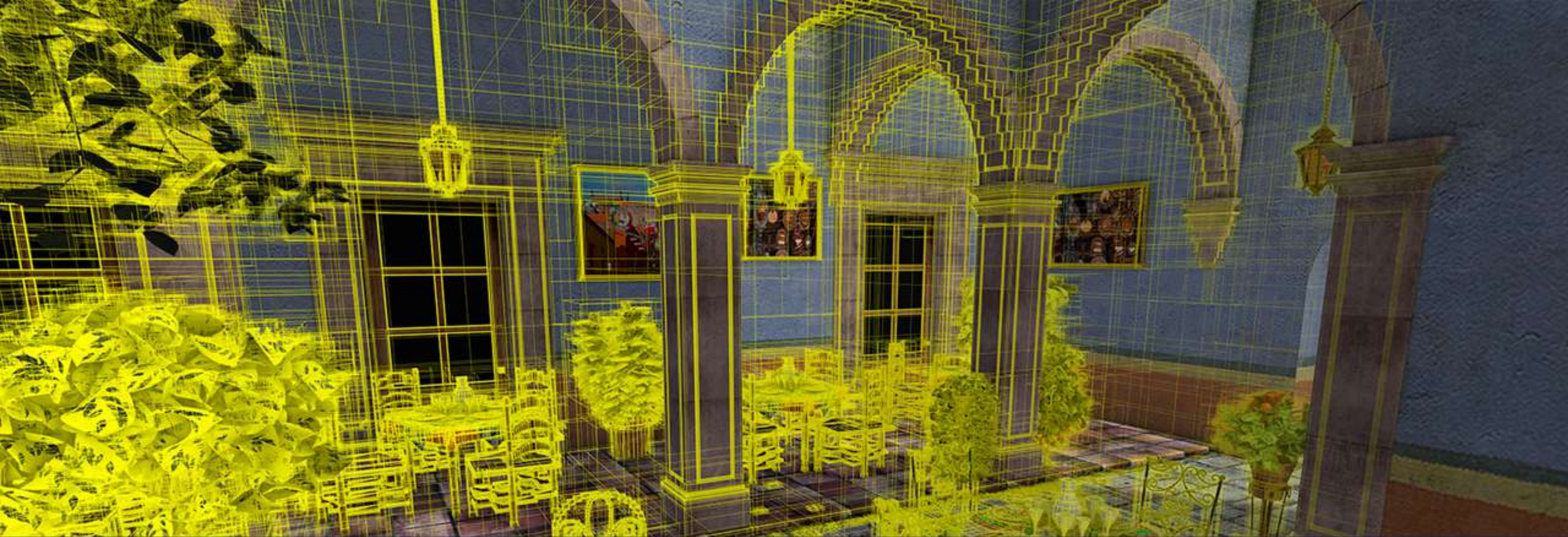




A GPU-accelerated LiDAR for generating labelled datasets

Alfonso López¹, Carlos J. Ogayar¹, Francisco R. Feito¹

¹ Departamento de Informática, Universidad de Jaén

CEIG21

Málaga - 22-24 de septiembre



Universidad
de Jaén

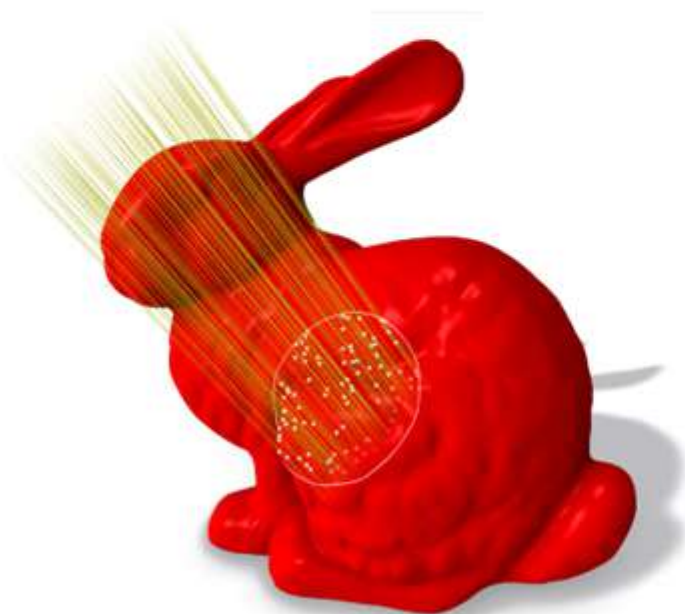
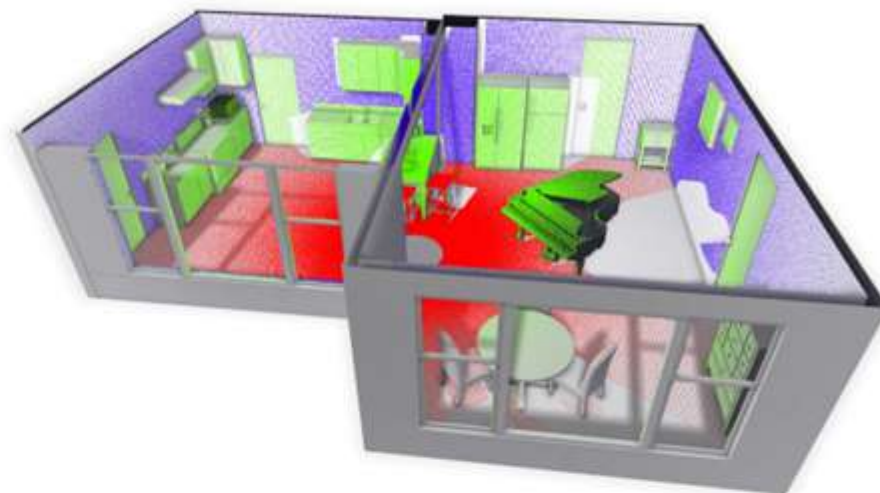


(Incluye narración)



► Relevancia de sensores LiDAR

- Elevado número de aplicaciones
- Gran variedad de sensores LiDAR
 - ▷ Propiedades: velocidad de adquisición de datos, rango, precisión, etc.
 - ▷ Plataforma desde la cual se opera



► Datos LiDAR relevantes para algoritmos de Inteligencia Artificial

- Ejemplos
 - ▷ Conducción autónoma
 - ▷ Segmentación/Clasificación de nubes de puntos
- Adquisición de datos clasificados
 - ▷ Escasez de conjuntos de datos etiquetados
 - ▷ Etiquetado manual

► Relevancia de la simulación para la obtención de conjuntos de datos LiDAR:

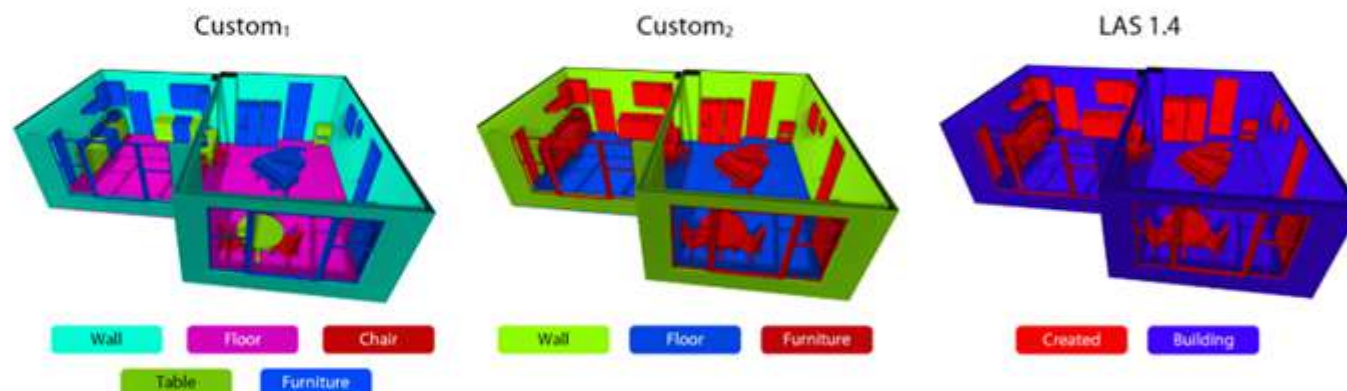
► Simulación de un sensor LiDAR parametrizado

