

KinemaFX

A Kinematic-Driven Interactive System for
Particle Effect Exploration and Customization

Yifei Zhang, Lin-Ping Yuan, Yuheng Zhao, Jielin Feng, Siming Chen

9/11

Jason Keyser

Particles

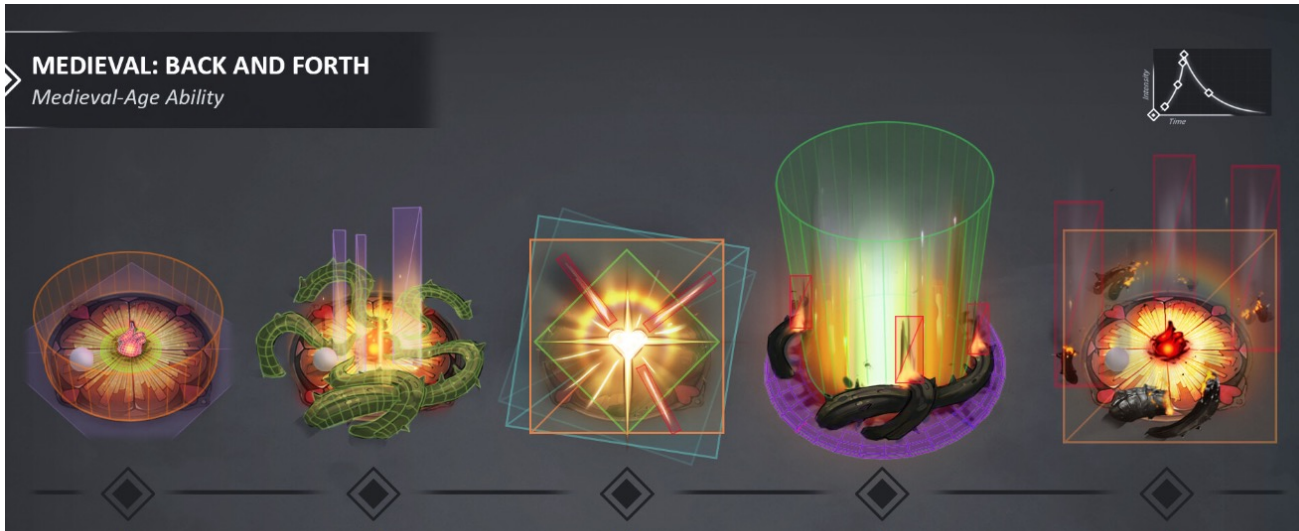
League of Legends
VFX reel

Asteroid Impact:
alternative camera view

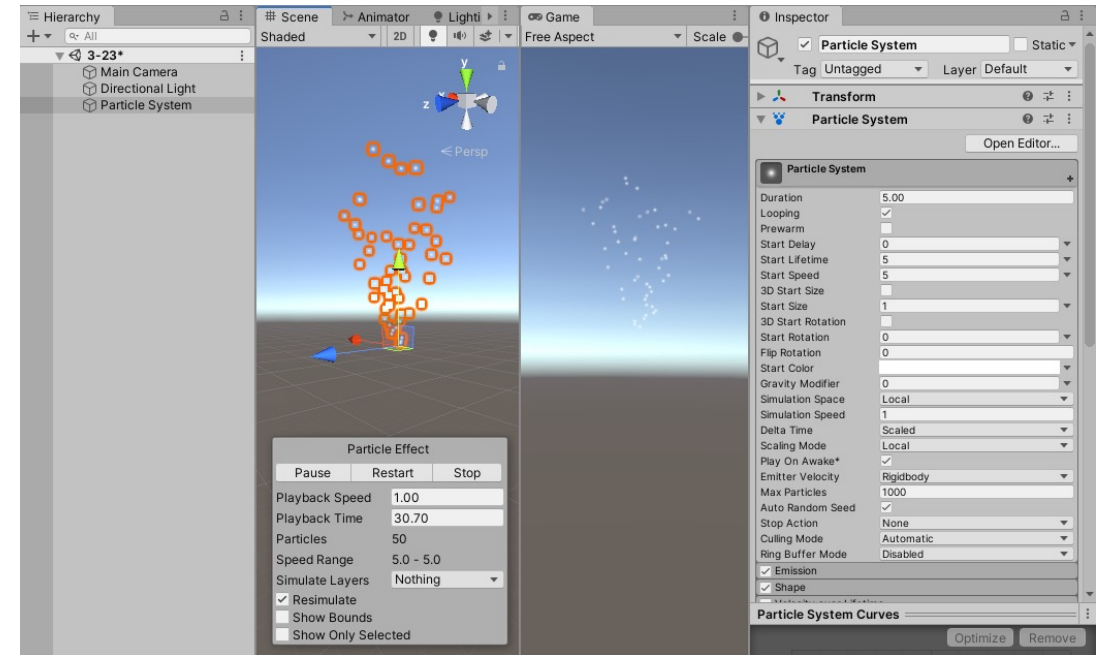


Background

Creating particle effects can be exhausting, especially for **non-expert** users.



