Audion VS Engine — User Guide

Русский · About the program · Console command · Presets · Every field

Contents

- Start
- Additional Components
- The Window
- Paths
- The Form of a Preset
- The Preview
- Running
- Profiles
- SERVICE
- SETTINGS and ABOUT
- The Command Line Under the Journal
- Menus on FZF
- When Something Goes Wrong

Start

`Start.exe` in the program's folder opens the window. The folder can lie on any drive and be copied to another machine: everything the program needs is inside it, nothing is installed into Windows. A second `Start.exe` brings the open window forward instead of opening another.

Additional Components

The processing engine does not ship with the program: VapourSynth, its plugins, FFmpeg and, with an NVIDIA RTX card, MLRT each have a licence of their own and are installed separately, from their

authors, into the program's folder — `system_core\vapoursynth\` and `Tools\ffmpeg\`; nothing is written into Windows.

- **At every start**, while one of them is not installed, the window asks: **Install additional components** — each component with what it is for and what will be downloaded (the files, their size), and the total. **INSTALL** installs them one after another — VapourSynth, the plugins, FFmpeg, MLRT; the terminal on the right shows the progress. **Install later** ticked says where to find them: the button **SETUP** in the header; **UNDERSTOOD** closes the question. **OPEN SETUP** closes the question and opens the sheet SETUP at once.
- **SETUP** in the header, apart from the buttons of the window itself, opens the sheet of the components: each with its version, its state and **INSTALL** or **UPDATE**; **INSTALL ALL** installs every component not yet here, one after another; **CHECK NOW** asks GitHub about a new VapourSynth (by itself at most once a day). A red dot on SETUP says that a component is not installed. The same sheet shows the machine — processor, video card, memory, the encoders of FFmpeg, the disk of the program — with **COPY** to send them with a report of a problem, and **Remove VapourSynth**: the engine with its plugins and MLRT to the Recycle Bin, asking first.
- Until the engine is installed, the page of a preset shows the installers in place of its form, in the same order.

MLRT. When the program finds an NVIDIA RTX card, the neural presets — DPIR, Real-ESRGAN 2x, Soft HD rebuild 2x — stand in RESTORATION, and MLRT is among the components (about 3.5 GB, trimmed to the card); until it is installed their page shows its card with the plan of the download. Without an RTX card these presets are not in the window. The first run of each model builds its TensorRT engine for 30–120 s; later runs start at once.

The Window

- **The header** — the name and the version; the badges of the engine: VapourSynth, the plugins (how many of the set the engine loads), MLRT with an RTX card, FFmpeg, the driver of the video card. A green dot — fresh; orange — a newer release is out; the tooltip says which. On the right: **SERVICE** and **SETUP**, then the buttons of the window itself — the language RU / EN, the theme, **SETTINGS**, **ABOUT**.
- **The palettes** — PRECISION, FILM LOOKS, RETRO, RESTORATION, PROFILES — large buttons with a line of what each is for.
- **The presets of the palette** — in groups by the kind of processing: a caption and its switches. PRECISION: Denoise · Deband · Master; FILM LOOKS: Film stock · Look; RETRO: Video · Photo;

RESTORATION: Interlace · Noise · Artefacts · Scale · Motion. A switch carries a short name; the full one and what the preset does are in its tooltip and over the form.

- **On the left** — the page of the preset: its name, **SAVE A PROFILE**, **COMMAND**, **RUN**, back to the defaults; under them a line of what the preset does, the form, and at the bottom the preview.
- **On the right** — the paths (SOURCE, TARGET and their buttons), the journal, the command line.
- **At the bottom** — the status line, the progress bar, the clock and **STOP** while a run goes.

The divider between the left and the right is dragged; the width is remembered. **The wide journal** button gives the journal the whole window and back.

Paths

- **SOURCE** — what is processed: a folder (every video in it and in its subfolders) or one file. A path is typed, picked (folder, **Add a file...**), taken from the history, or dropped onto the window. An empty field is the folder `input\` of the program.
- **TARGET** — where results go; empty — `output\`.
- Beside each: pick, history (pinned paths first), pin, open in Explorer.
- The window opens on `input\` and `output\`; the path of the last time comes back only when it is pinned.
- **RESET** — both paths back to `input\` and `output\`; the history is cleared, pinned paths stay; no file is touched.
- **LIST** — the videos of the Source a run takes, with their folders, in the journal.
- **DETAILS** — what ffprobe knows of every video of the Source: size, frame rate, length, codec, range, colour.
- **TO SOURCE** — the result becomes the Source of the next operation: the Source is emptied into the Recycle Bin, then the contents of the Target are moved into it. The window asks first and names both folders.
- **CLEAR** — the contents of the Source, of the Target or of both to the Recycle Bin; the folders themselves stay. Asked first.

The Form of a Preset

- **PARAMETERS** — the fields of the preset. A number is a slider with its value beside it: drag the slider, or type the value and press Enter; the arrow keys step it. A choice is a row of switches. The tooltip of each field says what it does and its range.

- **PROCESSING** — where the heavy part runs: **Auto**, **CPU**, **GPU**. It stands only on the presets where it changes something: BM3D (Shadow Denoise SOTA, Filmic Rebuild — Vulkan on a video card of any maker, else the processor) and the neural presets (TensorRT on RTX, DirectML on another card, or the processor).
- **ENCODING** — four cards: x264, x265, ProRes, DNxHR. The chosen card is lit by its rim; inside it the encoder (CPU, NVIDIA, QuickSync, AMF), the quality 14 / 17 / 21, the container, or the profile and the wrapper of ProRes, the profile of DNxHR. A hardware encoder this machine has not passed is greyed with the reason in its tooltip.
- **RUN** — **Overwrite**: make a result again even when it is already there.