- Elimina la necesidad de disponer de múltiples dispositivos LiDAR
- Planificación de sesión de captura de datos LiDAR

► Clasificación de entorno

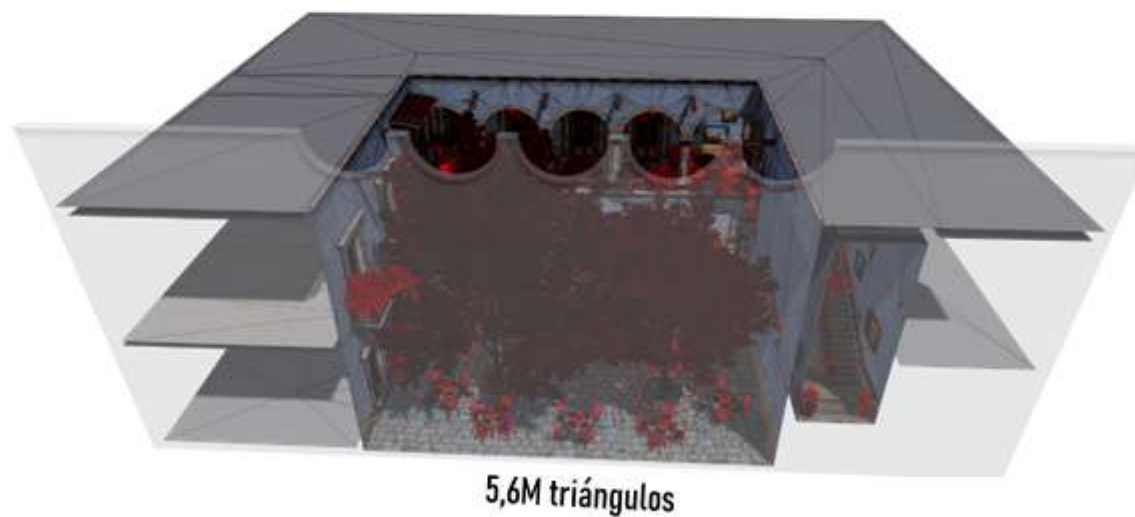
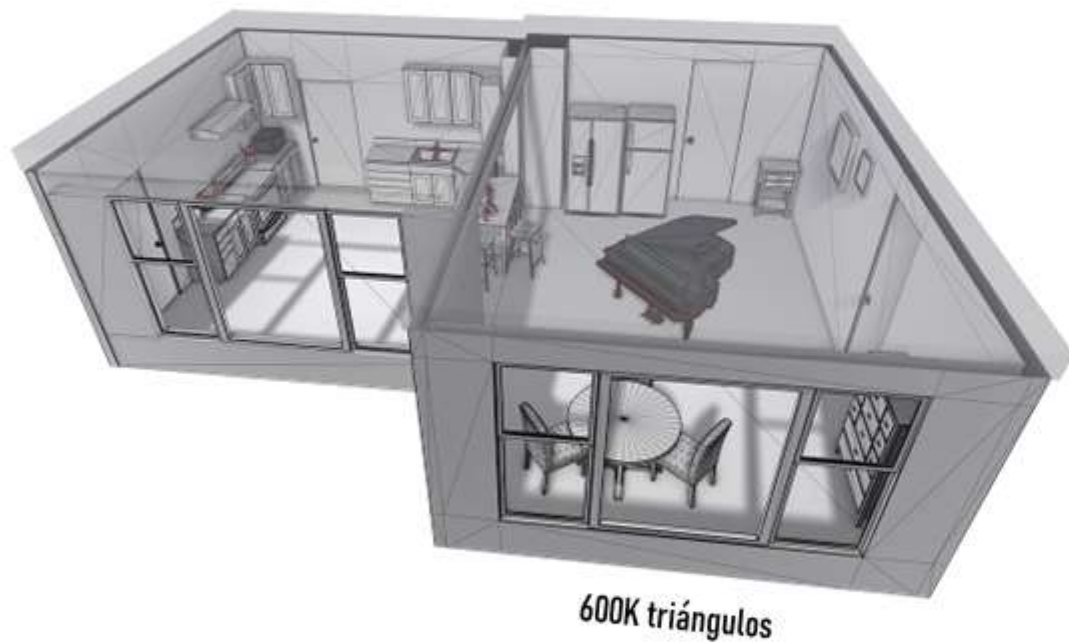
- A cualquier nivel de detalle
- Sin errores de etiquetado

► Aceleración GPU



► Entornos sintéticos

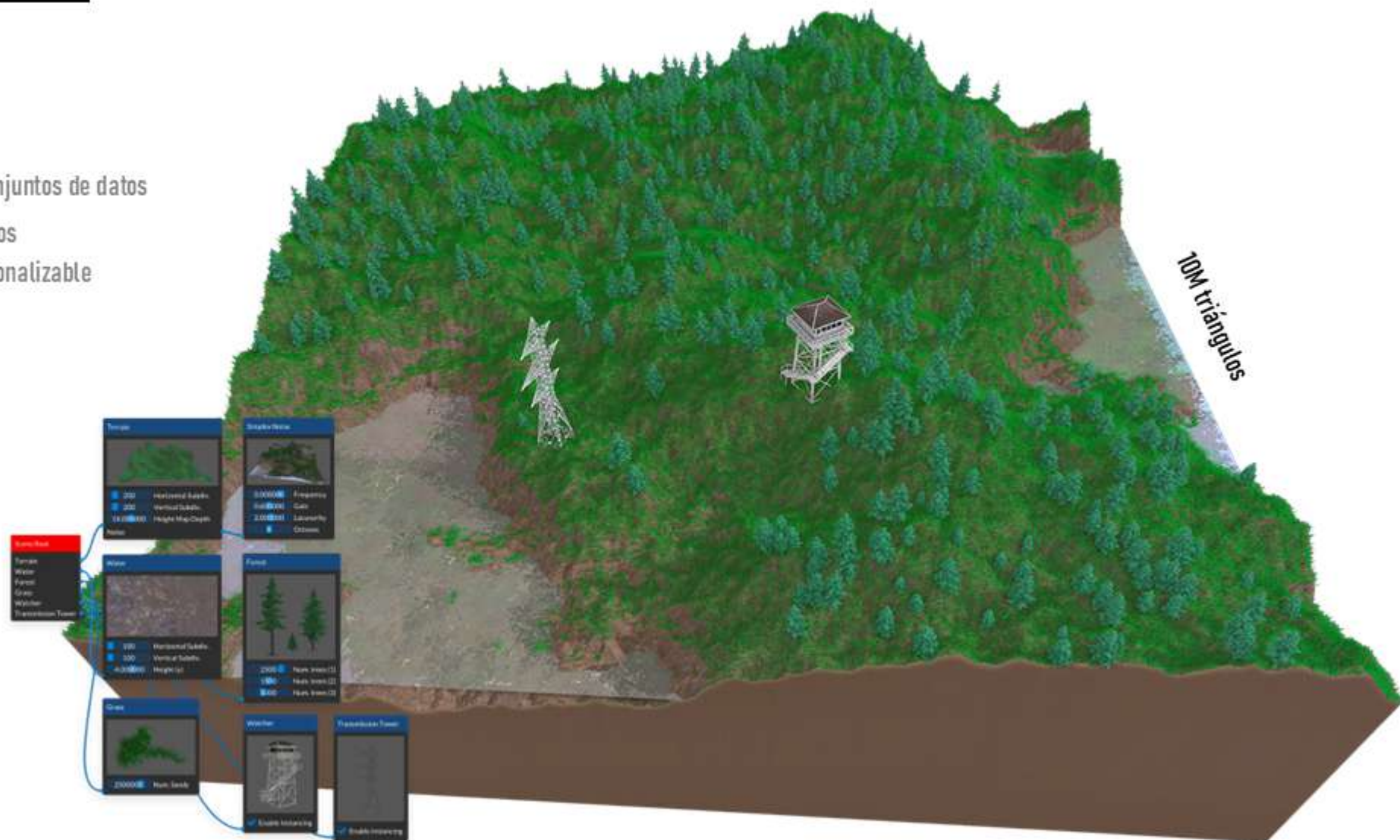
- Modelado previamente (Entorno estático)



