

Thomas Wagensommerer

COMMUNAL TOOLS · OPEN SOURCE APPLICATIONS

OPEN SOURCE APPLICATIONS

Communal Tools

sediment	2026
sediment · live	2026
sediment · fx	2026
sediment · pod	2026
ligature	2026
sortcut	2026
scorecut	2026
datamosh station	2026
scanline	2026
framegrab	2026

AVjoin	2026
dubtrans	2026
subtrans	2026
voicebox	2026
TerraForge	2026
meanland	2026
GNM head pipeline	2026
AutoRagdoll	2026
Resonode	2026

SOUND · 2026

sediment

FIELD	Sound — corpus-based composition and live processing
VERSION	1.0
TECHNOLOGY	Python (numpy/scipy, aiohttp), browser UI, ffmpeg; Windows/macOS self-installer
SOURCE	sediment-1.0.zip

A composition environment that treats a folder of recordings as a corpus. Every file is analysed into frame-level descriptors (loudness, onset density, pitch and pitchedness, spectral shape, chroma and key, stereo width, an MFCC-derived timbre space); pieces of arbitrary length are then re-ordered according to explicit ordering principles — sorting, gliding, contrast, morphing, formal clustering, Markov grammars learned from the sources, serial rows, modal and harmonic retuning, audible sorting algorithms — and rendered as high-resolution audio with a full event log. A picture add-on cuts frame-exact video to the resulting sound. The tool makes the ordering itself the compositional material: structure becomes audible as process.

sound picture

01 SOURCE

/srv/tw/corpus browse scan

☑	l0.wav	30.0 s	A2	Gm	136
☑	l1.wav	36.0 s	A#1	C	133
☑	l2.wav	42.0 s	F2	F	129
☑	l3.wav	48.0 s	C3	F	148
☑	l4.wav	54.0 s	B2	Gm	126
☑	l5.wav	1:00.0 min	C#3	F	117

6/6 files · 4:30.0 min all invert random subset

02 BRACKET & LENGTH

piece min 40.0 ms
piece max 1.50 s
output length 1:00.0 min

type values: 12smp · 3ms · 0.4s · 2m · 1:30

03 SLICING

start snap free onset beat
pool size 3000
silence floor dB -55.0
end on onsets

04 ORDERING MODE

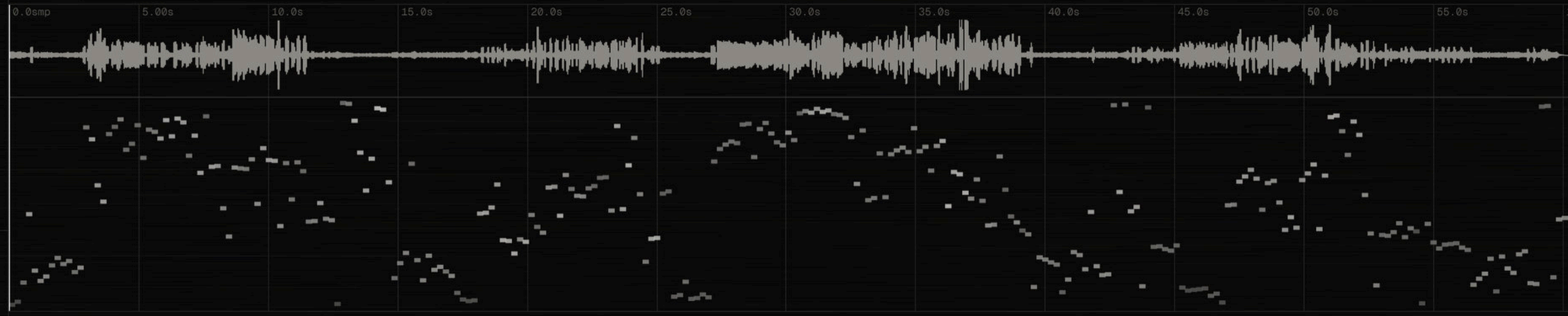
sort	follow	glide	contrast
morph	form	markov	serial
mode / scale	harmony	sorting process	drift
shortest path	spiral	mutating loop	scan
random			

0:00.000 / 1:00.208 loop

273 pieces · pool 3000 · 9 length bins · 48 kHz · peak -1.0 dBFS · 11.15 s

colour brightness height pitch

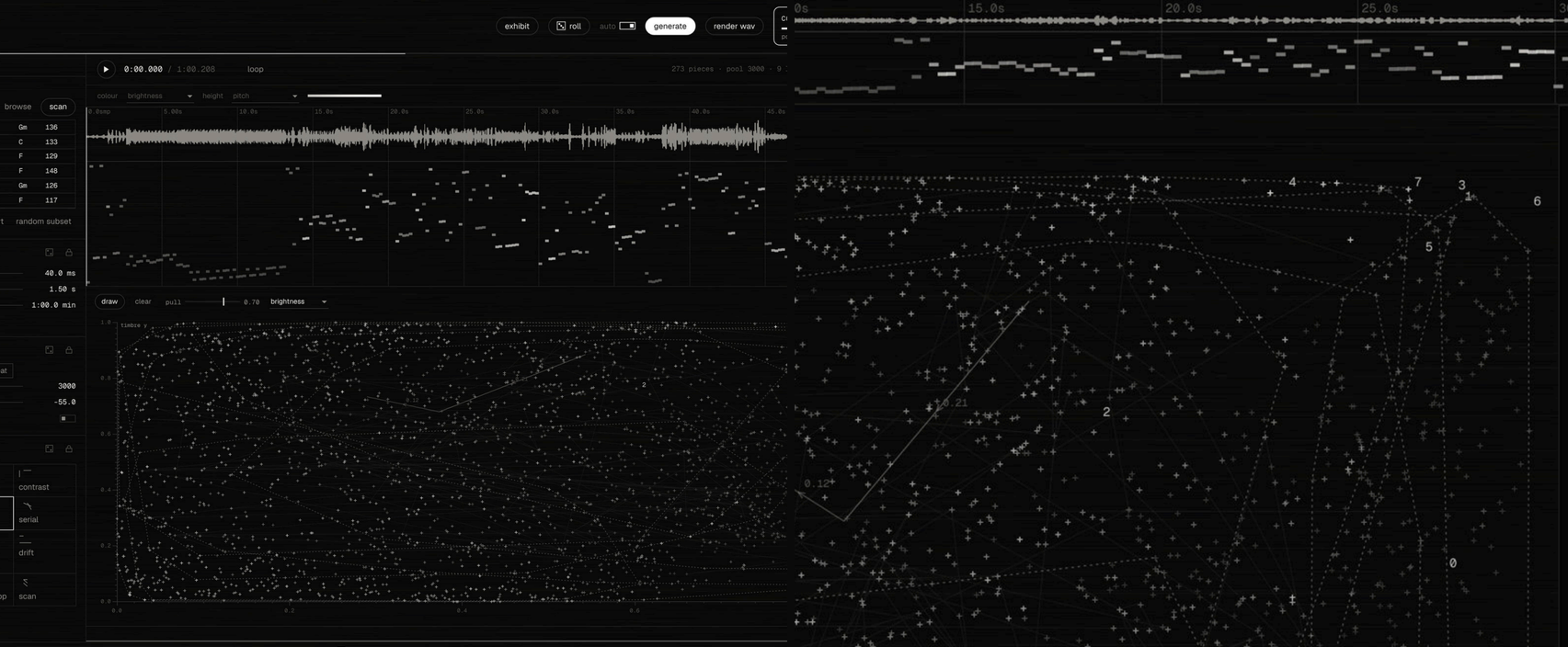
wheel zoom · drag pan · dbl-click reset · click piece = audition

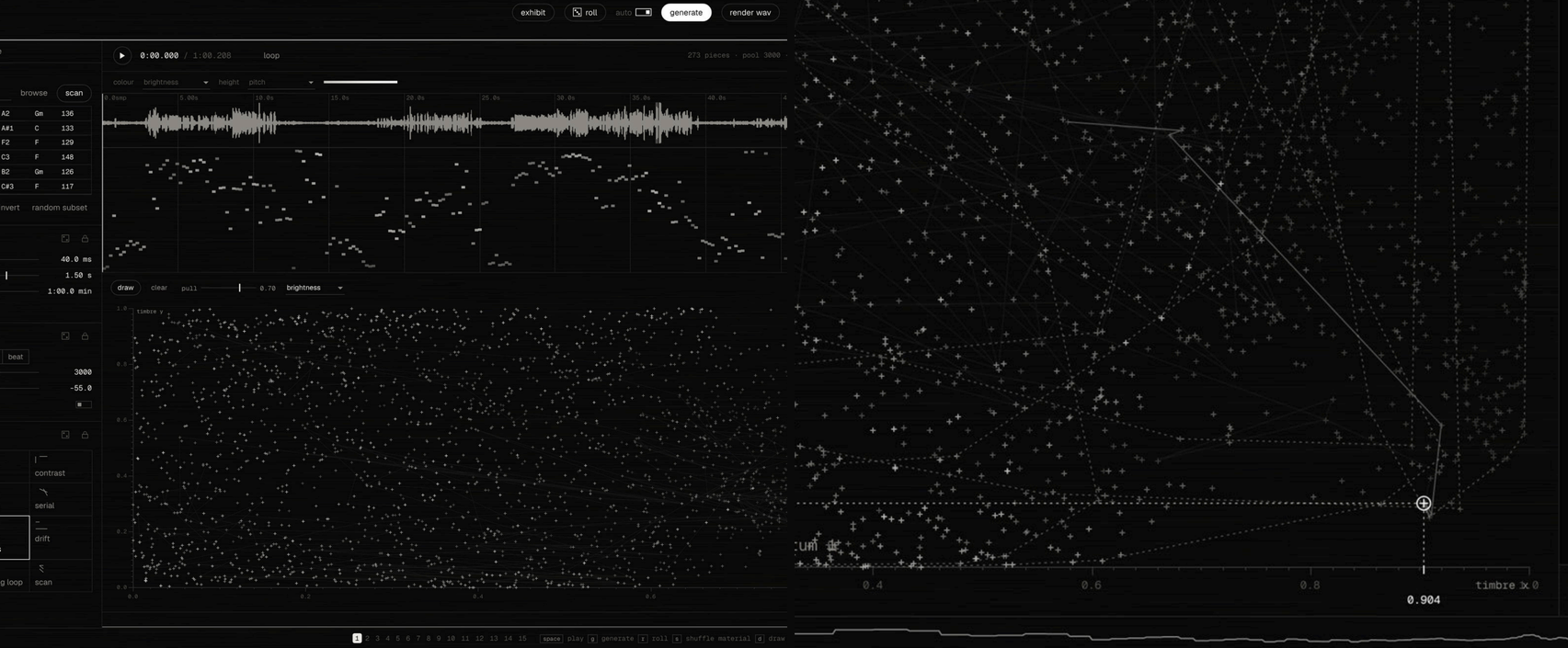


draw clear pull 0.70 brightness

glide · seed 1 · 3000 grains







picture addon

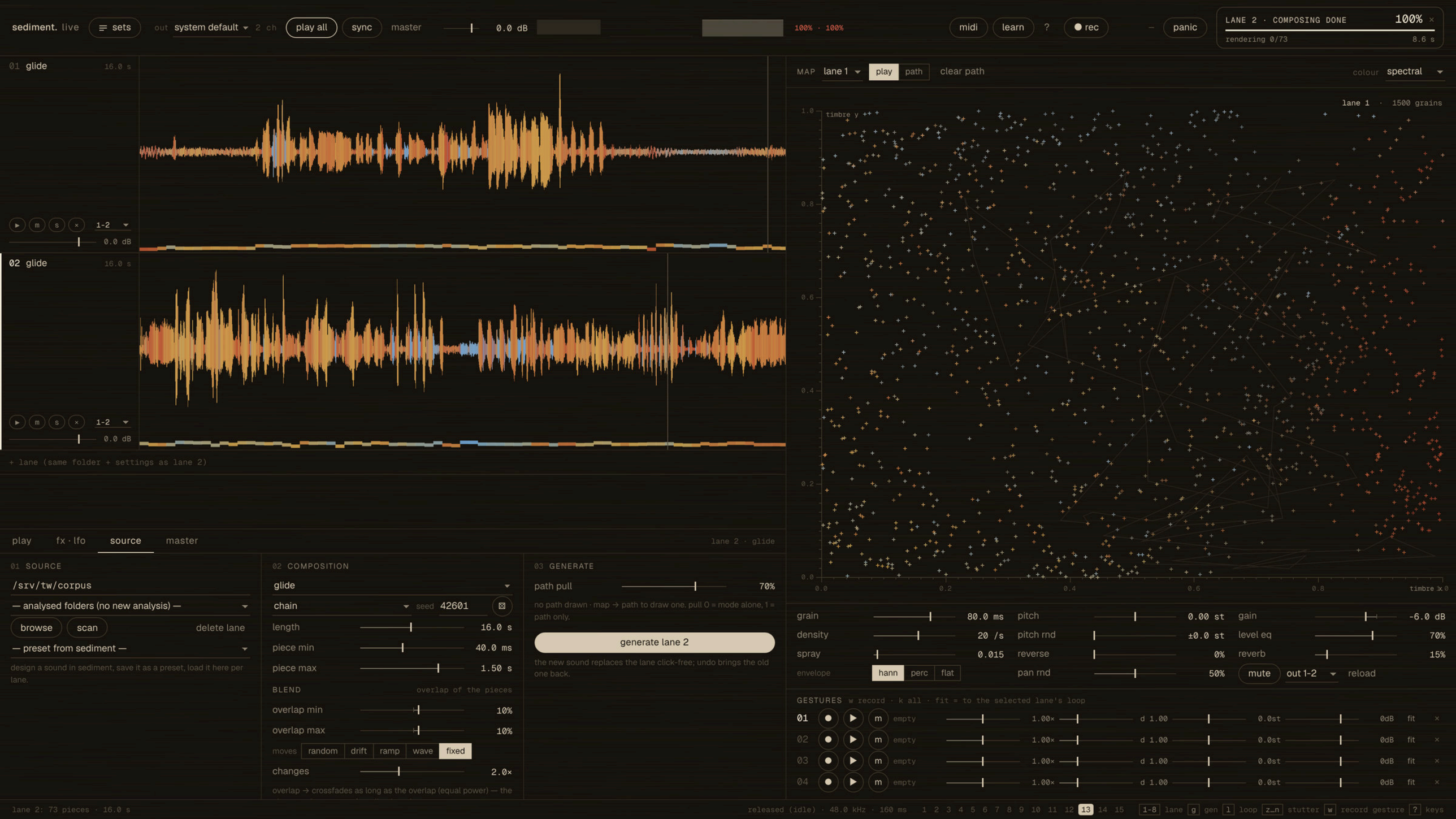
open **picture**, scan a folder of videos / frame sequences, and c

