Interaction with live2d using facial landmark detection, facial expression recognition and image stylization

Team09

蕭延儒 r09922139

方郁婷 r10922196

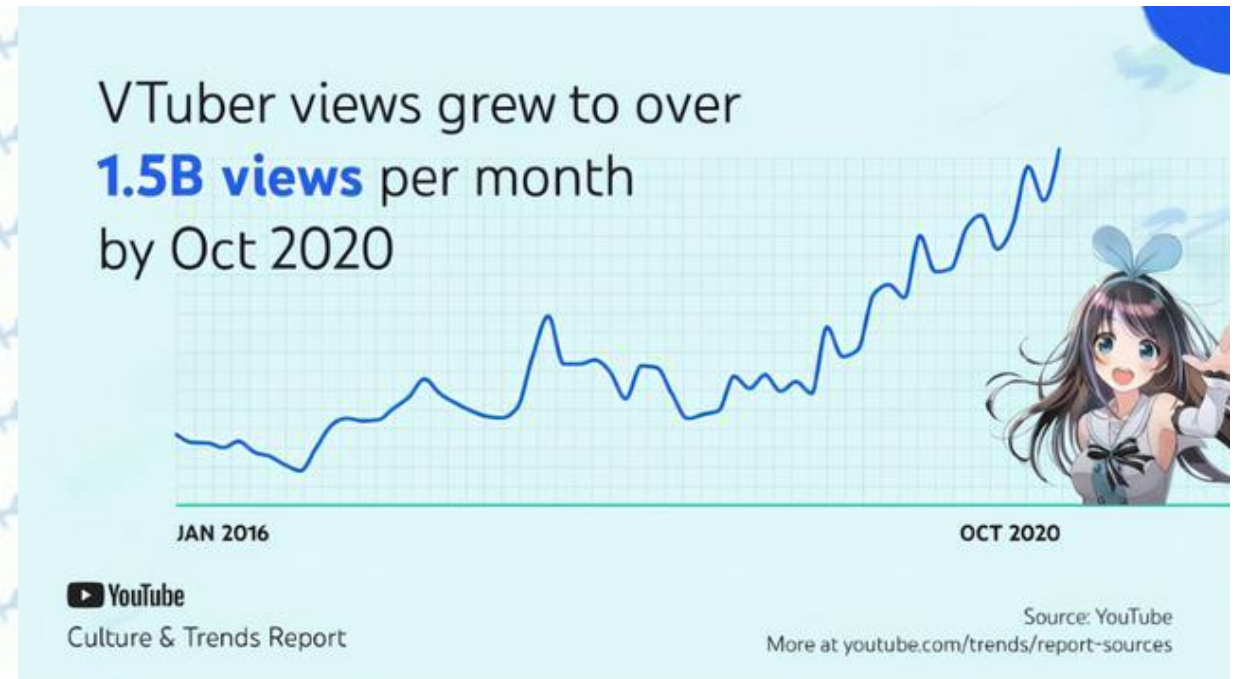林祐霆 r09944041

9

Motivation

Vtuber and machine learning is the trend of the times.
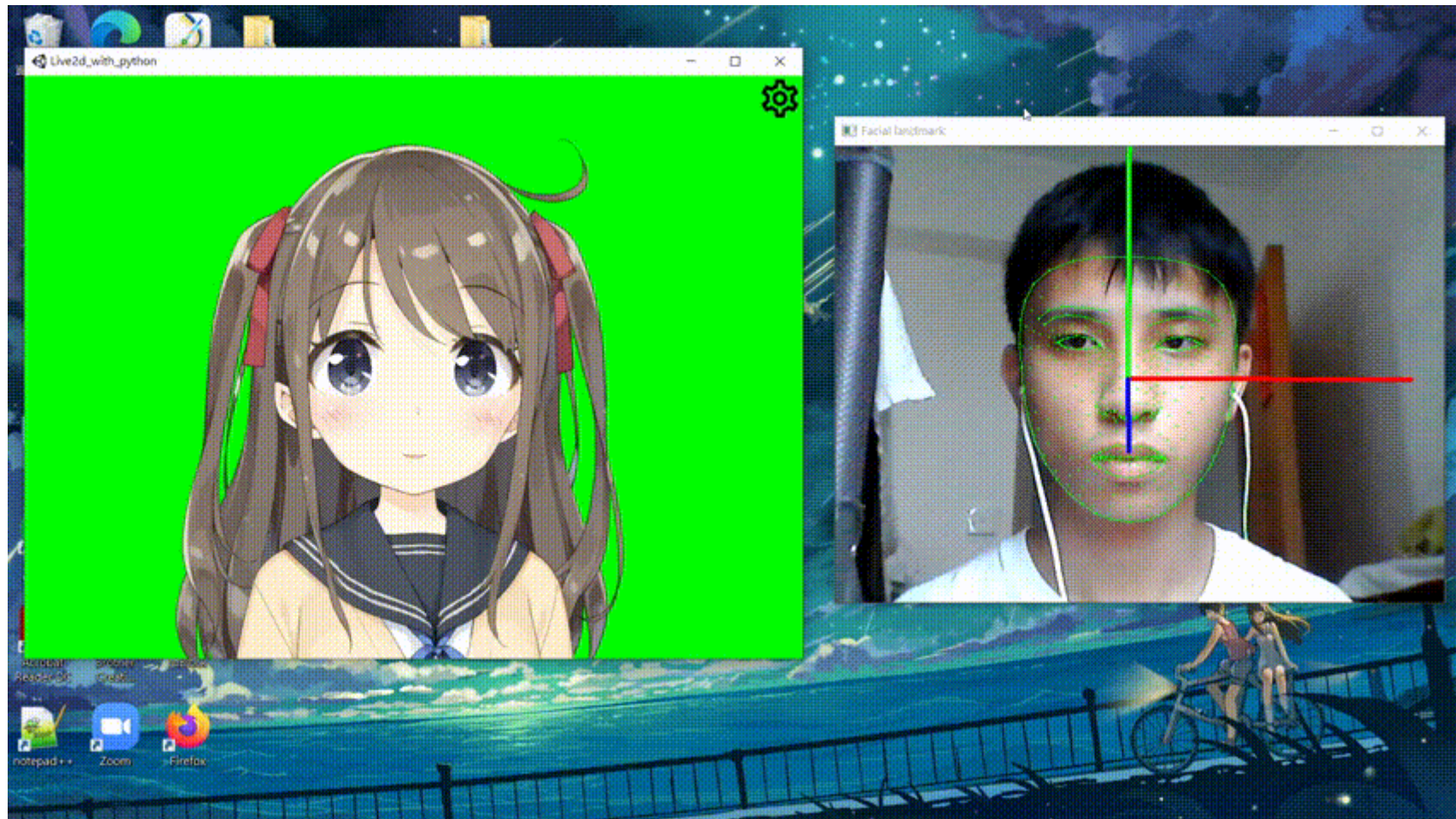

