LUND UNIVERSITY | LTH FACULTY OF ENGINEERING

Succinct Opacity Micromaps

Gustaf Waldemarson & Michael Doggett
Arm Sweden & Lund University

arm

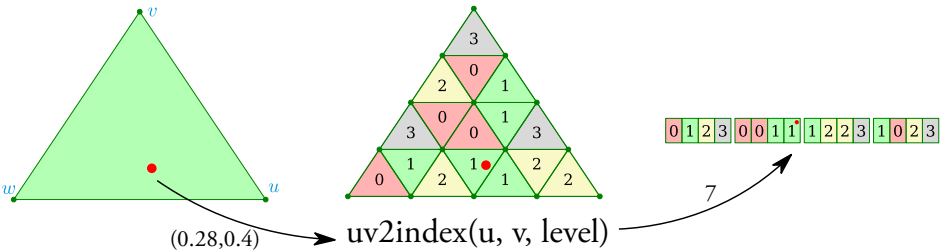
Agenda

1. What are Opacity Micromaps?
   - `uv2index` (`BarycentricsToSpaceFillingCurveIndex`)
2. *Succinct* Opacity Micromaps
   - Memory Footprint Comparison
3. Frametime Performance Evaluation

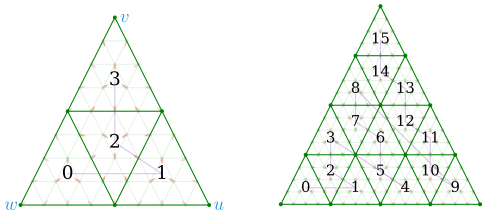
Opacity Micromaps

Opacity Micromaps

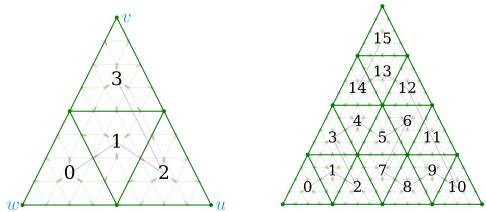
2-State		4-State	
0b0	Fully Transparent	0b00	Fully Transparent
0b1	Fully Opaque	0b01	Fully Opaque
		0b10	Unknown Transparent
		0b11	Unknown Opaque



Micromap Evolution



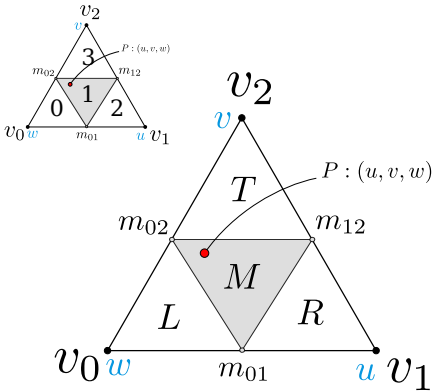
Gruen et al.



Vulkan® & DirectX® [Werness 2022]

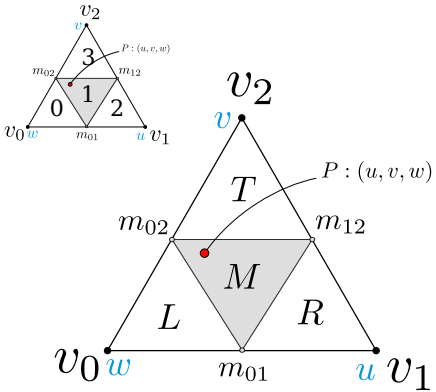
Opacity Micromaps Construction

`uv2index` `BarycentricsToSpaceFillingCurveIndex`



Opacity Micromaps Construction

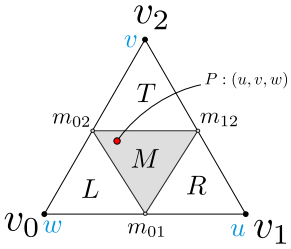
`uv2index` `BarycentricsToSpaceFillingCurveIndex`



$$\begin{cases} v_0 = m_{01} + m_{02} - m_{12} \\ v_1 = m_{01} + m_{12} - m_{02} \\ v_2 = m_{02} + m_{12} - m_{01} \end{cases}$$

