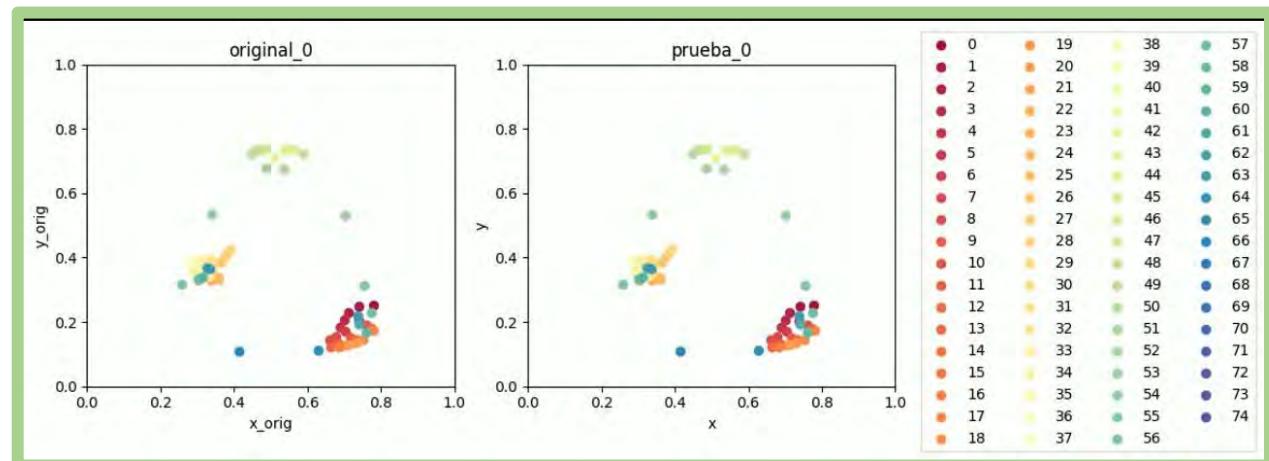
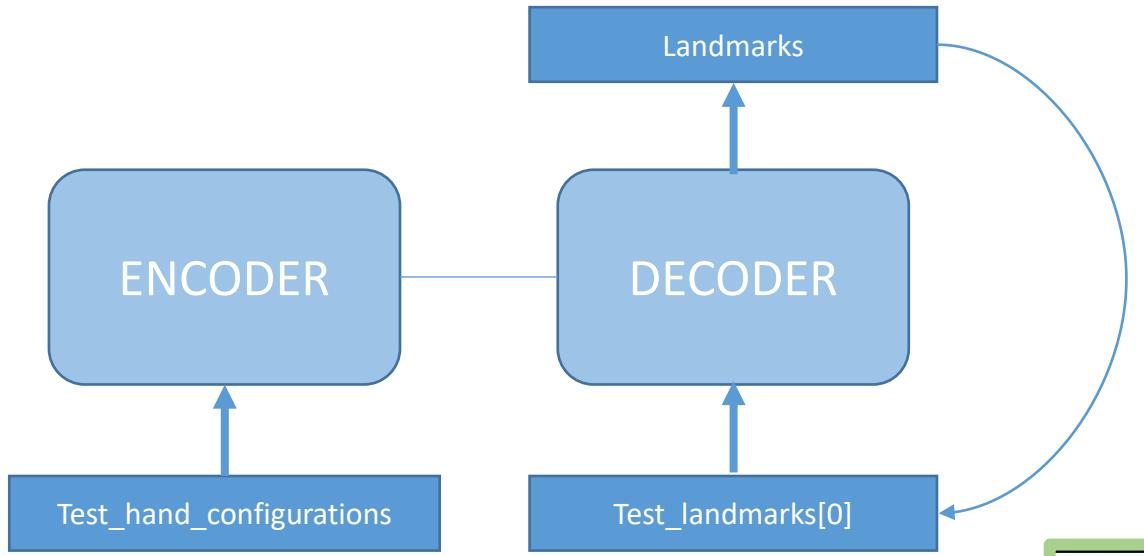
Separate loss functions for the generated landmarks and the stop token:

- loss_functions (landmarks , stop_token) = mse, binary_crossentropy
- metrics (landmarks, stop_token) = mse, binary_accuracy

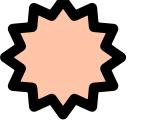
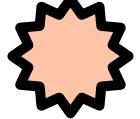
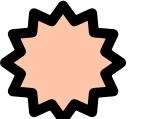
We train for 20 epochs in batches of 16 examples.



Testing



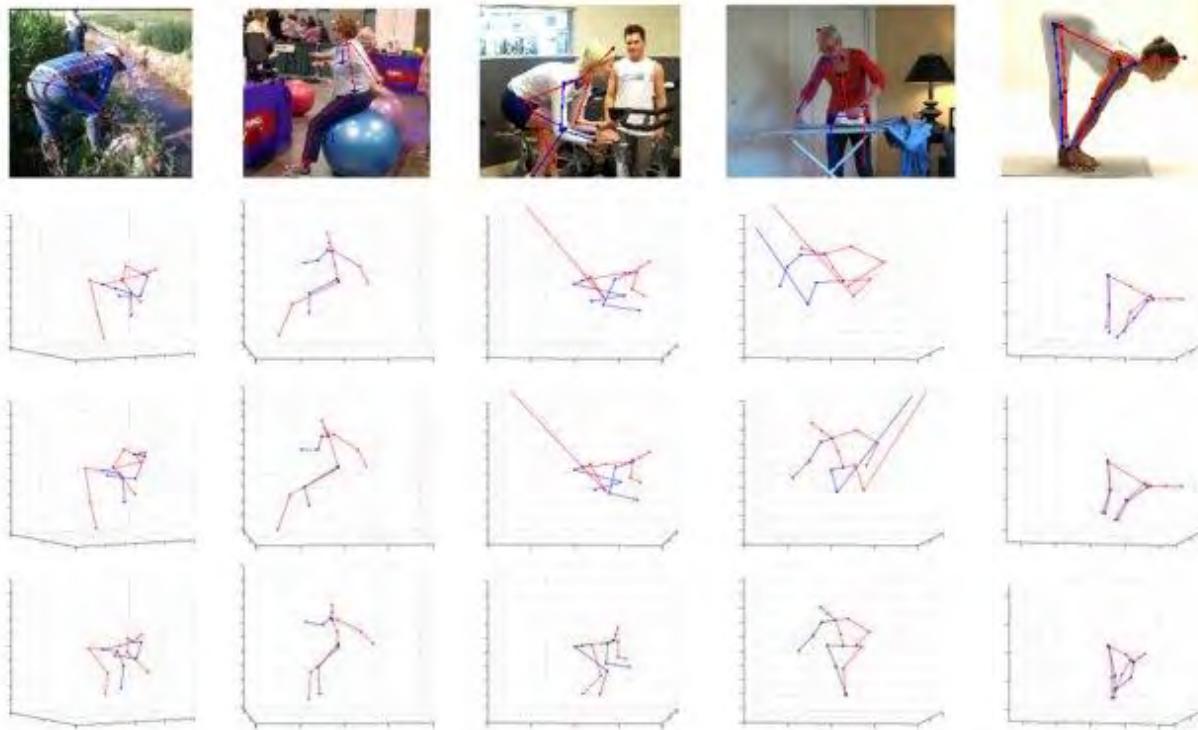
Future Lines

-  1 Improving the transformer performance
-  Changing the loss function  Changing the batch size and number of epochs
-  Try different Optimizers
-  2 Improving the data entries
-  Change the stop token design  Cleaning the landmarks inputs

Pose and spatial movement as input for dynamic content search & generation

Pose and spatial movement as input for dynamic content search & generation

Main purpose? To explore the potential of posture correctness analysis and multimodal feedback delivery for different applications (ergonomics, yoga, others).



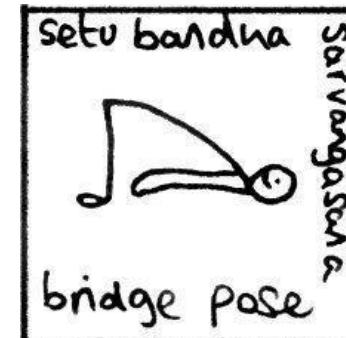
Tasks

- 1) Task scope definition ✓
- 2) Dataset search and evaluation. ✓
- 3) State of the art on building postural models ✓ and postural analysis.
- 4) Setting up an environment for posture classification from images. ✓
- 5) Model concept proposal for posture analysis, based on distances and normalizations. Built from reference datasets and literature. Limited scope.
- 6) Multimodal feedback by using virtual assets (e.g. avatar)
- 7) Incremental prototype set up.

Pose and spatial movement as input for dynamic content search & generation

Yoga-82 dataset

- 82 poses ~ 20k images
- Hybrid dataset published in 2020 → pictures for single and multiple practitioners, drawings and artistic pictures.



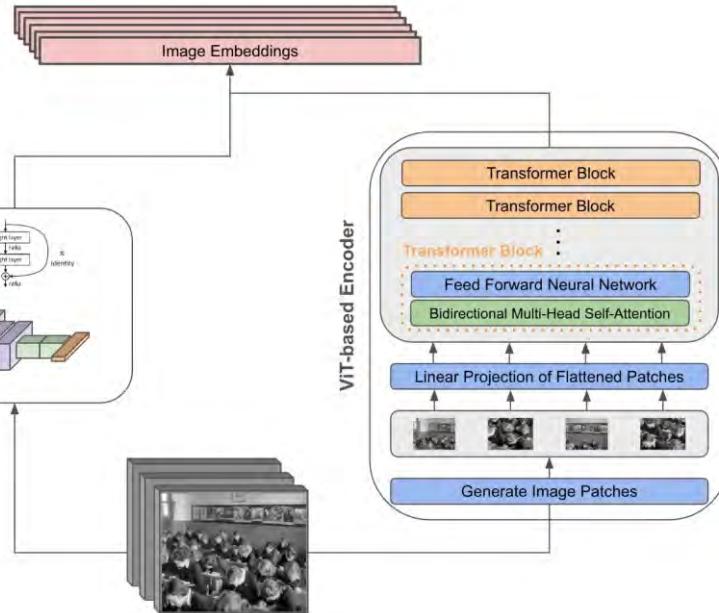
- Images provided as web links → Issue: unbalanced classes due to missing data

Pose and spatial movement as input for dynamic content search & generation

CLIP as classifier

- Our first approach has been to use CLIP as a classifier on a filtered 6-class dataset
 - Low zero-shot performance