► Entornos procedurales

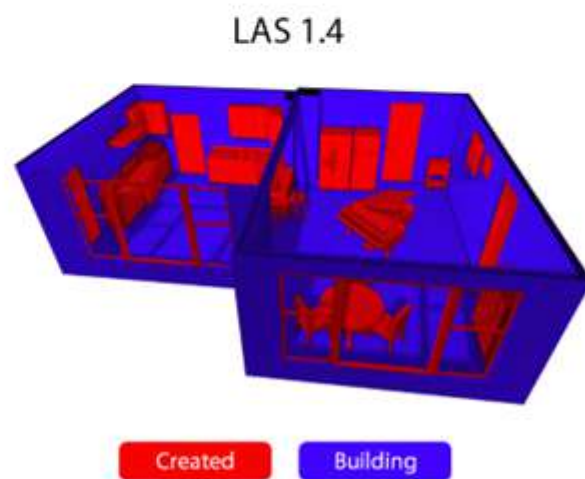
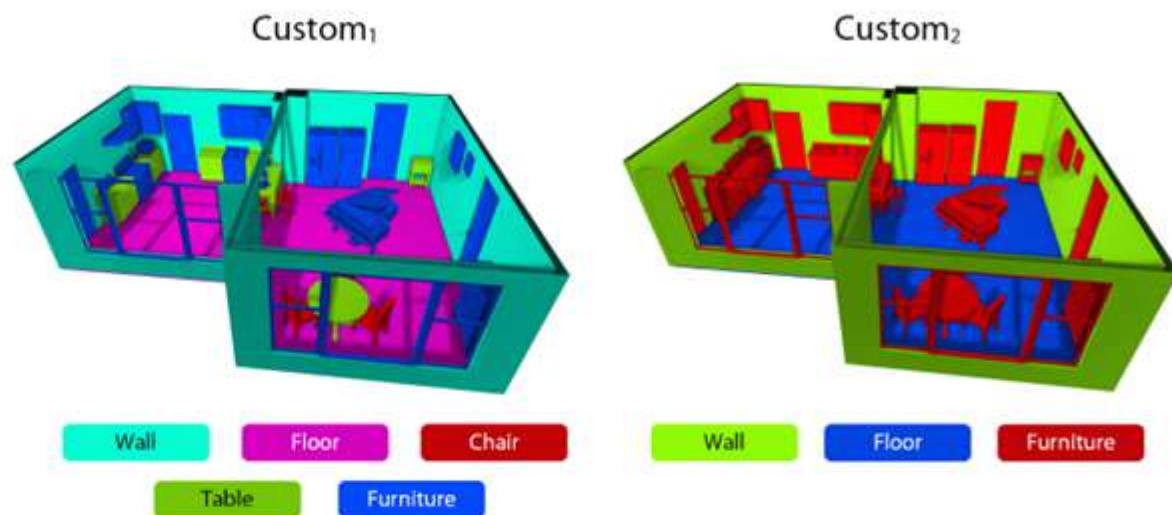
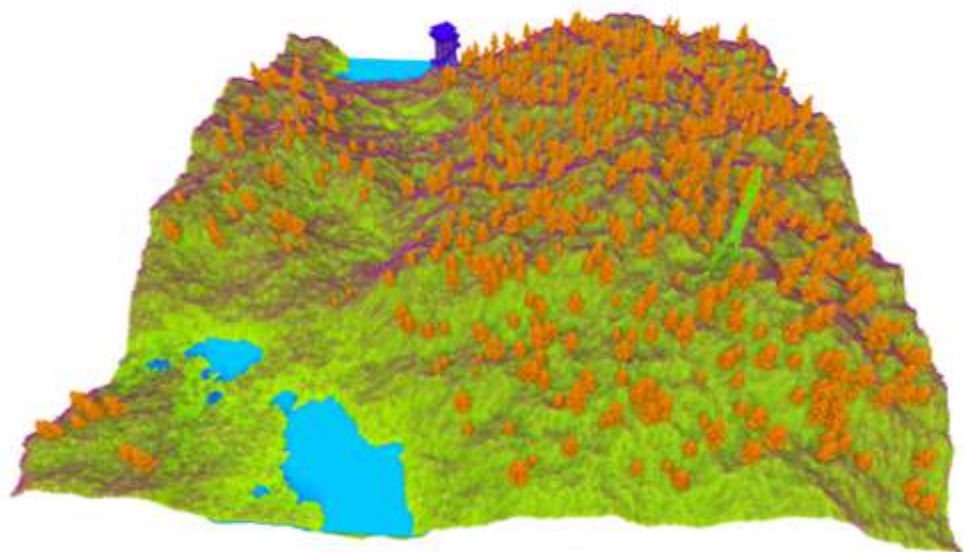
► Modelado procedural

- Generación de conjuntos de datos
- Diferentes entornos
- Complejidad personalizable



► Etiquetado de mallas de triángulos

- Custom
- Estándar LAS 1.4



Estructuración de entorno

► Optimización de consultas espaciales

► Ray-casting

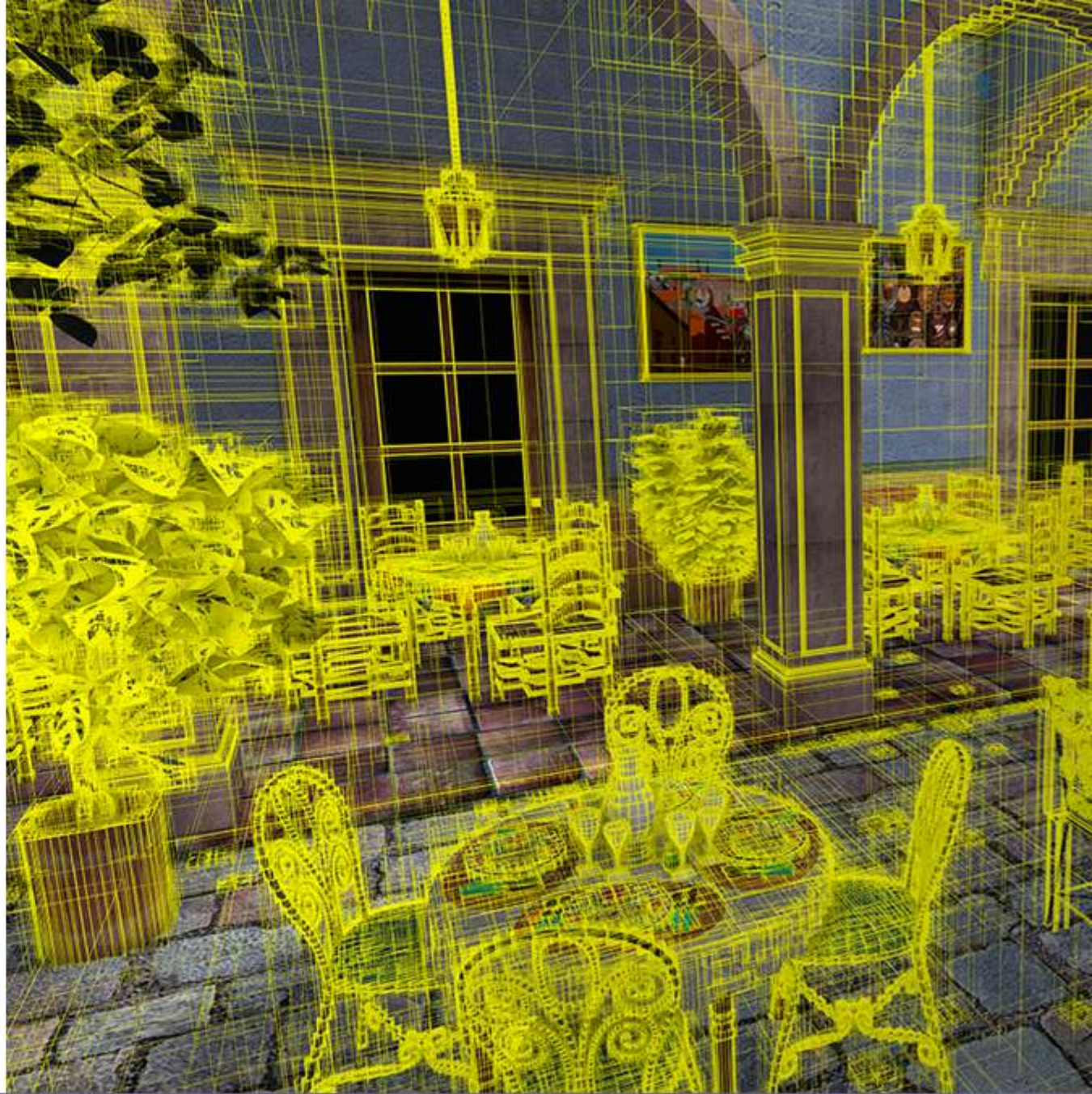
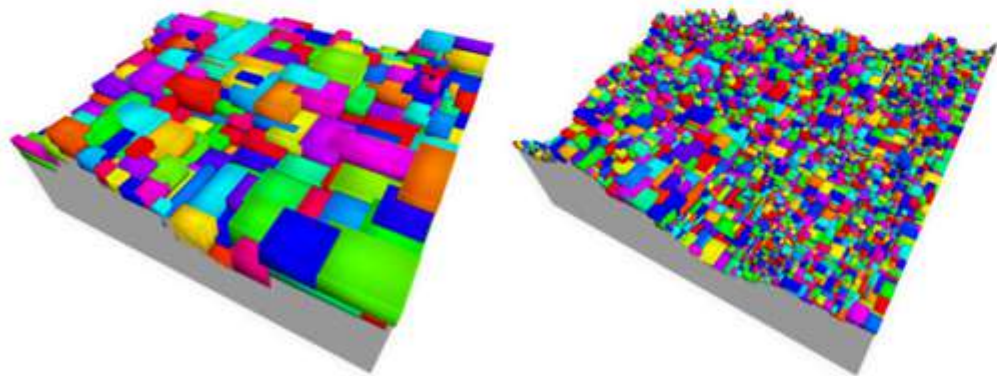
► Boundary Volume Hierarchy (BVH)

