MAT 265 FALL 2025

From Attention Visualization to Step-wise Analysis in Stable Diffusion

PRESENTED BY:

SHAW YIRAN XIAO
JINTONG YANG
SPIKE RAO

(1) MOTIVATION
(2) BACKGROUND
(3) ANALYSIS PIPELINE
(4) ATTENTION MAPS
(5) RESEARCH DIRECTION
(6) ART VISUALIZATION

MOTIVATION

Why do we care about interpreting diffusion models?

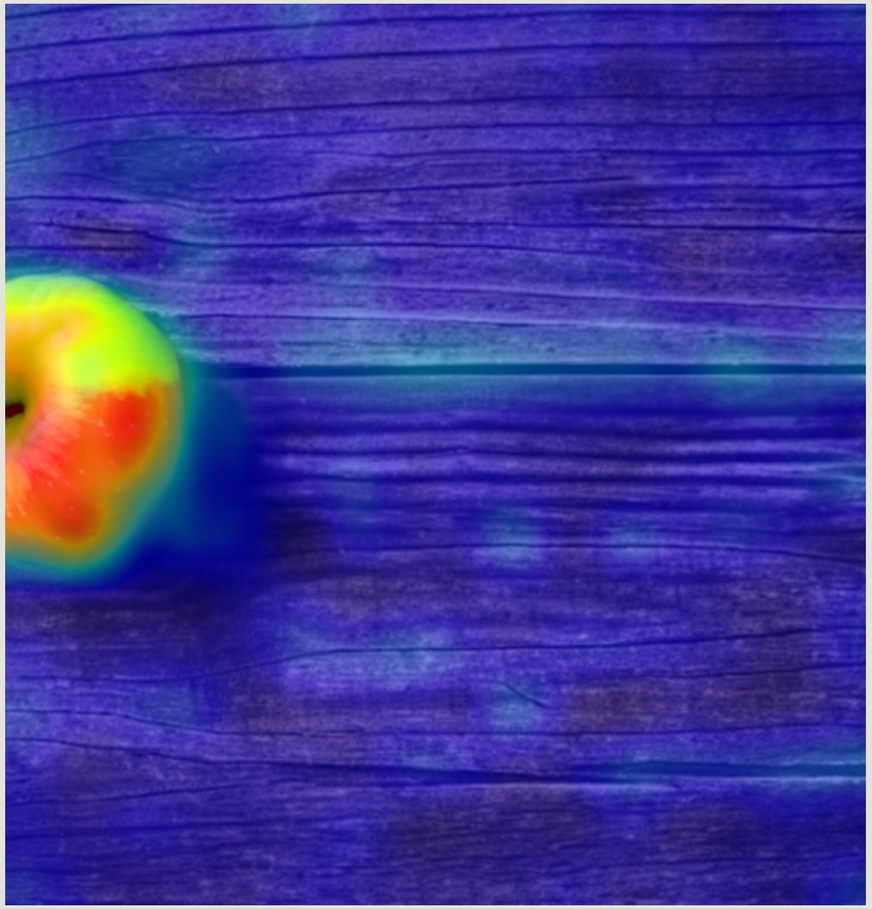
- Powerful, but highly opaque “prompt → image” process
- Hard to answer: What does the model look at? When are objects formed?

Limitations of existing work

- Many tools show static attention maps for a single final image
- Step-wise dynamics and object-level behavior are underexplored

Overall goal

- Build a practical analysis pipeline to:
- Visualize cross- and self-attention
- Track step-wise evolution of attention
- Enable systematic studies of how and when different concepts emerge



(1)

BACKGROUND

DAAM

- From “What the DAAM: Interpreting Stable Diffusion Using Cross Attention”
- Visualizes cross-attention (text → image) in Stable Diffusion
- For each word: where does the model focus in the image?

DAAM-I2I

- Extends attention analysis to self-attention (image ↔ image)
- Pixel-, box-, and contour-based self-attention heatmaps
- Reveals internal structure and relationships inside the image

TITAN

- Toolkit for object-level prompt analysis and annotations
- Extracts objects from prompts and produces COCO-style annotation structures

ANALYSIS PIPELINE

Input

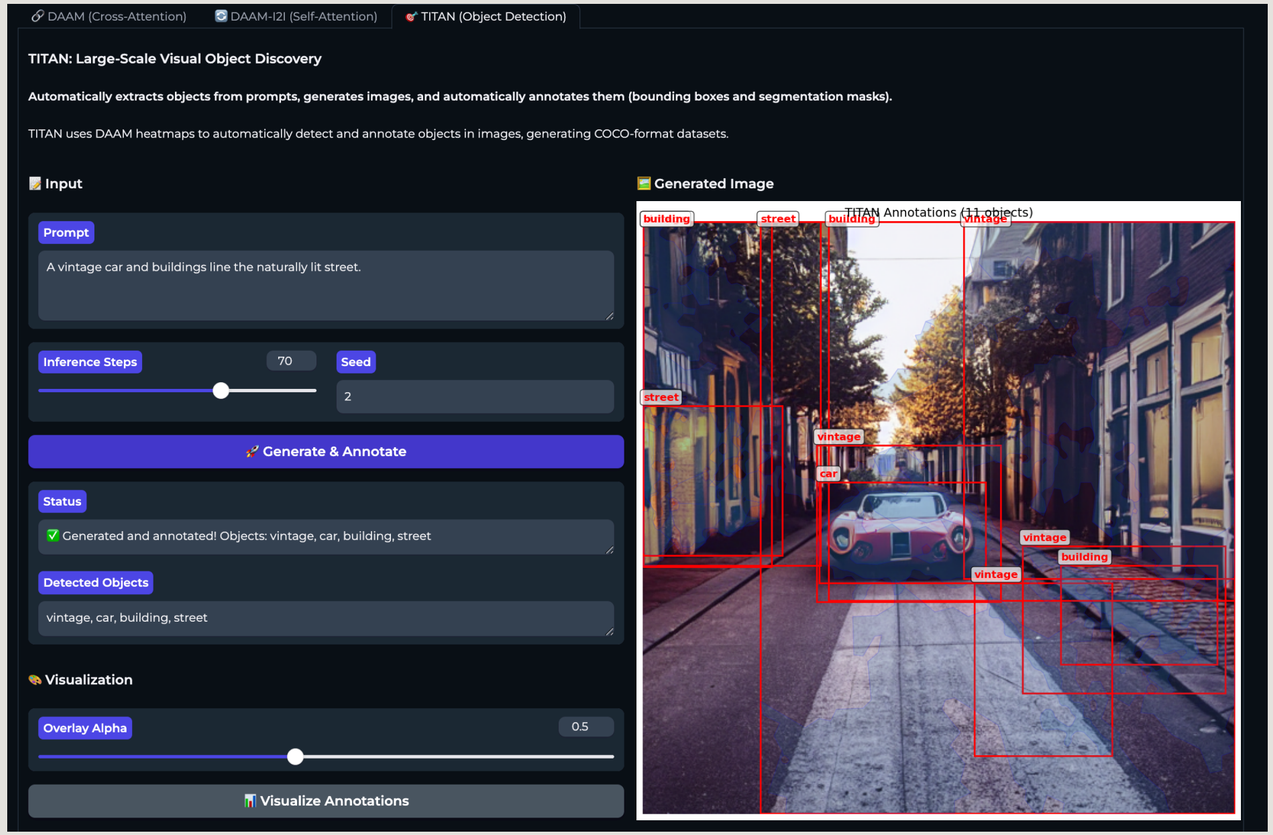
- Text prompt
- Stable Diffusion model (v1.5)
- Core analysis steps
- Generate image with full access to internal attention

