Optimized Rendering Techniques for Mobile VR

ARM

Roberto Lopez Mendez

Senior Software Engineer, ARM

DevGAMM 2016 – Moscow

13/05/2016

Agenda

- Ice Cave Demo

- Improving VR quality & performance
  - The concept of local cubemap
  - Dynamic soft shadows and reflections based on local cubemaps
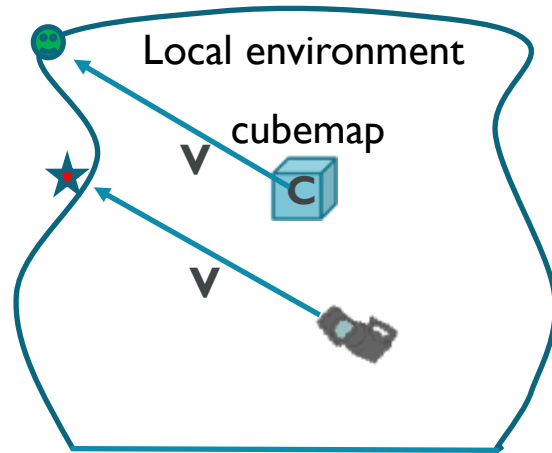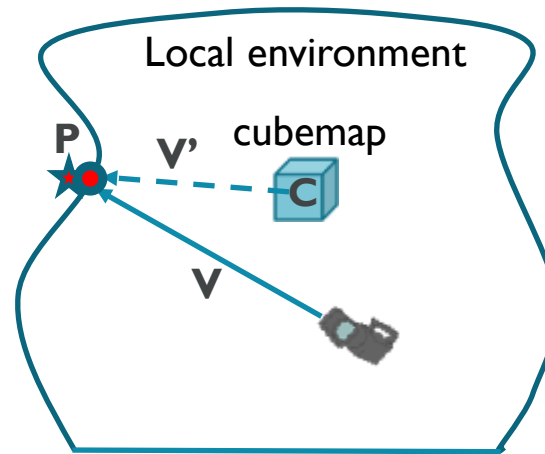 - Stereo reflections in VR

ARM

Demo Ice Cave

ARM

Quality and Performance for VR

# Optimized Rendering Techniques Based On Local Cubemaps

ARM

The Concept of Local Cubemaps

If we use the view vector **V** defined in WCS to fetch the texel from the cubemap we will get the smiley face instead of the star.

We need to use a new vector **V'** = **CP** to fetch the correct texel . We need to find the intersection point P of the view vector **V** with the boundaries of the local environment.
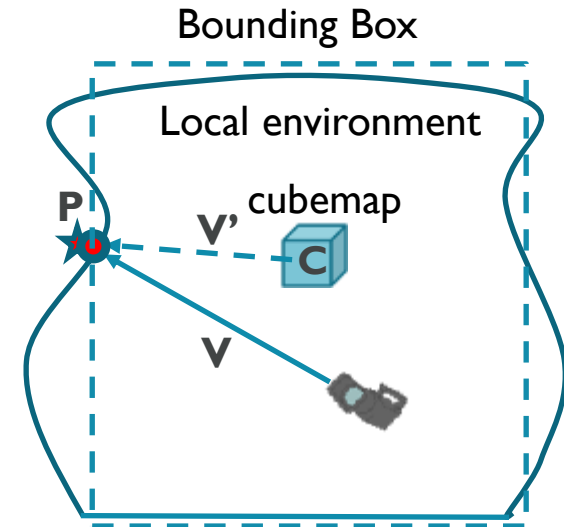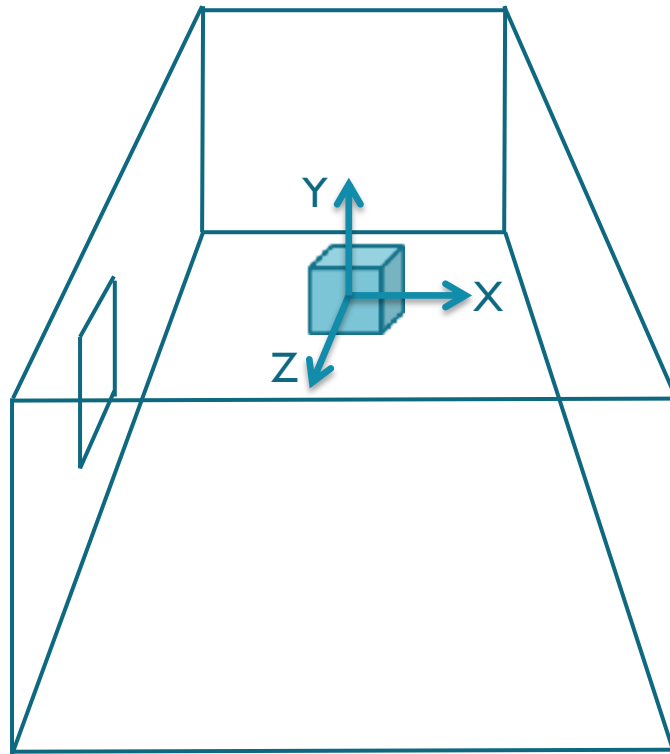
We introduce a proxy geometry to simplify the problem of finding the intersection point **P**. The simplest proxy geometry is the bounding box.

