Real-time GPU-based river surface simulation

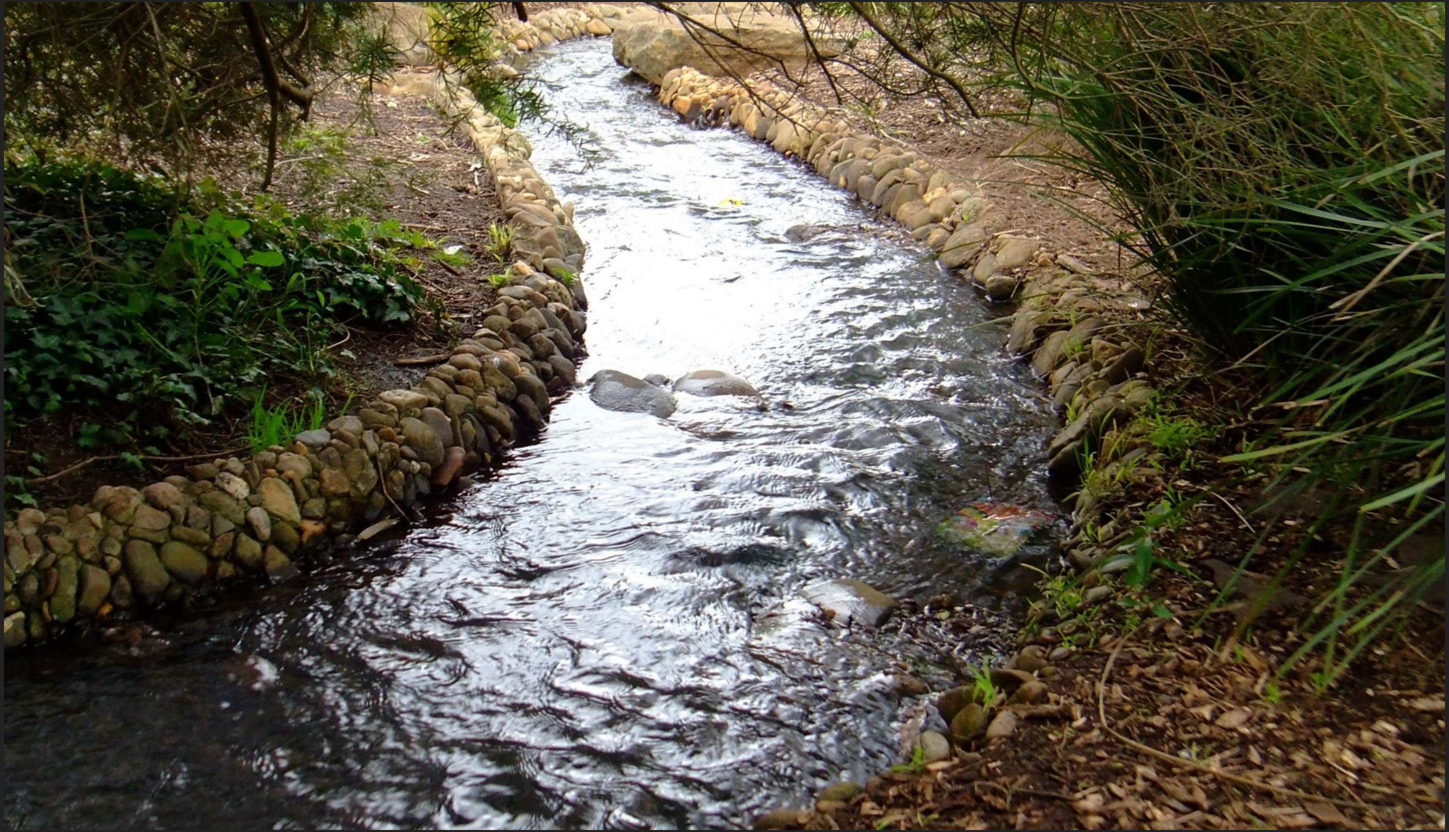
TER defence of Philip SCALES
Supervisor: Fabrice NEYRET, team MAVERICK (LJK/INRIA)