Opacity Micromaps Construction

`uv2index` `BarycentricsToSpaceFillingCurveIndex`

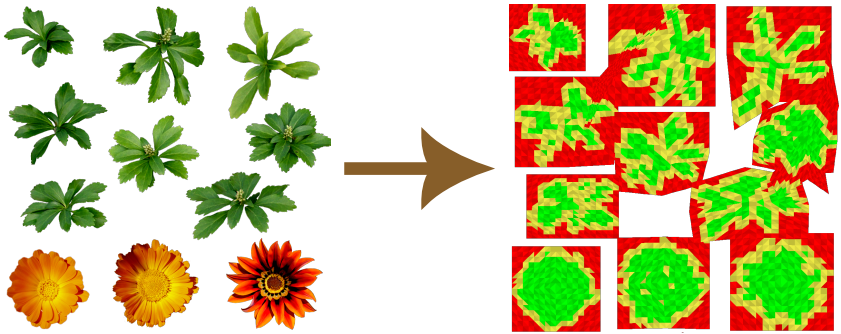


$$L \coloneqq \begin{cases} u_L &= u - v - w \\ v_L &= 2v \\ w_L &= 2w \end{cases} \qquad M \coloneqq \begin{cases} u_M &= u + v - w \\ v_M &= v + w - u \\ w_M &= u + w - v \end{cases}$$

$$R \coloneqq \begin{cases} u_R &= 2u \\ v_R &= v - u - w \\ w_R &= 2w \end{cases} \qquad T \coloneqq \begin{cases} u_T &= 2u \\ v_T &= 2v \\ w_T &= w - u - v \end{cases}$$