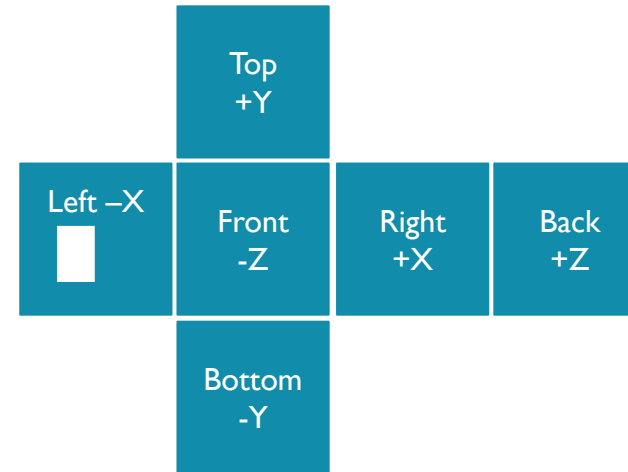**Local Cubemap = Cubemap + Cubemap Position + Scene Bounding Box + Local Correction**

ARM

Dynamic Soft Shadows Based on Local Cubemaps

ARM

Dynamic Soft Shadows Based on Local Cubemaps

Generation stage

Render the transparency of the scene in the alpha channel

	Top +Y		
Left –X	Front -Z	Right +X	Back +Z
	Bottom -Y		

Camera background alpha colour = 0

Opaque geometry is rendered with alpha = 1

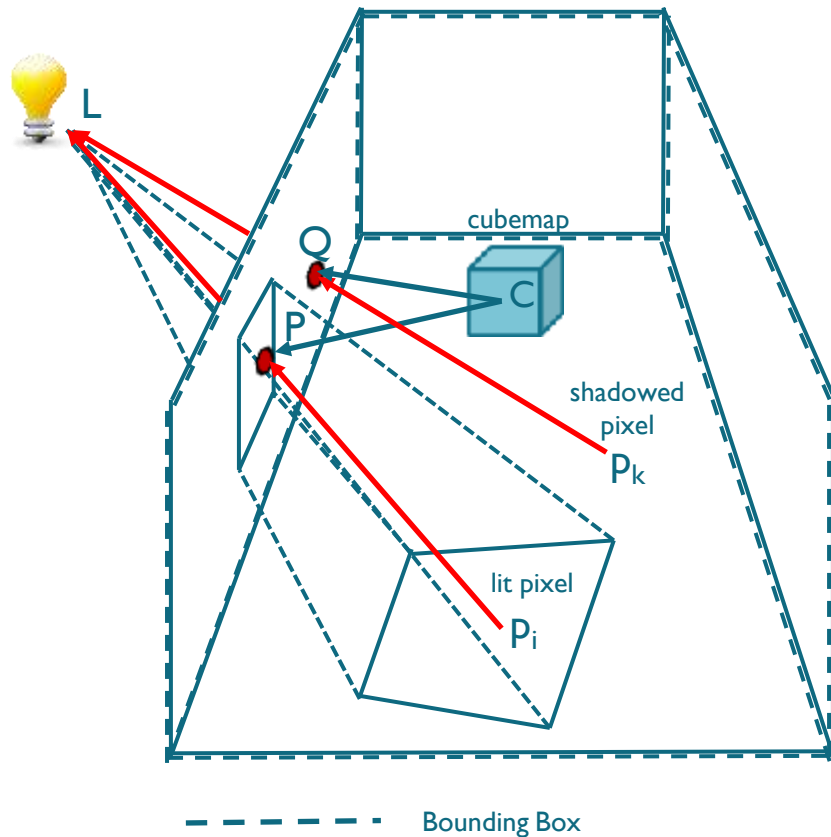Semi-transparent geometry is rendered with alpha different from 1

Fully transparent geometry is rendered with alpha 0

We have a map of the zones where light rays can potentially come from and reach the geometry.

No light information is processed at this stage.

ARM

Dynamic Soft Shadows Based on Local Cubemaps

Runtime stage

- Create a vertex to light source L vector in the vertex shader.

- Pass this vector to the fragment shader to obtain the vector from the pixel to the light position p_iL.

- Find the intersection of the vector p_iL with the bounding box.