▷ Estructura de datos más común en ray-tracing

- ▷ **Parallel Locally-Ordered Clustering for Bounding Volume Hierarchy Construction.** Daniel Meister, Jiri Bittner. IEEE Transactions on Visualization and Computer Graphics, IEEE. 2018.

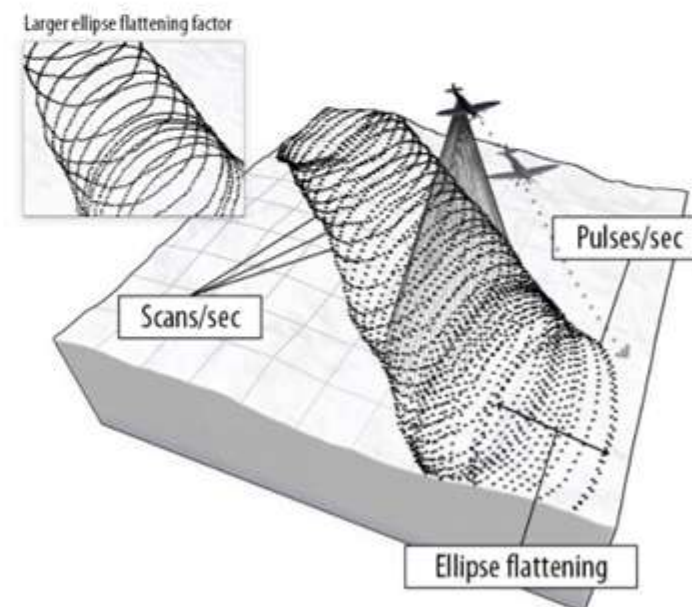
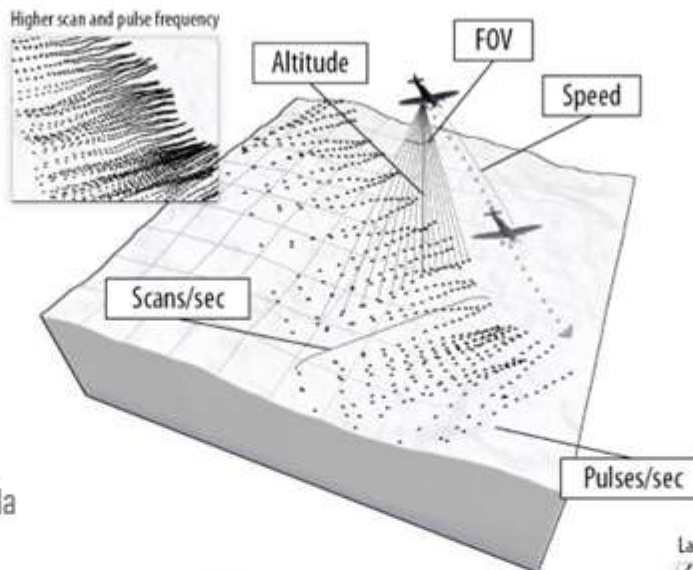
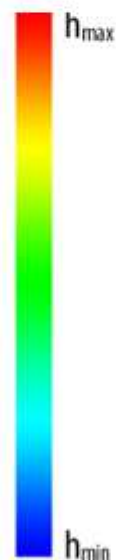
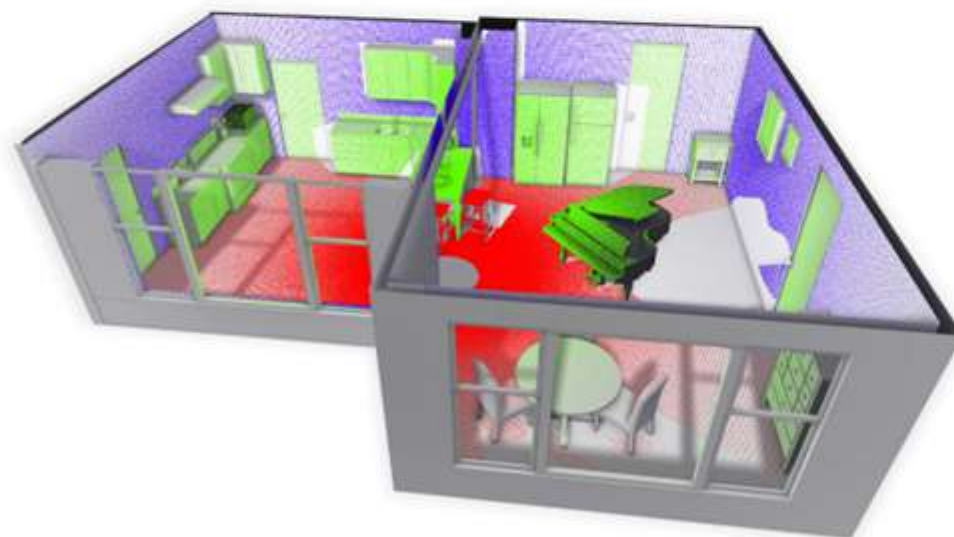
a) $d = 8$

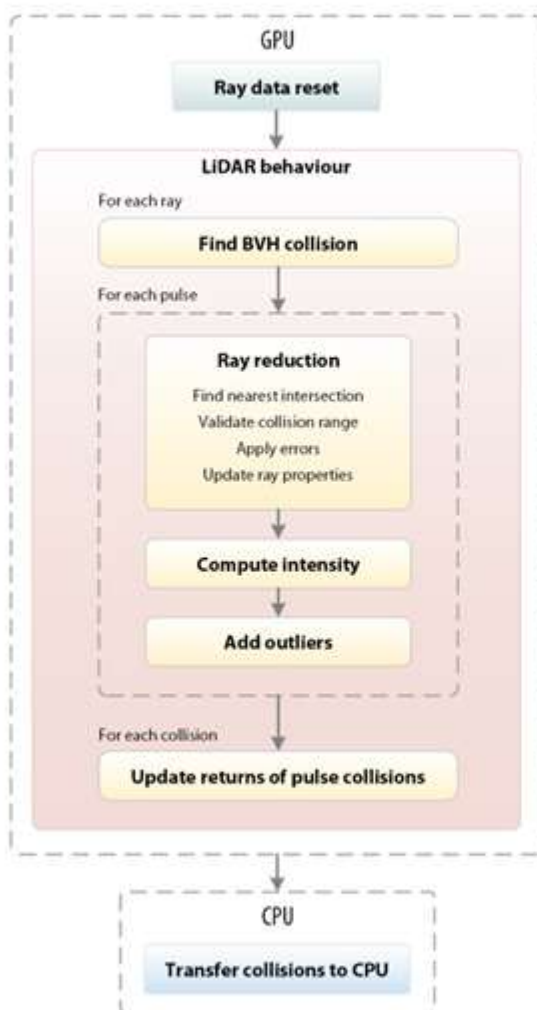
b) $d = 12$



► Generación de rayos

- Millones de rayos
- Construcción en GPU
 - ▷ LiDAR terrestre (TLS)
 - ▷ LiDAR aéreo (ALS)
 - ▷ Paralelo, zig-zag, elíptico, ruta personalizada