ResNet-based Encoder



ViT-based Encoder

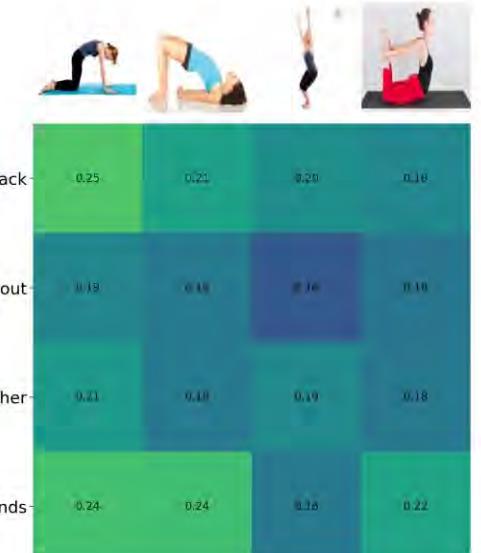
Image of a person leaning on the knees, facing forward or towards the ground, with a hunched back

Image of a person resting on his shoulders on his back, with his arms straight. May have one leg stretched out

Image of a person standing with knees bent. Non-stretched. His arms may be stretched out or with hands together

Image of a person resting on his stomach face down executing a yoga bow pose. He touches his feet with his hands

Cosine similarity between text and image features

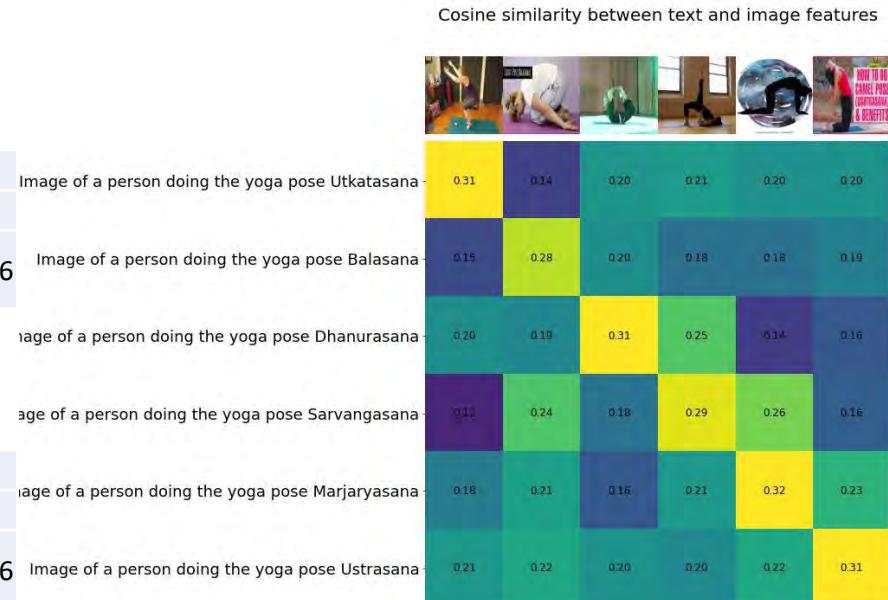


Pose and spatial movement as input for dynamic content search & generation

CLIP as classifier

- Fine-tuning on small-clean and complete-raw datasets**
- Significant improvement after fine tuning**

Model	RAW 6 CLASSES		Test Precision	Hyperparameters						
	Train images	Test images		Epochs	Batch size	Learning rate	Weight decay	Adam β_1	Adam β_2	Adam ε
ViT-B/32	1301	326	0,988	5	6	1,00E-05	1,00E-03	0,9	0,98	1,00E-06
			0,972			5,00E-06	1,00E-03			



Model	CLEAN 6 CLASSES		Test Precision	Hyperparameters						
	Train images	Test images		Epochs	Batch size	Learning rate	Weight decay	Adam β_1	Adam β_2	Adam ε
ViT-B/32	434	109	1,000	5	6	1,00E-05	1,00E-03	0,9	0,98	1,00E-06
			0,972			5,00E-06	1,00E-03			

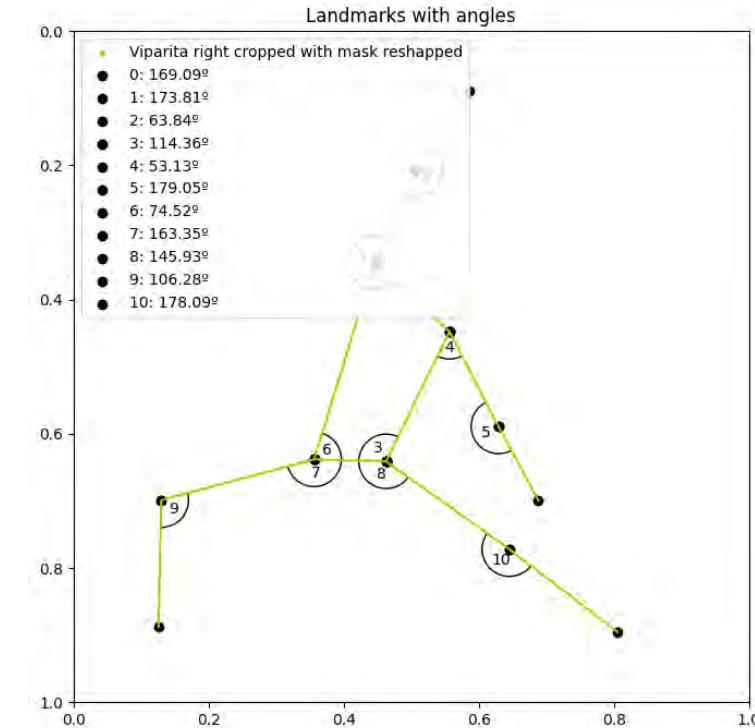
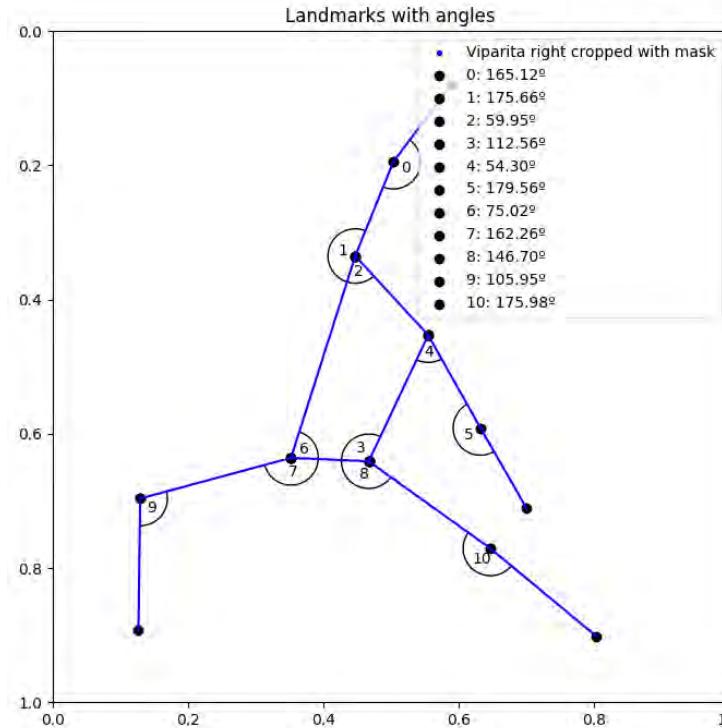
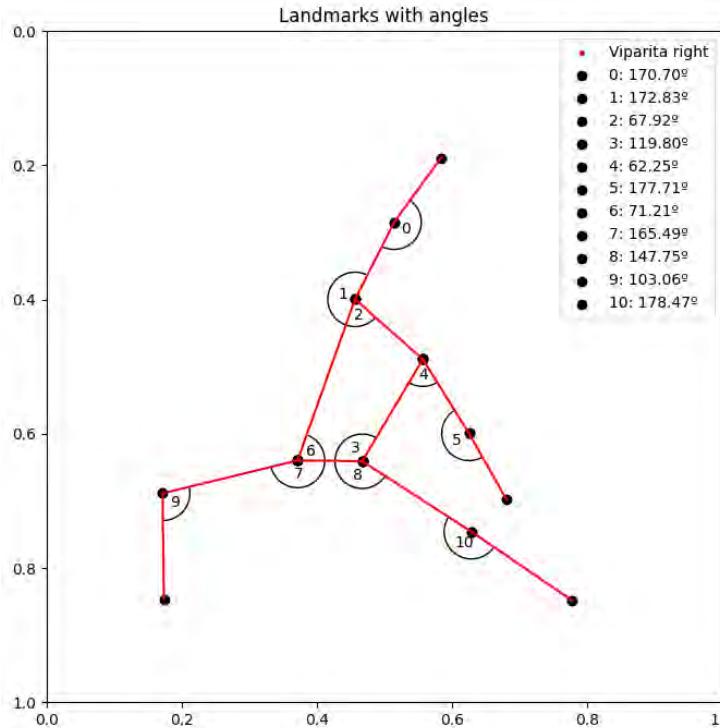
Precision on 82 poses RAW
dataset → 85,2%

Model	RAW 82 CLASSES		Test Precision	Hyperparameters						
	Train images	Test images		Epochs	Batch size	Learning rate	Weight decay	Adam β_1	Adam β_2	Adam ε
ViT-B/32	15301	3826	0,852	5	82	1,00E-05	1,00E-03	0,9	0,98	1,00E-06

Pose and spatial movement as input for dynamic content search & generation

Mediapipe for pose evaluation

- **Mediapipe for landmarks extraction and angle computing**
 - Some problems: sensitive to face position, scale and orientation in the input image
 - For the same input image, we obtain different angles depending to the preprocessing applied



Pose and spatial movement as input for dynamic content search & generation

Mediapipe for pose evaluation – Angle difference (error) estimation depending on preprocessing

