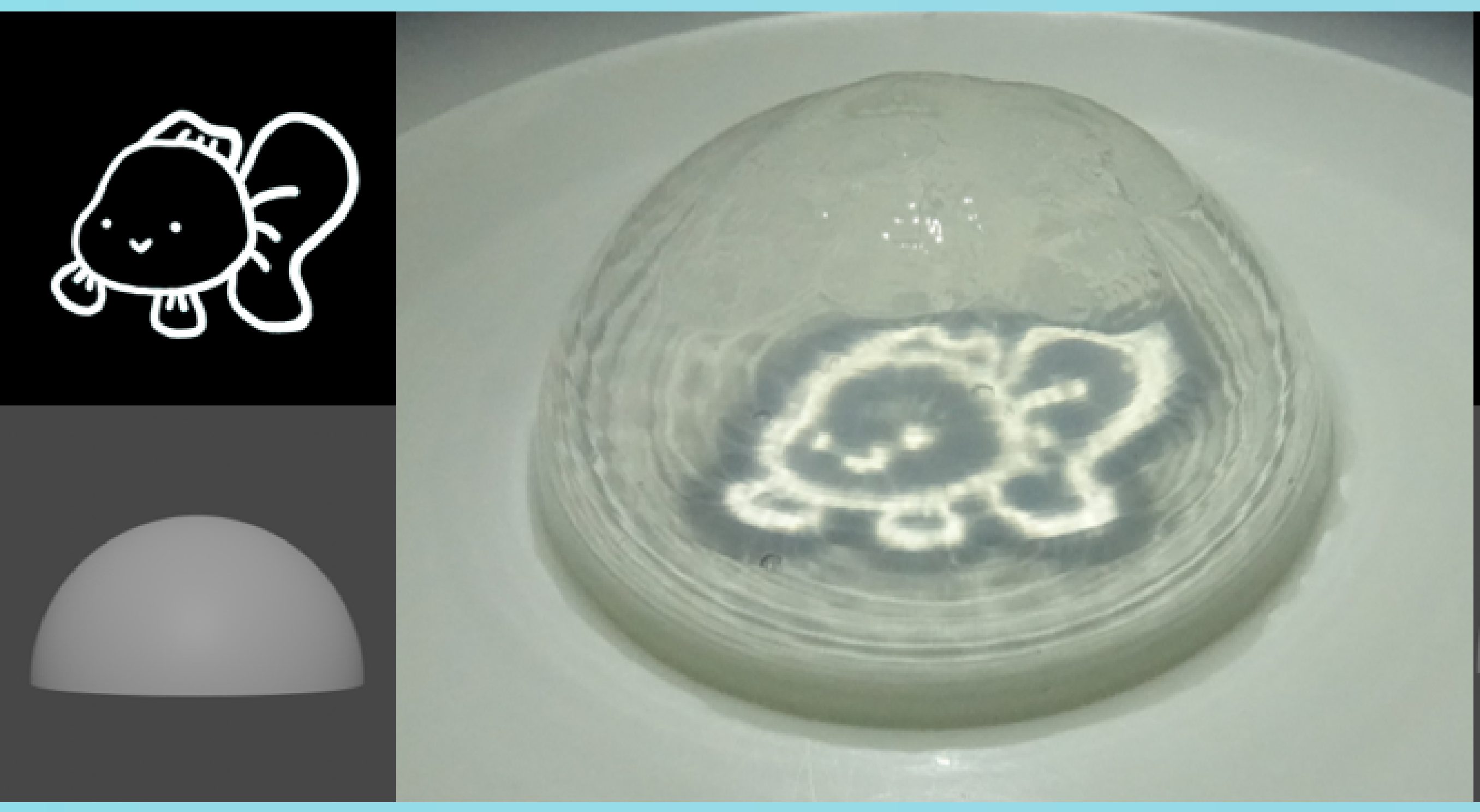
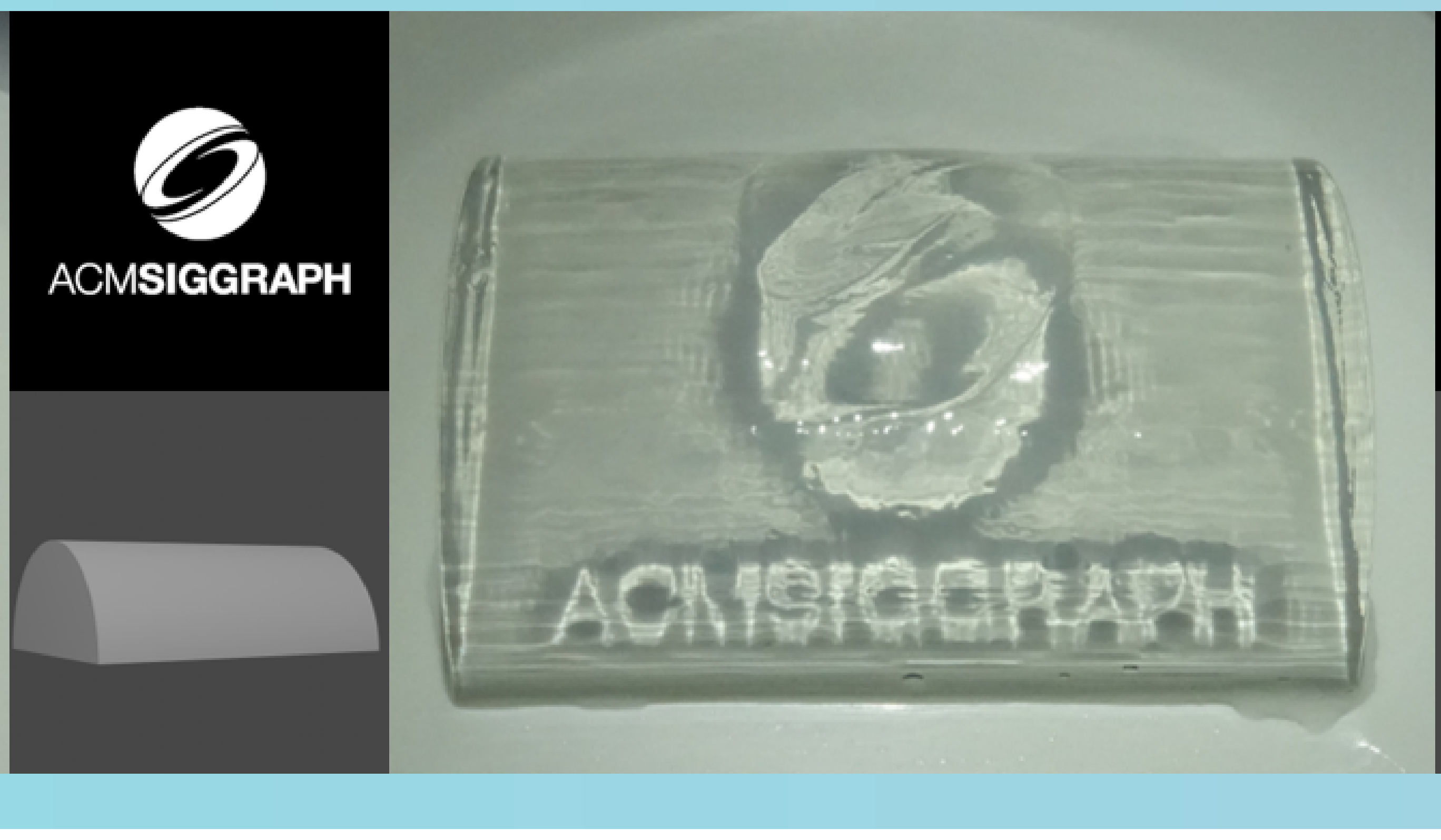
