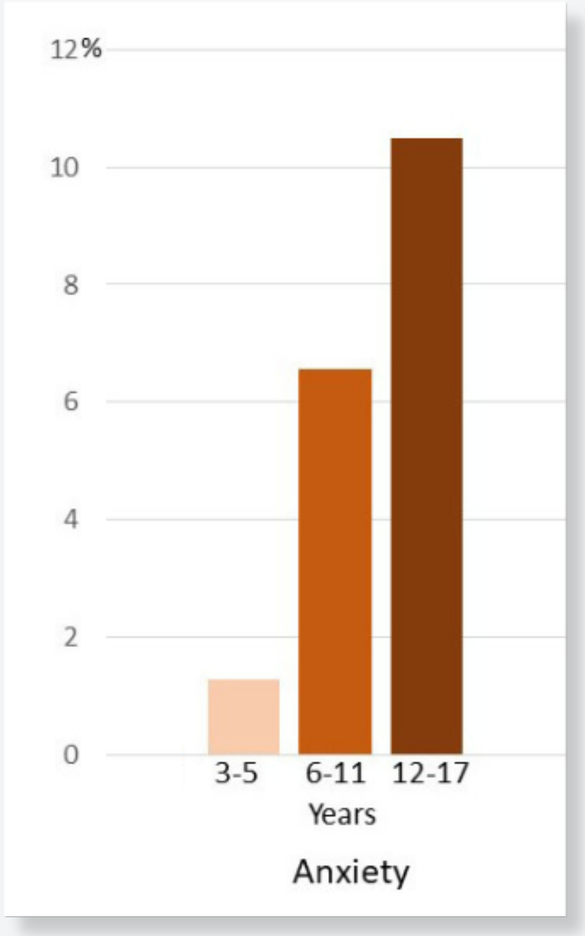
A DIVE INTO MINDFULNESS

Discover **tranquility** beneath the waves, delve into the **depths** of your **breath**, and emerge free from the **stresses of school life**

RICCARDO CALCAGNO X LUCA VIAN

NEEDS AND GOALS



The pupils' goal tackled is related to learning how to **manage anxiety**, especially derived from:

- **Bullying**
- **Interpersonal struggles**
- **Academic hardships**

This can be achieved through:

- Practice **deep breathing** and muscle relaxation
- Think of a **safe place**
- **Gradually** facing fears
- Praise and **reward** brave behavior

PERSONA

Name: Silvia

Personality Traits: Introverted, *observant*, strives to *make parents proud*.

Challenges: *Difficulty following her rhythm*, episodes of anxiety.

Desire for Guidance: *Seeks help* to navigate challenges and *develop strategies*.

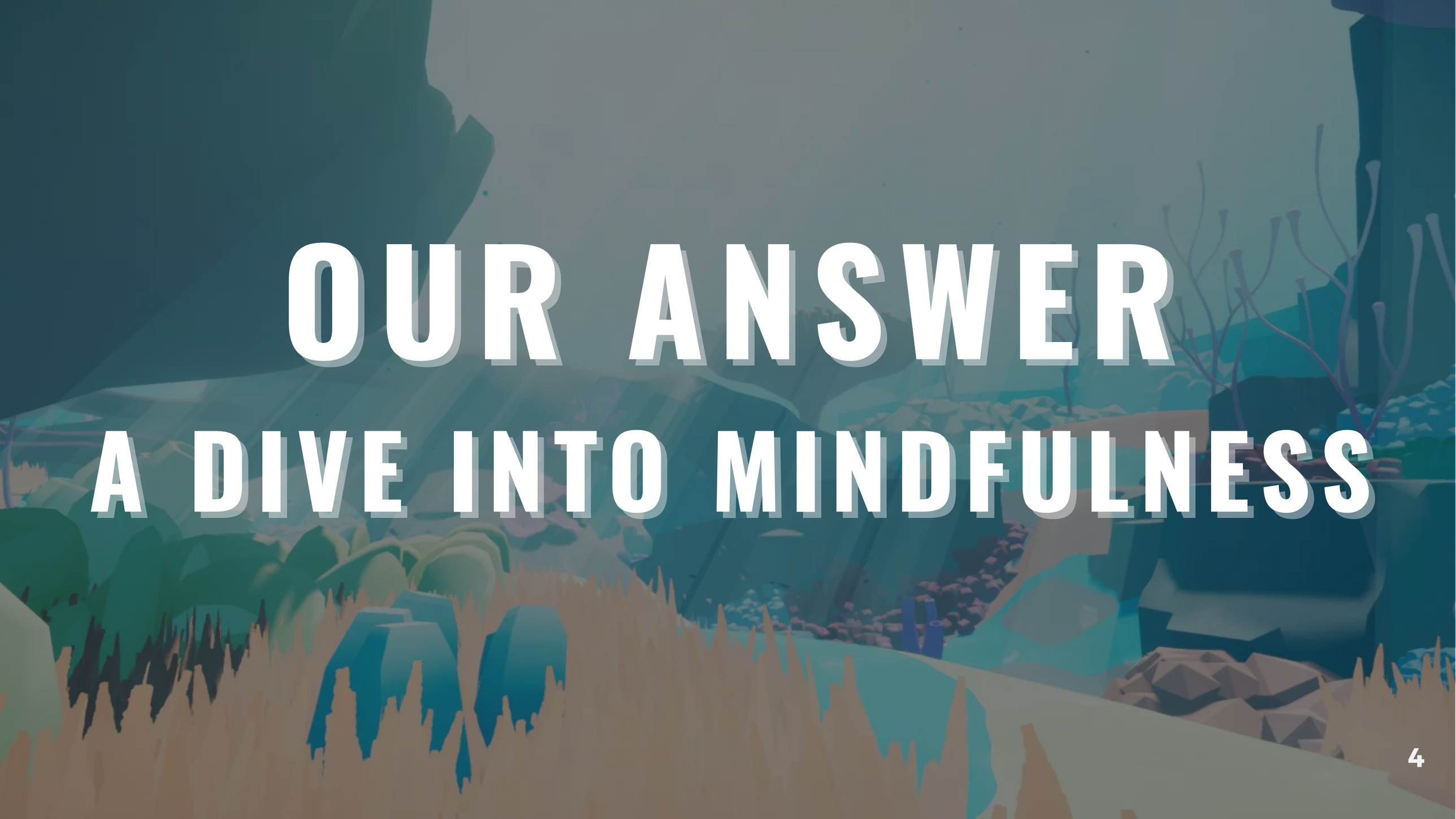
Support System: Needs *assistance* in identifying *triggers and managing anxiety*

Lack of Support: *Missing peer support* in new environment. Struggling with unfamiliar teaching style

Exam Anxiety: Fear of *speaking during exams*, leading to anxiety episodes and *speechlessness*

Recent Change: *Moved houses, changed schools mid-semester*.

Academic Pressure: *Struggling to adapt to new school*'s high pace, especially oral exams



