PreviewVideoMonitorPro User Manual

Version 5.2 | Professional Monitor and Video Inspection for ComfyUI

Welcome! 🖐

PreviewVideoMonitorPro transforms how you review AI-generated video and images in ComfyUI.

Whether you're creating videos at home or managing a studio pipeline, this tool gives you frame-accurate playback, technical inspection capabilities, and professional comparison tools—all without leaving your workflow.

This manual is designed for everyone: if you're new, follow the guided walkthrough. If you're an advance/tech/coder user, check the Advanced Notes at the end for details.

Quick Start: Your First Session

We´ll be making a video tutorial soon. Check for that link on https://github.com/CrateTools/comfyui-preview-video-monitor

Setting Up the Node

Add the Node In ComfyUI search `PreviewVideoMonitorPro`

Connect Your Input Connect either: - Video source (MP4, etc.) to the `video` input - Image sequences data to the `images` input (vae decode) - Image (single) to the `images` input (vae decode or a load image node)

source Selects the input type: video or images. Use video when connecting a VIDEO output, or images when connecting an IMAGE sequence or single image.

monitor The `monitor` parameter lets you select which physical screen displays the PVM viewer

power_state Set `power_state` to ON (true). Turn it Off to switch to another physical monitor and then back to On.

target_resolution Sets the internal processing resolution for the selected `monitor`: 1920x1080 or 3840x2160 denpending on your physical monitor resolution, GPU video card and graphics settings.

generations_name A custom label prefix for saved generations (e.g., "Generation", "the project name", "Test", "Final"). This name appears in the Generations dropup and is used when naming cached files.

Useful if you´ll be creating a lot of iterations and you need a base name for all gens or a "project name" prefix, etc. This can be changed later at any point on the Generations dropup.

Snapshot Workflow When enabled, saving a snapshot will also embed the workflow JSON that generated it. This lets you recover the exact workflow settings later.

Snapshot Path Where snapshots are saved. Use smart for the default location (snapshots/ inside the node´s folder in custom_nodes), or enter a custom directory path.

video Output

At the moment, we need this output for technical reasons. Passes through the input video, allowing the node to be chained with other video nodes. Not tested and would be a rare use for this node. Not recommended at all.

Run Your Workflow Queue your prompt. You'll see a waiting screen. The monitor window opens automatically on your selected display.

Keyboard Reference (Complete)

Key	Action
SPACE	Play/Pause
←	Previous frame (hold for scrubbing)
→	Next frame (hold for scrubbing)
END	Display selected Generation
I	Set IN point
O	Set OUT point
P	Reset IN OUT Points
1	Toggle 1:1 mode
2	Width mode
3	Height mode
4	Fit mode
5	Toggle Fullscreen
R	Toggle Red channel
G	Toggle Green channel
B	Toggle Blue channel
W	Enter and Exit wipe/SBS modes
Shift+Q	Close monitor window and end it´s running process
Mouse Wheel	Zoom in/out
Right-Click + Drag	Pan (when zoomed)
Middle-Click	Reset zoom

Understanding the Monitor Window

When the monitor opens, you'll see:

The Main Viewing Area Your video displays here. This is where all the magic happens—playback, inspection, comparison.

The Toolbar (Bottom) A clean row of controls for everything you need: - Timeline with frame counter - Playback controls (IN/OUT markers, play/pause, frame stepping) - FIT modes (how video scales to screen) - Vision controls (technical inspection) - Generations dropdown (manage multiple runs) - Snapshot (save workflows) - Clear Cache (manage storage)

Chapter 1: Playback & Navigation

Basic Playback

Play/Pause Click the ►/❚❚ button or press `SPACEBAR`

The video plays at the speed you set in the FPS control (more on that in a moment).

Frame Stepping - `← Left Arrow`: Previous frame - `→ Right Arrow`: Next frame - Hold arrow keys: Smooth 30fps scrubbing through your video

Scrubbing the Timeline Click and drag anywhere on the timeline bar to jump instantly to any frame. The frame counter updates in real-time.

