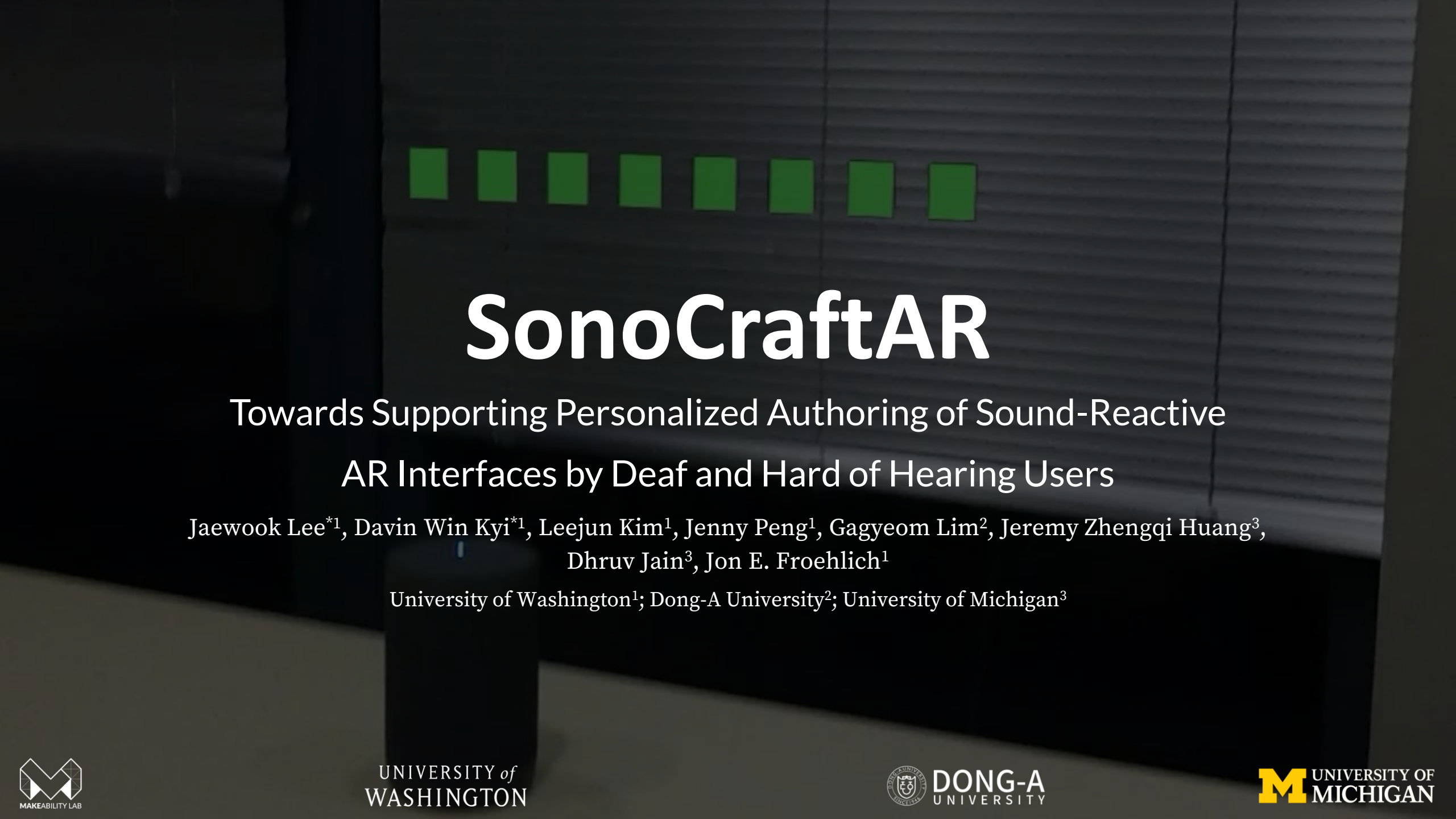


SonoCraftAR

Towards Supporting Personalized Authoring of Sound-Reactive AR Interfaces by Deaf and Hard of Hearing Users

Jaewook Lee[*1], Davin Win Kyi[*1], Leejun Kim[1], Jenny Peng[1], Gagyeom Lim[2], Jeremy Zhengqi Huang[3], Dhruv Jain[3], Jon E. Froehlich[1]

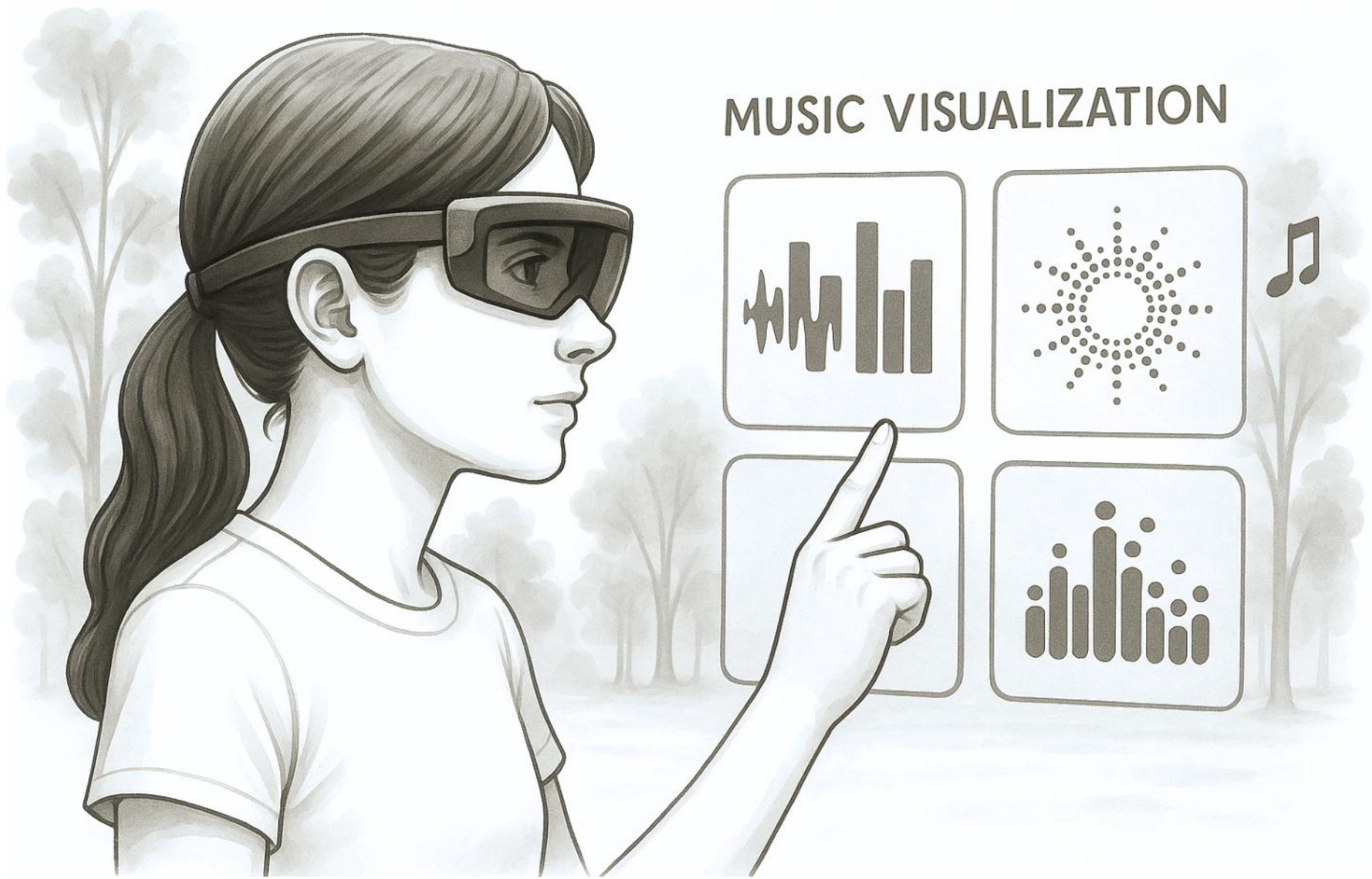
University of Washington[1]; Dong-A University[2]; University of Michigan[3]