OUR ANSWER

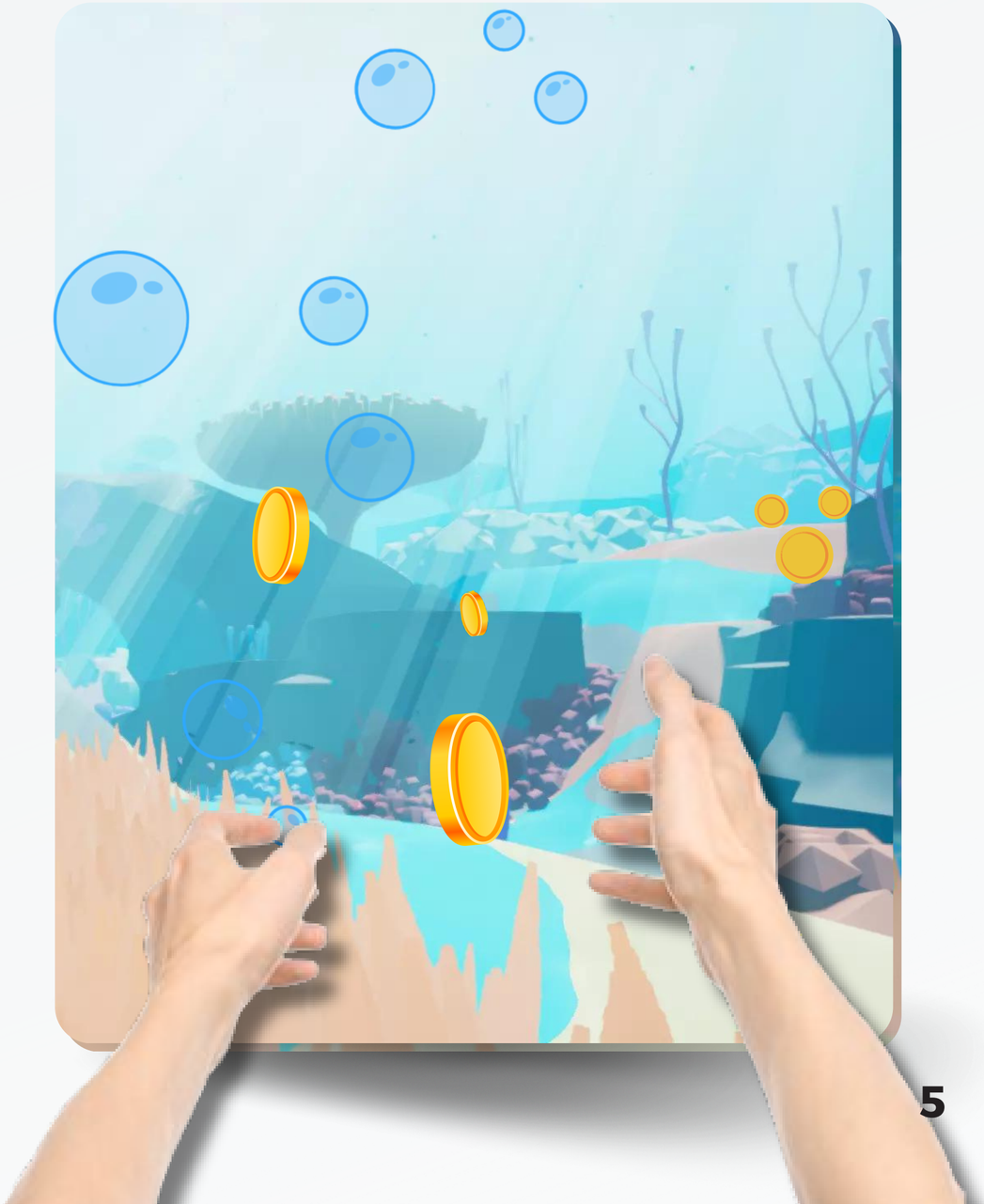
A DIVE INTO MINDFULNESS

DESCRIPTION

Projected in 1° person perspective on a screen, the serious game unfolds to children an underwater world **navigable** with the **movement of arms**, **gaze** and **breathing**.

The game challenges them to **explore** and **collect coins** hidden at **different depths**.

Relaxed breaths allow the avatar to **dive deeper**, adding a **mindful** element to the underwater adventure



INTEGRATING THE EXPOSURE

The aim of this SG is to gain confidence in **self-relaxation**.

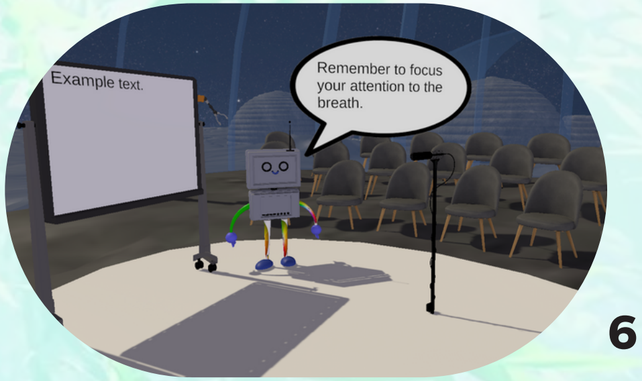
This further increment aims to **expose** the child to the same **stressing triggers:**

Homeworks, **exams**, **study plan,** ecc.

Underwater station, a link to the school world

Customizable Activities

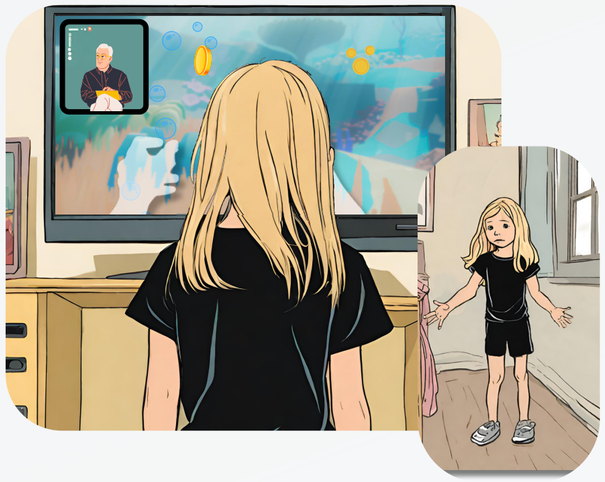
- Calls with tutors
- Edit study plan.
- Public speach exposure therapy



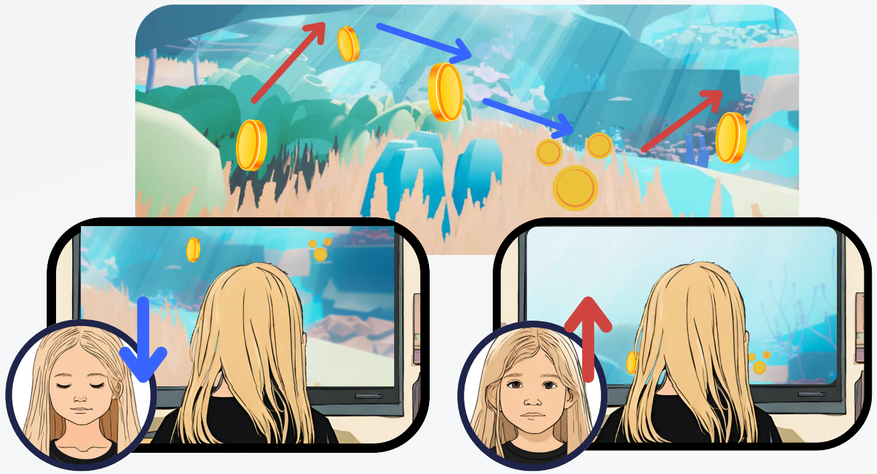

SILVIA'S
EXPERIENCE

Silvia describe to her tutor the **fears** about the **oral exams**

They play a public speaking simulation but Silvia does **not feel good** at all



Silvia needs a dive into the game and **reconnecting** with her **calm breathing**

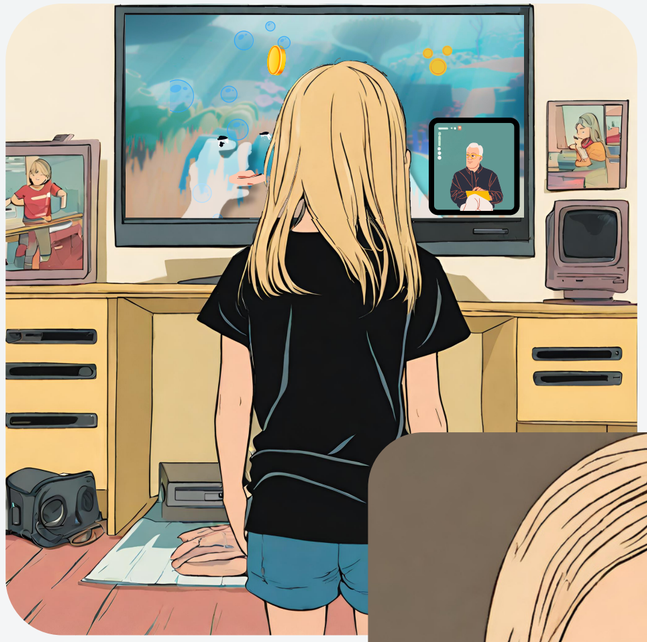


"*So satisfying the sound of coins collected*", she just needs to play between a **deep** state of **relaxation** and one more **active**