- **Approximation to the problem:**

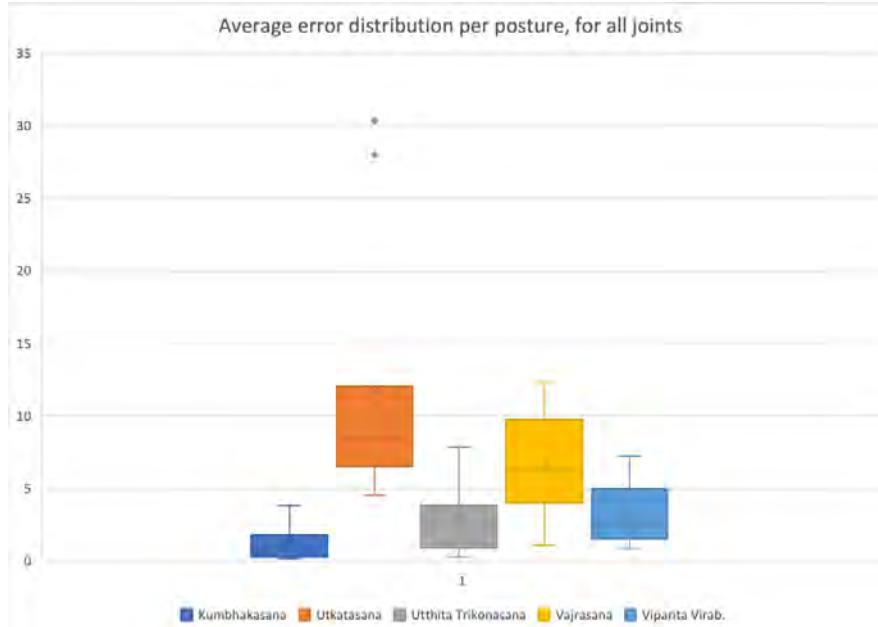
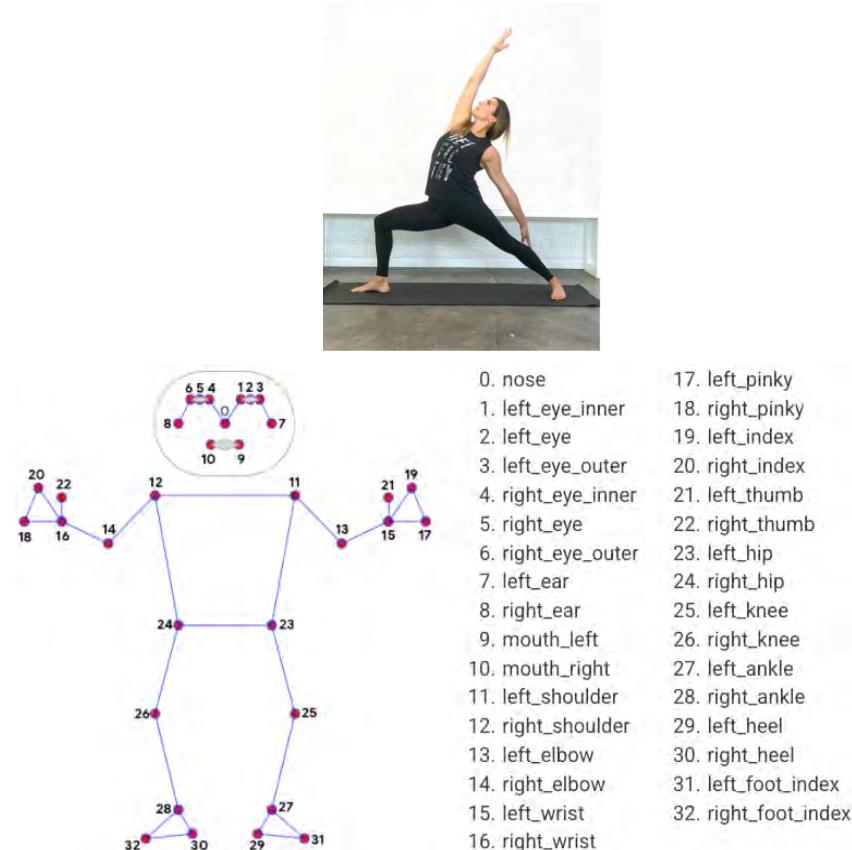
- Manual labeling of landmarks for 5 correct postures in a group of images
 - 1 image labeled by posture → Ground truth
- Computing the error between angles for different image preprocessing



Pose and spatial movement as input for dynamic content search & generation

Mediapipe for pose evaluation

- **Results:**
 - Smaller error when only the segmentation mask is applied. No resizing



Per posture

Average error by joint	Std deviation by joint	Joint
5,349	15,156	(12,14,16)
1,455	35,886	(14,12,24)
1,854	11,774	(24,12,11)
5,004	13,303	(11,23,24)
1,534	37,810	(23,11,13)
0,878	9,678	(11,13,15)
2,356	11,466	(23,24,12)
7,234	19,214	(26,24,23)
3,662	12,117	(24,23,25)
3,395	13,383	(28,26,24)
1,651	15,361	(23,25,27)

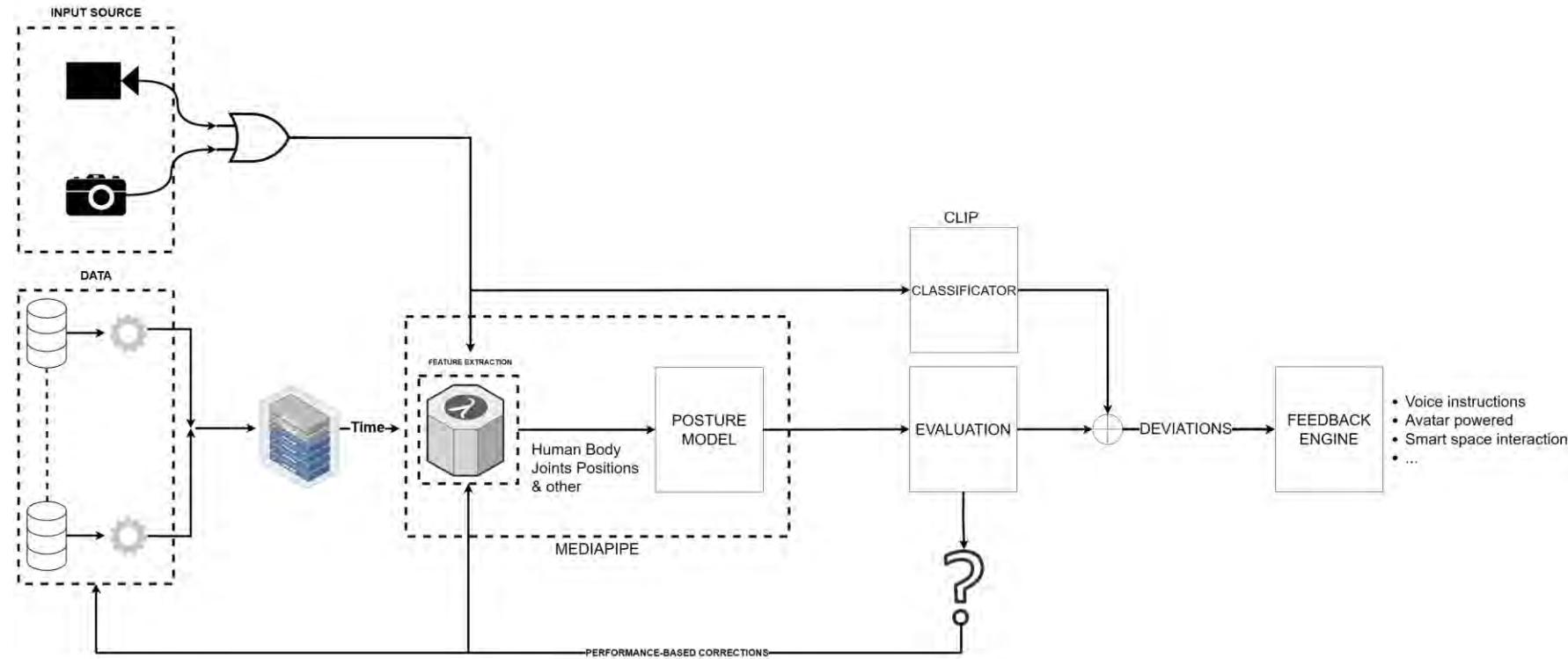
Per joint

Analysis based on 66 images by posture

Pose and spatial movement as input for dynamic content search & generation

Next Steps

- Evaluating feasibility of differentiating between "correct" and "wrong" postures with the angle difference approach and silhouette overlap.
- Setting-up a proof-of-concept system prototype with (some) feedback on posture evaluation.
- Bonus task: Check performance freezing layers and other CLIP models for classification.



Speaker diarization with multimodal inputs

1.- Brief Recapitulation



Face1

Face2

FaceN



2.- Experiments proposed

1

Image: Frames from a video at 5 fps

Audio: Corresponding slices (200 ms)

Encoders: ResNet50 both for audio and image

3

Image: Recurrent network using 5 fps videos

Audio: Using the audio correspondent to the 1 second video

Encoders: ResNet50 for audio and recurrent network image

2

Image: Frames from a video at 25 fps

Audio: Corresponding slices (40 ms)