Vtuber: Gawr Gura

Motivation

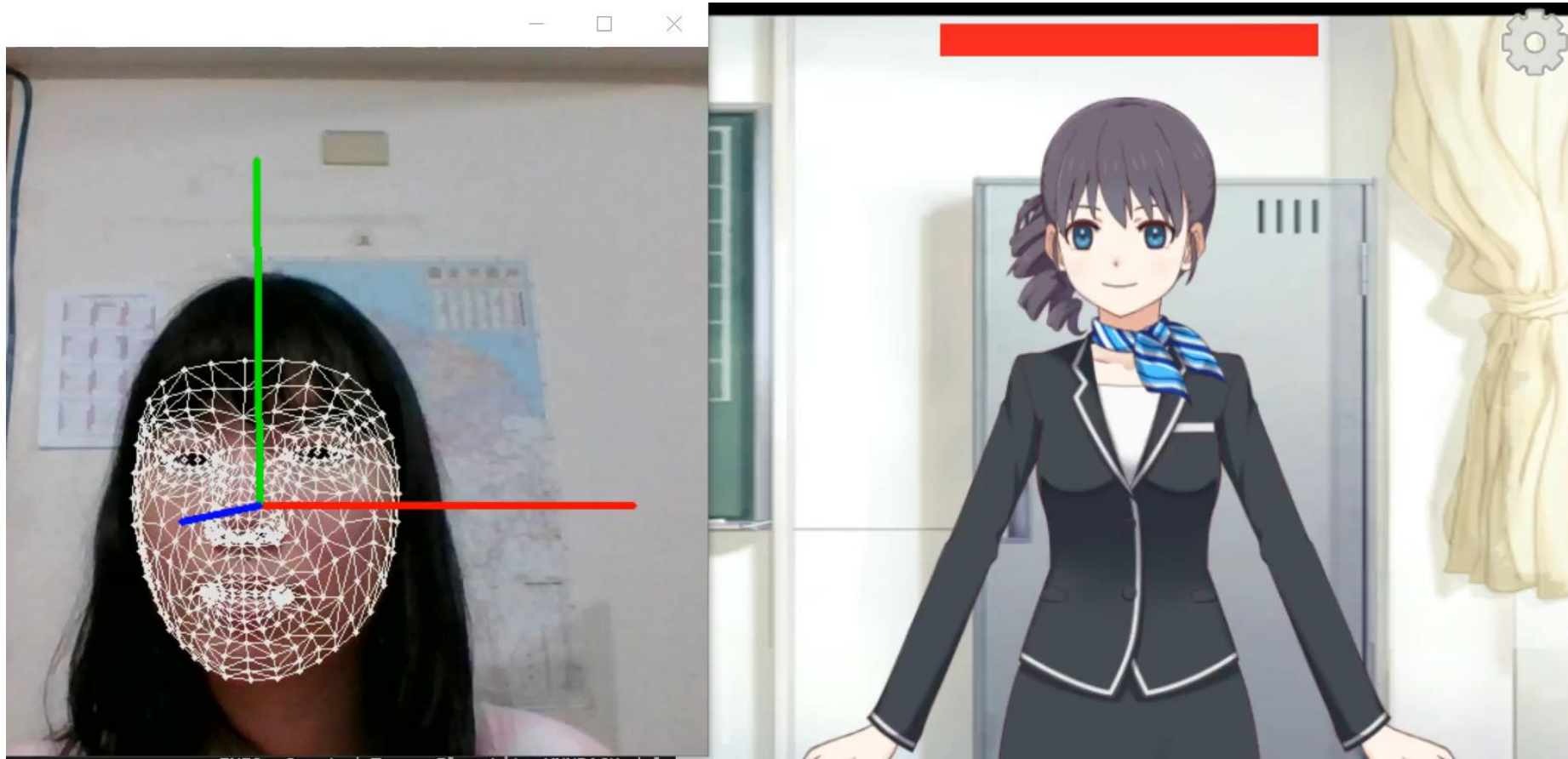Vtuber and machine learning is the trend of the times.

Motivation

We combine them with traditional algorithms, such as image stylization, to make the result more diverse.
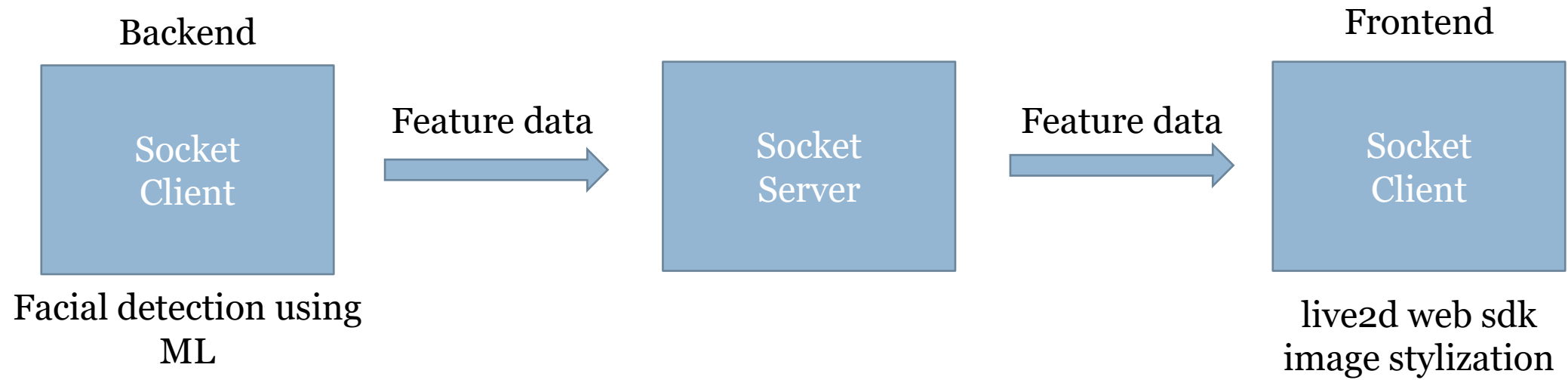
Problem definition