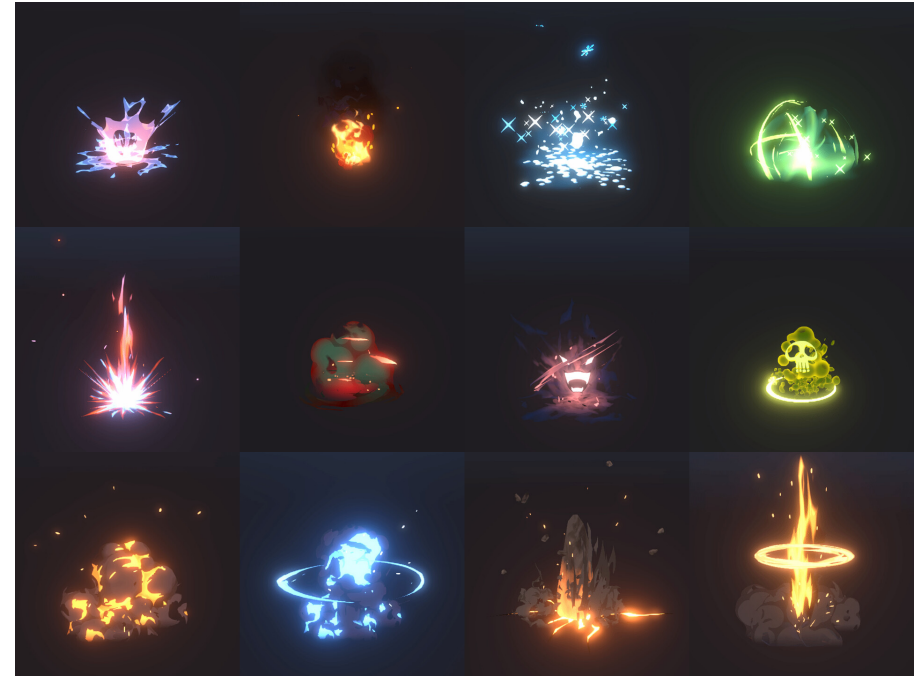
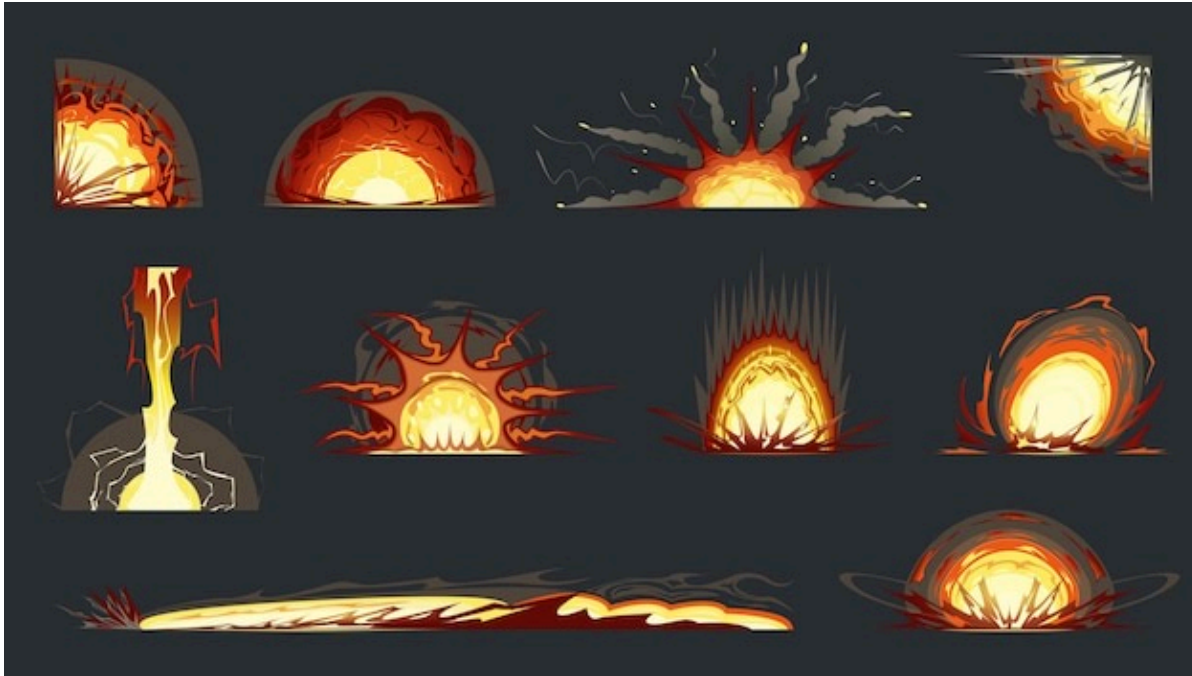
Artistic skills



Software learning curve

Background

Finding particle effects that match the user's intent is challenging.



Different styles and motion structures for explosion effects

Related Work

Facilitating effect artwork creation

Sketch-based

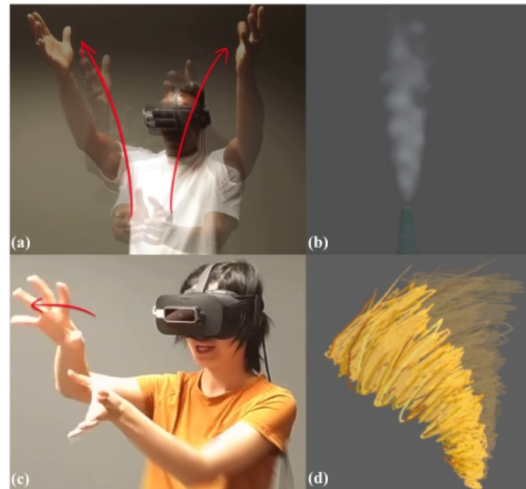
High requirements for
aesthetic literacy