- Build the vector CP from the cubemap position C to the intersection point P.

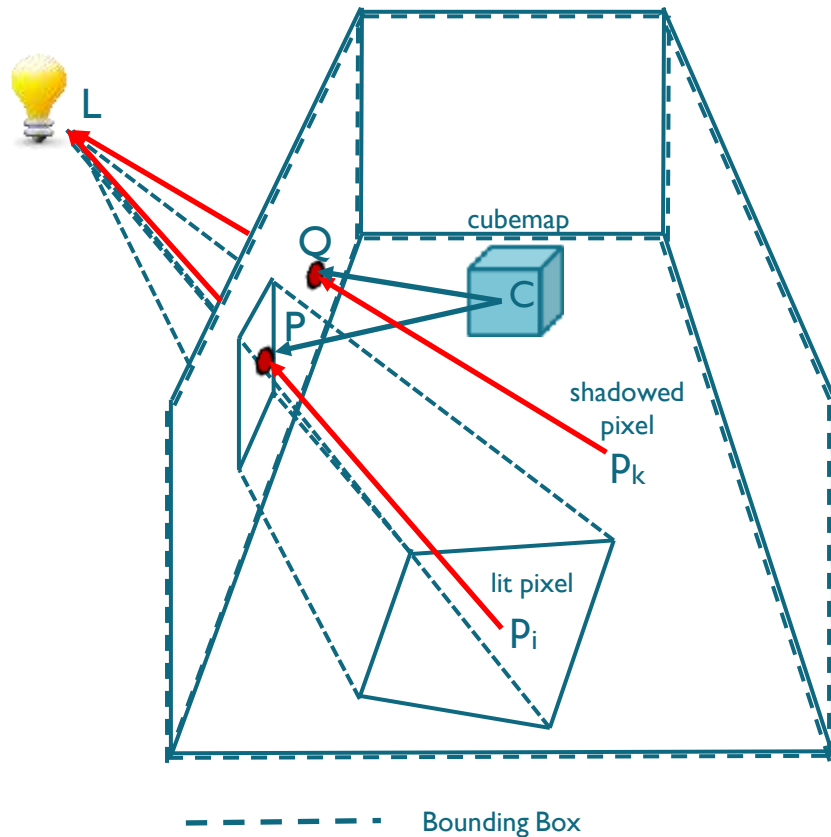- Use the new vector CP to fetch the texture from the cubemap.

  float texShadow = texCUBE(_CubeShadows, CP).a;

> ⓘ Source code in the ARM Guide for Unity Developers at MaliDeveloper.arm.com

ARM

Dynamic Soft Shadows Based on Local Cubemaps

Why soft?
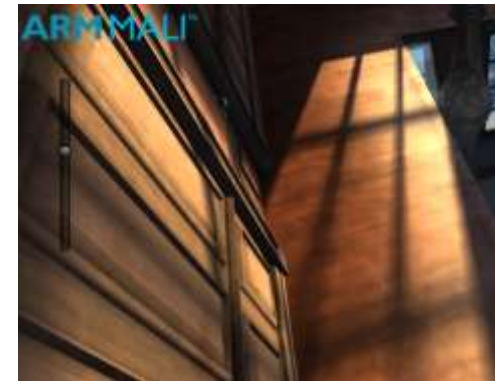
float texShadow = texCUBE(_CubeShadows, CP).a;

float4 newVec = float4`(CP, factor * length(p_iP))

float texShadow = texCUBElod(_CubeShadows, newVec).a;

The further from the object the softer the shadows.