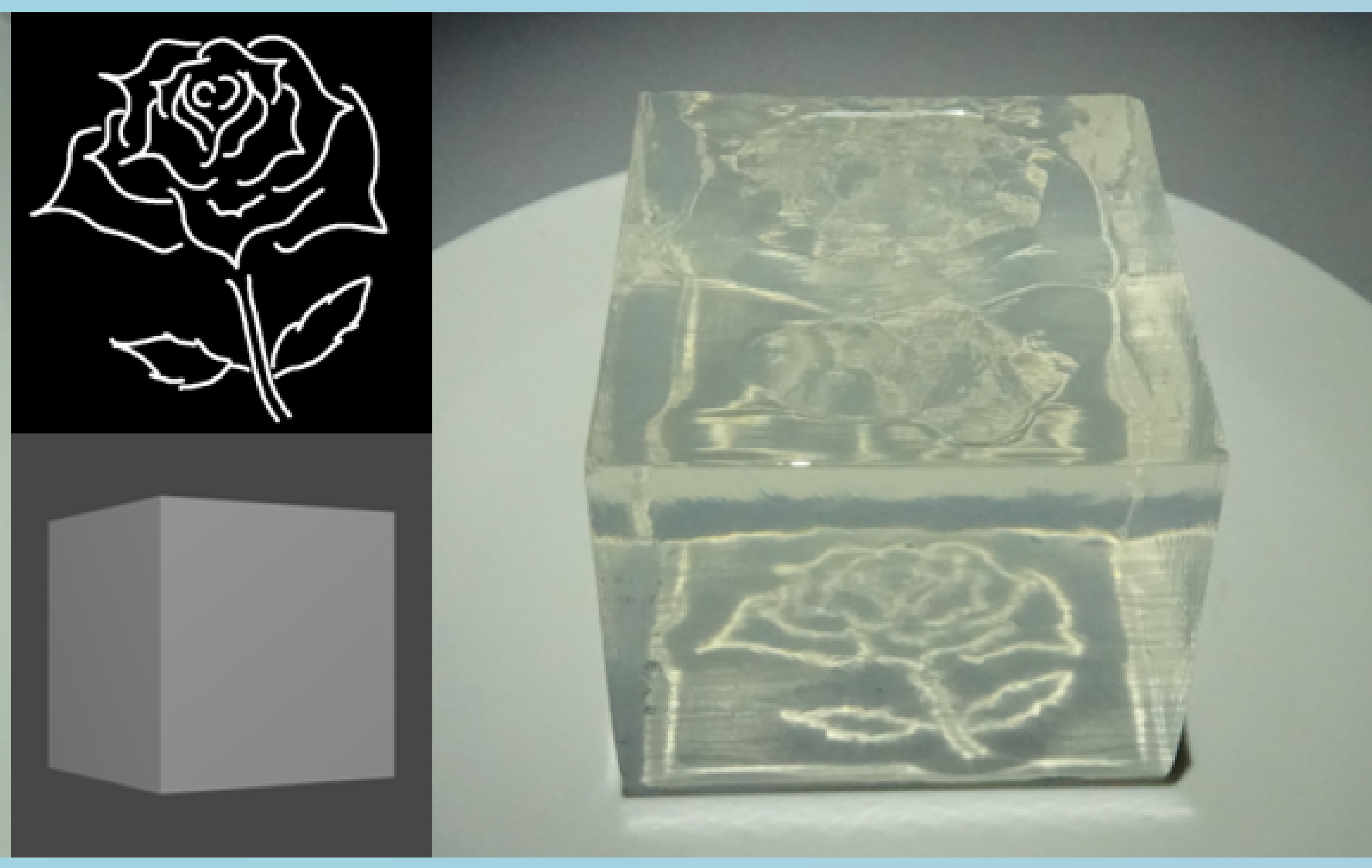
Edible Caustics:
Designing Caustics of Jelly via Differentiable Rendering