Energy-Brushes (UIST 2016)

Hand Gesture

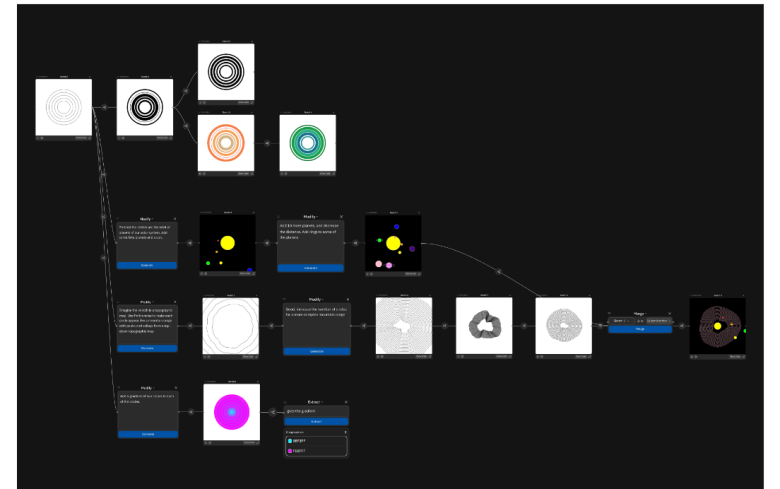
Fewer controllable parameters



MagicalHands (UIST 2019)

Semantic Input

Limited by the accuracy of
semantic expression and the
understanding ability of LLMs



Spellburst (UIST 2023)

Formative Study

Part I: Asset Collection

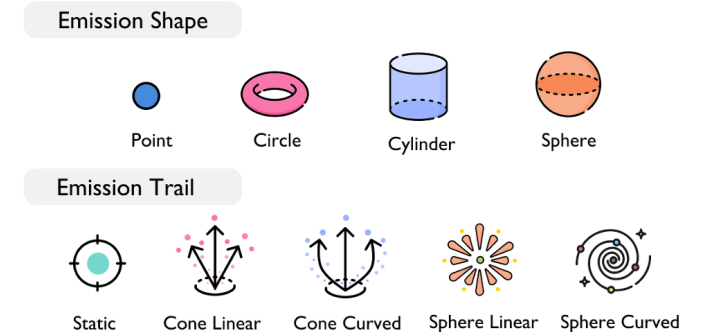
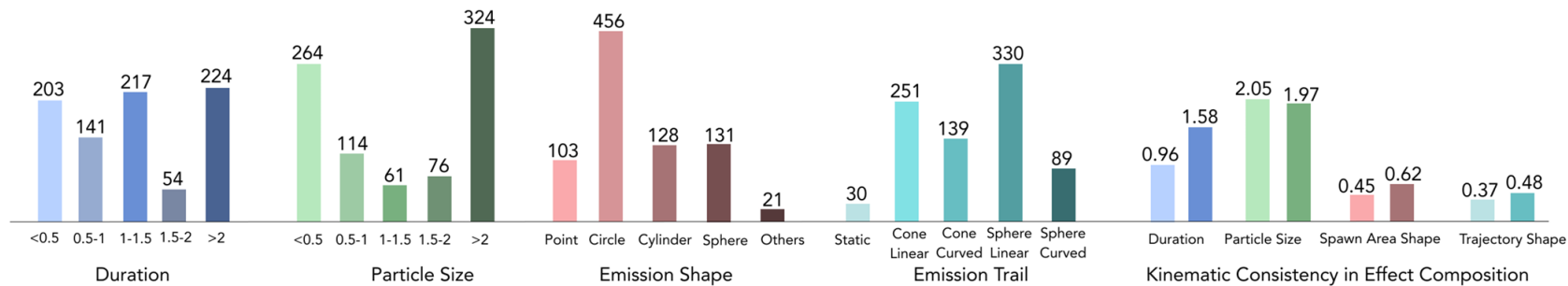


Part II: Semi-structured Interview

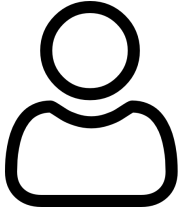


- Observation Task
- Conceptualization Task
- Exploration Task

Part III: Qualitative Analysis



Method Overview



How to intuitively express and efficiently control particle effects with complex kinematic behaviors ?

A Conceptual Model of Particle Effects

- The **semantic** component encodes the intended meaning and stylistic qualities.
- The **kinematic** component captures the spatiotemporal structure and particle motion.

Kinematic-driven System Design

- Implicit user preference-guided exploration **workflow**.
- Enhancing particle effect search and alignment through **structured representation** and distance metric.

Workflow

a Initial Intent Input

The interface for initial intent input is divided into two main sections: semantic input and graphical input.

- Semantic Input (a1):** Features a 'New Shape' dropdown set to 'Circle' and an 'Add' button. Below is a text input field containing 'trail'. A duration slider is set to '2.4s'. Three control buttons labeled 'Position', 'Rotation', and 'Scale' are shown. A 3D visualization of a trail with purple and red arrows is displayed, with a hand icon and the text 'Manipulate trail' indicating interactive control.
- Graphical Input (a2):** Features a 'New Shape' dropdown set to 'Sphere' and an 'Add' button. Below is a text input field containing 'explosion'. A duration slider is set to '0.8s'. Three control buttons labeled 'Position', 'Rotation', and 'Scale' are shown. A 3D visualization of a sphere with a grid and purple arrows is displayed.