► Procedimiento de simulación en GPU

9

1. Inicialización de información de rayos

- Energía

2. Búsqueda de colisiones

- Spatial queries (BVH)

3. Reducción de rayos

- Rayos dentro de un pulso actúan como una unidad
- 1 colisión por iteración
- Simulación de errores

4. Cálculo de intensidad

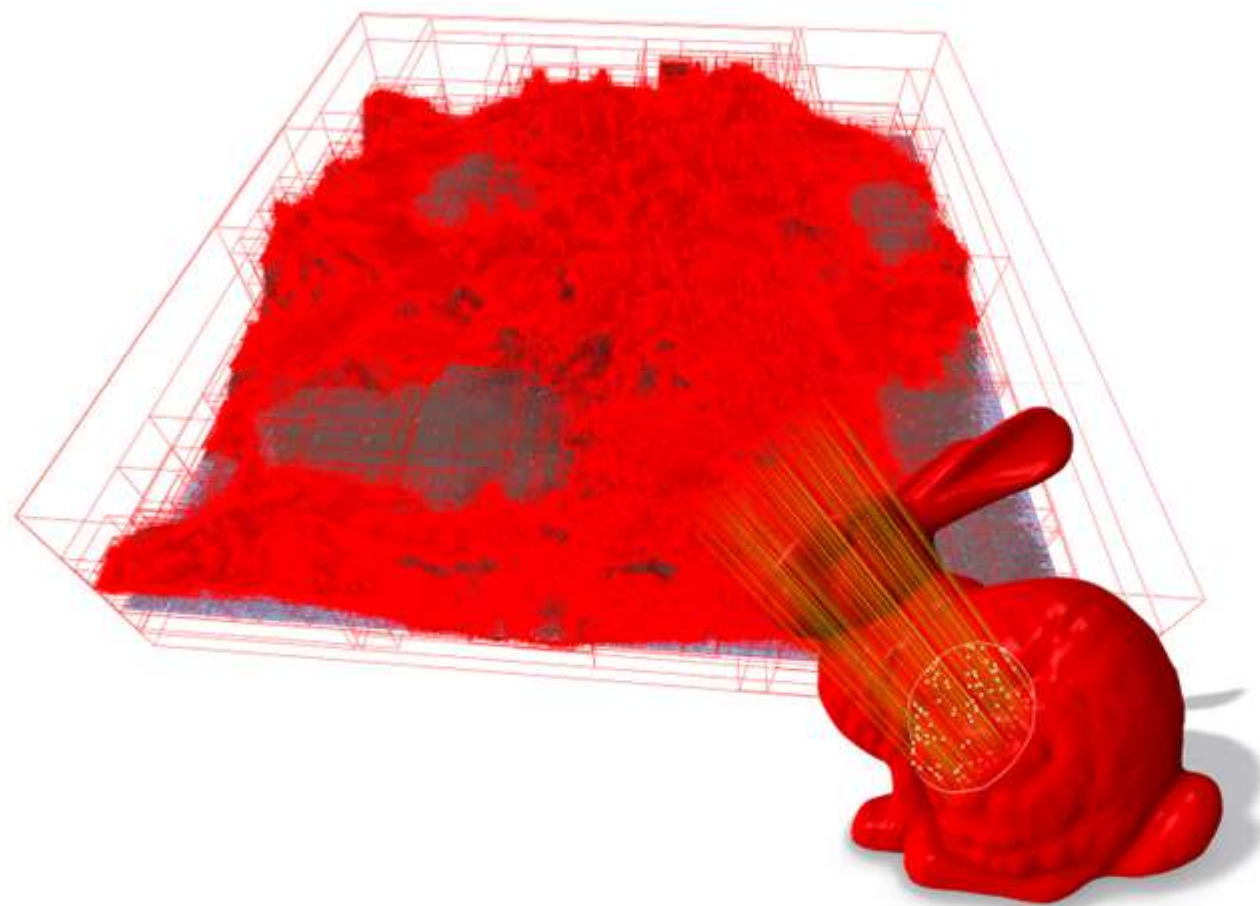
- Material de superficie
- Propiedades de colisión: ángulo, distancia, etc.

5. Generación de valores anómalos

- Distribución aleatoria personalizada
- Sustitución de colisiones válidas

6. Actualización de retornos

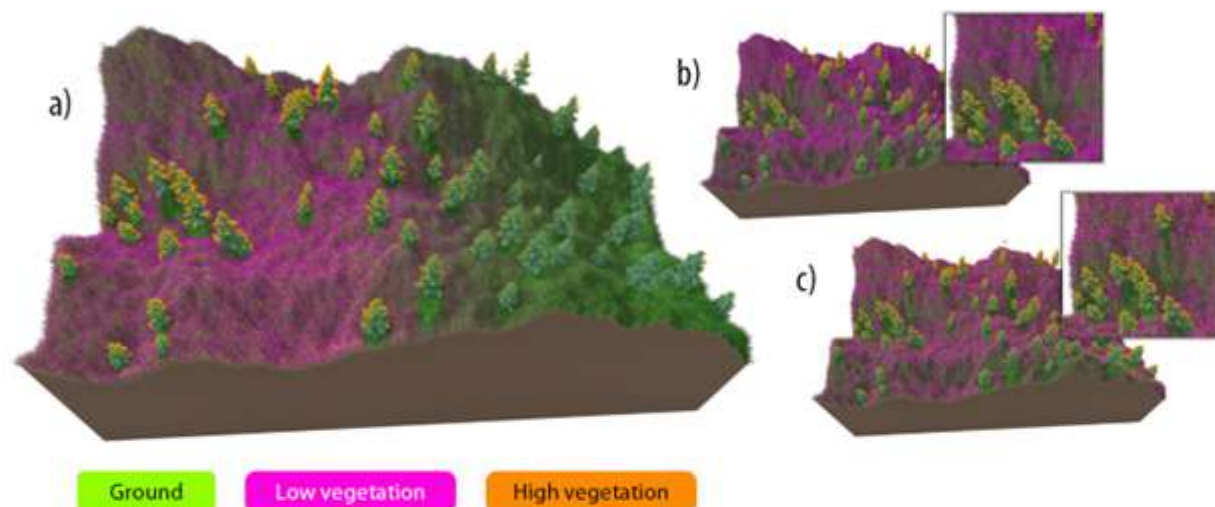
- Colisión → Id. de retorno, Número de retornos de pulso