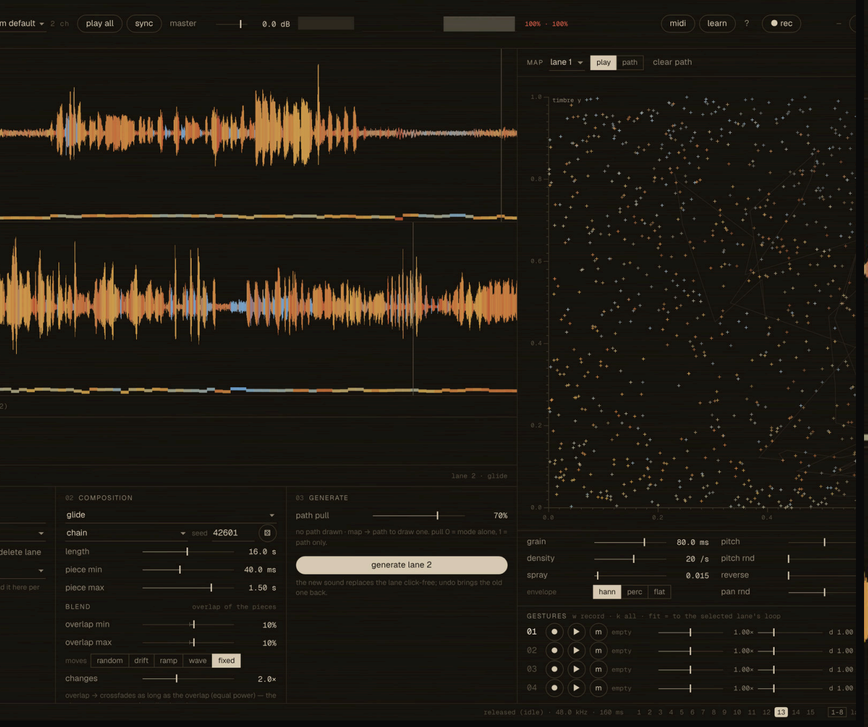
SOUND · 2026

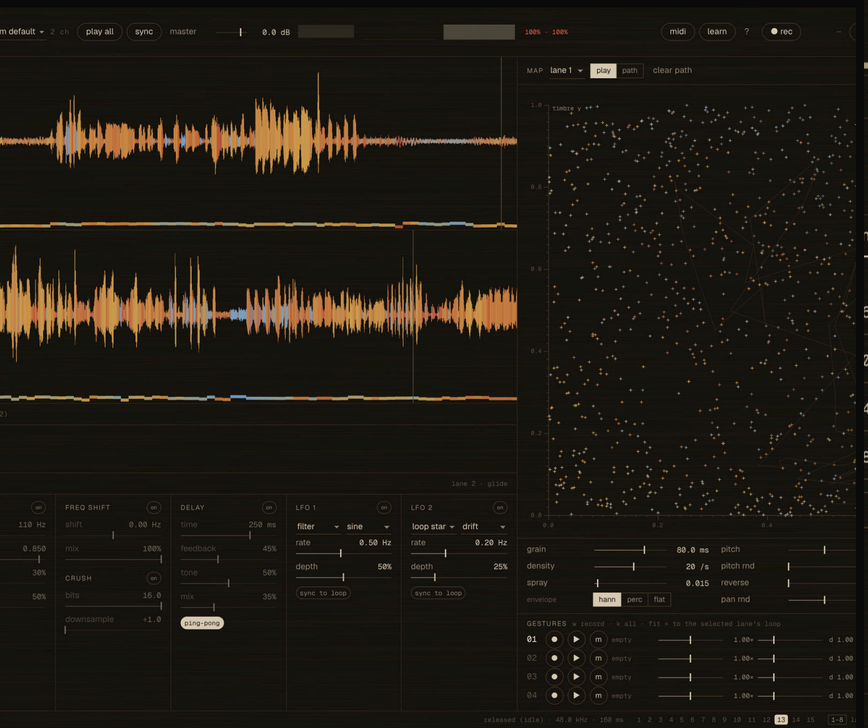
sediment · live

FIELD	Sound — corpus-based composition and live processing
VERSION	1.0
TECHNOLOGY	Python (aiohttp, sounddevice), Web Audio/browser UI, MIDI
SOURCE	sediment_live.zip

The performance branch of sediment. Up to eight lanes hold composed material that can be looped, stuttered, re-sorted and transformed in real time; the timbre map of each lane becomes a playable surface on which gestures are drawn, recorded and recalled. Sets store audio and state, MIDI learn binds every control. Designed to run alongside a DAW in concert.


sediment. live
sets
out system default 2 ch
play all
sync
master
0.0 dB
100% · 100%
midi
learn
?
rec
panic
LANE 2 · COMPOSING DONE
100%
rendering 0/73
8.6 s
01 glide
16.0 s
1-2
0.0 dB
02 glide
16.0 s
1-2
0.0 dB
+ lane (same folder + settings as lane 2)
MAP
lane 1
play
path
clear path
colour
spectral
lane 1 · 1500 grains
timbre y
timbre x
play
fx · lfo
source
master
lane 2 · glide
01 SOURCE
/srv/tw/corpus
— analysed folders (no new analysis) —
browse
scan
delete lane
— preset from sediment —
design a sound in sediment, save it as a preset, load it here per lane.
02 COMPOSITION
glide
chain
seed 42601
length 16.0 s
piece min 40.0 ms
piece max 1.50 s
BLEND
overlap of the pieces
overlap min 10%
overlap max 10%
moves random drift ramp wave fixed
changes 2.0×
overlap → crossfades as long as the overlap (equal power) — the
03 GENERATE
path pull 70%
no path drawn · map → path to draw one. pull 0 = mode alone, 1 = path only.
generate lane 2
the new sound replaces the lane click-free; undo brings the old one back.
grain 80.0 ms
density 20 /s
spray 0.015
envelope hann perc flat
pitch 0.00 st
pitch rnd ±0.0 st
reverse 0%
pan rnd 50%
gain -6.0 dB
level eq 70%
reverb 15%
mute
out 1-2
reload
GESTURES w record · k all · fit = to the selected lane's loop
01 m empty 1.00× d 1.00 0.0st 0dB fit
02 m empty 1.00× d 1.00 0.0st 0dB fit
03 m empty 1.00× d 1.00 0.0st 0dB fit
04 m empty 1.00× d 1.00 0.0st 0dB fit
lane 2: 73 pieces · 16.0 s
released (idle) · 48.0 kHz · 160 ms
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1-8 lane g gen l loop z…n stutter w record gesture ? keys


play all
sync
master
0.0 dB
midi
learn
rec
MAP lane 1
play
path
clear path
timbre y
02 COMPOSITION
glide
chain
seed 42601
length 16.0 s
piece min 40.0 ms
piece max 1.50 s
BLEND
overlap of the pieces
overlap min 10%
overlap max 10%
moves random drift ramp wave fixed
changes 2.0×
03 GENERATE
path pull 70%
generate lane 2
grain 80.0 ms
density 20 /s
spray 0.015
envelope hann perc flat
pitch
pitch rnd
reverse
pan rnd
GESTURES
lane 2 · glide


play all
sync
master
0.0 dB
midi
learn
rec
MAP lane 1
play
path
clear path
timbre y
FREQ SHIFT
DELAY
LFO 1
LFO 2
CRUSH
GESTURES
lane 2 · glide


system default
2 ch
play all
sync
master
0.0 dB
midi
learn
rec
MAP lane 1
play
path
clear path
timbre y
lane 2)
lane 2 · glide
02 OUTPUT
master 0.0 dB
latency 10 20 40 80 ms
graphics 60 30 15 fps
when silent release audio keep running
priority below ableton normal
cpu cores all not 1+2 upper half
restart engine
renders folder
reverb out out 1-2
everything → all to …
device: 2 channels. pairs above 1-2 need a device with more channels.
03 LOOK · CONTROL
colour 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
waves spectral plain
spectral: red = low · amber = mid · blue = high · grey = noisy · height = loudness
enable midi
clear mappings
learn (top right): click a control, move a knob or hit a pad. inspector controls follow the selected lane.
04 STORAGE
analysis cache 107 MB
resampled copies 0 kB clear
sediment previews 0 kB clear
live lanes 13 MB
sets 0 kB
renders / recordings 0 kB
ANALYSED FOLDERS
corpus · in use 107 MB forget
nothing grows per generate. the analysis cache grows per new sample folder (decoded audio). lanes are overwritten, sets / recordings only when you save / record.
grain 80.0 ms
density 20 /s
spray 0.015
envelope hann perc flat
pitch
pitch rnd
reverse
pan rnd
GESTURES w record · k all · fit = to the selected lane's loop
01 02 03 04 m empty 1.00×
released (idle) · 48.0 kHz · 160 ms 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

SOUND · 2026

sediment·fx

FIELD	Sound — corpus-based composition and live processing
VERSION	1.0
TECHNOLOGY	Python, numba-compiled audio engine, shared-memory audio process, ASIO/WASAPI/CoreAudio
SOURCE	sediment_fx_1.0.zip

A realtime effect unit applying sediment's ordering modes to live input. Incoming audio is written to memory and analysed as it arrives; up to eight lanes re-order pieces of the immediate past in time and route them to arbitrary discrete outputs. Architecturally it separates a compiled, allocation-free audio body from an analysing ‘brain’ that schedules events ahead, which makes multichannel operation at low latency possible.

SOURCES 2 / 8 — scan inputs +

1 in 1 — input — in 1 — mono — ×
trim — 0.0 dB
freeze | mute | dry — ons 50%

2 in 2 — input — in 2 — mono — ×
trim — 0.0 dB
freeze | mute | dry — ons 50%

1	on	scan	1 · in 1	2/s	→ 1-2	M S cap		0.0
2	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
3	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
4	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
5	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
6	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
7	on	mirror	1 · in 1		→ 1-2	M S cap		0.0
8	on	mirror	1 · in 1		→ 1-2	M S cap		0.0

lane | routing | engine

01 AUDIO

host ALSA
in default · 64 ch
out default · 64 ch
rate 48.0 kHz
block driver
latency in 8.0 ms · out 8.0 ms
engine ready (compiled in 4.34s)

change device …

02 TIMING · CPU

lookahead — 20.0 ms
max voices — 160
cpu guard — auto

the audio engine runs in its own high-priority process. the cpu view in the header: fill = system, line = busiest core, bar = audio engine (share of its time per block).

03 MASTER

master — 0.0 dB
limiter — on
ceiling — -0.3 dB

REVERB

size — 60%
decay — 3.50 s
damp — 45%
return — -6.0 dB
freeze — hold

MAP lane 1 · scan — spectral — fit map

lane 1 · scan
64 pieces
2 grains/s
memory 0.30 s · F#2

timbre y →
timbre x →

inputs — 64 ch · default

PLAYERS — files as inputs

a – no file — ▶ ⏮ load — 0.0 dB
b – no file — ▶ ⏮ load — 0.0 dB

CORPUS — a sample folder as memory

– none — folder × — 8 min

OUTPUTS 64 ch — meter all outputs — spectrum meter group

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64

master — 0.0 dB | reverb — -6.0 dB | decay — 3.50 s

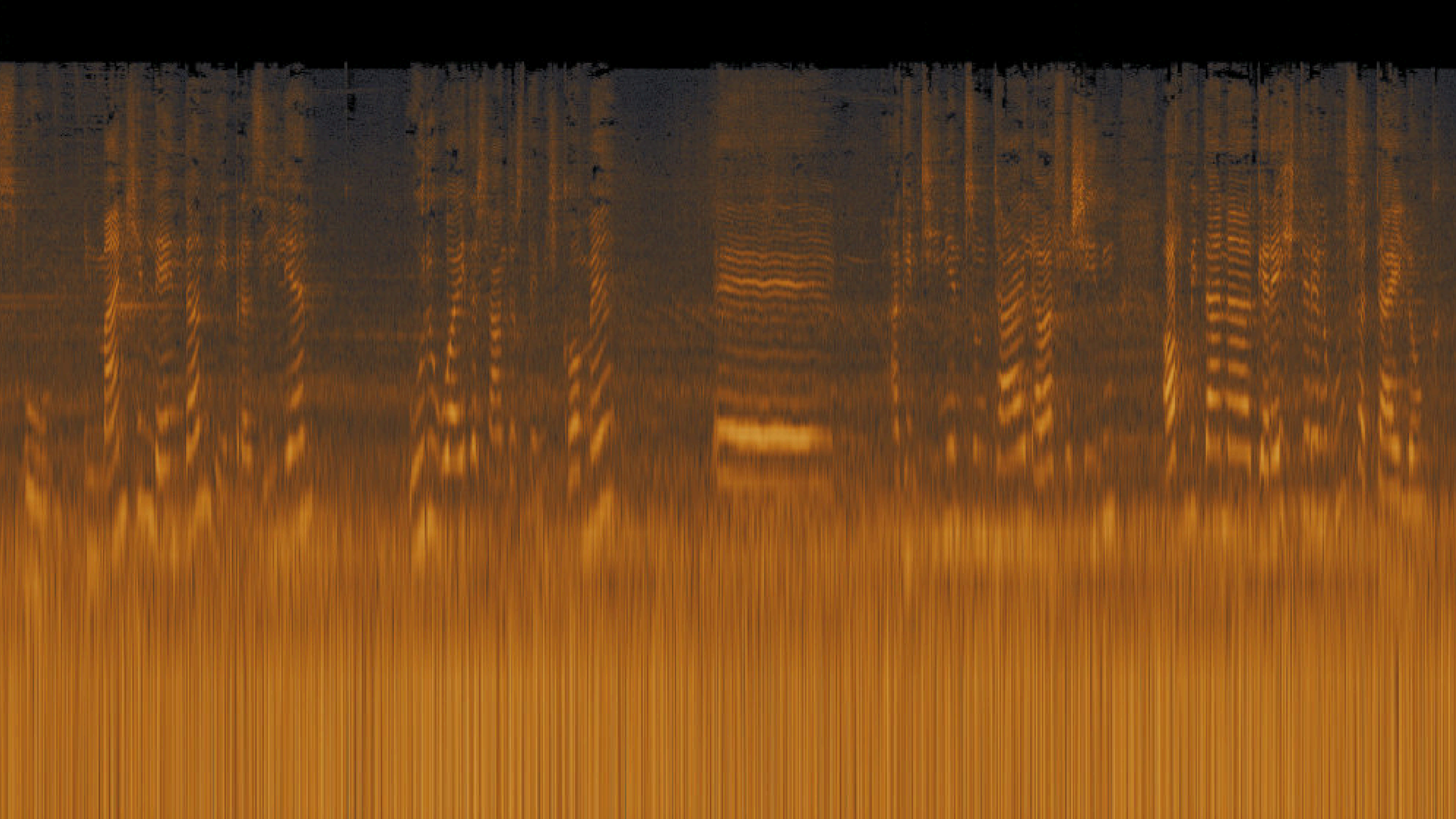

64 in · 64 out · 48.0 k
M -111.2 · S -81.3
I — LUFS · pk -20.8
100% · 100%
dsp 2%
midi
learn
?
rec
inputs
1 on scan 1·in 1 2/s → 1-2 M S cap 0.0
2 on mirror 1·in 1 → 1-2 M S cap 0.0
3 on mirror 1·in 1 → 1-2 M S cap 0.0
4 on mirror 1·in 1 → 1-2 M S cap 0.0
5 on mirror 1·in 1 → 1-2 M S cap 0.0
6 on mirror 1·in 1 → 1-2 M S cap 0.0
7 on mirror 1·in 1 → 1-2 M S cap 0.0
8 on mirror 1·in 1 → 1-2 M S cap 0.0
lane routing engine pick · time voice · fx
lane 1 · scan · in 1
01 PICK
scan mirror shadow follow glide drift contrast sort loop random
reads the live input directly — no analysis delay. delay 0 = the grains are as early as the input.
source 1·in 1
delay 0.00 ms
spray 250 ms
02 TIME
length min 40.0 ms
length max 180 ms
place cloud chain onset grid
density 18 /s
FOLLOW THE INPUT
gate gate
envelope 0%
MAP lane 1 · scan
timbre y →
OUTPUTS 64 ch
master 0.0 dB reverb
48.0 k · 0 v 1 2 3 4 5 6 7 8 9 10 11 12 13
routing
output →
s1 in 1 dry off
s2 in 2 dry off
L1 lane 1 · scan stereo
L2 lane 2 · mirror stereo
L3 lane 3 · mirror stereo
L4 lane 4 · mirror stereo
L5 lane 5 · mirror stereo
L6 lane 6 · mirror stereo
L7 lane 7 · mirror stereo
L8 lane 8 · mirror stereo
rv reverb
64 outputs on default. click / drag paints. per row: 1 channel = mono, 2 = stereo (lower = left), more = grains spread across them in channel order. dry: the input without processing. double-click an output number to name it.
in · 64 out · 48.0 k
M -45.4 · S -49.7
I -43.2 LUFS · pk -20.5
100% · 100%
dsp 2%
1 on scan 1·in 1 2/s → 1-2 M S cap 0.0
2 on mirror 1·in 1 → 1-2 M S cap 0.0
3 on mirror 1·in 1 → 1-2 M S cap 0.0
4 on mirror 1·in 1 → 1-2 M S cap 0.0
5 on mirror 1·in 1 → 1-2 M S cap 0.0
6 on mirror 1·in 1 → 1-2 M S cap 0.0
7 on mirror 1·in 1 → 1-2 M S cap 0.0
8 on mirror 1·in 1 → 1-2 M S cap 0.0
lane routing engine
01 AUDIO
host ALSA
in default · 64 ch
out default · 64 ch
rate 48.0 kHz
block driver
latency in 8.0 ms · out 8.0 ms
engine ready (compiled in 4.34s)
change device …
02 TIMING · CPU
lookahead 20.0 ms
max voices 160
cpu guard auto
the audio engine runs in its own high-priority process. the cpu view in the header: fill = system, line = busiest core, bar = audio engine (share of its time per block).
03 MASTER
master 0.0 dB
limiter on
ceiling -0.3 dB
REVERB
size 60%
decay 3.50 s
damp 45%
return -6.0 dB
freeze hold
MAP lane 1 · scan
timbre y →
OUTPUTS 64 ch
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
master 0.0 dB reverb
48.0 k · 0 v 1 2 3 4 5 6 7 8 9 10 11 12 13