Below these sections are three detailed graphical input panels for 'Position', 'Rotation', and 'Scale', each showing a 3D coordinate system with colored axes and arrows.

b Effect Exploration

The effect exploration interface is a dark-themed workspace with several key components:

- Top Panel:** A 'New Shape' dropdown set to 'Sphere' and an 'Add' button. Below are two buttons labeled 'trail' and 'explosion', each with a 'delete' button next to it.
- Central Area:** A large 3D visualization area showing various particle effects, including a red cloud, a yellow ring, and a blue starburst. A 3D coordinate system with colored axes is overlaid on the scene.
- Bottom Left (b1):** A 'Weighted searching' panel with a 'Kinematic' slider and a 'Semantic' slider. A 'Generate' button is located at the bottom right of this panel.
- Bottom Center (b2):** An 'Iterative selection' panel showing a sequence of particle effects being refined.
- Bottom Right (c1):** A 'Particle effect selection' panel with a 3D coordinate system overlaid on a selected particle effect. Below it are 'Speed: * 0.7' and 'Delay: 0.4s' sliders, and a 'Save' button.

c Effect Composition

The effect composition interface is a dark-themed workspace for finalizing and saving effects:

- Top (c2):** 'Transformation control' section showing a 'Drag and drop' operation between two particle effect visualizations.
- Middle:** 'Adjusting temporal features' section with 'Speed: * 0.7' and 'Delay: 0.4s' sliders, and a 'Save' button.
- Bottom (c3):** 'Effect artworks' section displaying two final rendered particle effect artworks side-by-side.

New Shape :

Circle ▾

- ✓ **Circle**
- Sphere**
- Cylinder**

Add

Start with a primitive shape

Kinematic



Semantic

Generate

Save

Workflow

a Initial Intent Input

New Shape: Circle

trail **a1 Semantic input**

Duration: 2.4s

Position

Rotation

Scale

a2 Graphical input

Position

Rotation

Scale

b Effect Exploration

New Shape: Sphere

explosion

Duration: 0.8s

Position

Rotation

Scale

b2 Iterative selection

b1 Weighted searching

Kinematic Semantic

c1 Particle effect selection

trail

explosion

Speed: * 0.7

Delay: 0.4s

c Effect Composition

c2 Transformation control

Drag and drop

Adjusting temporal features

Speed: * 0.7

Delay: 0.4s

c3 Effect artworks

New Shape :

Cylinder

Add

effect 1

delete

fire

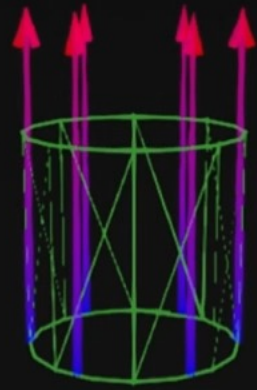
Duration: 1.2s



Position

Rotation

Scale



Kinematic



Semantic

Generate

Save

Workflow

a Initial Intent Input

New Shape: Circle

trail **a1 Semantic input**

Duration: 2.4s

Position **Rotation** **Scale** *Manipulate trail*

a2 Graphical input

Position **Rotation** **Scale**

The interface shows a control panel for creating a 'trail' effect. It includes a 'New Shape' dropdown set to 'Circle' and an 'Add' button. Below is a text input field containing 'trail' and a sub-label 'a1 Semantic input'. A duration slider is set to 2.4s. Three buttons labeled 'Position', 'Rotation', and 'Scale' are shown next to a 3D wireframe sphere with red arrows indicating manipulation axes. A 'Manipulate trail' label is next to a small 3D view of the trail. Below this, three separate 3D views illustrate the 'Position', 'Rotation', and 'Scale' controls, each with a yellow arrow pointing from the main interface to the respective view.

b Effect Exploration

New Shape: Sphere

explosion

Duration: 0.8s

Position **Rotation** **Scale**

b2 Iterative selection

b1 Weighted searching

Kinematic **Semantic**

The interface shows a control panel for creating an 'explosion' effect. It includes a 'New Shape' dropdown set to 'Sphere' and an 'Add' button. Below is a text input field containing 'explosion' and a duration slider set to 0.8s. Three buttons labeled 'Position', 'Rotation', and 'Scale' are shown next to a 3D wireframe sphere. A large, glowing red explosion effect is displayed in the center. Below the main interface, two rows of smaller effect thumbnails are shown. The first row is labeled 'b2 Iterative selection' and the second row 'b1 Weighted searching'. At the bottom, there are two sliders labeled 'Kinematic' and 'Semantic', and a 'Generate' button.

c Effect Composition

trail

explosion

c2 Transformation control

Drag and drop

Adjusting temporal features

Speed: * 0.7 **Delay: 0.4s**

c3 Effect artworks

Speed: * 0.7 **Delay: 0.4s**

The interface shows a control panel for effect composition. It includes two buttons labeled 'trail' and 'explosion', each with a 'delete' button. A large 3D scene is shown with a glowing yellow sphere and several blue and green particle trails. A white box highlights a specific particle effect, labeled 'c1 Particle effect selection'. Below this, a 'Drag and drop' label is shown next to a 3D view of the selected effect. Below that, a 'Adjusting temporal features' label is shown next to two sliders labeled 'Speed: * 0.7' and 'Delay: 0.4s', and a 'Save' button. At the bottom, a row of three effect artworks is shown, labeled 'c3 Effect artworks', with the same 'Speed' and 'Delay' sliders and 'Save' button below them.