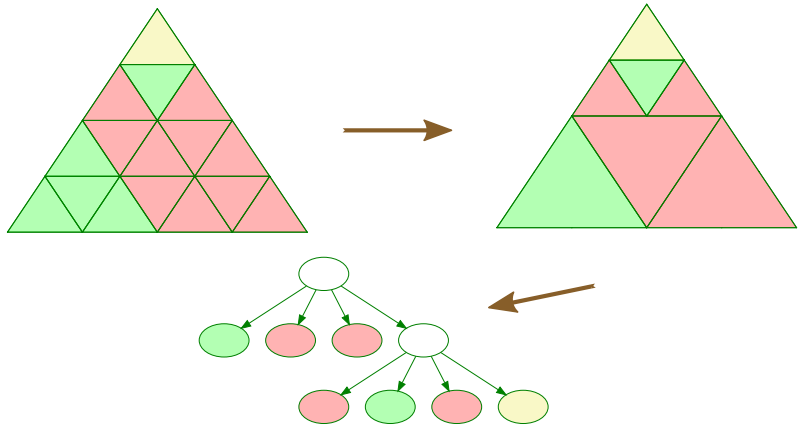
Succinct Opacity Micromaps

Succinct Opacity Micromaps



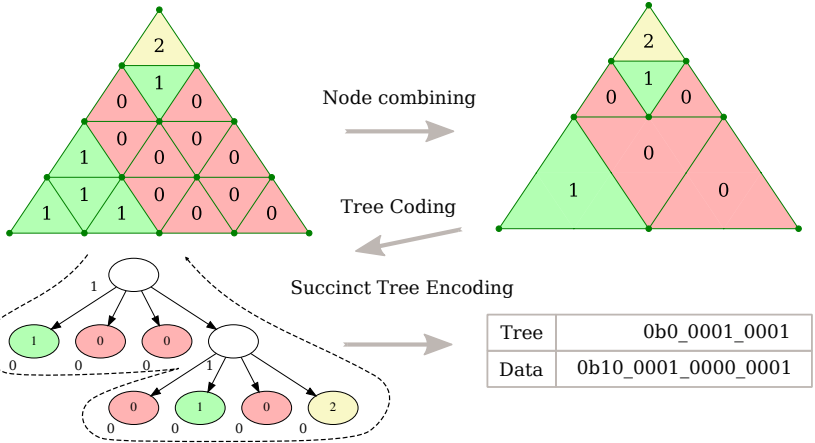
Succinct Opacity Micromaps

Tree Construction



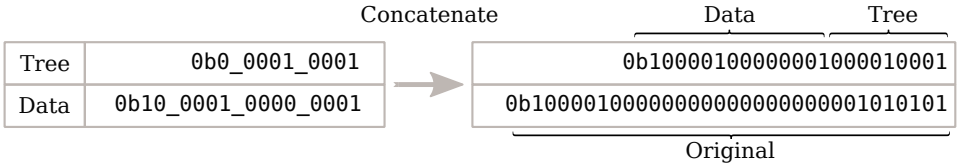
Succinct Opacity Micromaps

Tree Construction

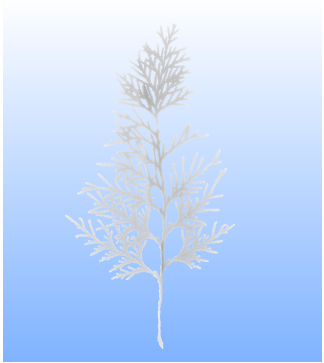


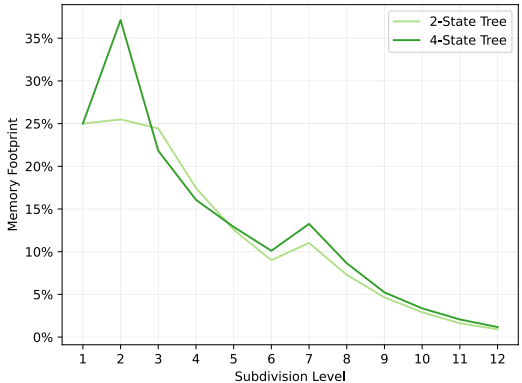
Succinct Opacity Micromaps

Tree Construction



Memory Footprint Improvement

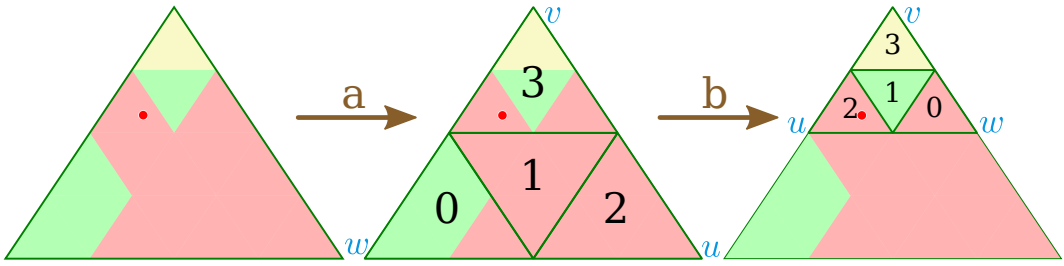


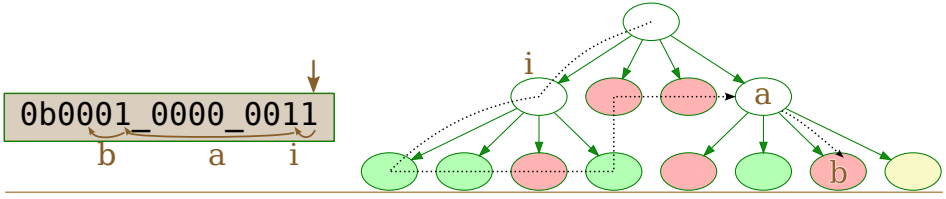


Best ≈ 1 % of the original size.

Succinct Opacity Micromaps

Look-up Algorithm

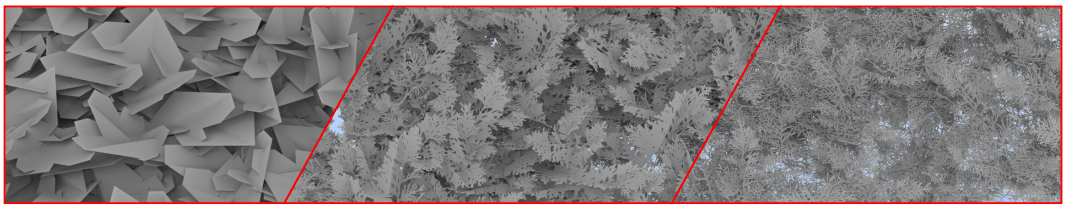




Frametime Performance

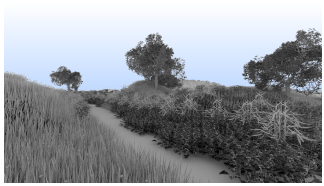
Methods

- Software Micromaps
- Succinct Tree
- Fast-Build Micromaps
- Fast-Trace Micromaps
- Bitmask
- Texture

