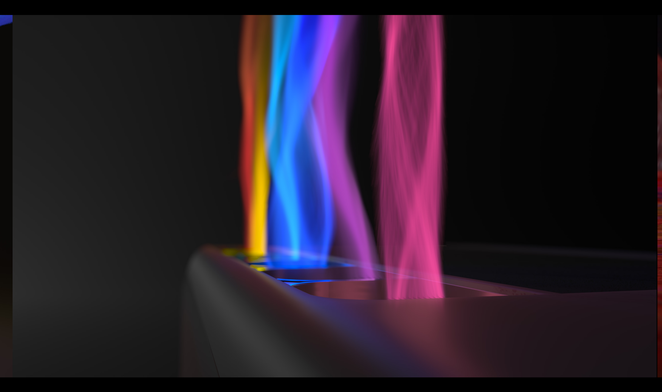
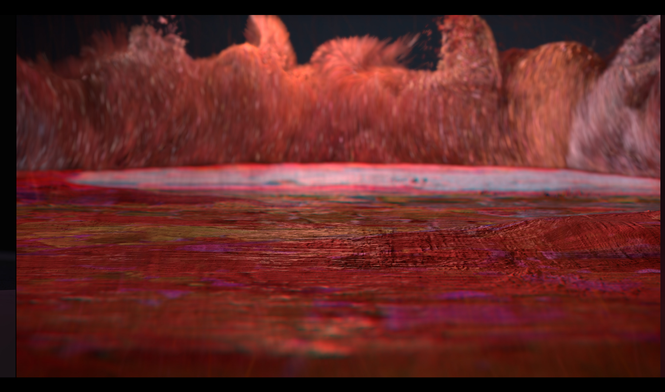
ONYX