Encoders: ResNet50 both for audio and image

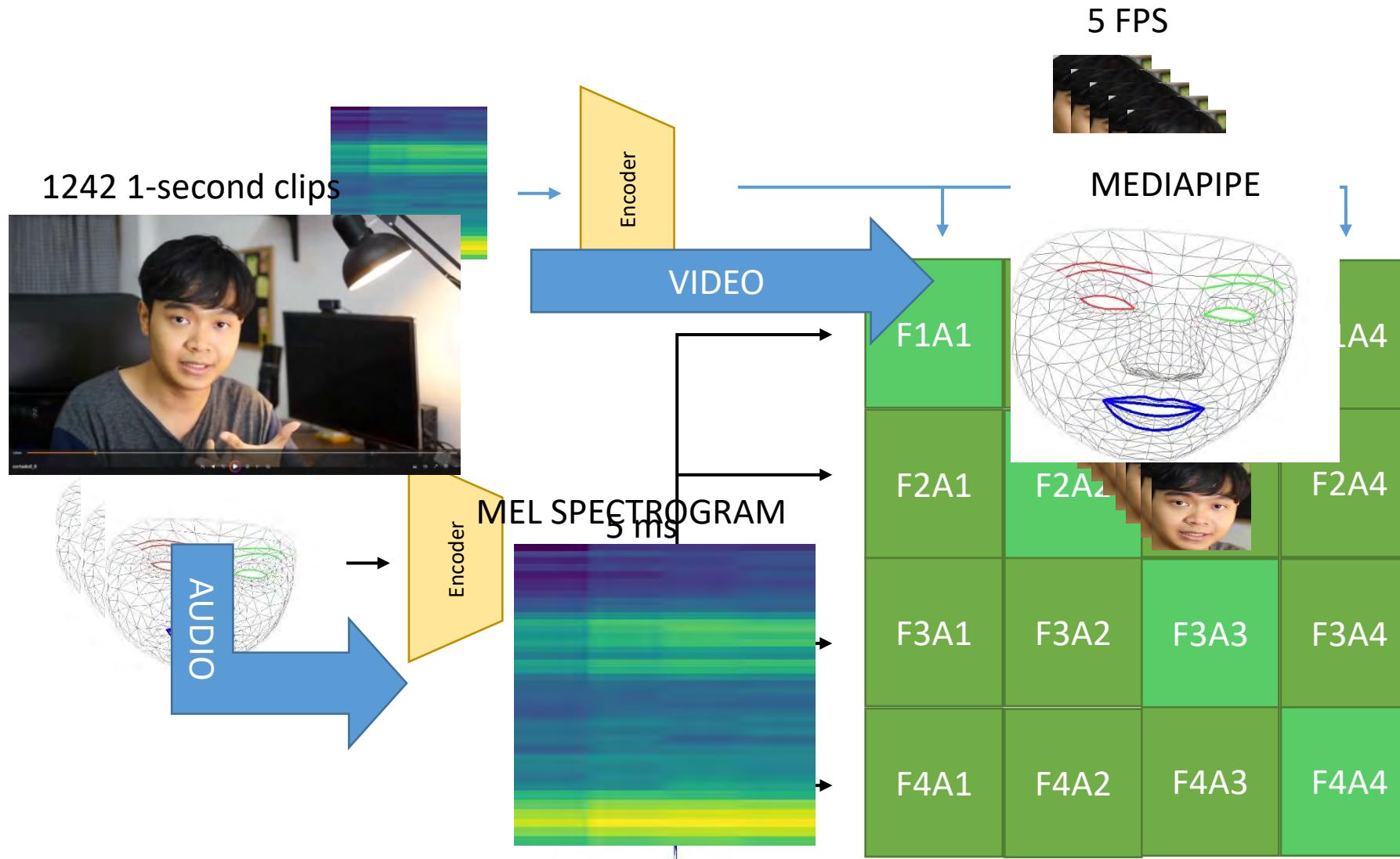
4

Inputs: Use the landmarks of the face in a DNN network

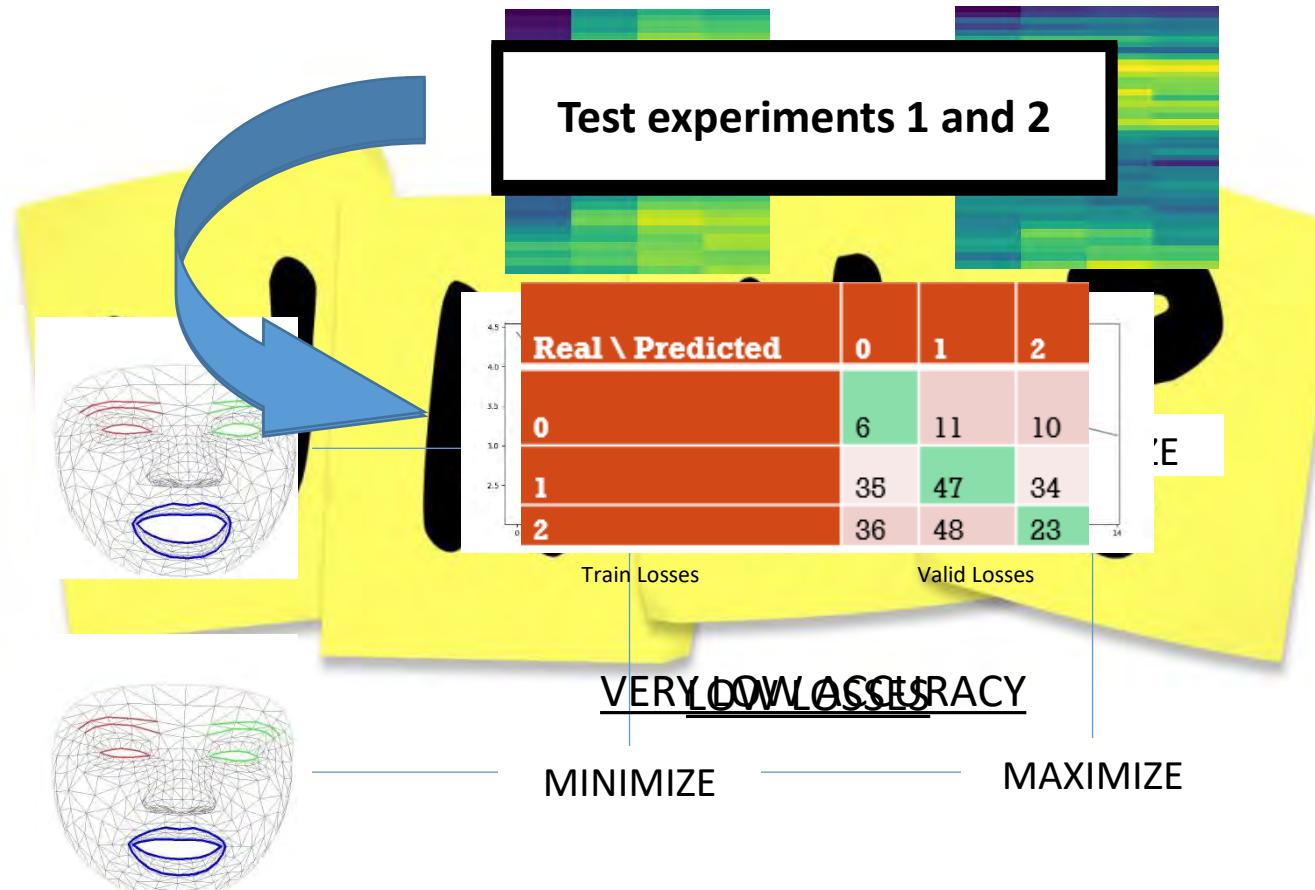
Audio: Use the corresponding audio slice

Encoders: ResNet50 for the audio

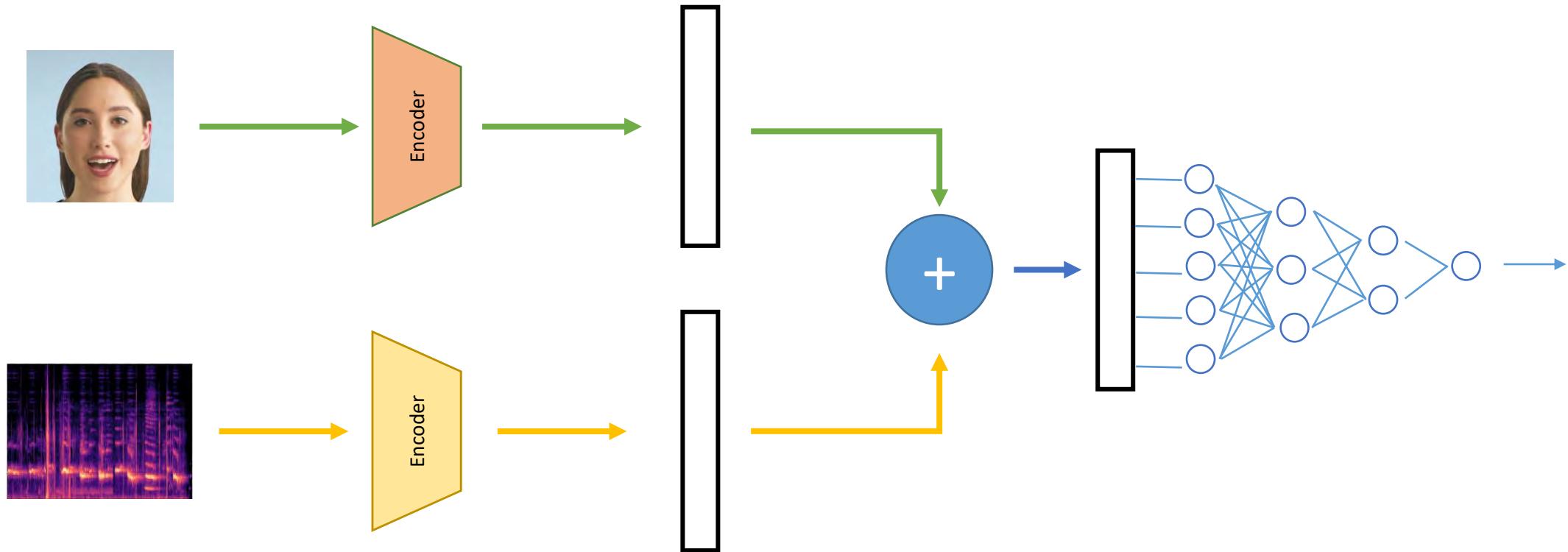
3.- Progress so far



4.- Problems found



5.- Solutions



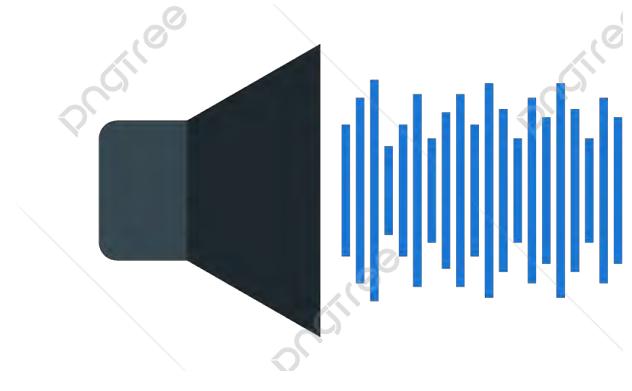
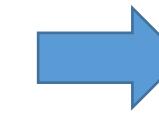
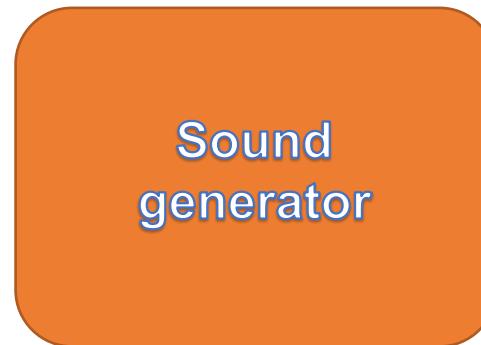
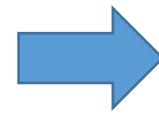
6.- Conclusion

- ❖ Due to the fact that CLIP is contrastive it may be harmful for our goal
- ❖ Implementing a DNN using the concatenation of the embedding both from audio and image may be a good approach
- ❖ Our problem can be reduced to a binary classification network
- ❖ The main difficult would be to achieve good accuracy by adjusting the models used by the encoders and to obtain a good-performing network

Audio Generation using Deep Learning

Sound generation

Our goal



Dog generated sound

Sound generation

What we have been working on



Datasets

Audio dataset & Video dataset



Initial Database

Source	Dog	Cat	Bird	TOTAL
Freesound	2015	312	909	3236
VGGSound	592	0	0	592
Imagenet	5180	4015	4149	13344