► Procedimiento de simulación en GPU

10

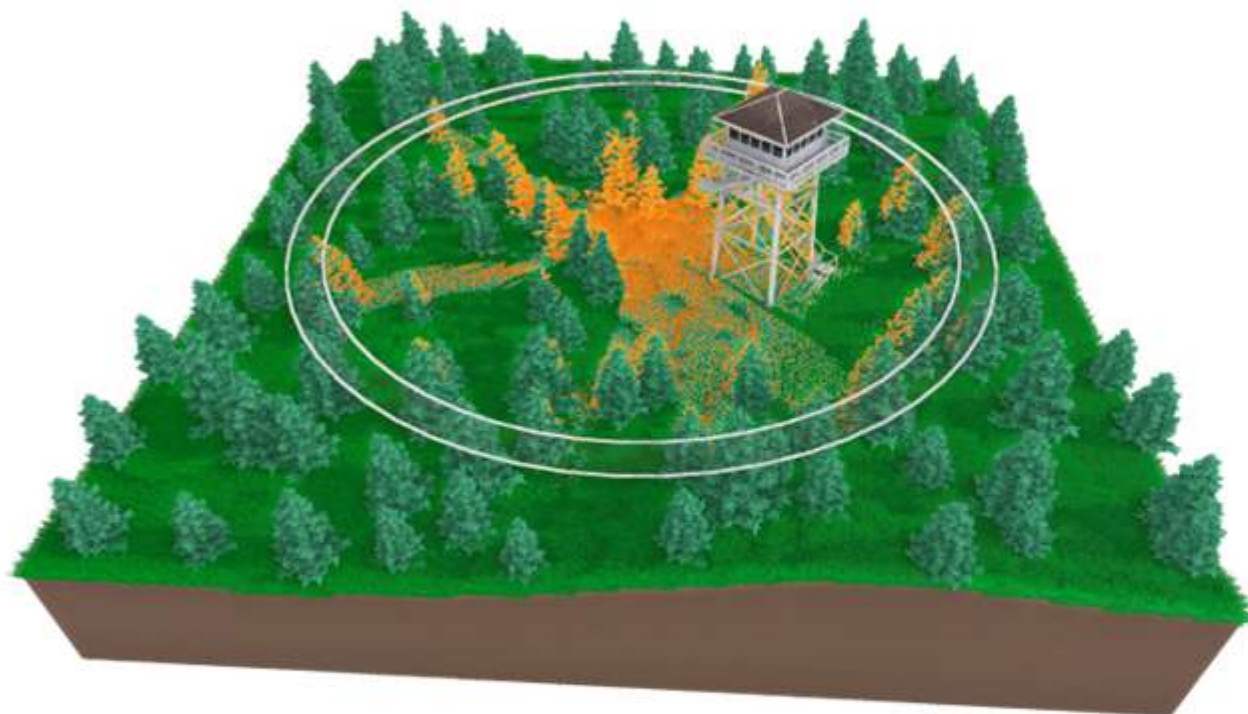
1. Inicialización de información de rayos
 - ▷ Energía
2. **Búsqueda de colisiones**
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión → Id. de retorno, Número de retornos de pulso



► Procedimiento de simulación en GPU

11

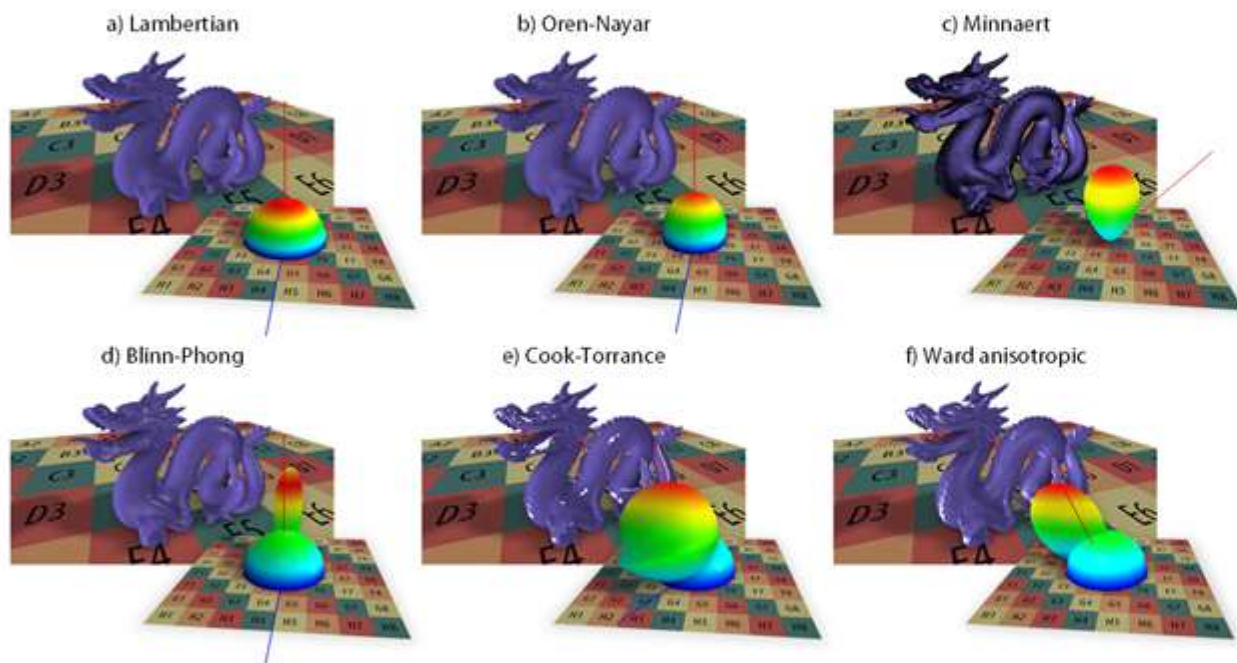
1. Inicialización de información de rayos
 - ▷ Energía
2. Búsqueda de colisiones
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión → Id. de retorno, Número de retornos de pulso



► Procedimiento de simulación en GPU

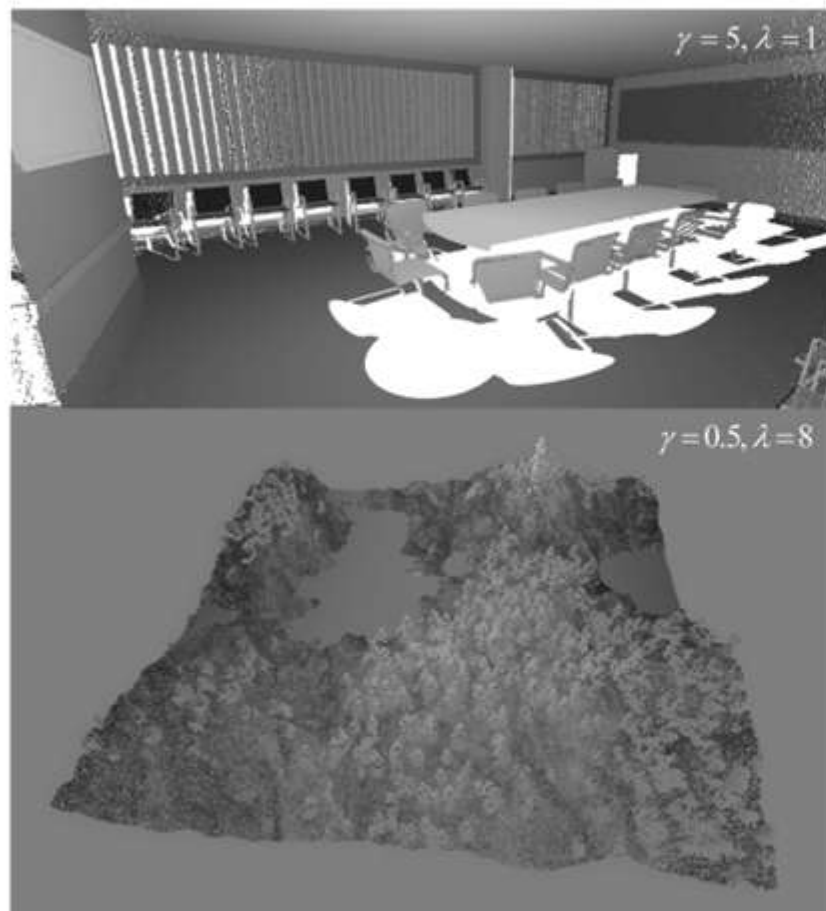
12

1. Inicialización de información de rayos
 - ▷ Energía
2. Búsqueda de colisiones
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión → Id. de retorno, Número de retornos de pulso



► Procedimiento de simulación en GPU

1. Inicialización de información de rayos
 - ▷ Energía
2. Búsqueda de colisiones
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión → Id. de retorno, Número de retornos de pulso