Introduction - Topic

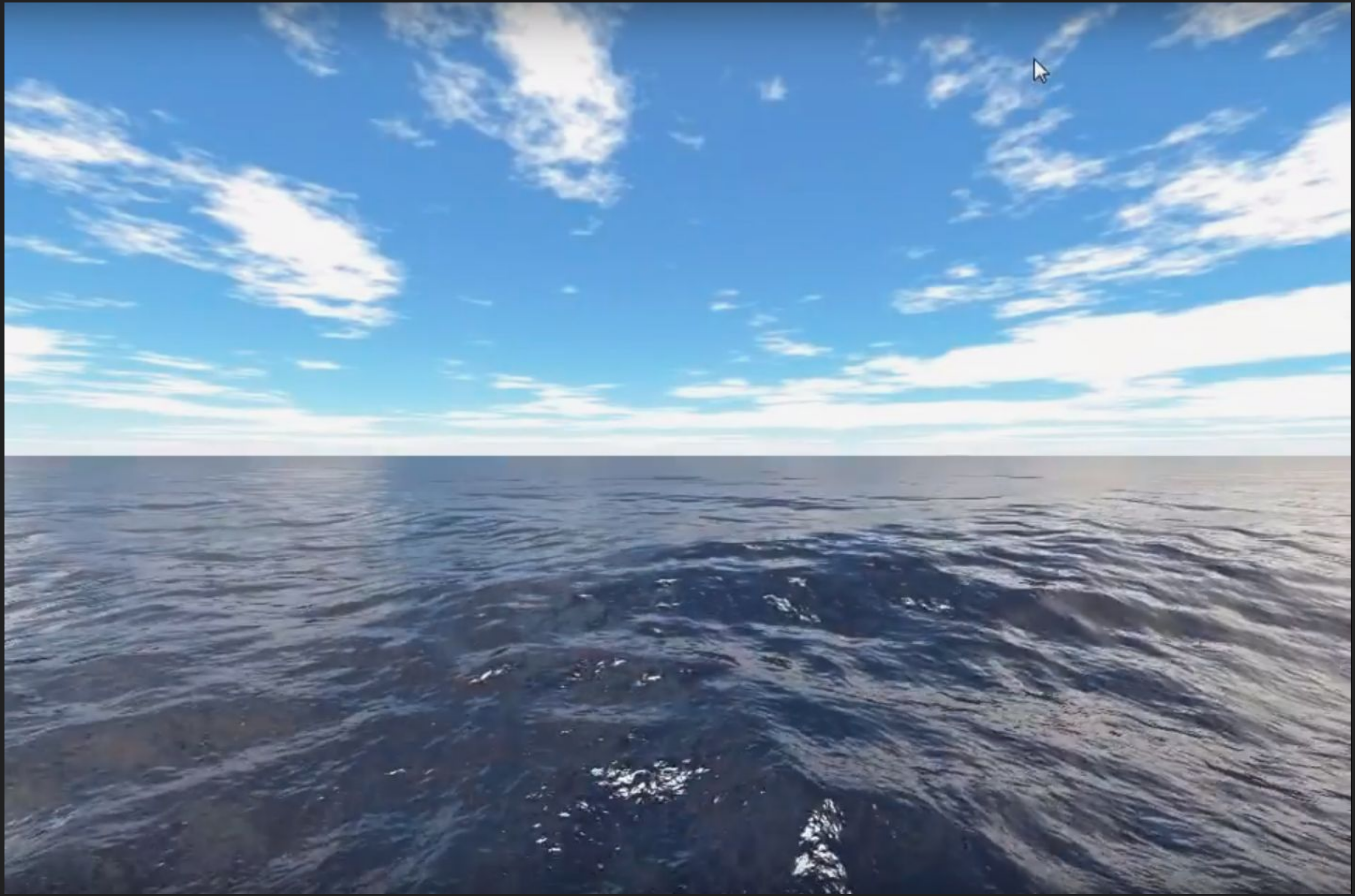
Mostly flat

Stationary waves
Small ripples

Reflected /
refracted light



Introduction - Current real-time techniques

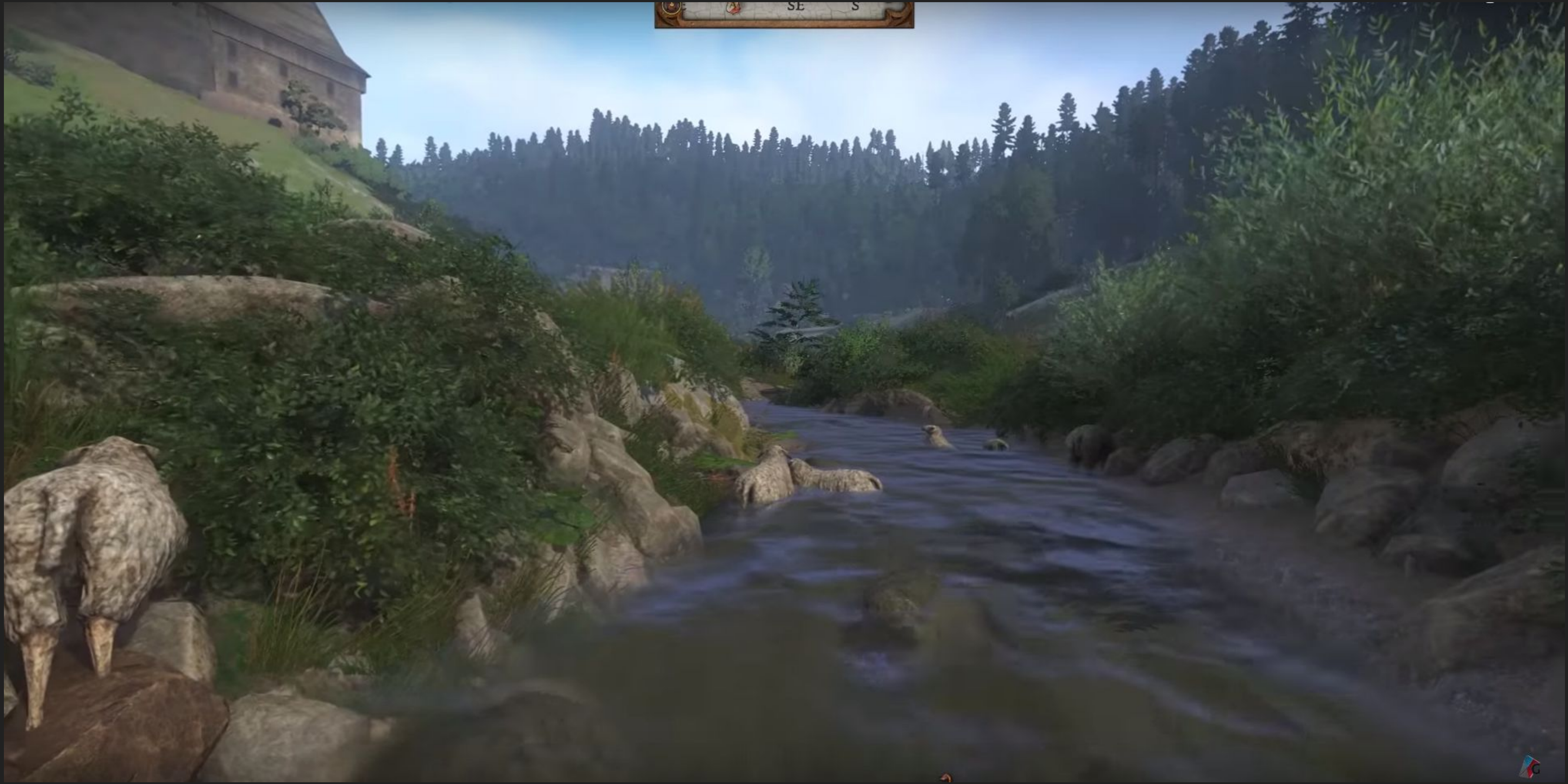


Scalable oceans in Proland

Introduction - Current real-time techniques

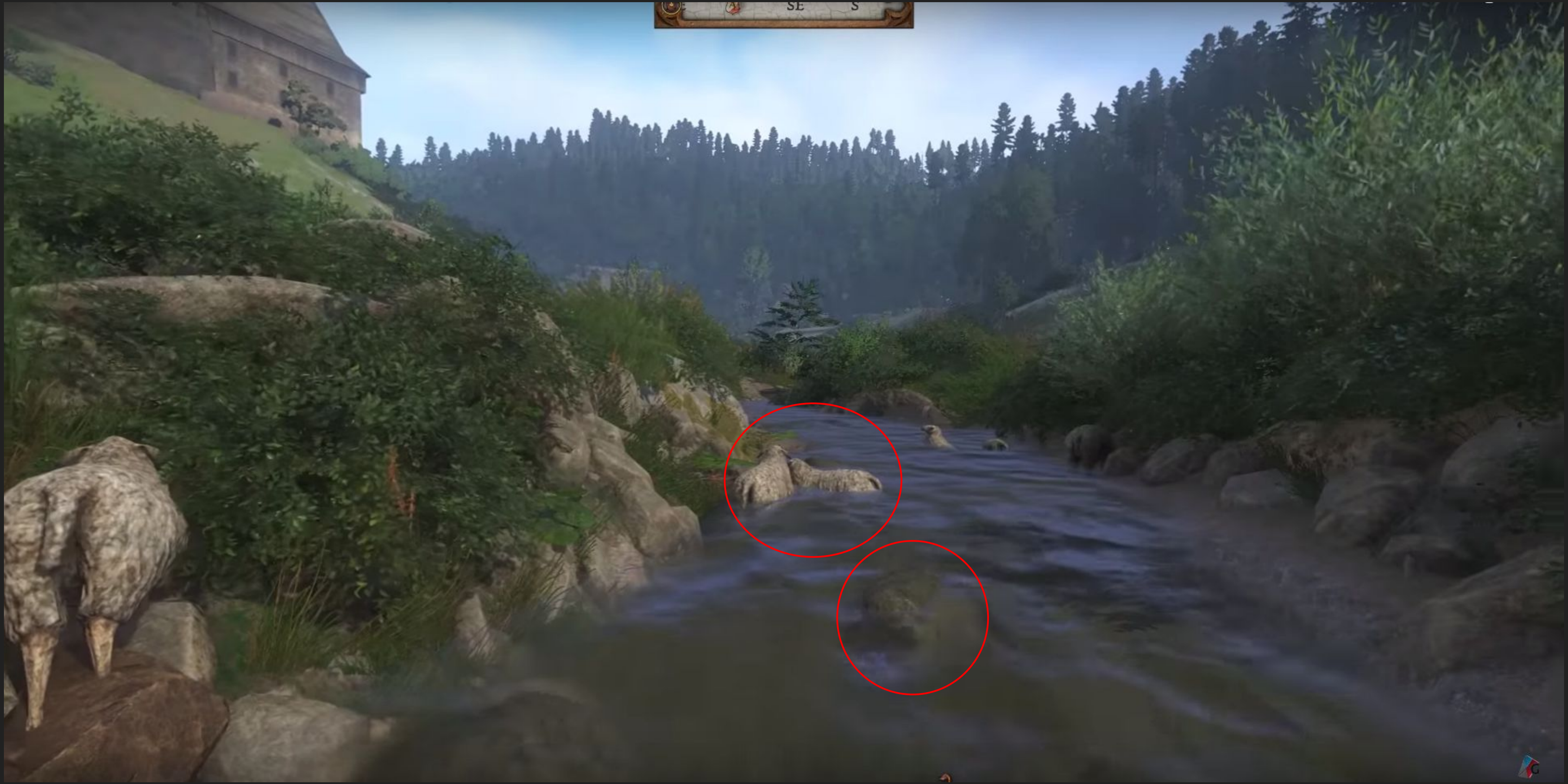
Rapids in Uncharted 4

Introduction - Current real-time techniques



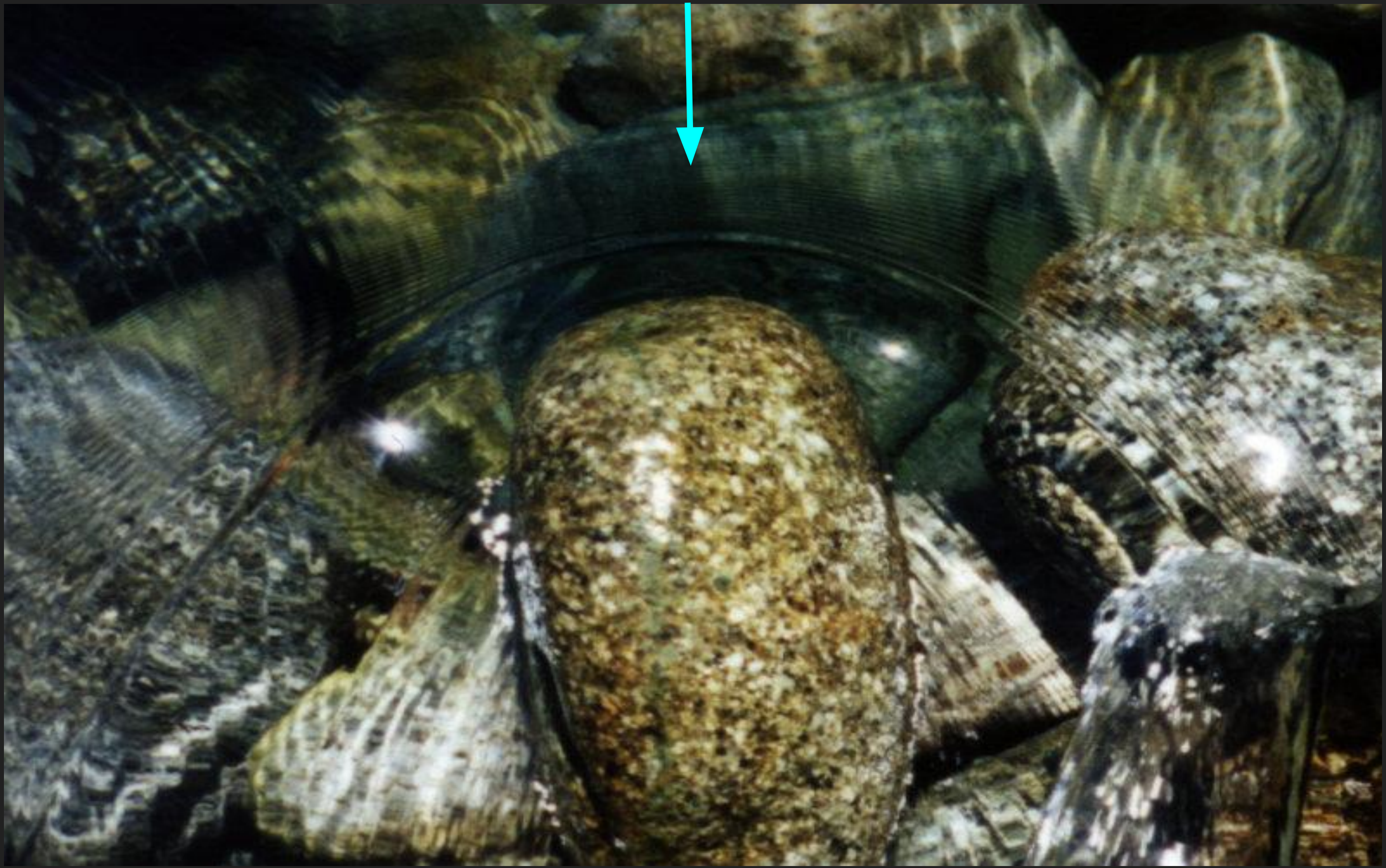
Calm river in Kingdom Come: Deliverance

Introduction - Current real-time techniques