Daiki Inokoshi[1], Yuki Yabumoto[1], Junpei Fujikawa[1], Yoshinori Dobashi[2], Takashi Ijiri[1]
[1]Shibaura Institute of Technology [2]Hokkaido University





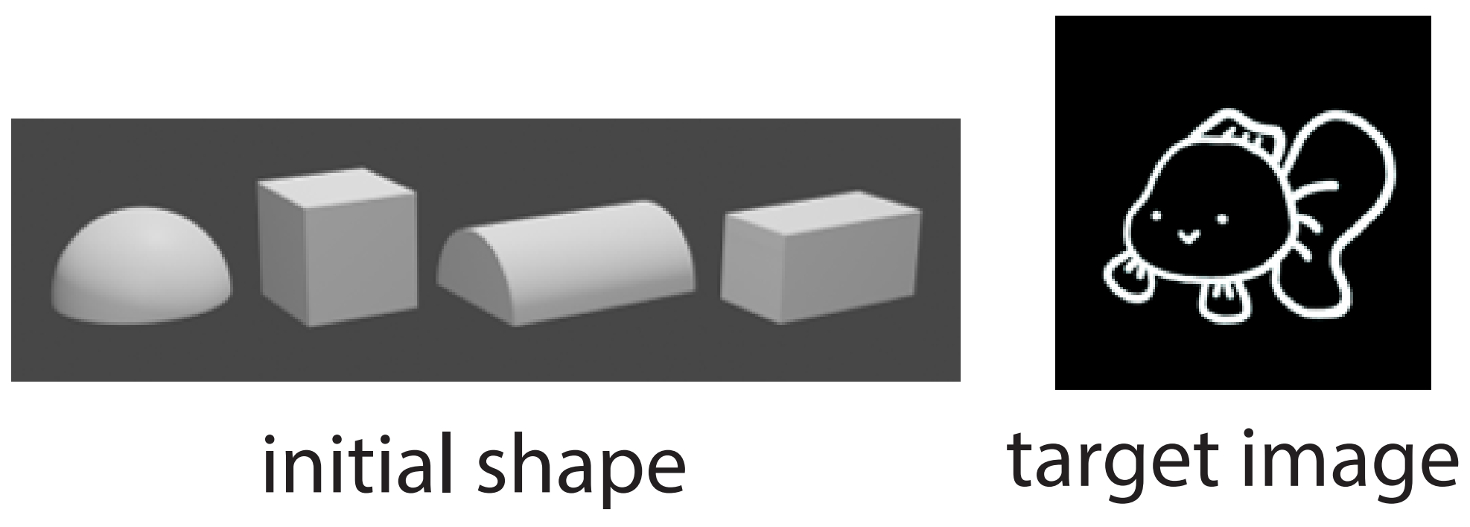


Backgrounds:
Transparent candies, such as jelly, are popular for their beautiful appearance.
They refract incoming light and produce caustic patterns in their surroundings.
However, modeling jelly shapes to provide specific caustic patterns is challenging.

Goal:
To create jellies with desired caustic patterns.