Final Database

Source	Dog	Cat	Bird	TOTAL
VGGSound orginal	4072	3824	3604	11500
Corrupt	556	2067	2762	5385
Non-corrupt	3516	1757	842	6115

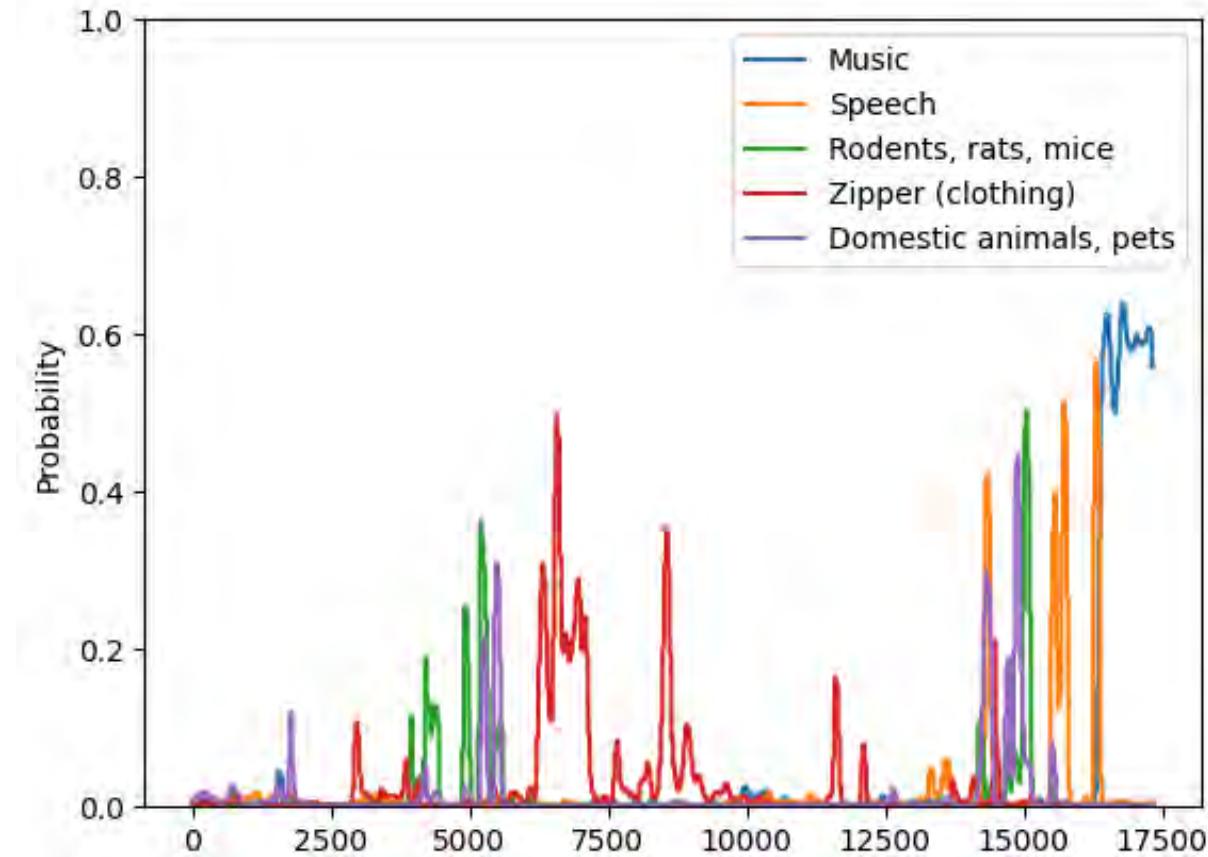
Dataset segmentation

Sound Event Detection

Downloaded Video



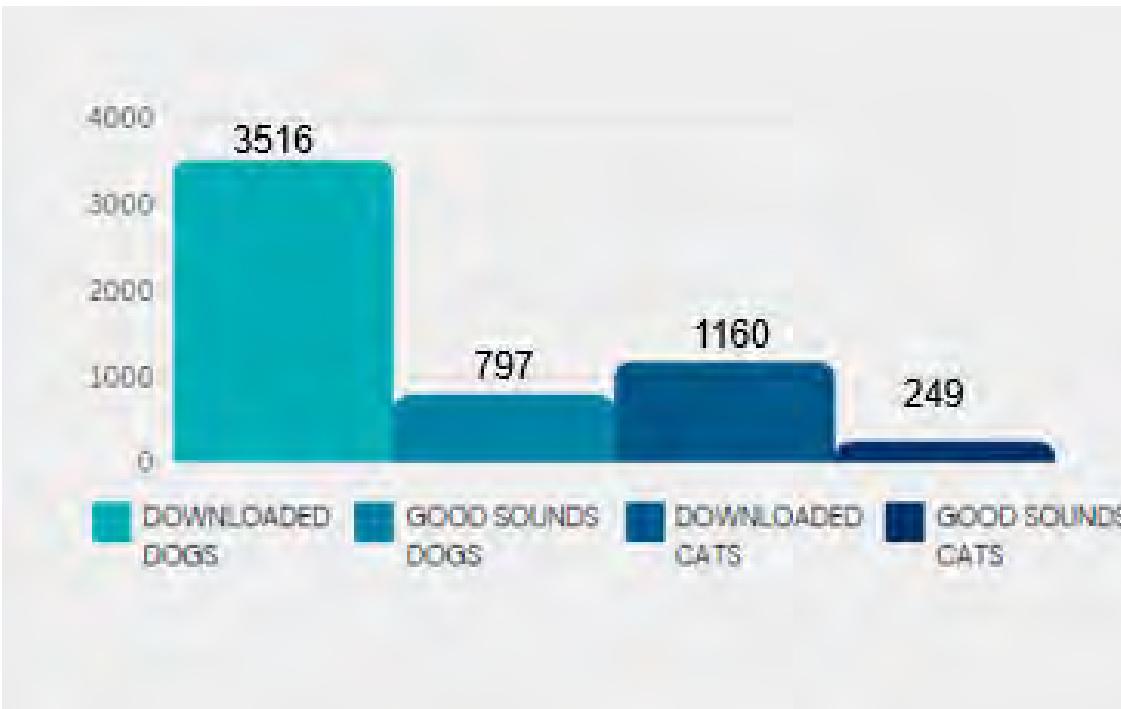
Using the Probabilistic Artificial Neural Network (PANN)



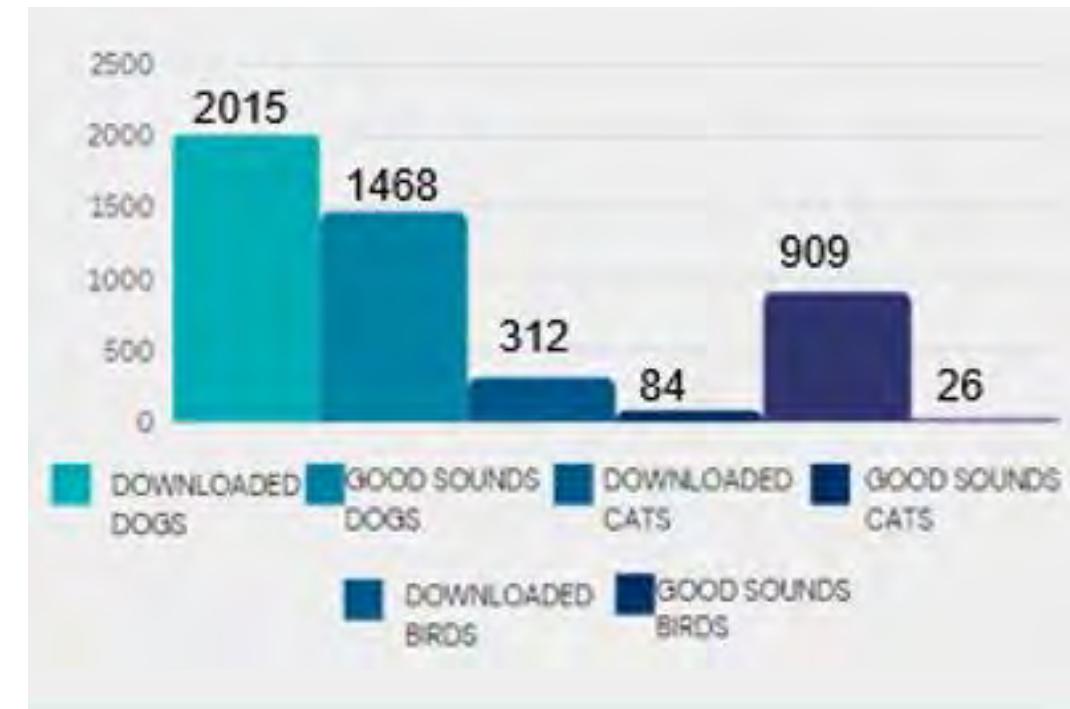
Dataset

Statistics

Audio preprocessing of videos (YouTube VGGSound)