Setting Your Playback Speed

The FPS Button (grey circle with a dot) Click it to open the FPS dropup menu:

FPS Presets: Choose fps preset - usual industry standards - 60, 59.94, 50, 48, 30, 29.97, 25, 24, 23.976, 15, 12, 8

Custom FPS (Yellow Box): Click the yellow-outlined field at the bottom, type your exact frame rate (e.g., 120), press Enter.

First Frame Number (Blue Box): Any video starts counting at 1, but sometimes you may want to change it to match editing or other departments so that everyone is talking about the "same frame."

Below the FPS field is a blue-outlined field. This sets what number your first frame displays as.

Example: If a shot/video is frame 1001-1200 of a larger sequence, set this to 1001 so the counter shows the correct frame numbers.

Marking IN and OUT Points

Shortcuts: `I O P`

Sometimes you only want to review a specific section of your video.

Set IN Point: Navigate to your desired start frame, click the IN button. A red pillar appears on the timeline.

Set OUT Point: Navigate to your end frame, click the OUT button. Another red pillar appears. The area between them turns yellow.

Playback with Markers: When markers are set: - Play starts at IN point - Stops at OUT point - Loops back to IN

Pong Mode: Click Pong to enable ping-pong playback (plays forward, then backward, continuously between IN/OUT or the full range)

Toggle Markers Off: Press P or Click IN or OUT yellow and red circles

Keyboard Shortcuts (Playback)

Key	Action
SPACE	Play/Pause
←	Previous frame (hold for scrubbing)
→	Next frame (hold for scrubbing)

Chapter 2: Viewing Modes

Understanding FIT Modes

Your video's resolution might not match your screen. FIT modes control how the video scales and positions.

The FIT Buttons: 1:1 Fit Width Height Fullscreen Reset

1:1 Mode (Keyboard: 1) Shows actual pixels—no scaling. Great for pixel-perfect inspection. Press 1 again to toggle back to Fit mode.

Fit Mode (Keyboard: 2) Scales video to fit screen while preserving aspect ratio. Black bars appear if needed. This is the default.

Width Mode (Keyboard: 3) Scales video to fill screen width. Height may exceed screen (you'll pan to see it all).

Height Mode (Keyboard: 4) Scales video to fill screen height. Width may exceed screen.

Fullscreen Mode (Keyboard: 5) Hides the toolbar, maximizes viewing area. Press 5 or ESC again to exit.

Reset Button Returns to default Fit mode and resets zoom/pan.

Zoom and Pan

Zooming In: Scroll your mouse wheel UP to zoom in. The video enlarges around your mouse cursor position—point at the detail you want to inspect!

Zooming Out: Scroll your mouse wheel DOWN to zoom out.

Panning: Once zoomed, right-click and drag to pan around the video. Zoom can go up to 16x!

Why Use Zoom: Inspect fine details, check edge artifacts, examine texture quality—essential for technical review.

Return to Fit: Middle mouse wheel click or Reset button in the main bar

Keyboard Shortcuts (Viewing)

Key	Action
1	Toggle 1:1

2	Width
3	Height
4	Fit
5	Toggle Fullscreen
Mouse Wheel	Zoom in/out
Right-Click + Drag	Pan (when zoomed)

Chapter 3: The Vision Module ⚡

Vision transforms your monitor into a technical inspection lab. These aren't creative color grading tools—they're diagnostic controls for analyzing what the AI actually generated.

Opening Vision

Click the **Vision** button (toolbar). A panel slides up with technical controls.

RGB Channel Isolation

The R/G/B Buttons: Click to isolate individual color channels: - **R (Red Channel):** See only red information (displayed as grayscale) - **G (Green Channel):** See only green information - **B (Blue Channel):** See only blue information - **RGB (Full Color):** Back to normal view

Keyboard Shortcut:

Use R G B keyboard keys to cycle RGB channels.

Why Use Channel Isolation: AI generation artifacts often hide in individual channels. Compression issues, color banding, and encoding problems become visible when channels are isolated.

The Vectorscope

What Is It: A circular display showing color distribution in your video. Each pixel's color is plotted as a dot.

How to Read It: - Center = No color (grayscale) - Edges = Saturated colors - Specific Directions = Specific hues (Red at 0°, Yellow at 60°, Green at 120°, etc.)

Why Use Vectorscope: Instantly see if your video has color casts, clipping, or unusual color distributions.