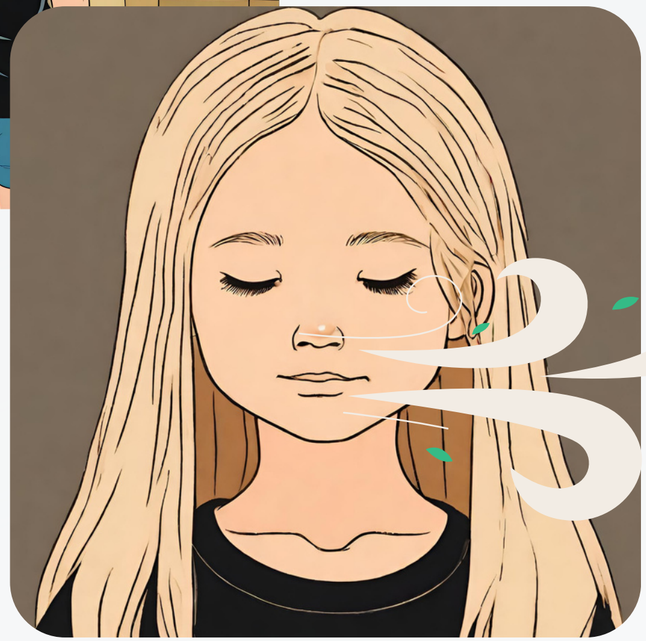
"*I should just **force my breath** into the correct rhythm*" (she thinks). Luckily for her, the system can not be tricked

"Subtle high frequencies detected!"

Silvia starts **following** the meditative **sounds** provided by the app. And after a little bit more **patience**, she is in the **flow**...



Now Silvia has a safe spot where to take a breath between stressful exercises...



...and that new gentle relationship that she has with her breath will lead her to feel her self everywhere...

...In particular, among her classmates.

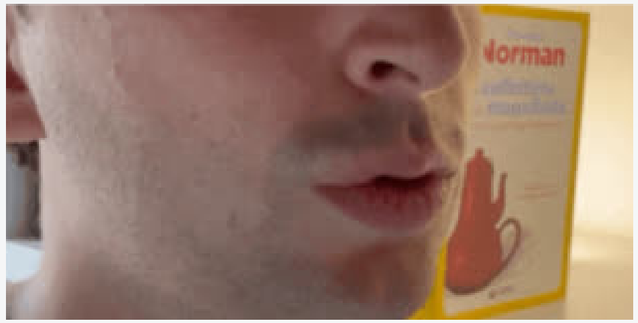
MODALITIES

BODY MOVEMENT

4 type of movement of both arms.
To direct the avatar: **towards**, **backwards**, pivoted rotation to the **left**, and **right**

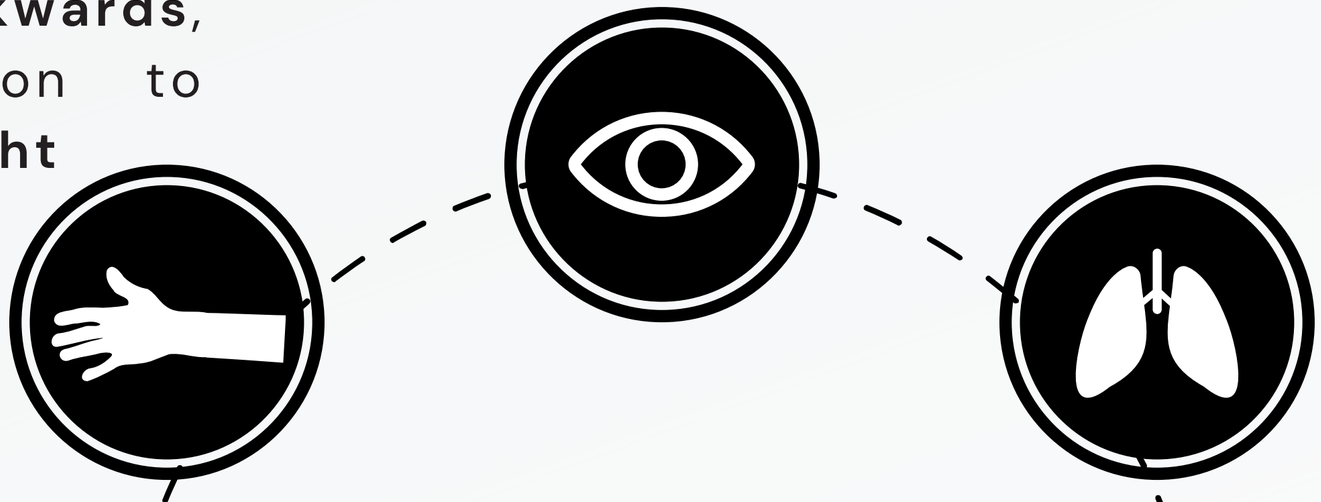
GAZE

Temporary, **constrained, delayed** rotation of the view. Allowing to perform curved trajectories.



BREATH SOUND

More relaxed it is more the avatar go in depth.



GESTURE CLASSIFICATION: SYMMETRY

Symmetry

- To distinguish **shift** (symmetric) from **rotation** (asymmetric).

```
def is_pose_symmetric(lhand, rhand, spine,threshold=0.22):

    h_asymmetry = compute_asymmetry(lhand, rhand, spine, "x")
    v_asymmetry = compute_asymmetry(lhand, rhand, spine, "y")

    asymmetry_score = (h_asymmetry + v_asymmetry) / 2

    if(asymmetry_score<threshold):
        return True
    else:
        return False
```

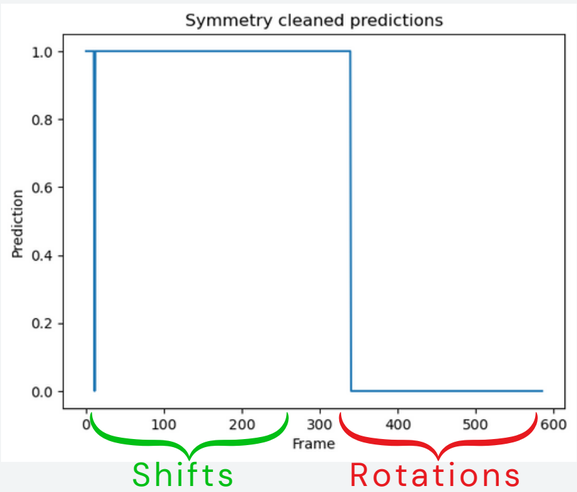
Where compute_asymmetry is implemented it in the following way:

$$hs(t_i) = \frac{\left||x^{tc}(t_i) - x^{LH}(t_i)| - |x^{tc}(t_i) - x^{RH}(t_i)|\right|}{|x^{tc}(t_i) - x^{LH}(t_i)| + |x^{tc}(t_i) - x^{RH}(t_i)|}$$

$$vs(t_i) = \frac{\left||y^{tc}(t_i) - y^{LH}(t_i)| - |y^{tc}(t_i) - y^{RH}(t_i)|\right|}{|y^{tc}(t_i) - y^{LH}(t_i)| + |y^{tc}(t_i) - y^{RH}(t_i)|}$$

Input Window: Symmetry computed considering one frame at the time

Output Window: Final prediction computed by checking the most frequent prediction in the last_frames_considered ones (at the moment initialized to 50)



GESTURE CLASSIFICATION: TRAJECTORY

Trajectory

- To distinguish between the **toward** and **backward** movement and between the **left** and **right** one.