Collect:

- Cross-attention maps over steps (DAAM)
- Self-attention maps in latent/image space (DAAM-I2I)
- Object lists and annotations (TITAN)

Outputs

- Word-specific and global attention heatmaps
- Step-wise attention trajectories for each token/object
- Object-level statistics and visualizations



(3)

OVERALL ANALYSIS PIPELINE

CROSS-ATTENTION: WORD-TO-IMAGE MAPS

Capabilities

- Generate images while recording cross-attention at every diffusion step
- For a selected word (or “Global”):

1. Aggregate across layers and heads
2. Upsample to image resolution
3. Overlay as a heatmap

What this lets us see

- Which regions each word attends to
- How attention focuses or diffuses across steps
- Global vs word-specific attention patterns

(4)

SELF-ATTENTION: IMAGE-INTERNAL STRUCTURE

Self-attention views (DAAM-I2I)

- Pixel-based heatmaps
  -Single latent pixel or range of pixels
- BBox-based heatmaps
  -Attention from a selected latent box
- Contour-based heatmaps
  -Attention from an arbitrary drawn contour
- Diffused pixel attention
  -Iterative expansion from a seed pixel

Interpretation

- Which regions form coherent structures?
- How do different parts of the image “talk” to each other?

(4)

OBJECT-LEVEL SEMANTICS VIA TITAN

Prompt understanding

- PromptHandler extracts object candidates from text
- Cleaned prompts and object lists (e.g., [cat, chair, window])

Annotation structure

- TITANDataset stores:

1. Image IDs, filenames
2. Object annotations (COCO-style)
3. Links to global attention maps

Why this matters

- Bridges token-level attention and object-level analysis
- Enables quantitative, dataset-like studies

(4)

GENERATION TIMELINES OF OBJECTS

Central research question

- During denoising, when do different semantic elements emerge?

1. Background vs main objects vs small details
2. Attributes (color, style) vs nouns (objects)

Why it’s interesting

- Reveals the model’s “drawing order” and internal decision process
- Fills a gap: most prior work shows static attention, not dynamics
- Can inform where to intervene for control or editing

(5)

STEP-WISE ATTENTION

Purpose

- Systematically export per-step cross-attention maps for analysis

Key design

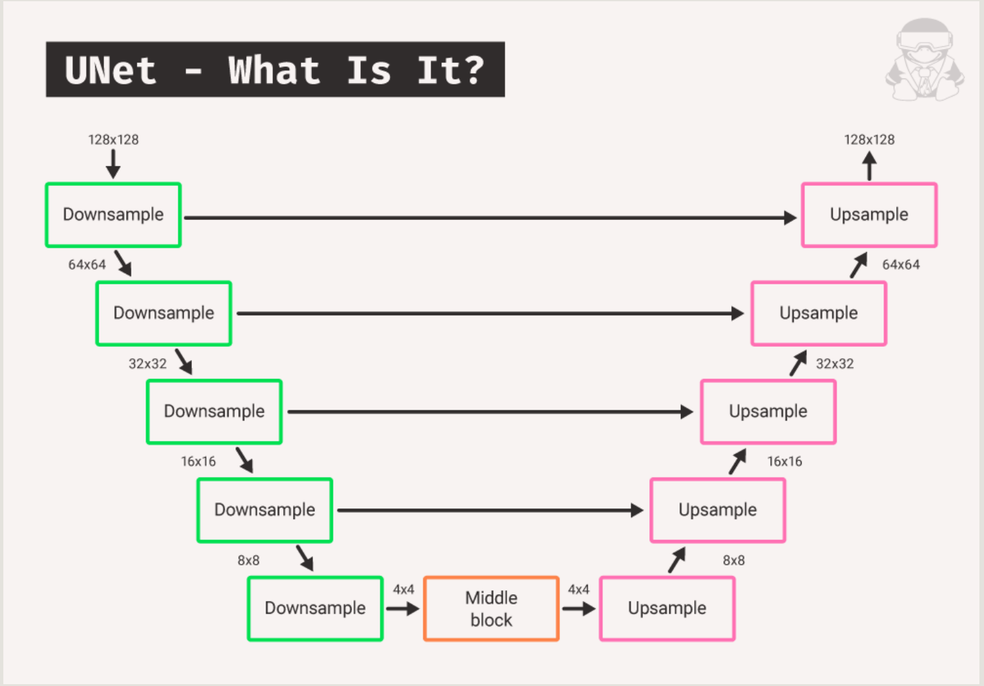
- Hook the UNet forward to track denoising steps
  -UNetForwardHook assigns a stable step_idx to each call

- Hook DAAM's internal update
  -StepAwareUpdateHook stores maps as {step → layer → head}

Outputs

- For each step:

1. Aggregated cross-attention map (tokens × H × W)
2. Word-specific heatmaps over the image (PNG files)
3. Global attention maps for comparison



(5)

Experiments and Impact

Planned analyses

- Prompt templates:

1. Multi-object: a cat and a dog in a living room
2. Attribute–noun: a red car in a watercolor painting

- For each word/object:

1. Compute attention mass curve over steps
2. Identify peak or stabilization step ranges

Expected outcomes

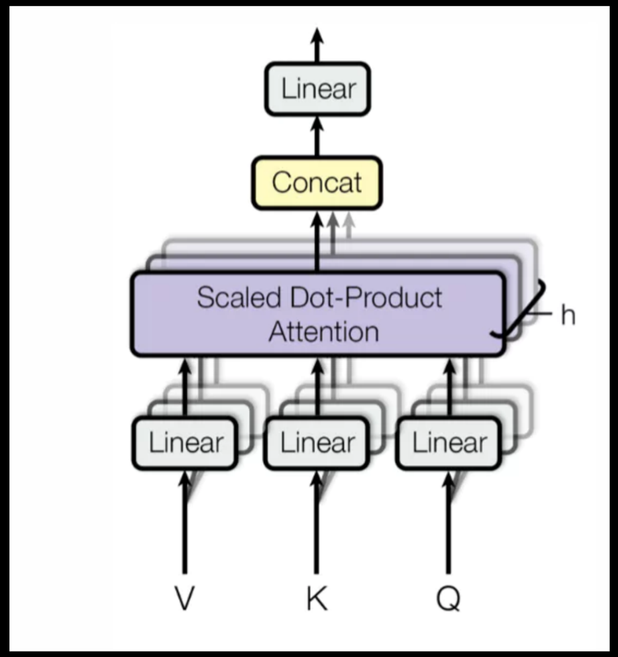
- Empirical patterns of:

1. Background vs foreground emergence
2. Object vs attribute timing

- Practical implications:

1. Guides when to apply constraints for controllable generation
2. Provides a solid interpretability study on diffusion dynamics

MODEL



HOOK ALL CROSS-ATTENTION Q AND K VECTORS

-Q (QUERY) = "WHAT AM I LOOKING FOR?"
-K (KEY) = "WHAT INFORMATION DO I HAVE?"
-V (VALUE) = "WHAT DATA SHOULD I RETRIEVE IF I MATCH?"

