

# VBridge Manual

---

[Overview](#)

[Getting Started](#)

[Base Settings](#)

[Editor DLC](#)

[Menus and UI](#)

[Settings](#)

[Output Editor](#)

[Overview](#)

[Usage and Examples](#)

[Equation Ex. 1](#)

[Equation Ex. 2](#)

[Equation Ex. 3](#)

[Min Default Max](#)

[Min Max Ex. 1](#)

[Min Max Ex. 2](#)

# Overview

VBridger is a tool that transforms live ARKit/FACS data into additional formats that add to the virtual avatar experience.

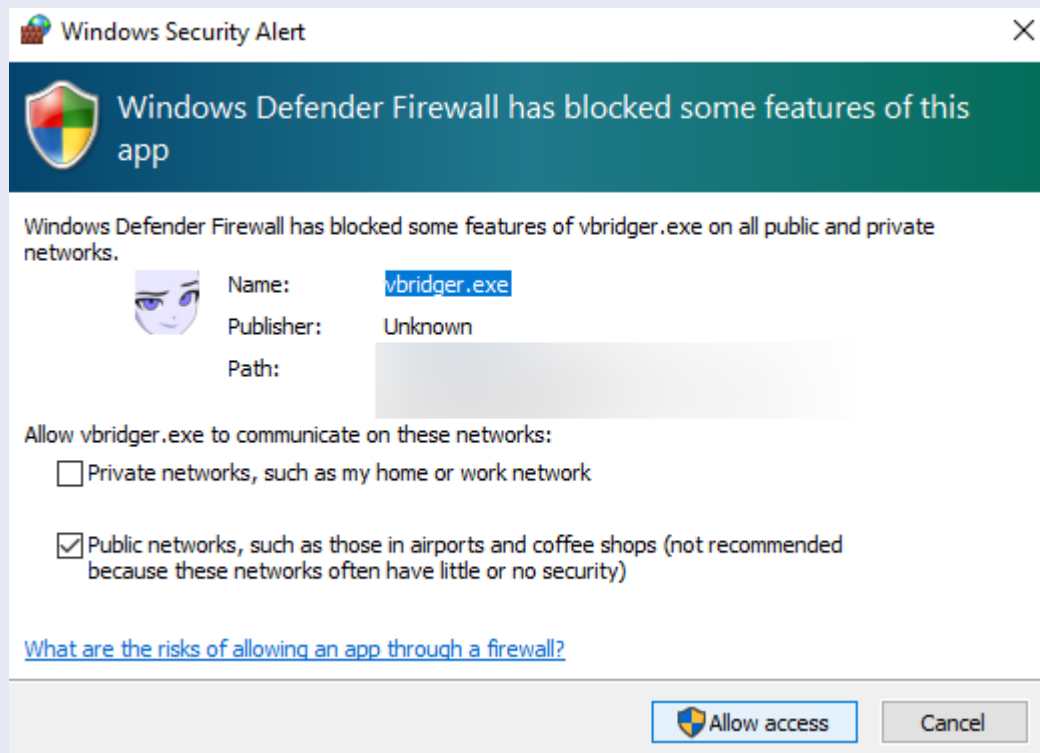
The primary focus of VBridger at the moment is to combine tracking data for use with the Live2D app Vtube Studio by Denchisoft. It functions as an addon that allows the user to make full use of the ARKit tracking data for more realistic model movements.

The base version of the app uses presets that have been officially created by PiPu Productions, and the DLC allows creators to create their own setting files. Anyone with the base app can load the settings generated by a creator.

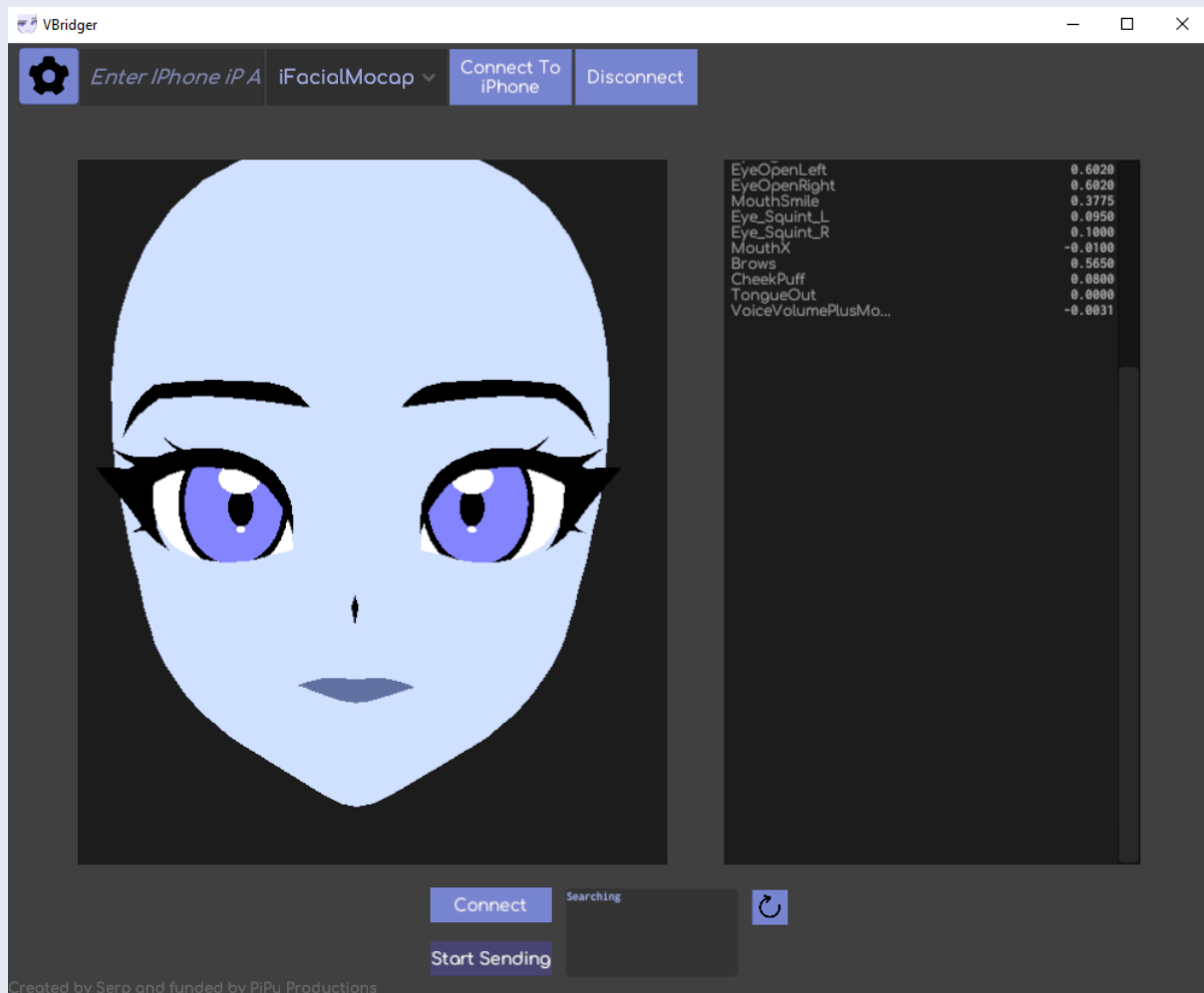
## Getting Started

1. Open the app via Steam.

Windows Defender may ask you to allow the app to use your internet, make sure it is allowed on Public networks or it won't be able to connect to your phone to receive tracking data.



