OLIVE (OpenCV **LIVE)** is a web-based, visual programming tool for real-time image, video, audio and 3D processing. It uses **nodes** to represent functions and **wires** (connections) to define the flow of media data between them, leveraging **OpenCV.js, Three.js** and **MediaPipe**.

I. System Requirements and Setup

- **Browser:** A modern web browser that supports **WebGL** and **JavaScript**.
- **Dependencies:** The application loads external libraries including opencv.js and three.js.
- **Status Check:** When the editor loads, the **OpenCV status** will be displayed in the interface, turning green once the library is ready.

II. Core Editor Concepts

Component	Description	Interaction
Node	A block representing a specific function.	**Add:** Drop the Node into the **Graph Editor.** **Remove:** Select and **Right-click** or press **Delete** (*continuous click on touchscreens*)**.**
Port	Connections for data flow. **Inputs** (left) receive data; **Output** (right) sends data.	**Create Connection:** Drag from an **Output Port** to an **Input Port** (*on a touchscreen, click the output first and then the input*).
Wire	Defines the flow of data from an output to an input.	**Delete Connection: Right-click** on the **Port** (*continuous clicking for touchscreens*).
Fullscreen	View media output in full screen.	**Double-click** on the image/video/canvas (*continuous click when on a touchscreen*).
File	Load and save graphs in **JSON** format.	**Click** the **Save** (💾) icon to save the project. **Click** the **Open** (📂) icon to load a project. Local media files must be manually uploaded as soon as the project (**.olive**) file is ready.

III. Node Catalog and Basic Functionality

Nodes are organized into categories based on their role in the workflow.

1. Input Nodes

These nodes provide the starting media source for your graph.

- **Image / Video / Audio:** Upload a static image, a video or a sound file (or provide a link).
- **Camera:** Capture a live video feed from your device's camera.
- **Parametric Lines:** Change the parameters and generate animated curves.

2. Processing Nodes

These nodes perform image and video manipulation using **OpenCV.js**.

Node Title	Primary Functionality	Key Controls & Parameters
Gamma Correction	Changes pixels' intensities.	Adjust **γ** value.
Histogram Equalization	Equalizes the histogram, globally or locally.	Adjust the **Grid Size** for **CLAHE** (Contrast Limited Adaptive Histogram Equalization)
Color Adjustment	Manipulates the HSV color space.	Use the **Color Picker** to define the color (or the target range) and the sliders for new **Hue** and **Saturation** values. Includes **Full Range** and **Invert** options.
Convolution	Applies spatial filtering.	Select **Filter Type** (e.g. Gaussian Blur, Sobel Edges) and adjust **Kernel Size** or define a **Custom Kernel**.
Morphology Rank	Applies morphological operations and ranking filters.	Select **Filter Type** (e.g. Erosion, Dilation, Median), **Kernel Size**, and **Kernel Shape** (Rectangle, Ellipse, Cross).
Polar Transformation	Warps the image using coordinate transformations.	Select **Effect Type** (Fish Eye, Cone, Swirl) and **Effect Strength**.
Glitch Effects	Applies dynamic visual distortions.	Select **Effect Type** (Shaking, Aberration, Fade, Glass) and **Effect Strength**.
Thresholding	Binarizes the color channels, using a global or local threshold.	Adjust the **Threshold Value** and select the **Type** (Binary, Otsu, Adaptive).
Matrix Operations	Adds, subtracts, multiplies, divides two images or applies min/max operations.	Select the **Operation Type** and the **Weights** for the inputs.
Channel Mixer	Remixes the RGB channels.	Adjust the **Percentage** of **Red, Green and Blue** at every color channel.
Color Blending	Replaces the color (e.g. Hue, Saturation) of an image with the color of another.	Select the **HSV** channels (Hue, Saturation, Value) to be replaced.

Concatenation	Merges the input images either horizontally or vertically.	Adjust the percentage of **Overlap** with the slider.
Transitions	Classic WebGL transitions between two inputs.	Select **Transition Type** (e.g. Fade, Wipe, Radial, Dissolve) and adjust the **Transition Duration**.
Connected Components	Detects superpixels in the input image.	Adjust the **Threshold Value** for the **Segmentation**.
Background Subtraction	Removes non-moving objects (background) from the input video.	No controls.
Optical Flow	Visualizes the optical flow with arrows.	Adjust the **Block Size**.
Skeleton	Applies the Distance Transform to the RGB channels.	No controls.

3. Rendering Nodes

They **r**ender the input texture onto a **3D geometry** using **Three.js**.