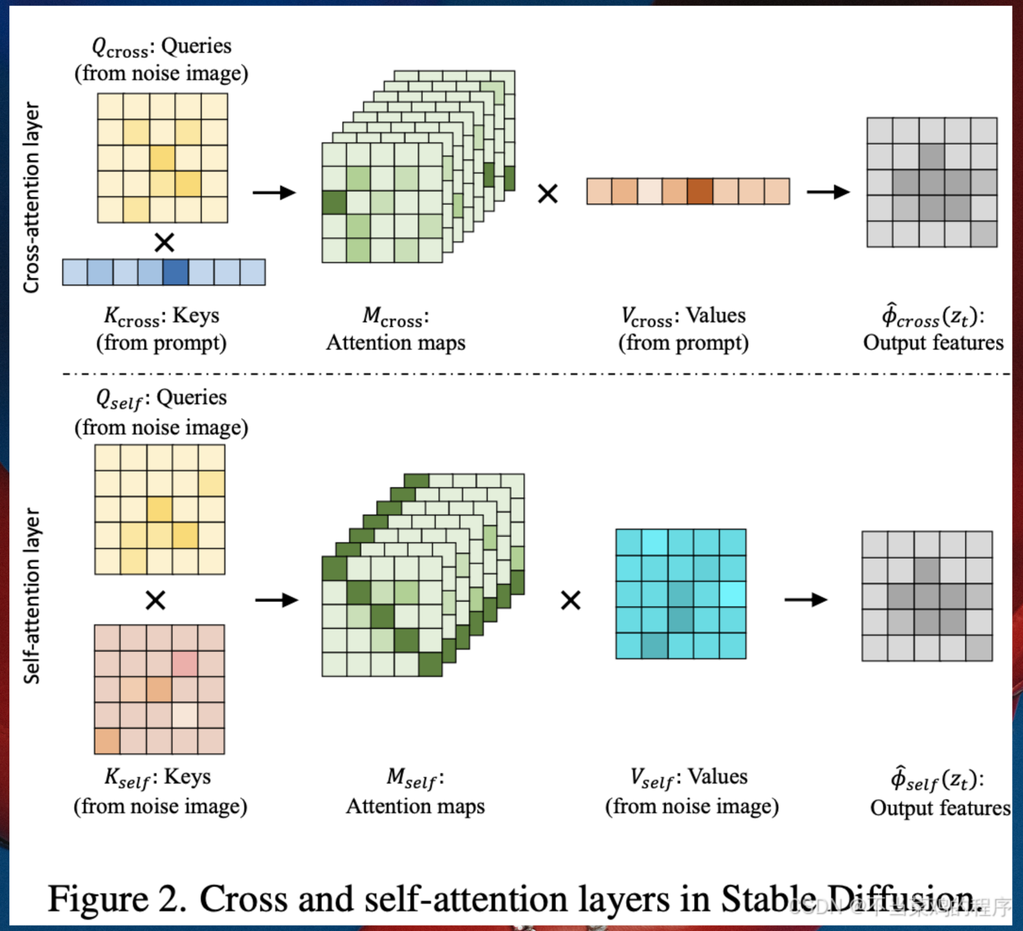
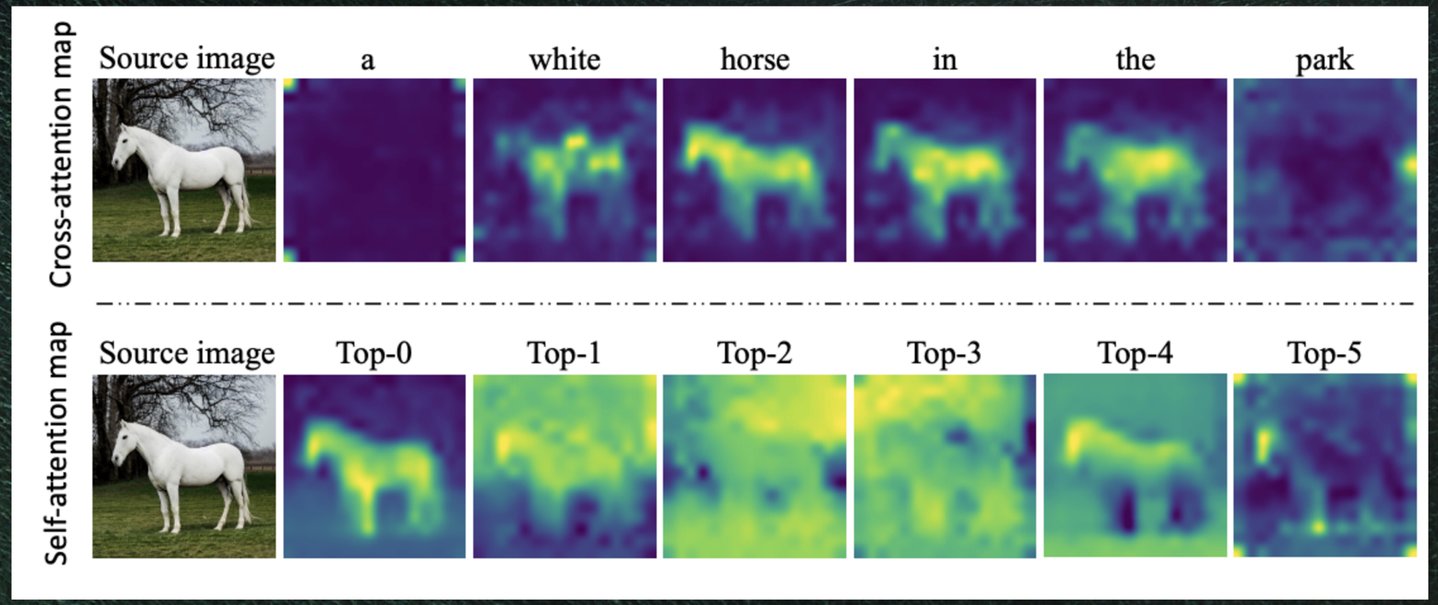


Figure 2. Cross and self-attention layers in Stable Diffusion.

ATTENTION MAP EXPERIMENT

SELF- & CROSS-ATTENTION TRACING, HEATMAP ANALYSIS



SPIKE RAO

CHALLENGE

- SELF-ATTENTION VISUAL CLARITY:
which heatmap types are most readable?

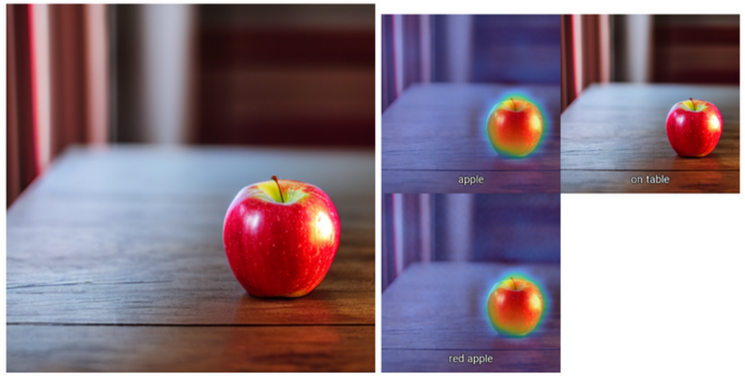
- INITIALLY MIS-HOOKED CROSS-ATTENTION IN THE UNET :
noisy / meaningless heatmaps

- DEBUG & FIX:
traced correct UNet tensors, normalize & aggregate properly (thanks to Shaw)

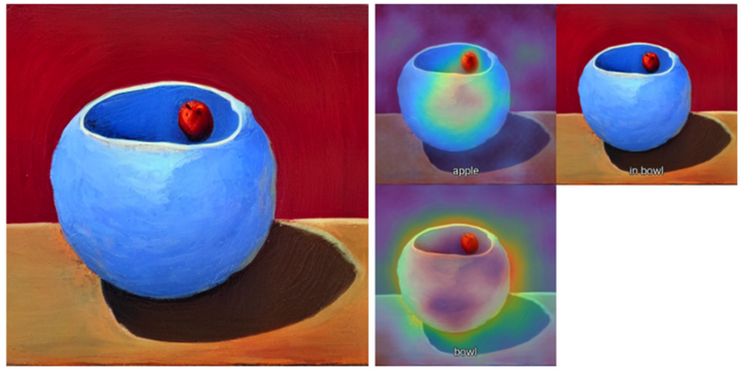
- ONGOING:
raw attention tensors are large; we reduced cost by aggregating heads, saving float16, and storing selected steps.