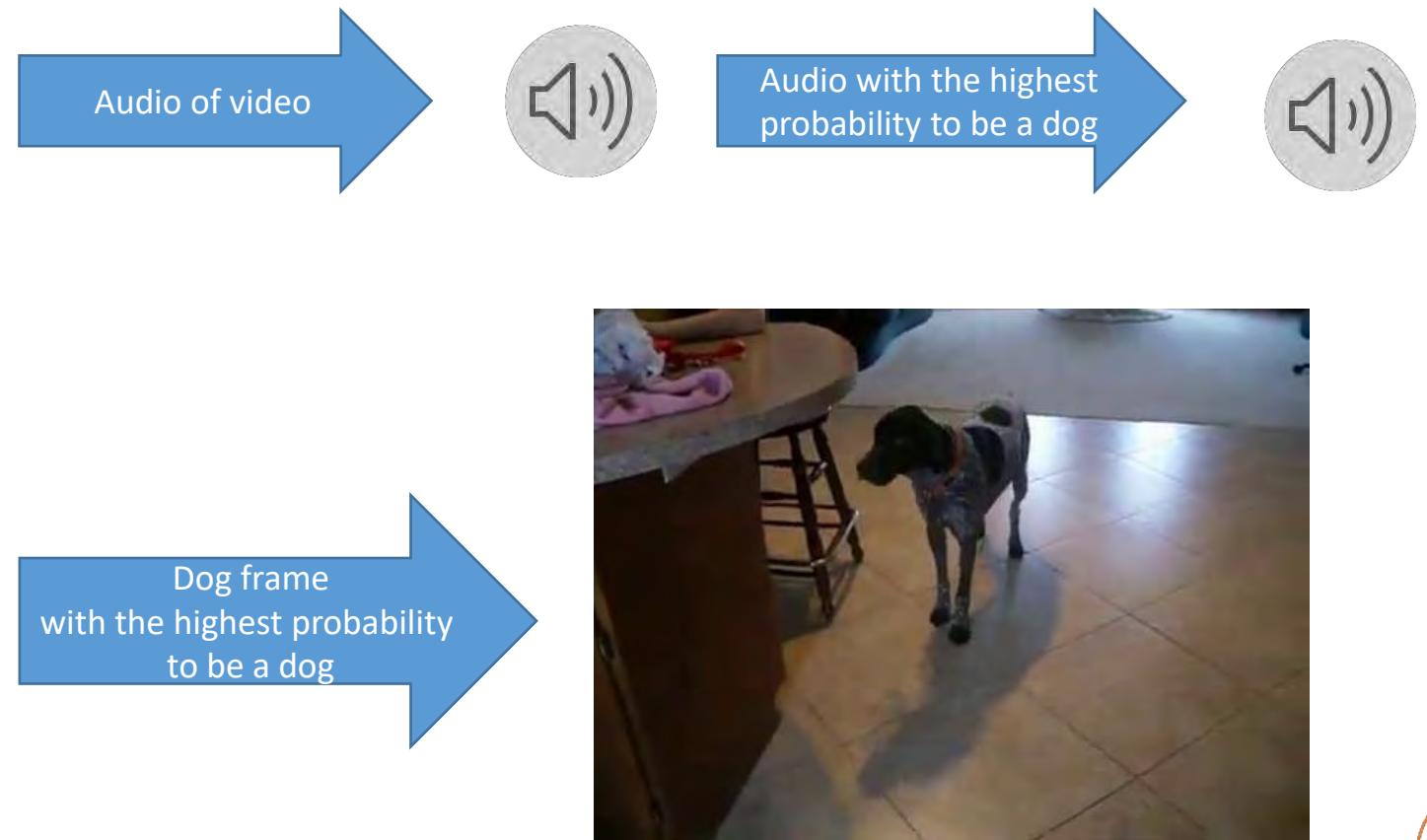
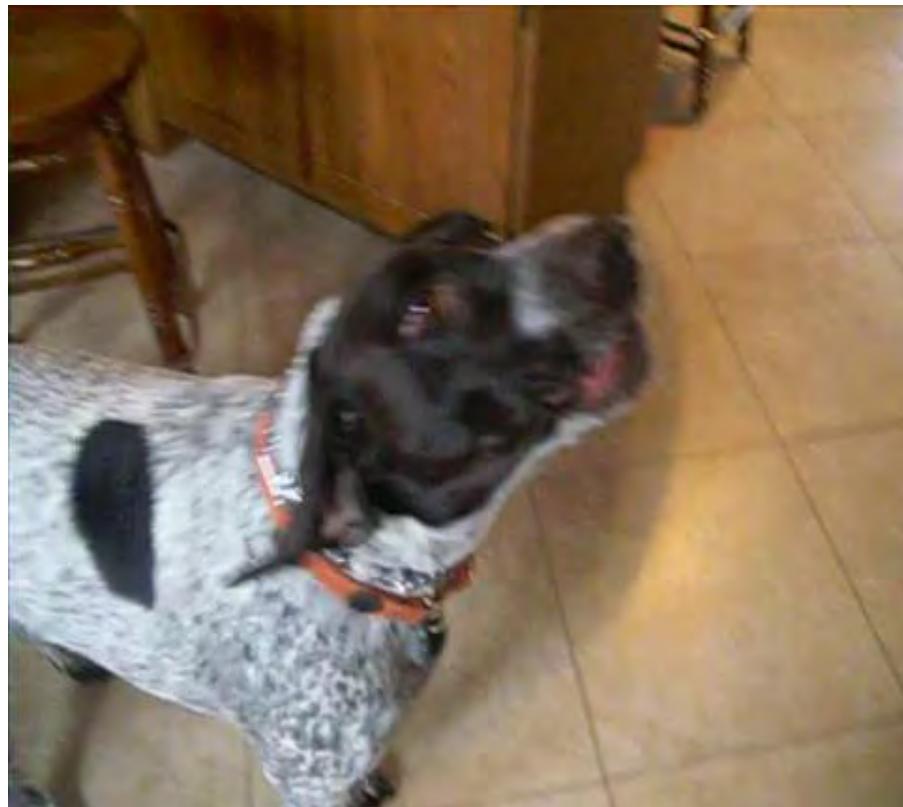


Audio preprocessing of Freesound



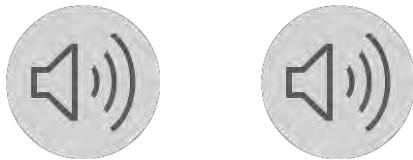
Dataset

Examples of data

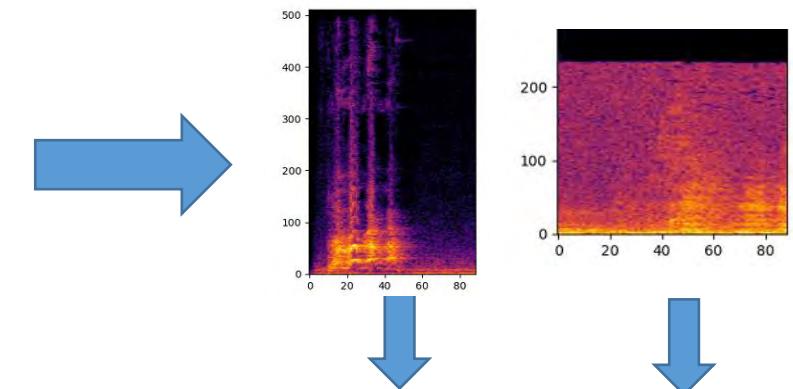
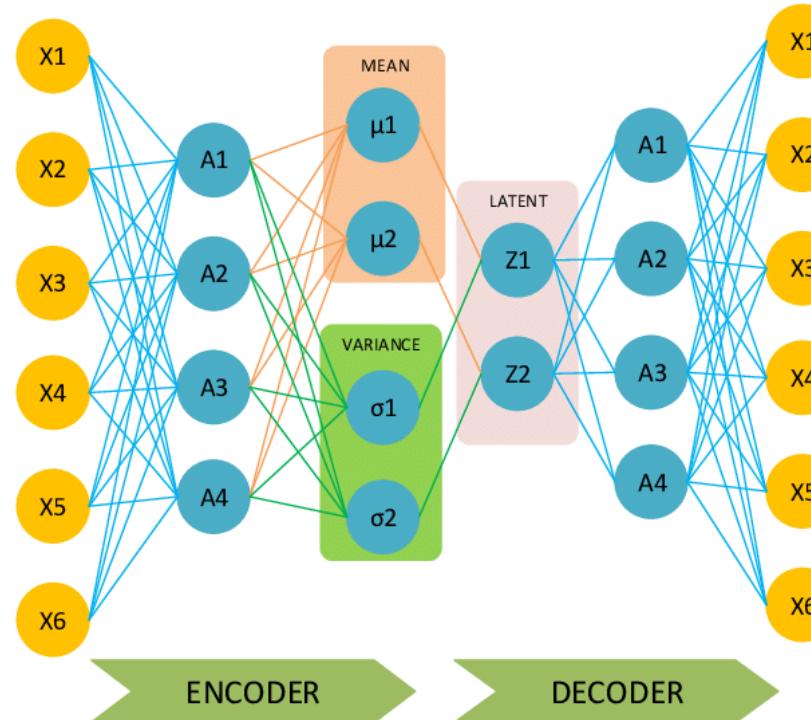
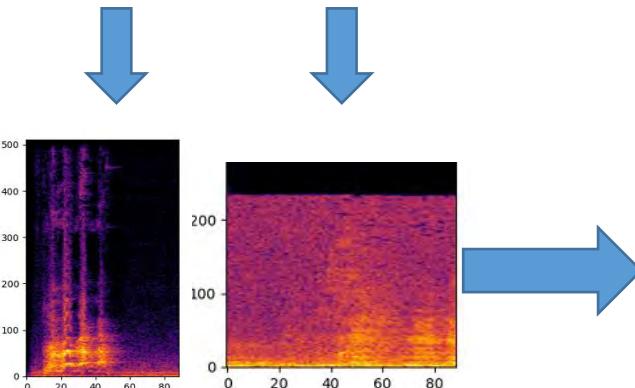


Variational Autoencoder

Example from our dataset



Preprocess_audio_VAE.py



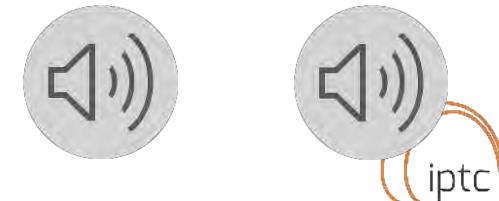
Features

Number of epochs = 2000

Latent dimension=20

Number of layers= 5 Convolutional Layers

Griffin-Lim algorithm



Variational Autoencoder

For audio

Laurafdez / VAE-MODEL Public

Code Issues Pull requests Actions Projects Security Insights

main 1 branch 0 tags

Go to file Code

Laurafdez Update VAE.py be76503 2 weeks ago 13 commits

Preprocess_audio_VAE.py Create Preprocess_audio_VAE.py 2 weeks ago

README.md Update README.md 2 weeks ago

VAE.py Update VAE.py 2 weeks ago

animal_sound_detector.py Create animal_sound_detector.py 2 weeks ago

download_video.py Create download_video.py 2 weeks ago

video_to_audio.py Create video_to_audio.py 2 weeks ago

About

This repository will show how a continuous vae works in the generation of new audios from latent space.