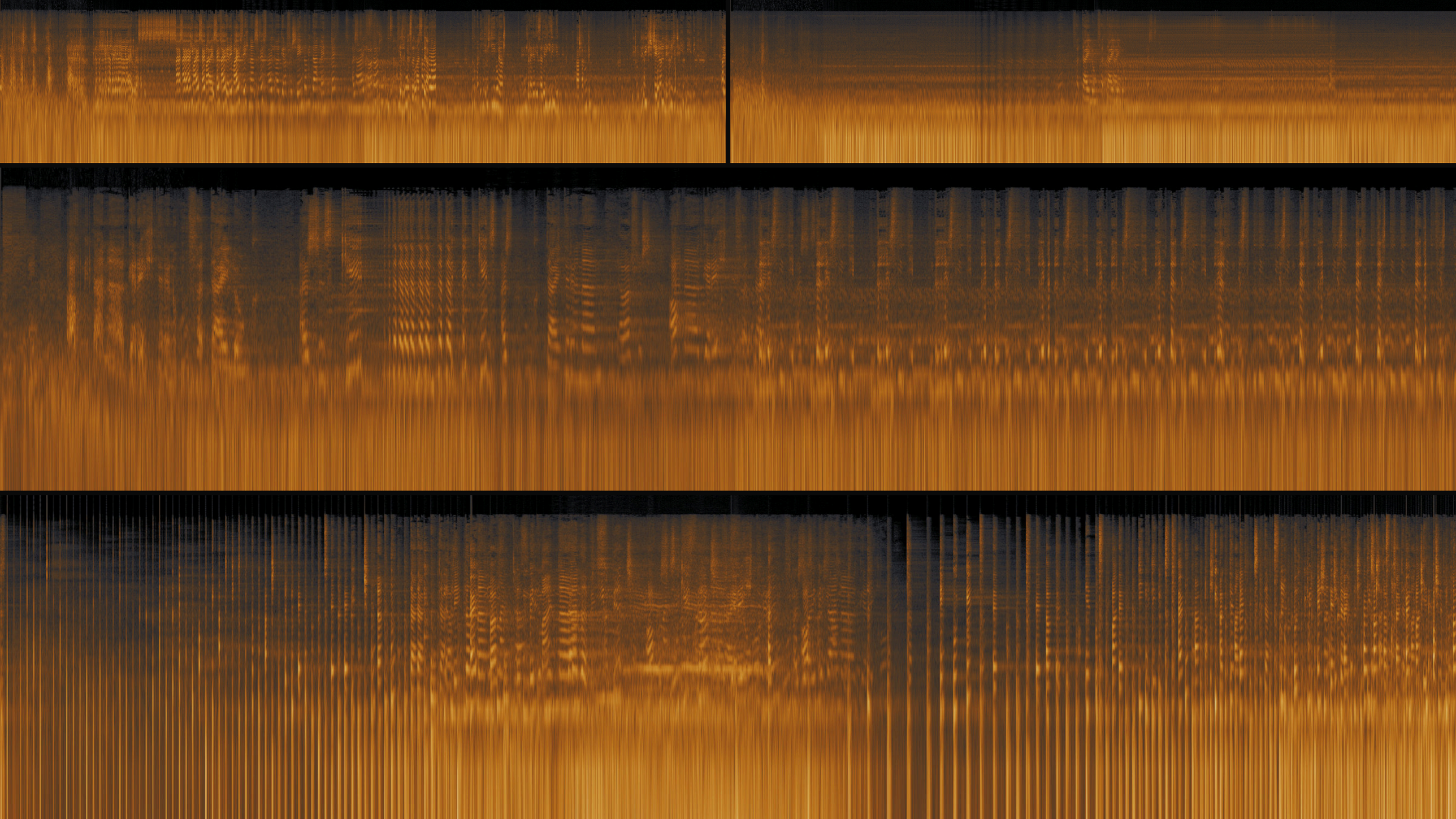
SOUND · 2026

sediment · pod

FIELD	Sound — corpus-based composition and live processing
VERSION	v3
TECHNOLOGY	C++ firmware for Electrosmith Daisy Pod (STM32H7); host-side offline renderer
SOURCE	sediment_pod_v3.zip

Firmware that ports sediment · fx to the Electrosmith Daisy Pod. Stereo line input is held in 87 s of memory; sixteen ‘moves’ (scan, loop, sorts by loudness, brightness, pitch, noise and flux, mirror, shadow, glide, contrast, drift, onset, grid) re-order pieces of it. One knob spans the grain continuum from 2 ms buzz to 2.5 s wash, the other controls the move. Plate: spectrograms rendered offline from the firmware's own engine on a radio recording.





PICTURE-BOUND SOUND · 2026

ligature

FIELD	Picture-bound sound — audiovisual scoring
TECHNOLOGY	Python engine (numpy/scipy), HTML/Web Audio UI, imageio-ffmpeg
SOURCE	261006_ligature.zip

Reads a video frame by frame and builds a soundtrack from a sample folder. Image descriptors — brightness, hue axes, motion and its direction, edge density, cut impulses — are routed through a modulation matrix to granular parameters: which sample sounds, where in it grains are read, pitch, grain size, scan speed, filtering, space and stereo position. Picture and sound are bound sample-accurately; regions can be re-rendered locally.

ligature video-bound scoring

▶ play | loop | 00:07.20 / 00:32.00 | undo | redo | final | render

cpu 100% engine 0% · 153MB
ui 0.0/0ms · 0 audio idle

render soundtrack · failed
load a sample folder first

source

1280×720 · 25.000 fps · 800 fr

/srv/tw/video/abstract.mp4 …

analyse / open | recent projects

samples

/srv/tw/corpus …

analyse folder | all | none

sample space

tone ↔ · grit ↕

dots = samples (size: activity). rings = what each layer is playing at the playhead. the image moves a target through this space.



frame 180

lanes brightness × | hue red↔cyan × | motion × | detail / edges × | pitch × | lowpass × | + lane

drag = mark · wheel = zoom · shift+wheel = scroll

fit

00:00.00 | 00:05.00 | 00:10.00 | 00:15.00 | 00:20.00 | 00:25.00 | 00:30.00

brightness
hue red↔cyan
motion
detail / edges
pitch
lowpass

no render yet — press render

regions 0 placed · drag on the timeline to mark a part to redo · enter regenerate · click a region to select it

mapping | layers | engine | fx | export

preset — load preset — | mutate | random | clear

	pitch	scan speed	grain size	density	spray	reverse	level	pan	width	lowpass	highpass	resonance	drive	chorus	pulse	echo	reverb	space size	match · tone	match · grit	match · activity	match · height
brightness																						
contrast																						
flicker																						
balance L↔R ±																						
hue red↔cyan																						
hue yellow↔blue																						
colour shift																						
saturation																						
warmth ±																						
colourfulness																						
detail / edges																						
sharpness																						
tonal entropy																						
motion																						
motion ← → ±																						
motion ↓ ↑ ±																						
zoom out↔in ±																						
rotation ±																						
motion pos x ±																						
motion spread																						
motion coherence																						
frame change																						
cut impulse																						

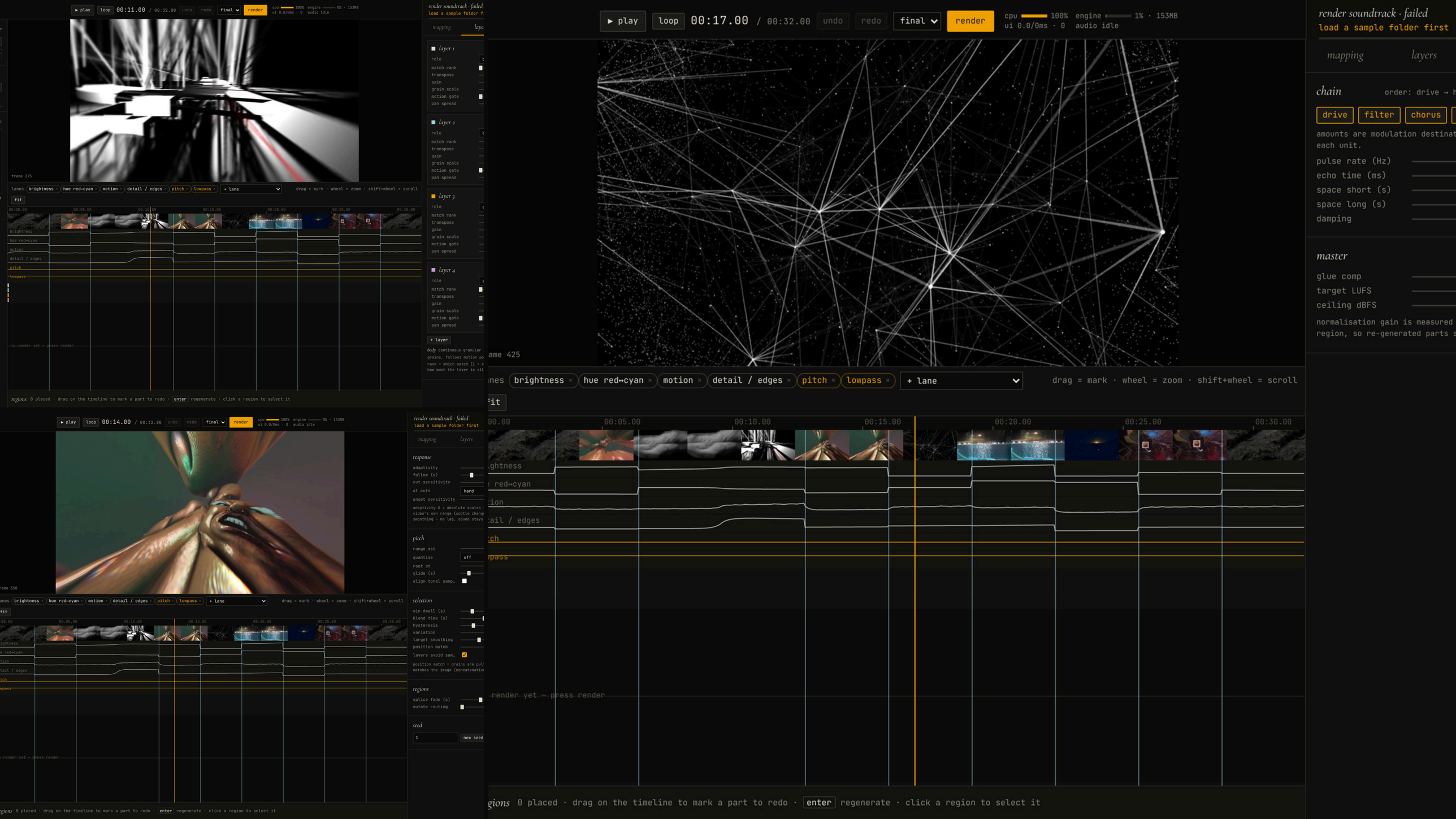
drag a cell up/down · double-click toggles · right-click clears

sensitivity 1.00

base values

value with zero modulation

pitch	0.50	scan speed	0.25
grain size	0.50	density	0.50
spray	0.12	reverse	0.00
level	0.80	pan	0.50
width	0.50	lowpass	0.85
highpass	0.06	resonance	0.08
drive	0.00	chorus	0.08
pulse	0.00	echo	0.06
reverb	0.30	space size	0.40
match · tone	0.50	match · grit	0.50
match · activ…	0.50	match · height	0.50


▶ play
loop
00:17.00 / 00:32.00
undo
redo
final
render
cpu 100% engine 1% · 153MB
ui 0.0/0ms · 0 audio idle
render soundtrack · failed
load a sample folder first
mapping
layers
chain
order: drive → h
drive
filter
chorus
amounts are modulation destinat
each unit.
pulse rate (Hz)
echo time (ms)
space short (s)
space long (s)
damping
master
glue comp
target LUFS
ceiling dBFS
normalisation gain is measured
region, so re-generated parts s
ame 425
brightness
hue red↔cyan
motion
detail / edges
pitch
lowpass
+ lane
drag = mark · wheel = zoom · shift+wheel = scroll
fit
00:05.00
00:10.00
00:15.00
00:20.00
00:25.00
00:30.00
render yet — press render
gions 0 placed · drag on the timeline to mark a part to redo · enter regenerate · click a region to select it

PICTURE-BOUND SOUND · 2026

sortcut

FIELD	Picture-bound sound — audiovisual scoring
VERSION	1.0
TECHNOLOGY	Python (numpy, OpenCV, torch/CUDA: CLIP, YOLO, RAFT, Depth Anything, MediaPipe), browser UI, NVDEC/NVENC, OpenTimelineIO
SOURCE	260929_sortcut.zip

A generative montage system for surveillance footage. Every frame of every camera is described once — colour, light, motion and optical flow, presence, composition, CLIP semantics, detections, faces, pose, depth and real clock time — and the material is then cut by sorting rather than by chance: 74 presets in families such as perceptual, continuity, semantic, surveillance, people, faces, image, depth, time, structure and audio. Continuity constraints, curve readings of every ranking and a use-once rule turn sorting into editing; each render is reproducible from its sidecar (CSV, OTIO, JSON) and can be streamed as OSC for sonification.