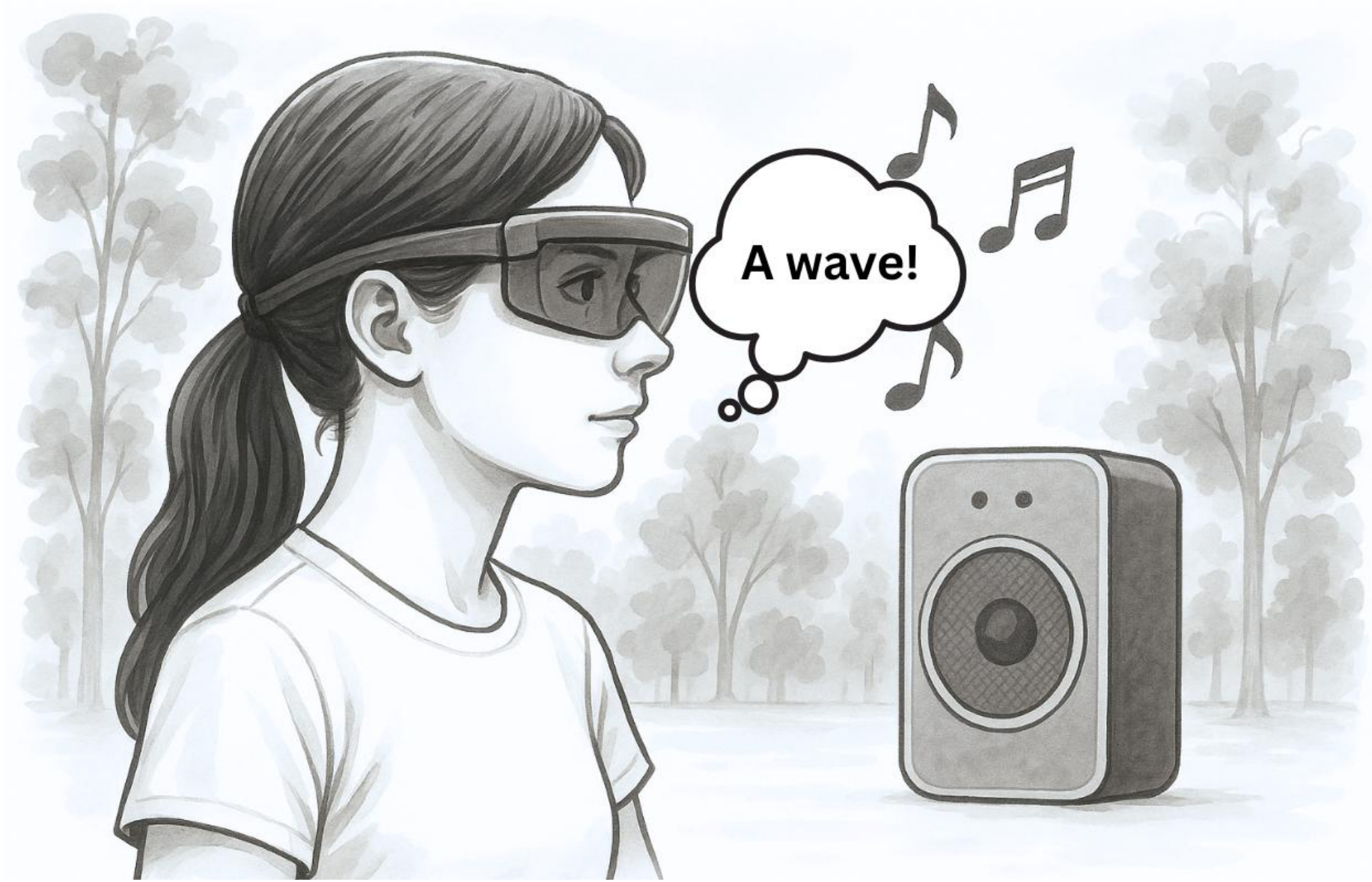
MAKEABILITY LAB

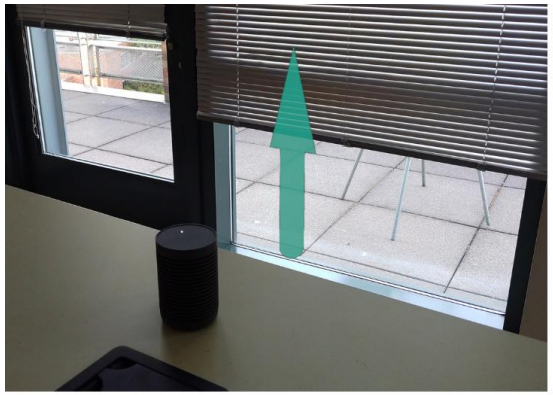
UNIVERSITY of WASHINGTON

DONG-A UNIVERSITY

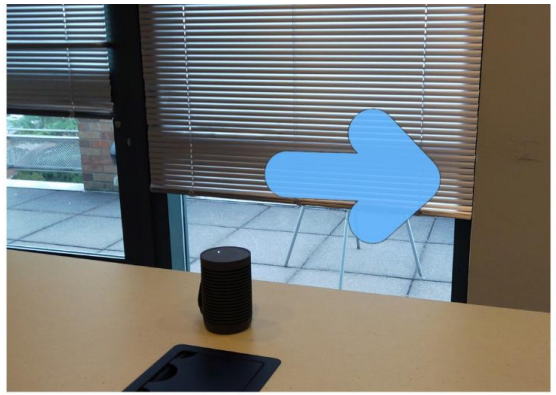
UNIVERSITY OF MICHIGAN


MUSIC VISUALIZATION


A wave!

