# 硬件光追引擎文档

队伍名称:提瓦特观光队 成员:余斐然、朱鹏辉、刘剑澎 学校:华南理工大学

#### 概述

本队伍提交作品为自制渲染器, 依赖库为 glm、imgui、volk、stb、tinygltf、Vulkan、glfw3。

#### 编译

- 1. 安装 Vulkan SDK
- 2. 使用 CMake 生成项目
  - (1) 首先解压项目压缩包, 在根目录下创建 build 文件夹, 打开 cmake 并填写源码路径和 build 路径。

🚞 > 此电脑 > 本地磁盘 (C:) > vulkan-study	r-cmake	✓ C 在 vulkan→	
名称 ^	CMake 3.24.0-rc4 - C:/vulkan-study-cmake/build	-	- o ×
🖿 .git	Eile Iools Options Help		
🛅 assets	Where is the source code: C:/vulkan-study-cmake		Browse Source
🛅 bin	Where to build the binaries: C:/vulkan=study=cmake/build	~	Browse Build
🔁 build	Sgarch:	d Entry Management	Environment
shaders	Name Value		
src			
🗖 third party			
j .gitattributes	Press Configure to update and display new values in red, then press Generate to	generate selected build	l files.
a .gitignore	Configure Cenerate Open Project Current Generator: None		
CMakel ists tyt			

(2) 点击 Configure,选择生成工程的 visual studio 版本,点击 Finish。

🛦 CMake 3.24.0-rc4 - C:\vulkan-study-cmake\build — 🗆 🗙					
<u>F</u> ile <u>T</u> ools <u>O</u> ptions <u>H</u> elp					
Where is the source code:	C:/vulkan-study-cmake	Browse Source			
Preset:	<custom> 2</custom>	×			
Where to build the binaries:	C:/vulkan-st	Browse <u>B</u> uild			
S <u>e</u> arch:	$\leftarrow \blacktriangle$	Entry Environment			
Name Press Configure to Configure Generate Ope	Specify the generator for this project Visual Studio 17 2022 Optional platform for generator(if empty, generator uses: x Optional toolset to use (argument to -T) • Use default native compilers • Specify native compilers • Specify toolchain file for cross-compiling • Specify options for cross-compiling • Einish Cancel	vild files.			

#### (3) 点击 Generate, 生成对应的工程文件

🛦 CMake 3.24.0-rc4 - C:\vulkan-study-cmake\build — 🗆 🗙						×		
<u>File T</u> ools <u>O</u> ptions <u>H</u> elp	<u>File Tools Options H</u> elp							
Where is the source code:	C:/vulkan-study-cmake					Browse <u>S</u> e	ource	
Preset:	<custom></custom>				~			
Where to build the binaries:	C:/vulkan-study-cmake/build				~	Browse Build		
S <u>e</u> arch:		🗹 Grouped	🔽 Advanced	🗣 Add Entry	🗱 <u>R</u> emove Entr	y E <u>n</u> viror	ment	
Name		Value						
> Ungrouped Entries > CMAKE > GLFW > Vulkan Press Configure to update and display new values in red, then press Generate to generate selected build files.								
BUILDING SHADER C:/vulkan-study-cmake/shaders/raytrace.rmiss BUILDING SHADER C:/vulkan-study-cmake/shaders/raytrace.rchit BUILDING SHADER C:/vulkan-study-cmake/shaders/raytraceShadow.rmiss BUILDING SHADER C:/vulkan-study-cmake/shaders/raytrace.rahit Configuring done								
e								

(4) 点击 Open Project, 打开生成的工程文件

A CMake 3.24.0-rc4 - C:\vulkan-study-cmake\build —						
<u>F</u> ile <u>T</u> ools <u>O</u> ptions <u>H</u> elp						
Where is the source code:	C:/vulkan-study-cmake				Browse <u>S</u> our	rce
Preset:	<custom></custom>			$\sim$		
Where to build the binaries:	C:/vulkan-study-cmake/build			×	Browse <u>B</u> ui	1d
S <u>e</u> arch:		🔽 Grouped 🔽 Advanced	🗣 Add Entry	🗱 <u>R</u> emove Entry	Environm	ent
Name		Value				
<ul> <li>&gt; Ungrouped Entries</li> <li>&gt; CMAKE</li> <li>&gt; GLFW</li> <li>&gt; Vulkan</li> <li>&gt; Press Configure to update and display new values in red, then press Generate to generate selected build files.</li> </ul>						
Configure Generate Open Project Current Generator: Visual Studio 17 2022 C:/vulkan-study-cmake/shaders/raytrace.rmiss BUILDING SHADER C:/vulkan-study-cmake/shaders/raytraceShadow.rmiss BUILDING SHADER C:/vulkan-study-cmake/shaders/raytrace.rahit Configuring done Generating done						