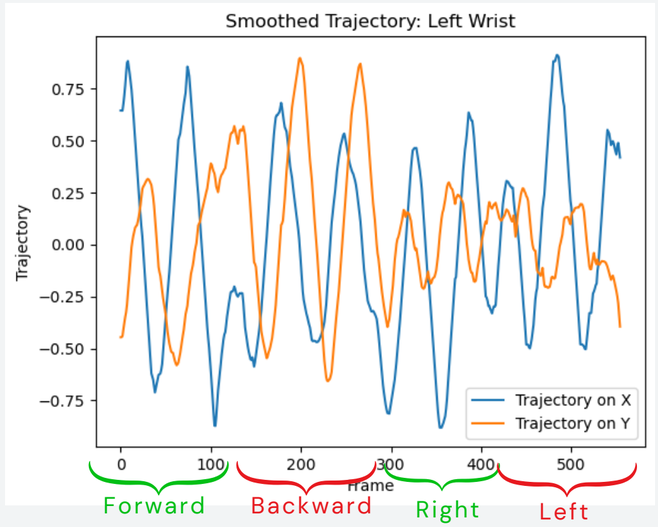
```
def compute_trajectory(lhand_past, lhand_now, rhand_past, rhand_now):
    direction_lhand_x =
    round((lhand_now["x"]-lhand_past["x"])/math.sqrt((lhand_now["x"]-
    lhand_past["x"])**2+(lhand_now["y"]-lhand_past["y"])**2),2)

    #in the same way is computed direction_lhand_y,
    direction_rhand_x, and direction_rhand_y

    return [direction_lhand_x,direction_lhand_y],
    [direction_rhand_x,direction_rhand_y]
```

So, the formula used to compute the trajectories is

$$\boldsymbol{d}^k(t_i) = \frac{\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-1})}{\|\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-1})\|}$$

Input Window: Trajectory computed considering comparing the actual position with the one of last_frames_considered frames ago (last_frames_considered is initialized to 2)

Output Window: Final prediction computed by smoothing the raw one with a convolution composed by the average over the window of window_size elements (at the moment initialized to 30)



Forward shift
A symmetric movement that has a higher **horizontal** component in the trajectory

Backward shift
A symmetric movement that has a higher **vertical** component in the trajectory

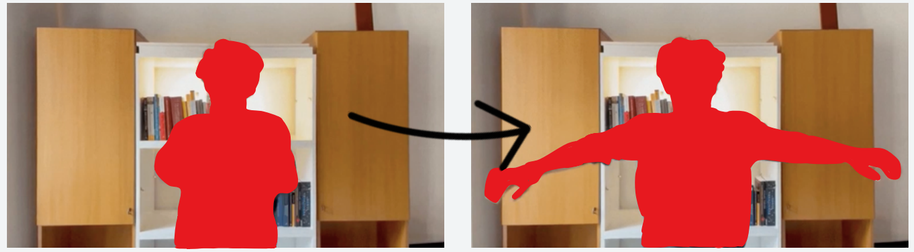
Right rotation
An asymmetric movement that has a horizontal component that is mostly **negative**

Left rotation
An asymmetric movement that has a horizontal component that is mostly **positive**

ACOUSTIC FEEDBACK: COMPACTNESS

Compactness

- The more the variation in compactness is, the more the water moved will be noisy.



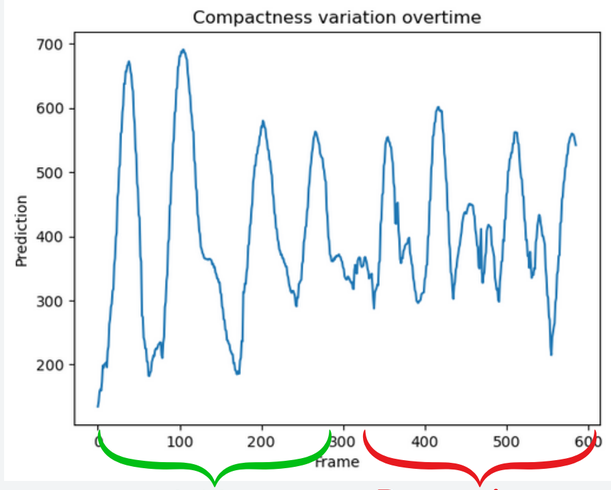
```
def compute_volume_variation(lhand, rhand, spine):
    volume = 0

    for hand in [lhand, rhand]:
        vector_x = spine["x"] - hand["x"]
        vector_y = spine["y"] - hand["y"]

        volume = volume + math.sqrt(vector_x**2 + vector_y**2)

    return volume
```

The magnitude of vector from hands approximate the compactness

$$as(t_i) = \|\boldsymbol{p}^{H}(t_i) - \boldsymbol{p}^{SP}(t_i)\|$$

Input Window: Compactness computed considering one frame at the time

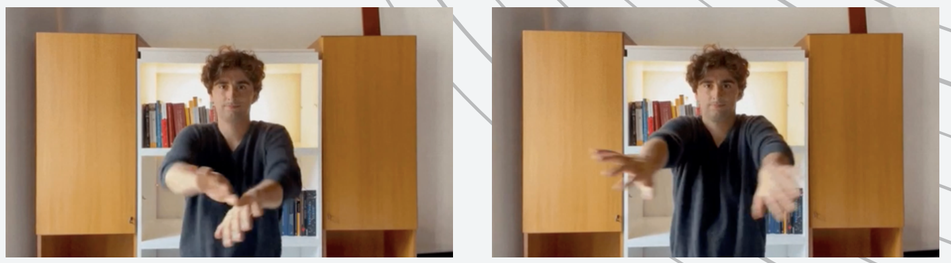
Output Window: Final prediction computed by checking if the current compactness index is, on average, higher or lower than the last_frames_considered ones (at the moment initialized to 15)



ACOUSTIC FEEDBACK: WRIST SPEED

Join speed

- The faster the movement is, the louder will be the acoustic feedback.



```
def compute_speed(lhand_past, lhand_now, rhand_past, rhand_now,dt):
    speed_lhand = math.sqrt(((lhand_past["x"] - lhand_now["x"])/dt)**2 +
    ((lhand_past["y"] - lhand_now["y"])/dt)**2)

    speed_rhand = math.sqrt(((rhand_past["x"] - rhand_now["x"])/dt)**2
    + ((rhand_past["y"] - rhand_now["y"])/dt)**2)

    return round((speed_lhand + speed_rhand),2)
```

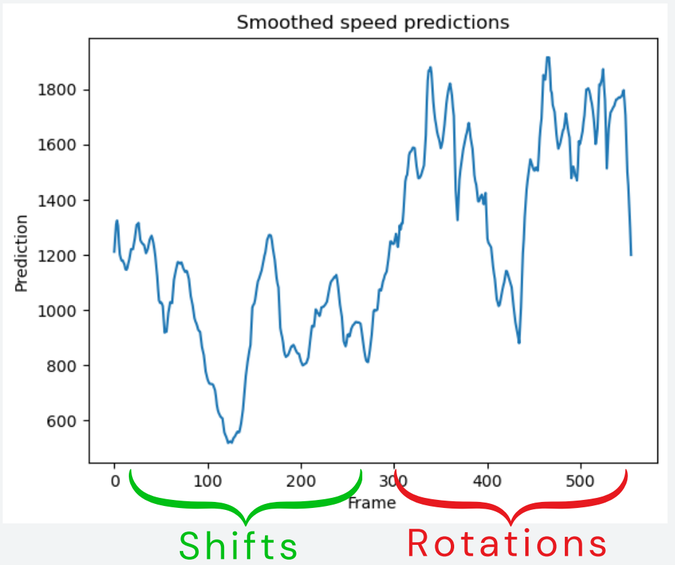
Where dt is 2/fps, because the speed is computed as follows:

$$\boldsymbol{v}^k(t_{i-1}) = \frac{\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-2})}{t_i - t_{i-2}}$$

$$v^k(t_i) = \sqrt{\left(v_x^k(t_i)\right)^2 + \left(v_y^k(t_i)\right)^2 + \left(v_z^k(t_i)\right)^2}$$

Input Window: Speed computed considering comparing the actual position with the one of last_frames_considered frames ago (last_frames_considered is initialized to 2)

Output Window: Final prediction computed by smoothing the raw one with a convolution composed by the average over the window of window_size elements (at the moment initialized to 30)



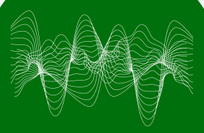
BREATH FEATURES

Emotions and respiratory patterns review and critical analysis

Frans A. Boiten,Nice H. Frijda,
Cornelis J.E. Wientjes

I/E ratio
> thoracically
< abdominally

THORACIC / ABDOMINAL



Irregular Inter-cyclic breathing
Volumes, rithms

IRREGULARITIES AND INTERRUPTIONS

Smoothness and angle of transition between I - E

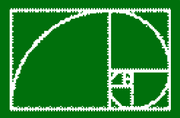
Clausen (1951)

- Inspiration phase
- inspiration pause
- expiration phase
- expiration pause

BREATH CYCLE SHAPE

mean frequency of cycles

RESPIRATORY RATE (RR)