You should see a screen like this after it loads.

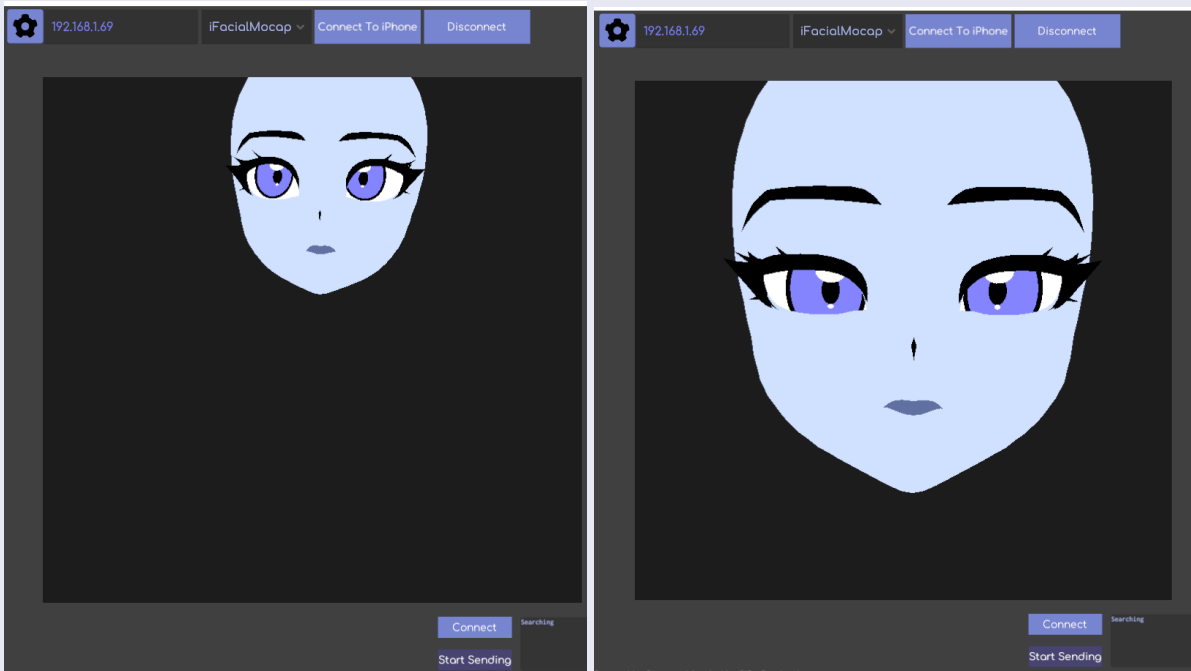
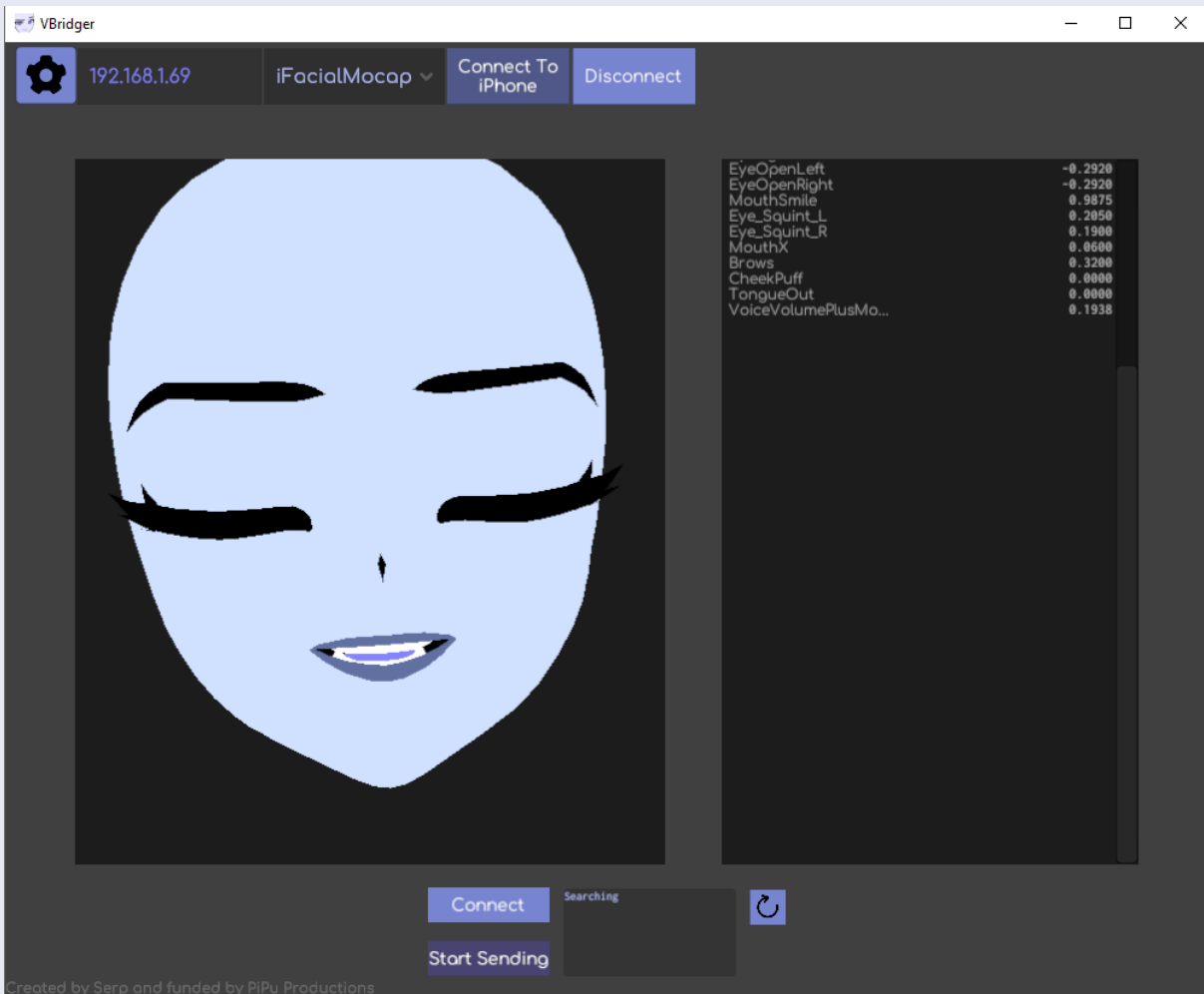


2. Find the local IP of your phone, it can generally be found in the Wifi settings. In apps like ifacialmocap it may be displayed along the top of your phone. It should look something like 192.168.1.XX (XX being random numbers). There will be no letters in the IP address.