cubemap

C

Q

P

shadowed
pixel

P_k

lit pixel

P_i

L

Bounding Box

ARM

Dynamic Soft Shadows Based on Local Cubemaps

ARM

Handling Shadows from Different Types of Geometries

High Quality Shadows from Static Environment
Use Local Cubemap technique

Shadows from Dynamic Objects
Use Shadow Mapping technique

All Shadows
Combine both types of shadows

ARM
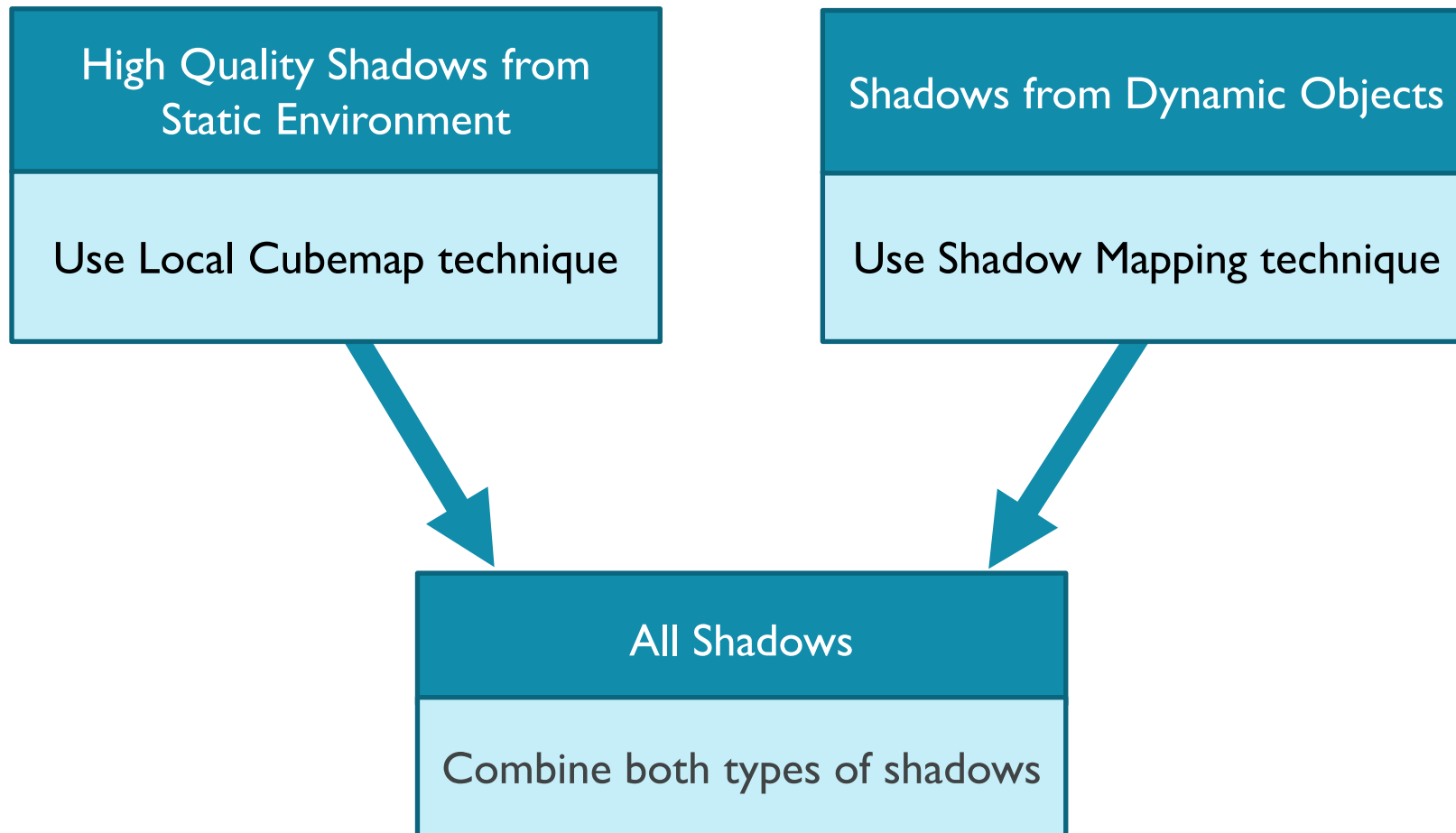
Combined Shadows in Ice Cave VR

ARMMALI™

Shadows from static Geometry using local cubemps

Shadows from dynamic Geometry using shadow mapping

ARM

Reflections Based on Local Cubemaps

ARM

Local Correction to Reflection Vector

Reflective surface

- - - - - Bounding Box
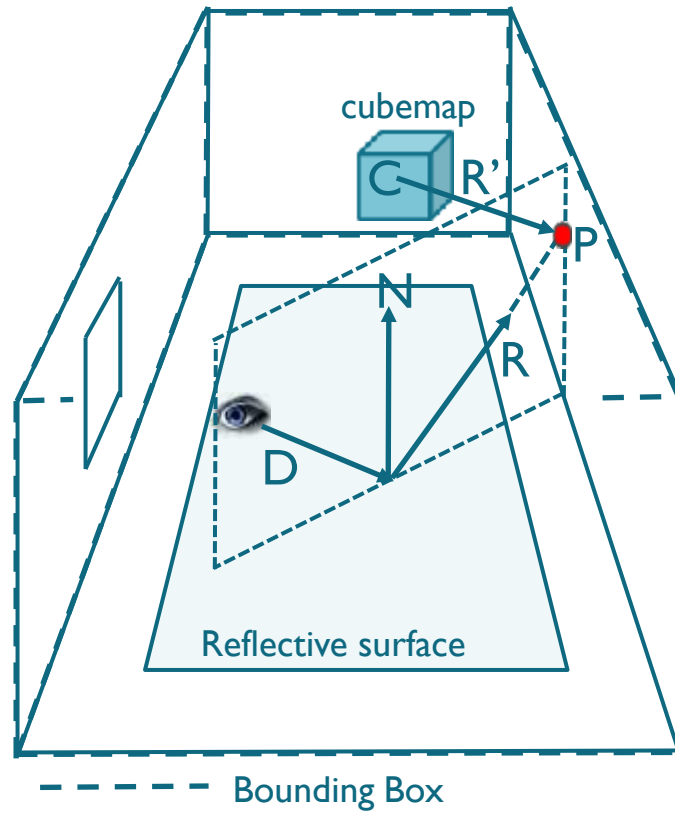
float3 R = reflect(D, N);