- 3. 在 Visual Studio 中编译运行(需按照 "配置" 章节放置资源文件)
  - (1) 在放置完资源文件后,将"vulkan\_study"设置为启动项目

	e	- 🗆 ×	
<ul> <li>○ · ○ 節 · ■ 回 ○ · ○ · Debug · x64 · ▶ 本地 Windows 调试器 • ▷ ● · ■ 同 ·</li> <li>解決方案资源管理器</li> <li>○ ○ ① ② ① ③</li> <li>② ③ ② ③ ② ③</li> <li>② ③ ③ ③ ③</li> <li>◎ ③ 称決方案资源管理器</li> <li>◎ ③ 称決方案</li> </ul>	#	生成(U) 重新生成(E) 清理(N) 査看(W) 分析和代码清理(Z) (双用于项目(J)	
<ul> <li>▶ ■ CMakePredefit</li> <li>▶ ■ GLFW3</li> <li>▶ ■ GLFW3</li> <li>▶ • ₩ imgui</li> <li>▶ • ₩ Shaders</li> </ul>	<b>2</b>	限定为此范围(S) 新建解决方案资源管理器视图(N) 生成依赖项(B)	
▷ o m volk ▷ o m vulkan_study	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	添加(D) 美向导(Z) Ct 管理 NuGet 程序包(N)	↓ rl+Shift+X
	¢	配置启动项目 设为启动项目(A) Ct 调试(G)	rl+Shift+O ►
	x	Git(l) 剪切(T) Ct	► rl+X
查找符号结果	6 ×	活動(P) CC 移除(V) Da 重命名(M) F2	rl+V el
■注 vulkan_study 項目量性 注意 ひ 根念名の间 (名約) 开发者 PowerShell 错误列表 輸出 查找符号结果 指定项目名称。		卸载项目(L) 加载重整化赖须 加载整个体兼关系树 重新扫描解决力案(S) 显示词恋赏规律模误 清除词恋戏超序错误	
□ 此项不支持预览 1↓ 0 / 0 ▲ 🖉 99+ 💱 main ▲ 🛱 vulkan-	<i>₹</i>	在文件资源管理器中打开文件夹(X) 在终端中打开	
	ه بر	Re性(R) Al	t+Enter

#### (2) 点击调试按钮,即可启动渲染器



配置

需要将比赛场景资源放置在 /assets/ 中, 即路径应为:

assets +

- Caustics
- Deferred
- GI
- PBR
- Shadow





画面分为红框中的 UI 区域,以及其余的非 UI 区域。

1. 镜头操作

在非 UI 区域中按下 左 ALT 键, 就可以控制摄像头旋转与移动:

- 鼠标控制视角旋转;
- WASD 控制水平移动;
- 按下 左 CTRL 键 下降, 空格键 上升
- 再次按下 左 ALT 键 结束控制
- 2. UI 区域

j	▼ Debug			
0	R:128	G:204	B:230	Clear color
100	🔽 Ray Tracer	r mode		
	🔻 Light		<u> </u>	
8	🔵 Point 🔵	Infinite		
8	2. 106	6.896	-0 <mark>,2</mark> 61	Position
		6.692		Intensity
	🕨 Ray Traci	ng		
		10000		Max Iteration 🙎
		5		Max Ray Depth
١		1		Samples Per Frame
	▶ Scenes			
١	GI			Scene 3
	Application a Frame index 8	verage 26.48 306	6 ms/frame (37.	8 FPS)

其中:

- 1) 改变光源参数, 可选点光源和环境光源, 改变位置、方向和光强;
- 光线追踪参数,可选最大累计帧数,光线迭代深度,以及每帧中每个像素的采样次数;
- 3) 可以切换场景。

此外,可以按下H键来隐藏和显示UI。

### 运行结果展示

设备信息:

CPU: AMD 5600X

RAM: 32GB DDR4 3200MHz

GPU: NVIDIA RTX3070 (8GB)

### a. Caustics



### b. Deferred



c. Gl



### d. PBR





## e. Shadow