New Shape :

Sphere



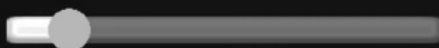
Add

effect 1

delete

explosion

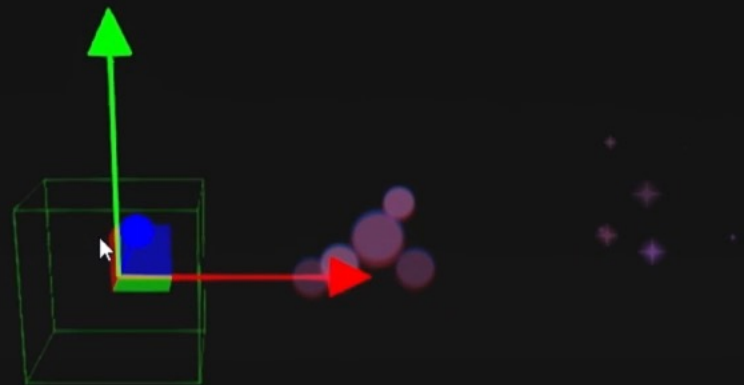
Duration: 0.5s



Position

Rotation

Scale



Click particle effect

Kinematic



Semantic

Generate

Speed: * 1



Delay: 0s



Save

Kinematic-Driven Method

$$R = (S, K)$$

$$K = (\text{shape}, \text{trail}, \text{duration})$$

Stage I: Convert constraints into structured representations




 Semantic and  graphical input

Initial description

Initial kinematic


• Semantic
• embedding

 Particle effect instance

Completed description


Simulation and sampling

• Kinematic
• shapeType

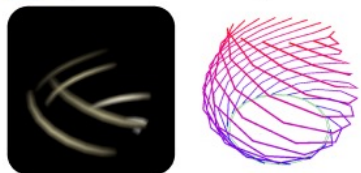
 Expansion of two effect instances

Description reasoning via LLM

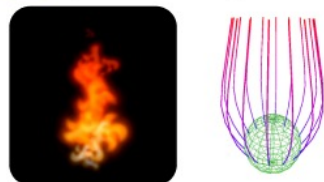
Interpolation/extrapolation

• duration
• trail 

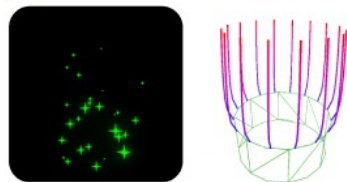
1 Spiral ascending trail



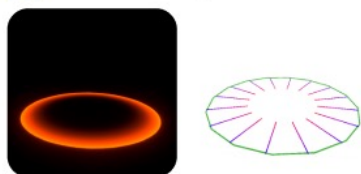
2 Upward-rising flame



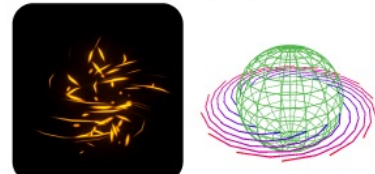
3 Vertically rising sparks



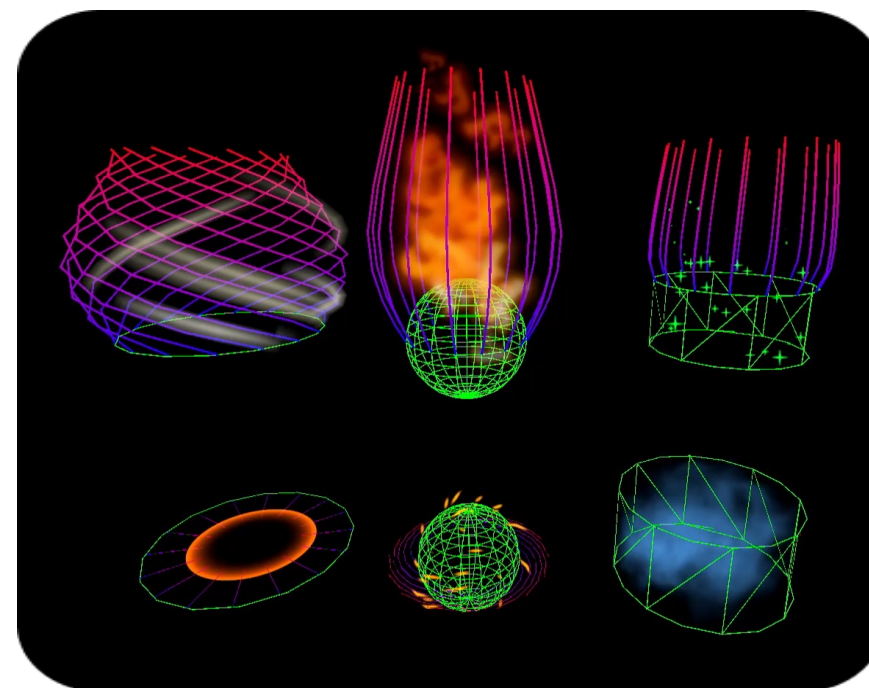
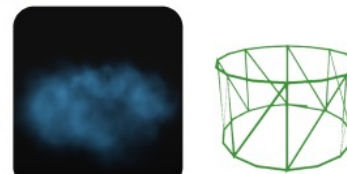
4 Contracting circular halo