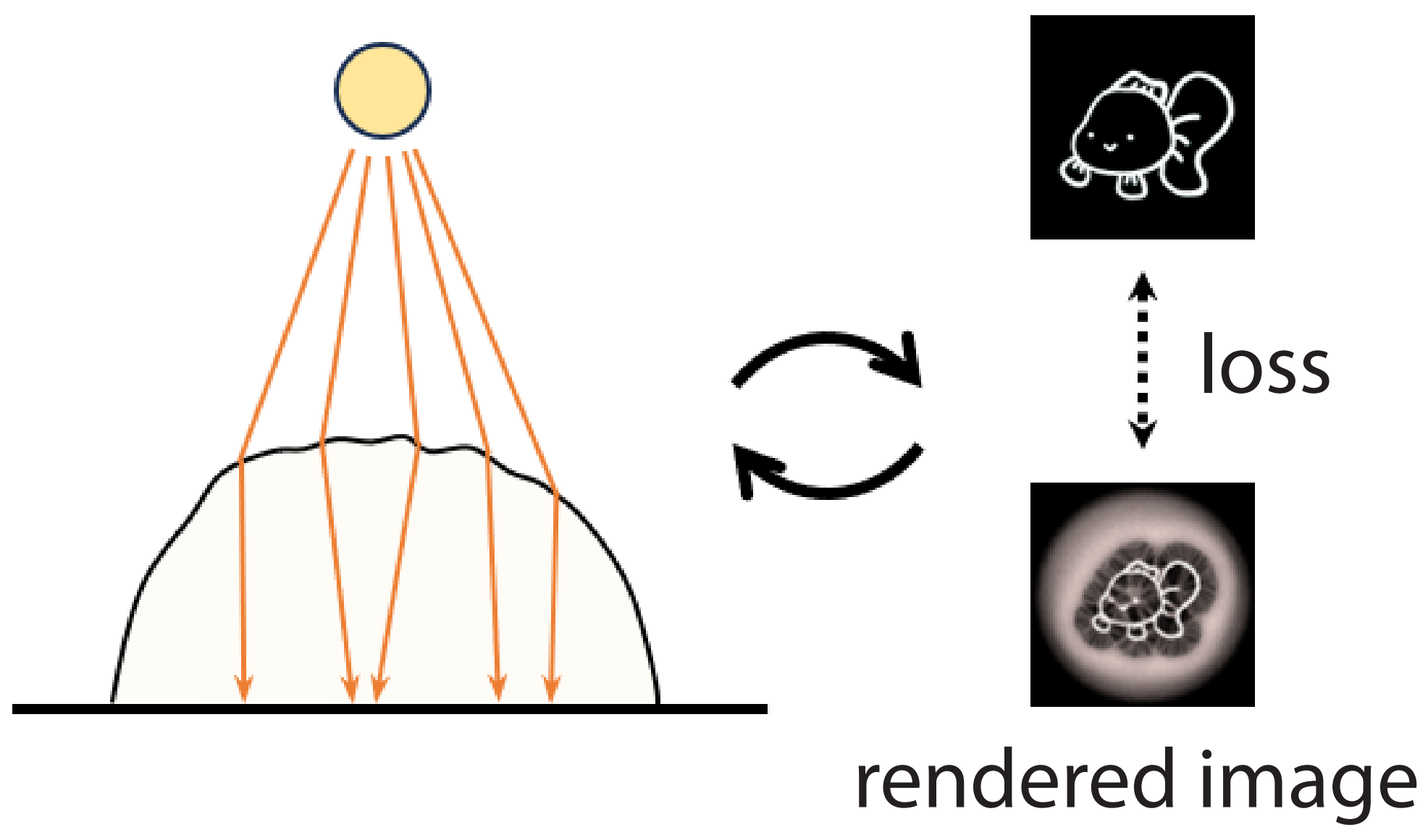
Approach:
We optimize jelly shapes using differentiable rendering to provide desired caustic patterns.

Method



a input

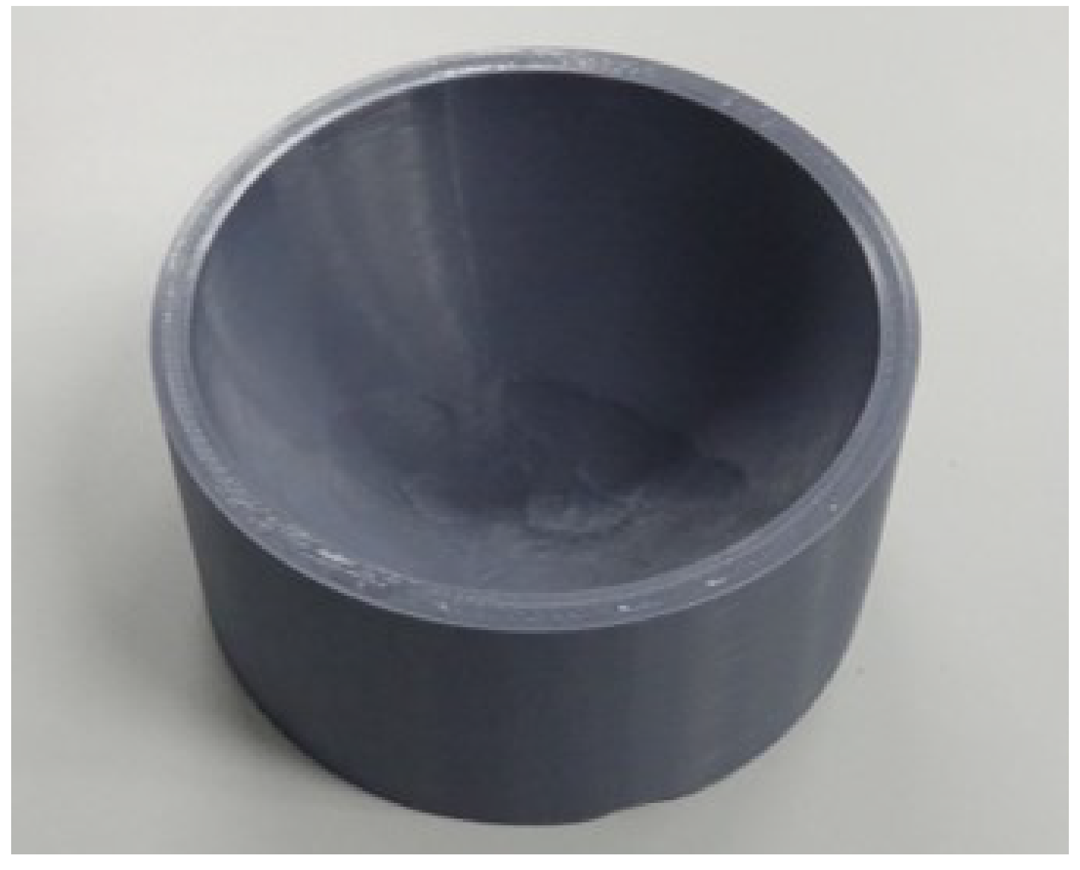
The user selects an initial shape and provides a grayscale target image.