We want to build a system that users can control the 3D model through a camera, and the output motion picture will depend on users' facial movement. We will focus on the facial expression and the overall image style.
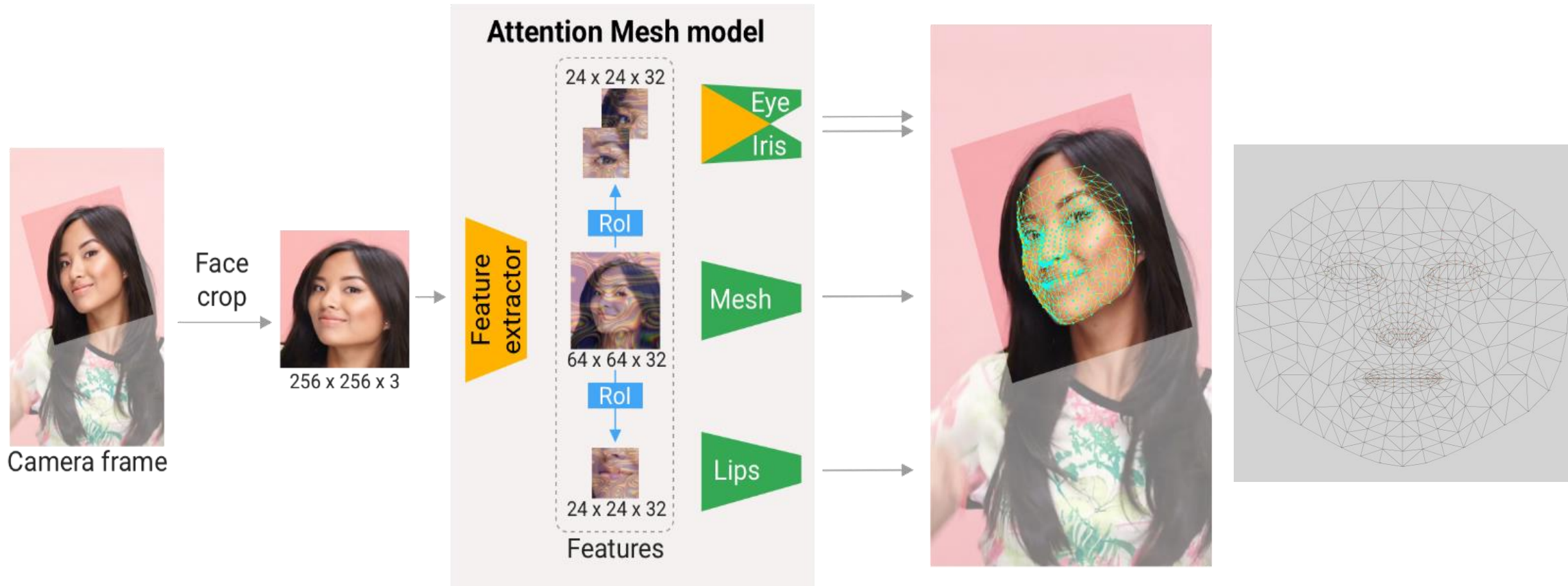
Algorithm

Below is our Architecture.

Backend   Frontend

| Socket Client | → Feature data → | Socket Server | → Feature data → | Socket Client |

Facial detection using ML   live2d web sdk image stylization
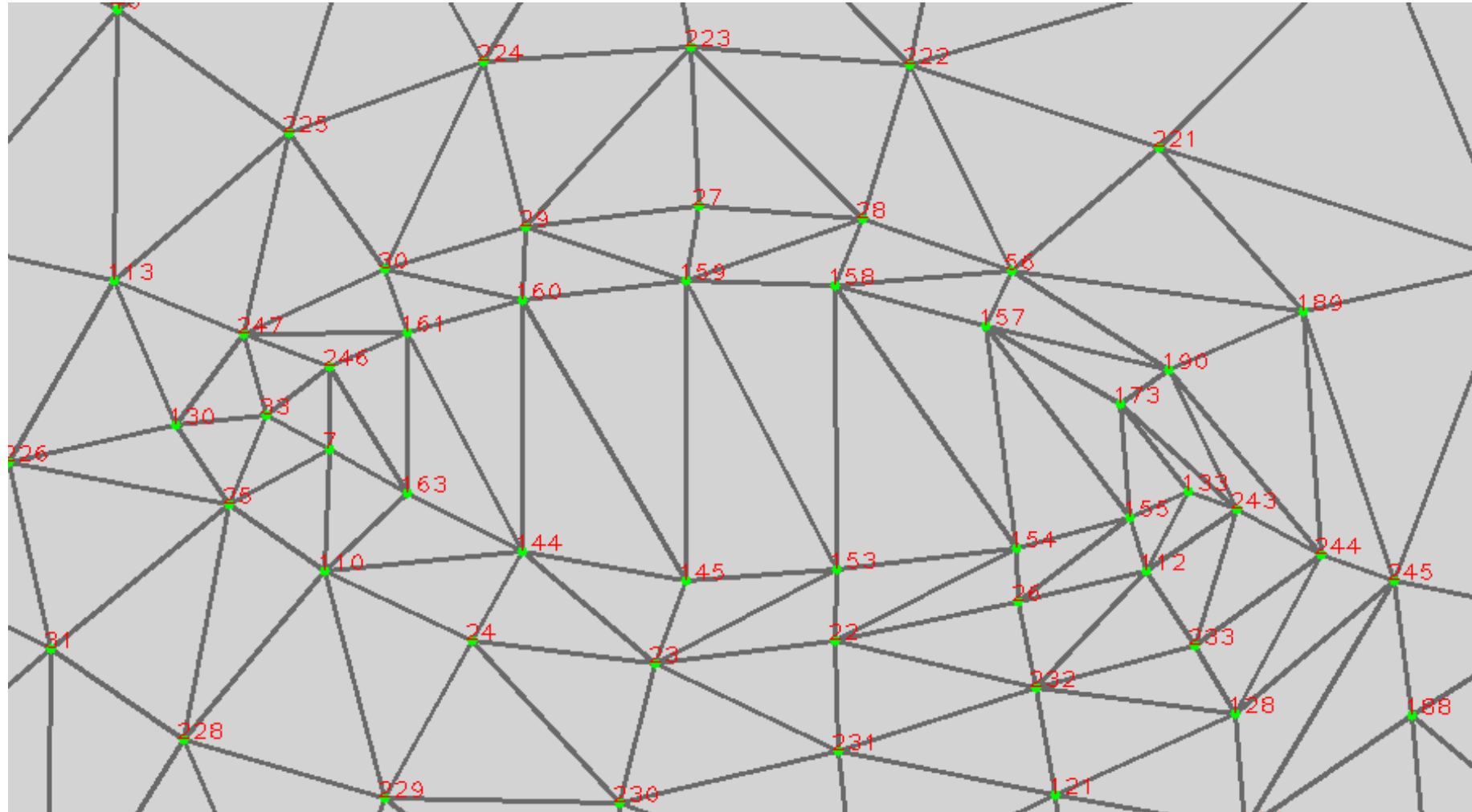
Facial Detection

Using Mediapipe to find landmark

The library is made by Google. It can provide excellent, gpu-standard face detection and landmarks detection using cpu only, providing 30 FPS smooth detection.