float4 col = texCUBE(Cubemap, R);
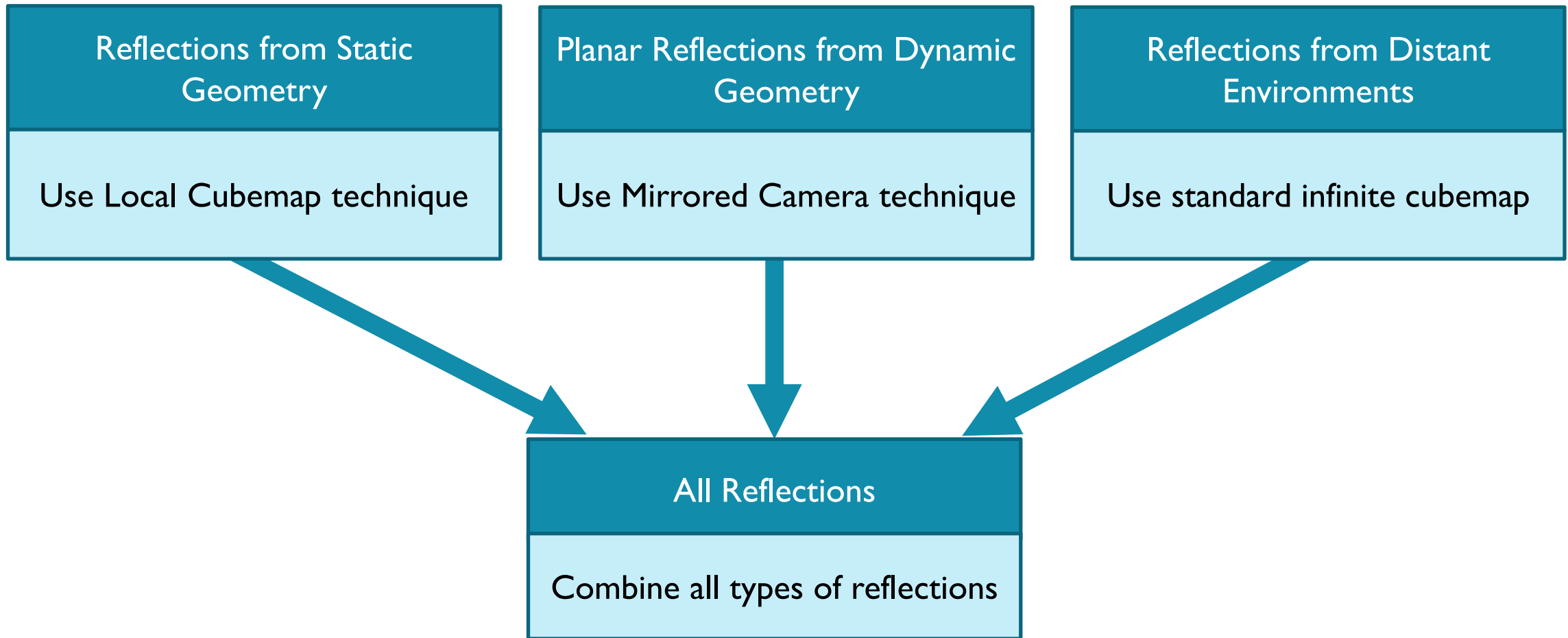
Find intersection point P
Find vector R' = CP
float4 col = texCUBE(Cubemap, R');

Source code in the ARM Guide for Unity Developers at MaliDeveloper.arm.com

ARM

Combined Reflections in Ice Cave VR

Reflections from Static Geometry	Planar Reflections from Dynamic Geometry	Reflections from Distant Environments
Use Local Cubemap technique	Use Mirrored Camera technique	Use standard infinite cubemap