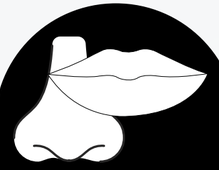


Inspiration over total duration of breath

I / E RATIO

Fast combo of Inspiration - pause - Inspiration

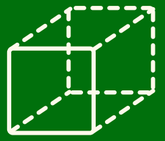
HIGH FREQUENCY OSCILLATIONS



Determinant of change in volume and time

Perez and Tobin, 1985

NASAL / ORAL



volume of air exchange In one cycle

TIDAL VOLUME (Vt) (Depth)

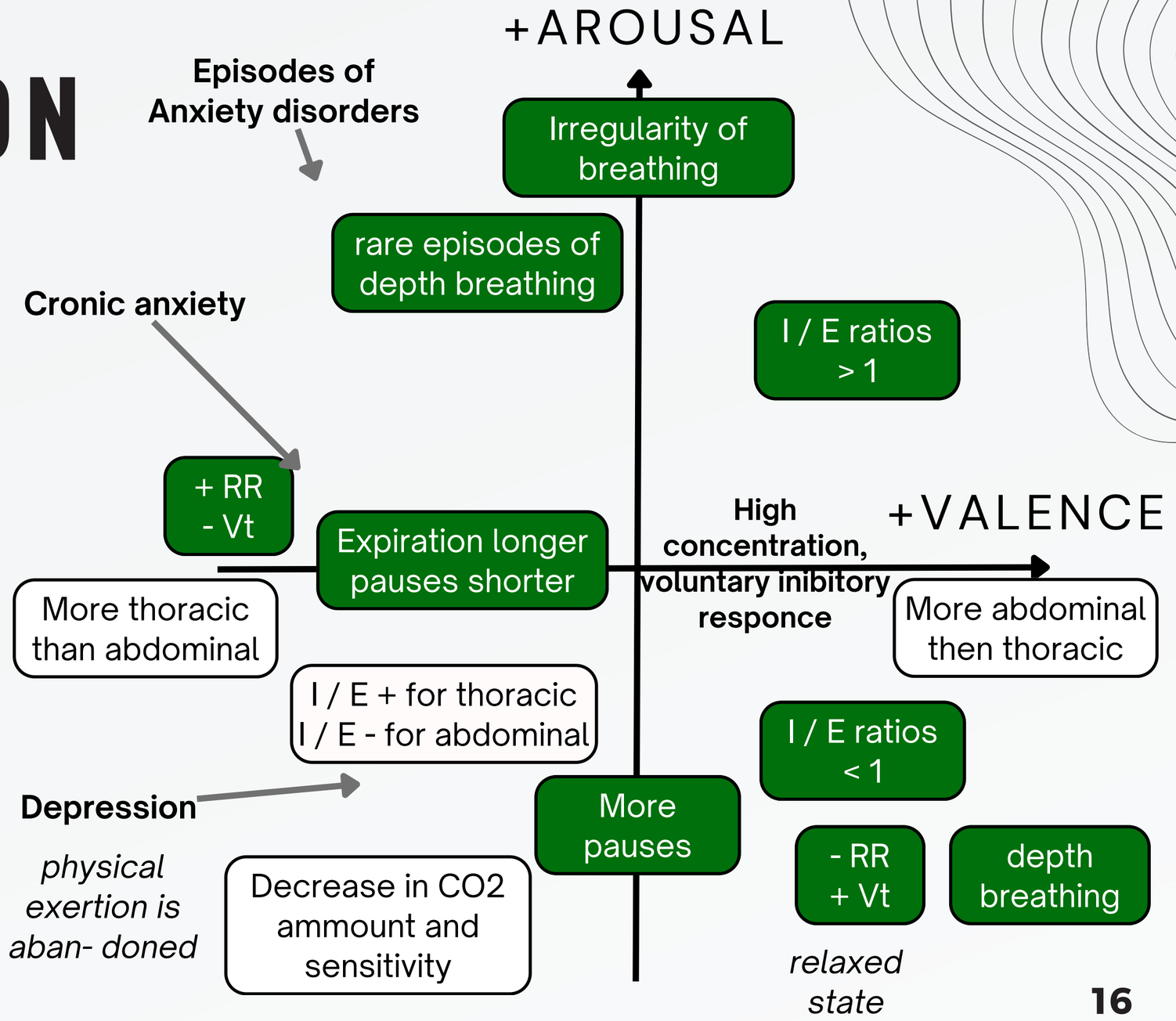
CLASSIFICATION

Slow breathing can reduce subjective indices of anxiety (Grossman, 1983)

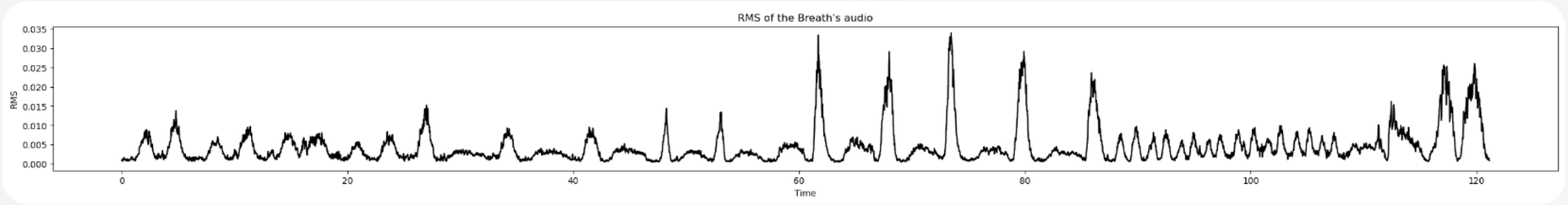
Excessive ventilation is a vicious cycle for panic, and bring to somatization (Wientjes, 1993)

Feleky 1914

Emotion	I/E		Work		Depth		Rate	
disgust	1.1	0	5.9	0	15.3	0	23.4	+
pleasure	1.1	0	9.5	+	27.3	+	20.4	+
anger	1.5	+	13.3	+ +	31.8	+ +	25.2	+ +
pain	1.5	+	8.0	+	29.7	+	16.2	0
wonder	2.5	+ +	9.5	+	38.6	+ +	9.6	0
fear	2.6	+ +	14.6	+ +	33.2	+ +	26.4	+ +
normal	0.8	0	4.7	0	13.9	0	20.4	+



RMS

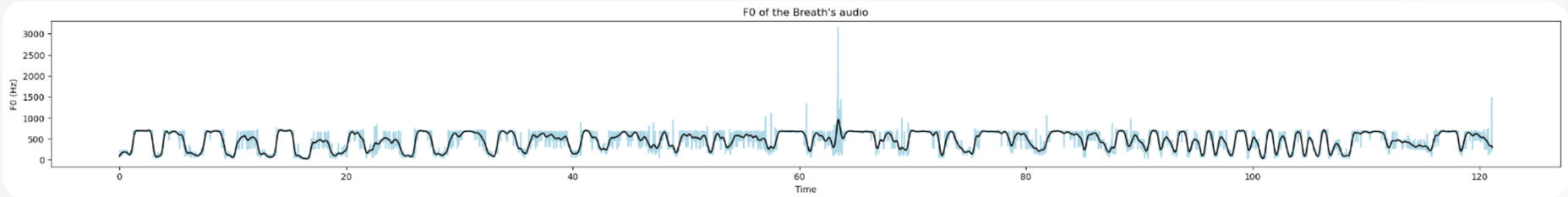


ExtractRMS()

ExtractRMS(Y)

librosa.feature.rms(Y)

F0 (FILTERED)



ExtractF0()

ExtractF0(Y)

librosa.autocorrelate(..)

argrelextrema(..)

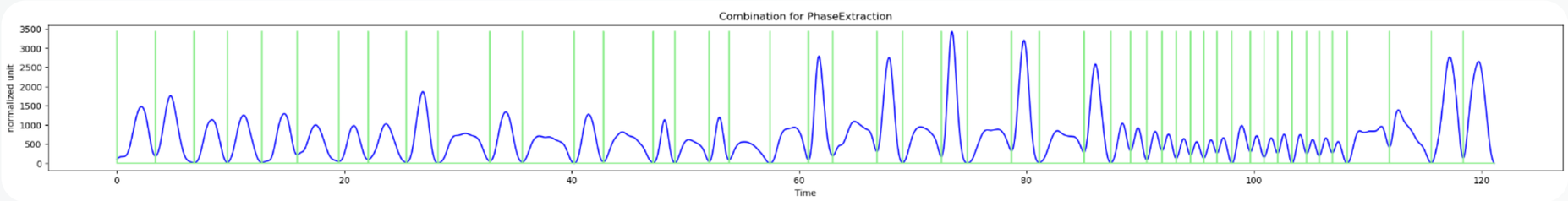
cleanAutocorrelation(Y)

FilterF0(..)