DAAM (original) — cross-attention

Maps which words influence which pixels



- a red apple on a wooden table, morning light



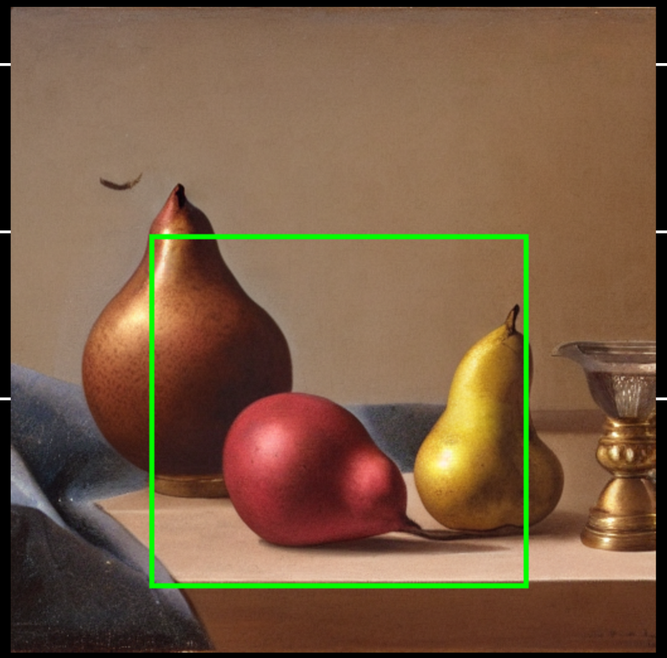
- a red apple in a blue ceramic bowl, still life

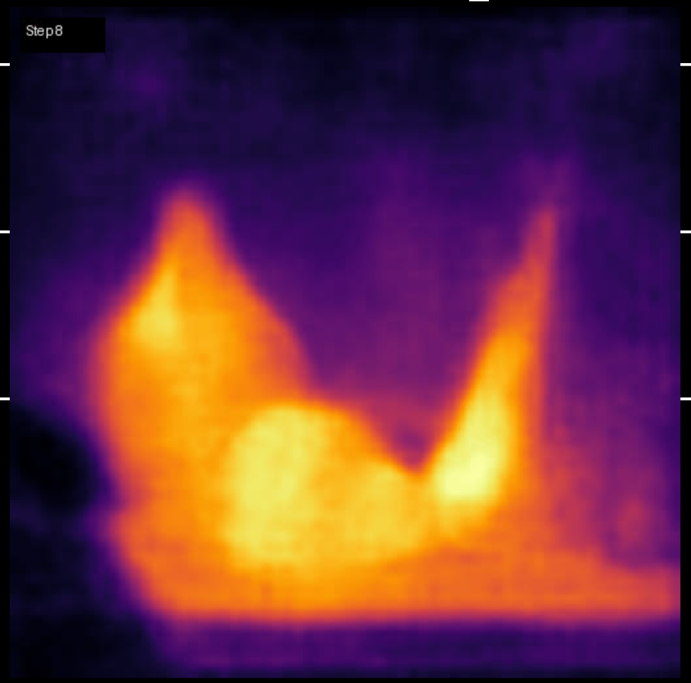
- a red apple under a cloth, partially revealed

KEY FINDING: PREPOSITIONS HAVE WEAK EFFECT ON OBJECT PLACEMENT; DAAM OFTEN SHOWS LITTLE ACTIVATION FOR THEM

DAAM-I2I — UNet self-attention tracing

Maps how image positions attend to each other over steps





EXPRIMENT

=SELF-ATTENTION

PROMPT：**A STILL LIFE OF PEARS AND A BRASS PITCHER ON A WOODEN TABLE, NATURAL WINDOW LIGHT**

STEP-WISE(20/20):

EARLY → MID → LATE

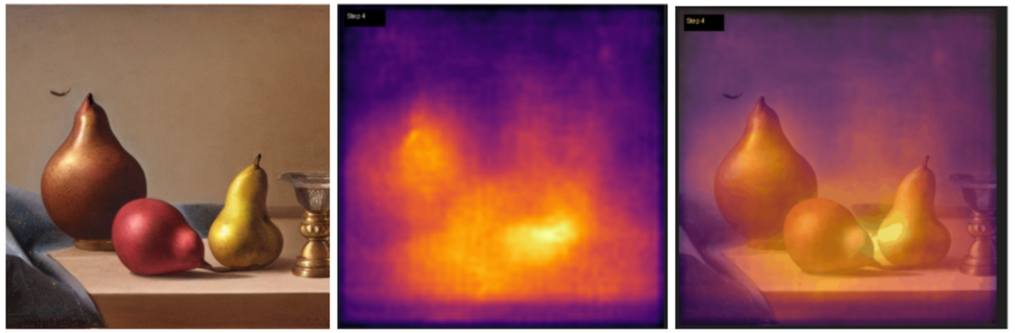
KEY FINDING:

BBOX AGGREGATION ARE MOST INTERPRETABLE

IT CLEARLY SHOWS THAT THE AREAS HIGHLIGHTED IN THE IMAGE INDEED RECEIVED MORE ATTENTION

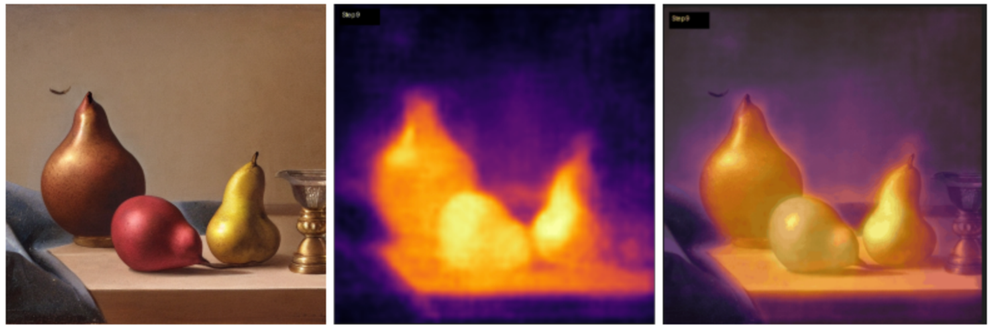
WHEN YOU SEE ATTENTION FOCUSING ON A REFLECTION, IT'S BECAUSE THE MODEL IS DRAWN TO THAT HIGH-CONTRAST SIGNAL WHILE IT'S STILL FIGURING OUT THE HIGHLIGHTS—THOSE PIXELS INFLUENCE THE REST OF THE IMAGE MORE STRONGLY THAN SMOOTHER, MATTE REGIONS.