VBridge Supports ifacialmocap(iPhone), FaceMotion3D(iPhone), as well as Meowface (Android). You can select which app you'd like to use by clicking the drop-down on the top bar.

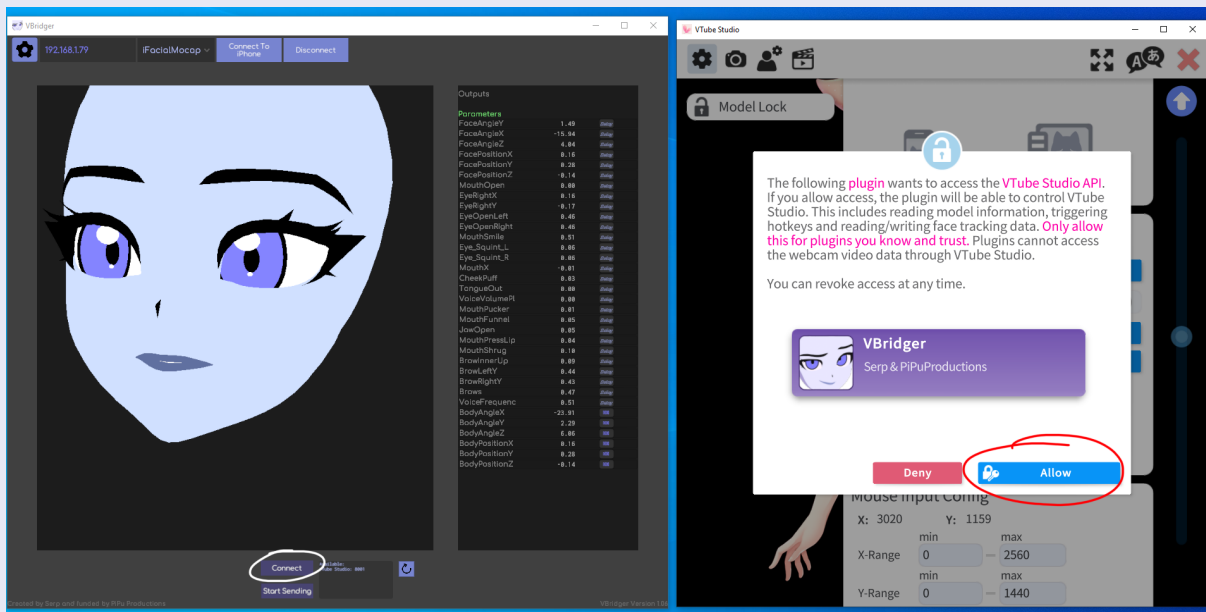
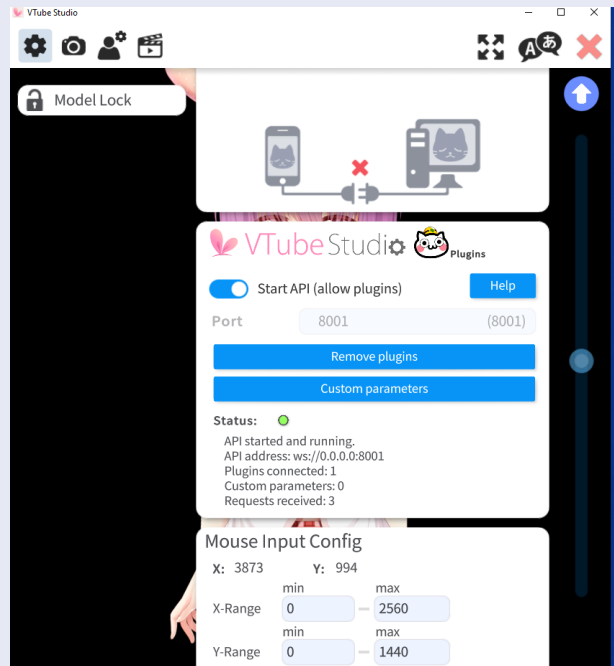


3. With your phone's IP in the address bar and the app open on your phone, select "Connect to iPhone"  
The face should begin moving now, matching what you do!