SOURCES

6/6 · 1440 F

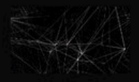
INGEST RESCAN ALL NONE



01 cam01
1280×720 · 240 f · 1.00 fps
time: index



02 cam02
1280×720 · 240 f · 1.00 fps
time: index



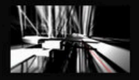
03 cam03
1280×720 · 240 f · 1.00 fps
time: index



04 cam04
1280×720 · 240 f · 1.00 fps
time: index



05 cam05
1280×720 · 240 f · 1.00 fps
time: index



06 cam06
1280×720 · 240 f · 1.00 fps
time: index

INGEST OPTIONS +

ANALYSIS

7/12 COMPLETE

- color 6/6
- motion 6/6
- background 6/6
- structure 6/6
- flow 6/6
- embed 6/6
- detect n/a
- face n/a
- pose n/a
- depth n/a
- ocr n/a
- rarity 6/6

ANALYZER OPTIONS +

redo

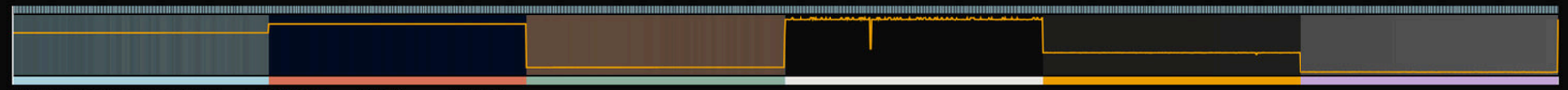
ANALYZE

CHROMATIC DRIFT

1440 UNITS → 1440 F · 48S · 1440 CUTS · STEP 0.014 · 100% UNIQUE · 0.14S



PLAY ⟲ START ×.25 ×.5 ×1 ×2 ×4 ■ loop 1 / 1440 0:00.0 / 0:48.0



SOURCE COLOUR PER FRAME — SORT KEY | CUTS ▬ CAMERA



PARAMETERS RENDER JOBS

METHOD

chromatic drift

- method: sort by feature
  rank all units by one feature, then read the ranking along a curve
- key: hue_dom – dominant hue
- desc: on off
- curve: linear
- cycles: 3.0
- circ start: auto fixed
- circ deg: 0
- gray mode: end start drop keep
- chroma min: 6.0

CONTINUITY

- once: on off
- tight: 0.90
- cont space: color+L,a,b

UNIT

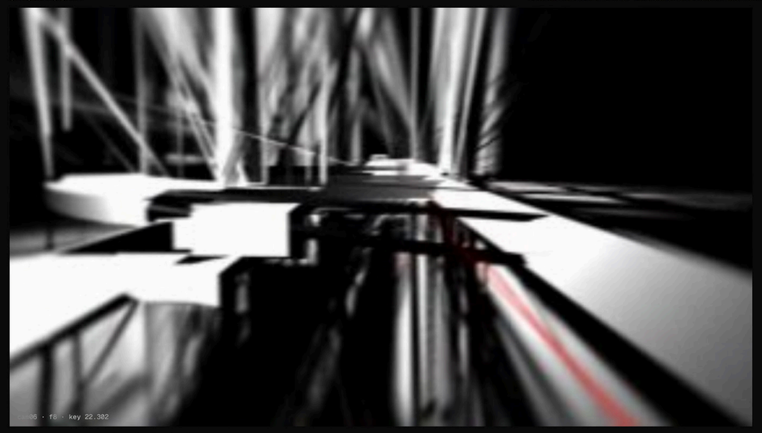
- unit: frame grain event scene
- grain len: 8
- grain jitter: 0.00
- grain dir: forward reverse pingpong random
- event signal: motion bgdiff persons fg_area objects
- event q: 0.75
- event min: 2
- event max: 60
- event pad: 1
- event merge: 3
- scene q: 0.97

48s · 1440 cuts

SEQUENCE

BROWSE OPEN

CHROMATIC DRIFT



PAUSE ⟲ START ×.25 ×.5 ×1 ×2 ×4 ■ loop 1320 / 1440 0:44.0 / 0:48.0

SOURCE COLOUR PER FRAME — SORT KEY | CUTS ▬ CAMERA



BROWSE OPEN

SEQUENCE EXPLORE RENDERS

PRESETS

PERCEPTUAL
01 chromatic drift
02 chromatic drift · grains
03 lab path
04 synthetic day
05 synthetic night
06 motion swell
07 motion breath
08 stillness

CONTINUITY
09 impossible camera
10 impossible camera · composition
11 match cut
12 match cut · semantic
13 anti-continuity
14 drift apart
15 motion relay
16 motion sweep
17 wind rose

SEMANTIC
18 prompt: empty → crowd
19 prompt: weather
20 prompt: waiting
21 unsupervised axis
22 principal axis
23 typology
24 object chain

SURVEILLANCE
25 · population
26 · ghost
27 rarity
28 rarity · only anomalies
29 panopticon
30 panopticon · long stare
31 absence
32 only absence

PEOPLE
33 · empty → crowded
34 · crowd swell
35 · approach
36 · arms rising
37 · falling
38 · pose rhyme
39 · stride

FACES
40 · eye line
41 · face lock · head turn
42 · face lock · morph
43 · face scale
44 · scream
45 · smile
46 · eyes closing
47 · gaze sweep
48 · expression rhyme

IMAGE
49 void → noise
50 texture scale
51 focus pull
52 rotating world
53 visual weight
54 monochrome → polychrome
55 cold → warm
56 symmetry

DEPTH
57 · tunnel → wall
58 · near → far
59 · depth rhyme

TIME
60 one day

STRUCTURAL
61 hue × light hilbert
62 semantic spiral
63 converge
64 fractal brightness
65 light then colour
66 · people then place
67 verticals
68 wall · per cam brightness
69 canon · 4 screens

AUDIO
70 audio: loudness → light
71 audio: brightness → motion
72 audio: stutter

BASELINE
73 original
74 chronological merge

the whole piece becomes one colour gradient across all cams

SEQUENCE history

ANALYZE

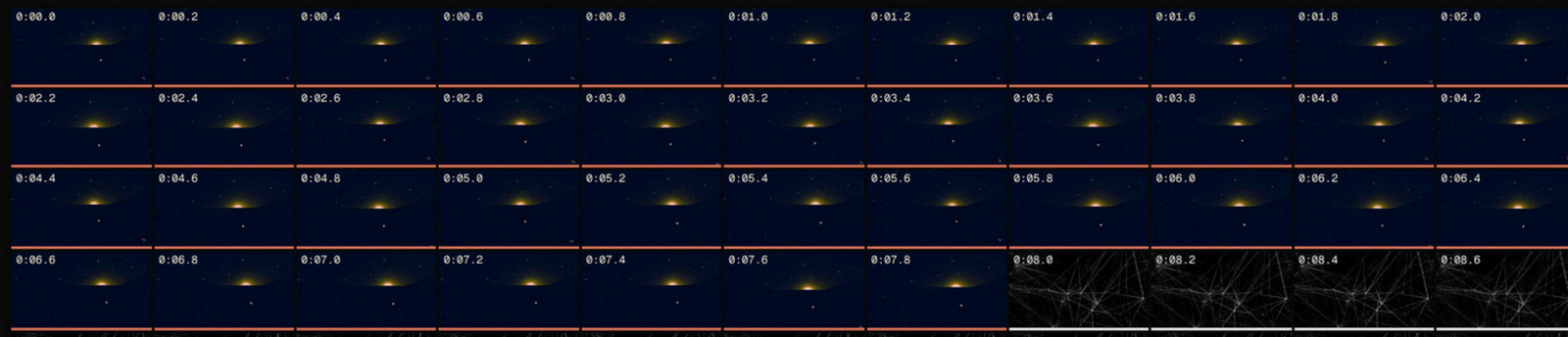
IMPOSSIBLE CAMERA

1440 UNITS → 1440 F · 48S · 1440 CUTS · STEP 0.022 · 100% UNIQUE · 1



PAUSE ⟲ START ×.25 ×.5 ×1 ×2 ×4 ■ loop 412 / 1440 0:13.7 / 0:48.

SOURCE COLOUR PER FRAME — SORT KEY | CUTS ▬ CAMERA



PICTURE-BOUND SOUND · 2026

scorecut

FIELD	Picture-bound sound — audiovisual scoring
VERSION	1.3
TECHNOLOGY	Python (numpy, scipy, OpenCV optical flow), embedded browser UI, ffmpeg
SOURCE	scorecut_1.3.zip

Companion to sortcut (T06): sortcut makes the cut, scorecut makes it sound. The cut video is decomposed into scenes and cameras; each camera becomes a ‘station’ of an audio corpus, switched at the exact frame of the cut, while image features drive a spectral and spatial treatment of the sound. Thirty presets in eight families; cut-aware smoothing guarantees that every change lands on its frame. Output: high-resolution audio, the stream-copied picture with new sound, and a cue list.

SOURCE

/srv/tw/video/abstract.mp4 pick

audio file or folde files folder

srv/tw/corpus/ ×

analyse cached after the first pass

PRESET revert

CHANNEL clean tuner resume dead-air
OPTIC focus luminance grain depth exposure
CHROMA fifths common warmth prism
KINETIC stillness drift pressure tape
SPECTRAL lattice shift entropy mirror spatial
VOID night infrared sodium
STRANGE height camera novelty total
MATCH corpus sorted

dead air + relay tick at every cut; image grain opens the carrier hiss

save as … save

BINDINGS

preset depth 1.0 ×
unleash off on
file dice roll ×
position dice roll ×

SCENES

SELECTION

file per scene
channel match sorted sequence random single
position in file
broadcast resume head match song
variety 0.40
distinct neighbours off on
avoid silence off on
sort axis
bright energy tone texture mass
luma ↔ centroid 1.00
motion ↔ onsets 0.80
saturation ↔ tonality 0.60



▶ play ◉ audition ◌ proxy ↻ loop → flow files positions 00:00:06:15 / 00:00:32:00



MAPPINGS · IMAGE → SOUND + map

sensor noise → noise floor ×
0 … 0.35 × 1 route frame scene
lin exp log s st4 st8 gate
smooth 120 ms

MOTION LAYER · OPTICAL FLOW + map

FLOW off imprint carve tide doppler scrub flutter swarm rotor impulse gaze vertical wake full

custom layer

motion layer off on
layer depth 1.0 ×
imprint
off spotlight carve boost
imprint depth 18 dB
imprint pan 0.60
imprint curve 1.0
imprint attack 6 ms
imprint release 240 ms
bottom row 70 Hz
top row 12000 Hz
flutter shape sine pulse
flutter restarts at cut off on

LAYER MAPPINGS — ADDED ON TOP OF ANY PRESET

pick a flow preset above or + map — flow sources are in the 'flow' group

CORPUS 6 files

file	dur	key	LUFS	centr	scn
10.wav	0:30	G#m	-22.5	426	1
11.wav	0:36	A#m	-23.5	107	1
12.wav	0:42	C#	-22.2	228	2
13.wav	0:48	C#	-19.3	359	2
14.wav	0:54	A#m	-21.2	385	2
15.wav	1:00	C#	-19.1	317	2

render cancel 800 frames · 6 files

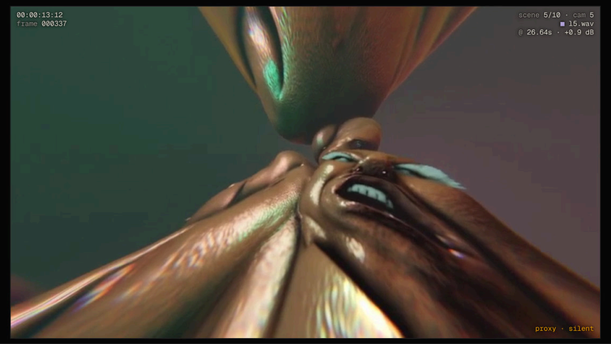
abstract.mp4 1280×720 25.000 fps · cfr 800 frames 0:32 flow · dis



play audition proxy loop flow files positions 00:00:20:07 / 00:00:32:00

positions re-rolled #471097

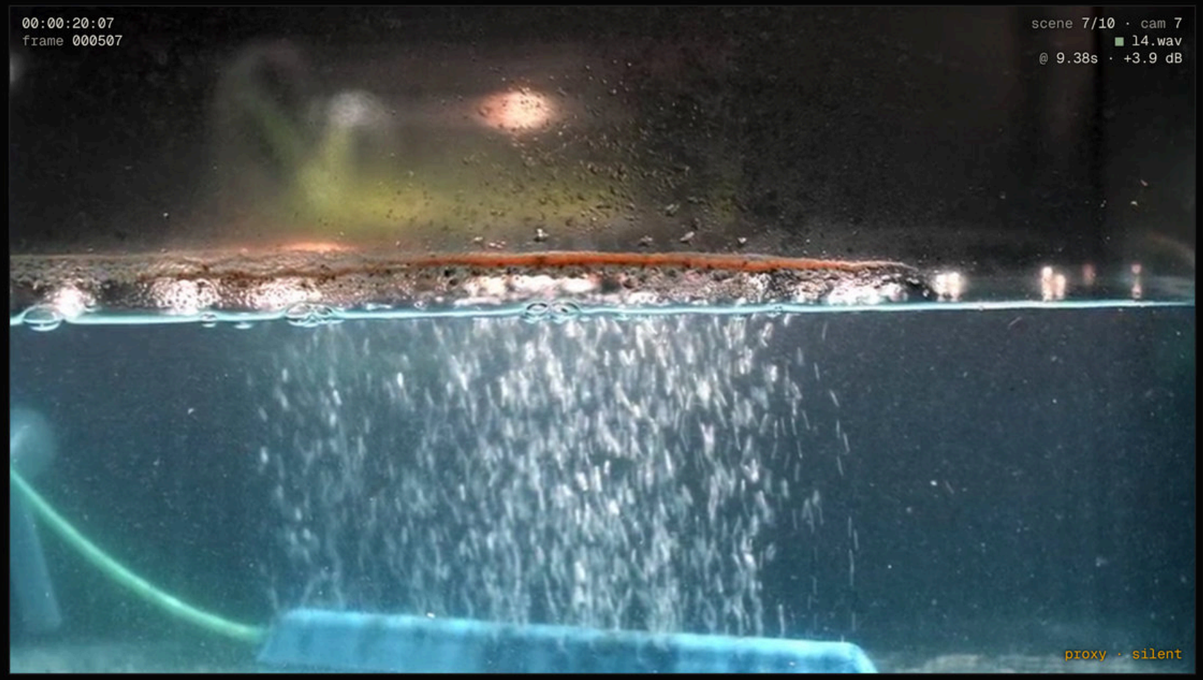
abstract.mp4 1280×720 25.000 fps · cfr 800 frames 0:32 flow · dis



▶ play audition proxy loop flow files positions 00:00:13:12 / 00:00:32:00

800 frames · 6 files

abstract.mp4 1280×720 25.000 fps · cfr 800 frames 0:32 flow · dis



play audition proxy loop flow files positions 00:00:20:07 / 00:00:32:00

rientation → harm ↔ perc

etail → resolution loss

ntropy → diffusion

800 frames · 6 files

MOVING IMAGE · 2026

datamosh station

FIELD	Moving image — process, analysis, capture
VERSION	2
TECHNOLOGY	Python (numpy/OpenCV or CUDA/torch), browser UI, ffmpeg/xvid
SOURCE	260801_Datamosher2.zip

A standalone tool for controlled image decay. Two chainable engines: a flow engine in which measured motion is reshaped through 45 parameters and applied to a canvas that is never refreshed — repeatable at any resolution — and a bitstream engine that deletes I-frames and duplicates P-frames in real MPEG-4 data, leaving the damage to the decoder. Thirty-two presets in five families (smear, glitch, fluid, dream, brutal); identical results on CUDA and CPU back-ends.

DATAMOSH *a station for controlled decay*

■ torch is not installed — CPU engine only

SOURCES 1

DROP A VIDEO

or a local path / folder | ADD

abstract.mp4
1280×720 · 25 fps · 32s · h264

RENDERS 0

abstract.mp4 1280×720 source



the decay, one column per frame

IN 0 | LEN all | SEED 0 | AUDITION | RENDER | STOP | DRIFT | RESET

PRESETS 32

SMEAR *long, continuous, gravitational*

tar
everything sinks. slow, black, unrecoverable.

silk
a long soft pull. the image is being drawn through fabric.

undertow
the picture is pulled sideways by something under it.

long_exposure
motion accumulates until the frame is a single gesture.

pull
row by row. horizontal drag, paint on glass.

GLITCH *discontinuous, blocky, hostile*

shredder
macroblocks with no loyalty to their neighbours.

macroblock_storm
the codec giving up, loudly and on schedule.

corrupt_index
the file's frame index is lying. nothing lands where it should.

teeth
fast, bright, bitten. short attention span.

packet_loss
bands of the frame simply never arrive.

FLUID *smooth motion fields, no hard edges*

oil_slick
thick, iridescent, always moving. petroleum on water.

current
a steady body of water carrying the image downstream.

mercury
heavy, reflective, reluctant to move and impossible to stop.

breathing_water
in and out. the surface never settles.

vapour
barely holding together. thin, drifting, cold.

DREAM *soft, luminous, forgetful*

haze
soft, warm, forgetful. nothing is quite where you left it.

ectoplasm
figures leave residue. the room remembers them.

soft_bloom
cuts open into light instead of into the next shot.

milk
everything dissolving into a pale field. very little left.

sleepwalk
slow stutter. the image keeps moving after it stops.