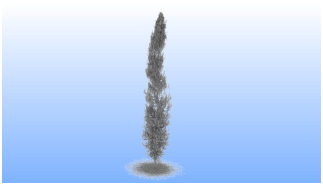


Scenes

CryTek Sponza [2011]

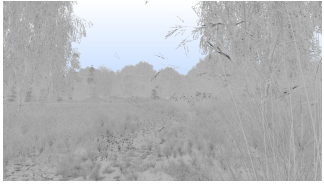


Ecosys [1998]



New Sponza [2022]

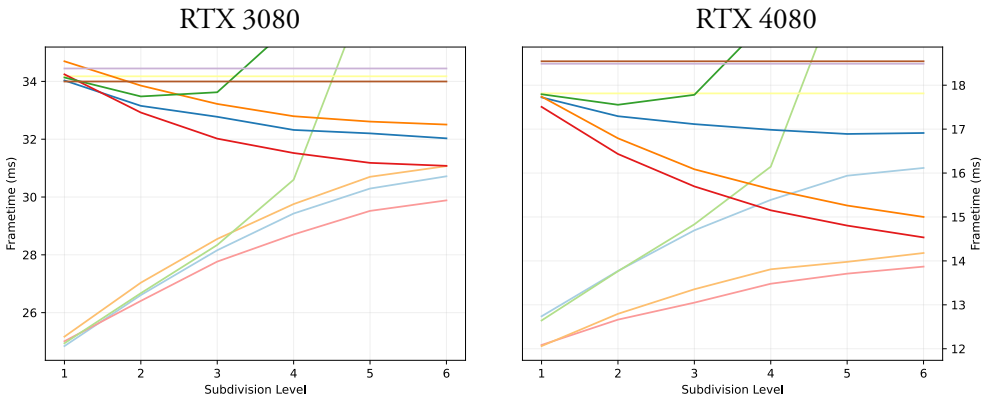
San Miguel [2010]



Landscape [2016]

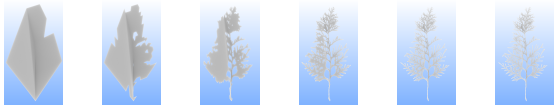
Frametime Performance

General Results — Landscape [2016]

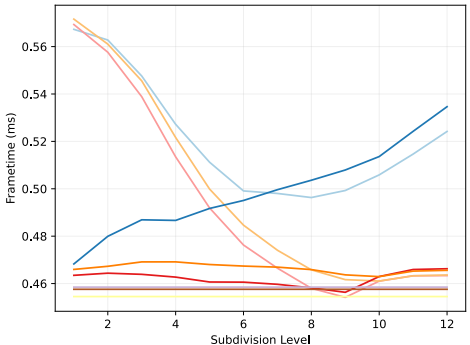


Frametime Performance

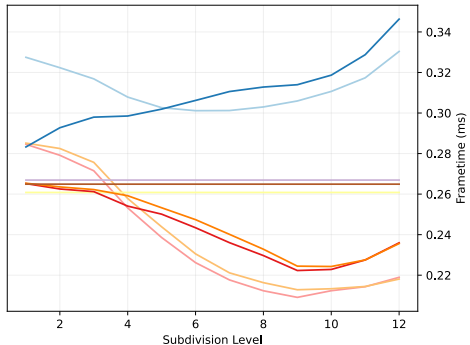
New Sponza [2022]



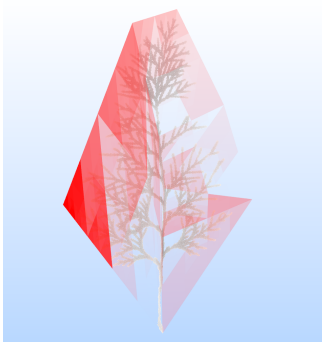
RTX 3080

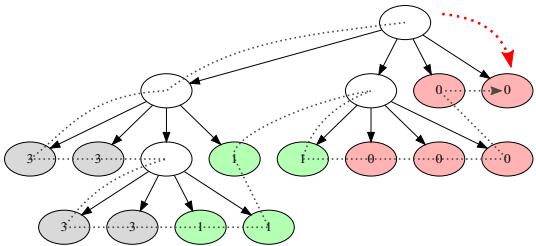


RTX 4080



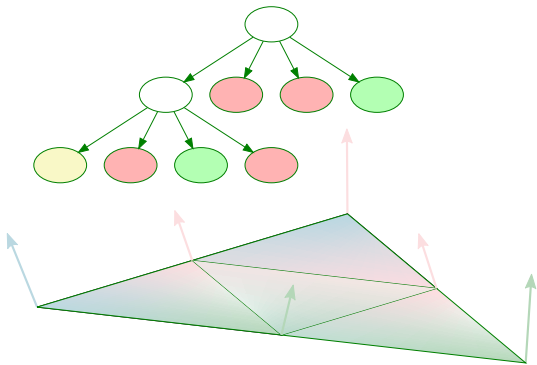
The Problem(s)