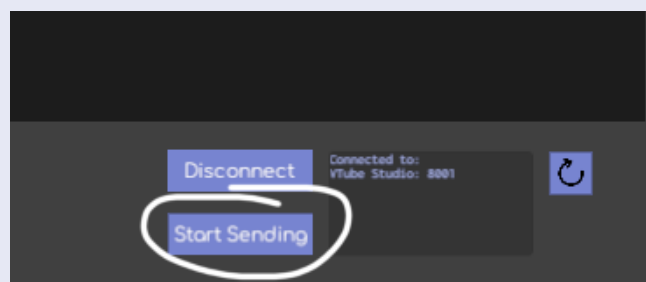


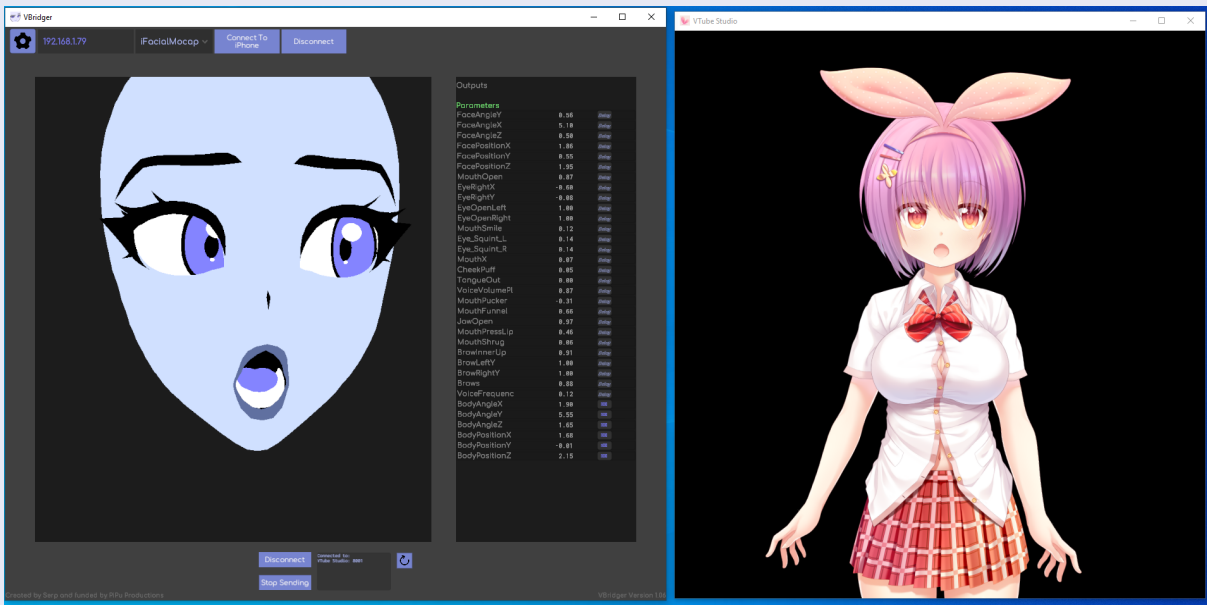
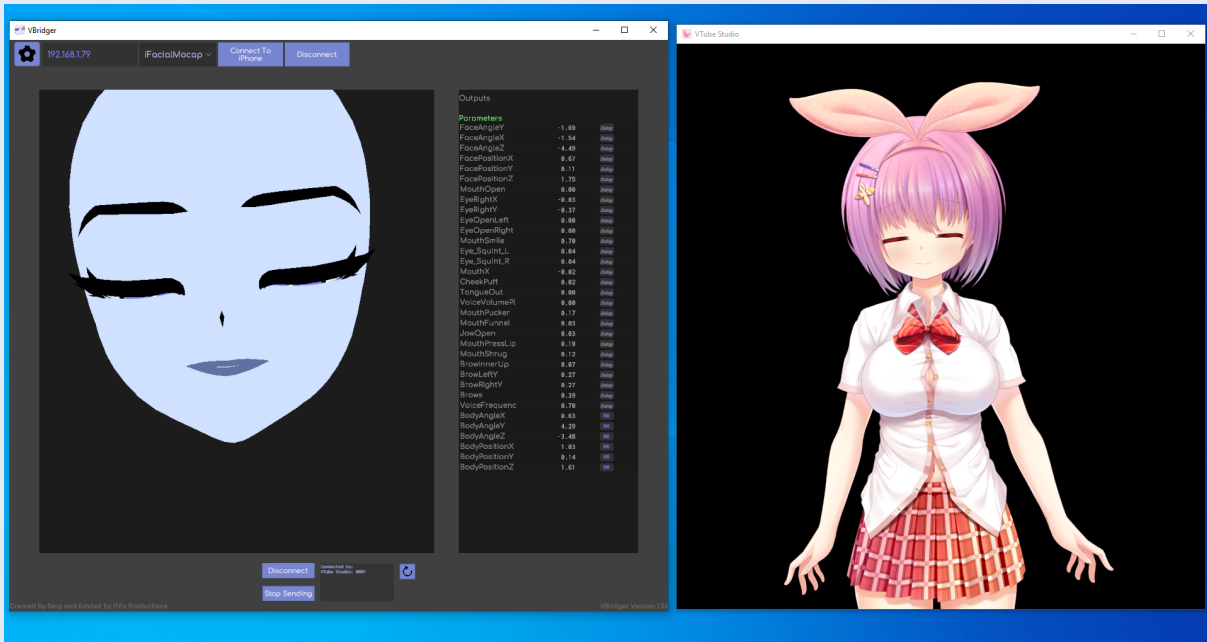
If your head is small and high up, press the "reset position" button on your phone, otherwise the position information will be incorrect!

- By default VBridger loads a file that creates outputs compatible with VTube Studio.
- In order to Connect to Vtube Studio, open the Vtube studio app. Make sure that the "Start API" is switched on. Then hit "Connect" On VBridger. Vtube studio will ask you to allow the plugin. Hit "Allow"

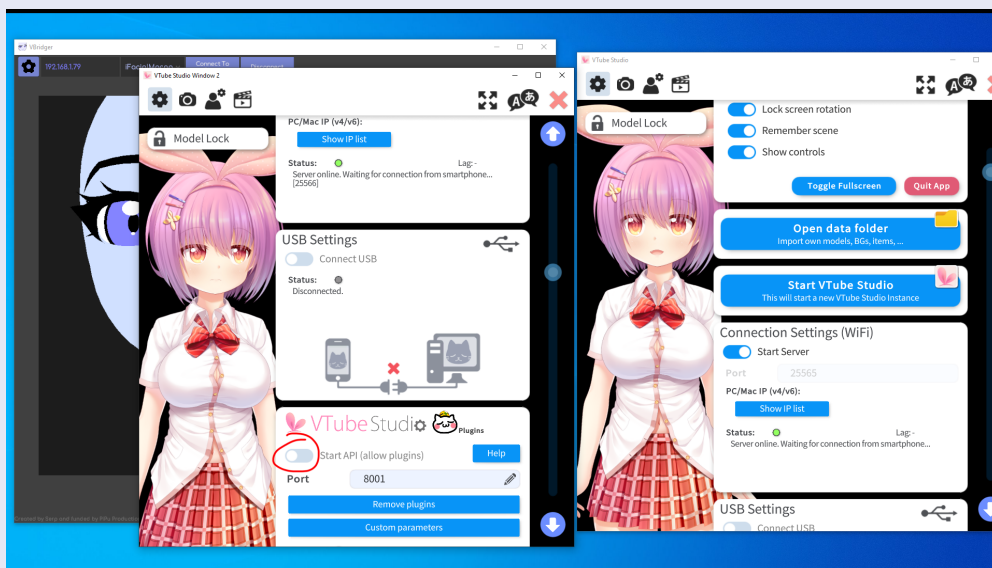


- Now all you need to do is hit "Start Sending" On VBridger and the model will start moving