b shape optimization

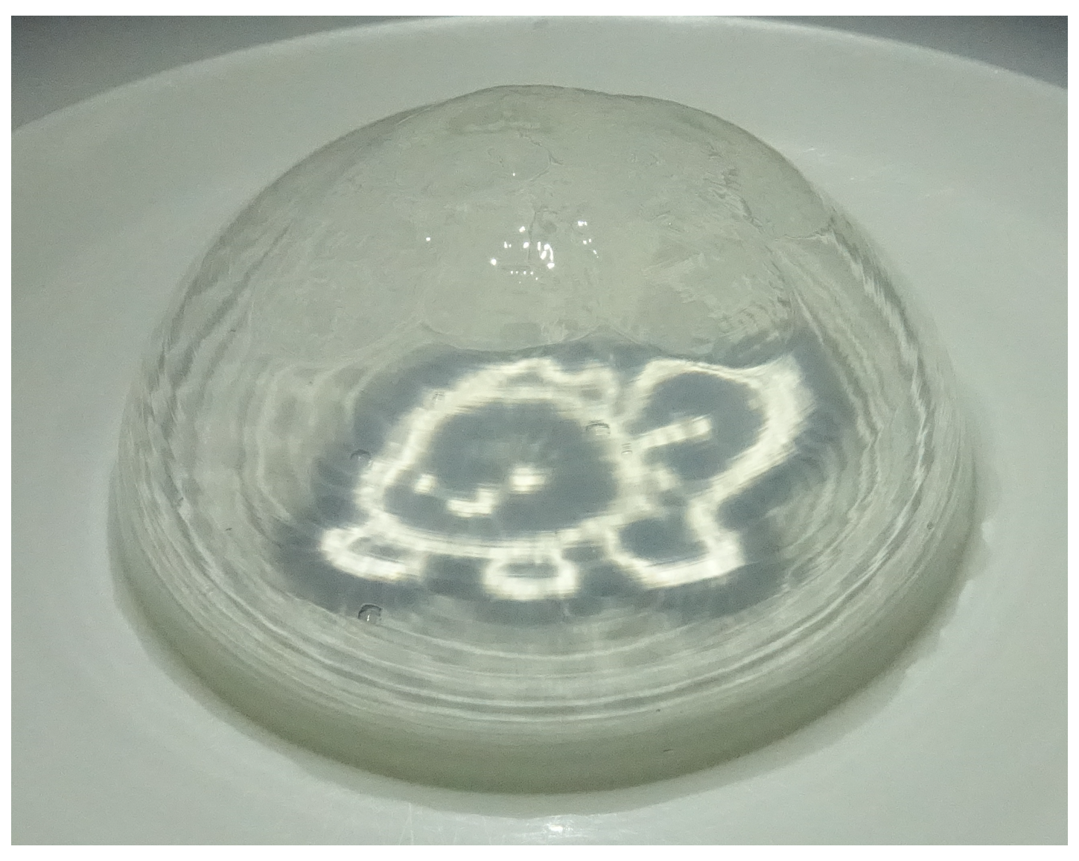
We iteratively update the jelly shape using Mitsuba 3[1] to minimize the following function.

