Audion VS Engine

English · Руководство · Консольная команда · Пресеты · Решения

Содержание

- Зачем это сделано
- Решает картинка
- Четыре палитры
- Пресеты
- Нейросетевые пресеты
- Принципы
- Что внутри
- Дальше
- Техническая справка
  - Перед первым запуском
  - FFmpeg и драйвер NVIDIA
  - RAW камер
  - Папки
  - Кодирование
  - Профили
  - Требования
  - Лицензии

Портативная программа для работы с картинкой видео на VapourSynth и FFmpeg: шумодав, дебанд и зерно, плёночные и ретро-образы, реставрация старого и пережатого материала, а на видеокарте NVIDIA RTX — нейросетевое увеличение, шумодав и дорисовка кадров.

Зачем это сделано

VapourSynth умеет то, чего нет ни в одном монтажном пакете из коробки: деинтерлейсер эталонного качества, шумодав, который работает только в тенях, обратный телесин, шумодав с компенсацией движения, увеличение нейросетью. Но пользоваться им — значит писать скрипт на Python, помнить имена плагинов и собирать конвейер руками для каждого ролика.

Программа держит эту работу в пресетах и даёт ей окно: палитры, пресеты по группам вида обработки, форма каждого пресета, запомненные пути, терминал с ходом каждого кадра и видимая команда. Каждая кнопка окна — операция консольной команды `vs_cli.cmd`, поэтому та же работа идёт из пакетного файла или через ИИ-агента, которого попросили обычными словами.

Решает картинка

Пресет выбирают глазами. Под формой каждого пресета стоит его превью: кадр из середины видео Источника до и после пресета со значениями формы — рядом или по одному. Он считается сам через мгновение после последнего изменения, тем же скриптом и с теми же значениями, что и прогон; ставший ненужным расчёт сразу гасится. Рядом — сведения о кадре: размер, частота, номер кадра, а у шумодава — шум, измеренный в исходнике, и что пресет из него сделал.

Четыре палитры

Они независимы: только техническая обработка, только образ, или одно за другим.

Палитра	Пресетов	Что делает
ТОЧНОСТЬ	8	шумодав — в том числе только по теням, — дебанд, контролируемый возврат микрозерна. Технический слой **до** колориста
ПЛЁНКА	7	плёнки: Kodak 35 мм, 16 мм, Super 8, IMAX 70 мм; образы: кинематографичный, отбеливание, блики анаморфота
РЕТРО	3	аналоговый характер: VHS и кинескоп, камкордер 90-х, полароид
РЕСТАВРАЦИЯ	10	то, чего нет в монтажных пакетах: деинтерлейс, обратный телесин, шумодав с компенсацией движения, блоки, радуга, дымка, увеличение; на vs-mlrt — DPIR, Real-ESRGAN, пересборка мягкого HD

Порядок важен. ТОЧНОСТЬ — технический слой, который идёт *до* цветокоррекции: чинить шум после того, как колорист вытянул тени, поздно.

Пресеты

Каждый пресет — там, где он стоит в окне, — и вызов консольной команды, который делает то же:

Раздел	Группа	Пресет	Вызов
ТОЧНОСТЬ	Шумодав	Мягкий шумодав	`vs_cli.cmd run --operation mild_denoise`
ТОЧНОСТЬ	Шумодав	Шумодав теней SOTA	`vs_cli.cmd run --operation shadow_denoise_sota`
ТОЧНОСТЬ	Шумодав	Чистка цветового шума	`vs_cli.cmd run --operation chroma_cleanup`
ТОЧНОСТЬ	Дебанд	Мягкий дебанд	`vs_cli.cmd run --operation deband_safe`
ТОЧНОСТЬ	Дебанд	Дебанд с тонким зерном	`vs_cli.cmd run --operation deband_fine_grain`
ТОЧНОСТЬ	Мастер	Плёночная пересборка	`vs_cli.cmd run --operation filmic_rebuild`
ТОЧНОСТЬ	Мастер	Чистый архив	`vs_cli.cmd run --operation archive_clean`
ТОЧНОСТЬ	Мастер	Подготовка к грейдингу	`vs_cli.cmd run --operation pregrade_prep`
ПЛЁНКА	Плёнка	Плёнка 35 мм	`vs_cli.cmd run --operation film_35mm`
ПЛЁНКА	Плёнка	Плёнка 16 мм	`vs_cli.cmd run --operation film_16mm`
ПЛЁНКА	Плёнка	Super 8	`vs_cli.cmd run --operation super8`