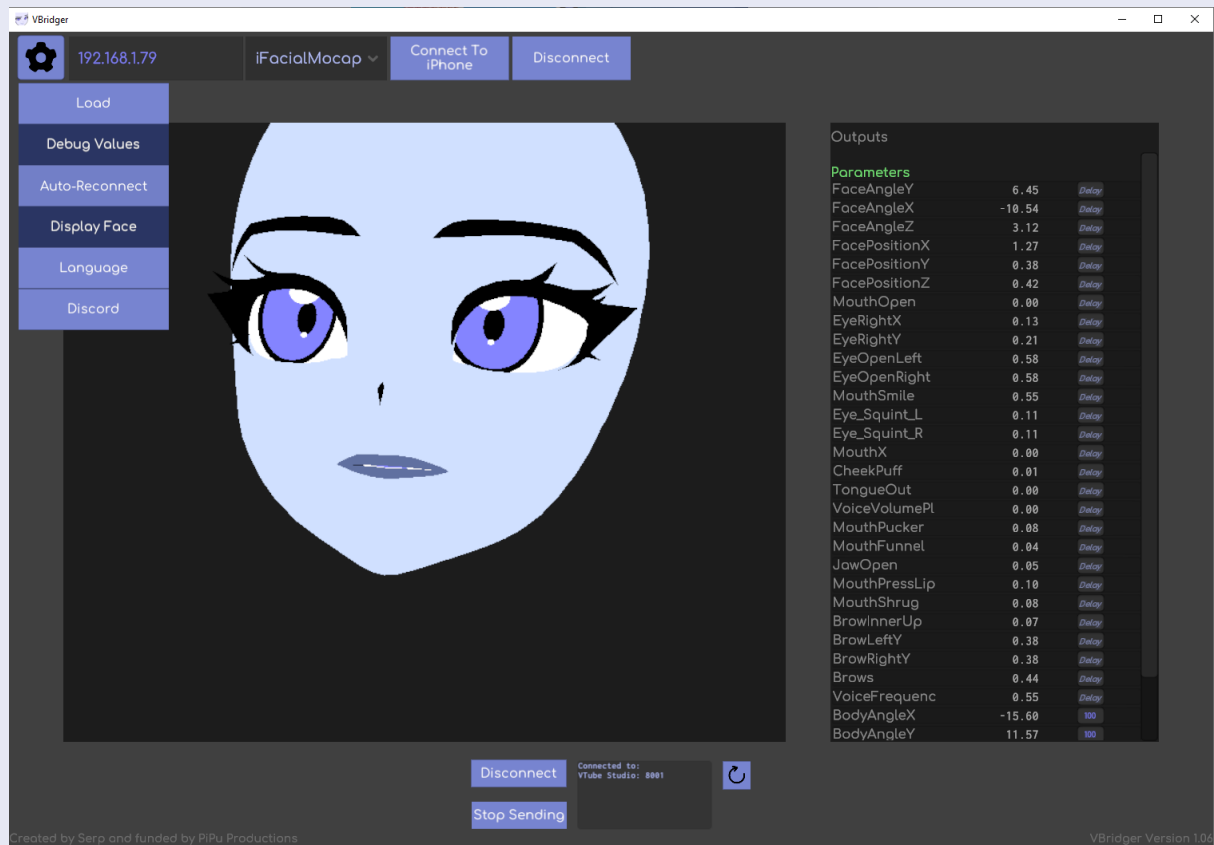




- By default, VBridge will send your tracking data to Vtube Studio and any additional windows you open. In order to stop VBridge from sending to a window of VTube Studio, turn off the API.



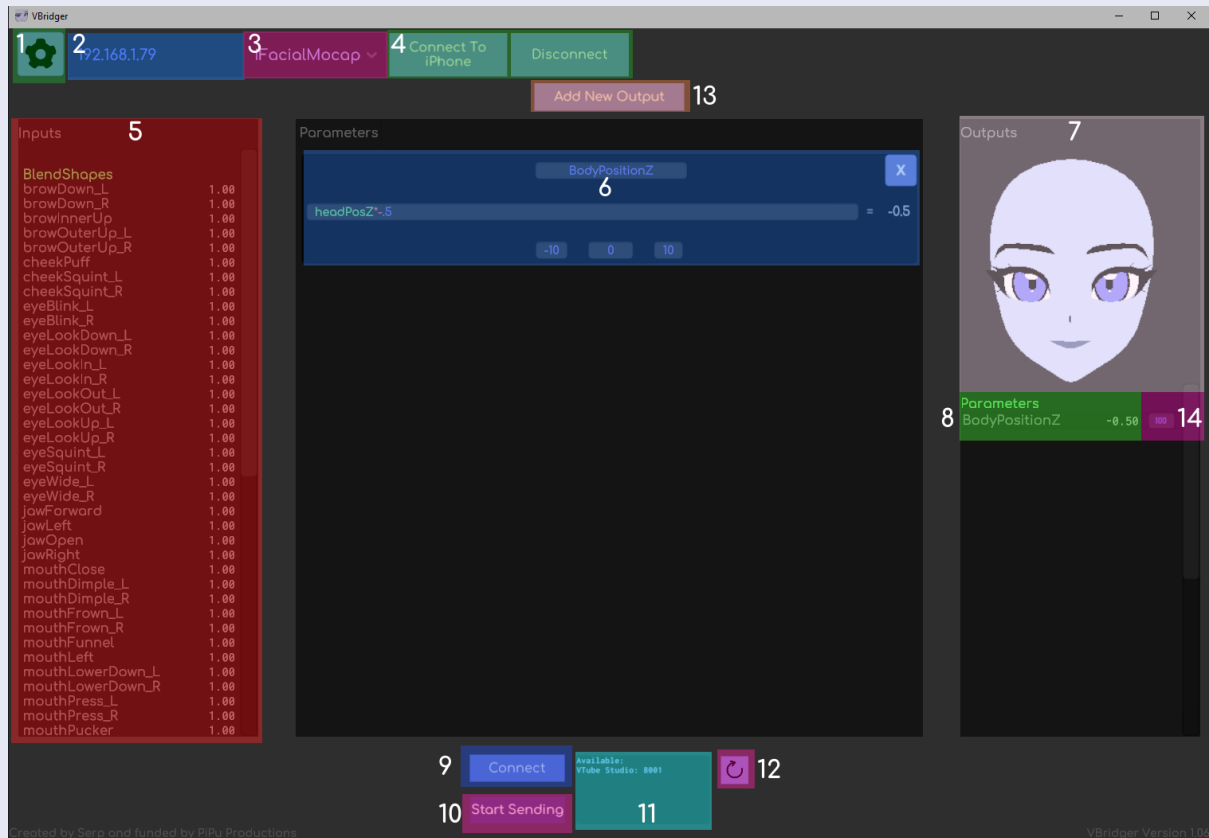
# Base VBridger Settings



1. **Load**  
Loads Saves
2. **Save**  
Save the Delay edits
3. **Debug Values**  
Shows the values next to the Outputs
4. **Display Face**  
Toggles the 3D face preview
5. **Language**  
Change language settings. Does not affect output names, just UI.
6. **Delay**  
Change the Delay (milliseconds) of the parameter Output

# Editor DLC

## Menus and UI

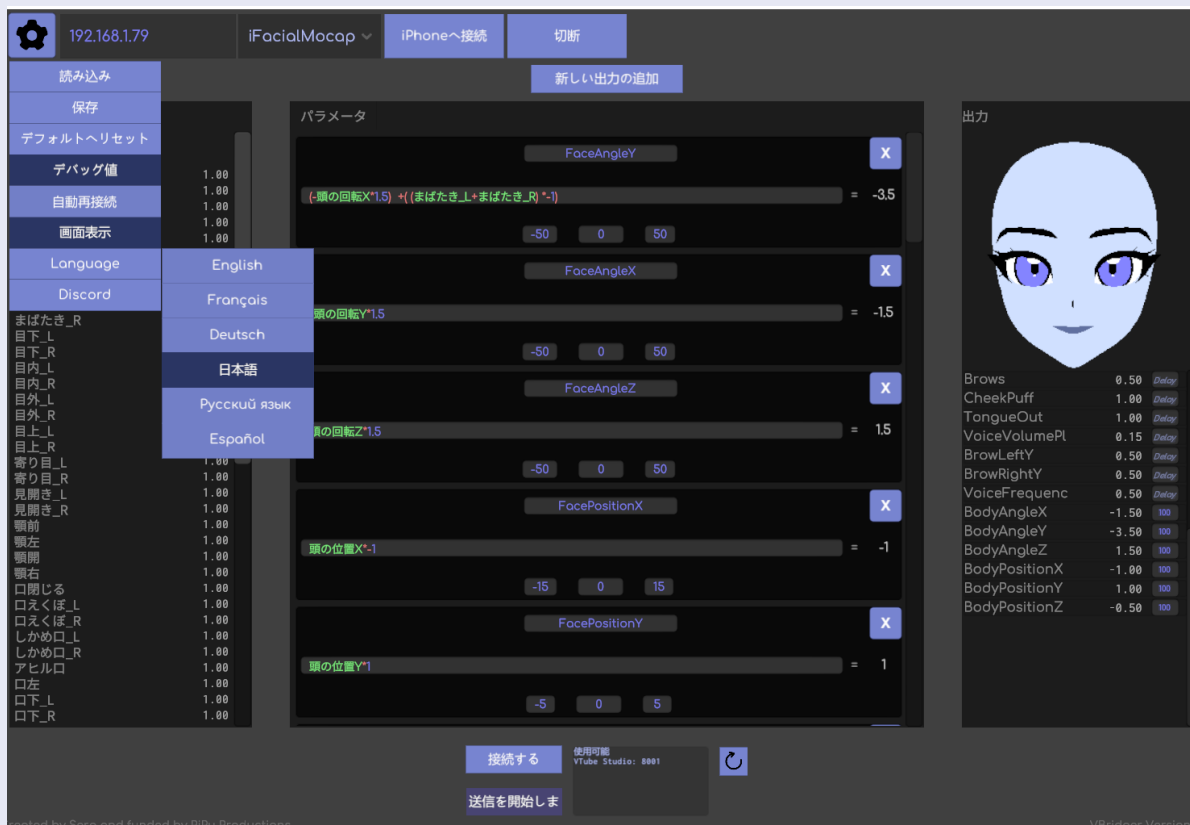
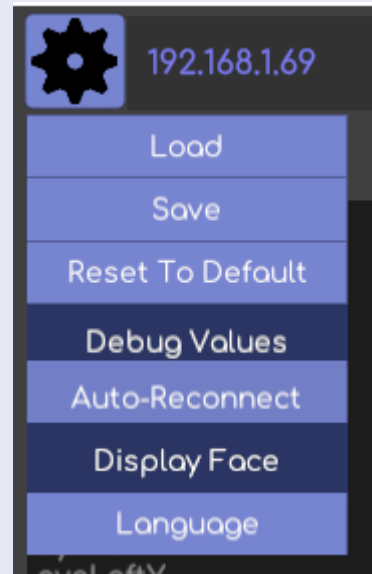