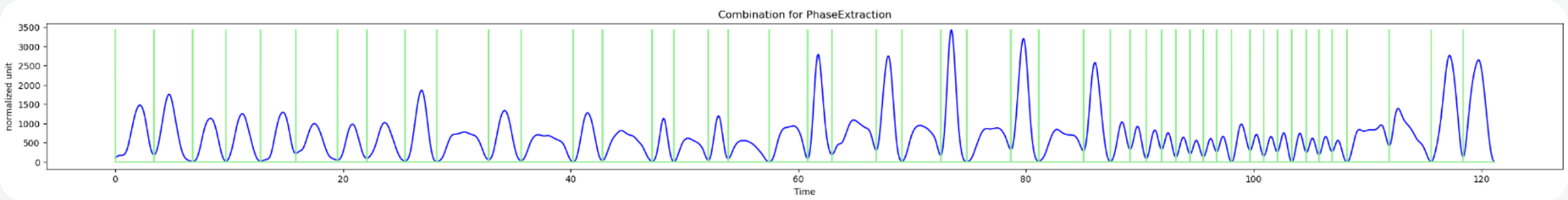
butter_lowpass_filter(..)

scipy.signal

LINEAR COMBINATION OF RMS & F0



UNITIZATION



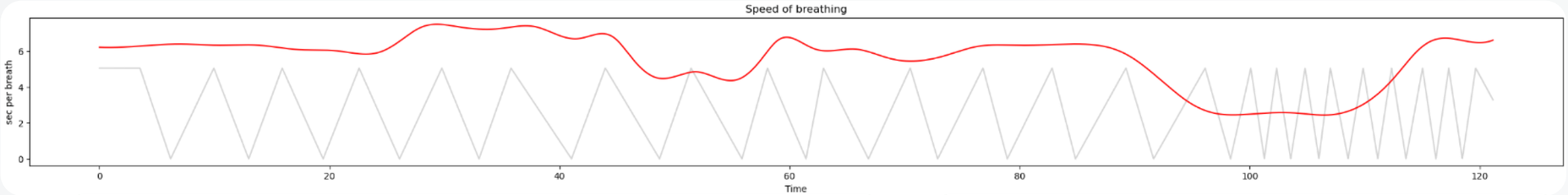
GetMinimums()

GetMinimums(Y = linerC_RMS_FØ)

argrelextrema(Y)

FilterMinimum(.. , tollerance)

BREATHING FREQUENCY



ExtractFrequency(AirFlow)

or

ExtractFrequency(linerC_RMS_FØ)

or

FrequencyFromMinimums (GetMinimums()

ExtractFrequency(Y)

ExtractFrequency_Rec(

ExtractFrequency_Rec(Y , null)

SubtractBaseLine(..)

)

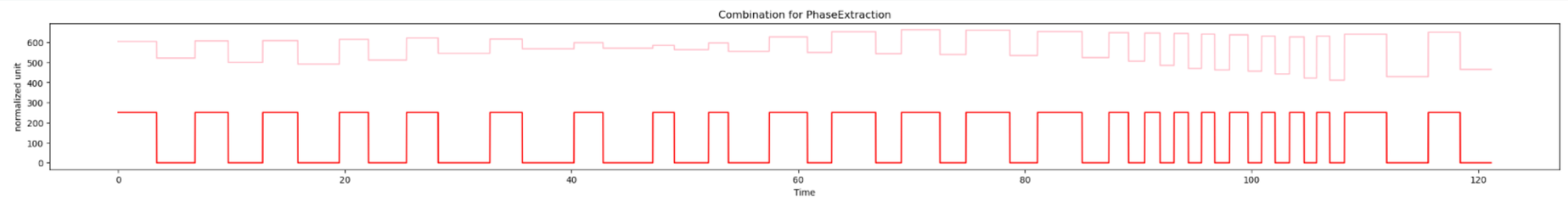
ExtractFrequency_Rec(Y, appriximation)

for w in getWindows(appriximation)

librosa.autocorrelate(w)

argrelextrema(..)

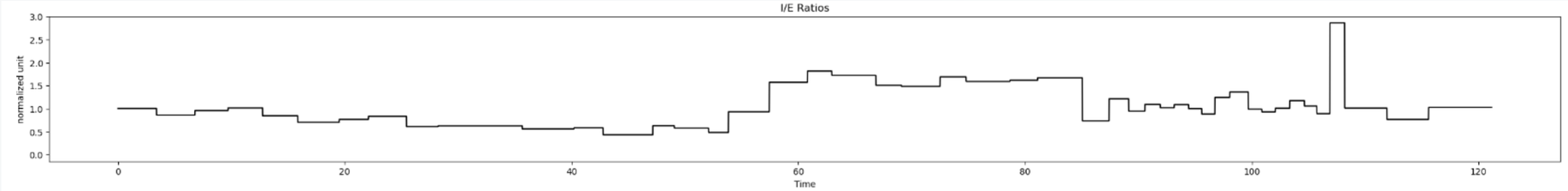
CLASS {I, E}



`ClassifyIE()`

```
ClassifyIE()
    NormalizeFeature(
        GetMinimums()
        FilteredFØ()
    )
    Sign(curr - before)
```

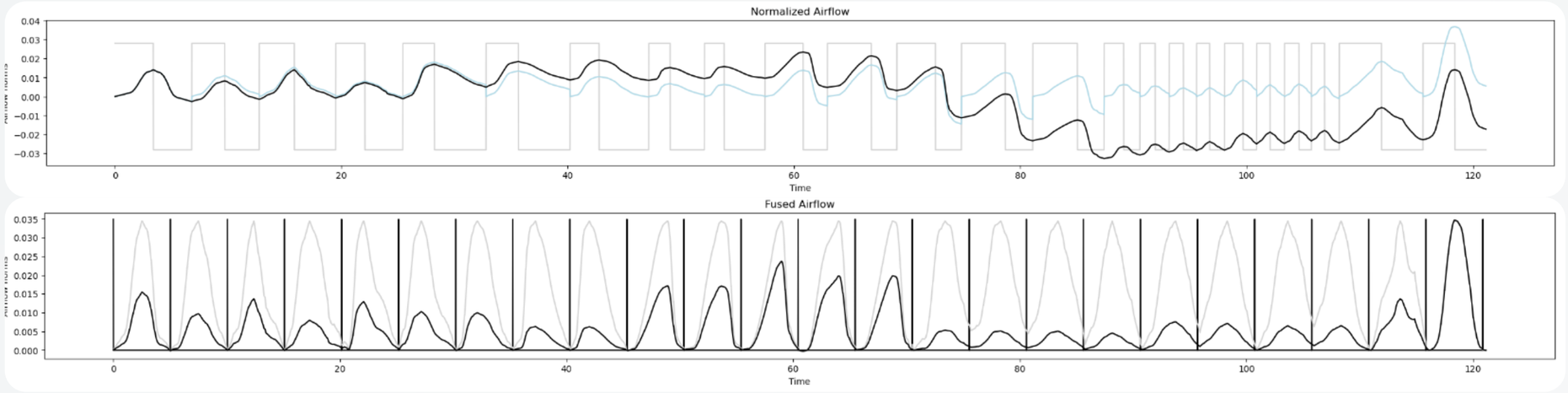
(I, E) RATIO



`CalculateIERatios()`

```
CalculateIERatios()
    MeasureDuration( ClassifyIE() )
```