Facial Detection

e.g., Right eye and the mark number
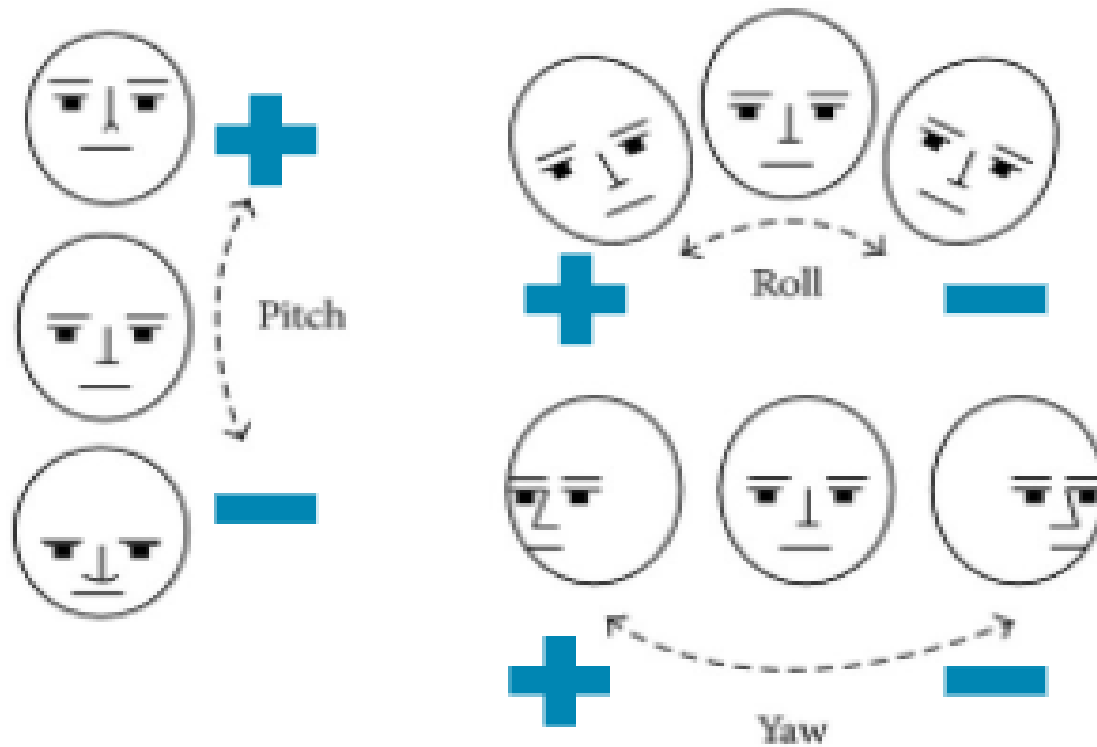
Facial Detection

Feature data

- Rotation Vector-> Face Direction
- Eye aspect ratio -> Blinking
- Iris detection -> Iris movement
- Mouth aspect ratio -> Speaking (open、close)
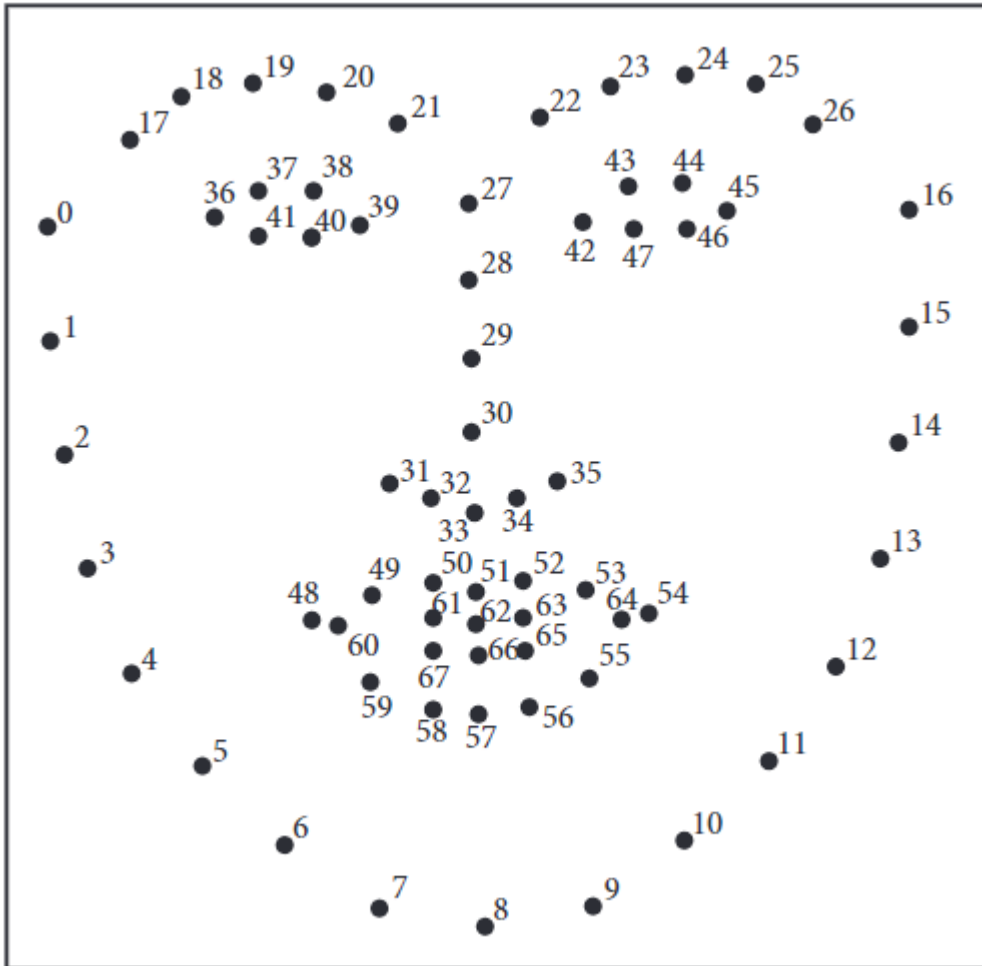- Mouth Distance -> Mouth form (shape)

Facial Detection

Rotation Vector

- using solvePnP function of Opencv to estimate rotation vector.
- convert the rotation vector to roll-pitch-yaw
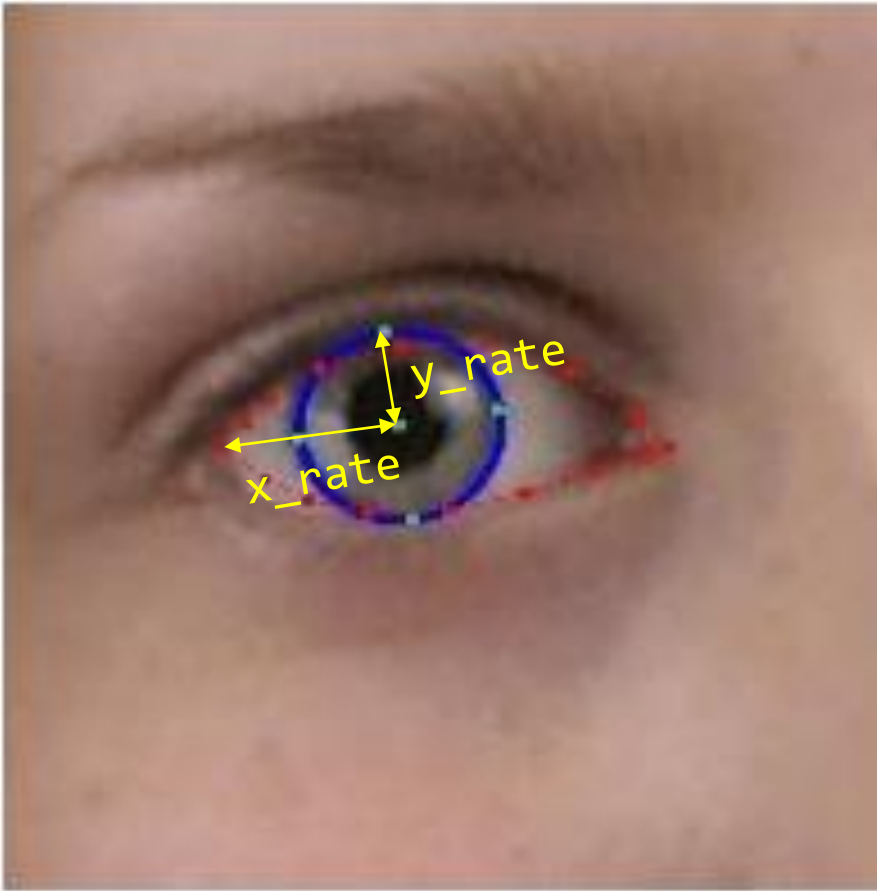
Facial Detection