Tone Mapping

Hable, J., 2010. *Uncharted 2: HDR Lighting*. Game Developers Conference

► Procedimiento de simulación en GPU

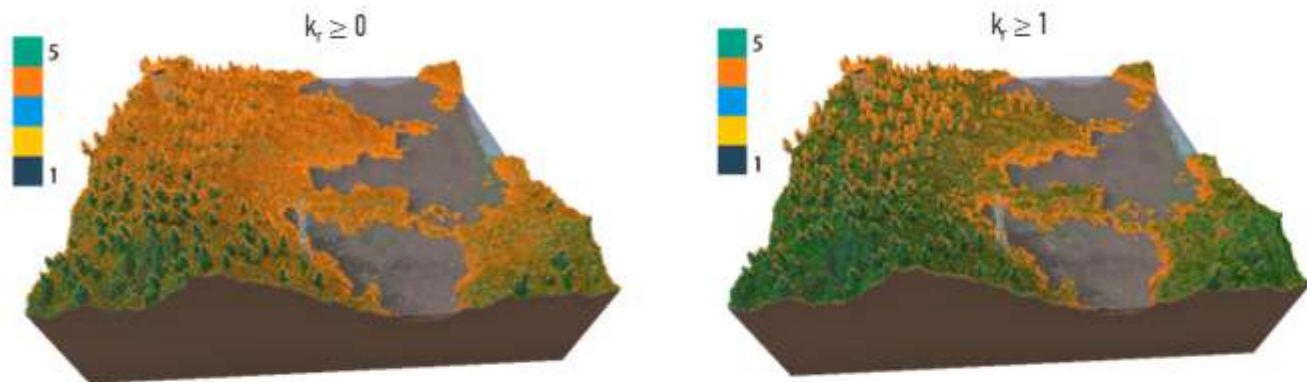
1. Inicialización de información de rayos
 - ▷ Energía
2. Búsqueda de colisiones
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión $\rightarrow$ Id. de retorno, Número de retornos de pulso



► Procedimiento de simulación en GPU

15

1. Inicialización de información de rayos
 - ▷ Energía
2. Búsqueda de colisiones
 - ▷ Spatial queries (BVH)
3. Reducción de rayos
 - ▷ Rayos dentro de un pulso actúan como una unidad
 - ▷ 1 colisión por iteración
 - ▷ Simulación de errores
4. Cálculo de intensidad
 - ▷ Material de superficie
 - ▷ Propiedades de colisión: ángulo, distancia, etc.
5. Generación de valores anómalos
 - ▷ Distribución aleatoria personalizada
 - ▷ Sustitución de colisiones válidas
6. Actualización de retornos
 - ▷ Colisión $\rightarrow$ Id. de retorno, Número de retornos de pulso



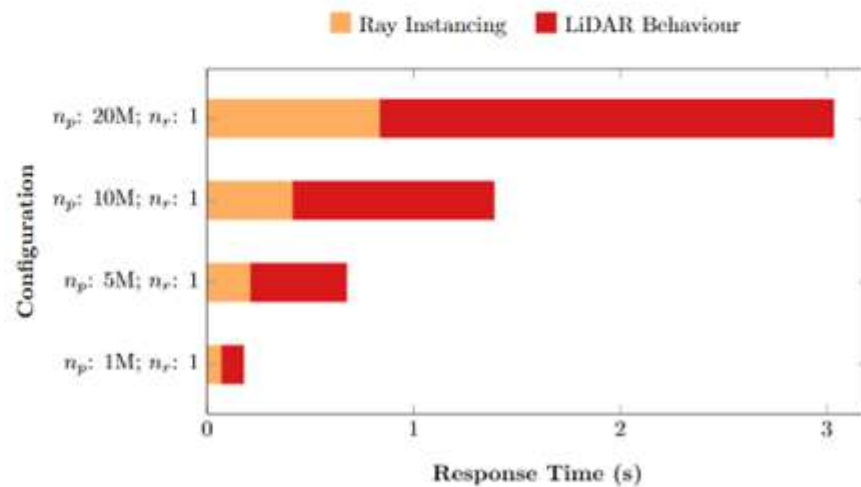
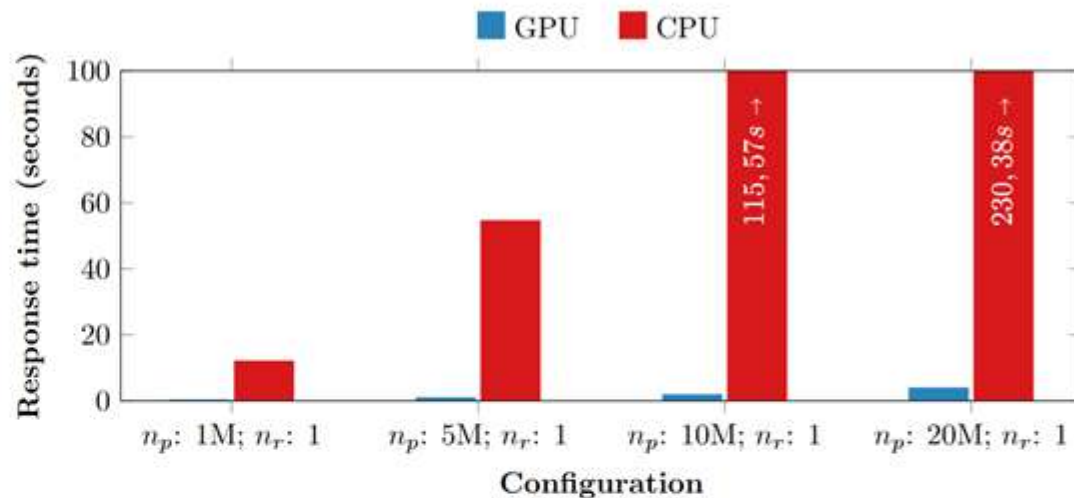
6. Actualización de retornos

- Colisión → Id. De retorno, Número de retornos de pulso
- Filtro por valor absoluto
- Filtro por valor normalizado



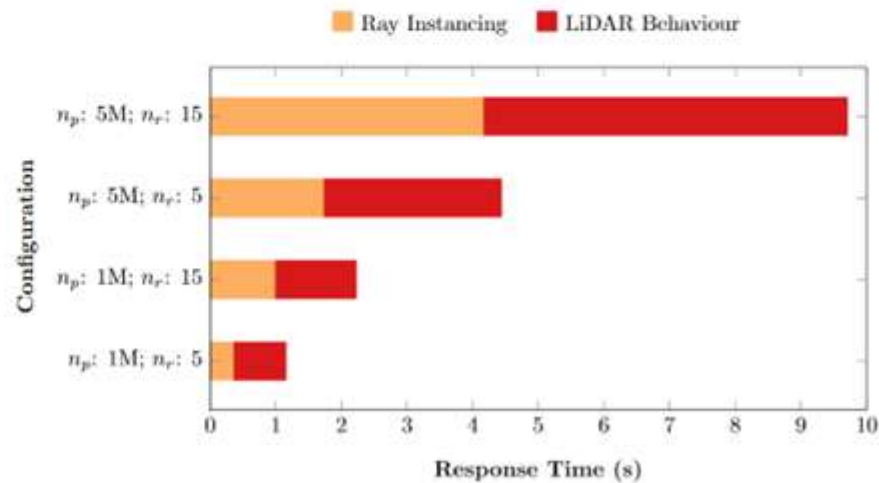
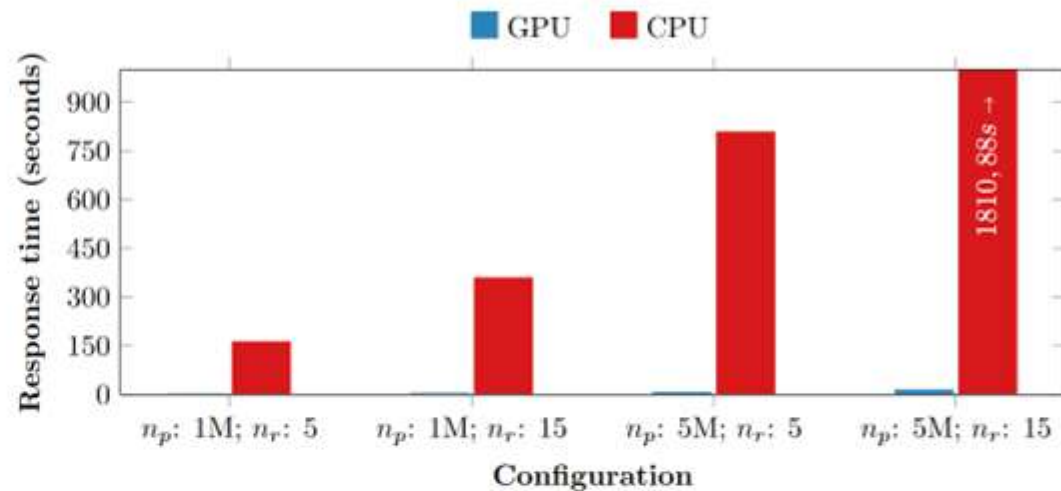
► Tiempo de respuesta de LiDAR terrestre