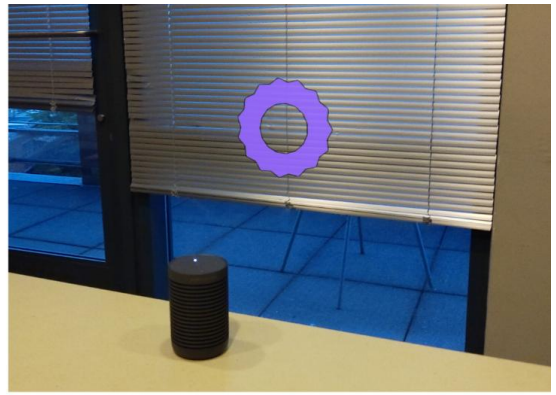


(A) “An arrow that changes in length”



(B) “A cute arrow that changes in size”

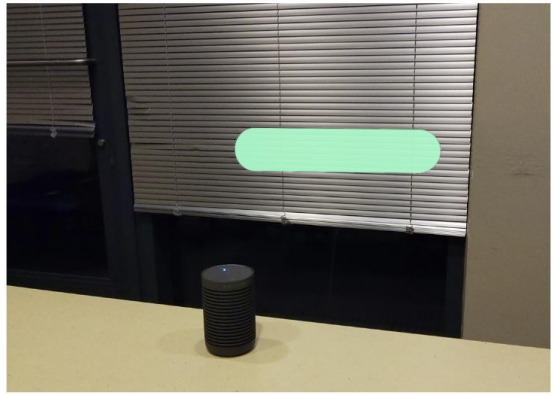
(C) “A pulse resembling a wifi signal”



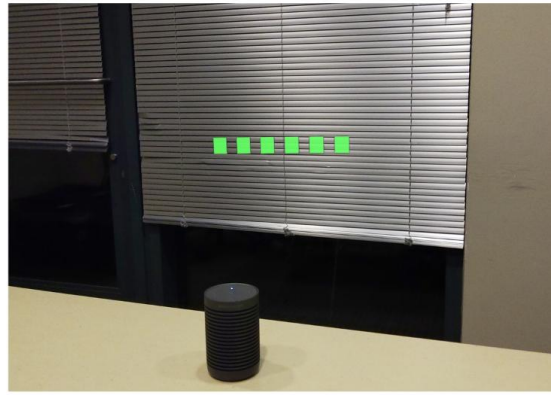
(D) “A purple music disc”

(E) “A wave”

(F) “A bright rainbow wave”

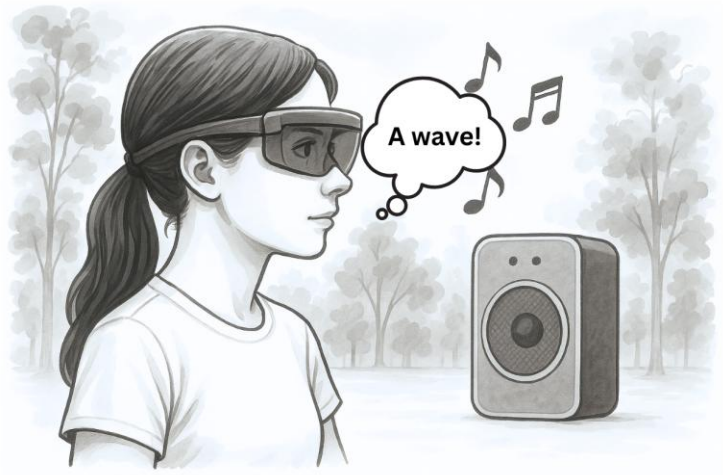


(G) “A sound bar with rounded corners”

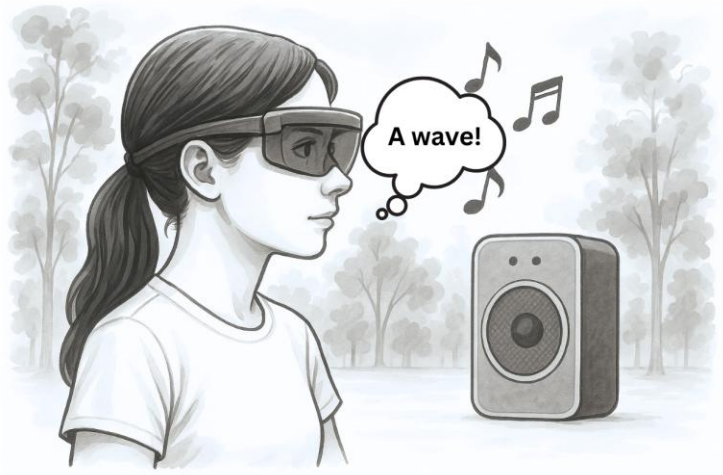


(H) “A retro TV volume control bar”

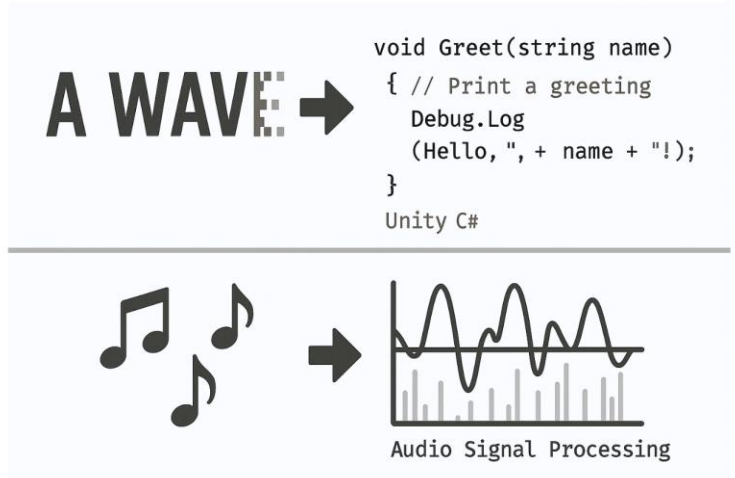
We introduce **SonoCraftAR**, a wearable AR system that empowers Deaf and hard-of-hearing (DHH) users to create sound-reactive AR interfaces from typed input.