Раздел	Группа	Пресет	Вызов
ПЛЁНКА	Плёнка	IMAX 70 мм	`vs_cli.cmd run --operation imax_70mm`
ПЛЁНКА	Образ	Кинематографичный	`vs_cli.cmd run --operation cinematic`
ПЛЁНКА	Образ	Отбеливание	`vs_cli.cmd run --operation bleach_bypass`
ПЛЁНКА	Образ	Анаморфот	`vs_cli.cmd run --operation anamorphic_scope`
РЕТРО	Видео	VHS и кинескоп	`vs_cli.cmd run --operation vhs_crt`
РЕТРО	Видео	Камкордер 90-х	`vs_cli.cmd run --operation camcorder_90s`
РЕТРО	Фото	Полароид	`vs_cli.cmd run --operation polaroid`
РЕСТАВРАЦИЯ	Чересстрочность	Деинтерлейс QTGMC	`vs_cli.cmd run --operation qtgmc_deinterlace`
РЕСТАВРАЦИЯ	Чересстрочность	Обратный телесин TIVTC	`vs_cli.cmd run --operation tivtc_ivtc`
РЕСТАВРАЦИЯ	Шум	MVTools MCDegrain	`vs_cli.cmd run --operation mvtools_mcdegrain`
РЕСТАВРАЦИЯ	Шум	Шумодав DPIR [1]	`vs_cli.cmd run --operation dpir_denoise`
РЕСТАВРАЦИЯ	Артефакты	Блоки H.264	`vs_cli.cmd run --operation deblock_h264_artefacts`
РЕСТАВРАЦИЯ	Артефакты	Радуга и перекрёстные помехи	`vs_cli.cmd run --operation derainbow_decross`
РЕСТАВРАЦИЯ	Артефакты	Дымка и локальный контраст	`vs_cli.cmd run --operation dehaze_local_contrast`

Раздел	Группа	Пресет	Вызов
РЕСТАВРАЦИЯ	Масштаб	Увеличение NNEDI3 2X	`vs_cli.cmd run --operation nnedi3_upscale_2x`
РЕСТАВРАЦИЯ	Масштаб	Real-ESRGAN 2x [1]	`vs_cli.cmd run --operation vsmlrt_realesrgan_2x`
РЕСТАВРАЦИЯ	Масштаб	Пересборка мягкого HD 2x [1]	`vs_cli.cmd run --operation soft_hd_rebuild_2x`
ПРОФИЛИ		Запустить профиль	`vs_cli.cmd run --operation profile_run`

[1] на vs-mlrt: виден при видеокарте NVIDIA RTX или MLRT.

Словами — что делает, на каком материале, какие у него параметры — в справочнике пресетов; каждое поле с диапазоном и вариантами — в OPERATIONS_RU.md.

Нейросетевые пресеты

DPIR, Real-ESRGAN 2x и пересборка мягкого HD 2x работают на vs-mlrt: TensorRT на NVIDIA, ONNX Runtime, модели. Видеокарту NVIDIA RTX программа ищет сама; если она есть, MLRT предлагается среди дополнительных компонентов — около 3,5 ГБ закачки под эту карту, всё в папку программы. Без карты RTX и без MLRT этих пресетов в окне нет. Где работает сеть, задаёт блок ОБРАБОТКА её страницы: TensorRT на RTX, DirectML на другой видеокарте или процессор. Первый прогон каждой модели собирает её движок TensorRT 30–120 с, дальше прогоны начинаются сразу.

Принципы

- **Исходник не меняется.** Результат ложится в Назначение — в папку палитры и пресета с кодером, с папками и именем исходника.
- **Файл либо целый, либо его нет.** Результат пишется под своим именем и получает настоящее, только когда дописан; СТОП гасит vspipe и FFmpeg и убирает недописанный файл. Готовый результат пропускается, так что остановленный прогон просто запускают снова.
- **Шумодавы меряют шум.** Семь из них измеряют шум исходника до начала и усиливаются вместе с ним; ниже своих настроек они не опускаются, так что чистый материал получает ровно настройки.

- **Уровни сохраняются.** Полнодиапазонный исходник остаётся полнодиапазонным — через фильтры и в кодировщике.
- **Один файл — одни и те же кадры.** У зерна постоянное начальное число; BM3D на процессоре повторяет себя байт в байт.
- **Вне папки ничего не пишется.** Движок, плагины, MLRT, индексы исходников и кэши установщиков живут в ней; в Windows, реестр и PATH ничего не попадает.