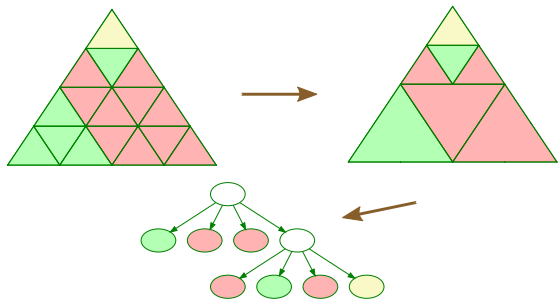
Future Work

- Need a different (succinct) tree
- Other micromap types?
  - Lossy micromaps?
  - Generalized micromaps?



Conclusion

1. Indexing Algorithm:
   - `uv2index`
2. Succinct Opacity Micromaps
3. Performance Comparison

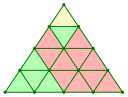
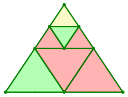


Thanks for Listening

Questions

- Thanks for listening!
- Questions and Answers

Acknowledgements


Succinct Opacity Micromaps

Gustaf Waldemarson Michael Doggett

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

Special thanks to…
- Rikard Olajos @ LUGG
- Simone Pellegrini @ Arm
- Mathieu Robart @ Arm

arm

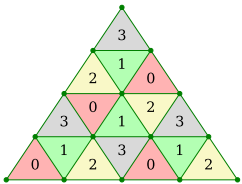
arm

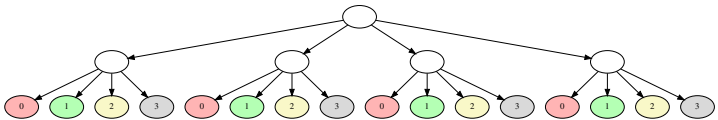
The End

Extras

Where may we *Lose* performance?

Degenerate Trees

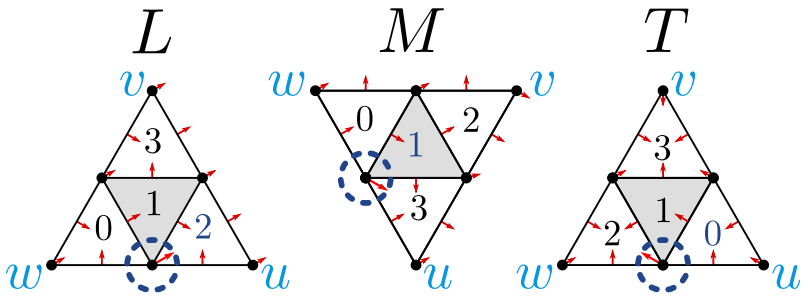




Example degenerate case for the succinct tree compression: Instead of actually reducing the number of nodes, it is forced to add 5 internal ones.