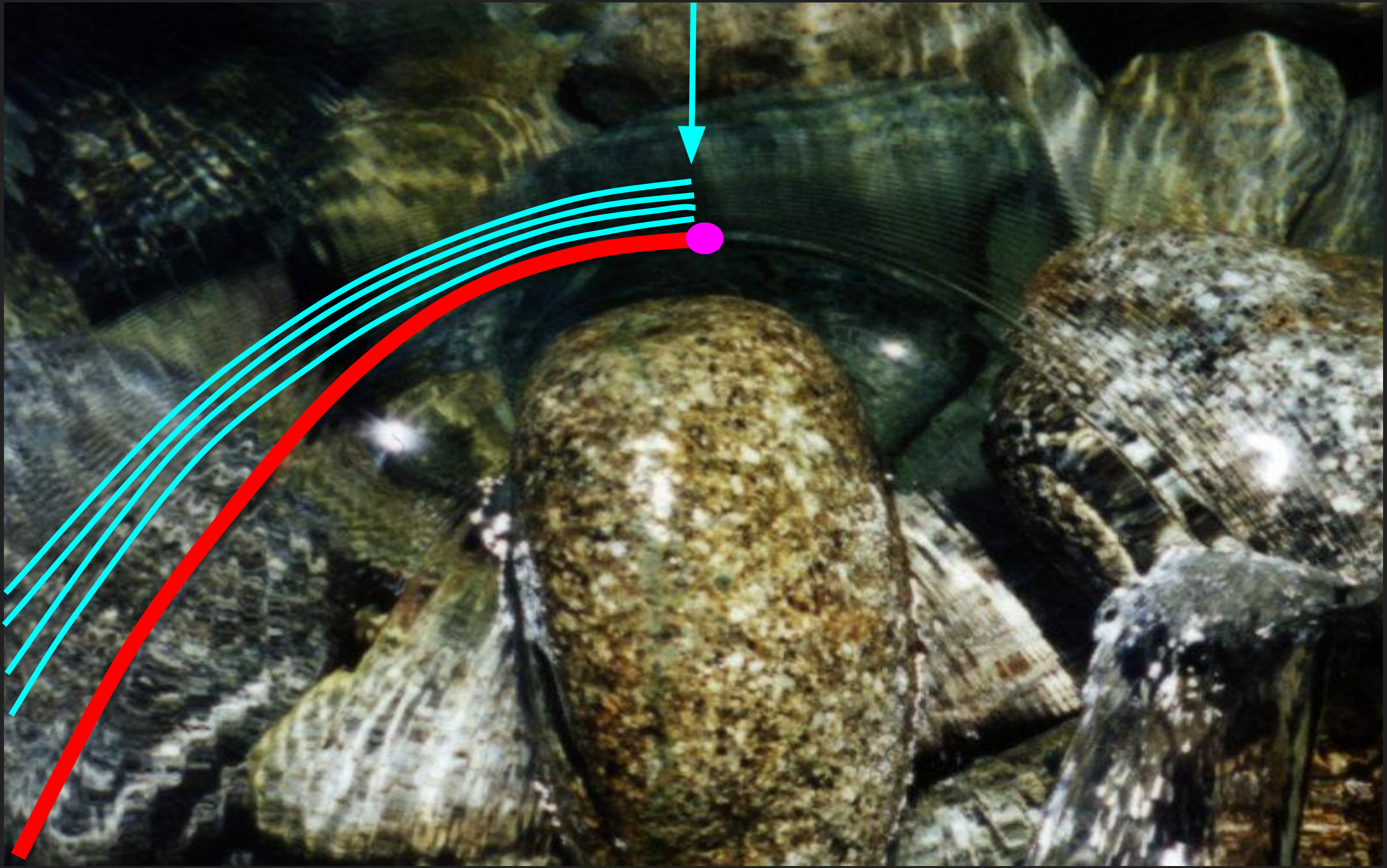
Calm river in Kingdom Come: Deliverance

Introduction - Surface details



Stationary shockwave caused by rock

Introduction - Surface details



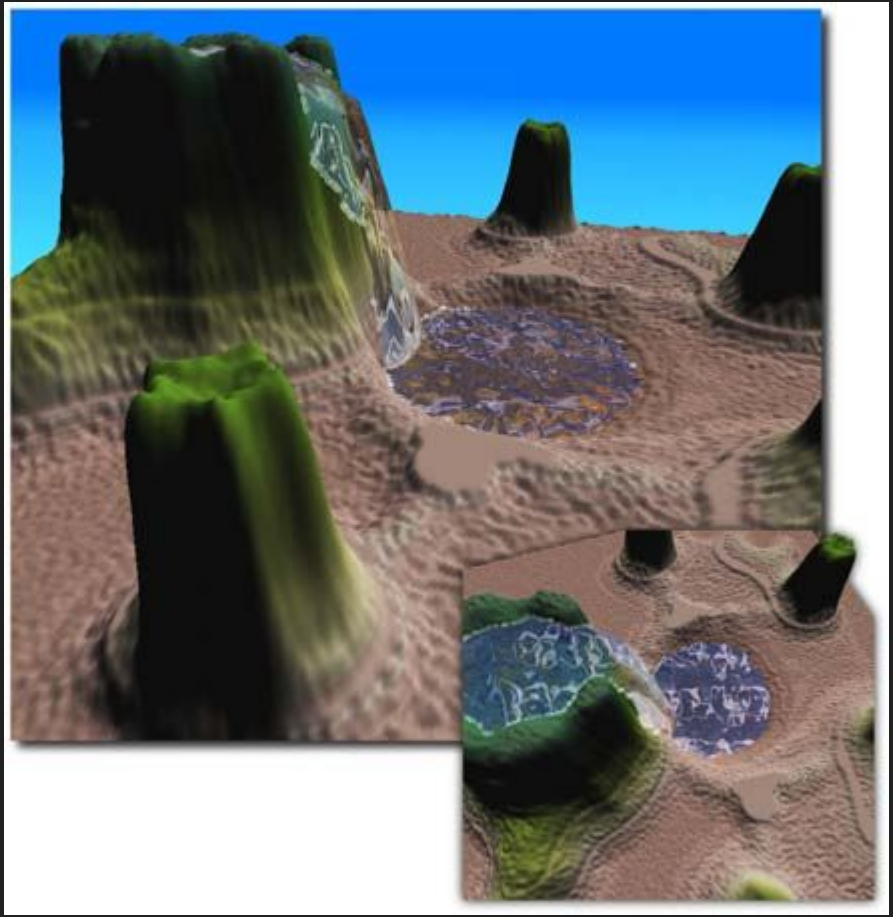
Stationary shockwave caused by rock

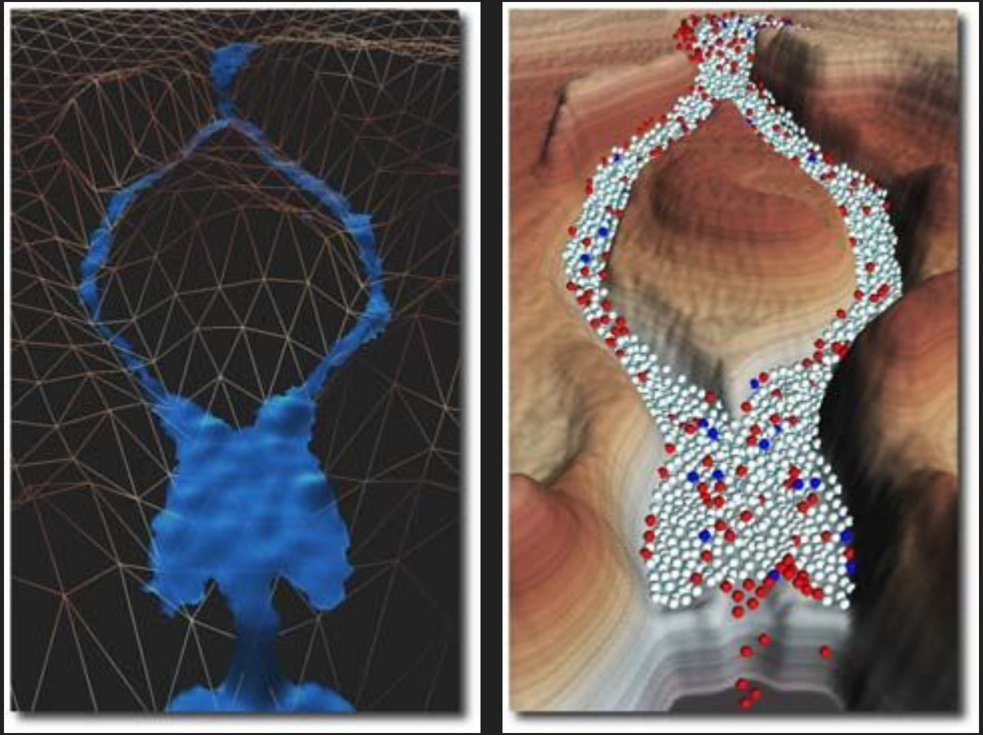
Introduction - Our Contribution

Real-time, GPU-based, meshless method

For simulating surface details such as stationary shockwaves

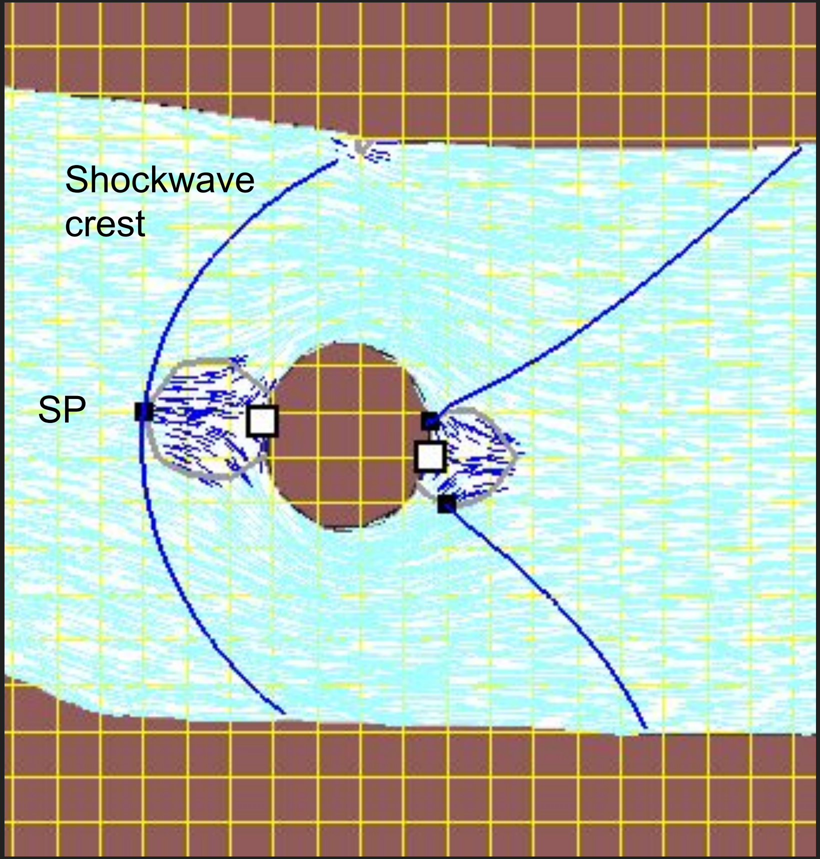
Related work - Full 3D approach *[Kipfer and Westermann, 2006]*