Diagnostic Sliders

Gain: Lightens or darkens the image. Drag right to reveal shadow detail, left to check highlight clipping.

Gamma: Adjusts midtone brightness without affecting pure blacks or whites. Reveals detail in shadows (drag right) or highlights (drag left) while preserving contrast endpoints.

Saturation: Boosts or reduces color intensity. Crank it up to spot subtle color artifacts.

Why These Aren't Creative Tools: You're not grading the video—you're testing it. Push brightness to extremes to see if shadow areas hold detail. Max out saturation to reveal color noise.

These controls help you evaluate generation quality—they don't produce an output at all.

Keyboard Shortcuts (Vision)

Key	Action
R	Toggle Red channel
G	Toggle Green channel
B	Toggle Blue channel

Chapter 4: Comparison Tools

When you generate multiple versions, you need to compare them side-by-side. PreviewVideoMonitorPro gives you two professional comparison modes.

The Generations Dropup

Click **Generations** to see all your cached runs. Also navigate them with up and down arrow keys. Then press END key to display the selected one, same as clicking it. - A timestamp (when it was created) - A prefix (v_ for video, i_ for images), what´s node input belongs to, images or video - A comparison button (small square on the right)

Loading a Generation: Press Enter to deploy generations menu or click the Generations button in the main bar. Click any generation name to load it into the viewer.

Renaming: While a generation is selected, press Enter again to rename, then press Enter again to confirm.

Deleting: Middle mouse wheel click over a generation, confirm. The generation is removed from disk.

At the moment of this release, the Generations dropup has a visible max of 10 gens. If more are created they will acttually "be there" but you have to move with the UP and DOWN arrows to "slide/move" the menu vertically and scrub them in. We will add the mouse wheel in the next version of PVM to make this more intuitive.

Comparison Mode: WIPE ◉

Activating Wipe: Click the small square next to any generation to compare to. It turns yellow.

What Happens: As soon as you click and hold, the yellow wipe interface divider disappears and you see a clean wipe that you can move side to side. You can even zoom and pan while in wipe mode.

As soon as you release the left click, the yellow wipe divider interface comes back right where your pointer is. Deactivate wipe with W shortcut.

Wipe Behavior: - Respects your current zoom and pan - Both videos move together - The line stays at your chosen position as you scrub frames

You can grab the yellow line from the small square or anywhere else on the line. The line has a capture radius of 12 pixels which makes it comfortable to use.

Deactivating Wipe: Press `W` (instant exit) or click the square again.

Keyboard Shortcut: `W` = Exit comparison modes

Comparison Mode: SIDE-BY-SIDE ⬗

Activating SBS: Click the yellow square (wipe mode) again. It turns blue.

What Happens: Screen splits in half with a fixed blue line at center: - **LEFT half:** Your current video - **RIGHT half:** Comparison video

SBS Behavior: - Blue line never moves (fixed at screen center) - Each video stays in its own half (clipped at boundaries) - Zoom is synchronized: both videos zoom equally - Zoom anchor: where your mouse is when you scroll - Pan is synchronized: both videos pan together

Why This Matters: Point your mouse at a point on the left generation, zoom in—both generations zoom to the same relative position. You're inspecting the same detail on both generations simultaneously!

Frame-Space Zooming: Unlike wipe mode (which zooms the whole view), SBS zooms each frame independently within its window. The frames can grow huge (extending beyond their boundaries), but clipping keeps them separated.

Cycling Through Modes: - Grey Square = No comparison - Yellow Square (1st click) = Wipe mode - Blue Square (2nd click) = Side-by-side mode - Grey Square (3rd click) = Back to normal

Keyboard Shortcut: `W` = Exit any comparison mode (wipe and SBS) instantly

---

Chapter 5: Managing Your Work

Generations and RAM

How Generations Work: Every time you queue your ComfyUI workflow, PreviewVideoMonitorPro: 1. Decodes your video/images 2. Caches all frames in RAM (for instant playback) 3. Saves frames to disk as JPEG sequence (for later reload) 4. Registers the generation in the Generations dropup

Why Cache to RAM: Reading frames from RAM is 100-1000x faster than decoding from disk. This enables: - Instant scrubbing - Smooth playback - Real-time vision slider feedback - Lag-free comparison modes

The Trade-Off: RAM usage! Just keep an eye on free RAM if you are working with hi-res video and a lot of generations.