Eye aspect ratio

$$EAR_{Lt} = \frac{\|LM_{37} - LM_{41}\| + \|LM_{38} - LM_{40}\|}{2 * \|LM_{36} - LM_{39}\|} * \frac{\|LM_{19} - LM_{33}\|}{\|LM_{27} - LM_{33}\|},$$

$$EAR_{Rt} = \frac{\|LM_{43} - LM_{47}\| + \|LM_{44} - LM_{46}\|}{2 * \|LM_{42} - LM_{45}\|} * \frac{\|LM_{24} - LM_{33}\|}{\|LM_{27} - LM_{33}\|}.$$

Ref: Using Eye Aspect Ratio to Enhance Fast and Objective Assessment of Facial Paralysis
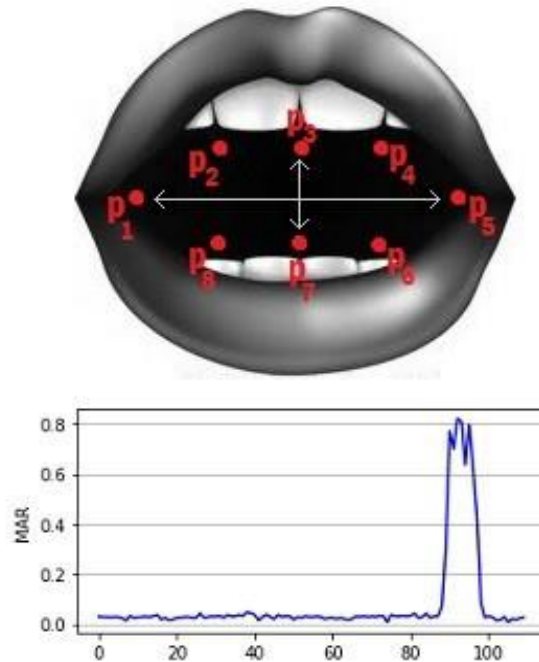
11

Facial Detection

Iris detection

x_rate: how much the iris is toward the left.
0 means totally left and 1 is totally right.

y_rate: how much the iris is toward the top.
0 means totally top and 1 is totally bottom.

Facial Detection

Mouth aspect ratio

$$\mathbf{MAR} = \frac{\|p_2 - p_8\| + \|p_3 - p_7\| + \|p_4 - p_6\|}{2\,\|p_1 - p_5\|}$$

Ref: Asleep at the Wheel: A Computer Vision and Deep Learning Approach to Detecting Drowsiness | by Jaynish P. Vaghela | Medium

Mouth Distance = ‖ *p1 - p5* ‖

live2d

live2d can simulate the effect of 3D. How dose it works?

model is divided into different segments

live2d

➢ **translation**

$$\begin{bmatrix} x_k \\ y_j \end{bmatrix} = \begin{bmatrix} u_q \\ v_p \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$
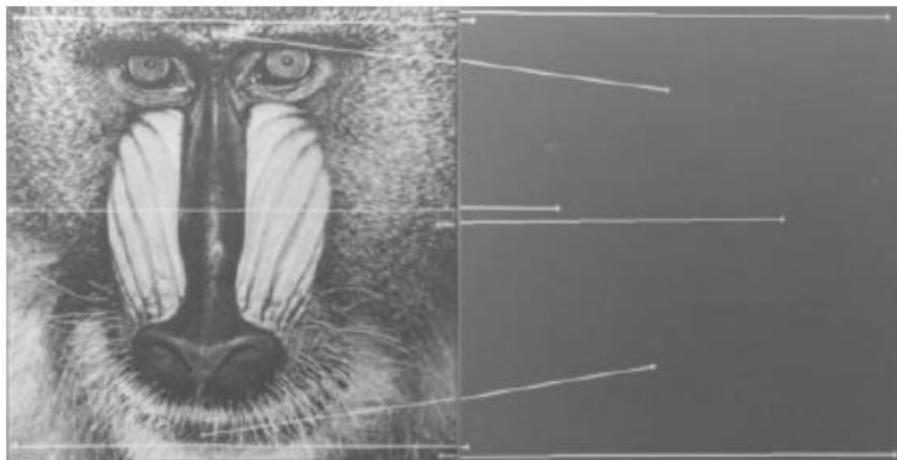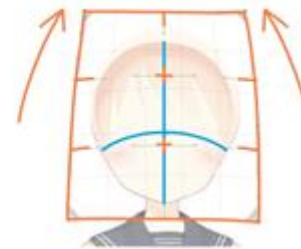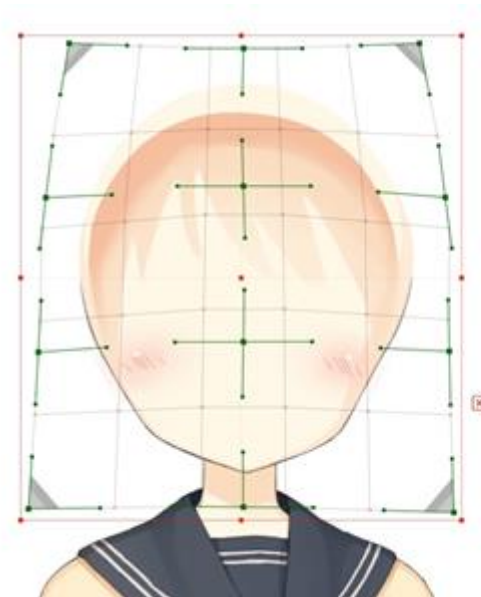
live2d