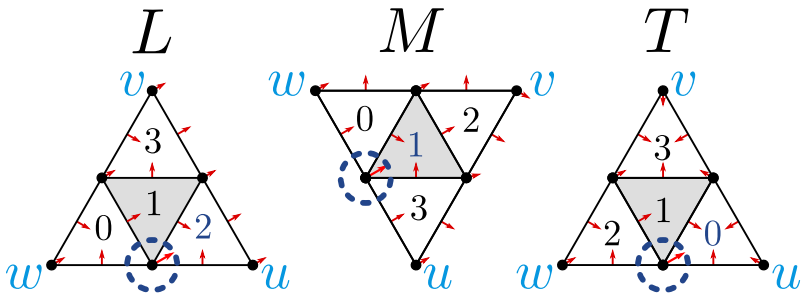
Opacity Micromaps Splits

Rounding Issues



Opacity Micromaps Splits

Rounding Issues



Opacity Micromaps Lookup Function – 1

```
def uv2index(u, v, level):
    w = 1.0 - (u + v)
    def rec(idx, d, u, v, w):
        if d == level:
            return idx
        L, M, R, T = 0, 1, 2, 3
        if w > 0.5:
            return rec(4 * idx + L, d + 1, 2*u, 2*v, (w - u - v))
        elif v >= 0.5:
            return rec(4 * idx + T, d + 1, 2*w, (v - u - w), 2*u)
        elif u >= 0.5:
            return rec(4 * idx + R, d + 1, (u - v - w), 2*v, 2*w)
        else:
            return rec(4 * idx + M, d + 1, (u + v - w), (w + u - v), (v + w - u))
    return rec(0, 0, u, v, w)
```

Opacity Micromaps Lookup Function – 2

```
def uv2index(u, v, level):
    w = (1.0 - (u + v))
    def rec(i, d, mflip, tflip, u, v, w):
        if d == level: return i
        L, M, R, T = (2, 1, 0, 3) if tflip else (0, 1, 2, 3)
        if w > 0.5:
            return rec(4 * i + L, d + 1, mflip, tflip, 2*u, 2*v, (w - u - v))
        elif v >= 0.5 and not (v == 0.5 and mflip):
            return rec(4 * i + T, d + 1, mflip, not tflip, 2*u, (v - u - w), 2*w)
        elif u >= 0.5 and not (v == 0.5 and mflip):
            return rec(4 * i + R, d + 1, mflip, tflip, (u - v - w), 2*v, 2*w)
        else:
            return rec(4 * i + M, d + 1, not mflip, tflip,
                       (u + v - w), (w + u - v), (v + w - u))
    return rec(0, 0, False, False, u, v, w)
```

Succinct Opacity Micromaps

Look-up Algorithm — 1

```
uint t = 0, d = 0;
while (true)
{
    bool is_internal = tree_bit(t);
    if (is_internal)
    {
        t += 1;
        uint c = step();
        t, d = bitscan(tree_len, c, t, d);
    }
    else
    {
        return opacity_value(tree_len, d);
    }
}
```

Succinct Opacity Micromaps

Look-up Algorithm — 2

```
uvec2 bitscan(uint tree_len, uint child, uint t, uint d)
{
    while (t < tree_len && c > 0)
    {
        bool is_internal = opacity_tree_bit(t);
        if (is_internal)
            child += 4;
        else
            d += 1;
        t += 1;
        child -= 1;
    }
    return uvec2(t, d);
}
```

References I

- Oliver Deussen, Pat Hanrahan, Bernd Lintermann, Radomı⊠r Mech, Matt Pharr, and Przemyslaw Prusinkiewicz. 1998. "Realistic modeling and rendering of plant ecosystems." In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH '98). Association for Computing Machinery, New York, NY, USA, 275–286. ISBN: 0897919998. DOI: 10.1145/280814.280898.
- Morgan McGuire Frank Meinl Marko Dabrovic. 2011. *CryTek Sponza*. https://www.cryengine.com/asset-db/product/crytek/sponza-sample-scene. (2011).
- Holger Gruen, Carsten Benthin, and Sven Woop. Aug. 2020. "Sub-Triangle Opacity Masks for Faster Ray Tracing of Transparent Objects." *Proc. ACM Comput. Graph. Interact. Tech.*, 3, 2, (Aug. 2020). DOI: 10.1145/3406180.
- Timm Dapper Jan-Walter Schliep Burak Kahraman. 2016. *Landscape*. https://www.laubwerk.com. (2016).
- Guillermo M. Leal Llaguno. 2010. *San Miguel*. https://www.pbrt.com. (2010).

arm

LUND UNIVERSITY

References II

Frank Meinl, Katica Putica, Cristiano Siqueria, Timothy Heath, Justin Prazen, Sebastian Herholz, Bruce Cherniak, and Anton Kaplanyan. 2022. *Intel Sample Library*. https://www.intel.com/content/www/us/en/developer/topic-technology/graphics-processing-research/samples.html. (2022).

Eric Werness. Aug. 24, 2022. *VK_EXT_opacity_micromap*. The Khronos Group Inc. (Aug. 24, 2022). Retrieved May 11, 2023 from `https://registry.khronos.org/vulkan/specs/1.3-extensions/man/html/VK_EXT_opacity_micromap.html`.

arm

LUND UNIVERSITY