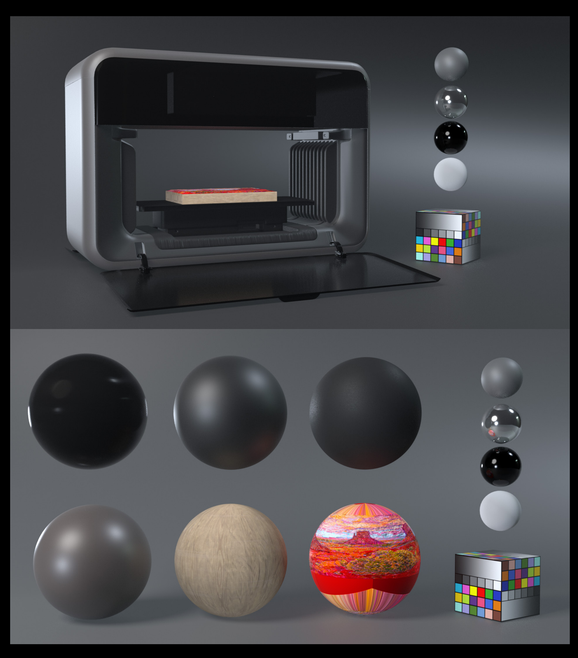
SANM 560 WINTER 2026

SOFIA MANCIAS ▪ BENJI HOPKINS ▪ YIQI ZHENG
RILEY HO ▪ ITIM KONGSAKULVATANASOOK

SCAD · HARBOR
ETC · CAT

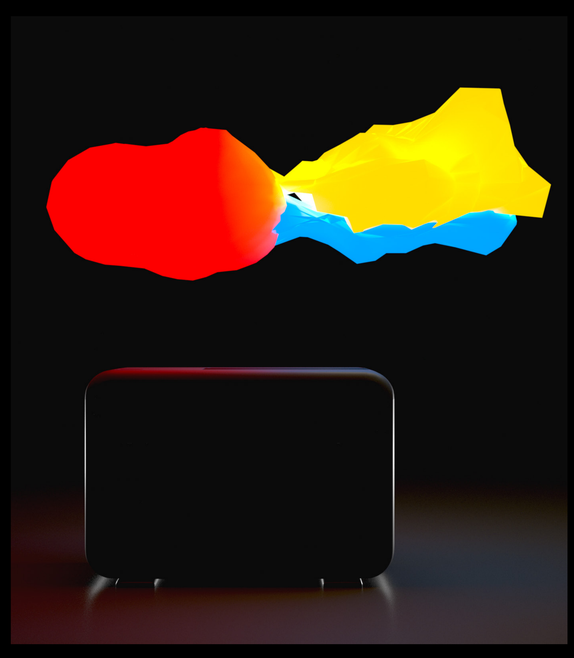



Look Development



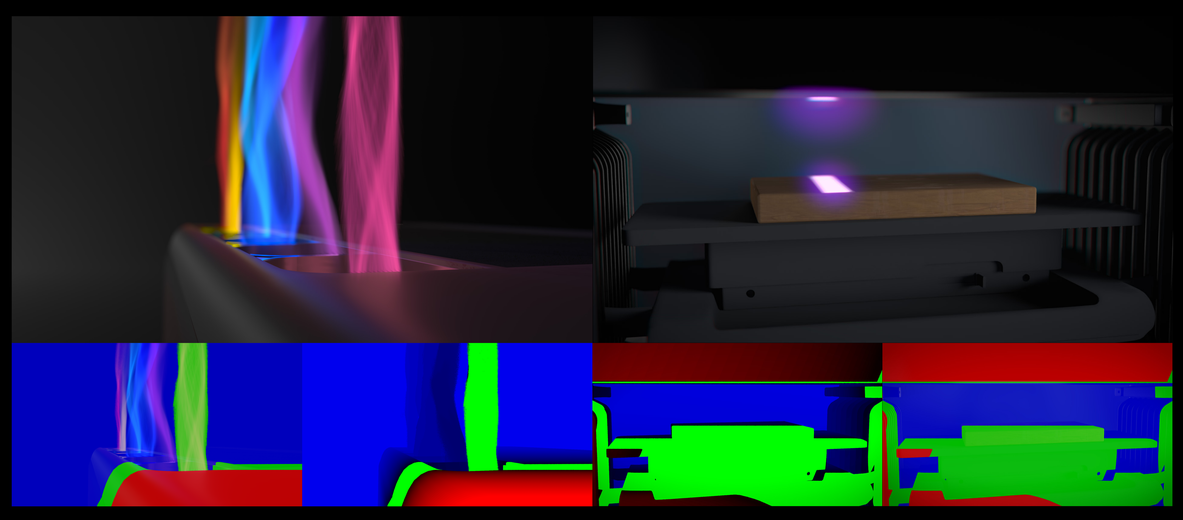
- **PRINTER MATERIALS:** Black reflective plastic, black matte plastic, and warm silver metallic plastic
- **PAINTING TEXTURE:** Created in Substance Painter for precise control of displacement and bump maps
- **SHADER BASE:** Karma Material Builder

Lighting



- **LIGHTING SETUP:** Dark scene with 1 key light and 1–2 fill lights depending on the shot
- **COLOR SPILL:** Used animated mesh lights, rendered separately for better control over spill brightness

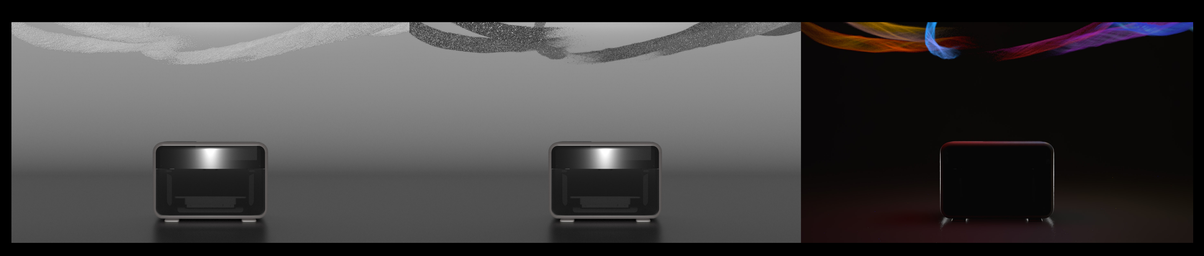
Compositing



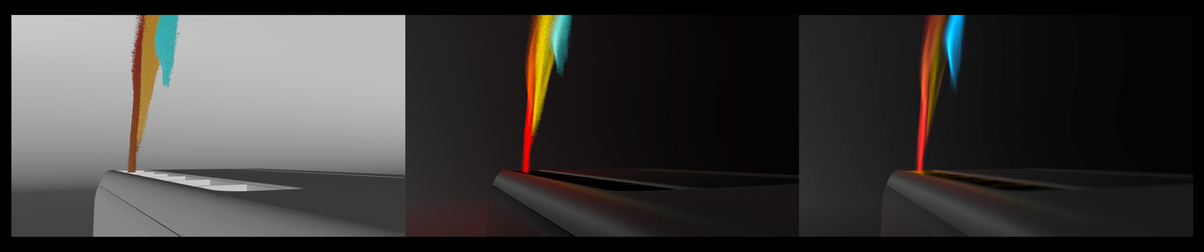
Depth of Field

- **COLOR CORRECTION**
  Color correction was performed using the provided AOV layers to refine the final look of the image. Adjustments to grade, saturation, brightness, and lighting were made to maintain balanced colors, preserve vibrancy, and ensure visual continuity between shots. Roto masks were used in certain shots to isolate specific elements for targeted color and lighting adjustments.
- **DEPTH OF FIELD**
  Depth of field was implemented in Nuke using Z-depth data from the Karma render to keep the main product in focus while softly blurring the background. Since the render passes were separated into object and background layers, the depth data was reconstructed to combine the necessary depth information for the effect to work correctly.