Generated image(left) raw heatmap(middle) Overlayed image(right)



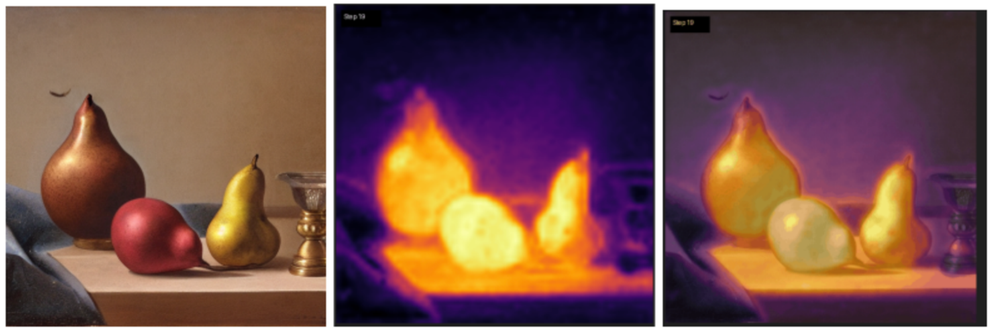
(step4/20)

light, scattered highlights — the model is sketching broadly, testing many possibilities.



(step9/20)

certain regions grow brighter — the model starts selecting key objects and positions (composition emerges)



(step19/20)

bright, focused areas and fine structure — details, textures and reflections are being finalized.

ATTENTION MAP VISUALIZATION

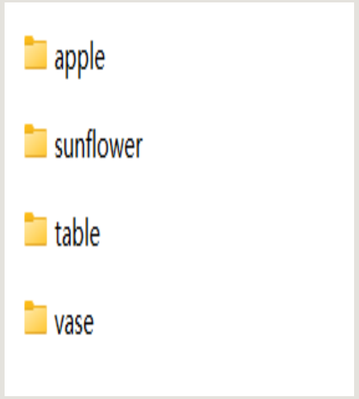
JINTONG YANG

MAIN CHALLENGES

- Handling large attention data while keeping the system efficient
- Updating many layers of attention maps in real time
- Designing a simple and stable way to map attention scores to particle behavior

PREVIOUS METHOD

- Used attention scores
- Hard to translate into images
- Unclear & indirect visual output

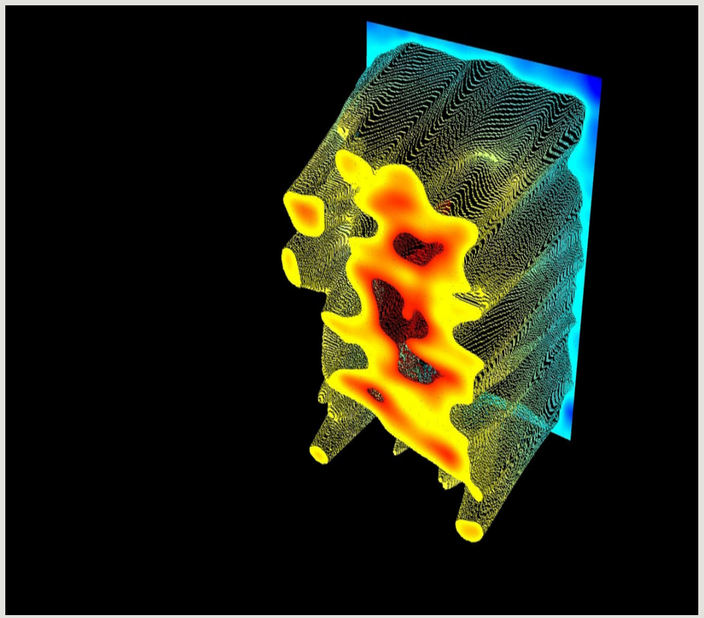


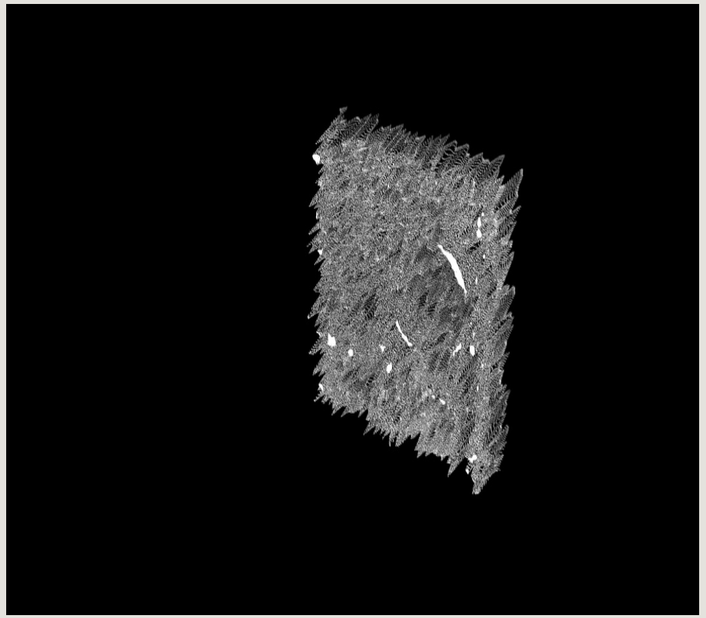
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	0.00548	0.00548	0.00548	0.00548	0.00556	0.00573	0.00589	0.00606	0.00622	0.00639	0.00655	0.00672	0.00668	0.00643	0.00619	0.00594	0.0057	0.005
2	0.00548	0.00548	0.00548	0.00548	0.00556	0.00573	0.00589	0.00606	0.00622	0.00639	0.00655	0.00672	0.00668	0.00643	0.00619	0.00594	0.0057	0.005
3	0.00548	0.00548	0.00548	0.00548	0.00556	0.00573	0.00589	0.00606	0.00622	0.00639	0.00655	0.00672	0.00668	0.00643	0.00619	0.00594	0.0057	0.005
4	0.00548	0.00548	0.00548	0.00548	0.00556	0.00573	0.00589	0.00606	0.00622	0.00639	0.00655	0.00672	0.00668	0.00643	0.00619	0.00594	0.0057	0.005
5	0.00545	0.00545	0.00545	0.00545	0.00553	0.00568	0.00583	0.00599	0.00614	0.00629	0.00645	0.0066	0.00656	0.00633	0.0061	0.00588	0.00565	0.005
6	0.00539	0.00539	0.00539	0.00539	0.00545	0.00558	0.00571	0.00584	0.00597	0.0061	0.00623	0.00636	0.00632	0.00613	0.00594	0.00574	0.00555	0.005
7	0.00533	0.00533	0.00533	0.00533	0.00538	0.00549	0.00559	0.0057	0.0058	0.00591	0.00601	0.00612	0.00609	0.00593	0.00577	0.00561	0.00545	0.005
8	0.00527	0.00527	0.00527	0.00527	0.00531	0.00539	0.00547	0.00555	0.00563	0.00571	0.00579	0.00587	0.00585	0.00573	0.0056	0.00548	0.00535	0.005
9	0.00521	0.00521	0.00521	0.00521	0.00523	0.00529	0.00535	0.00541	0.00546	0.00552	0.00558	0.00563	0.00562	0.00553	0.00543	0.00534	0.00525	0.005
10	0.00514	0.00514	0.00514	0.00514	0.00516	0.00519	0.00523	0.00526	0.00529	0.00533	0.00536	0.00539	0.00538	0.00532	0.00527	0.00521	0.00515	0.00
11	0.00508	0.00508	0.00508	0.00508	0.00509	0.0051	0.00511	0.00511	0.00512	0.00513	0.00514	0.00515	0.00514	0.00512	0.0051	0.00508	0.00505	0.005
12	0.00502	0.00502	0.00502	0.00502	0.00501	0.005	0.00498	0.00497	0.00495	0.00494	0.00492	0.00491	0.00491	0.00492	0.00493	0.00494	0.00496	0.004
13	0.00501	0.00501	0.00501	0.00501	0.005	0.00497	0.00495	0.00492	0.0049	0.00487	0.00485	0.00482	0.00483	0.00486	0.00489	0.00492	0.00495	0.004
14	0.00505	0.00505	0.00505	0.00505	0.00504	0.00502	0.005	0.00498	0.00496	0.00494	0.00492	0.0049	0.0049	0.00493	0.00497	0.005	0.00503	0.005
15	0.00509	0.00509	0.00509	0.00509	0.00508	0.00506	0.00505	0.00503	0.00501	0.005	0.00498	0.00497	0.00498	0.00501	0.00505	0.00508	0.00512	0.005
16	0.00513	0.00513	0.00513	0.00513	0.00512	0.00511	0.0051	0.00508	0.00507	0.00506	0.00505	0.00504	0.00505	0.00509	0.00513	0.00516	0.0052	0.005
17	0.00517	0.00517	0.00517	0.00517	0.00516	0.00515	0.00515	0.00514	0.00513	0.00512	0.00511	0.00511	0.00512	0.00516	0.0052	0.00525	0.00529	0.005
18	0.00521	0.00521	0.00521	0.00521	0.0052	0.0052	0.0052	0.00519	0.00519	0.00518	0.00518	0.00518	0.0052	0.00524	0.00528	0.00533	0.00537	0.005
19	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00525	0.00527	0.00532	0.00536	0.00541	0.00546	0.00
20	0.00529	0.00529	0.00529	0.00529	0.00529	0.00529	0.0053	0.0053	0.0053	0.00531	0.00531	0.00532	0.00534	0.00539	0.00544	0.00549	0.00554	0.005
21	0.00527	0.00527	0.00527	0.00527	0.00528	0.00529	0.0053	0.00531	0.00531	0.00532	0.00533	0.00534	0.00537	0.0054	0.00544	0.00548	0.00552	0.005
22	0.00521	0.00521	0.00521	0.00521	0.00522	0.00524	0.00525	0.00526	0.00528	0.00529	0.0053	0.00532	0.00533	0.00535	0.00537	0.00538	0.0054	0.005
23	0.00515	0.00515	0.00515	0.00515	0.00516	0.00518	0.0052	0.00522	0.00524	0.00526	0.00527	0.00529	0.0053	0.00529	0.00529	0.00528	0.00528	0.005