What a page holds is remembered for the next start; **back to the defaults** returns it. PROCESSING and the cards of the encoder are one for the whole window: set on one page, they are set on all.

The Preview

Under the form: the middle frame of a video of the Source, before and after the preset with the values on the form.

- **Before** — the frame of the source as the presets load it; **After** — with the preset; **Side by side** — both.
- **‹ 1 / 5 ›** — the videos of the Source in turn.
- It is counted by itself about a second after the last change of the form, of the page or of the Source; a count that is no longer wanted is ended. **Again** counts it now; **fold** gives the form all the room.
- Under the controls: the size and frame rate of the source, which frame it is; the size of the result when the preset changes it (2x presets); for a denoiser — the noise it measured and the strength it chose.
- When the form is taller than the room above the preview, the form scrolls and the preview stays.

Running

1. Choose the palette and the preset, set the form; look at the preview.
2. **COMMAND** — the commands of vspipe and FFmpeg the run would start, for every file of the Source, in the journal; nothing is started.
3. **RUN** — the preset goes over every video of the Source. The first line of the journal is the same run as a command of `vs_cli.cmd`. Under the journal a live line: the share, the frames done and their

number, the speed, the time left, the file; the bar at the bottom shows the whole run.

4. **STOP** — vspipe and FFmpeg end, the half-written file is removed, the finished ones stay. A second press a few seconds later ends the run at once. Closing the window during a run asks first, then stops it the same way.

Where the result goes: `<Target>\<Palette>\<preset>_<encoder>\<the folders of the Source>\<name>.<ext>`, the name of the source kept — for example `output\Precision\mild_denoise_h264_crf14\clip.mp4`. A result already there is passed over (`[SKIP]` in the journal) unless **Overwrite** is on, so a stopped run simply starts again. The journal of the run is in `logs\`, its report in `report\` — **LOGS** and **REPORT** over the journal open them.

A run is refused before it begins when it cannot begin — no engine, no video in the Source, a neural preset without MLRT; the reason is in the journal, and nothing is written.

Profiles

A profile is a preset with its values and an encoder, saved under a name.

- **SAVE A PROFILE** on the page of a preset asks a name (Latin letters, digits, `-`, `_`) and saves the preset with the values of the page and the encoder of the cards into `config\profiles\<name>.json`.
- **PROFILES → Run a profile**: choose the profile — its encoder goes onto the cards — and **RUN**. The results go to `<Target>\Profiles\<profile>_<encoder>\`.

Five profiles come with the program: `shadow_clean_quick`, `filmic_warm_35mm`, `archival_master`, `resolve_handoff`, `vhs_dreamy`.

SERVICE

Item	What it does
Doctor	checks the whole engine: both Pythons, VapourSynth, every plugin, FFmpeg, BM3D on the video card
File list	the videos of the Source the next run takes
Hardware capabilities	short encodes with NVENC, QuickSync and AMF: which hardware encoders work on this machine
Check every preset	every preset on a test clip, a frame each: what runs here and how fast

Item	What it does
BM3D: video card or CPU	BM3D through Vulkan against BM3D on the processor on the first video of the Source

The components are installed in **SETUP** (above).

SETTINGS and ABOUT

- **SETTINGS** — the look of the window: the theme (Audion Code Dark, Audion Vault Dark, Audion Dark, Audion Light), the fonts and their sizes (**TO DEFAULTS** returns them).
- **ABOUT** — the version, the folders of the program, what it is made of and under which licences.

The Command Line Under the Journal

PS — the PowerShell 7 of the program (`App\AudionPs.exe`), **CMD** — cmd.exe. Enter runs the command; its lines come into the journal, STOP ends it. The history keeps the commands, pinned first; the file and folder buttons add a path.

Menus on FZF

The same work without the window, in a console with a quick picker:

- `launcher_project.cmd` / `launcher_project_ru.cmd` — the palettes, the profiles, the tools;
- `cli\launcher_<palette>.cmd` / `_ru.cmd` — the presets of a palette: the questions of the preset, then `vs_cli.cmd run`;
- `launcher_tools.cmd` — the checks;
- `builder_main.cmd` — the installers of the engine, MLRT, the cleaning of the install cache, the doctor.

When Something Goes Wrong

- **The page shows installers instead of the form** — the engine is missing: install it from the top down.
- **A neural preset is not in RESTORATION** — the program found no NVIDIA RTX card and MLRT is not installed.

- **"refused before it began"** — the journal names the reason in a `[PREFLIGHT] FAIL` line: no video in the Source, no engine, no MLRT, an inverse telecine on a source under 29 fps, camera RAW.
- **Camera RAW** (ProRes RAW, ARRIRAW, Sony RAW and X-OCN, RED R3D, Nikon N-RAW, Blackmagic RAW, Canon Cinema RAW) — the engine does not read it: VapourSynth sees no picture in it. The line names the program that opens it and its page: convert it there into ProRes 4444 or DNxHR 444 and run the preset on that. A Source of RAW alone is refused before the run; among other videos such files fail one by one with that line, the rest are made. The window names such files once, in a window of its own, when the Source is chosen, and keeps them out of the preview. The table of the programs is in the README.
- **An encoder is greyed** — this machine has not passed it: SERVICE → Hardware capabilities checks again.
- **The preview says it could not count** — the reason follows; the same reason stops a run, so it is worth reading.
- **The folder was moved** — the next start of the program repairs the launchers of its Pythons itself.
- **Doctor** in SERVICE tells the state of the whole stack.