FX



Particles are guided along a curve using POP Curve Force to produce a controlled swirl motion. Small particle scales create a dense, fluid-like appearance through overlap. Randomized scale, emission masks, and opacity masks are applied to control particle distribution and introduce visual variation.

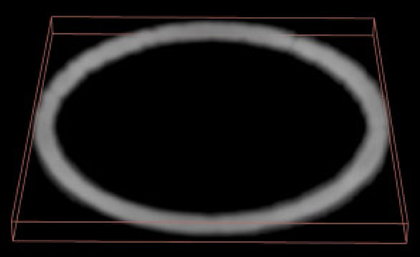


To avoid a flat appearance in the front flow, particle velocity was adjusted to achieve more controlled motion. A TimeShift node was used to stagger particle timing so they enter the holes sequentially. VEX-based size variation was applied, producing smaller particles in the front and larger particles in the back to enhance depth and visual variation.

```
// ---------------- Random ----------------
float randomVal = rand(@ptnum);

float randOmin = chf("rand_omin");
float randOmax = chf("rand_omax");
float randNmin = chf("rand_nmin");
float randNmax = chf("rand_nmax");
randomVal = fit(randomVal, randOmin, randOmax, randNmin, randNmax);

// ---------------- Age ----------------
float ageOmin = chf("age_omin");
float ageOmax = chf("age_omax");
float ageNmin = chf("age_nmin");
float ageNmax = chf("age_nmax");

float age = fit(@age, ageOmin, ageOmax, ageNmin, ageNmax);

// ---------------- Life ----------------
float lifeOmin = chf("life_omin");
float lifeOmax = chf("life_omax");
float lifeNmin = chf("life_nmin");
float lifeNmax = chf("life_nmax");

float life = fit(@life, lifeOmin, lifeOmax, lifeNmin, lifeNmax);
f@emissMask = clamp(age + life + randomVal, 0, 1);

@Cd = f@emissMask;
```

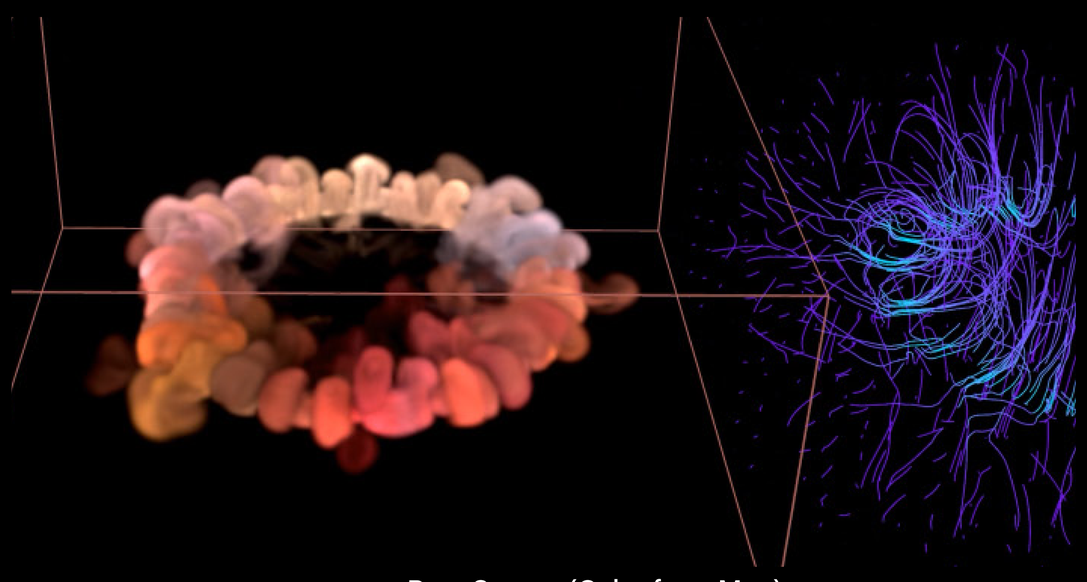
- **EMISSIVE MASK ON PARTICLES**
  Generate a mask for the emissive material using particle age and life attributes.
  Combine the mask with a random value generated by the rand() function to introduce variation.
  Use custom parameters to control the overall amount of emissive area.

FX



Pyro Source

Pyro Source (Color from Map)



Pyro Source (Color from Map)

Particles Advect by Volumes

Two particle layers are advected through a Pyro simulation, with color driven by a painting map via Attribute From Map. Pyro volumes are generated using a Pyro Solver and used directly as the velocity field. Pscale is dynamically adjusted based on camera frustum proximity to normalize apparent particle size across varying shot distances.