AIRFLOW
+NORMALIZED

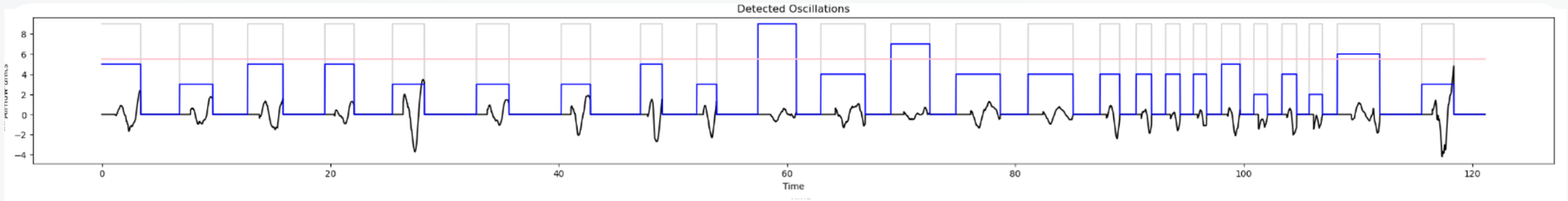


ExtractAirFLow()

```
ExtractAirFLow(data, phaseClasses)
    For d in data
        addDeltaToCurve( GetDelta(..) )
```

```
GetDelta(data, FØ, RMS, phaseClass)
    polinomial regression => delta
    if phasesClass == INSPIRATION
    return delta else return -delta
```

HIGH FREQ. OSCILLATIONS



DetectOscillations()

```
DetectOscillations(airflow, phasesClass)
    FilterInspirations()
    HighPassFilter(filteredAirflow)
    librosa.zero_crossings()
```

HARDWARE COMPONENTS

Speaker

Battery life: Up to 5 hours

Response: 180Hz – 20kHz Bluetooth: 4.1

Computer

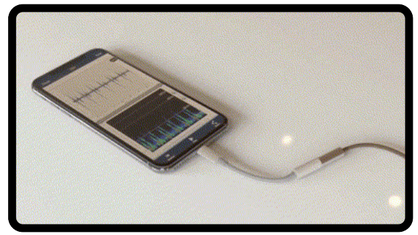
With Windows 10+
or MacOS > Catalina
Webcam: 1 MP, 30 fps

Projector

WiFi or USB cable

1920x1080 pixels

ThinkLabs One
Digital Stethoscope



Audacity Compatibility

3.5mm headphone jack

WMQ Single Head
Digital Stethoscope
51,69€ Alibaba

TOOLS FEATURES EXTRACTION

Openface2, to extract **gaze** features

Mediapipe, to infer **pose** landmarks

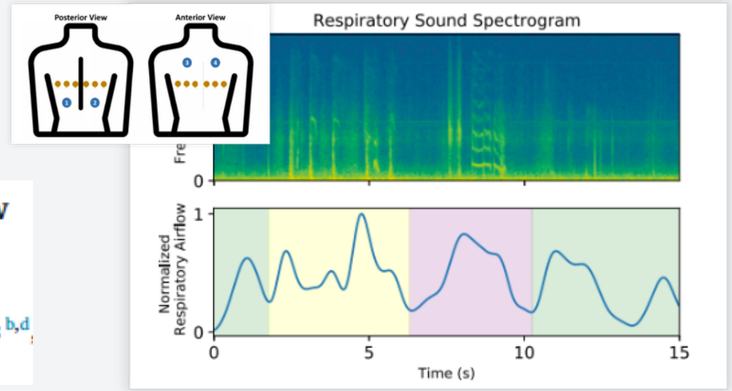
Audacity, to record and minimal breathing dataSet

Librosa, to extract **audio** features

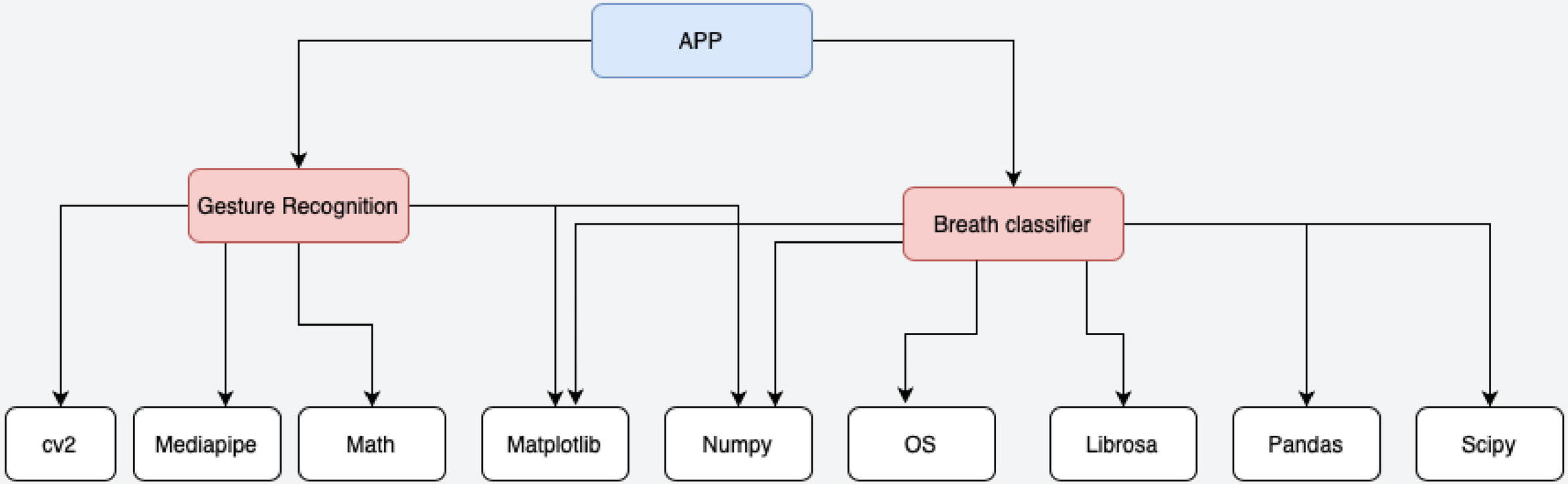
OpenSource deep learning model,
For simplified AirFlow Extraction

Ensemble deep learning model for dimensionless respiratory airflow estimation using respiratory sound

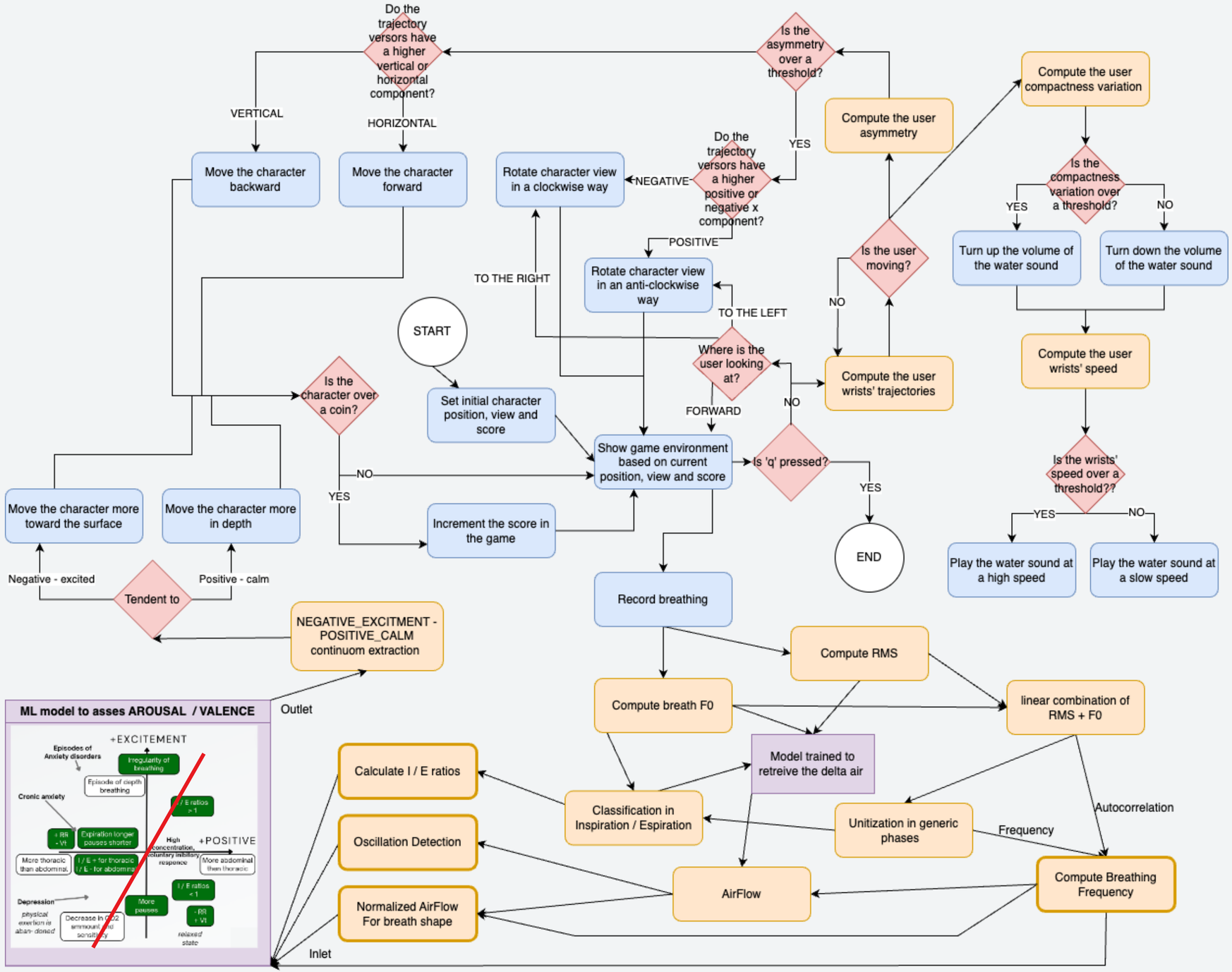
Diogo Pessoa [a,*], Bruno Machado Rocha [a], Maria Gomes [b], Guilherme Rodrigues [b], Georgios Petmezas [c], Grigorios-Aris Cheimariotis [c], Nicos Maglaveras [c], Alda Marques [b,d], Inéz Frerichs [e], Paulo de Carvalho [a], Rui Pedro Paiva [a]