$$\arg\min_{\pi} \underbrace{\|I(\pi) - I_t\|_2^2}_{\text{difference of target and rendered image}} + \underbrace{\lambda \sum_{\mathbf{v}_i \in \pi} \|L(\mathbf{v}_i)\|_2^2}_{\text{smoothing term}}$$



c mold 3D printing

We fabricate a mold of the optimized jelly shape using a 3D printer.



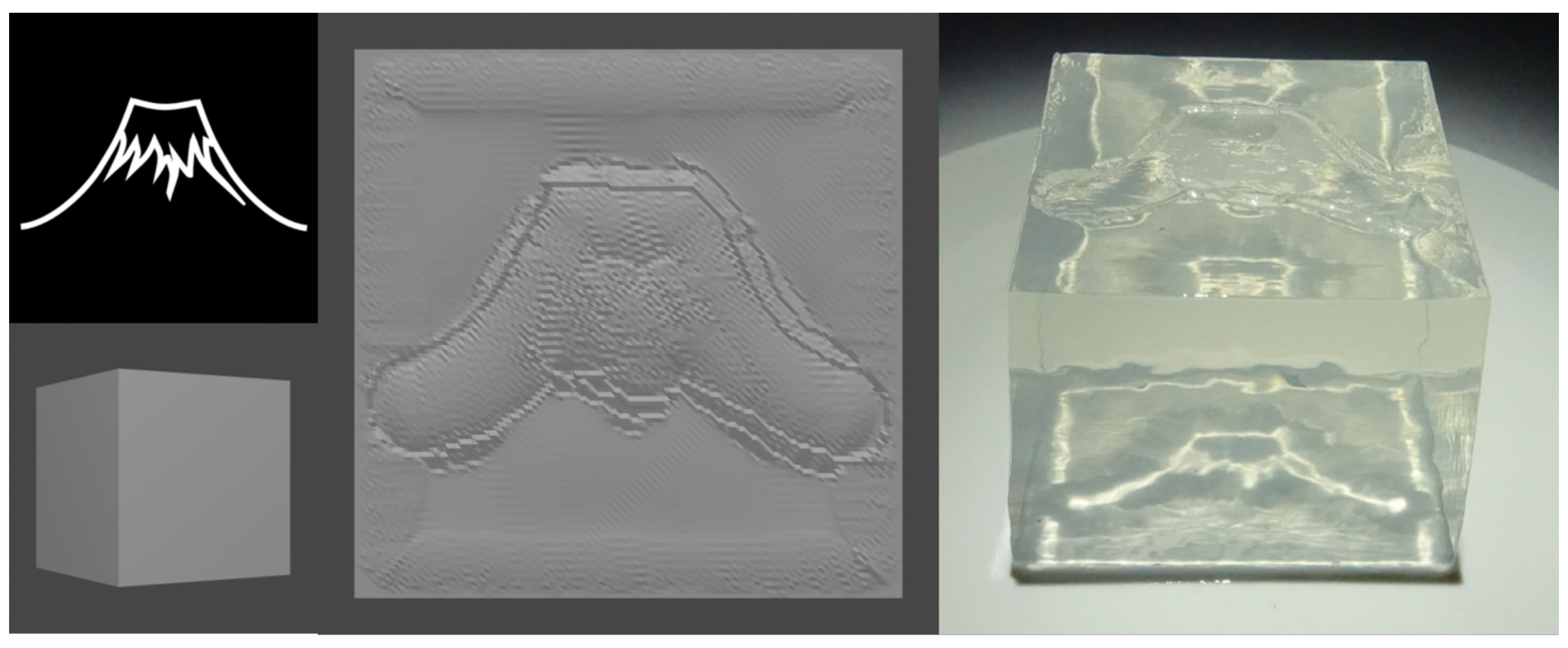
d jelly making

We create jelly using the mold and illuminate it to visualize the caustic pattern.