5 Spiral-diverging trail

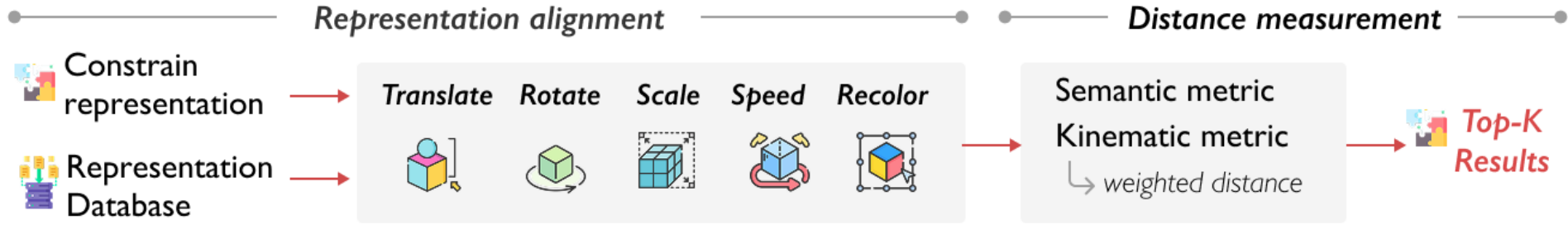


6 Static smoke



Kinematic-Driven Method

Stage 2: Search via alignment and distance metric



For each instance in the dataset, we apply transformations to make it as close as possible to the constraints.

$$R^* = \arg \min_{R_i \in \mathcal{D}, T \in \mathcal{T}} D(R_c, T(R_i)),$$

The structured representation makes it easier for distant metric.

$$D(R_c, R_i) = w_s \cdot D_s(S_c, S_i) + w_k \cdot D_k(K_c, K_i)$$

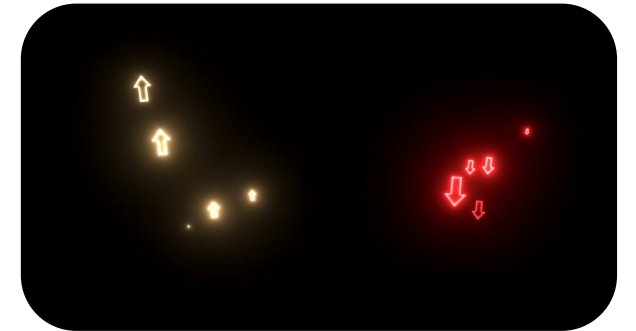
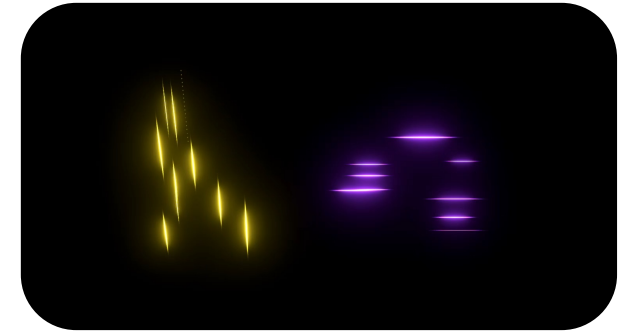
$$D_s(S_c, S_i) = 1 - \frac{S_c \cdot S_i}{\|S_c\| \|S_i\|}$$

$$D_k(K_c, K_i) = D_{\text{trail}}(\text{trail}_c, \text{trail}_i) \cdot f(\text{duration}_c, \text{duration}_i).$$

$$D_{\text{trail}}(\text{trail}_c, \text{trail}_i) = \sum_{j=1}^N D_{\text{shape}}^{(j)}$$

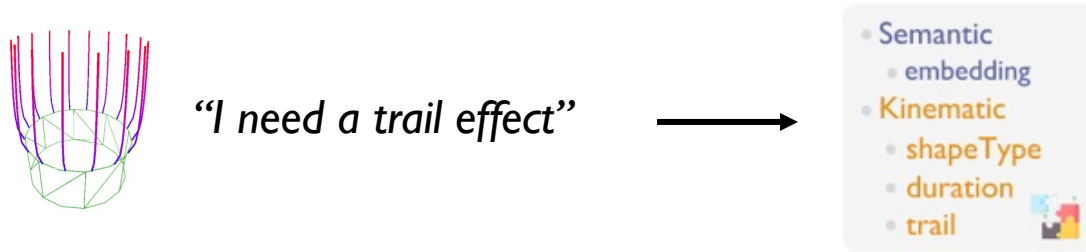
$$D_{\text{shape}}^{(j)} = H(\text{shape}_c^{(j)}, \text{shape}_i^{(j)}) + \lambda(1 - \cos(\Delta\phi_j)).$$

$$f(\text{duration}_c, \text{duration}_i) = 1 + \alpha \left(\max \left(\frac{\text{duration}_c}{\text{duration}_i}, \frac{\text{duration}_i}{\text{duration}_c} \right) - 1 \right)^2, \quad (7)$$