Understanding the Cache System (Simple)

Two Caches:

RAM Cache (Fast, Temporary) - Decoded frames in memory - Used for instant playback - Cleared when you close monitor - **Disk Cache (Slow, Persistent)** - JPEG sequences saved in `runs_cache` folder - Survives closing ComfyUI - Can reload quickly (JPEGs decode fast!) loads into RAM again when displayed

Normal Workflow: Generate → Loads to RAM + saves to disk → You review → Close monitor → RAM cleared, disk remains → Open monitor later → Reloads from disk to RAM

We use an advanced content detection system that prevents a Generation about to enter the node´s inputs to be writen into cache if it already exists there.

The Clear Cache System

Click **ClearCache** button to open options:

"Clear All Cache" (Current Option) - Deletes all generation folders from disk - Clears RAM (FUTURE VERSION - NOT READY) - Empties the Generations dropdown - Frees up disk space

Coming in v5.3: Advanced Cache Management Future versions will offer: - Clear RAM Only (keep files on disk, free memory) - RAM usage display (see how much memory you're using) - Automatic management (smart cleanup when RAM is full - selected by user)

For Now: Manage your generations manually—delete old ones you don't need via the Generations dropdown.

Snapshots: Saving Your Workflow

What Is Snapshot: Captures your current ComfyUI workflow and saves it alongside your generation in a WebP file/image. Think of it as version control for your prompts and settings. If node it set to "smart" in "snapshot_path", snaps go to _nodes If node it set to "your custom path" in "snapshot_path", snaps go to that path Type "smart" again and the node will automatically know you´re sending to _nodes

How It Works: 1. Enabled by default (`snapshot_workflow: true` in node) 2. Automatically saves workflow in temp directory when generation completes 3. Stored in `snapshots` folder

Snapshot Button: Opens a menu to manage saved workflows

EXIT the Preview Video Monitor Window

`Shift+Q`

Closes monitor window and exit the tool (and it´s Windows/Pygame process). Not technically necessary to set `power_state` to Off unless you want to continue doing stuff in ComfyUI running promps and not want them to trigger PVM to re-open Also you need `power_state` before you change to a different physical display.

Advanced Notes: For Technical Directors, Pipeline Managers, Developers and Coders in general.

Pipeline Integration

Node Parameters: | Parameter | Description | |————-|————-| | `source` | "video" or "images" | | `monitor` | "Monitor 0", "Monitor 1", etc. (auto-detects displays) | | `power_state` | Boolean (true = on, false = off) | | `target_resolution` | "1920x1080" or "3840x2160" (UI resolution) | | `generations_name` | String (prefix for generation folders) | | `snapshot_workflow` | Boolean (auto-save workflows) | | `snapshot_path` | "smart" (auto) or custom directory path |

Cache Architecture

Storage Location:

`ComfyUI/custom_nodes/PreviewVideoMonitorPro/runs_cache/`

Generation Folder Structure:

```
runs_cache/
├── v_Generation_14-23-45_a1b2c3d4/  (video generation)
│   ├── frame_0000.jpg
│   ├── frame_0001.jpg
│   └── ...
├── i_Generation_14-25-12_e5f6g7h8/  (image generation)
│   ├── frame_0000.jpg
│   └── ...
└── snapshots/
    ├── workflow_a1b2c3d4.webp
    └── ...
```

Naming Convention:

`{prefix}_{generations_name}_{timestamp}_{uuid}`

- **Prefix:** `v_` (video) or `i_` (images)
- **Timestamp:** HH-MM-SS (filesystem safe)
- **UUID:** 8-char unique identifier

Metadata System

File: `generations_metadata.json`

Structure:

```json
{
  "gen_id_uuid": {
    "folder_path": "path/to/generation",
    "display_name": "v_Generation_14-23-45",
    "timestamp": 1735382625.123,
    "runs_name": "Generation",
    "source_type": "video",
    "total_size": 123456789,
    "frame_count": 1000,
    "format": "jpg"
  }
}
```

Duplicate Detection

Mechanism: Before registering a new generation, checks for existing entries with: - Identical `total_size` (folder size in bytes) - Identical `frame_count`

Behavior: If duplicate found: 1. Deletes newly created folder 2. Loads existing generation 3. Logs: "Duplicate content detected"

Edge Case: If cache is completely empty (`len(generations_metadata) == 0`), duplicate check is skipped. This ensures re-generation works after full cache clear.