1. Settings
2. Phone IP
3. Tracking App Selection
4. Connect/Disconnect from Phone
5. Input Values
6. Output Editor
7. 3D Preview
8. Outputs
9. Connect (to Vtube Studio)
10. Start Sending (Begins sending Data)
11. Available Vtube Studio Windows
12. Refresh (must be pressed when opening/closing Vtube Studio windows!)
13. Add Output Button (Creates a new equation)
14. Delay Editor



# Settings

1. **Load**  
Load any settings files
2. **Save**  
Save any Settings file
3. **Reset to Default**  
Resets Outputs to whatever is in the "Default.vbridger" file in the Saves.  
If you overwrite your Default file, you can "Verify the Integrity of Files" via steam, and it'll re-download the factory settings.
4. **Debug Values**  
Toggles all the live number values
5. **Display Face**  
Toggles the 3D preview face
6. **Language**  
Change language. Language affects the equations editor so the equations can be typed in the language of the program.



# Output Editor

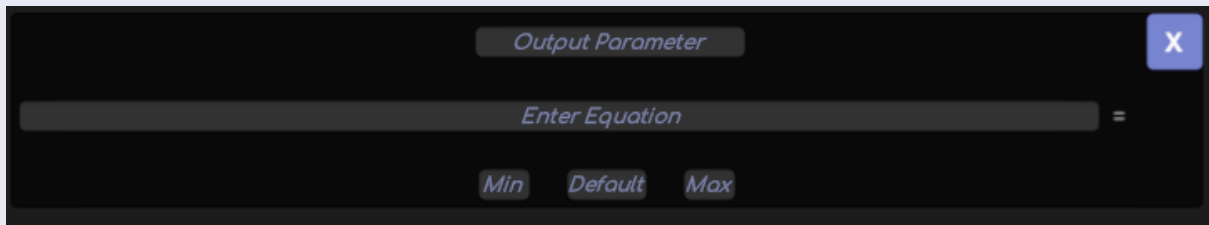
## Overview

In the output editor, the Inputs can be used as variables in equations. Each output added will create a box where all the relevant information for that Output exists.

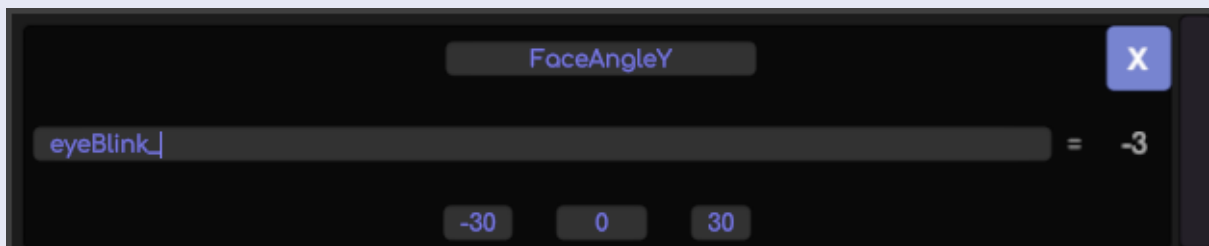
The top box is the name of the Output. This cannot be changed while VBridger is connected to any external app.

The middle box is where you can enter your driver equation.

The bottom boxes are used to communicate information to the receiving app, to let it know what range the values of the equation should fall within.



The names must be typed perfectly, and will turn green once recognized.



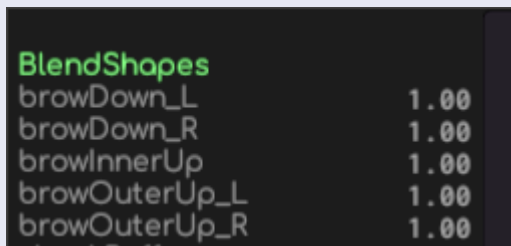
If typed incorrectly, the text will remain blue and when you hit enter, the box will turn red to show you that the equation isn't viable.



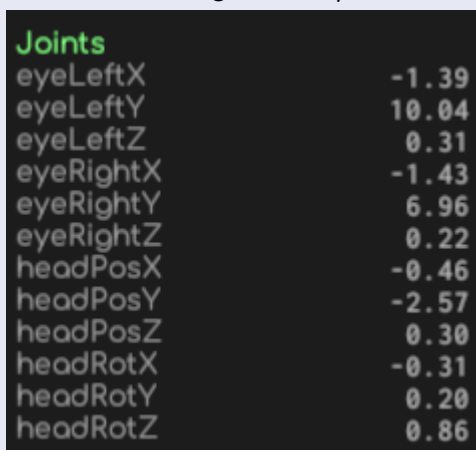
All math will be pinkish and plain values will remain blue.



All blendshape inputs will be a value of 0-1



Joint values are less clear, the Eye rotations are generally -15 to 15, and Head rotations are generally -30 to 30.