BRUTAL *no refresh, no ceiling*

rcury · audition 640×360 · 99 frames

render



LEN all SEED 0 AUDITION RENDER STOP DRIFT RESET

37s · done

n/ectoplasm · audition 640×360 · 100 frames

render

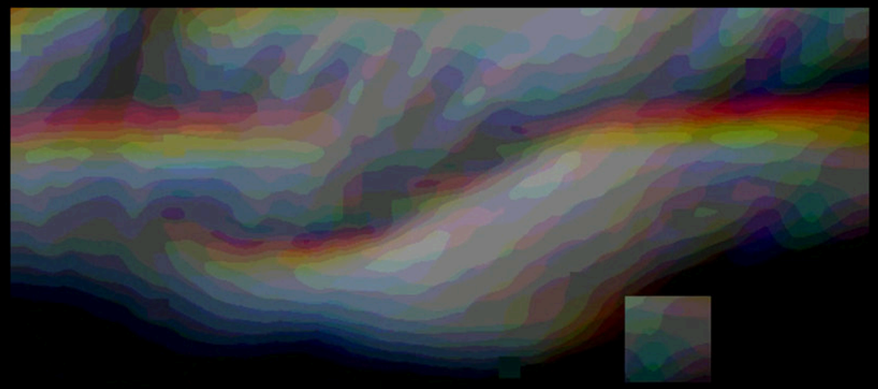


16.2 LEN all SEED 0 AUDITION RENDER STOP DRIFT RESET

100 21.7s · done

/macroblock_storm · audition 640×360 · 99 frames

render



.5 LEN all SEED 0 AUDITION RENDER STOP DRIFT RESET

00 13.2s · done

MOVING IMAGE · 2026

scanline

FIELD	Moving image — process, analysis, capture
VERSION	1.1
TECHNOLOGY	Python (numpy, OpenCV, optional depth model), browser UI, ffmpeg
SOURCE	scanline_v1.1.zip

A sweeping bar crosses the moving image and reveals, behind it, one of 35 analytical readings — terrain (slope, curvature, ruggedness), photometric (glint, water index, haze), motion (flow, divergence, vorticity, time-to-contact), detection (landmass, horizon, rectilinear anomalies) and depth. The frame is thus never fully in one epistemology. Output: an image sequence, a preview film and a per-frame scan log.

scanline. V1.1

SOURCE

video /srv/tw/video/abstract.m OPEN

output auto — next to source SET

frames 0 0

width 1920

AXIS where the bar travels

behaviour vertical

angle step 23°

period 4.0 s

REVEAL how it lands

behaviour stratigraphy — tail + residue

band 1.6 %

tail 10 %

residue 0.10

comb teeth 4

drift 10 %

seed 7

the dispersed reveals hold their grain for the whole run — change the seed to get a different one. they analyse a wider region, so they cost more per frame.

RENDER crispness

style auto — per mode

opacity 0.85

gamma 1.15

levels 14

line weight 1.0

trace amp 3.5 %

stipple, contour and hatch stay hairline at any resolution — field is the only style that fills.

INSTRUMENT readout

hud full — + histogram, scale, rot

max labels 8

vector gain 3.0

flow scale 0.45

jpg quality 94

sweep bar profile trace

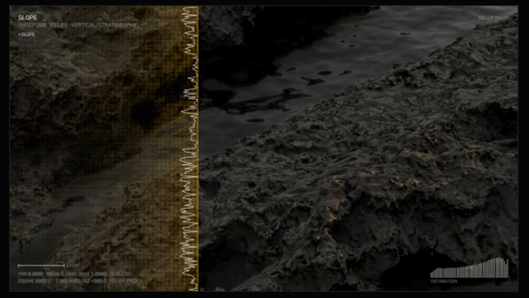
overlay only alpha-ready, black base

SWEEPING STOP



MODE	FRAME	WRITTEN	LABELS	MEAN	PROGRESS
slope	000014	14	0	0.5945	1.8 %

11:58:00 sweep started
11:58:02 abstract.mp4 · 1280x720 · 25.000 fps · 800 frames
11:58:02 out 1280x720 · 4 modes · vertical/stratigraphy · auto







MODE	FRAME	WRITTEN	LABELS	MEAN
slope	000004	4	0	0.5896

11:58:00 **sweep started**
11:58:02 abstract.mp4 · 1280x720 · 25.000 fps · 800 frames
11:58:02 out 1280x720 · 4 modes · vertical/stratigraphy · auto

MOVING IMAGE · 2026

framegrab

FIELD	Moving image — process, analysis, capture
VERSION	fixed-step
TECHNOLOGY	Python/tkinter, Windows Graphics Capture, RIFE (GPU) interpolation, ffmpeg; Unity fixed-step render bridge (C#)
SOURCE	260918_FrameGrabberFixedStep_noIntAudio.zip

Captures every frame an application actually presents — including irregular, frame-generated or lagging sources — with high-resolution timestamps, and conforms them to an even timeline (hold, blend, optical flow or RIFE interpolation). A render mode lets Unity drive capture frame by frame with synchronised audio over UDP, turning real-time engines into deterministic offline renderers.

source

◉ application ○ screen ↻

(nothing found)

☒ client area only (no title bar) ☐ cursor

output

FOLDER /home/dot/Videos/framegrab …

TAKE auto

FRAMES AS bmp min CPU · max disk RAM GB 16 ☐ low priority

record

F9 rec / pause F10 stop

ARM ● REC ❚❚ PAUSE ■ STOP idle

☐ render mode unity drives it frame by frame · audio in sync · udp 8964

FRAMES	SOURCE FPS	RECORDED	DUPES	STALLS
—	—	—	—	—

BUFFER 0 MB

conform

TAKE …

FPS 60 SPEED 1.0

TIMING realtime FILL rife · gpu

CPU FLOW medium MAX GAP ms 250

SEQ AS png QUALITY 16

VIDEO h264 · nvenc RIFE SCALE auto

altime = keep wall-clock timing, fill the gaps
ames = each captured frame becomes one output frame
eed = 4 turns a 0.25x slowed-down app back to normal speed
fe scale auto = 1.0 up to 1440p, 0.5 above (lower = faster, coarse

CONFORM OPEN FOLDER ☒ auto after stop

—

log

12:01:35 global hotkeys unavailable (No module named 'keyboard')
– use the buttons

source

○ application ◉ screen ↻

(nothing found)

☒ client area only (no title bar) ☐ cursor

output

FOLDER /home/dot/Videos/framegrab ...

TAKE auto

FRAMES AS png lossless · smaller · more CPU M 16 ☐ low priority

record

F9 rec / pause F10 stop

ARM ● REC ❚❚ PAUSE ■ STOP idle

☒ render mode unity drives it frame by frame · audio in sync · udp 8964

FRAMES	SOURCE FPS	RECORDED	DUPES	STALLS
—	—	—	—	—

BUFFER 0 MB

conform

TAKE ...

FPS 60 SPEED 1.0

TIMING realtime FILL rife · gpu

CPU FLOW medium MAX GAP ms 250

SEQ AS png QUALITY 16

VIDEO h264 · nvenc RIFE SCALE auto

altime = keep wall-clock timing, fill the gaps
ames = each captured frame becomes one output frame
eed = 4 turns a 0.25x slowed-down app back to normal speed
fe scale auto = 1.0 up to 1440p, 0.5 above (lower = faster, coarse

CONFORM OPEN FOLDER ☒ auto after stop

—

log

```
12:01:35  global hotkeys unavailable (No module named 'keyboard')
— use the buttons
12:01:41  render mode: frames as png (4-6x smaller than bmp,
costs no render time in lock-step)
12:01:41  render mode: ARM here, then start FixedStepRender in
unity (F7). unity starts + stops the take, conform runs 1:1 with
audio.
```

MOVING IMAGE · 2026

AVjoin

FIELD	Moving image — process, analysis, capture
TECHNOLOGY	Windows batch, ffmpeg/ffprobe
SOURCE	260913_AVjoin.bat

Lossless concatenation of numbered video parts (name_partNN) by stream copy: groups by base name, sorts numerically, verifies identical stream parameters across parts and checks the merged duration against the sum of the parts. Counterpart to AVsplitter.

```
AVjoin

 ------------------------------------------------------
  A V j o i n      lossless part merger
 ------------------------------------------------------
  folder : D:\renders\terrain
  pattern: *_partNN.ext

  ffmpeg : on PATH

 ------------------------------------------------------
  group  : terrain_take03

        1  terrain_take03_part01.mov   [0:00:03.200]
        2  terrain_take03_part02.mov   [0:00:03.200]
        3  terrain_take03_part03.mov   [0:00:03.200]
        4  terrain_take03_part04.mov   [0:00:03.200]

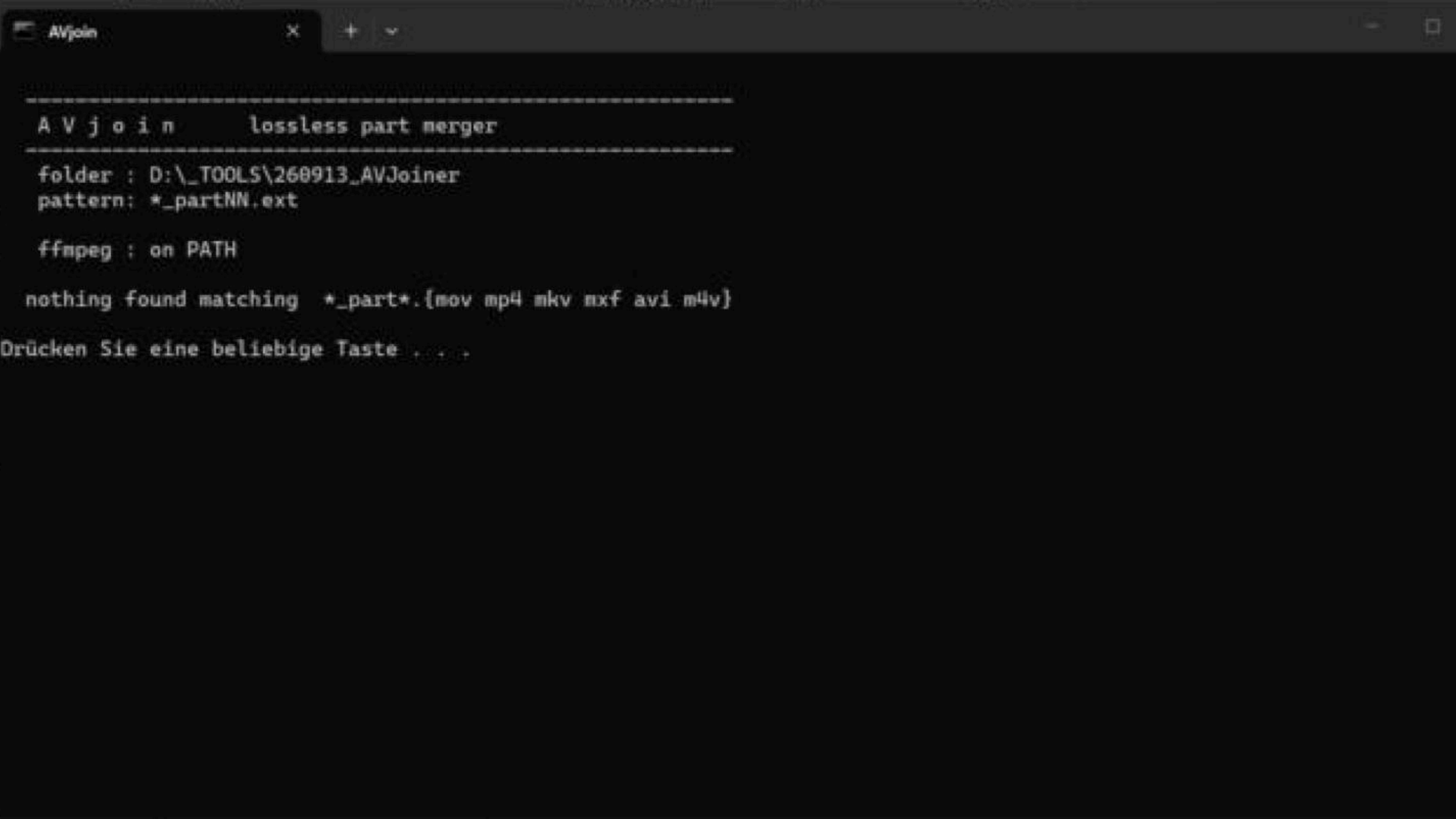
  parts  : 4
  sum    : 0:00:12.800

  writing: terrain_take03_MERGED.mov
size=  137073kB time=00:00:12.76 bitrate=88001.8kbits/s speed=7.21x

  merged : 0:00:12.800
  delta  : 0 ms

 ------------------------------------------------------
  done.

Drücken Sie eine beliebige Taste . . .
```


AVjoin
A V j o i n lossless part merger
folder : D:_TOOLS\260913_AVJoiner
pattern: *_partNN.ext
ffmpeg : on PATH
nothing found matching *_part*.{mov mp4 mkv mxf avi m4v}
Drücken Sie eine beliebige Taste . . .

VOICE AND LANGUAGE · 2026

dubtrans

FIELD	Voice and language
TECHNOLOGY	Python, faster-whisper, Ollama (local LLM), Chatterbox TTS, RVC/Applio, Demucs, rubberband, ffmpeg
SOURCE	261018_Dubtrans.zip

Fully local translation dubbing for long recordings. Speech is transcribed (Whisper, language locked after detection), translated by a local language model that is told each line's duration, synthesised, fitted into the original timing by condensation and time-stretching, converted through a trained RVC voice model and mixed over the original or a separated background. Output: video with English dub and original track plus subtitles.

dubtrans

any language → english · one voice · yours

idle

run queue | cancel

queue

audio or video, any length. folders are scanned recursively.

add files | add folder | clear | open output

or paste a path — file or folder | add

log

```
ffmpeg: /usr/bin/ffmpeg  (rubberband stretch)
dubtrans ready — http://127.0.0.1:8974
```

listen | translate | voice | timing | mix | lines

transcription

spoken language	auto	auto = detected on the first chunk, then locked
whisper model	large-v3	large-v3 ≈ 5 GB VRAM
device	cuda	
compute type	float16	
beam size	5	
chunk length (min)	30	resumable — a crash costs one chunk
VAD min silence (ms)	500	VAD is what keeps whisper sane over hours
initial prompt		names / jargon to bias the transcript

sentences

max sentence length (s)	12	
split at pause ≥ (s)	0.8	shorter = more, smaller dub lines

test run

preview only (min)	0	0 = whole file. 10 = tune everything on a short stretch first

idle

listen translate voice timing mix lines

English lines are placed at the original sentence start. A line that doesn't fit its slot (until the next sentence) is first **condensed** by the translator, then **time-stretched** up to the speed limit; anything left over pushes the following lines later (**drift**), which the next pause absorbs.

fitting

max speed-up 1.3 — 1.30 = never more than 30 % faster

shorten lines that still don't fit (ollama)

min gap between lines (s) 0.12

render

rvc block length (min) 5 — cut in pauses; smaller = less vram per rvc pass

release gpu models

idle

listen translate voice timing mix lines

Every speaker in the source becomes **one english voice**. Chatterbox speaks the english; if a reference clip is set it clones that voice zero-shot. With RVC on, the whole timeline then goes through your trained Applio model — once per block, so the model loads once, not per line.

speech (chatterbox)

engine chatterbox — dry = beeps, checks timing + mix without a gpu

reference voice … — optional — 10-30 s of the target voice

exaggeration 0.5 — 0.5 neutral, higher = more dramatic

cfg / pacing 0.5 — lower = slower, steadier

temperature 0.8

seed 1234

rvc (your trained model, via applio)

run the dub through the rvc model

applio folder … — the folder with core.py — models are read from its logs\

trained model — nothing in applio\logs — rescan

model .pth …

index .index … — optional — falls back to no index if it breaks

pitch (semitones) 0

index rate 0.5

protect consonants 0.33

f0 method rmvpe

extra applio args — passed through verbatim, e.g. --clean_audio True

applio python — empty = <applio>\env\python.exe

listen

This is a test of the

idle

listen translate voice timing mix lines

bed under the english voice

bed: original, ducked under the voice

voice gain (dB): 0

bed gain (dB): -4

duck ratio: 8 — how hard the bed dips while the voice speaks

duck threshold: 0.02

demucs model: htdemucs

output

output folder … — empty = next to each source

video container: mkv — video is stream-copied, never re-encoded

audio-only output: flac

keep the original audio as a second track

embed english subtitles (timed to the dub)

aac bitrate: 256k

subtitle chars / line: 42

VOICE AND LANGUAGE · 2026

subtrans

FIELD	Voice and language
TECHNOLOGY	Python/tkinter, faster-whisper, Ollama, ASS subtitles, ffmpeg/NVENC
SOURCE	260908_Subtitler.zip

Local subtitling pipeline for recordings of three to seven hours: chunked transcription with voice-activity detection and word timestamps, translation through a local language model, typographic styling as ASS, and either soft muxing or hardware-encoded burn-in. Batch-capable; shares its listening and translation core with dubtrans.