Surface construction from Solid Particle Hydraulics

Previous works - Original work *[Neyret and Praizelin, 2001]*



Representation of shockwave description

Previous works - Original work *[Neyret and Praizelin, 2001]*

Input:
- Terrain
- Obstacles

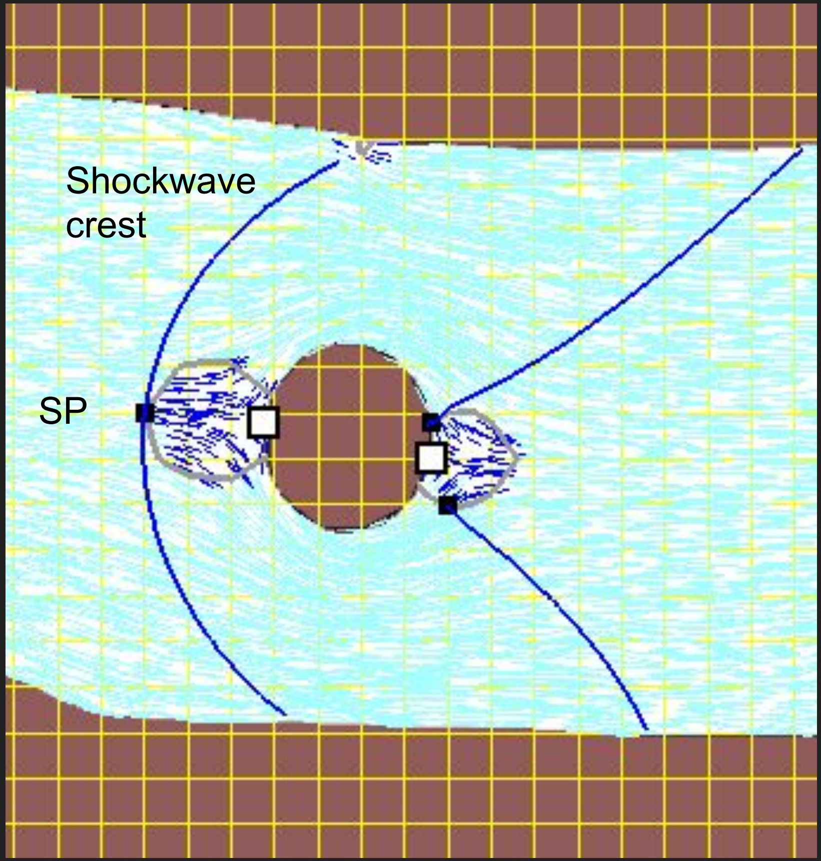
Velocity map

Analysis: wave theory, shallow water physics

Output: start points (SP) + line segments for crest

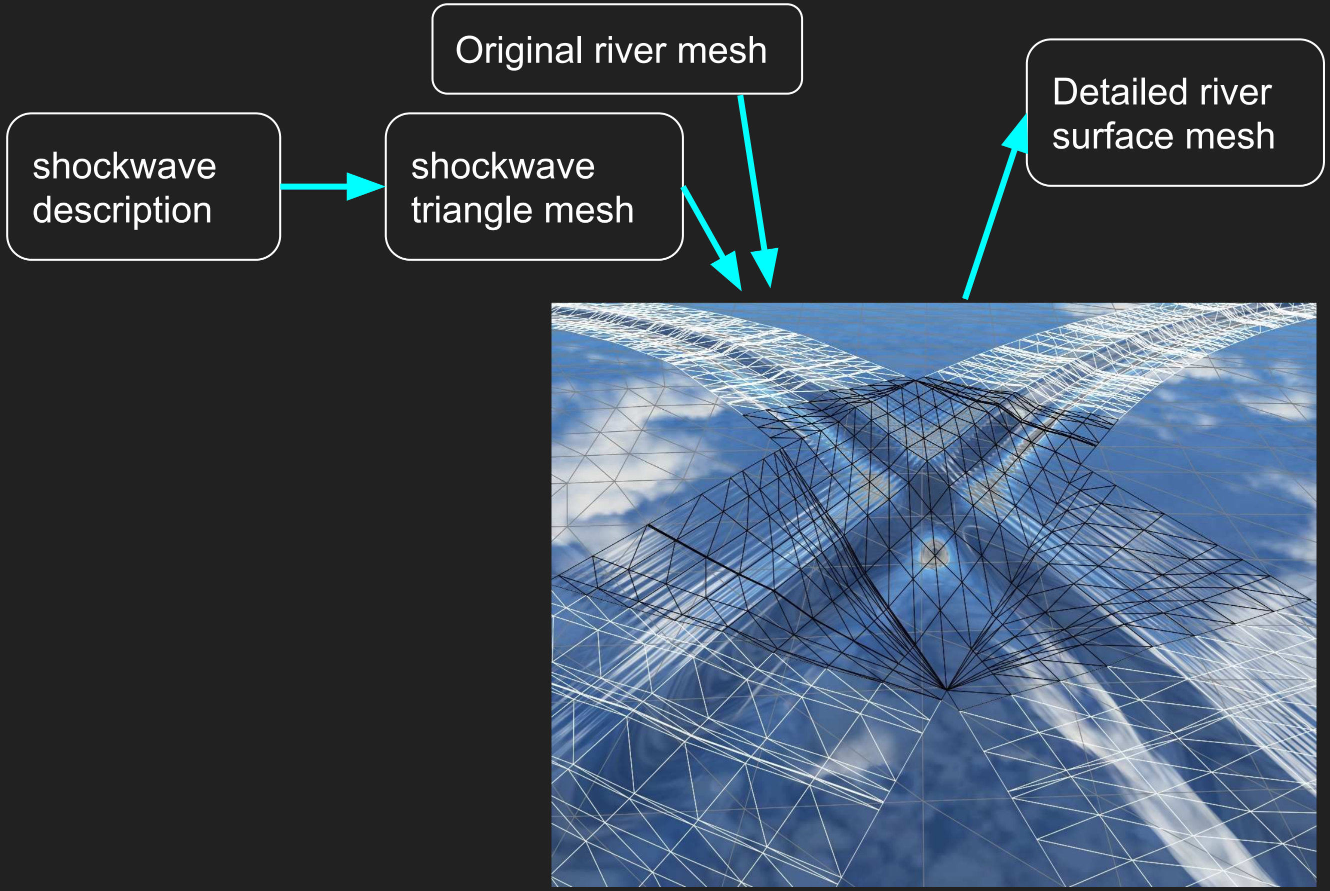
Start Point 1	...	Start Point n
Seg 1	→ ...	→ Seg m
Seg 1	→ ...	→ Seg m