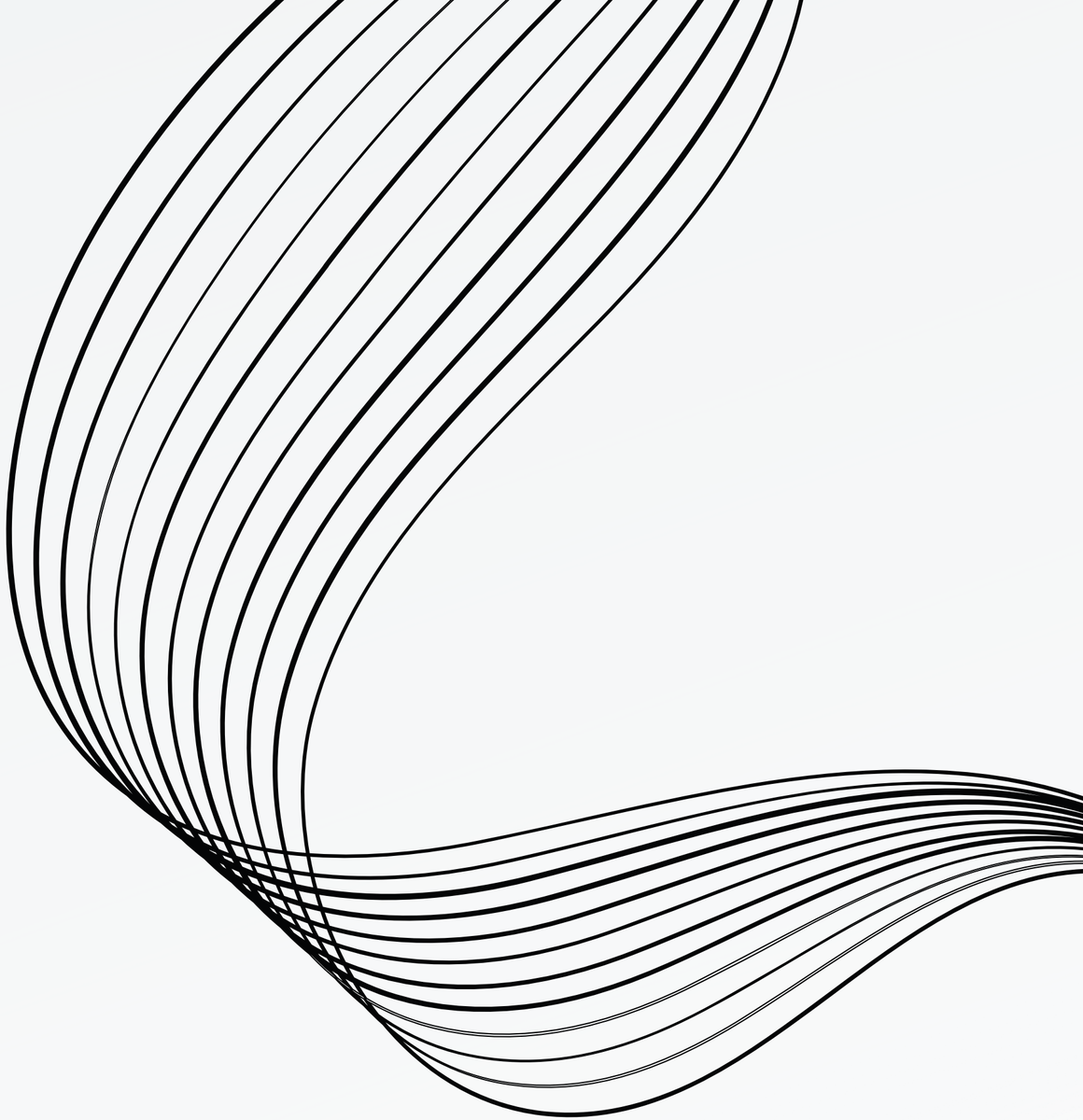
UML COMPONENT DIAGRAM



UML STATE CHART



READY FOR A DEMO?



CONCLUSIONS

Breath recording wise it is **difficult** to, **recreate** the same **conditions** in the **training** fase and in the **use**. The stetoskope solution would reduce noise and allow music to be played on top.

We are **confident** in the potential for obtaining a **good approximation** of **airflow**. Each frame requires **calculating** a **delta** (+-) which can be obtained **through** a **regression task** (or **ANN** with backpropagation) on the already computed features: classification **I/E, RMS, FO**.

The gesture features extracted seem to be adequate and meaningful for both the tasks of **gesture classification** and to produce **acoustic feedback**. The fact that the features can be used in an online way, makes the approach suitable in a **real context**.

Having the possibility to work in a **group**, allowed us to face the project in a more challenging way, with two different but **complementary** styles.

LIMITATIONS

The **audio** feature **extraction** module incurs an average **delay** of **1.5 breaths** (using a window size of 3 breaths and a hop factor of 1.5).

The gesture **classifier** proposed needs to see a **full wrist rotation** before classifying the movement. In addition, it cannot say when no movement is **present** or when it’s not recognized with high confidence.

FURTHER IMPROVEMENTS

Make the system **real-time with SSI**. **Normalization of breath** could though give use a **prediction** of the next breath, for the research of minimums.

There is the necessity to build a **machine learning classifier** that would be able to work with the classification-related gestures extracted in order to obtain better movements’ predictions, especially for what regards the trajectories analysis.

REFERENCES

- https://www.cdc.gov/childrensmentalhealth/data.html
- https://www.cdc.gov/childrensmentalhealth/features/anxiety-depression-children.html
- https://www.healthychildren.org/English/family-life/family-dynamics/Pages/help-your-child-manage-fears-and-anxieties.aspx
- https://www.nature.com/articles/s41598-018-32046-5
- Emotions and respiratory patterns review and critical analysis (Frans A. Boiten,Nice H. Frijda, Cornelis J.E. Wientjes)
- Respiratory therapy for the treatment of anxiety Meta-analytic review and regression (Teresa M. Leyroa, Mark V. Versellaa, Min-Jeong Yanga, Hannah R. Brinkmana, Danielle L. Hoyt a, Paul Lehrer)
- Upper-Body Geometric Features for Traffic Command Gesture Recognition (Sijia Wang , Kun Jiang , Junjie Chen , Mengmeng Yang and Zheng)

THANK'S FOR WATCHING

ARE THERE ANY QUESTIONS?