RAM Management (Current Implementation)

Frame Storage:

```
self.cached_original_frames = [numpy_array, ...]  # All frames
self.cached_comparison_frames = [numpy_array, ...]  # Comparison
frames
```

Memory Calculation:

```
Per frame: width × height × 3 (RGB) bytes
Example: 1920 × 1080 × 3 = 6,220,800 bytes ≈ 6MB
1000 frames: 6MB × 1000 = 6GB RAM
```

Caching Strategy (v5.2): All frames loaded into RAM on generation load. No lazy loading or LRU eviction.

Planned Improvements (v5.3+): - Lazy loading (cache ±50 frames around current position) - LRU eviction (oldest frames dropped when memory limit reached) - Manual RAM clear (free memory without deleting disk files) - Compressed RAM cache (store JPEGs in memory, decode on-the-fly)

GPU Acceleration

OpenCV CUDA Support: If `opencv-contrib-python` with CUDA is installed: - Frame resizing uses `cv2.cuda.resize()` - Significant performance boost for 6K/8K video - Fallback to CPU if CUDA fails

Detection:

```
CV2_CUDA_AVAILABLE = cv2.cuda.getCudaEnabledDeviceCount() > 0
```

Log Message:

```
GPU acceleration enabled: cv2_cuda=True, torch_cuda=True
```

Performance Impact: | Resolution | Improvement | |————|————- | | 1080p | Negligible (PCIe overhead > benefit) | | 4K | ~10-20% faster | | 6K/8K | ~50-70% faster |

Known Performance Considerations

Zoom Slowdown (v5.2): At high zoom levels (5x+), rendering slows due to pygame blitting large surfaces (25-100x more pixels).

Root Cause:

```
# Current implementation
zoomed_surface = scale(frame, huge_size)  # e.g., 9600×5400
```

```
        screen.blit(zoomed_surface, position)    # Processes all pixels,
even clipped ones
```

Planned Fix (v5.3): Crop visible portion BEFORE scaling:

```
    visible_crop = crop(frame, visible_area)  # Only what's on screen
    zoomed = scale(visible_crop, screen_size) # Always ~2M pixels
    screen.blit(zoomed, (0, 0))               # Fast regardless of zoom
level
```

Threading Model

Main Thread: Pygame event loop, rendering, UI

Worker Thread: Video decoding, frame caching (started on new generation)

Synchronization: - `cache_lock`: Protects frame cache access - `new_content_ready`: Event signals cache completion - `running` flag: Controls thread lifecycle

---

Extension Points

Custom Vision Filters: Add to `_apply_vision_to_frame()` function. Input: RGB numpy array, Output: modified RGB array.

Custom Comparison Modes: Follow wipe/SBS pattern in `_draw_wipe_overlay()` and `_draw_sbs_overlay()`.

Custom Metadata: Extend `generations_metadata` structure in `_register_new_generation()`.

Environment Variables

Not currently used but could be added for: - `PVMP_CACHE_DIR`: Override default cache location - `PVMP_MAX_RAM_GB`: Set RAM limit for auto-cleanup - `PVMP_JPEG_QUALITY`: Adjust disk cache quality

---

Troubleshooting

Monitor window doesn't open: - Check `power_state` is set to ON (true) - Verify all dependencies are installed - Check ComfyUI console for error messages

Performance issues: - Close other applications using RAM - Reduce number of cached generations - For 6K/8K, install `opencv-contrib-python` with CUDA

Can't see comparison video: - Ensure comparison generation has same frame count - Check that both videos loaded successfully - Try exiting comparison (press W) and re-entering

Keyboard shortcuts not working: - Click the monitor window to give it focus - Check if you're editing text (shortcuts disabled during text entry)

Thank you for using PreviewVideoMonitorPro!

We're committed to making video review in ComfyUI fast, intuitive, and professional. Happy creating! 🎬✨

---

end of line