Readme 0 stars 3 watching 0 forks Report repository

Releases

No releases published

Packages

No packages published

Contributors 2

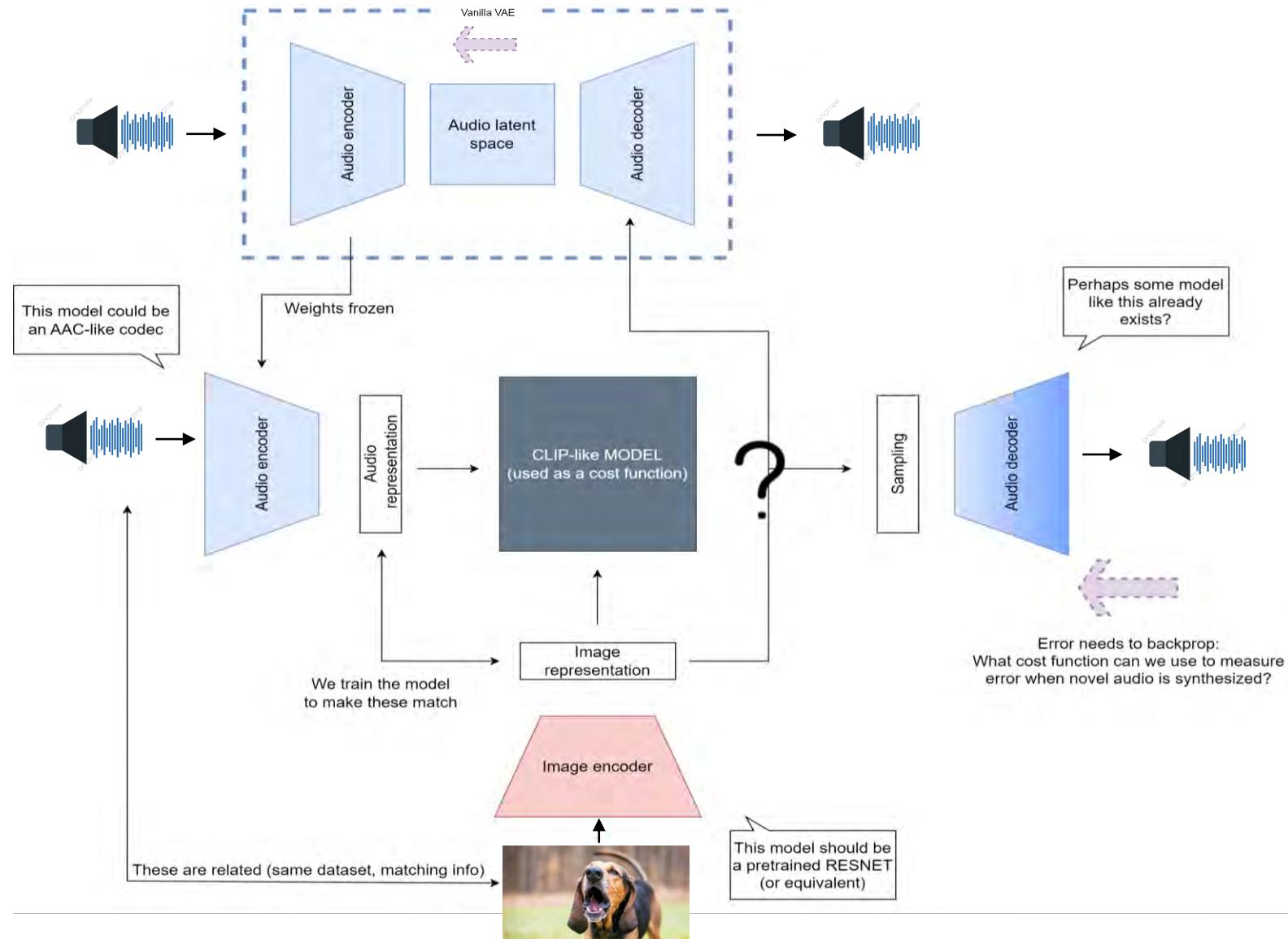
Laurafdez Laura Fernández Galindo
mariasanruiz

[Our GitHub]



How can we approach our goal?

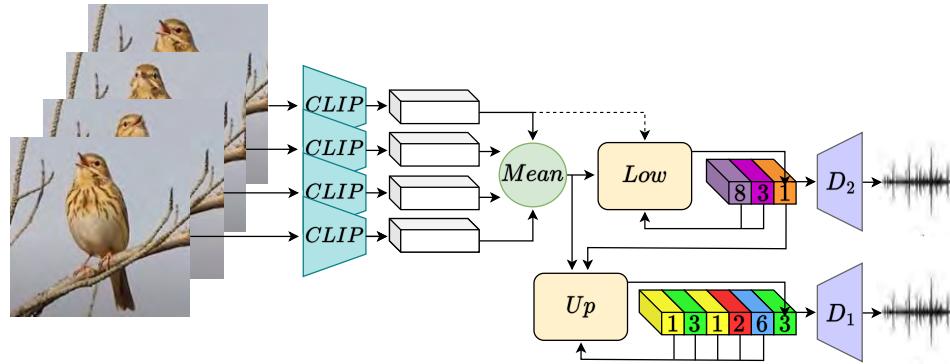
Our goal: generate audios from images. Discussion



Literature review: Discrete Latent Space

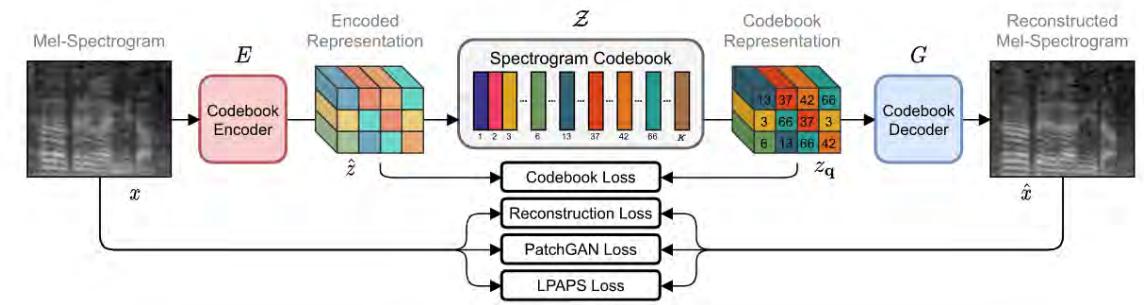
im2wav, SpecVQGAN

im2wav



[\[Paper\]](#) [\[Github\]](#)

SpecVQGAN

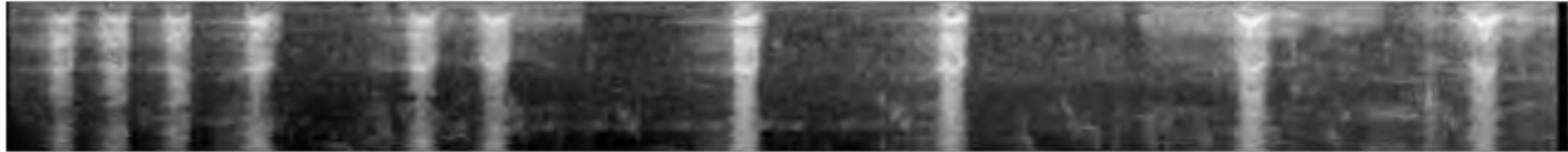


[\[Paper\]](#) [\[Github\]](#)

Example SpecVQGAN

Using a sample of our dataset

Original spectrogram:



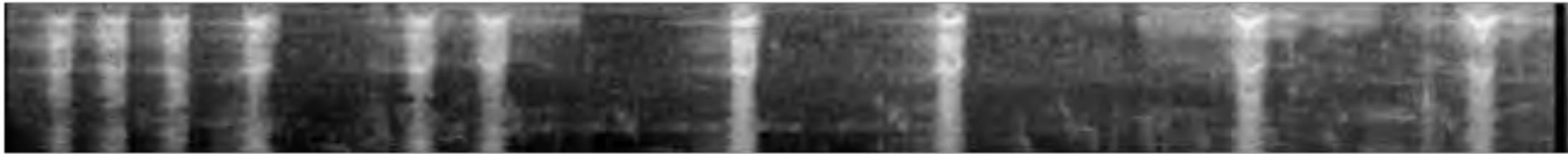
Reconstructed spectrogram:



Example SpecVQGAN

Random permutation of quantized representation

Original spectrogram:

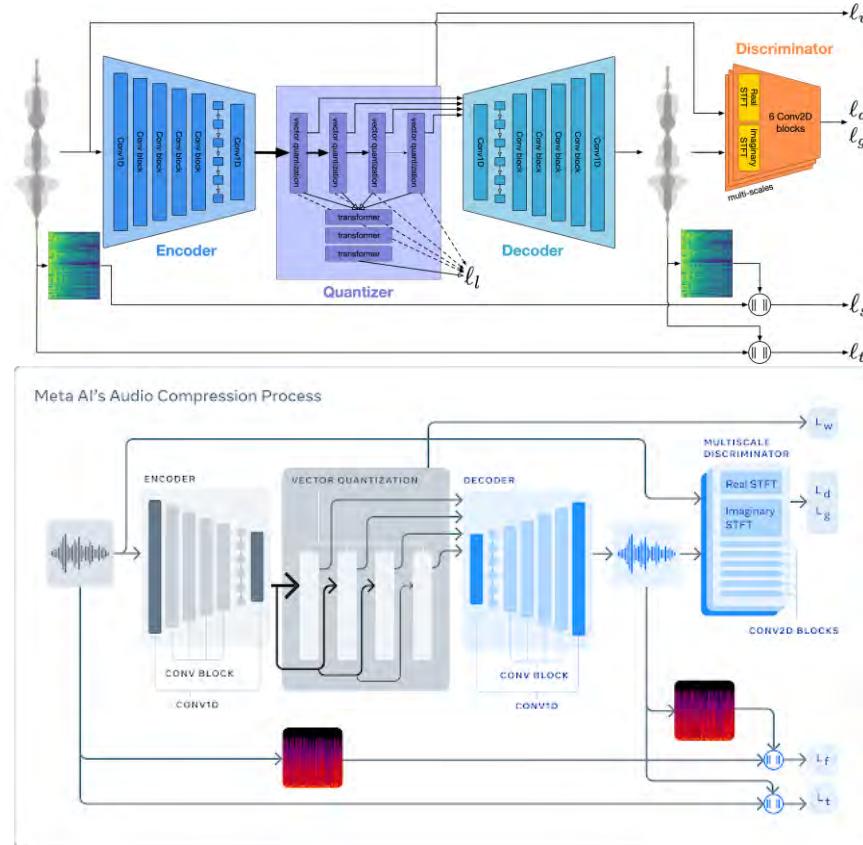


Reconstructed audio shuffled:

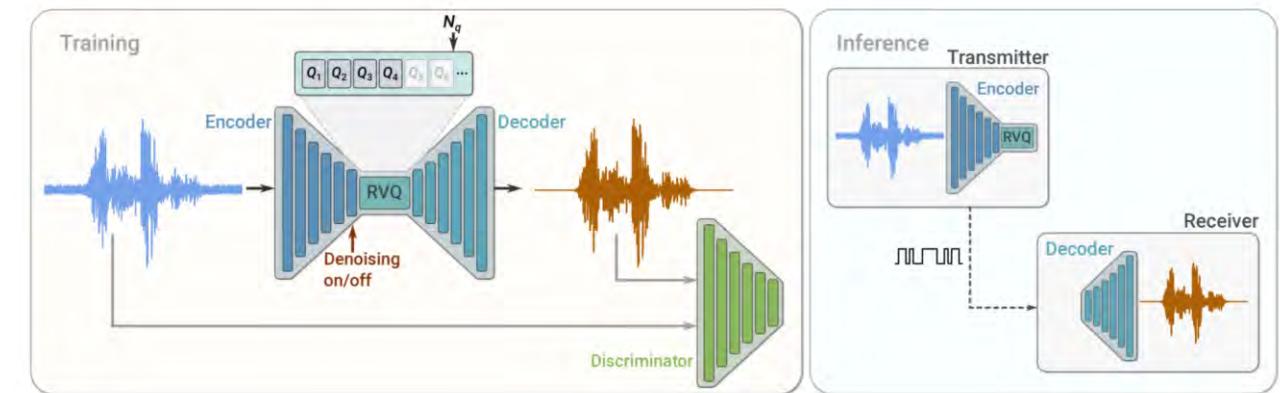


VQ-VAEs in High-Quality Audio Encoders

Encodec (Meta) + SoundStream (Google)



Encodec [[Meta](#)]

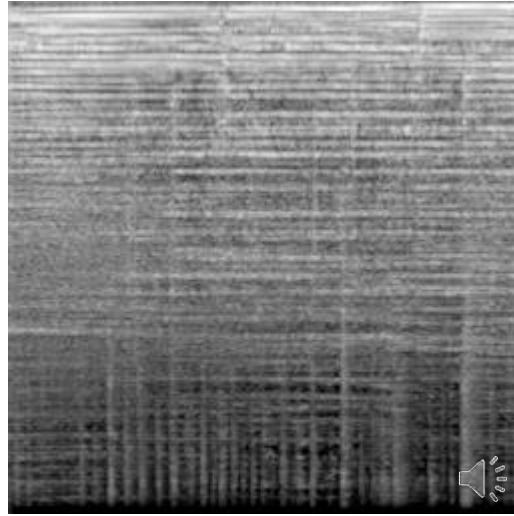


SoundStream [[Google](#)]

Literature review: Continuous Latent Space

Conditioning the latent space using diffusion models

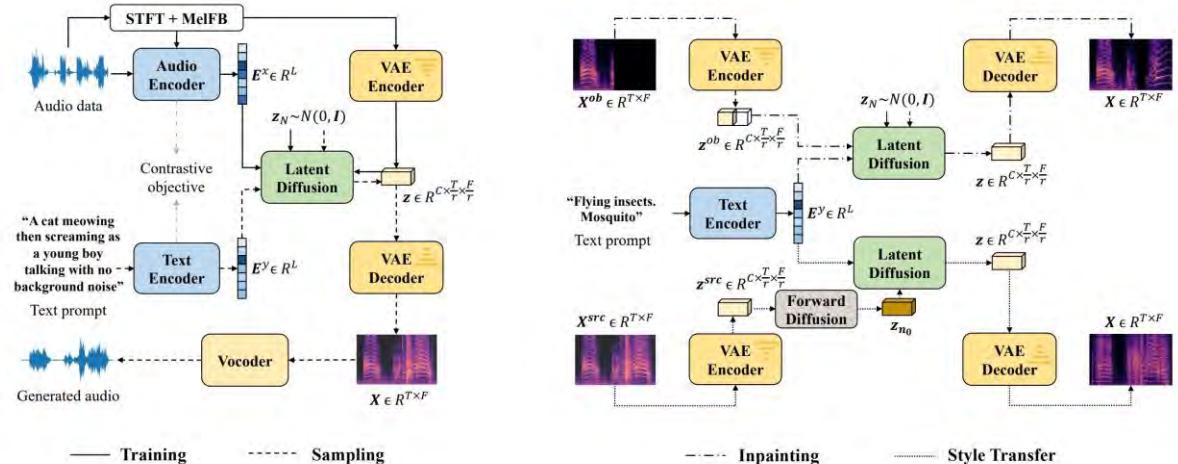
AudioDiffusion



teticio/latent-audio-diffusion-ddim-256

[\[Paper\]](#) [\[Colab\]](#) [\[Github\]](#)

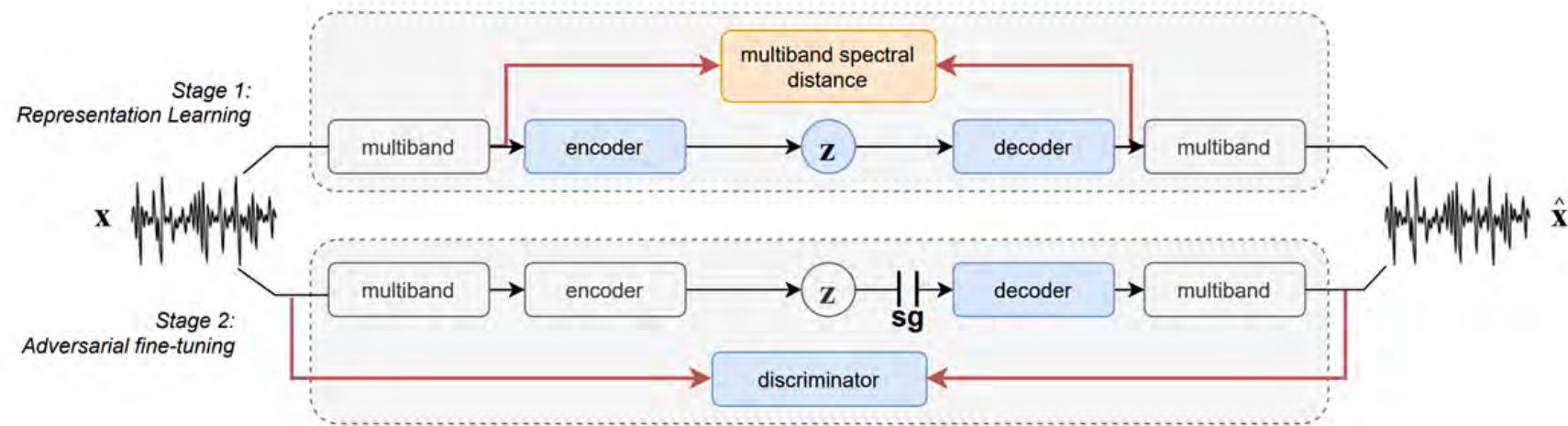
AudioLDM



[\[Paper\]](#) [\[Colab\]](#) [\[Github\]](#)

Continuous VAE in High-Quality Audio Synthesis

RAVE (Realtime Audio Variational autoEncoder)



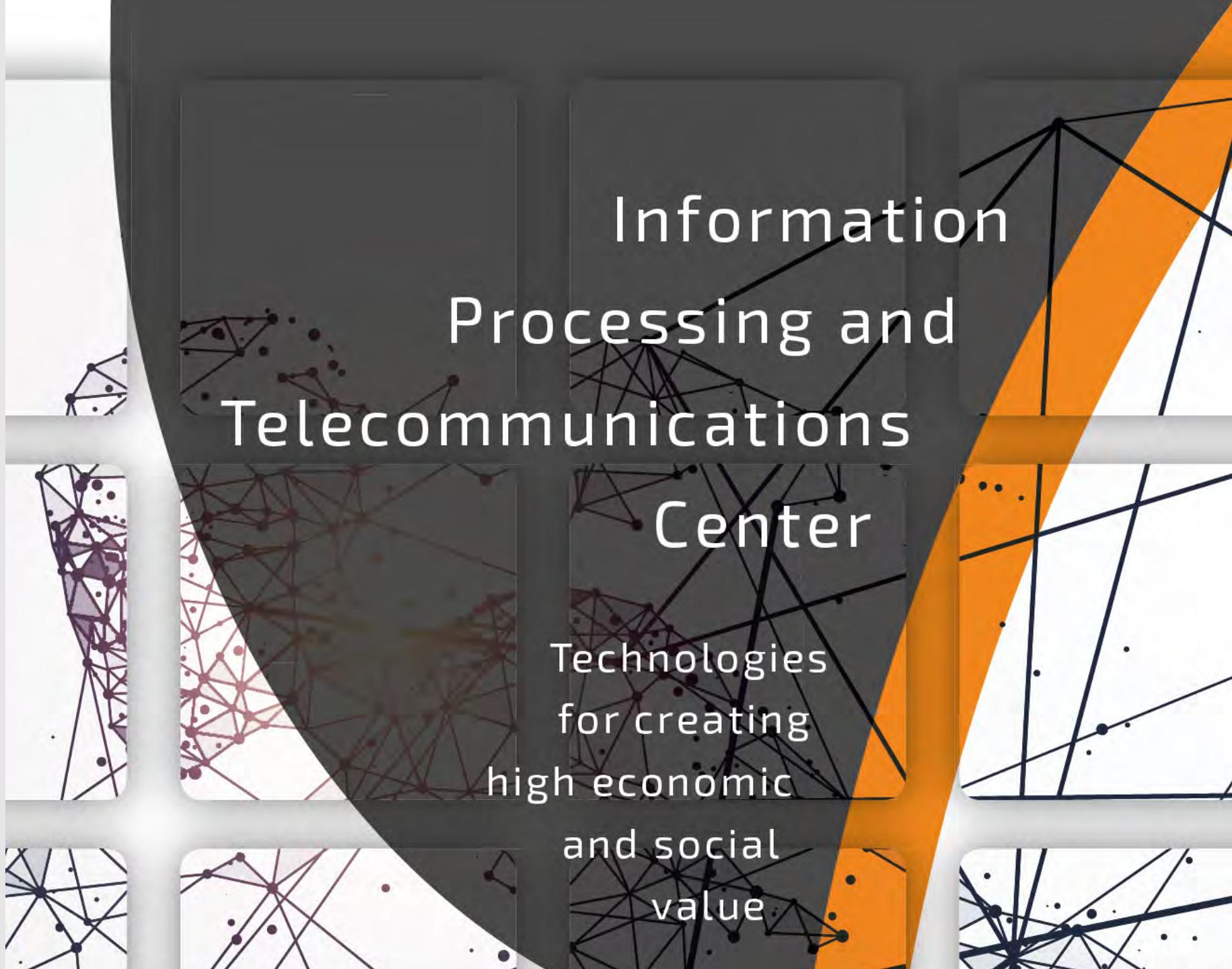
[[Paper](#)] [[Colab](#)] [[Available models](#)] [[DEMO](#)]

Many thanks



POLITÉCNICA

www.iptc.upm.es



Information Processing and Telecommunications Center

Technologies
for creating
high economic
and social
value