- **Projection:** A specialized node that uses the input as a **projected texture** onto the scene. Use the mouse to **orbit and zoom** the camera. A spherical indicator represents the **Projector** which can be dragged using **Drag Controls**.
- **Mapping / Lighting:** UV mapping of the input image/video on a **GLTF Model** or a Plane, Cube, Sphere etc. The spherical indicator represents a **Point Light**. Adjust light's color and intensity with the color picker and the sliders.
- **3D Text:** renders custom 3D text. Adjust content, font, color, size and camera position.

4. Pose-Estimation Nodes

They track the human body using the **MediaPipe** library.

- **Human Pose:** Applies segmentation to the person (if any) and removes the background.
- **Character Animation:** pose-driven movement of an uploaded **VRM humanoid model**.

5. Output Node

This node does not have an Output Port.

- **Canvas Viewer:** This is the final step in the graph. It displays the result of the connected node.

IV. Additional Functions

In this section, special features (not described above) are presented.

1. Node Bypass

To temporarily deactivate a connected node, uncheck the box next to the Node Title (top right corner). This allows you to *test* the effect of a particular node on the data flow.

2. Control Ports

The **Human Pose Node** sends *pose estimation result data* through its Control Output. Connect this port to the Control Input (if available) of any node, in order to trigger specific events (e.g. *play* a video/sound or a transition, *pause* a video/sound already playing or *reverse* a transition, *position* the camera, *adjust* a slider value, *move* the model in a Character Animation Node etc.). You may change the default gesture, that is raising and waving the right hand, from the **Settings** (⚙) menu on the top right corner of the Graph Editor.

3. Keyframes

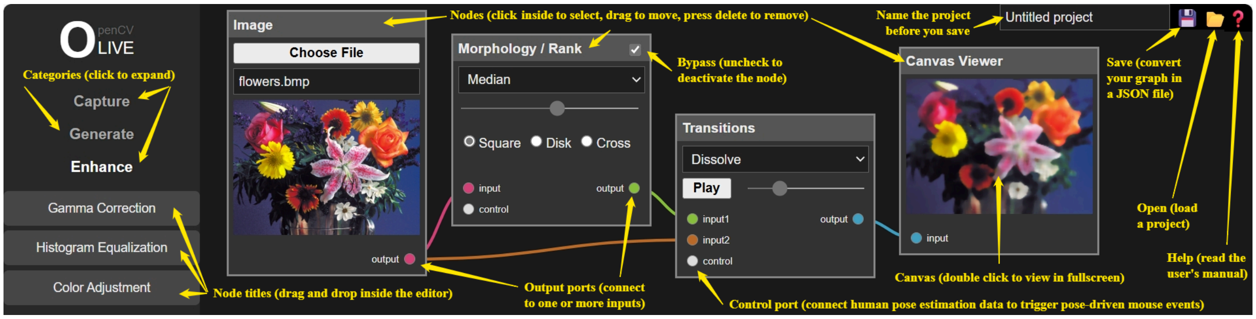
The **Countdown / Keyframes Node** allows for *time-driven* events triggering. Set the **Countdown Timer**, determine a **Target Value** (or two) for the affected parameters and connect the Control Output to any node's Control Input (except from the Character Animation Node that accepts only **MediaPipe** data from the Human Pose Node). The countdown starts by clicking the node's **Trigger Button**, or by another node's Control Output.

4. Sound Filters & Effects

This node takes an audio (or video) as input and applies audio processing (e.g. Band Pass, Reverb, Flanger, Panning). It has a Control Port as well and no output. No more than one such node may be connected to a video/audio Output Port.

V. Graphical User Interface

https://opencv-live.github.io/examples/images/gui.png


OpenCV LIVE
Categories (click to expand)
Capture
Generate
Enhance
Gamma Correction
Histogram Equalization
Color Adjustment
Node titles (drag and drop inside the editor)
Image
Choose File
flowers.bmp
output
Nodes (click inside to select, drag to move, press delete to remove)
Morphology / Rank
Median
Square
Disk
Cross
input
control
output
Output ports (connect to one or more inputs)
Bypass (uncheck to deactivate the node)
Transitions
Dissolve
Play
input1
input2
control
output
Control port (connect human pose estimation data to trigger pose-driven mouse events)
Name the project before you save
Untitled project
Canvas Viewer
input
Canvas (double click to view in fullscreen)
Save (convert your graph in a JSON file)
Open (load a project)
Help (read the user's manual)

VI. Examples

https://github.com/opencv-live/opencv-live.github.io/tree/main/examples

To run the examples: hover over the **Open** (📂) icon at the top right corner of the Graph Editor and select a **Sample Graph** from the list. You may also download the **.olive files** (not the images folder) to your PC and then load any of them.