Что внутри

- **Окно** — `Start.exe` (`App\AudionVSEngine.exe`, C# на Avalonia): палитры, группы пресетов, форма, превью, пути, терминал, СЕРВИС, SETUP, настройки вида.
- **Консольная команда** — `vs_cli.cmd` на собственном Python программы: каждая операция, итог в JSON, коды выхода, кадр до и после в PNG.
- **Пресеты** — скрипты `.vpy` в `system_core\presets\<палитра>\`, у каждого своё описание.
- **Движок** — VapourSynth со своим Python и плагинами, его ставит программа; FFmpeg — в `Tools\ffmpeg\bin\`.
- **Меню на FZF** — `launcher_project_ru.cmd`, `launcher_tools.cmd`, палитры в `cli\`, сборщик `builder_main.cmd`.
- **Профили** — пресет со значениями и кодером, сохранённый под именем.

Дальше

- Руководство — установка движка, окно, превью, профили, меню.
- Консольная команда — `vs_cli.cmd`: команды, ключи, коды, итог.
- Пресеты — каждый пресет словами, какой для какого материала.
- Решения — почему так устроено.
- Изменения.

Техническая справка

Перед первым запуском

Программа поставляется без VapourSynth и его плагинов — это около семидесяти модулей, у каждого своя лицензия. Она ставит их сама, от их авторов. Пока движка нет, страница пресета

показывает установщики по порядку, у каждого — что он скачает; при запуске окно предлагает поставить их все или открыть SETUP одной кнопкой, они же есть в SETUP в шапке и в сборщике `builder_main.cmd`:

№	Пункт меню	Что делает
10	`VAPOURSYNTH`	самый новый стабильный VapourSynth со своим Python 3.12
11	`VS PLUGINS`	фильтры, которые зовут пресеты, затем проверка, что движок загружает каждый
12 / 13	`FFMPEG …`	портативный FFmpeg, подобранный под драйвер NVIDIA
14	`REMOVE VAPOURSYNTH`	убирает движок — с плагинами и MLRT — в Корзину; 10 и 11 возвращают его
71	`VERIFY / DOCTOR`	проверяет весь стек

VapourSynth R80 и новее загружает только плагины под свой нынешний интерфейс (API 4); шаг 11 ставит именно такие и не трогает ядро, которое поставил шаг 10.

vs-mlrt для нейросетевых пресетов — не шаг сборщика: окно предлагает его при первом запуске и в SETUP, только когда находит видеокарту NVIDIA RTX, и ставит под эту карту.

FFmpeg и драйвер NVIDIA

Каждая сборка FFmpeg собрана под свою версию заголовков аппаратного кодирования и требует свой наименьший драйвер. Самая новая сборка на старом драйвере ничего не ускоряет — она ломает аппаратный путь. Сборка подбирается под драйвер.

Драйвер NVIDIA	Какую FFmpeg ставит программа
видеокарты NVIDIA нет	самую новую стабильную сборку Gyan, не ниже 9.0.2
610.88 (LTS) и новее	самую новую стабильную сборку Gyan, не ниже 9.0.2
570.0 – 610.87	8.0.1 (заголовки NVENC n13.0.19.0) — доступны не все операции
551.76 – 569.xx	7.1 (заголовки NVENC n12.2.72.0) — доступны не все операции
ниже 551.76	никакую: установщик останавливается и говорит почему

ProRes RAW нужна 9.x. В мае 2026 года в FFmpeg исправили декодер ProRes RAW: теперь он читает записанную в файле кривую линеаризации, и искажения тонов и цвета на картинке уходят. Исправление вошло в 9.0; 8.0.1 и раньше декодируют ProRes RAW с этим искажением. Ещё 9.x обрезает кадр до активной области сенсора и передаёт параметры цвета камеры с каждым кадром.

610.88 — LTS-драйвер, под который собраны программы. С ним или без видеокарты NVIDIA ставится самая новая стабильная FFmpeg — выйдет 9.1, будет 9.1 — и никогда не старше 9.0.2. Если самая новая сборка не прошла пробу NVENC на этом драйвере (новая FFmpeg может потребовать драйвер новее), установщик возвращается к 9.0.2. Ниже 610.88 ставится прежняя ветка, при которой работает NVENC, и установщик, план закачки и бейдж FFmpeg в шапке говорят, что доступны не все операции — те, которым нужна FFmpeg 9.x, например правильный ProRes RAW. Сборка подбирается при установке: сменился драйвер — FFmpeg, поставленная снова в SETUP, подберётся под него. 7.1.1 старому драйверу не поможет: она собрана с теми же заголовками, что 8.0.1, и требует тот же 570.0 — помогает только 7.1.