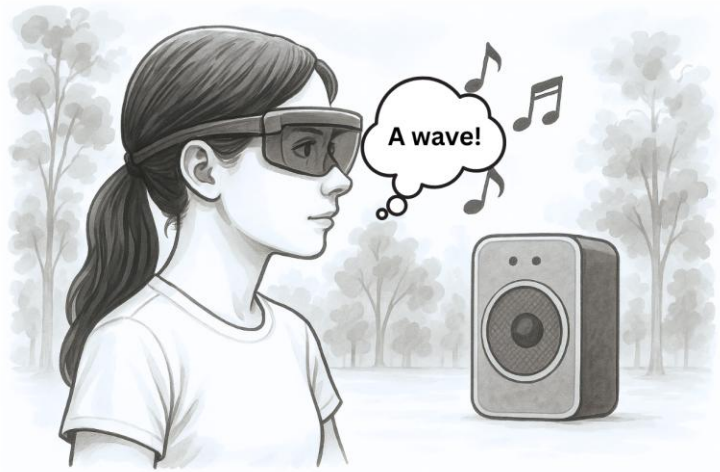
A DHH user types a description of their desired sound-reactive AR interface.



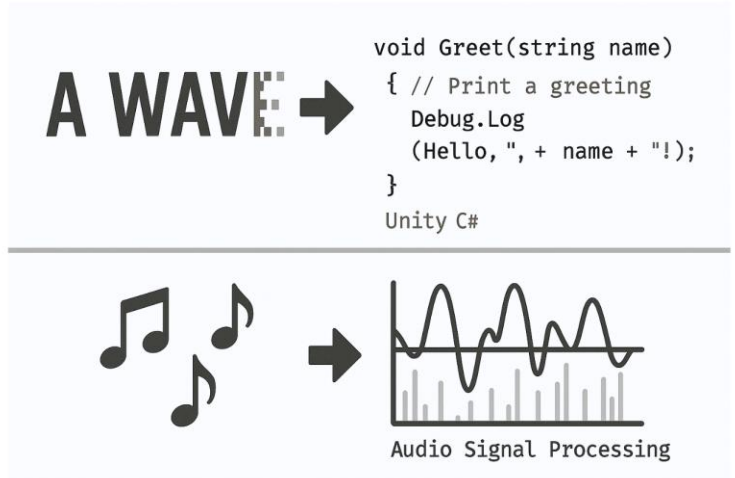
A DHH user types a description of their desired sound-reactive AR interface.



LLM agents generate UI and animation code, while signal processing extracts sound features.

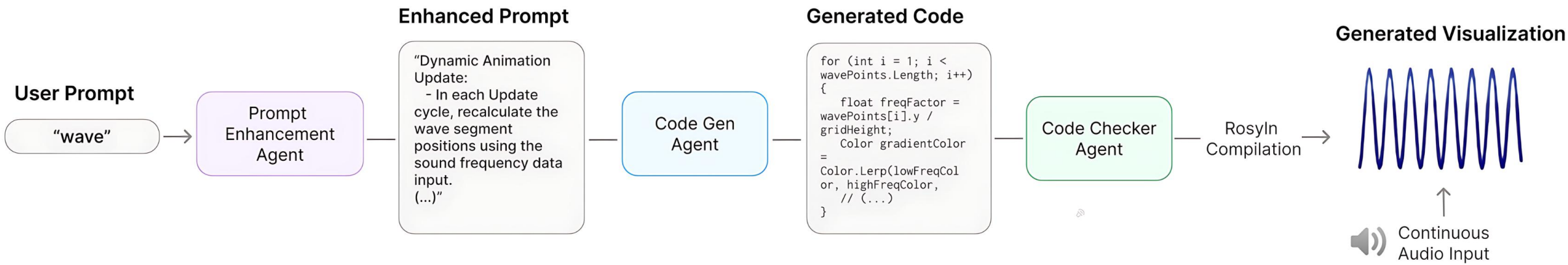


A DHH user types a description of their desired sound-reactive AR interface.



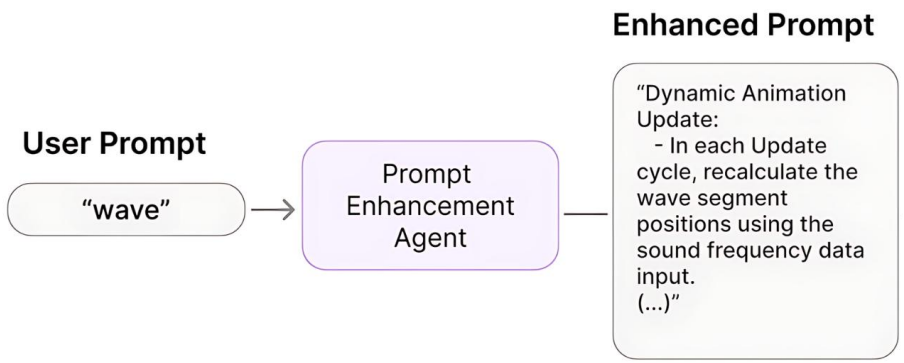
LLM agents generate UI and animation code, while signal processing extracts sound features.