➤ **rotation**

$$\begin{bmatrix} x_k \\ y_j \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u_q \\ v_p \end{bmatrix}$$

live2d

morphing

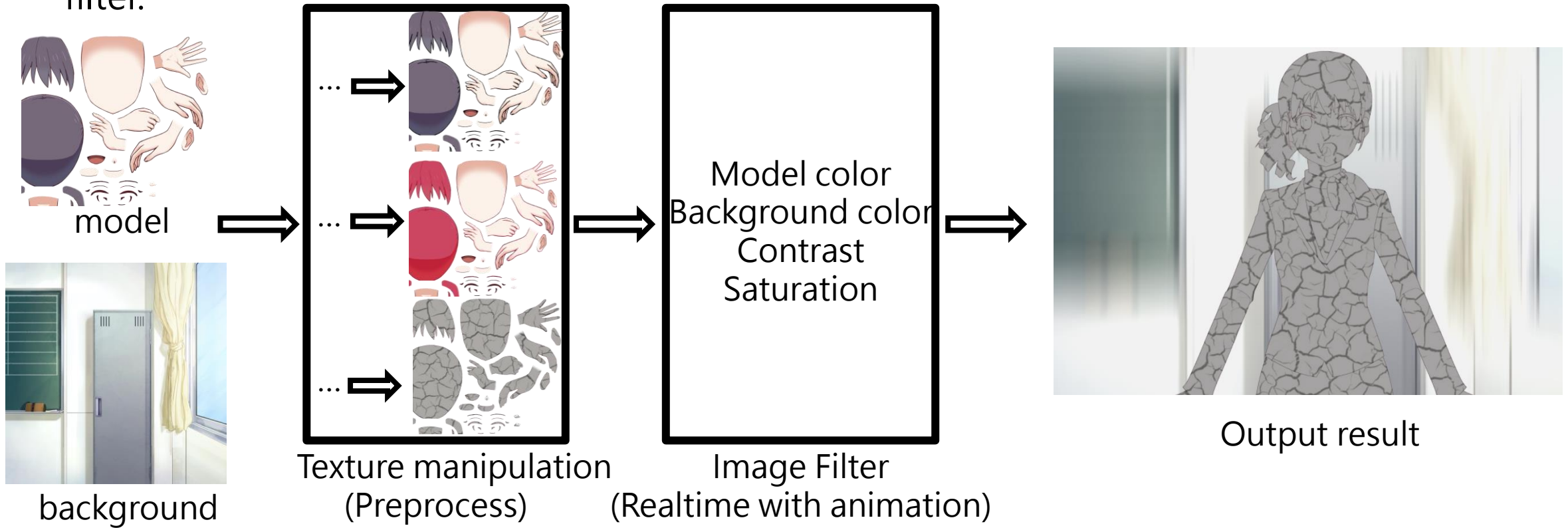input   control points   desired output

Style transfer

In this section, we introduce our method of transferring character style.

Except for original, we have 4 extra style corresponding to character expression: Happy, angry, surprise, and closing eyes.

Processing pipeline

We preprocess the texture into different style, and then real-time tuning using image filter.

model

background

Texture manipulation
(Preprocess)

Model color
Background color
Contrast
Saturation

Image Filter
(Realtime with animation)

Output result

Happy

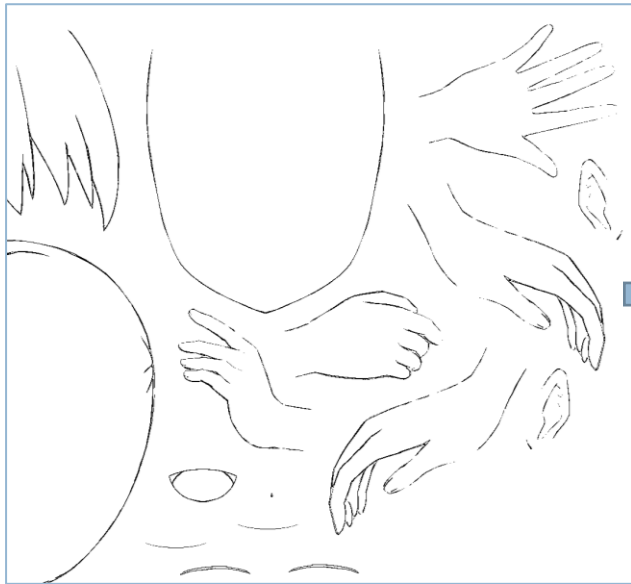Our idea is to make the character more energetic.

Step 1: Extract dark color edges from the model
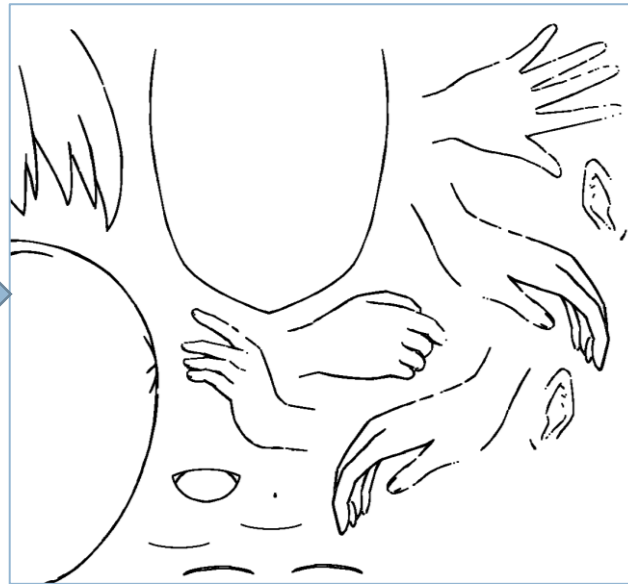Step 2: Thicken extracted edges and put it back to model
Step 3: (Real-time) Enhance background intensity and overall saturation

Happy

Pipeline:

Extract dark color edges

Dilate dark color edges

Output
(With intensity and saturation enhancement)

Angry

Our idea is to make the image dramatic and powerful.

Step 1: Extract gray color from model and turn them into blue.

Step 2: Decrease color temperature.

Step 3: Distort the background image using sine wave and Pincusion distortion.

Step 4: (Real-time) Decrease overall intensity and enhance contrast.
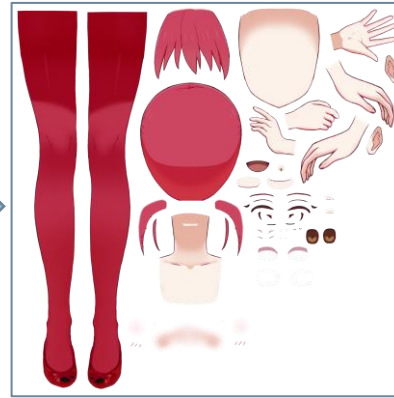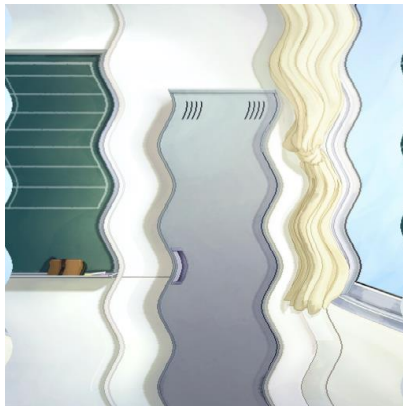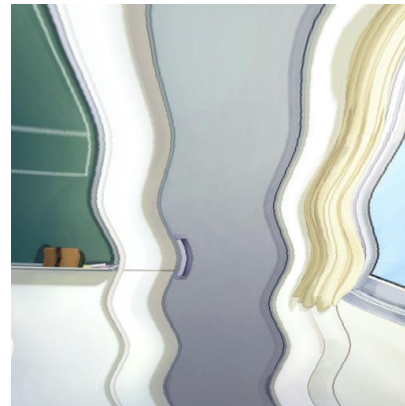
Angry