apple_step_49

PREVIOUS TESTS

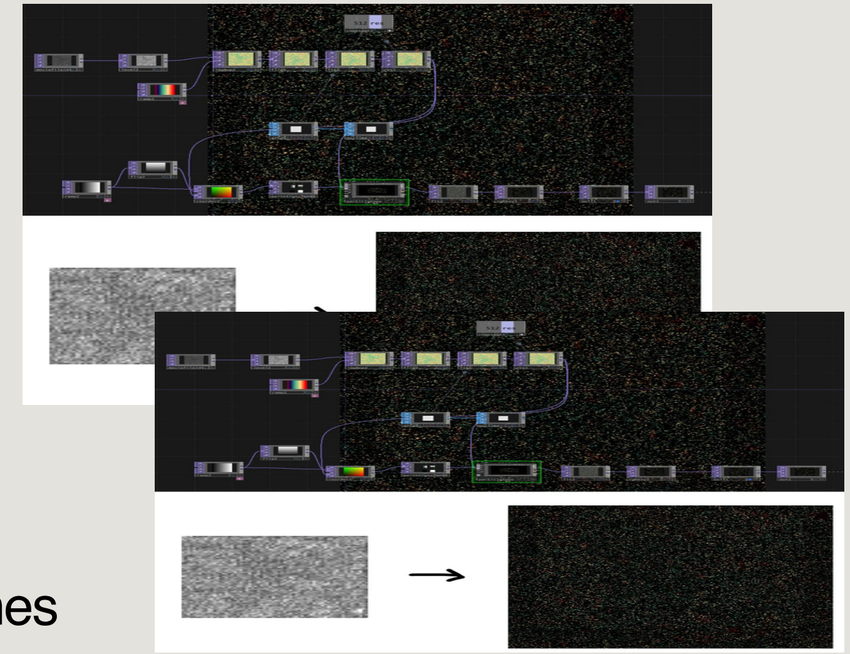
- Used attention scores to visualize the “height” of the focused area.
- The visual effect was not very clear.