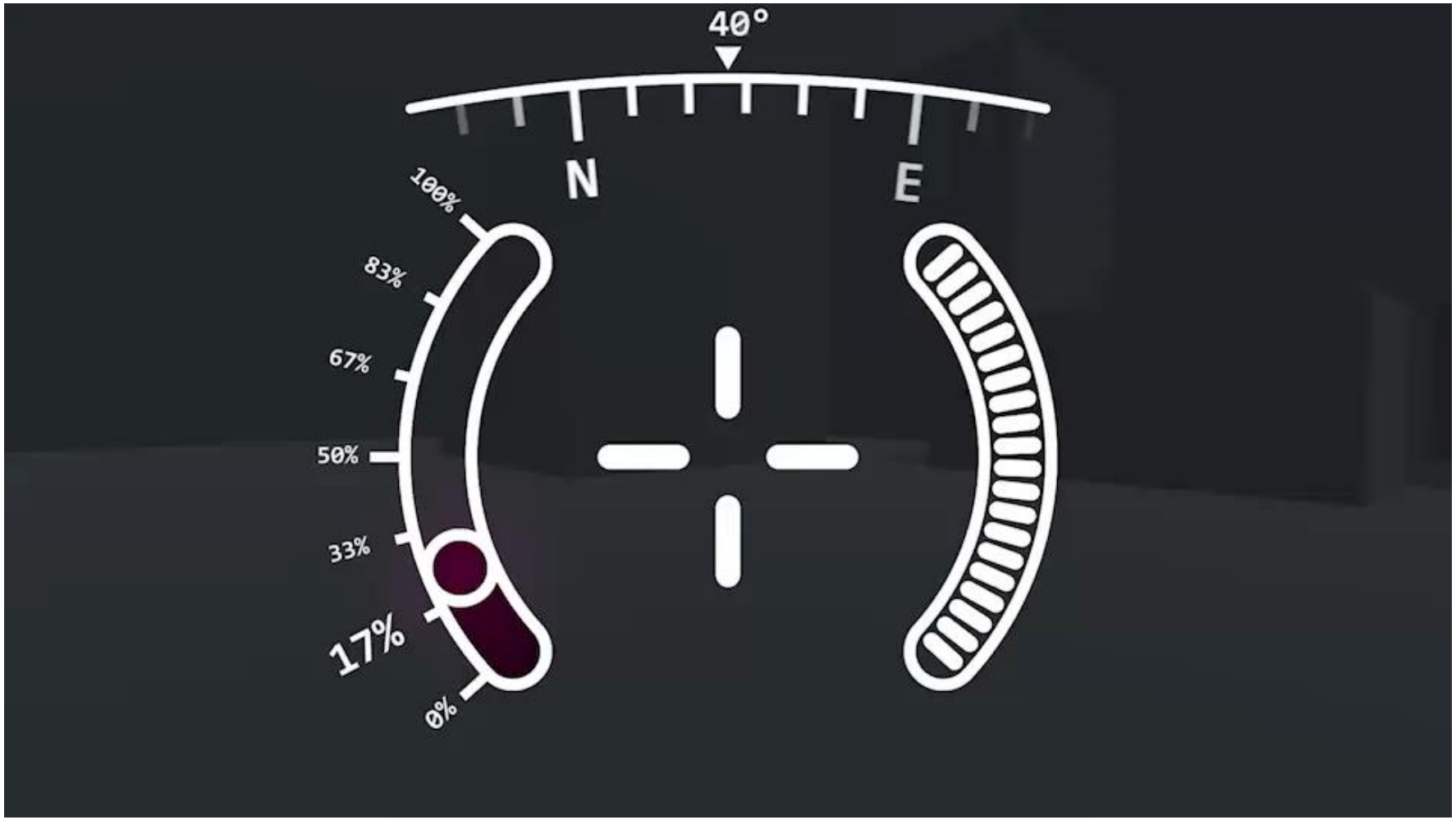
SonoCraftAR renders a personalized AR interface that reacts dynamically to sound.


User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}
Code Checker Agent
Rosyln Compilation
Generated Visualization
Continuous Audio Input

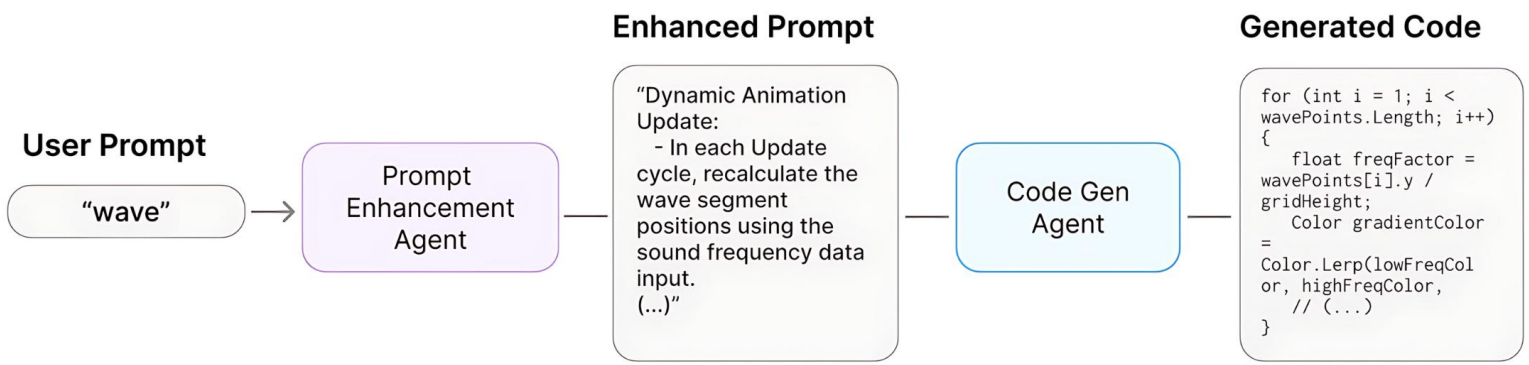
User Prompt

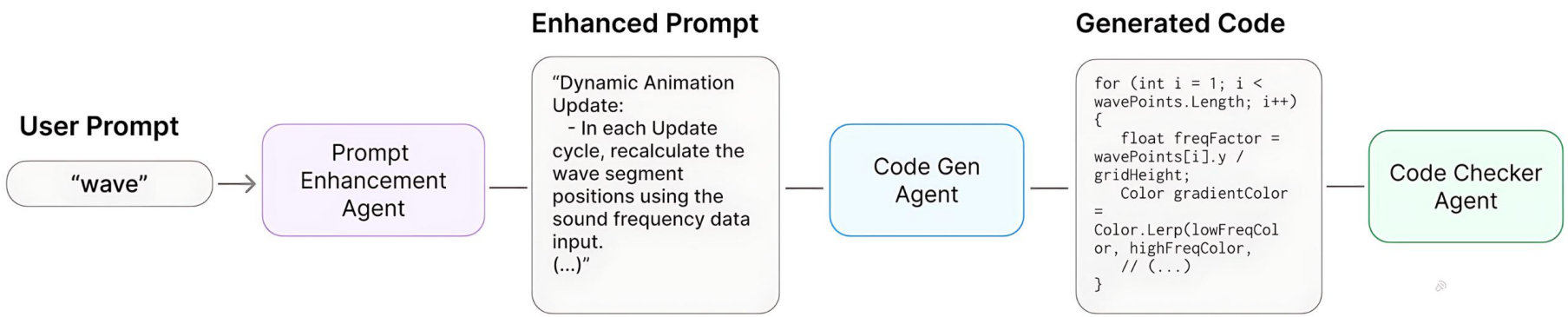
“wave”

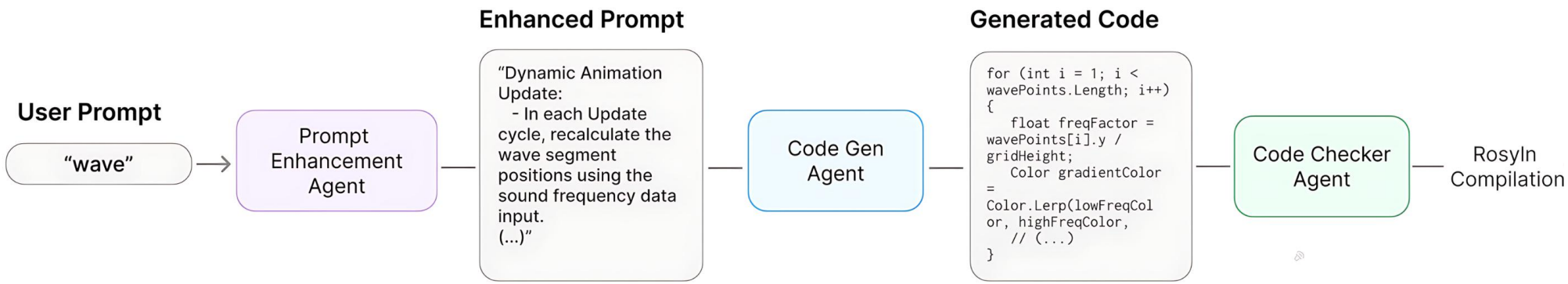

User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”