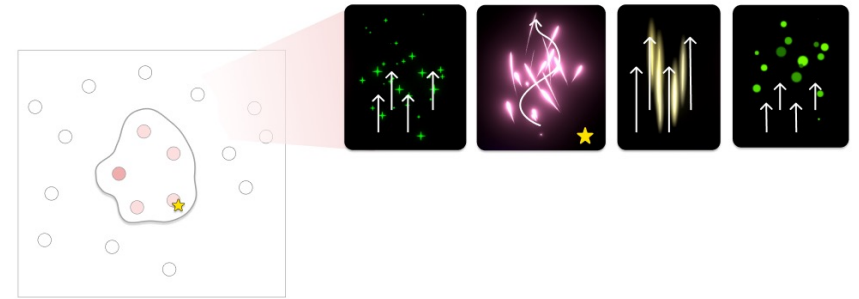


Offline Step: instances in the dataset → structured representation

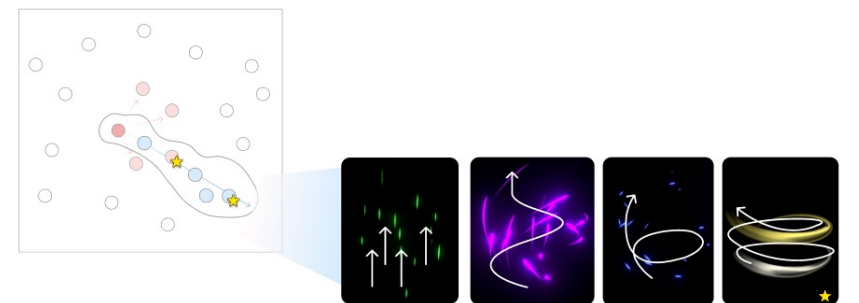
Step 0: User input → structured representation (as the initial constraint)



Step 1: Find the nearest particle effects around the current constraint, and the user select favorite one (as the next constraint)

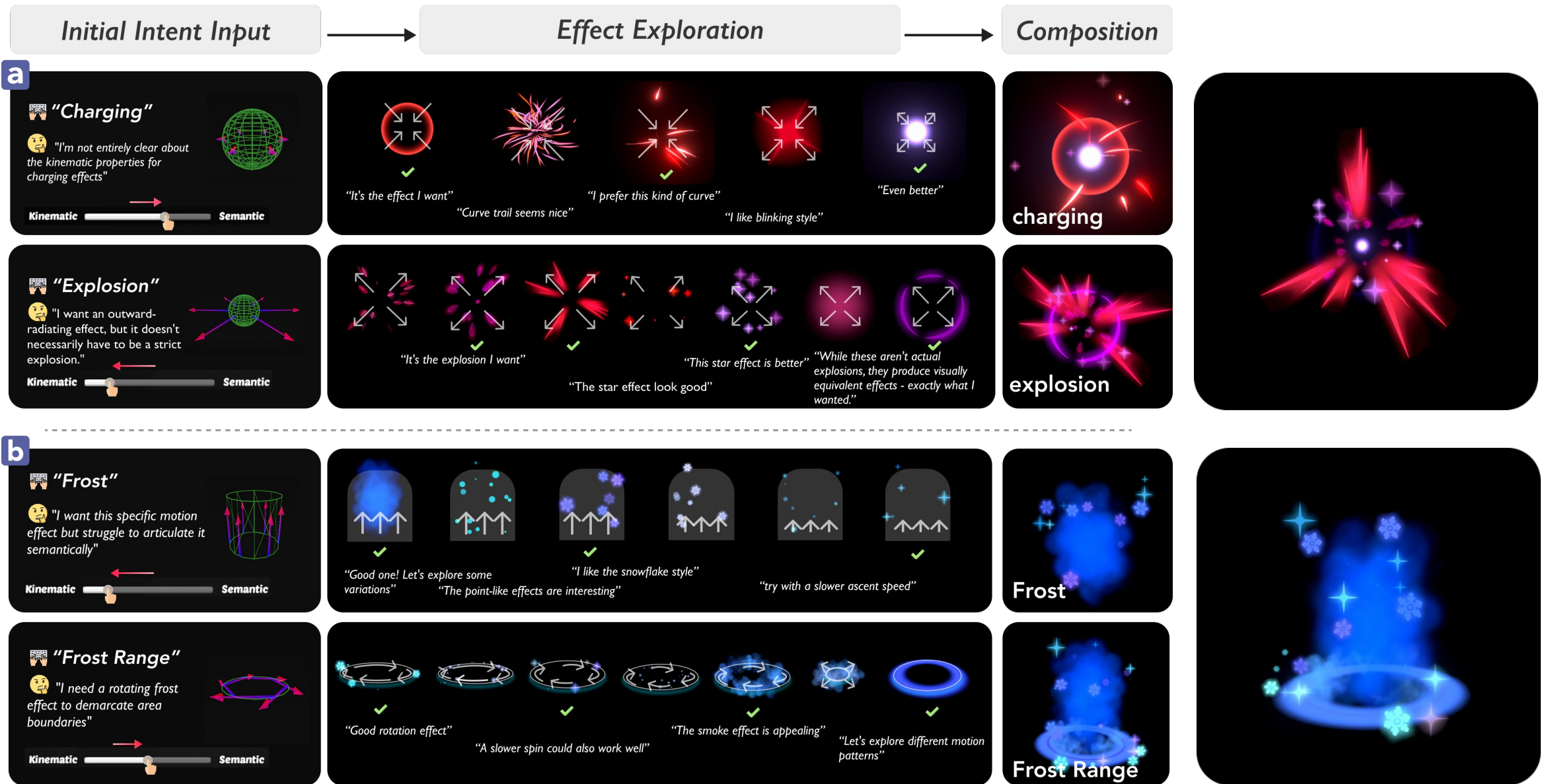


Step 2: Along the direction from the current constraint to the next one, we construct new representations through interpolation and extrapolation of the kinematic representation as well as semantic reasoning, and then retrieve the instances that best match these representations