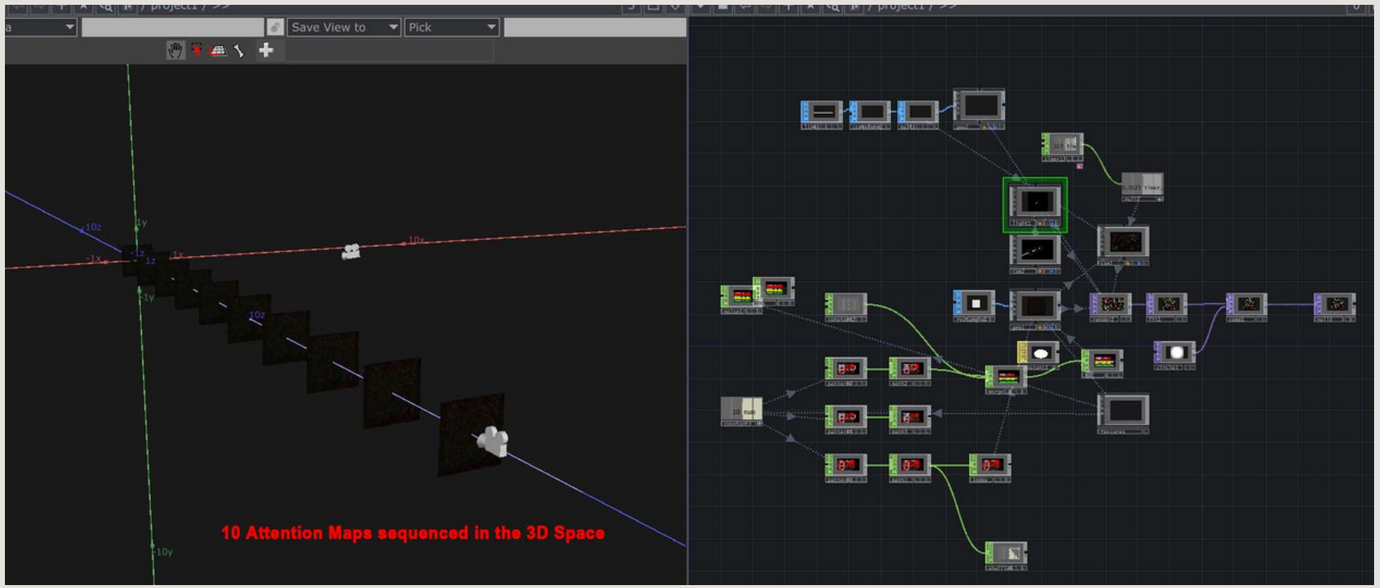


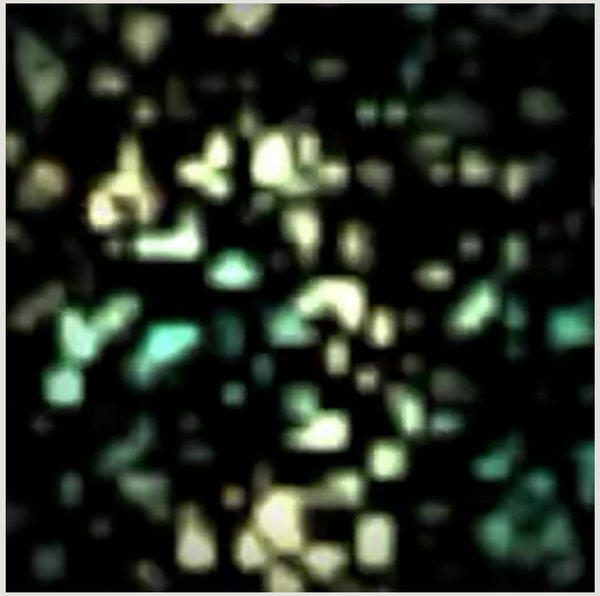


PREVIOUS TESTS

- List attention maps in 3D space
- Camera moves through attention layers
- Each layer as particles forming an image
- Problems: heavy GPU load → laggy;
- TD cannot render 3D textures → images flatten into 2D planes







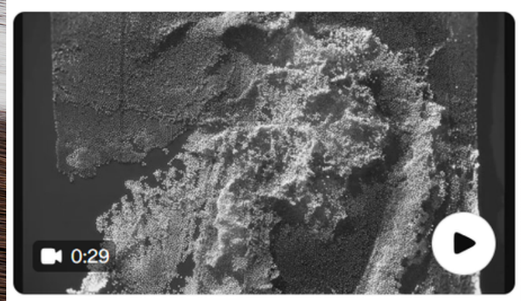
INSPIRATION

[TOE] POP Trails Portraits

Today, I share with you a patch to create a messy pointcloud and particles from any single photo....

[TOE] Webcam Optical Flow 2025

Hello I've been away from my computer for the last few days, but, here it is, the Touchdesigner...



[TOE] FLEX Liquid Webcam

Hello everyone. Today, let's make waves together. A touchdesigner project, based on optical flow...

Bertrand de BECQUE (B2BK)

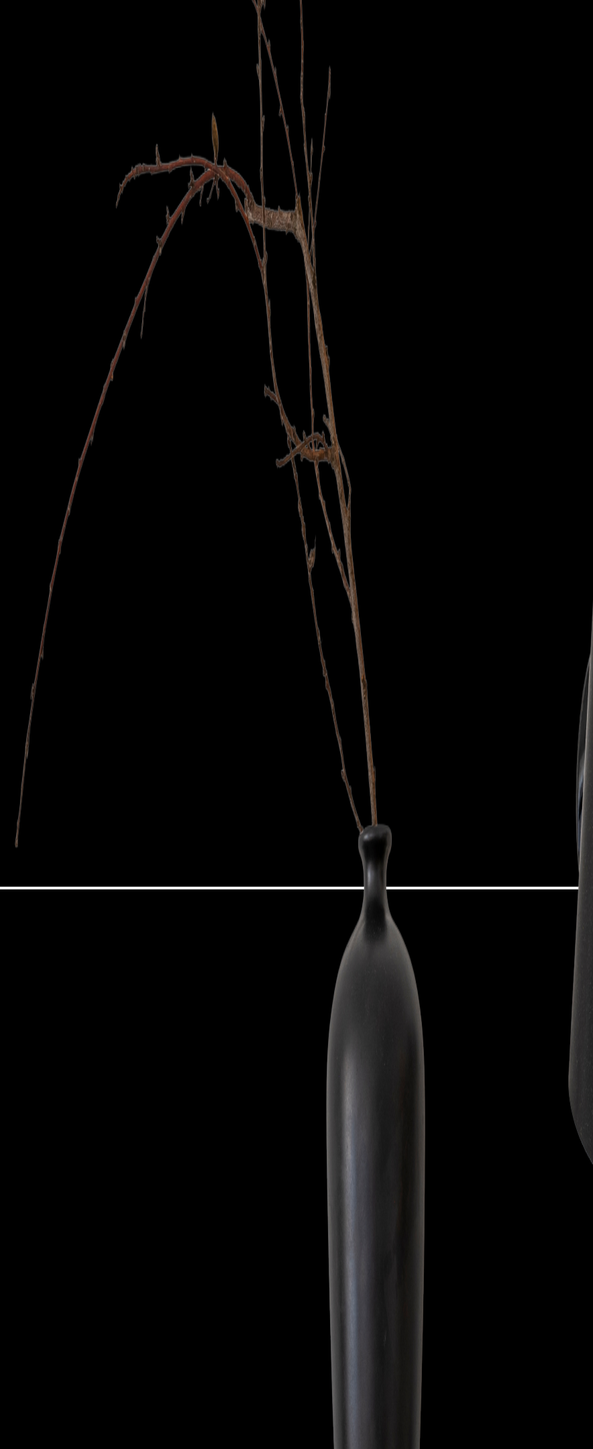
https://www.patreon.com/cw/b2bk?vanity=b2bk

- A French visual artist
- 2D → 3D Touchdesigner
- Helped refine structural design

VISUALIZATION (IN-PROGRESS)

1) Attention map morph

2) Token-level visualization



ATTENTION MAP MORPH

Divide the 50 attention steps into 3 phases:

- Phase 1 (1–10): Noise
- Phase 2 (11–30): Basic structure begins to form
- Phase 3 (31–50): More details become visible