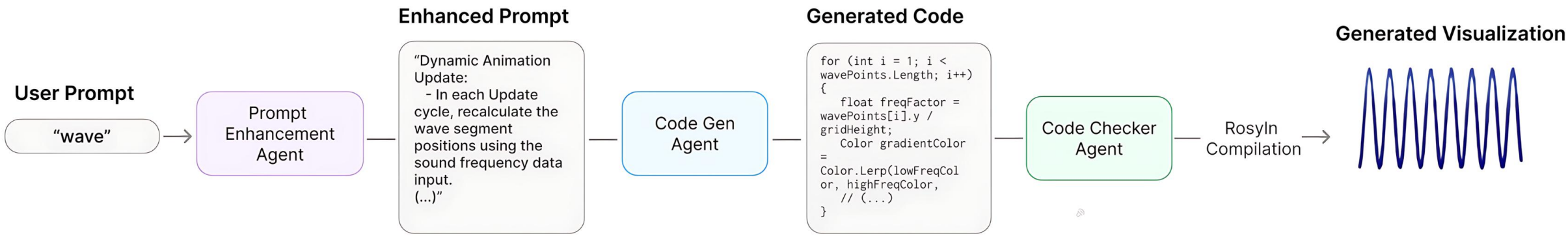


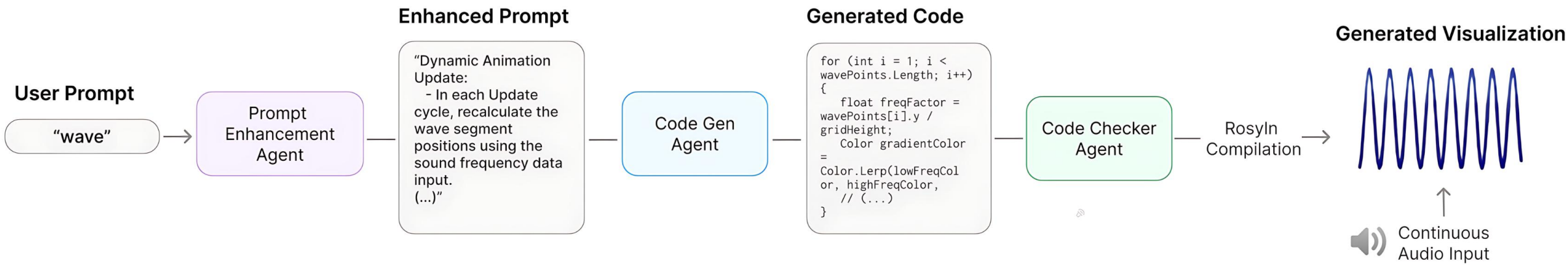
Shapes by Freya Holmér

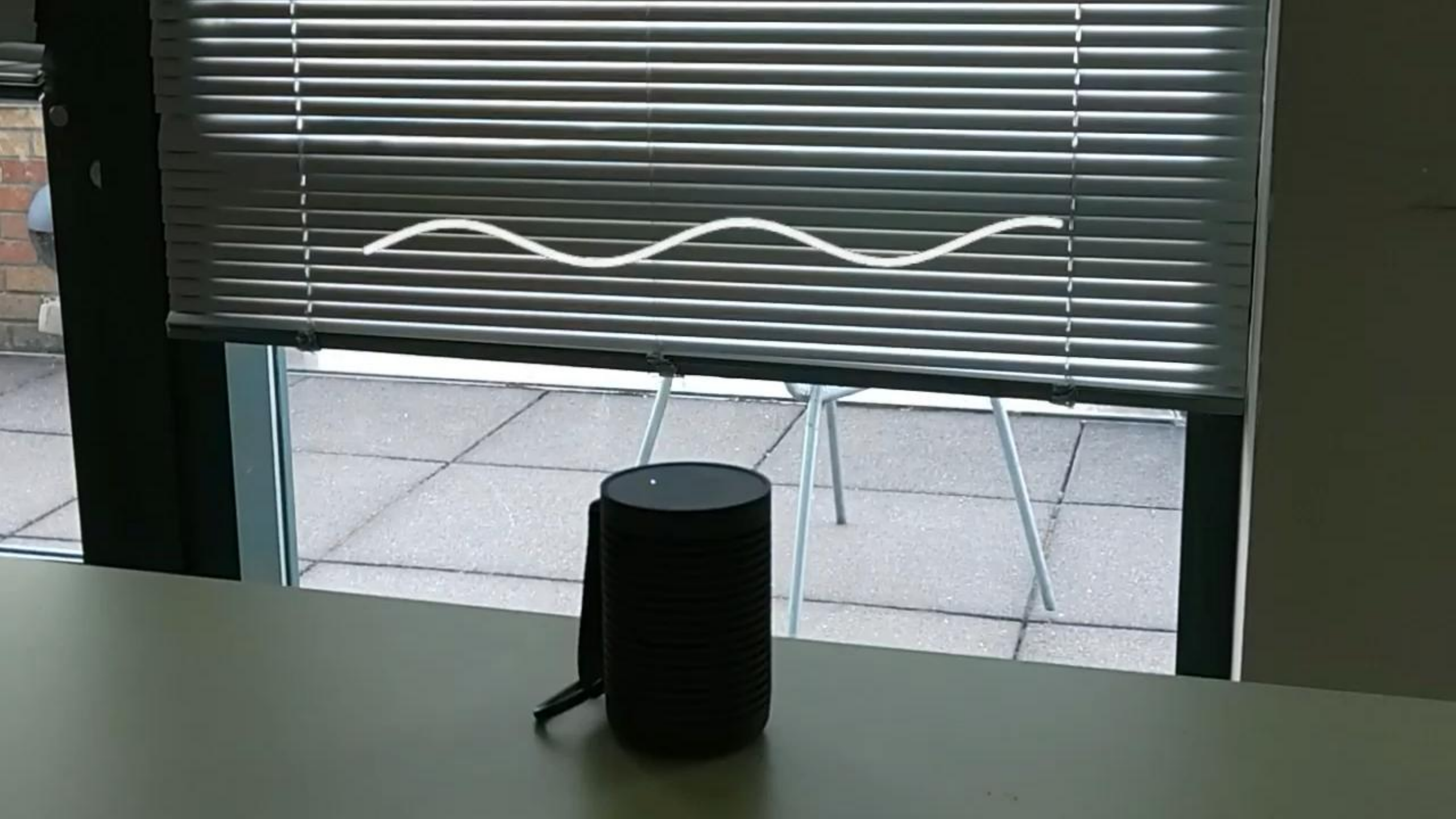

User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}

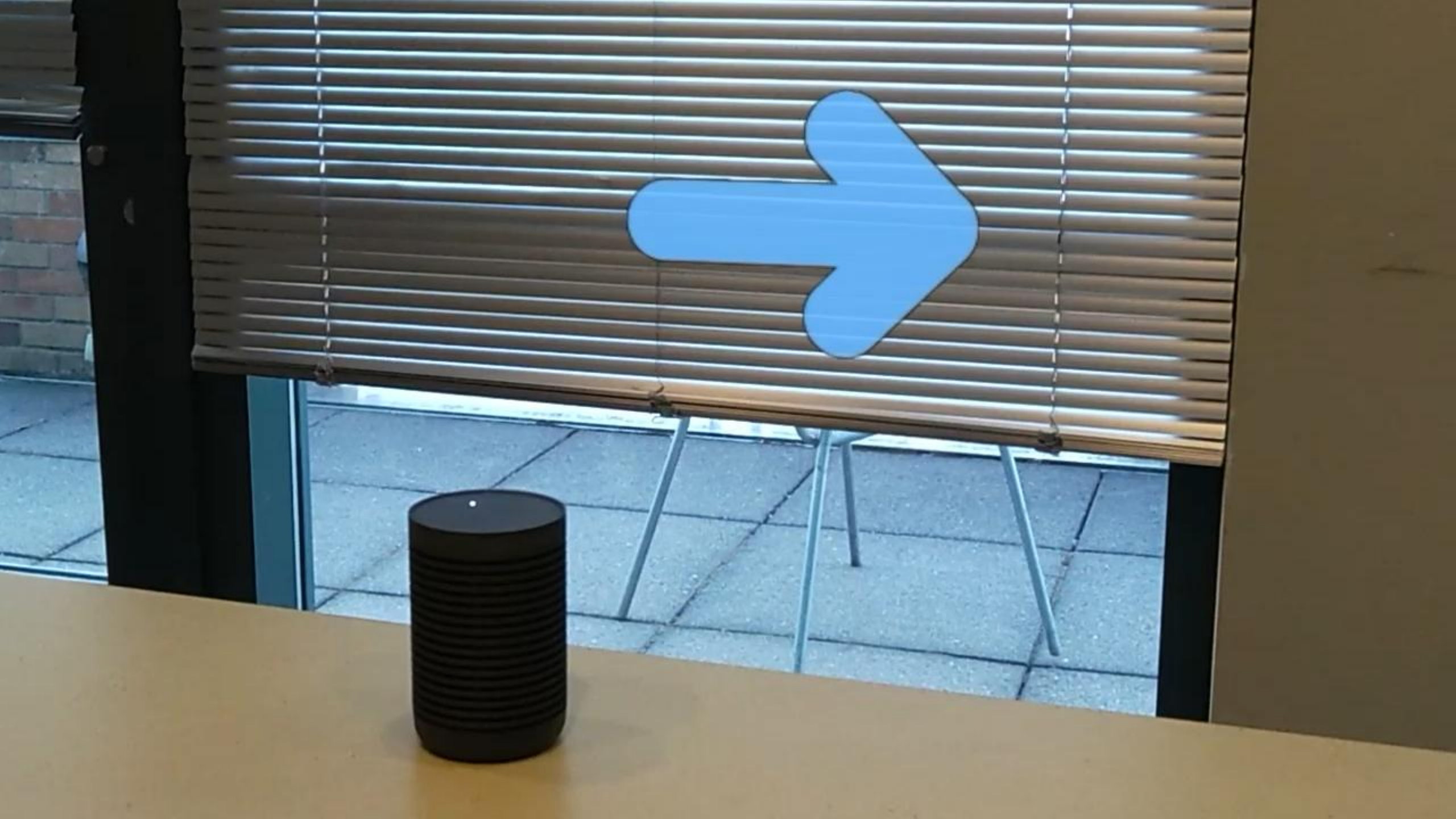

User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}
Code Checker Agent

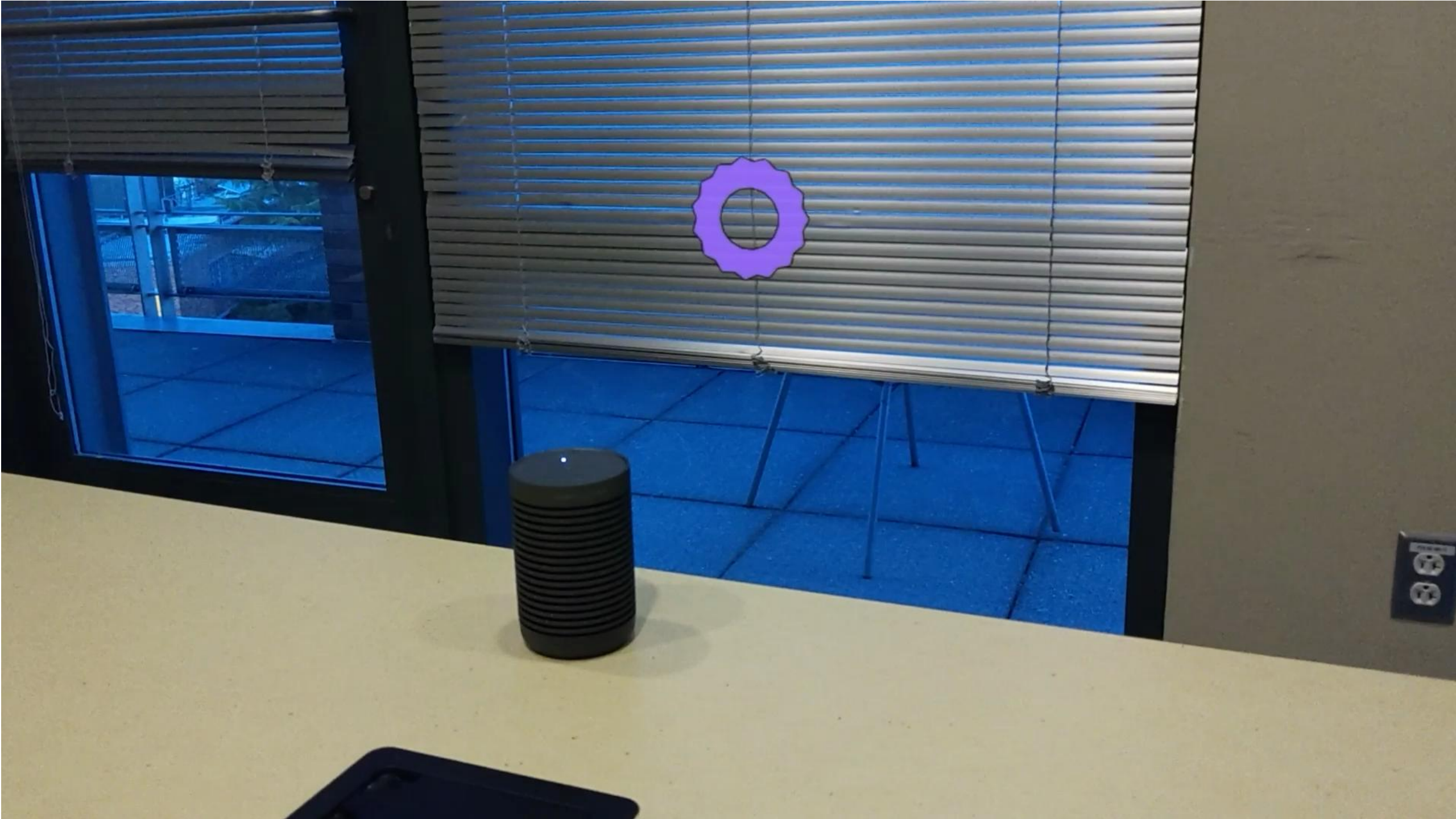

User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}
Code Checker Agent
Rosyln Compilation

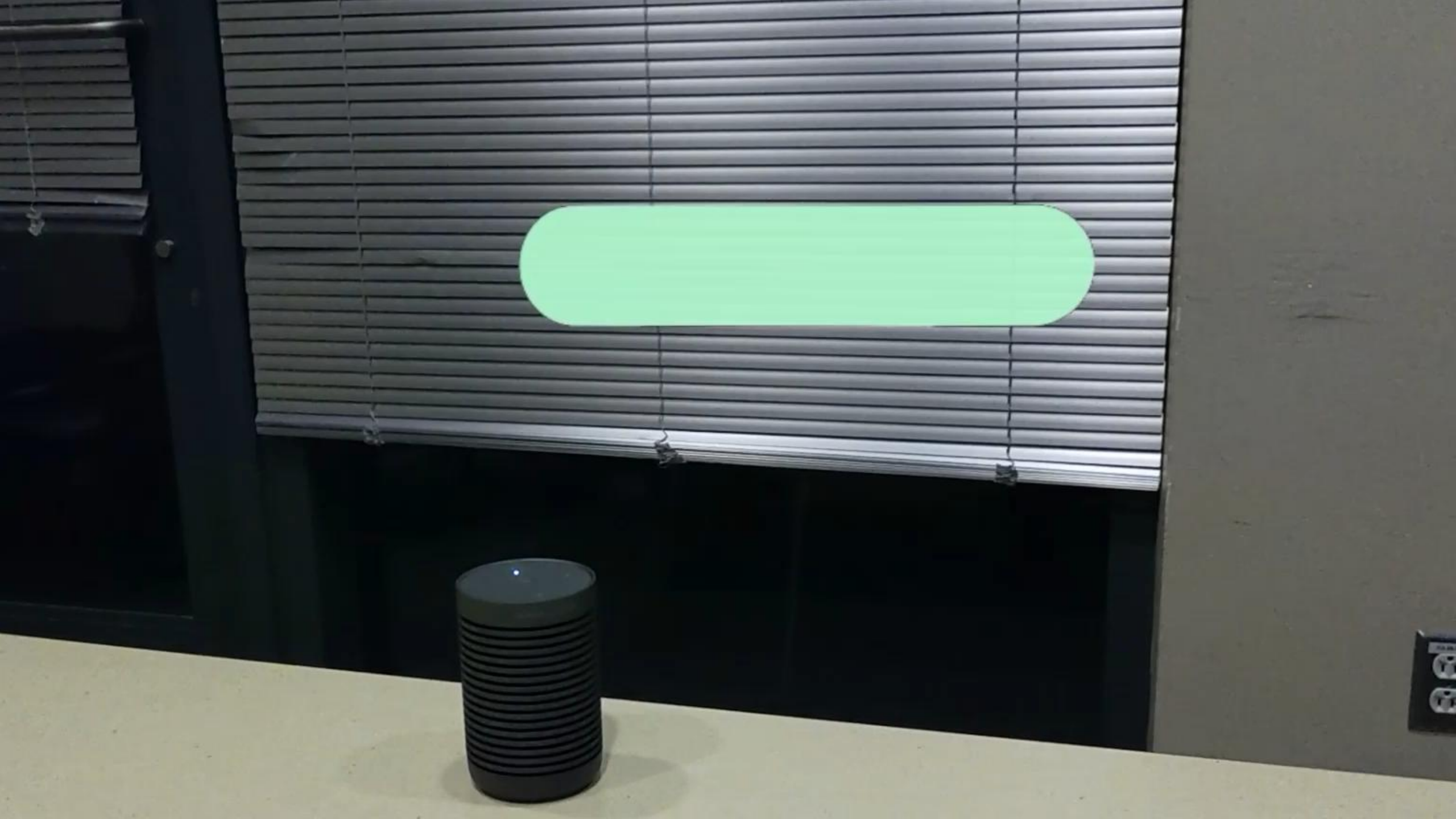

User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}
Code Checker Agent
Rosyln Compilation
Generated Visualization


User Prompt
“wave”
Prompt Enhancement Agent
Enhanced Prompt
“Dynamic Animation Update:
- In each Update cycle, recalculate the wave segment positions using the sound frequency data input.
(...)”
Code Gen Agent
Generated Code
for (int i = 1; i < wavePoints.Length; i++)
{
float freqFactor = wavePoints[i].y / gridHeight;
Color gradientColor = Color.Lerp(lowFreqColor, highFreqColor,
// (...)
}
Code Checker Agent
Rosyln Compilation
Generated Visualization
Continuous Audio Input









Personalization vs. Effective Sound Representation

- **Trade-off:** Balancing creativity and personalization with clear sound representation.
- **SonoCraftAR:** Enables creativity and flexibility but may reduce visual clarity.
- **Curated designs:** Ensure clarity and polish but limit personalization.
- **Future direction:** Hybrid model — curated templates with AI-assisted customization.

Future Work

- Support multiple sound sources and features
- Add more input modalities like sketching and voice
- Add error recovery features like undo
- Provide multiple designs or templates for users to choose from
- Reduce latency for real-time authoring

Thanks for listening!

Let’s Connect!

Email jaewook4@cs.washington.edu

Website https://jaewook-lee.com

Twitter https://twitter.com/jaewook_jae