Then, the user select favorite instance as the current constraint, and get back to **step 1**

Usage Scenario



New Shape :

Circle 

Add

Add a circle shape

Kinematic



Semantic

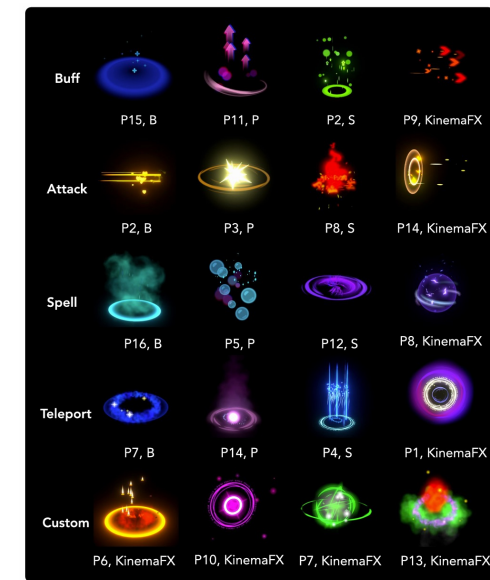
Generate

Save

User study

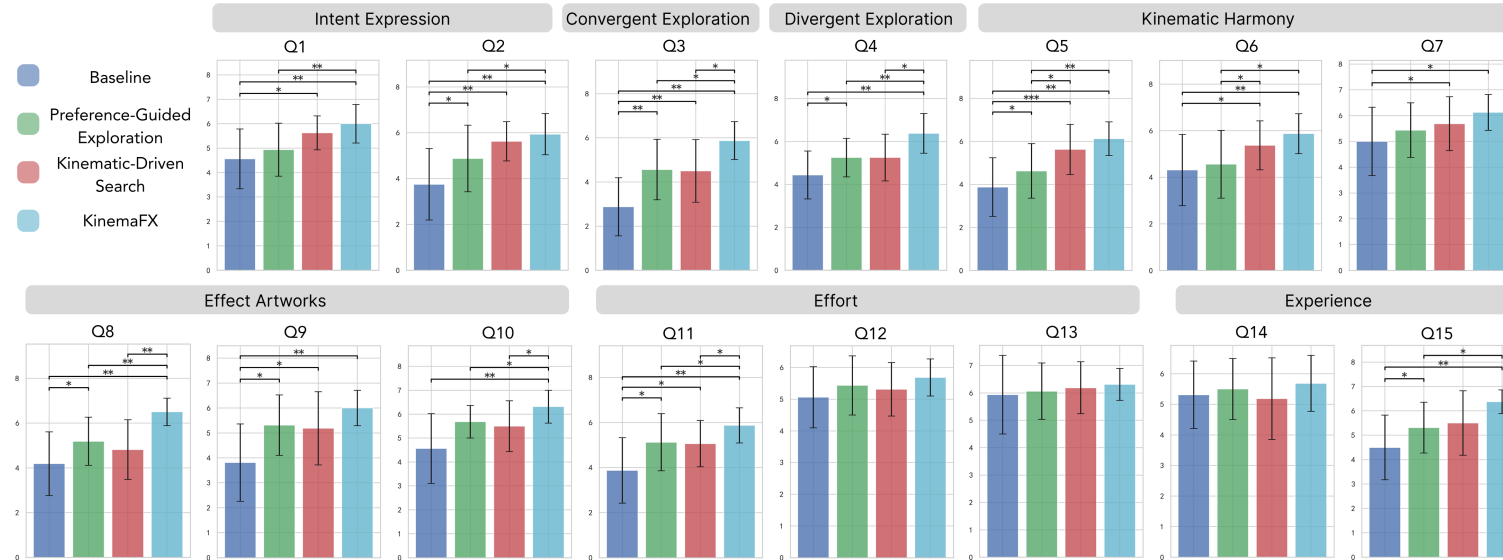
- 16 participants,
- 4 study conditions
- 4 effect themes

ID	Question
Q1	I was able to clearly express my intent.
Q2	I liked the input method for expressing intent.
Q3	I was able to effectively influence the iteration direction.
Q4	I discovered surprising and satisfying results.
Q5	I was satisfied with the shape and motion of the search results.
Q6	I didn't need to manually adjust much when combining effects.
Q7	The spatiotemporal relationship of effects was harmonious.
Q8	I was satisfied with the effect artwork I created.
Q9	The effect artwork contained the core features I intended.
Q10	The final effect was diverse beyond the core features.
Q11	I did not feel a heavy cognitive load.
Q12	I did not feel a heavy physical load.
Q13	The system was easy to learn.
Q14	The system was easy to use.
Q15	I would like to use this tool in the future.



Questionnaire Items

Effect artworks created by users



Results across seven dimensions: 1) Intent Expression; 2) Convergent Exploration; 3) Divergent Exploration; 4) Kinematic Harmony; 5) Effect Artworks; 6) Effort; 7) Experience

Gallery

(effect artworks from our user study)



UIST 25

BUSAN, KOREA | SEP. 28TH – OCT. 1ST 2025

Thank you !

KinemaFX:

A Kinematic-Driven Interactive System for
Particle Effect Exploration and Customization