Pipeline:

Extract Gray

Enhance its blue channel

Decrease color temperature

Vertical Sine wave distortion

Pincusion distortion

Output
(With intensity decrease)

Surprise

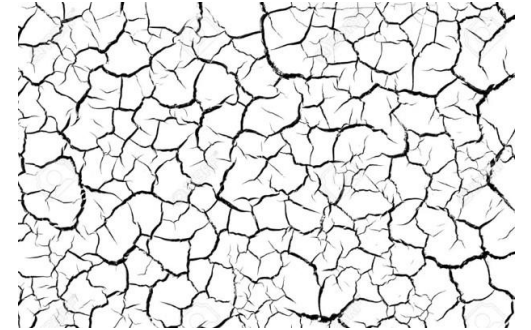Our idea is to make the cartoonish shock effect .

cracked.jpg

Step 1: Extract edges from the model

Step 2: Turn the whole model into color gray except for edges

Step 3: Add cracked texture on the model

Step 4: Add vertical motion blur to the background image
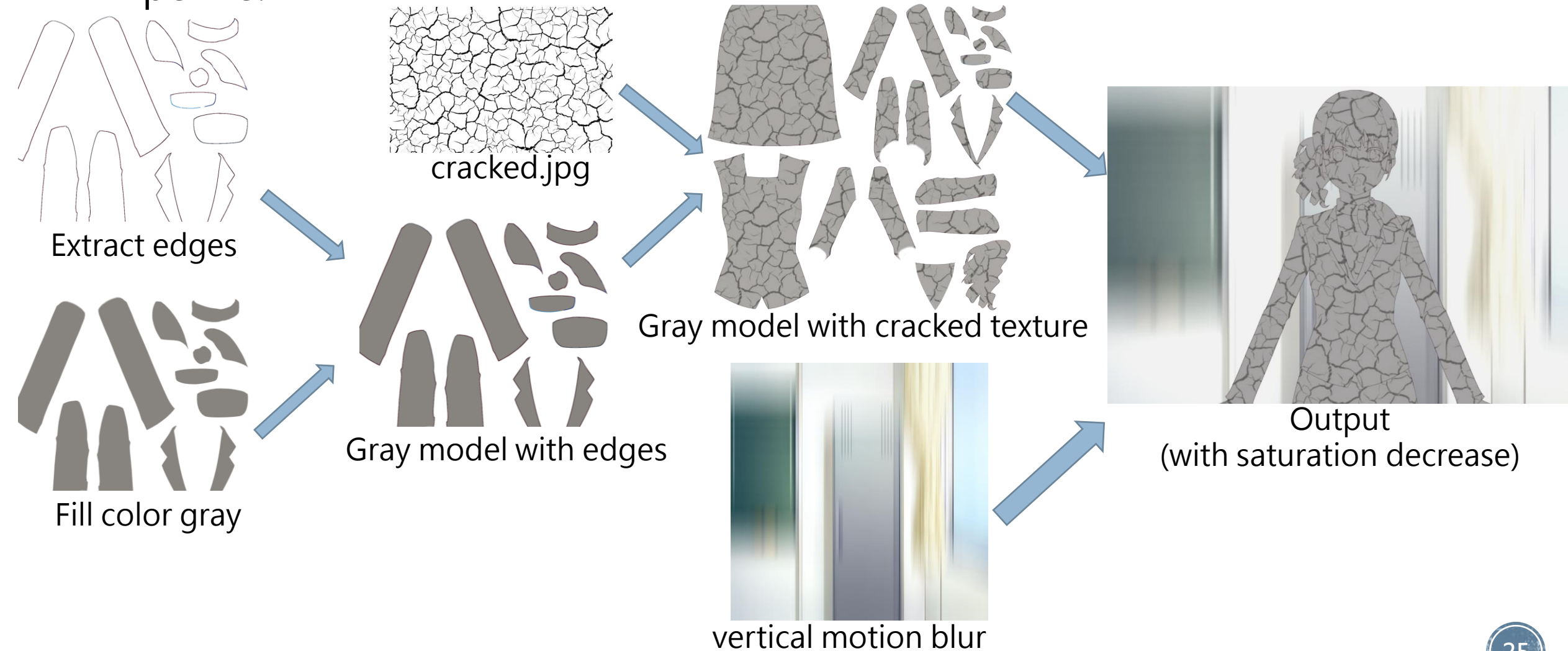
Step 5: (Real-time) Decrease the overall saturation

Surprise

Pipeline:

Extract edges

cracked.jpg

Fill color gray

Gray model with edges

Gray model with cracked texture

vertical motion blur

Output
(with saturation decrease)

25

Closing Eyes

Our idea is to make the image artistic.

Step 1: Extract edges from model

Step 2: Calculate the x, y-axis gradient of model

Step 3: Use gradient to calculate LIC(Line Integral Convolution)