All Reflections

Combine all types of reflections

ARM

Combined Reflections in Ice Cave VR

Reflections from static geometry using local cubemap

Reflections from the sky using infinite cubemap
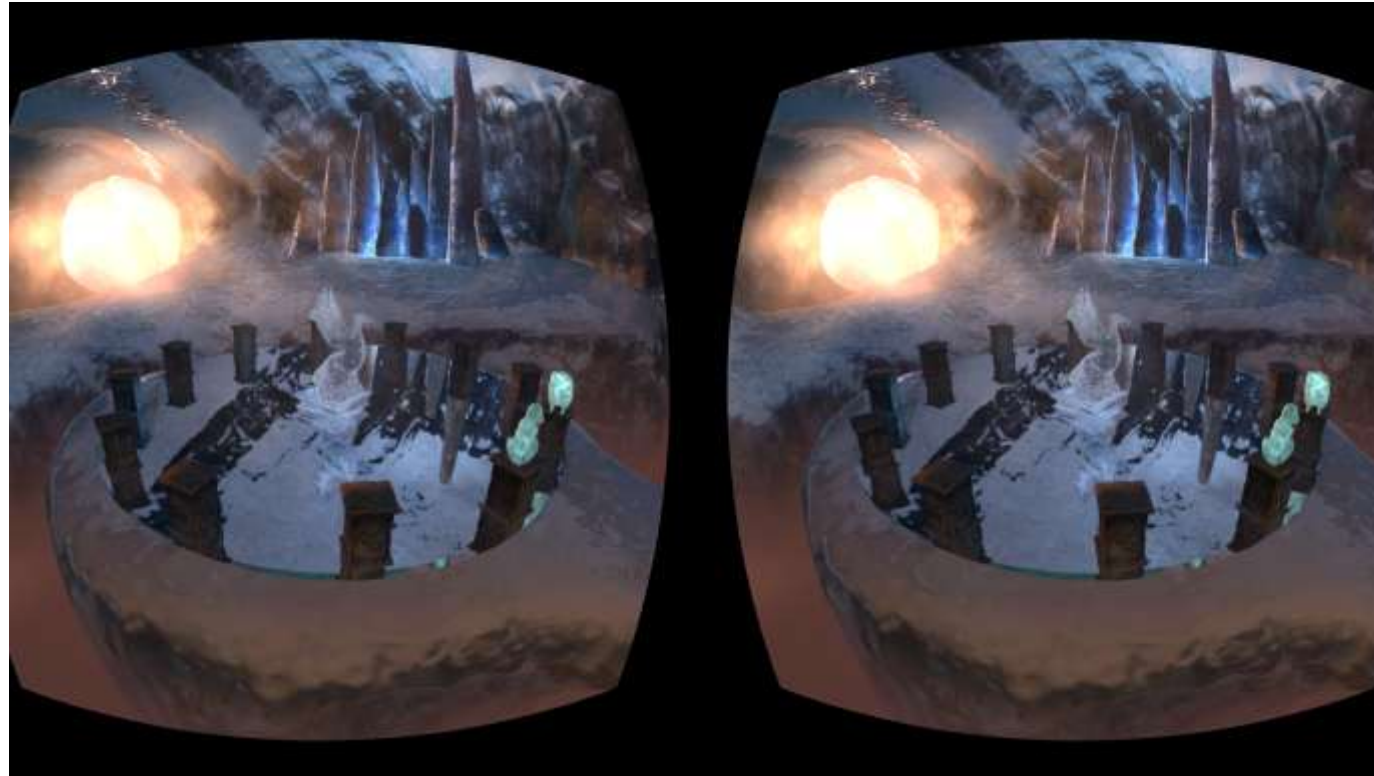
Reflections from dynamic geometry using a mirrored camera

ARM

Stereo Reflections in Unity

ARM

Why Stereo Reflections?

- Using the same texture for right/left eyes reflections in VR looks plain and breaks the sensation of full immersion.

ARM

Implementing Stereo Reflections in Unity

- Add two new cameras targeting left/right eye respectively and disable them as you will render them manually.

- Create a target texture the cameras will render to

- Attach the script to each camera, the SetUpCamera function places the camera in the right position for rendering reflections for each eye.

```
void OnPreRender()
{
    SetUpCamera ();
    // Invert winding
    GL.invertCulling = true;
}
void OnPostRender()
{
    // Restore winding
    GL.invertCulling = false;
}
```