Linked lists of shockwave segments



Representation of shockwave description

Previous works - Latest work *[Yu et al., 2011]*



shockwave description

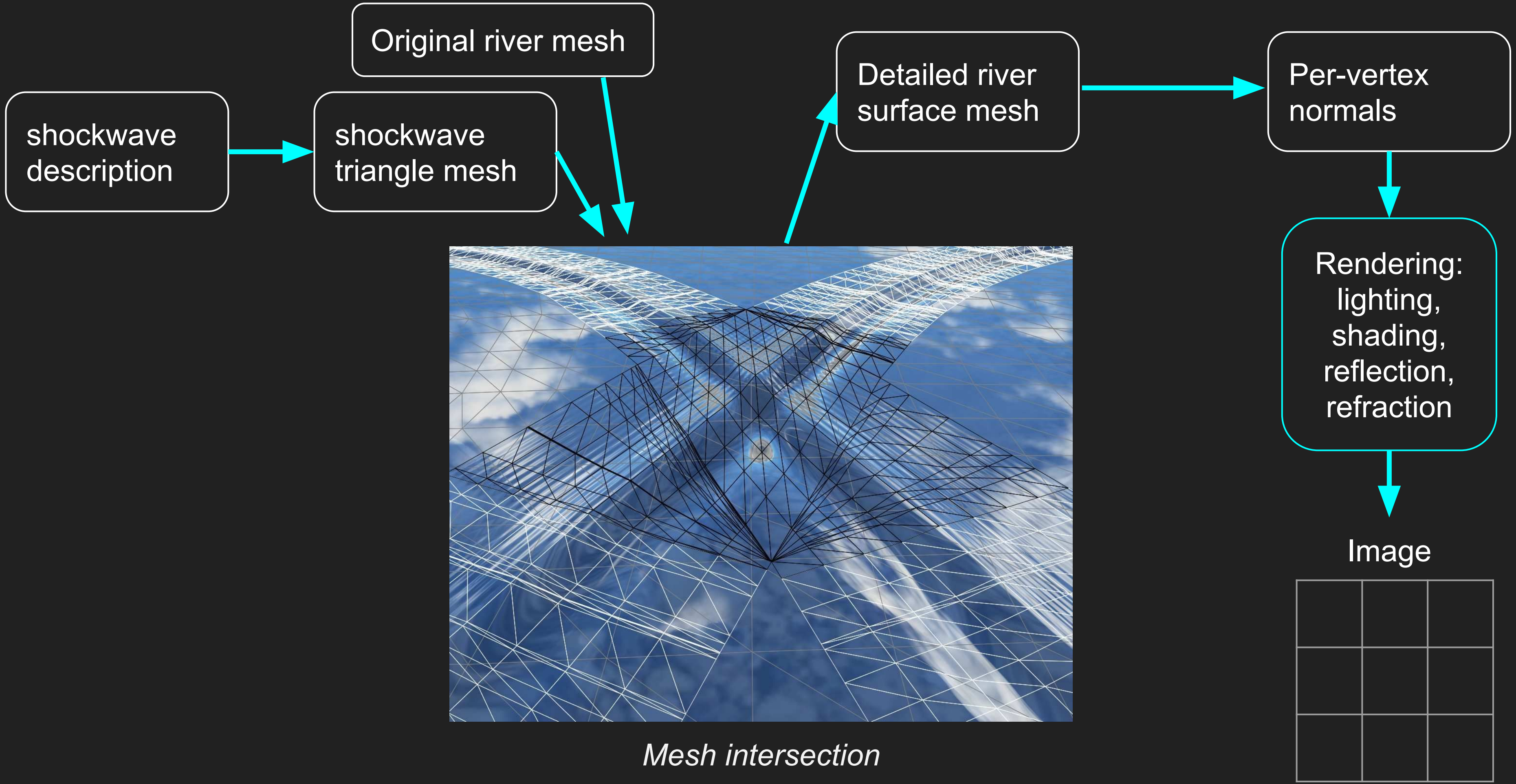
shockwave triangle mesh

Original river mesh

Detailed river surface mesh

Mesh intersection

Previous works - Latest work *[Yu et al., 2011]*



shockwave description

shockwave triangle mesh

Original river mesh

Mesh intersection

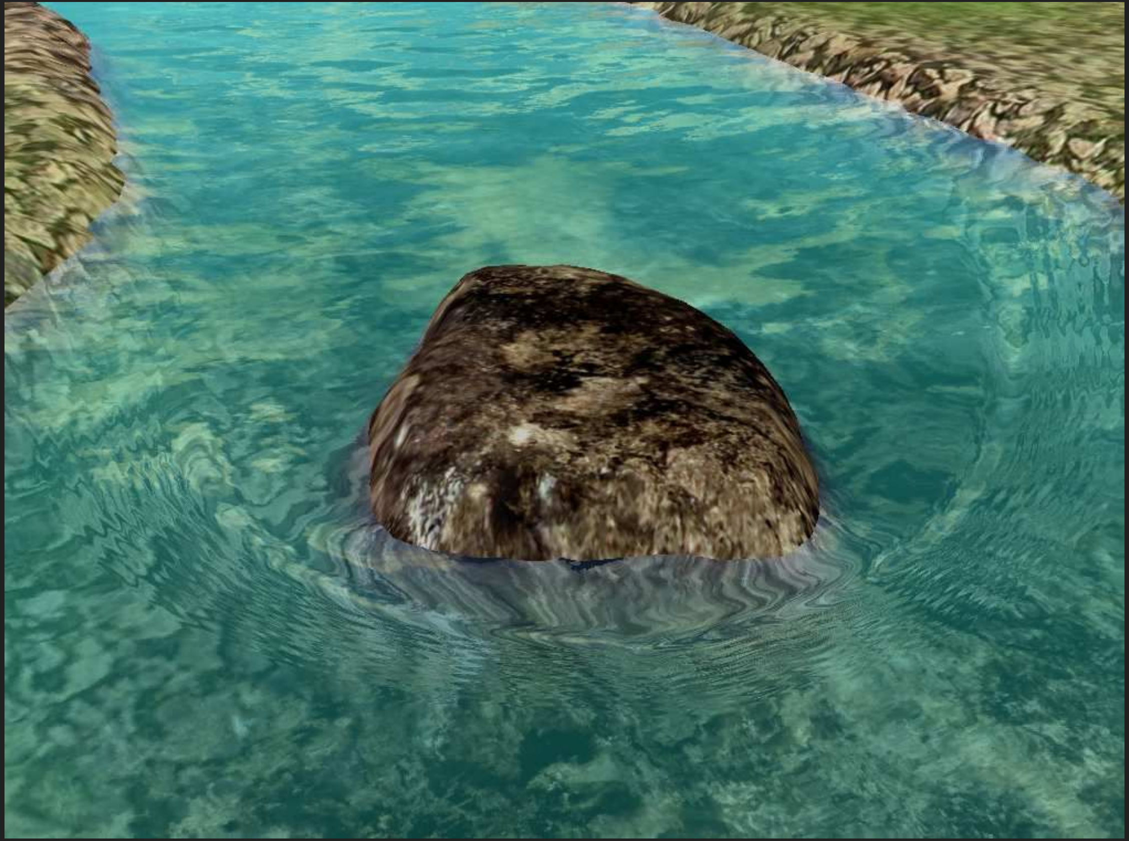
Detailed river surface mesh

Per-vertex normals

Rendering: lighting, shading, reflection, refraction

Image

Previous works - Latest work *[Yu et al., 2011]*



Realistic rendering with front shockwave

Our work - Overview

- Avoid meshes: directly compute per-pixel normals
- Add life and interaction: dynamic velocity field

GPU is a good fit!

Our work - Overview

- Avoid meshes: directly compute per-pixel normals
- Add life and interaction: dynamic velocity field

GPU is a good fit!

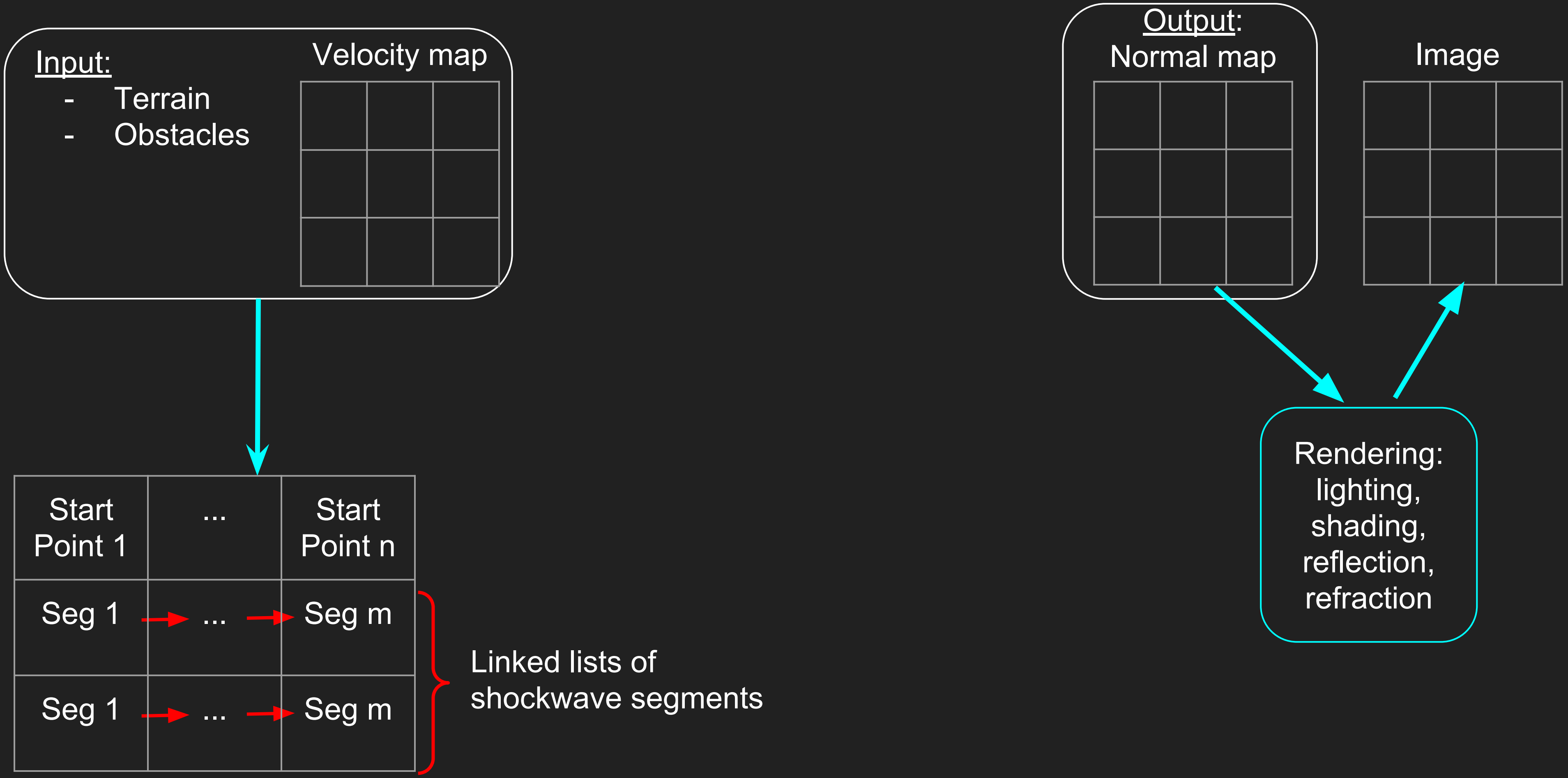
Tools:

- Data storage: Buffers
  - 4 floats per fragment
  - fragment corresponds to a pixel
  - Image buffer: R,G,B,A
- Computation: Fragment shaders
  - Runs once per fragment per frame
  - Input: fragment screen coords
  - Output to a Buffer: 4 floats

Fragment 0, 0	Fragment 1, 0	...	Fragment N-1, 0
Fragment 0, 1	...	...	Fragment N-1, 1
...	...	...	...
Fragment 0, M-1	Fragment 1, M-1	...	Fragment N-1, M-1

Buffer for an N by M screen

Our work - Method



Input:

- Terrain
- Obstacles

Velocity map

Start Point 1	...	Start Point n
Seg 1 →	... →	Seg m
Seg 1 →	... →	Seg m

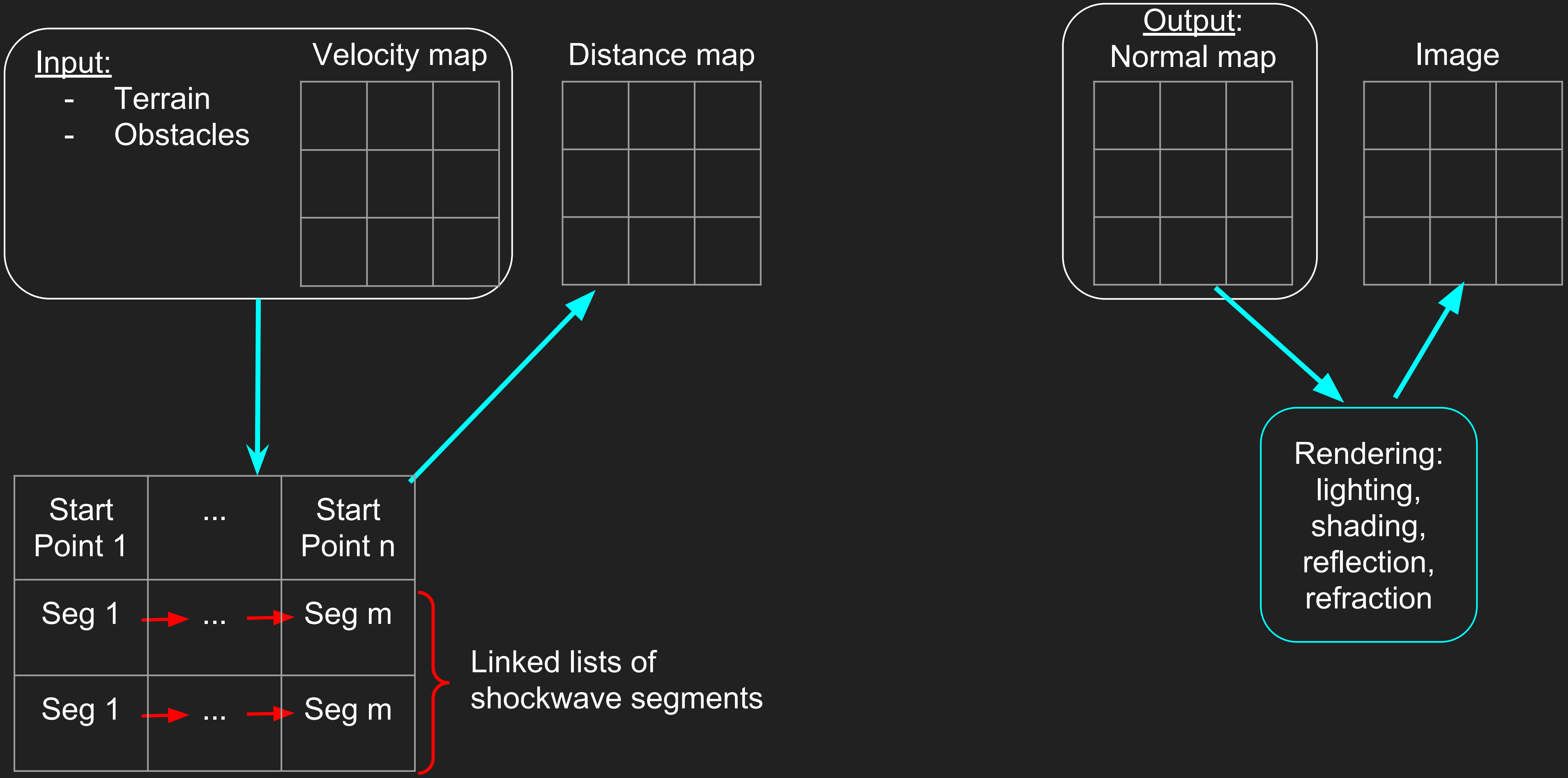
Linked lists of shockwave segments

Output:
Normal map

Image

Rendering: lighting, shading, reflection, refraction

Our work - Method



Input:

- Terrain
- Obstacles

Velocity map

Distance map

Output:
Normal map

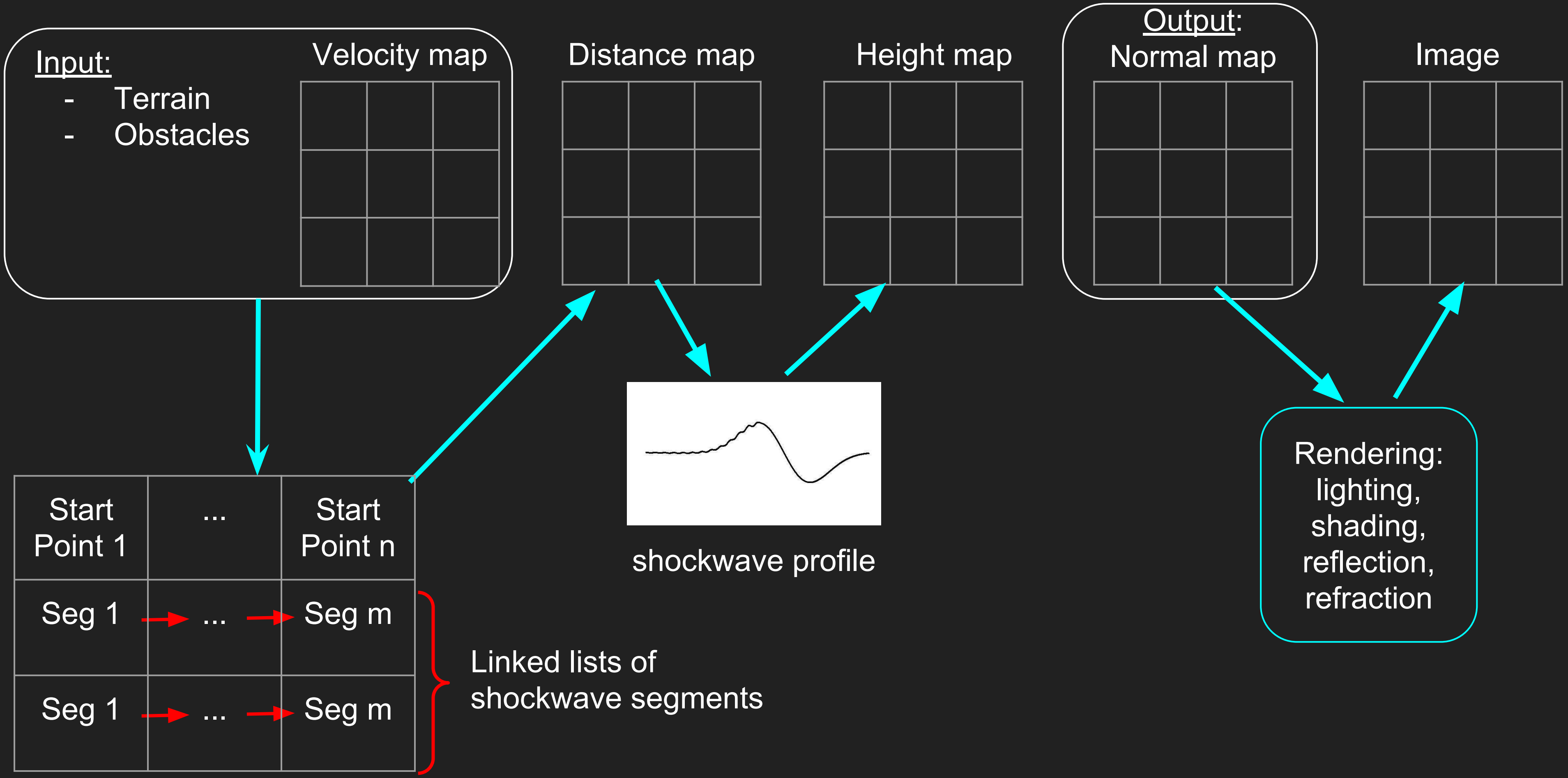
Image

Start Point 1	...	Start Point n
Seg 1 →	... →	Seg m
Seg 1 →	... →	Seg m

Linked lists of shockwave segments

Rendering:
lighting,
shading,
reflection,
refraction

Our work - Method



Input:

- Terrain
- Obstacles

Velocity map

Distance map

Height map

Output:
Normal map

Image

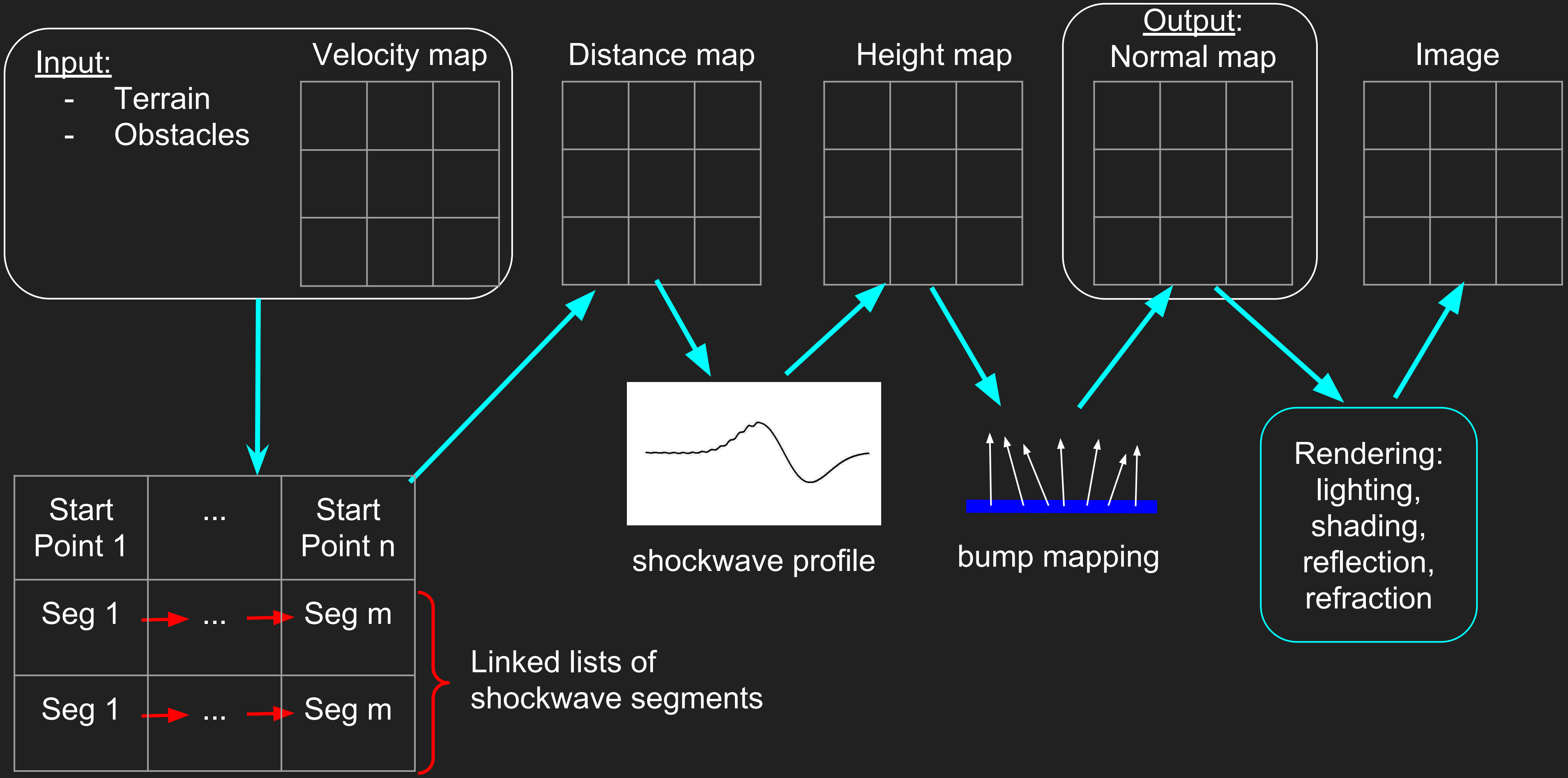
Start Point 1	...	Start Point n
Seg 1 →	... →	Seg m
Seg 1 →	... →	Seg m