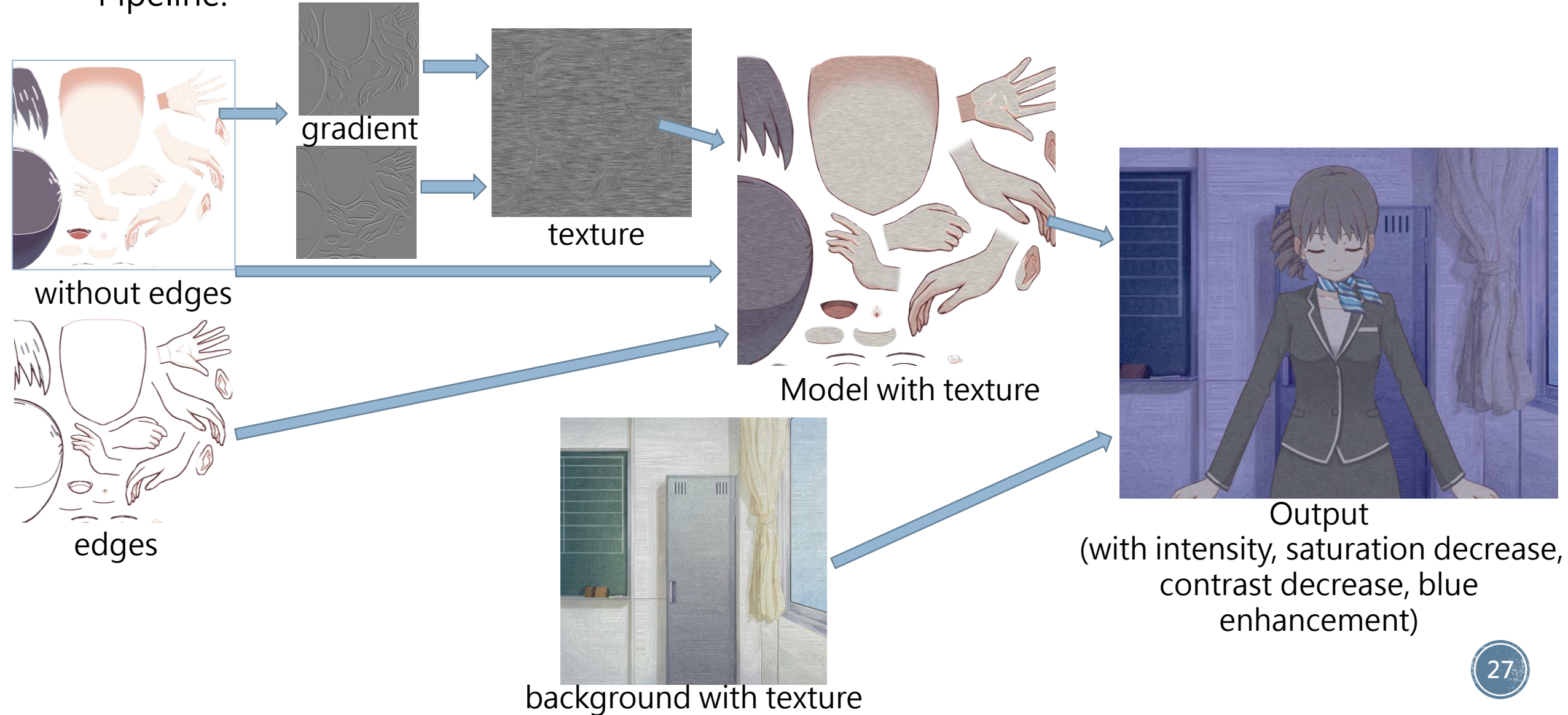Step 4: Use LIC as texture and add it to model

Step 5: Paste edge back to model

Step 6: Repeat Step 2~4 for the background image.

Step 7: (Real-time) Lower overall intensity, contrast and saturation; enhance color blue

Closing Eyes

Pipeline:

gradient

texture

without edges

edges

Model with texture

background with texture

Output
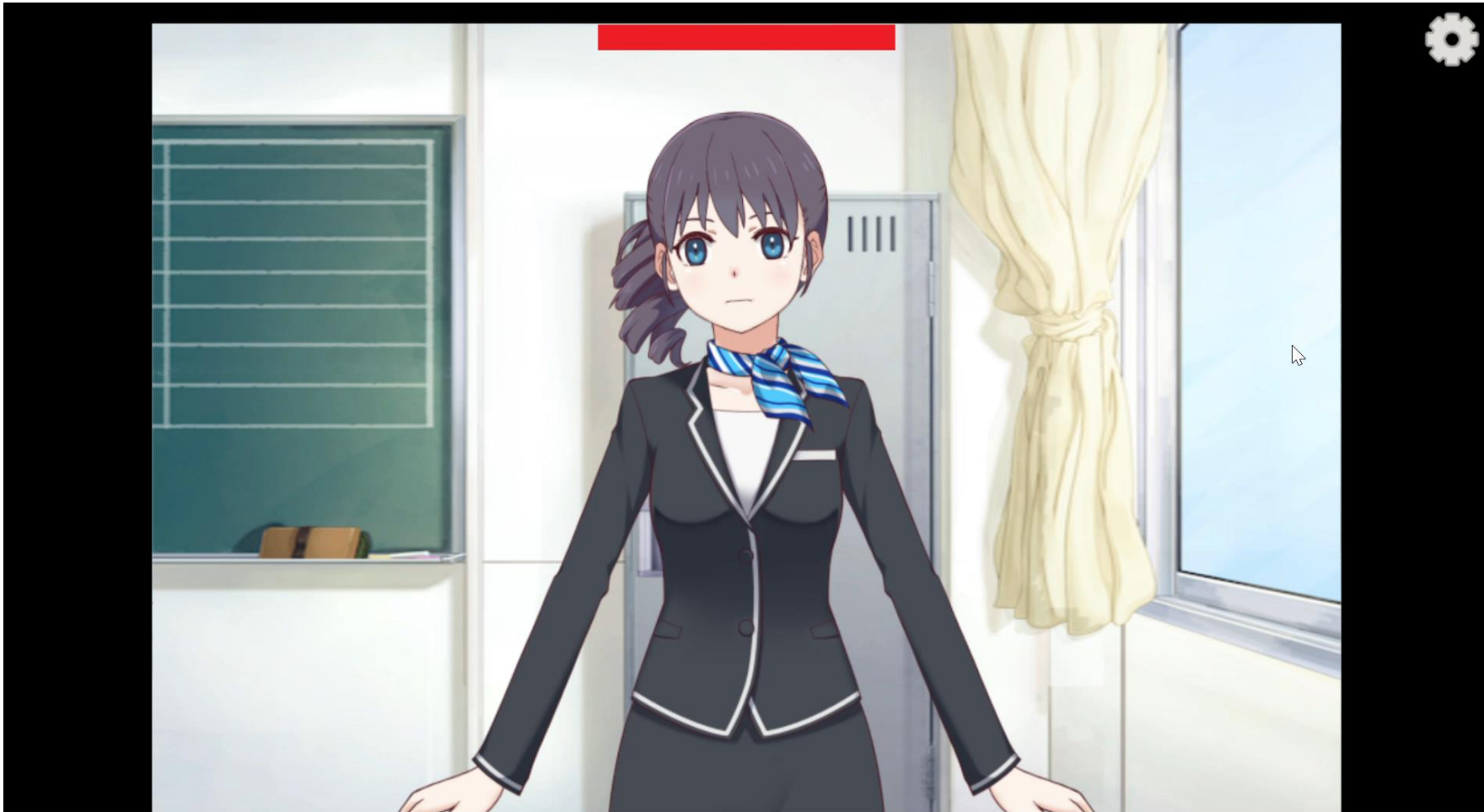(with intensity, saturation decrease, contrast decrease, blue enhancement)

Results

User can control live2d model smoothly, and image style can change through specific expression.

Reference / Related Work

Jialing Feng, Zhexiao Guo, Jun Wang and Guo Dan. (2020). Using eye aspect ratio to enhance fast and objective assessment of facial paralysis. *Computational and Mathematical Methods in Medicine.* 1–11. https://doi.org/10.1155/2020/1038906

Jaynish P. Vaghela. (2019, Dec 16) *Asleep at the Wheel: A Computer Vision and Deep Learning Approach to Detecting Drowsiness.* Medium. https://medium.com/@jaynishvaghela/driver-drowsiness-detection-using-lstm-network-2966c49a5400

Live2D. (2018). CubismWebSamples [Source code]. https://github.com/Live2D/CubismWebSamples

Mmmmmm44. (2021). VTuber-Python-Unity [Source code]. https://github.com/mmmmmm44/VTuber-Python-Unity

thank you all