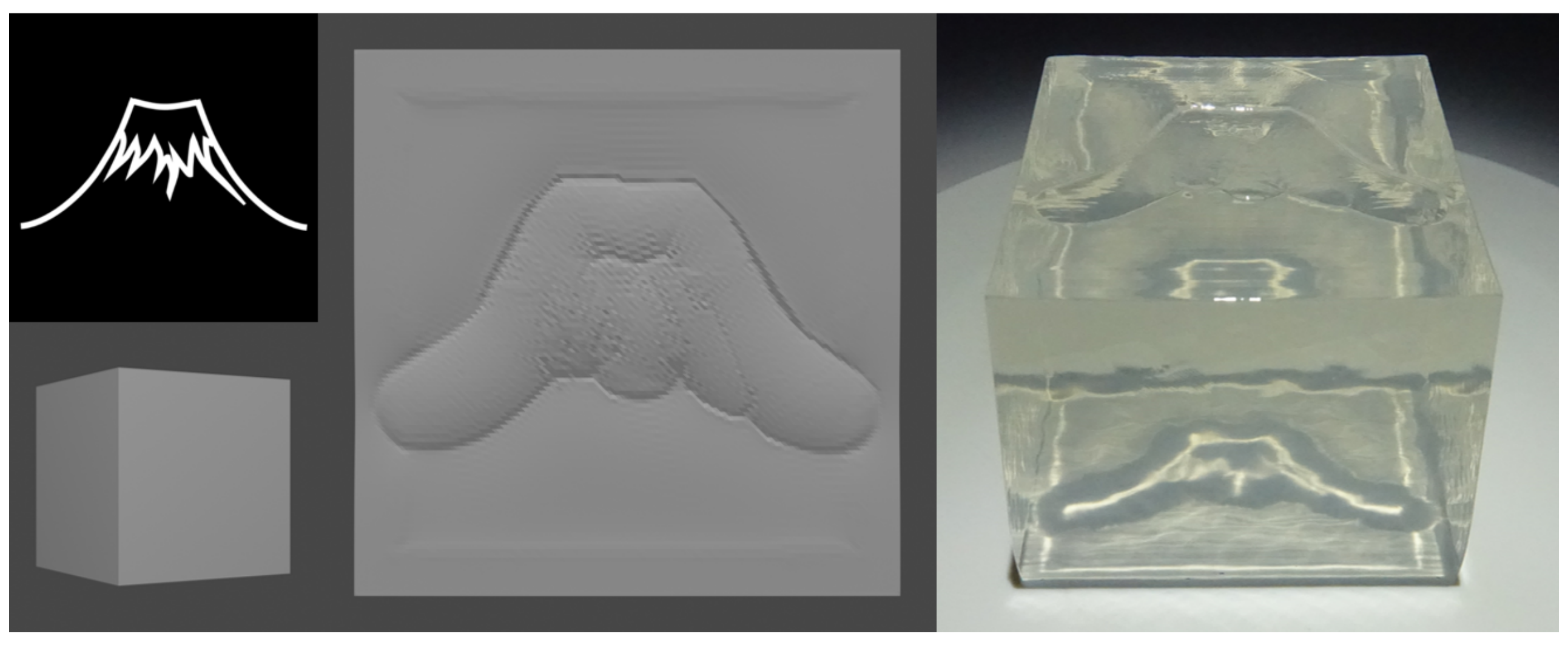
Results and Discussion

- We created jellies with various caustic patterns (see above).
- We produced jellies with and without the smoothing term to evaluate its effectiveness.
- When employing the smoothing term, we obtained a smoother shape, resulting in clearer caustic patterns.

Without smoothing term



With smoothing term



Future Work

- Produce more precise caustic patterns for target images with gradients.
- Generate colored caustic patterns.
- Investigate the minimum thinness and complexity of caustic lines that our method can reproduce.

[1] W. Jakob, S. Speierer, N. Roussel, and D. Vicini. 2022. DR.JIT: A Just-in-Time Compiler for Differentiable Rendering. ACM ToG 41, 4 (2022), 124:1–124:19