ARM

# Implementing Stereo Reflections in Unity

Attach the script to the main camera

```
public class RenderStereoReflections : MonoBehaviour
{
        public GameObject reflectiveObj;
        public GameObject leftReflCamera;
        public GameObject rightReflCamera;
        int eyeIndex = 0;

        void OnPreRender(){          ← Optimize further by adding a condition about reflective object's visibility
                if (eyeIndex == 0){
                        // Render Left camera
                        leftReflCamera.GetComponent<Camera>().Render();
                        reflectiveObj.GetComponent<Renderer>().material.SetTexture("_DynReflTex", leftReflCamera.GetComponent<Camera>().targetTexture );
                }
                else{
                        // Render right camera
                        rightReflCamera.GetComponent<Camera>().Render();
                        reflectiveObj.GetComponent<Renderer>().material.SetTexture("_DynReflTex", rightReflCamera.GetComponent<Camera>().targetTexture );
                }
                eyeIndex = 1 - eyeIndex;
        }
}
```
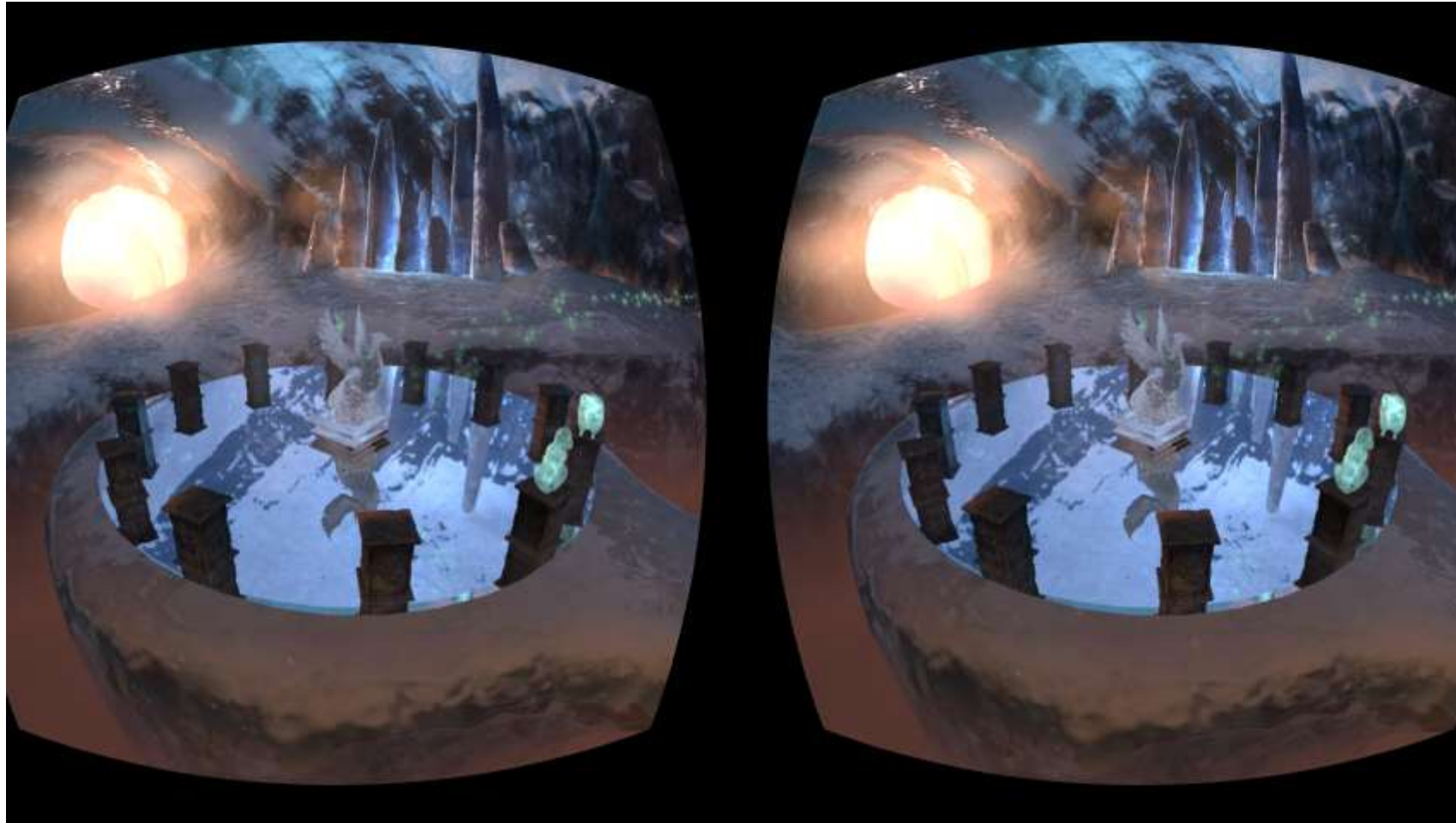
ARM