files | transcribe | translate | style | output

font Garamond must be installed, or set a fonts folder

size 54 in source-resolution px

text #FFFFFF #RRGGBB

outline #000000

border 1 1 = outline+shadow, 2 = box

box colour #000000

box alpha 110 0 opaque … 255 invisible

outline w 2.4

shadow 3.0

tracking 0.0

align 5 numpad — 2 bottom centre, 8 top

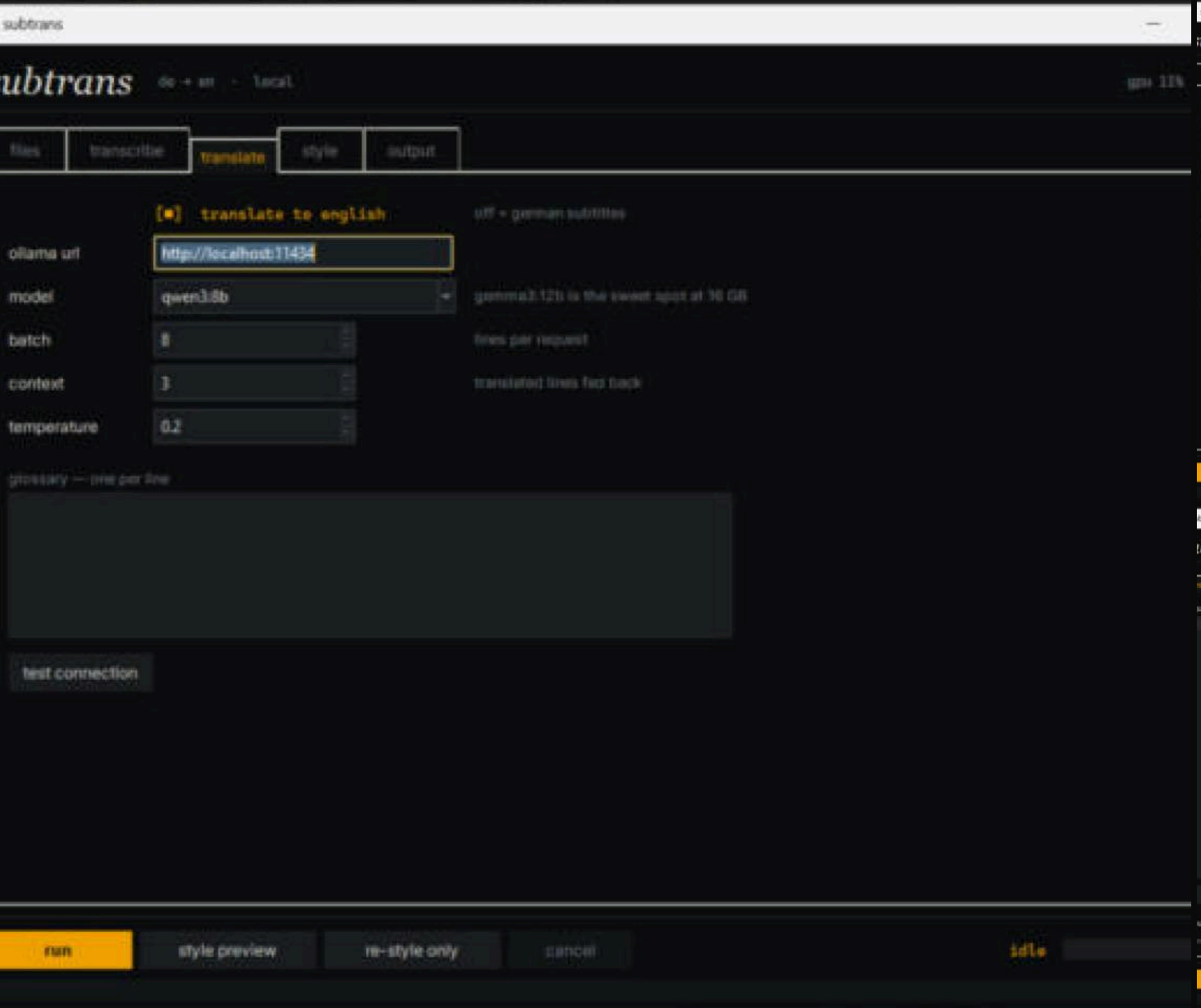
margins L 90 R 90 V 60

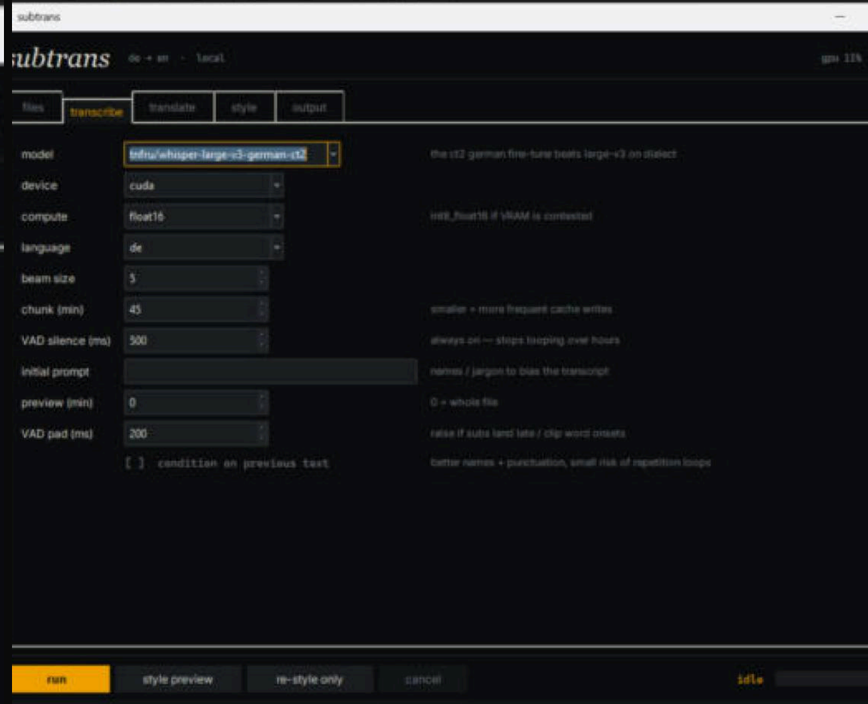
[ ] bold [ ] italic [ ] uppercase

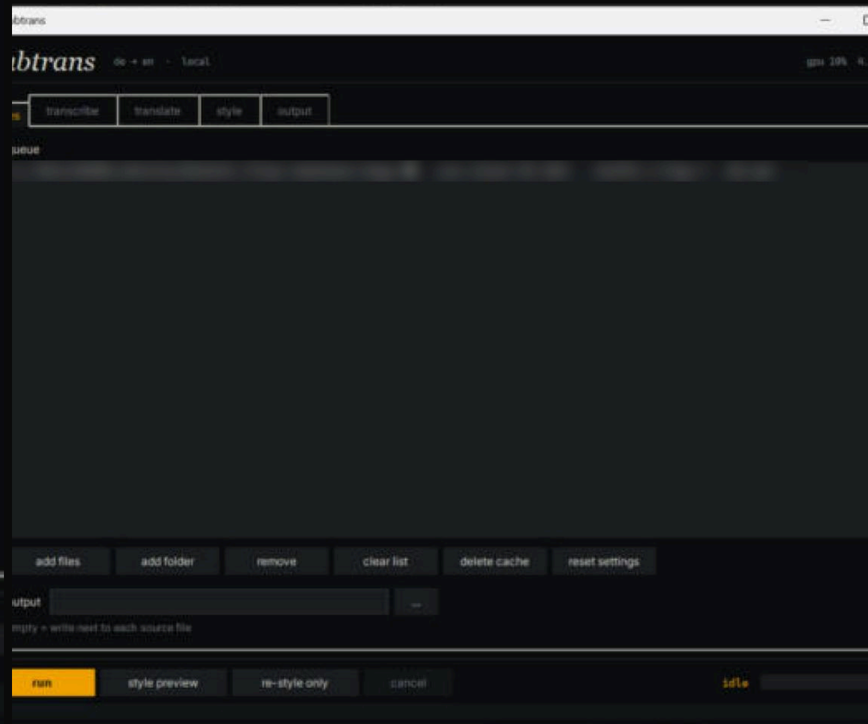
chars / line 42

max lines 2

cue seconds min 1.0 max 6.0


subtrans
ubtrans
de → en · local
gpu 11%
files
transcribe
translate
style
output
[■] translate to english
off = german subtitles
ollama url
http://localhost:11434
model
qwen3:8b
gemma3:12b is the sweet spot at 16 GB
batch
8
lines per request
context
3
translated lines fed back
temperature
0.2
glossary — one per line
test connection
run
style preview
re-style only
cancel
idle


subtrans
ubtrans
de → en · local
gpu 11%
files
transcribe
translate
style
output
model
tnfru/whisper-large-v3-german-ct2
the ct2 german fine-tune beats large-v3 on dialect
device
cuda
compute
float16
int8_float16 if VRAM is contested
language
de
beam size
5
chunk (min)
45
smaller = more frequent cache writes
VAD silence (ms)
500
always on — stops looping over hours
initial prompt
names / jargon to bias the transcript
preview (min)
0
0 = whole file
VAD pad (ms)
200
raise if subs land late / clip word onsets
[ ] condition on previous text
better names + punctuation, small risk of repetition loops
run
style preview
re-style only
cancel
idle


ubtrans
de → en · local
transcribe
translate
style
output
ueue
add files
add folder
remove
clear list
delete cache
reset settings
utput
mpty = write next to each source file
run
style preview
re-style only
cancel
idle

VOICE AND LANGUAGE · 2026

voicebox

FIELD	Voice and language
TECHNOLOGY	Python, Chatterbox TTS, Applio/RVC, aiohttp web UI
SOURCE	260716_VoiceCloner.zip

A local text-to-speech environment whose voice can be a model trained on arbitrary, including non-vocal, audio. Typed text or any input file is synthesised and passed through a trained RVC model; reference voices, expressive tags, extreme slow-down and stereo-preserving conversion are available, and training jobs can be queued from the same interface.

inline tags work: [laugh] [cough] [chuckle] [sigh]

CHATTERBOX

reference voice: default voice | + upload

exaggeration: 0.50

pacing / ref: 0.50

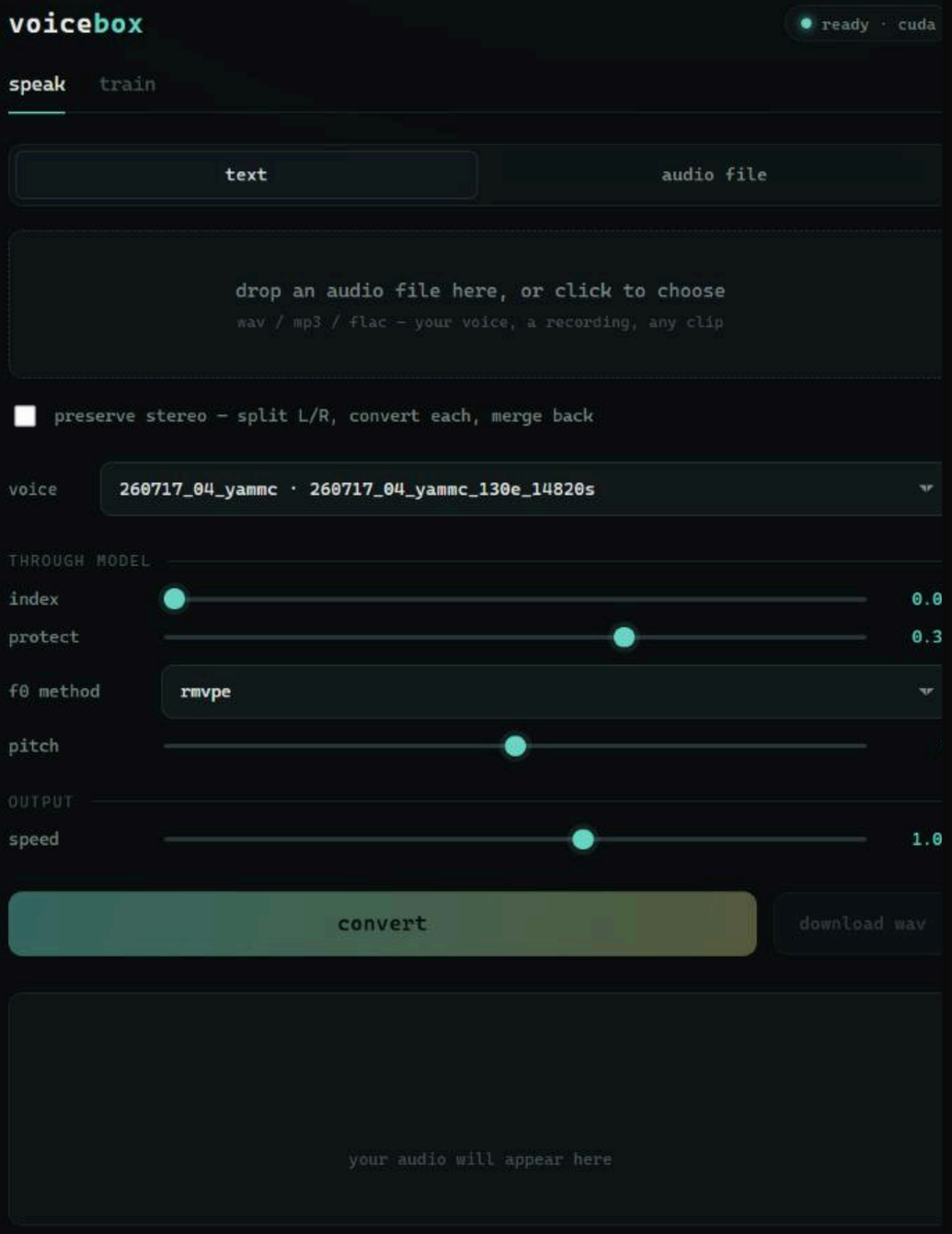
temperature: 0.80

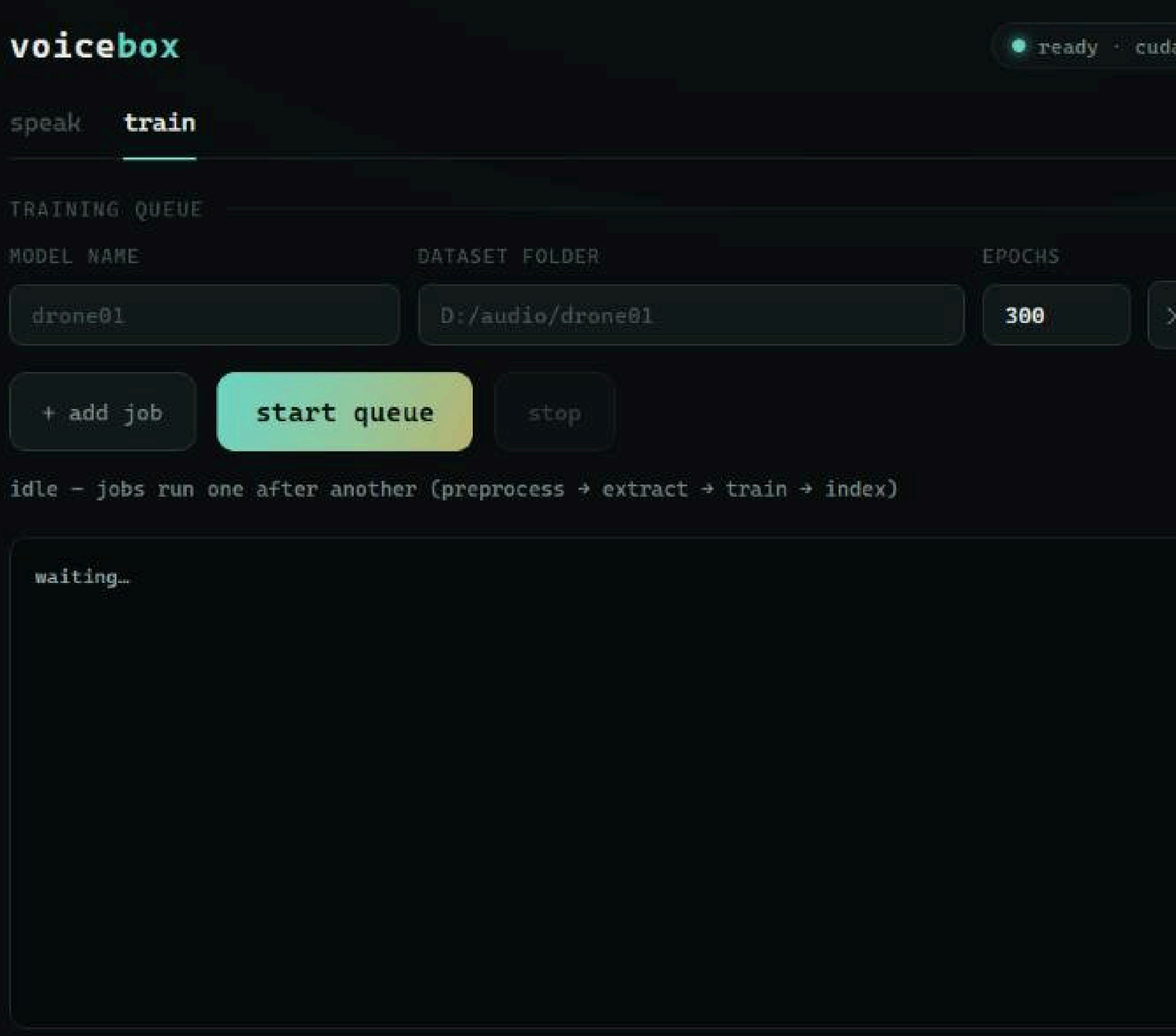
seed: 0

voice: clean — chatterbox

OUTPUT

speed: 1.00


voicebox
ready · cuda
speak
train
text
audio file
drop an audio file here, or click to choose
wav / mp3 / flac — your voice, a recording, any clip
preserve stereo — split L/R, convert each, merge back
voice
260717_04_yammc · 260717_04_yammc_130e_14820s
THROUGH MODEL
index
0.0
protect
0.3
f0 method
rmvpe
pitch
OUTPUT
speed
1.0
convert
download wav
your audio will appear here


voicebox
ready · cuda
speak
train
TRAINING QUEUE
MODEL NAME
DATASET FOLDER
EPOCHS
drone01
D:/audio/drone01
300
+ add job
start queue
stop
idle — jobs run one after another (preprocess → extract → train → index)
waiting…