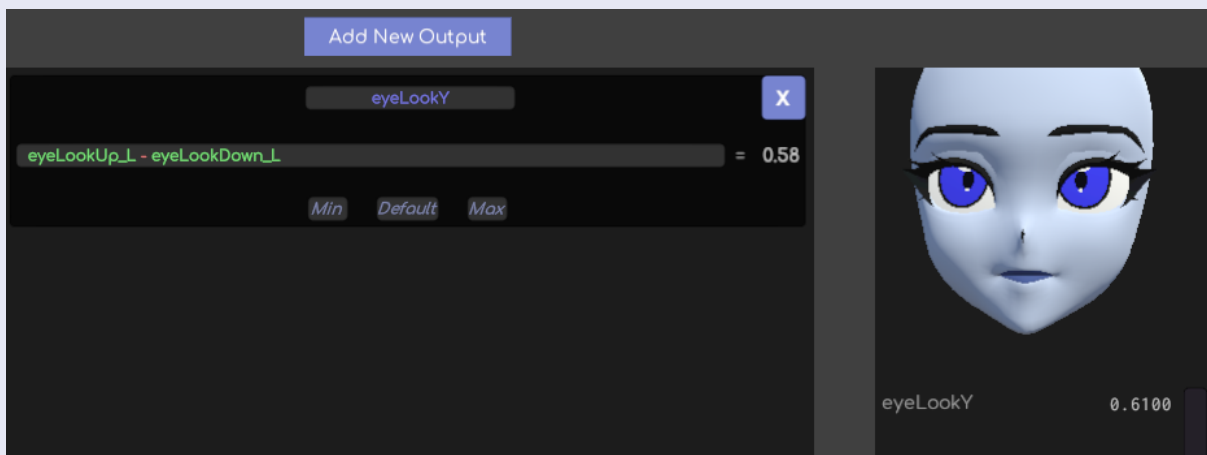
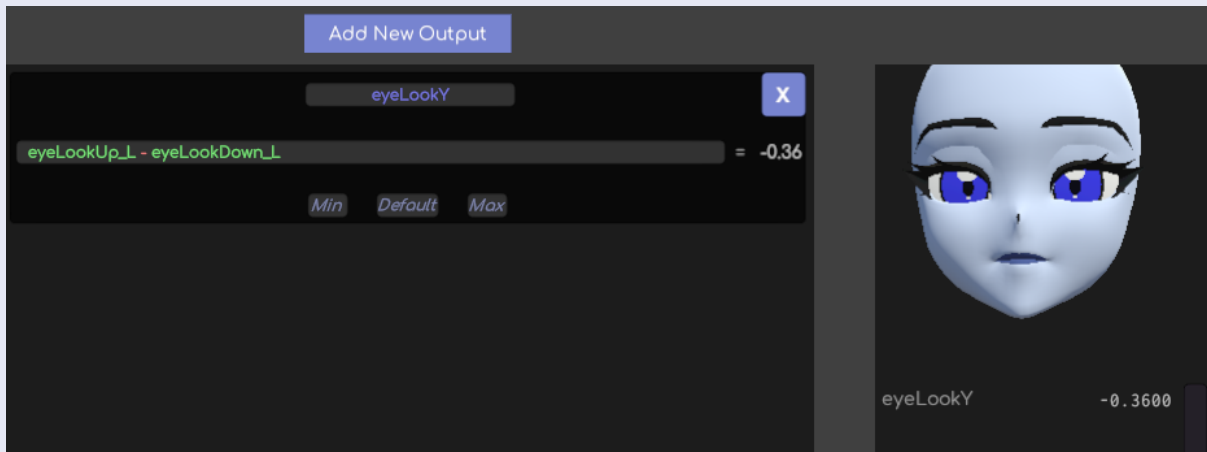


## Usage and Examples

### Equation Ex. 1

Eye Look Down is subtracted from Eye Look Up, so when I look down on the phone the output value goes into the negatives, and looking up stays in the positives.

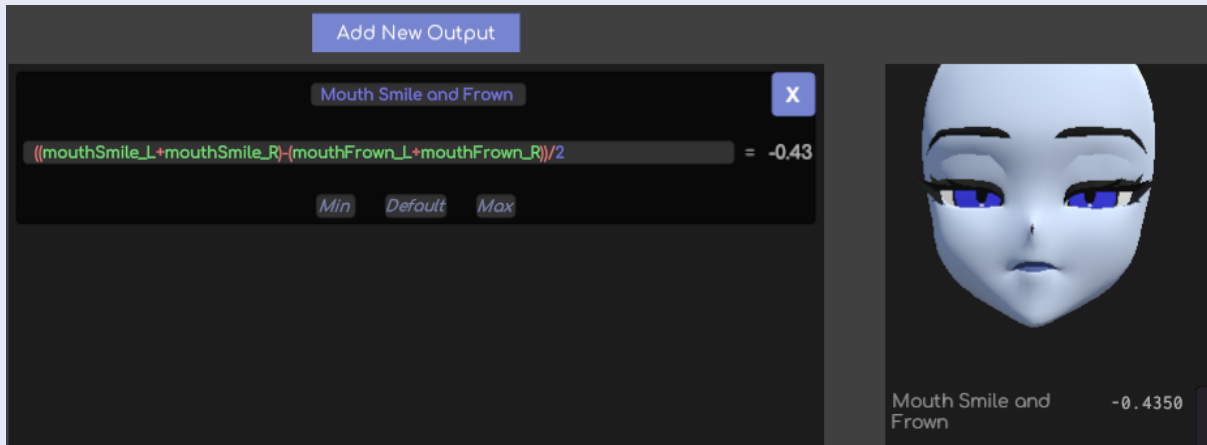
By doing this, we've combined two inputs into one output, simplifying the data for Vtube Studio.



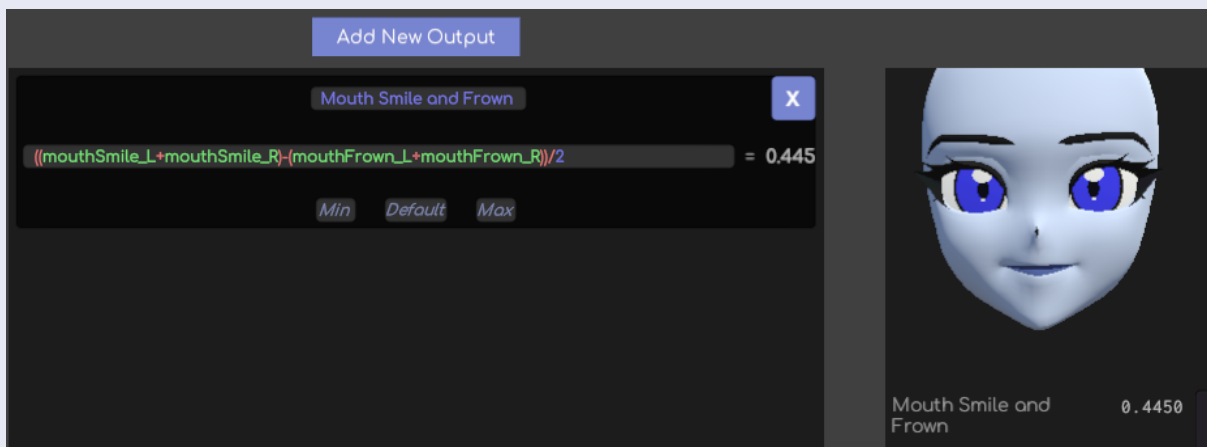
## Equation Ex. 2

Iphones track Smiling and Frowning on both left and right sides of the face, which creates 4 parameters. We can simplify this to just 1 by averaging smiling and frowning, and doing what we did above by placing the frown on the negative value.