- ▷ 300.000 triángulos
- ▷ 1 rayo por pulso
- ▷ 1 retorno
- ▷ 360°x150°
- ▷ Nvidia RTX 2080 Ti



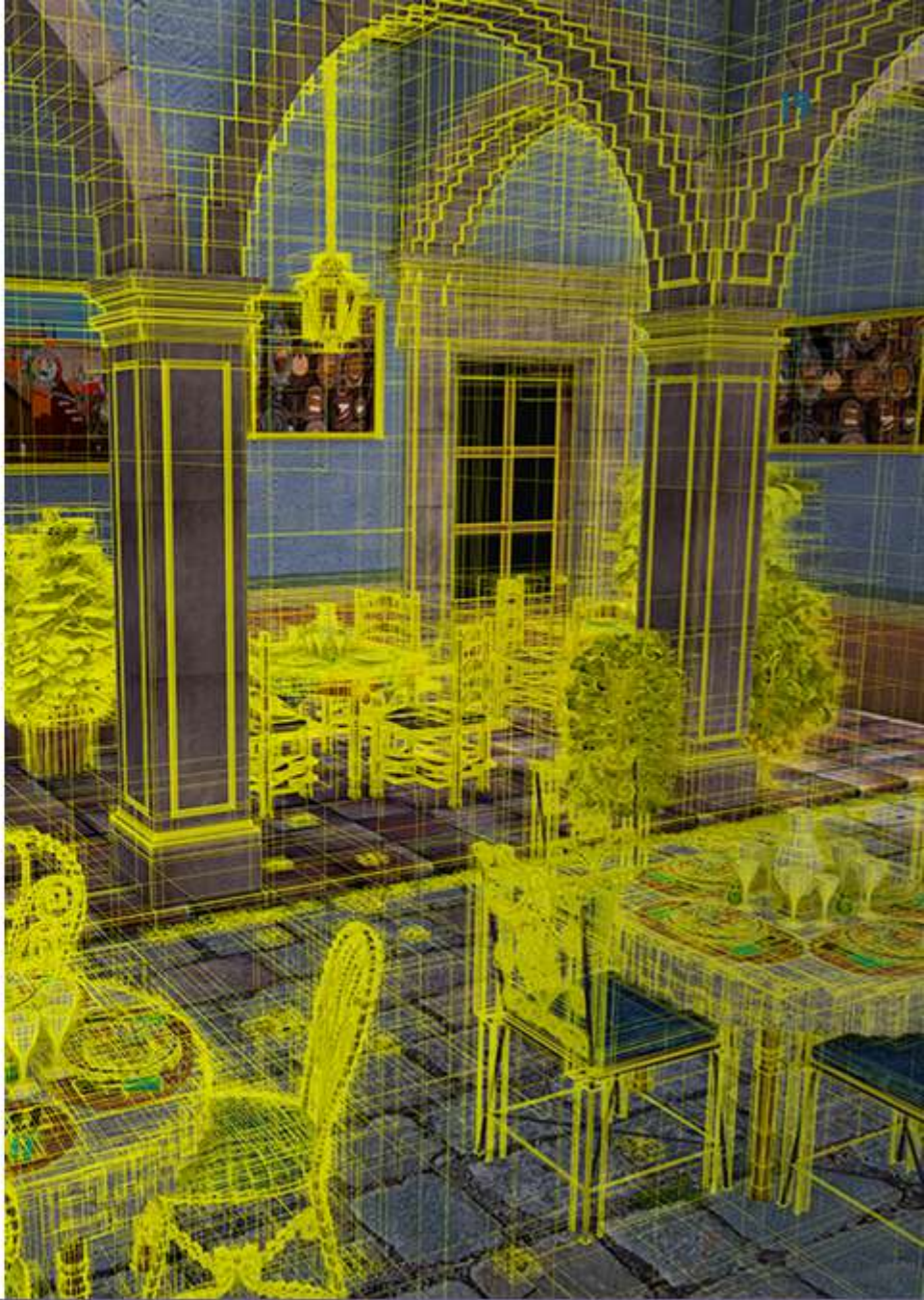
► Tiempo de respuesta de LiDAR aéreo

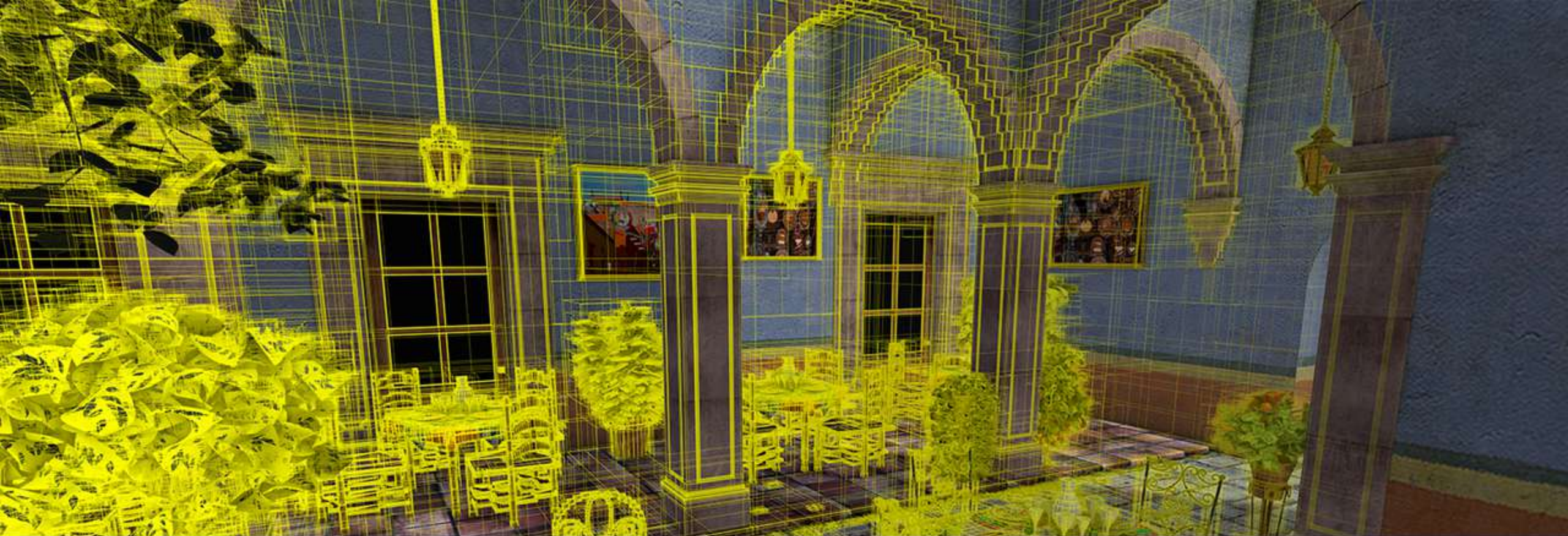
- ▷ 10M triángulos
- ▷ 5/15 rayos por pulso
- ▷ 5 retornos
- ▷ Nvidia RTX 2080 Ti



Trabajo futuro

- ▶ **Evaluación de conjuntos de datos sintéticos en aplicaciones de Inteligencia Artificial**
 - ▶ Aplicabilidad de datos sintéticos en entrenamiento de redes neuronales
 - ▶ Análisis de precisión obtenida
 - ▷ Una precisión similar a la obtenida con conjuntos reales respaldaría el uso de datos sintéticos
- ▶ **Optimización de simulación**
 - ▶ Paralelización de rayos en pulsos
 - ▶ Nuevas extensiones de OpenGL
 - ▷ Comunicación entre hilos
- ▶ **Reducción de uso de memoria durante la simulación**
 - ▶ Adaptación a commodity hardware





A GPU-accelerated LiDAR for generating labelled datasets

Alfonso López¹, Carlos J. Ogayar¹, Francisco R. Feito¹

¹ Departamento de Informática, Universidad de Jaén

CEIG21

Málaga - 22-24 de septiembre



Universidad
de Jaén