RAW камер

Движок не читает RAW камер: источники VapourSynth не видят в нём картинки. Источник только из RAW получает отказ до начала, среди других видео такой файл проваливается сам по себе, остальные делаются; превью говорит то же. Строка называет программу, которая его открывает, и её страницу: там его переводят в ProRes 4444 или DNxHR 444 и запускают пресет на результате. Программы ставит человек, с сайтов производителей и на их условиях: VS Engine их не скачивает и не несёт.

RAW	Файлы	Чем открыть
ProRes RAW, ProRes RAW HQ	`.mov`	DaVinci Resolve, Final Cut Pro
ARRIRAW, ARRIRAW HDE	`.mxf`, `.ari`, `.arx`	ARRI Reference Tool, DaVinci Resolve
RED R3D	`.r3d`	REDCINE-X PRO, DaVinci Resolve
Nikon N-RAW	`.nev`	DaVinci Resolve, REDCINE-X PRO
Blackmagic RAW	`.braw`	Blackmagic RAW Player, DaVinci Resolve
Sony RAW, X-OCN	`.mxf`	Sony RAW Viewer, DaVinci Resolve

RAW	Файлы	Чем открыть
Canon Cinema RAW, Cinema RAW Light	`.rmf`, `.crm`	Canon Cinema RAW Development, DaVinci Resolve

Папки

Папка	Что в ней
`input\`	Источник по умолчанию: все видео папки и её подпапок
`output\`	Назначение по умолчанию: `<Палитра>\<пресет>_<кодер>\<папки Источника>\<имя>.<расш>`; профиль — `Profiles\<профиль>_<кодер>\`
`logs\`	журнал каждого прогона
`report\`	отчёт каждого прогона (`run_report.json`, `.md`) и JSON каждого файла
`config\`	операции и их тексты (`tool_manifest.yaml`), раскладка окна, профили, настройки окна

Кодирование

34 кодера: x264 и x265 по лестнице качества 14 / 17 / 21, ProRes LT, 422, 422 HQ в MOV или MXF, DNxHR LB, SQ, HQ, HQX и та же лестница на NVENC, QuickSync и AMF. В окне это четыре карточки — x264, x265, ProRes, DNxHR, — внутри кодер, качество и контейнер. Аппаратный кодер, который эта машина не прошла (СЕРВИС → «Возможности оборудования»), выключен с причиной. Звук копируется; ProRes, идущий в x264 или x265, получает AAC 384 кбит/с, MXF — PCM 24 бит 48 кГц.

Профили

Имя	Пресет
`shadow_clean_quick`	Шумодав теней SOTA — быстрая чистка шума в тенях тёмного цифрового материала
`filmic_warm_35mm`	Плёнка 35 мм — Kodak 250D с интенсивностью 1,0
`archival_master`	Чистый архив — нейтральный архивный мастер в ProRes 422 HQ
`resolve_handoff`	Подготовка к грейдингу — минимум перед DaVinci Resolve, ProRes 422 HQ
`vhs_dreamy`	VHS и кинескоп с интенсивностью 1,2

«Сохранить профиль» на странице пресета сохраняет его со значениями страницы и кодером карточек в `config\profiles\<имя>.json`.

Требования

- Windows 10/11 x64; около 1,2 ГБ на диске вместе с движком; MLRT добавляет около 3,5 ГБ.
- Системные Python, VapourSynth и FFmpeg не нужны.
- Видеокарта с драйвером Vulkan не обязательна: на ней работает BM3D; без неё BM3D идёт на процессоре.
- Для установки нейросетевых пресетов нужна видеокарта NVIDIA RTX; установленные, они работают и на другой карте через DirectML или на процессоре — во много раз медленнее.

Лицензии

Движок — отдельные программы со своими лицензиями: программа их зовёт, а не встраивает. VapourSynth — под LGPL 2.1, каждый плагин — под своей (GPL, LGPL, MIT, ISC — `install\vs_plugins.json`), сборка FFmpeg — GPL v3, её исходники лежат рядом в `Tools\`, vs-mlrt и его среды — TensorRT, ONNX Runtime — под своими. В выпуск ничего из этого не входит: программа ставит всё от авторов.