Check Left/Right Reflection Synchronization

ARM
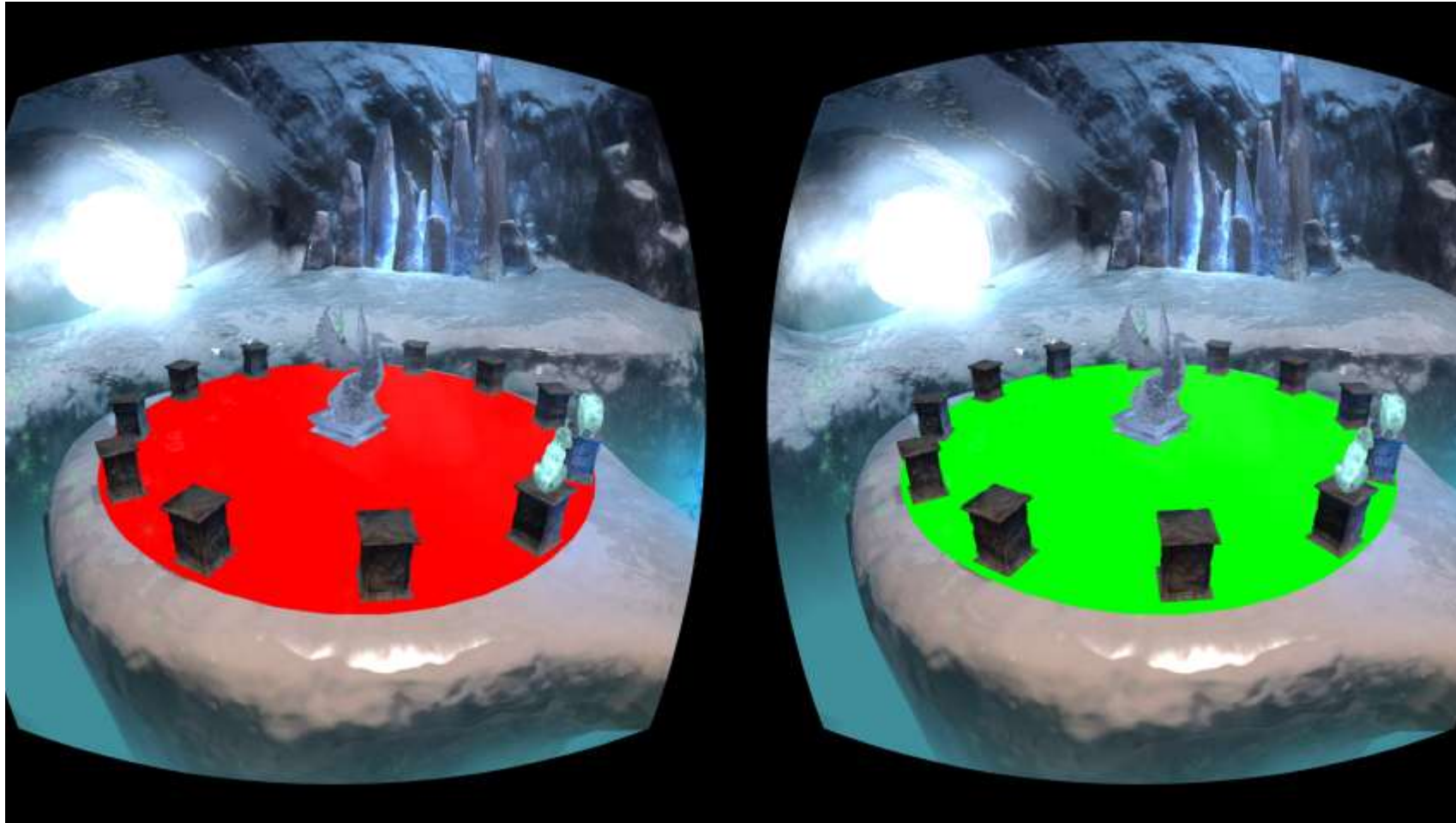
Stereo Reflections in Ice Cave VR

ARM

Wrap Up

- The use of techniques based on local cubemap allows rendering high quality and efficient shadows and reflections from static geometry. These methods can be combined with other runtime rendering techniques to render reflections from dynamic objects.

- Stereo reflections in VR improve user experience

  https://community.arm.com/groups/arm-mali-graphics/blog/2016/03/10/combined-reflections-stereo-reflections-in-vr

- Check out The ARM Guide for Unity Developers for very efficient rendering techniques to make the most out of Unity when developing mobile VR games

  ⓘ ARM Guide for Unity Developers at MaliDeveloper.arm.com

ARM

Thank you!

ARM

To Find Out More….

- Find out more about techniques based on local cubemaps at:
  - https://community.arm.com/groups/arm-mali-graphics/blog/2016/03/10/combined-reflections-stereo-reflections-in-vr
  - http://malideveloper.arm.com/documentation/developer-guides/arm-guide-unity-enhancing-mobile-games/
  - http://community.arm.com/groups/arm-mali-graphics/blog/2015/04/13/dynamic-soft-shadows-based-on-local-cubemap
  - http://community.arm.com/groups/arm-mali-graphics/blog/2014/08/07/reflections-based-on-local-cubemaps
  - http://community.arm.com/groups/arm-mali-graphics/blog/2015/04/13/refraction-based-on-local-cubemaps
  - http://community.arm.com/groups/arm-mali-graphics/blog/2015/05/21/the-power-of-local-cubemaps-at-unite-apac-and-the-taoyuan-effect

ARM