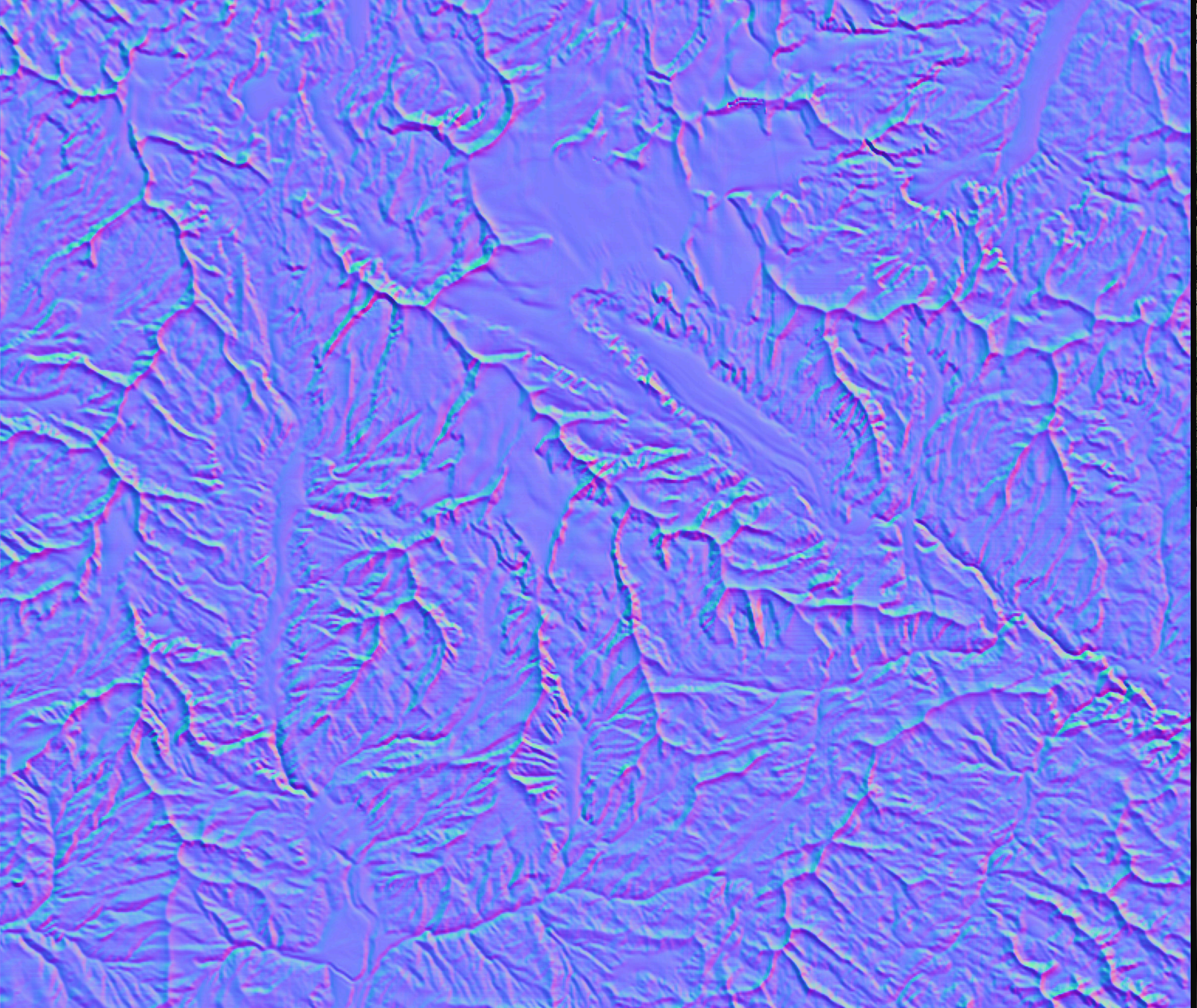
SPACE, TERRAIN AND BODY · 2026

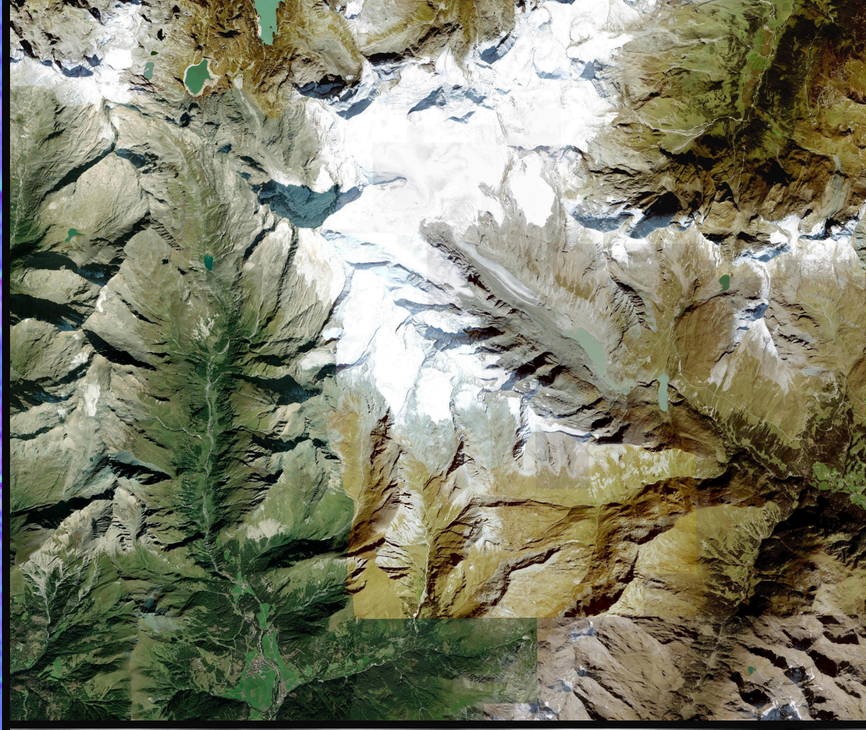
TerraForge

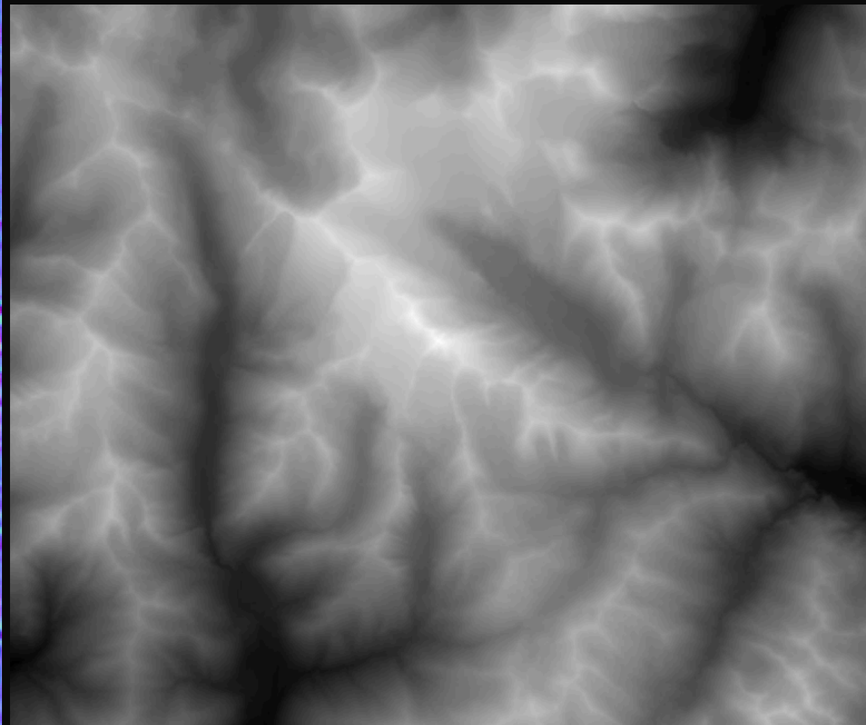
FIELD	Space, terrain and body — 3D and game-engine tools
VERSION	1.0
TECHNOLOGY	Blender add-on (Python/numpy), open elevation tiles (Terrarium), Esri/USGS/Mapbox imagery
SOURCE	terraforge.py

A Blender add-on that fetches elevation and satellite imagery for any region on earth, builds a metrically correct (un-warped Mercator) terrain mesh and derives normal, 16-bit height and ambient-occlusion maps, exported with dated textures as FBX/OBJ for game engines. Plate: Großglockner, 20 × 20 km.


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Scene
ViewLayer
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | terrain_grossglockner
Region
Region: grossglockner
Manual bbox
Latitude 47.070702
Longitude 12.693900
Zoom 12
Span (tiles) 3.00
Zoom in Zoom out
Preview region
19.99 × 19.86 km
imagery: 307...px (6.51 m/px)
DEM: 768 × 767 px (26.0 m/...
~169 tiles ·...udget 4096)
Sources & quality
Esri World I...abels, no key)
Imagery zoom 14
AWS Terrain Tiles (terrariu...
DEM zoom 12
Clear tile cache
Mesh
Match DEM pixels
Mesh resolution 768
Vertical exaggeratio 1.30
Scale 0.001000
Un-warp Mercator
Centre at origin
Shade smooth
Clamp below sea level
Smooth heightfield
Radius (px) 1.50
Amount 1.00
Maps
Normal map
Item Tool View TerraForge
Scene Collection
Collection
terrain_grossglockner
terrain_grossglockner
terrain_grossglockner
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 1.000
Y 1.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Visibility
Viewport Display
Line Art
Custom Properties
Playback Keying View Marker
1
Start 1 End 250
1 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250
4.0.2







SPACE, TERRAIN AND BODY · 2026

meanland

FIELD	Space, terrain and body — 3D and game-engine tools
VERSION	v5
TECHNOLOGY	Python, Depth Anything 3 (CUDA), browser UI, FBX export
SOURCE	meanland_v5.zip

Fuses several image sets or videos through Depth Anything 3 (depth and camera pose) into an averaged landscape heightfield or a free-standing sculpture, exported as textured FBX for Unity, and as per-frame point clouds for 4D sequences.

SETS · INPUT/

+ FOLDER + FILES

Insid vid

weight 1.00

Stephansdom Vienna 4K St. Ste… vid

weight 1.00

MODE

SCULPTURE LANDSCAPE

each set stays a body in 3d space. they are arranged, bridged with tendrils and welded into one object. orbit it in the viewer.

QUALITY

DRAFT FULL INSANE

model large

max views / set 24

video fps 2

DA3 res 896

point stride 2

conf gate 0.15

far clip % 97





a mean landscape of all depths

torch cuda

weld

h 0.10

0.035

pike 1.0

ses 12

pt res 320

9

0.04

s form

ir 24

ness 0.020

r 0.15

s

25

0.12

1.00

1

0

nity u) 100

2048

lender)



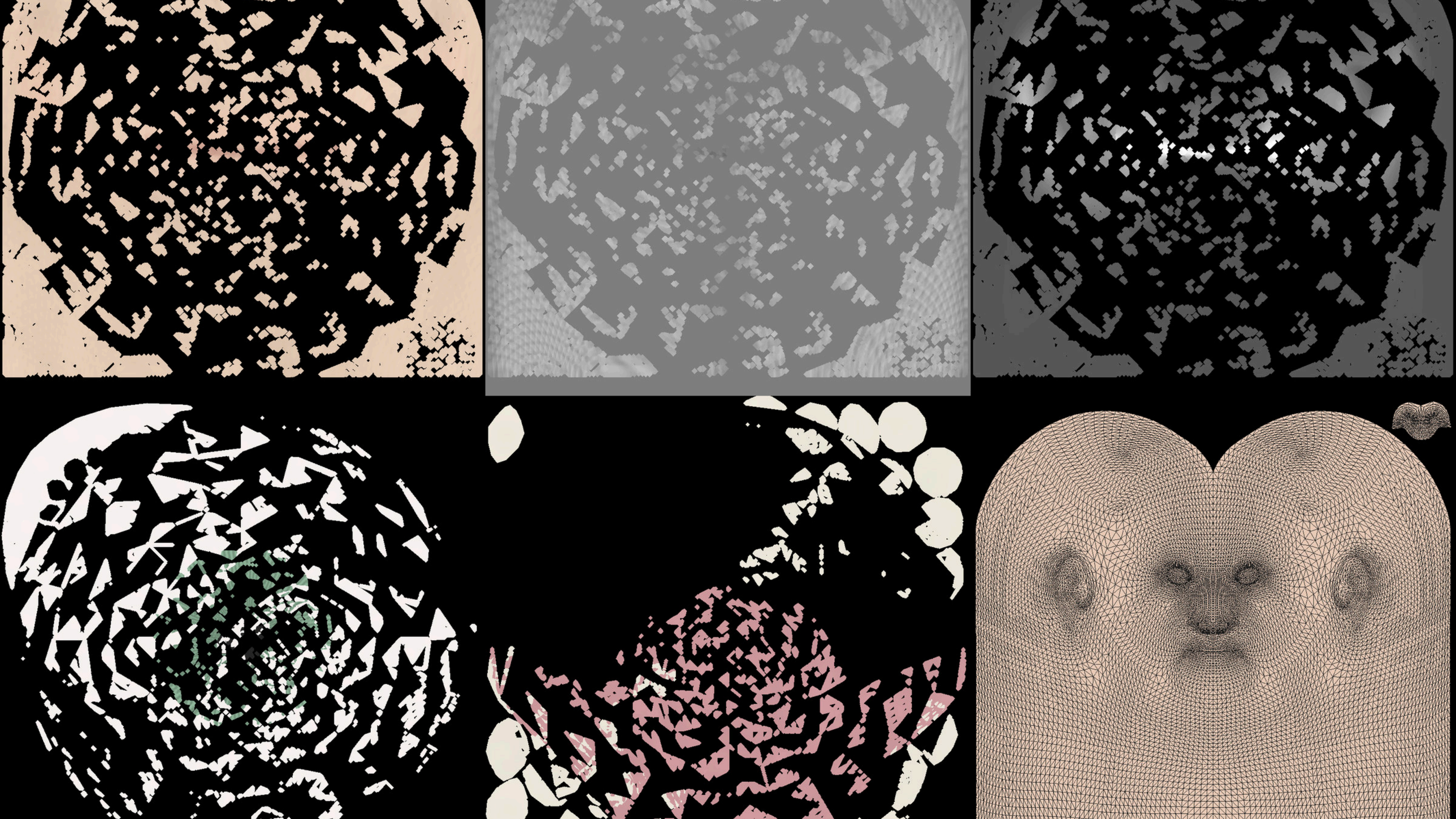
Idle

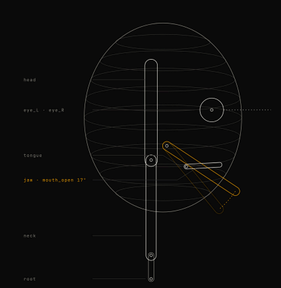
SPACE, TERRAIN AND BODY · 2026

GNM head pipeline

FIELD	Space, terrain and body — 3D and game-engine tools
VERSION	rig v0.6 · texture kit v1.0 · projector 2
TECHNOLOGY	Blender add-ons (Python/numpy), mediapipe/manual landmarks, thin-plate-spline warp; Unity HDRP
SOURCE	gnm_unity_rig.py · 260722_GNM_textureKit.zip · 260722_GNM_textureKit_projector2.zip

A pipeline for neural-generated (GNM) heads. The rig add-on classifies regions from material slots, solves the anatomical frame from eye and teeth centroids and builds a reproducible Root/Neck/Head/Jaw/Tongue/Eye rig with baked viseme and blink shape keys for Unity. The texture kit bakes position and normal maps per UV region and synthesises seed-driven skin, eyes and mouth; the projector conforms any frontal (e.g. AI-generated) face photo onto the head via landmark alignment and thin-plate-spline warping.