The whole thing is divided by 2 to average out the left and right sides instead of just adding them together.



The screenshot shows a software interface with a dark background. At the top, there is a blue button labeled "Add New Output". Below it, a panel titled "Mouth Smile and Frown" contains a text input field with the equation  $((\text{mouthSmile\_L} + \text{mouthSmile\_R}) - (\text{mouthFrown\_L} + \text{mouthFrown\_R})) / 2$  and a value of  $= -0.43$ . Below the input field are three buttons: "Min", "Default", and "Max". To the right of the panel is a 3D model of a character's face with a neutral, slightly downturned expression. Below the 3D model, the text "Mouth Smile and Frown" is followed by the value  $-0.4350$ .



The screenshot shows the same software interface as above. The "Mouth Smile and Frown" panel now displays the equation  $((\text{mouthSmile\_L} + \text{mouthSmile\_R}) - (\text{mouthFrown\_L} + \text{mouthFrown\_R})) / 2$  with a value of  $= 0.445$ . The 3D model of the character's face now has a slight smile. Below the 3D model, the text "Mouth Smile and Frown" is followed by the value  $0.4450$ .


### Equation Ex. 3

Mouth Dimple activates when the mouth is closed, where mouth Stretch activates when the mouth is Open. Pucker is the opposite shape of both, so we can average the Dimple/Stretch Inputs and then subtract Pucker to create a single Output from the 5 inputs!

Kissy Face Stretch Face X

$(\text{mouthDimple\_L} + \text{mouthDimple\_R} + \text{mouthStretch\_R} + \text{mouthStretch\_L}) / 4 - \text{mouthPucker} = -0.667$

*Min* *Default* *Max*




Kissy Face Stretch Face -0.6675

Kissy Face Stretch Face X

$(\text{mouthDimple\_L} + \text{mouthDimple\_R} + \text{mouthStretch\_R} + \text{mouthStretch\_L}) / 4 - \text{mouthPucker} = 0.402$

*Min* *Default* *Max*



Kissy Face Stretch Face 0.4025

## Min Default Max

These are values that Vtube studio uses when you set a VBridger output as a tracking input there.

These don't affect the equation, but they change how vtube studio interprets it.

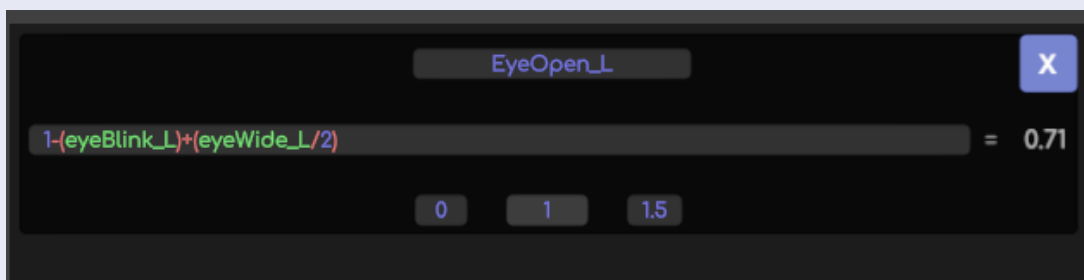
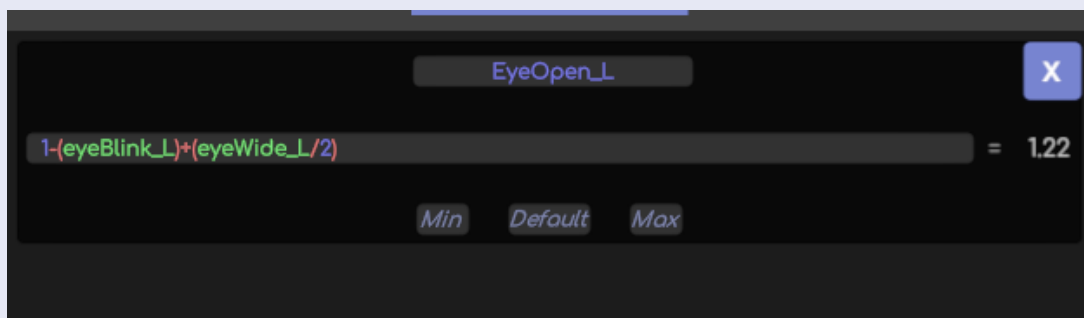
The Min is what Vtube studio expects to be your minimum value. The Max is what Vtube studio expects the highest value to be. The Default is somewhat unimportant, but it should be whatever value your equation uses as a base. Most of the time it will be 0, but a blink for example might be 1 when the eyes are open or in "standard" position.

## Min Max Ex. 1

For this EyeOpen equation we are combining blink and eye wide to create one parameter. In Live2D, the blink value is 0 when the eye is closed, so we are going to do some math to reformat the values to match how the live2D parameters work.

We can subtract eyeBlink from 1 to invert the value. Now instead of being 1 when blinking, it will be 0, and instead of 0 while the eye is open, it will be 1. This means that our default value should be 1. Our minimum value will be 0 because we reach the blink when the value hits 0. We won't need to go into the negatives at all.

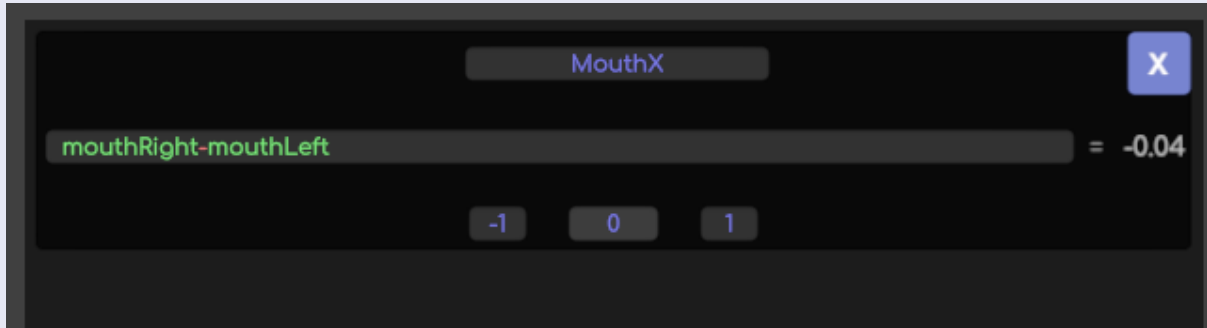
Eye Wide is usually rigged at 1.5 in Live2D parameters, so we want to halve the value. So when the eye is widest, the highest possible value is 1.5 instead of 2. So we will add a divided eyeWide to the blink equation. We'll set the Max to 1.5, and our Output is finished.



## Min Max Ex. 2

### Mouth X

Like the smile/frown equation, MouthX has a negative and positive value. The value will only go to -1 at minimum and 1 at max, and the neutral position (0) will be our default.



## Delay

Delay can be added to the Output as a modifier.

This will delay the sending of the information for the amount specified (in ms). This allows you to play with the timing of parameters.