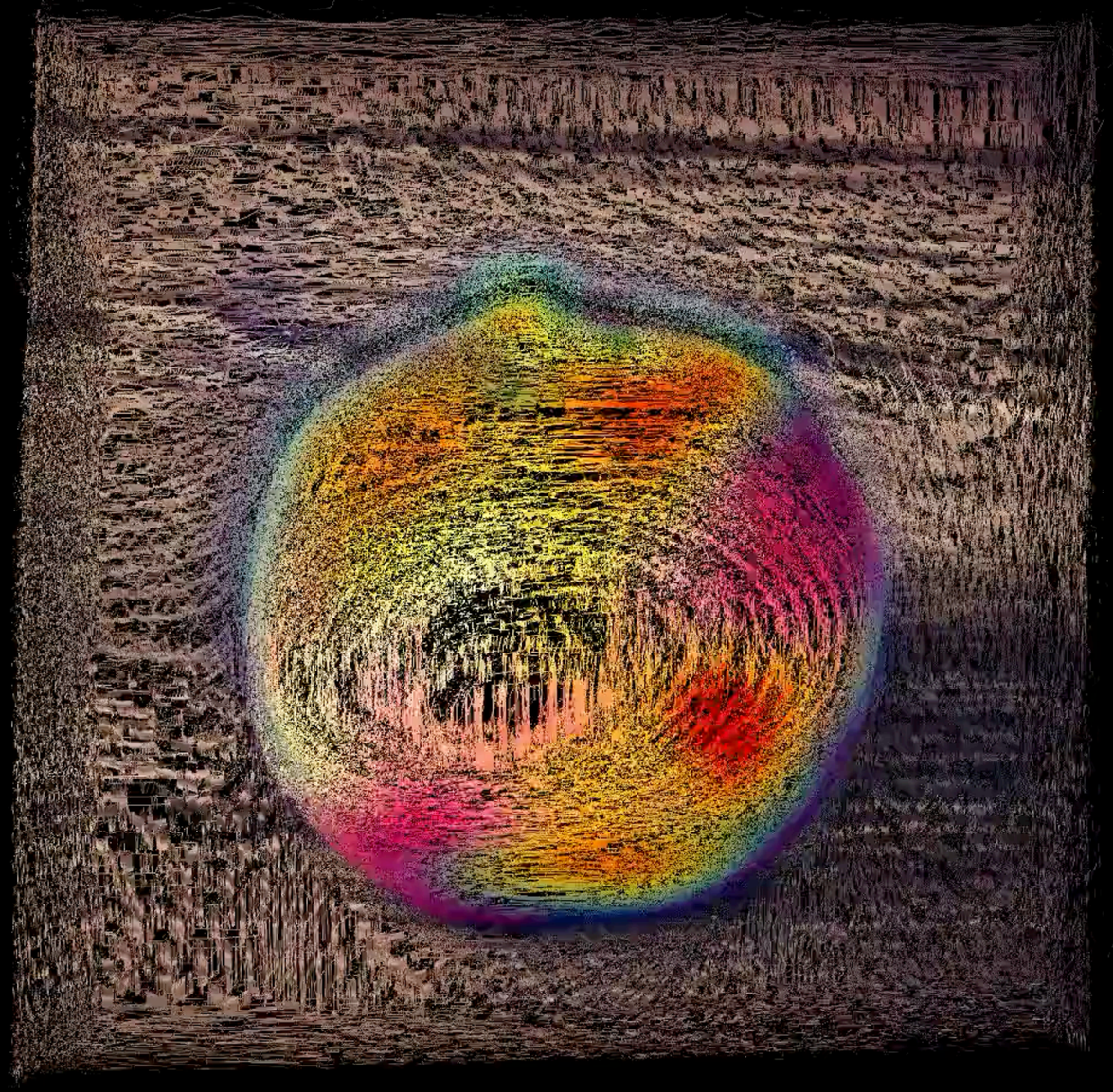
Each phase uses moving particles with generated depth

This also avoids pushing too much load onto the CPU/GPU

Transition Method

- The three phases morph smoothly into one another
- Instead of plainly listing all layers and flying a camera through them
- This avoids hard cuts or switching like: layer → black → layer

ATTENTION MAP MORPH



TOKEN-LEVEL VISUALIZATION

Input:

Dynamic attention maps (per timestep, for one token)

Masks:

- StabilityMask = attention
- ChaosMask = 1 – attention

Forces:

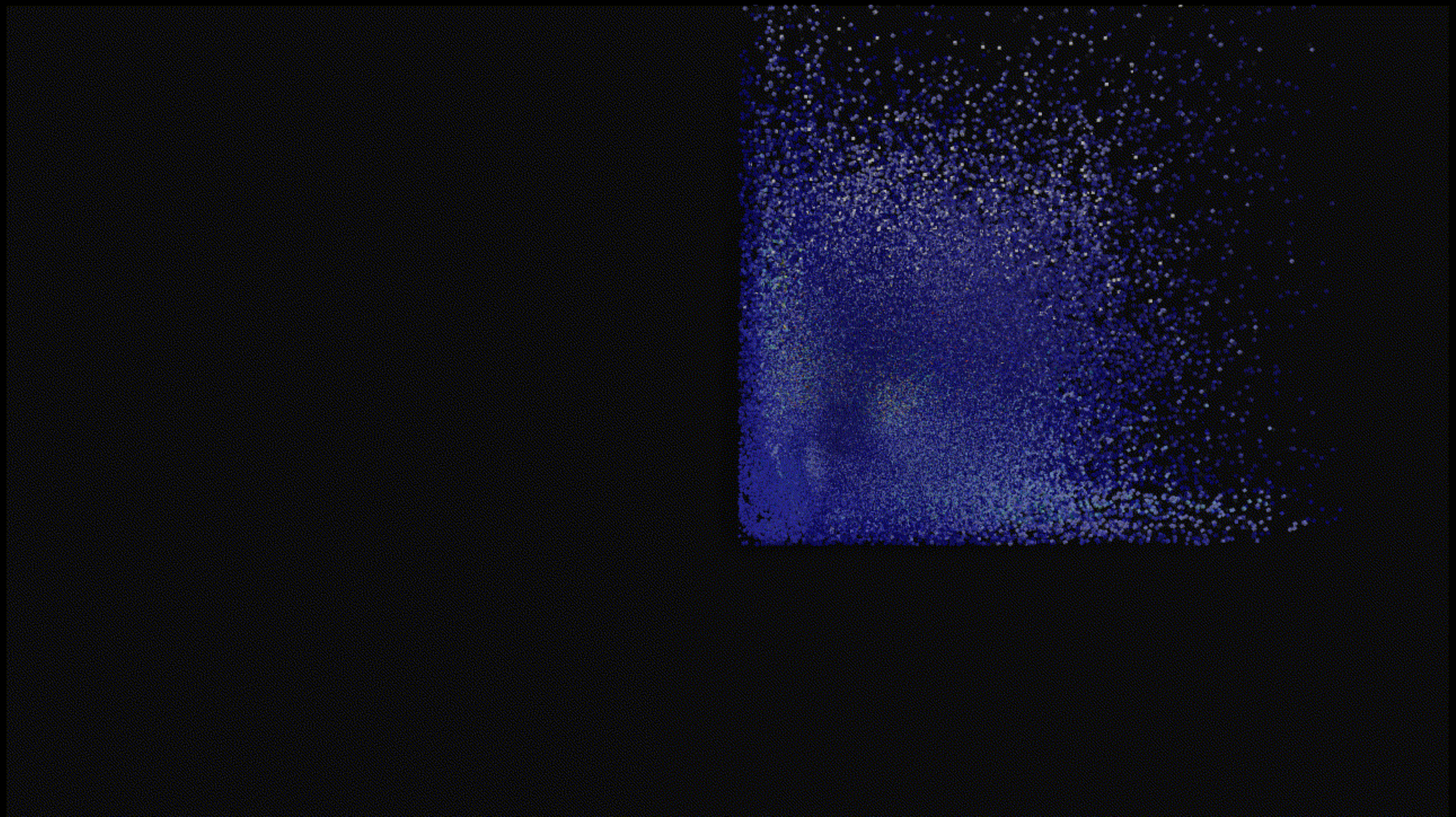
- Chaos = Wind × ChaosMask
- Stability = Attraction × StabilityMask
- Blended over time via Phase

Outcome:

A living visualization of how a token’s meaning solidifies through time.

TOKEN-LEVEL VISUALIZATION

TOKEN-LEVEL VISUALIZATION



FUTURE WORK

- Complete the second composition
- Combine the two visual levels into one flow
- Add interactive navigations for exploration