Resol... 2048

Analyze + Bake Maps

Procedural Skin

Seed 42

Randomize All From Seed

Eyes Mouth

Iris:

Teeth Yellow 0.15

Assign To Materials

Generate Textures

Batch (many heads)

Count 8

Out Dir: //textures/

Batch Generate To Folder

UV Template (paint / AI)

Skin

Wire Fill

Export UV Template

Photo Projection

Yaw 0.00

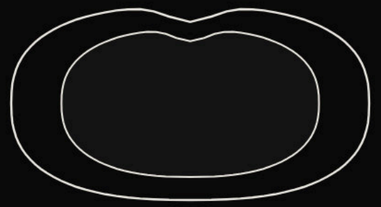
Pitch 0.00

FOV Scale 1.00

Facing Falloff 1.50

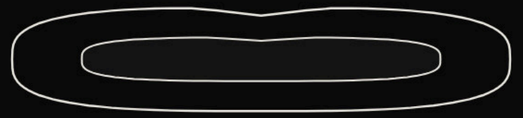
Project Photo Onto Skin

Tool View GNM Tex



viseme_AA

o 1.00 · w 0.20 · r 0.00



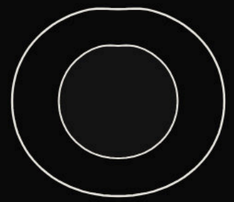
viseme_EE

o 0.25 · w 1.00 · r 0.00



viseme_IH

o 0.30 · w 0.50 · r 0.00



viseme_OH

o 0.60 · w 0.00 · r 0.80



viseme_OU

o 0.15 · w 0.00 · r 1.00



viseme_MBP

o 0.00 · w 0.00 · r 0.15



viseme_FV

o 0.15 · w 0.30 · r 0.00



viseme_L

o 0.50 · w 0.10 · r 0.00



viseme_S

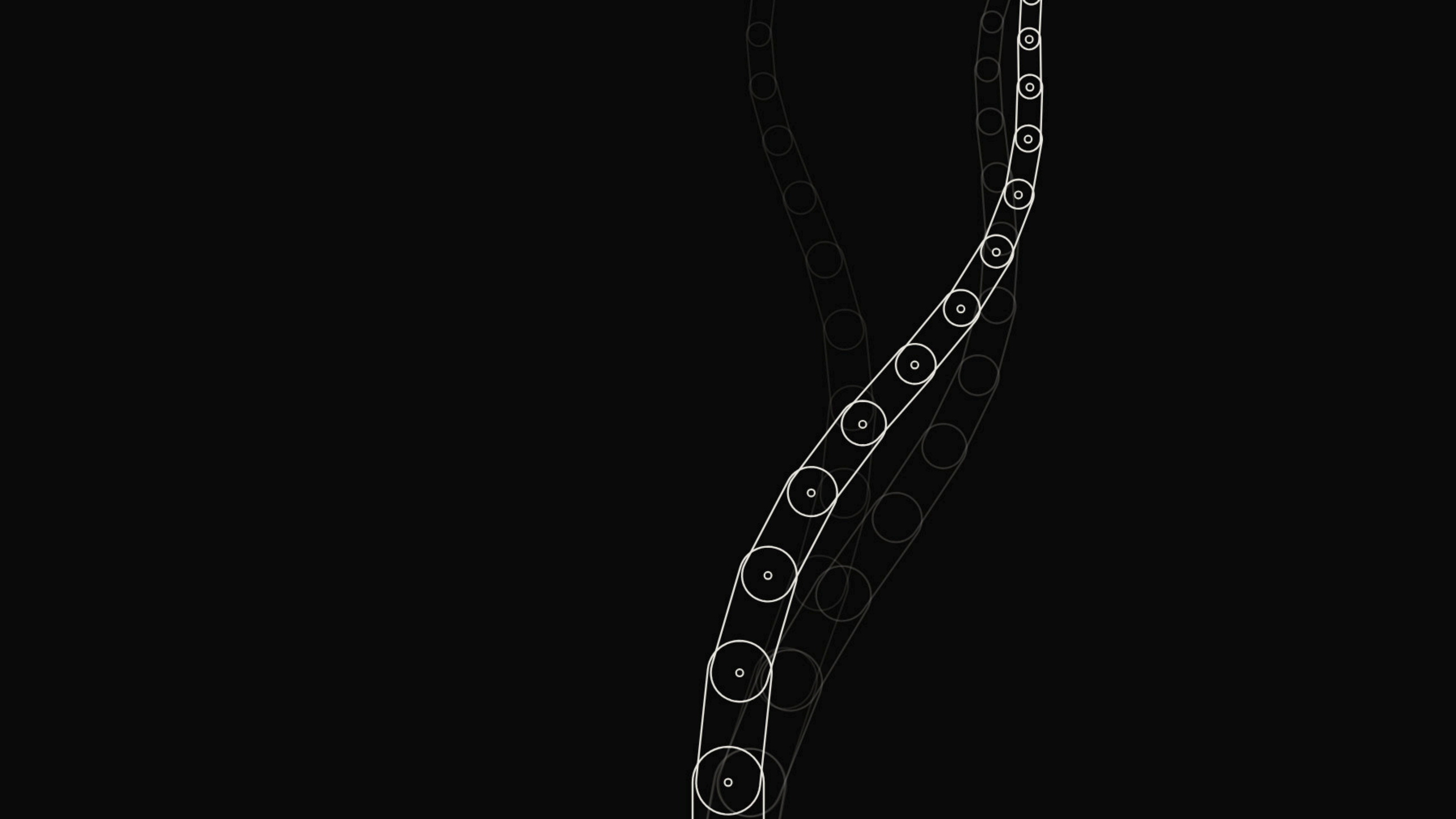
o 0.10 · w 0.60 · r 0.00

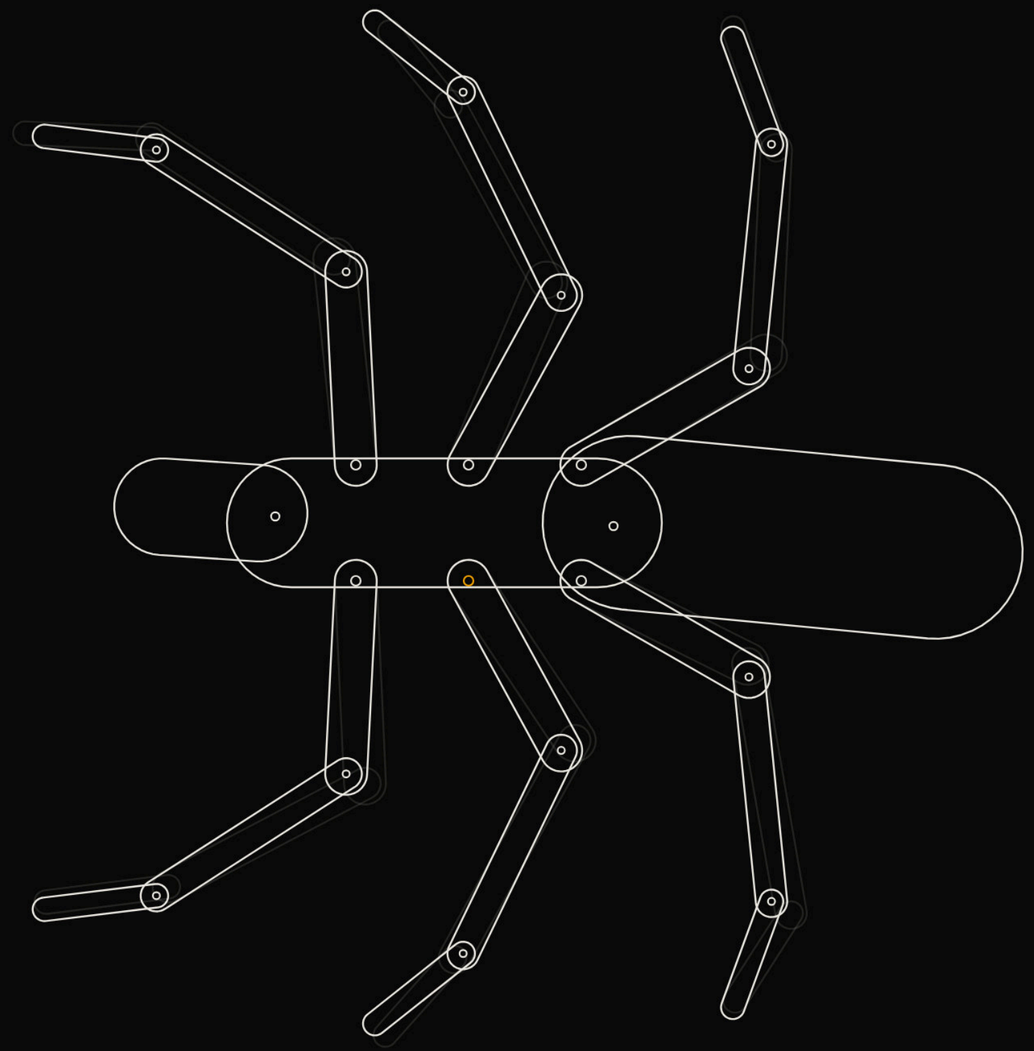
SPACE, TERRAIN AND BODY · 2026

AutoRagdoll

FIELD	Space, terrain and body — 3D and game-engine tools
VERSION	Unity 6.5
TECHNOLOGY	Unity 6.5 HDRP, C# (runtime + editor), PhysX
SOURCE	AutoRagdoll_Unity65.zip

A rig-agnostic ragdoll generator for Unity: colliders and joints are fitted from skin weights through the bind pose, so bipeds, quadrupeds, insects, tentacles and mechanical rigs work without humanoid mapping; unrigged meshes are chunked into bodies. A driver adds eight motion layers, 26 presets and audio input.





xapod · 21 bodies · no humanoid map



rigged mesh · 18 chunks · fitted joints

PERFORMANCE CONTROL · 2026

Resonode

FIELD	Performance control
TECHNOLOGY	Node.js OSC bridge, HTML interface; Resolume Avenue 6
SOURCE	Resonode.zip

An OSC control surface for Resolume built for slow performance: long fades, few gestures, nothing cutting that should glide. The fade engine runs in the bridge rather than the browser, so a twelve-minute crossfade keeps its exact curve even if the interface is closed; parameters, soft triggers, clip grids and slow sequences are arranged as nodes, with cues and remote control.


RESONODE
bridge
to 127.0.0.1 7000 from 7001 send 30 Hz
time 1.00×
colours spectrum
Stop all
Black
Float
Pin
Presets four layers
four layers
Load
name
Save
keep current IP & ports when loading
Export
Import
Cues
Capture
01 black wait
02 layer 1 alone wait
03 one and two 30s
04 everything up wait
GO space
Stop
no cue running
Remote
Let another machine drive this one: point Max, Reaktor or another sender at the "from" port and map an address here.
master MST float
/composition/master
0.000
TRIGGER to 1 in 8 s sineInOut
layer 1 → next clip L1 C1 soft
/composition/layers/1/clips/1/connect
dip a fader
layer 1
FADE INTO IT
out 12 sineInOut hold 0.3
in 12 sineInOut from 0 to 1
...and out again
clips L1–
layers 1 – 4 clips 1 – 8
fade the clip in over 6 s shift-click does the other c
L1 1 2 3 4 5
L2 1 2 3 4 5
L3 1 2 3 4 5
L4 1 2 3 4 5
layers
end · fade 20.0s go
layer 1 L1 → layer 2 float
/composition/layers/1/video/opacity
0.000
TRIGGER to 1 in 20 s sineInOut
hand · cut +2s go
layer 2 L2 → layer 3 float
/composition/layers/2/video/opacity
0.000
TRIGGER to 1 in 20 s sineInOut
layer 3 L3 float
/composition/layers/3/video/opacity
0.000
TRIGGER to 1 in 20 s sineInOut
layer 4 L4 float
/composition/layers/4/video/opacity
0.000
TRIGGER to 1 in 20 s sineInOut
slow sequence
START
6 clips
stop when faded
hold 180 s · crossfade 90 s equal power
01 L 1 C 1
02 L 2 C 1
03 L 3 C 1
04 L 1 C 2
05 L 2 C 2
06 L 4 C 2
+ clip
one pass ≈ 21:00 + 2 waiting for you
Send 5
/composition/…
1 float Send
Check every address
Asks Resolume about every address on the canvas and marks the ones it doesn't answer on. Needs OSC Output to port 7001. For an exact address: Shortcuts → Edit OSC in Resolume, then click the control.
Incoming
listening for OSC on 7001
loaded preset "four layers"
OSC out → 127.0.0.1:7000 · rate 30 Hz
+ parameter
+ soft trigger
+ clip grid
+ clip chain
+ frame
Tidy
Fit
100%
drag a header to move · drag a right-hand port onto another node to hand off · a add · f fit



Resonode

An OSC control surface for Resolume Avenue 6, built for slow work: long fades, few gestures, nothing cutting glide.

What it is

Two files do the work. `bridge.js` is a small server that talks OSC to Resolume and serves the interface interface, which you use in a browser. There is nothing to install beyond Node.js — no npm packages, connection needed at the venue.

The one thing worth understanding: the fade engine runs in the bridge, not in the browser.

Browsers throttle timers in background tabs, which would visibly wreck a ten-minute fade. Here the bro view. A twelve-minute crossfade keeps its exact shape whether the window is in front, hidden, or close can shut the browser mid-fade and the fade finishes. Open it again and you rejoin a show already in pr devices — laptop and phone — stay in sync on the same state.

Install and run · Windows

`Windows 10 / 11`

1. Install **Node.js** from nodejs.org — the LTS build. Accept the defaults.
2. Put the `resonode` folder somewhere sensible (Documents is fine — not inside a zip, and not in Prog
3. Double-click **start.bat**.

A black console window appears and stays open — that is the engine; leave it running. The interface o automatically at `http://127.0.0.1:8962`, in a chromeless Chrome or Edge window if you have one.

To stop: close the interface window, then press `Ctrl` + `C` in the console, or just close the console.

If Windows Defender asks about the firewall

Allow it on private networks. Sending to Resolume works either way, but blocking it stops OSC *coming* Resolume, which quietly disables clip names, playing-clip highlighting and the address checker.

If the port is busy

Open a command prompt in the folder and start it on another port:

```
node bridge.js --ui-port 8963
```

THOMASWAGENSOMMERER.XYZ