Linked lists of shockwave segments

shockwave profile

Rendering: lighting, shading, reflection, refraction

Our work - Method



Input:
- Terrain
- Obstacles

Velocity map

Distance map

Height map

Output:
Normal map

Image

Start Point 1	...	Start Point n
Seg 1 →	... →	Seg m
Seg 1 →	... →	Seg m

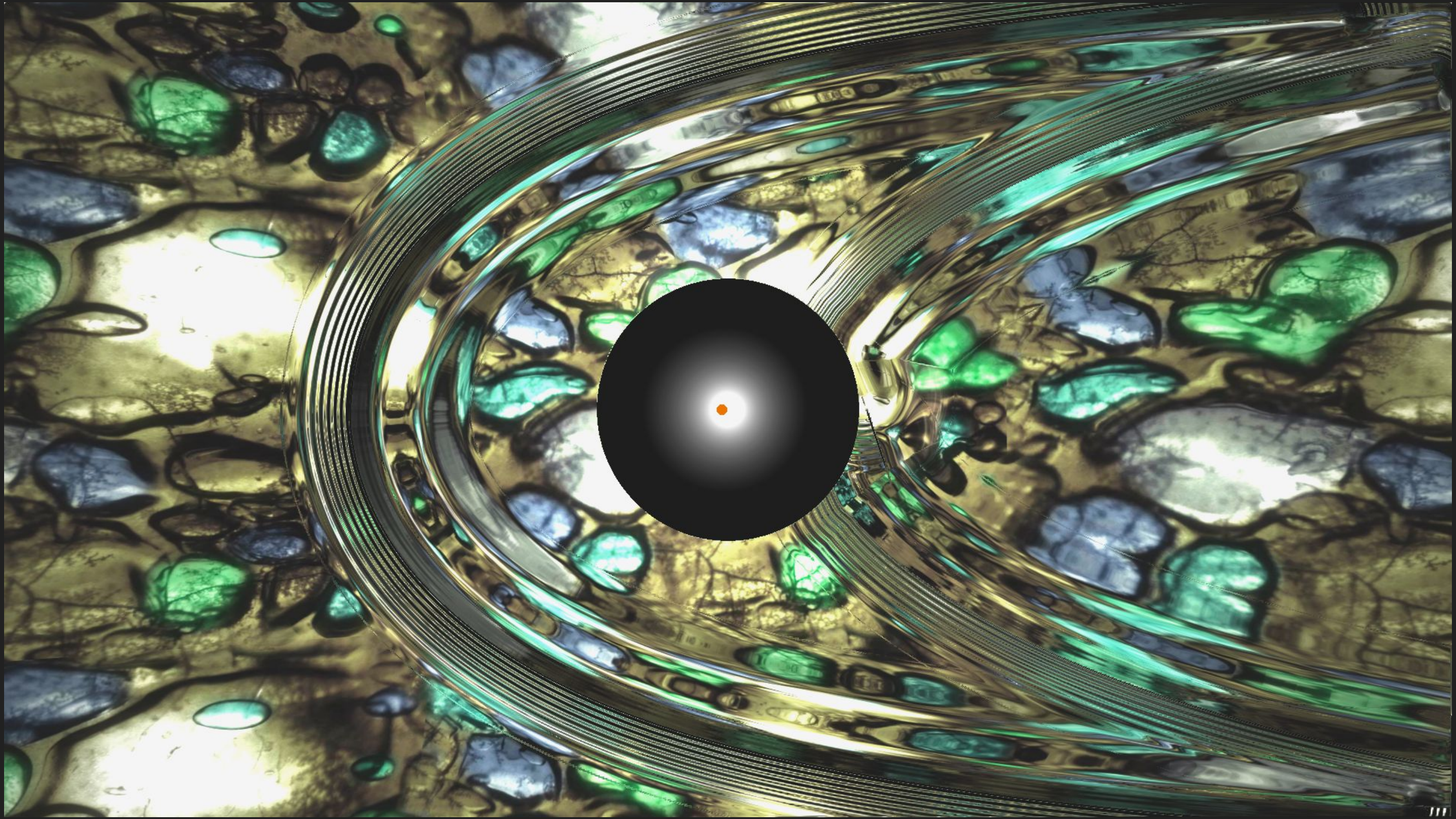
Linked lists of shockwave segments

shockwave profile

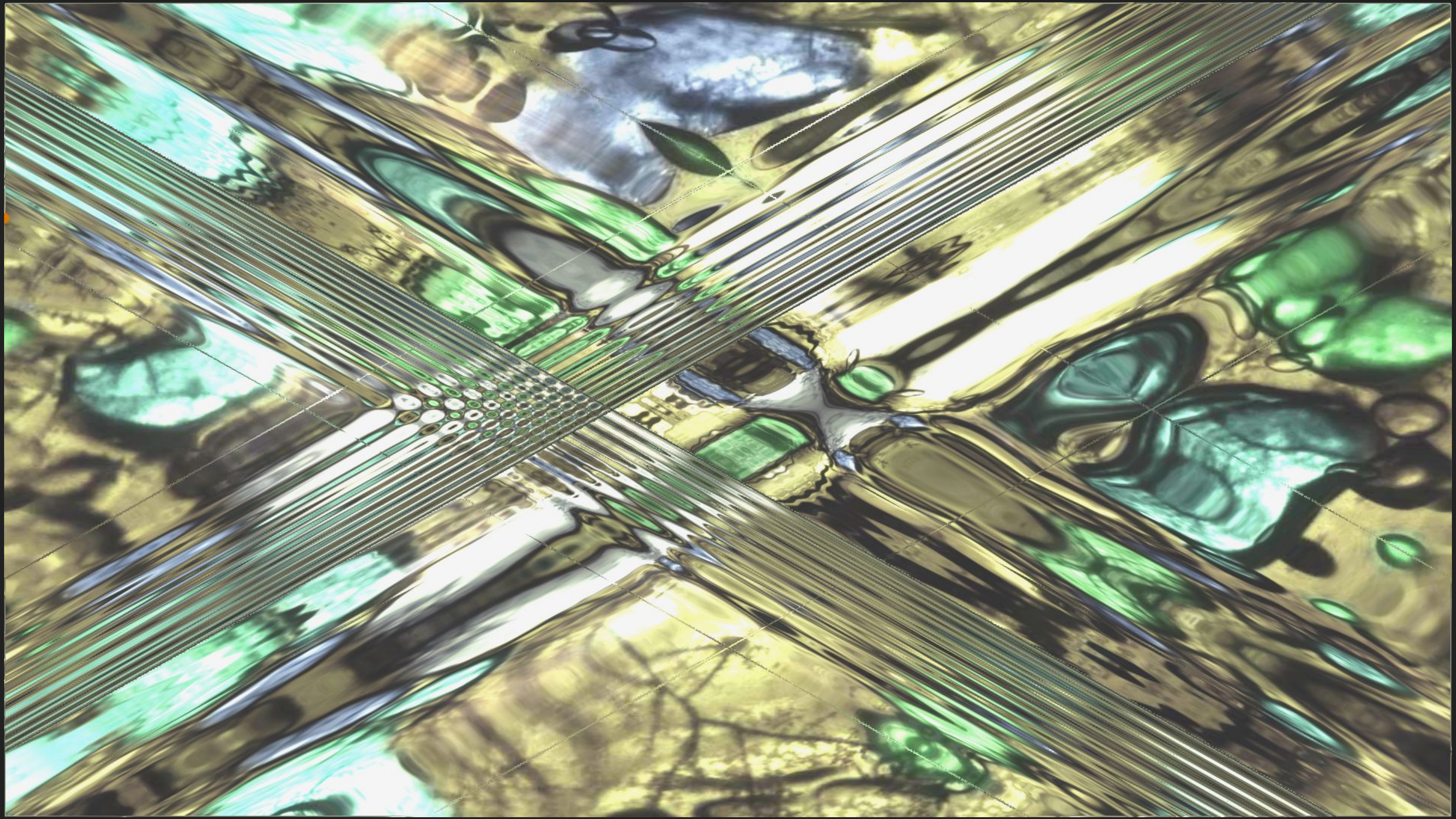
bump mapping

Rendering: lighting, shading, reflection, refraction

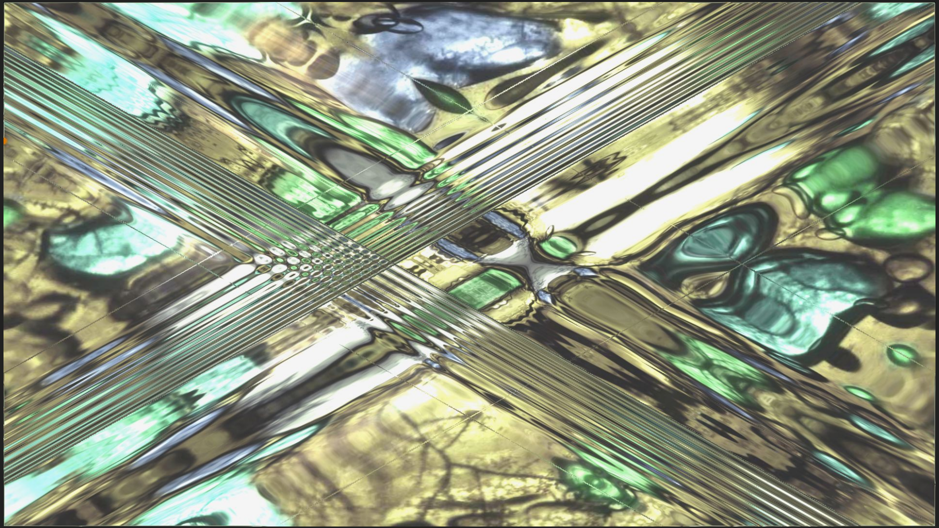
Results - Single rock with three shockwaves

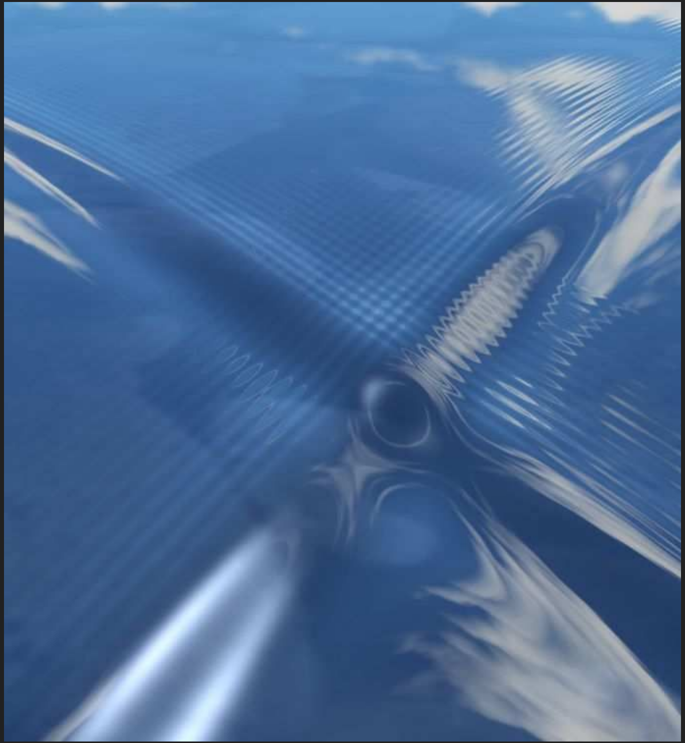


Results - Intersecting shockwaves



Results





Shockwave intersection [Yu et al., 2011]

- Decent results with basic implementation
  - Comparable shockwave shape
  - Rear shockwaves accounted for
  - Real-time achievable

Conclusion

- Our GPU adaptation seems promising
    - More options for velocity field
    - Avoids costly mesh constructions
    - Heightmap approach eases intersections
    - Takes advantage of parallelism

- Ongoing project...

References

[Neyret and Praizelin, 2001]
Fabrice Neyret and Nathalie Praizelin. Phenomenological Simulation of Brooks. In Eurographics Workshop on Computer Animation and Simulation (EGCAS), pages 53–64, Manchester, United Kingdom, September 2001. Eurographics, Springer.

[Yu et al., 2011]
Qizhi Yu, Fabrice Neyret, and Anthony Steed. Feature-based vector simulation of water waves. 22:91–98, 04 2011.

[Kipfer and Westermann, 2006]
Kipfer, Peter and Westermann, Rüdiger. Realistic and Interactive Simulation of Rivers. Proceedings of Graphics Interface 2006.

Video / Image sources

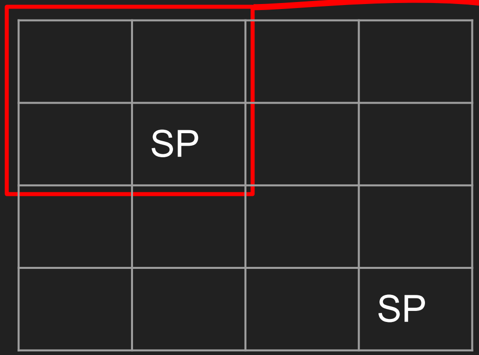
Kingdom come deliverance - calm river, flow through rocks, no interaction

Uncharted 4 - rapids, not what we want, but has local effects + waves,

Proland - ocean waves, with small details too

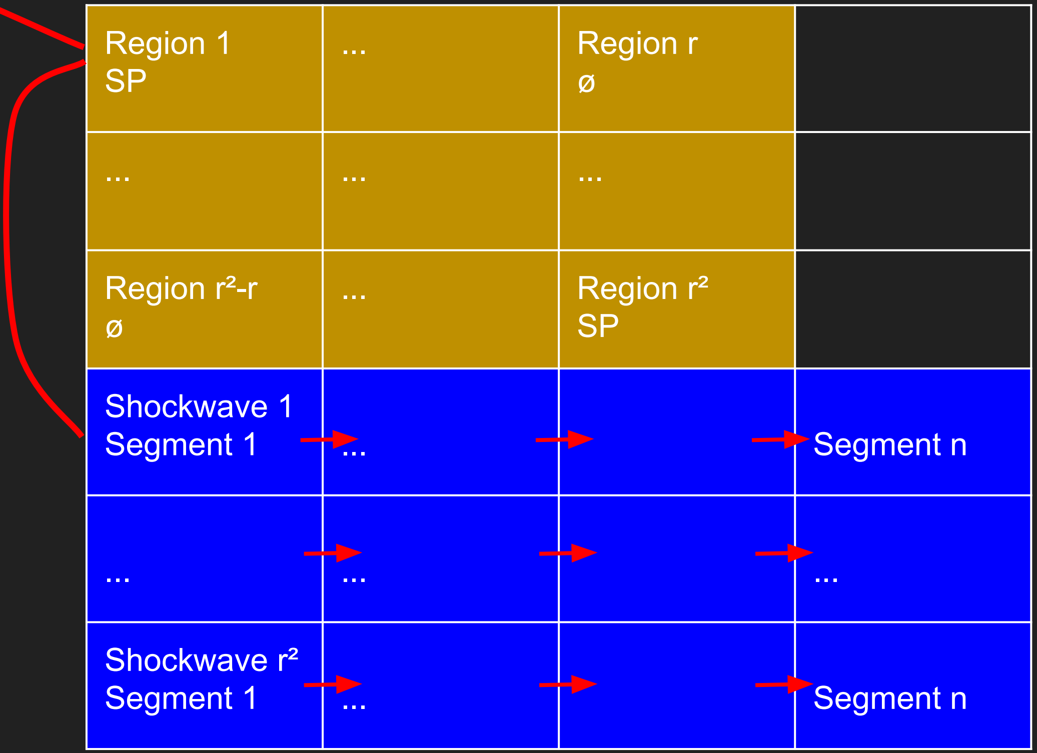
Bump map diagram

Our work - Details - Start Points and Shockwaves



	SP		
			SP

Start point marker buffer



Region 1 SP	...	Region r ø	
...	...	...	
Region r²-r ø	...	Region r² SP	
Shockwave 1 Segment 1	...		Segment n
...	...		...
Shockwave r² Segment 1	...		Segment n

Shockwave buffer for r² regions and shockwave length n

- Finding start points
  - Each fragment checks criteria
  - Marks itself in marker buffer
- Tracking start points
  - Subdivide screen into regions
  - Parallel search
- Constructing shockwaves
  - Read SP (if any) from its region
  - Parallel segment update along list

Results

Not quite real-time for high res / complex scenes

- Major bottleneck: computing pixel-shockwave distance
  - Naïve point-segment distance algorithm
  - Solve with acceleration structure OR different shockwave representation

Conclusion

- Future works:
  - Shockwave dissipation
  - Represent more feature types

Shockwaves caused by submerged obstacle

Hydraulic jump