SIGGRAPH ASIA 2020 VIRTUAL

Pre-recorded sessions:
From 4 December 2020

Live sessions:
10 – 13 December 2020

SA2020.SIGGRAPH.ORG
#SIGGRAPHAsia | #SIGGRAPHAsia2020

CPPM: Chi-squared Progressive Photon Mapping

Zehui Lin, Sheng Li, Xinlu Zeng, Congyi Zhang, Jinzhu Jia, Guoping Wang, Dinesh Manocha

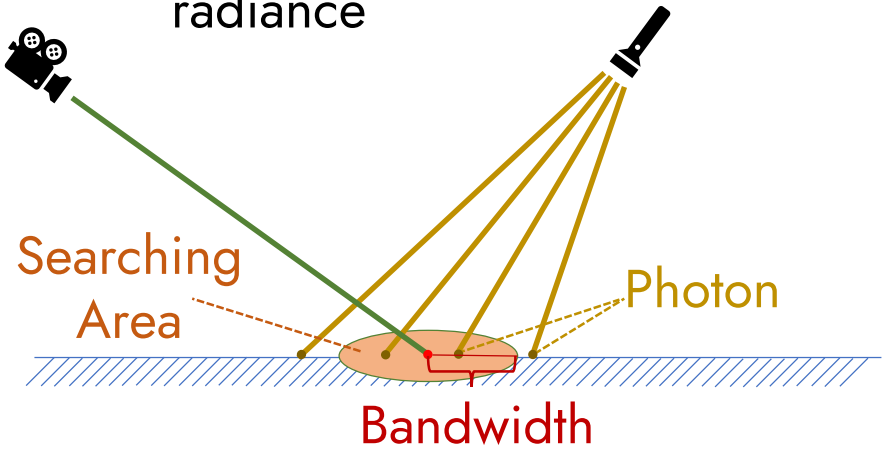
SIGGRAPH ASIA 2020 VIRTUAL

Outline

- Background
- Challenge
- Key Idea
- Algorithm
- Results
- Limitation
- Summary

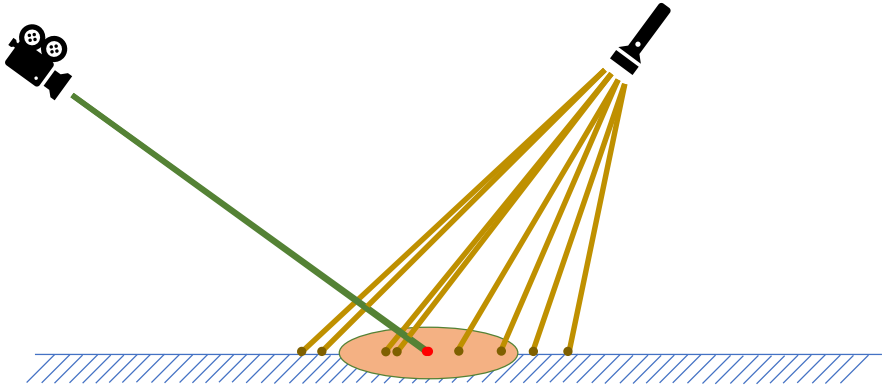
Background

- Photon Mapping [Jensen 1996] can solve S-D-S paths:
  - Light sources emit photons
  - Collect photons to estimate radiance



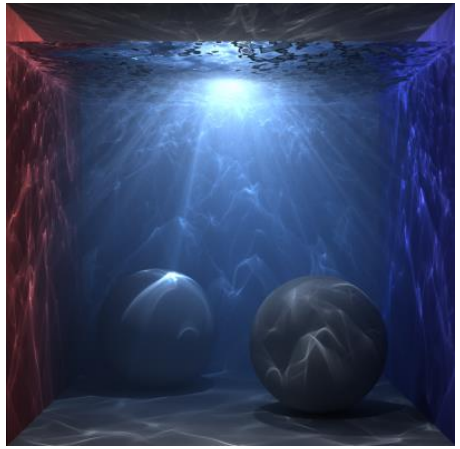
Background

- SPPM [Hachisuka and Jensen 2009] converges to the correct pixel measurement:
  - Multiple iterations
  - Bandwidth converges to infinitesimal



SIGGRAPH ASIA 2020 VIRTUAL

Background



Progressive Photon Mapping: A Probabilistic Approach

[Knaus and Zwicker 2011]

Adaptive Progressive Photon Mapping

[Kaplanyan and Dachsbacher 2013]

Deep Kernel Density Estimation for Photon Mapping

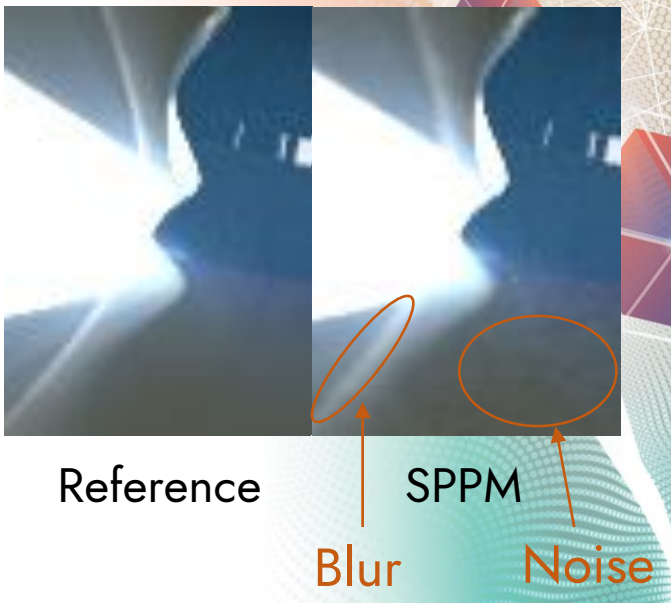
[Zhu et al. 2020]

SIGGRAPH ASIA 2020 VIRTUAL

Challenge

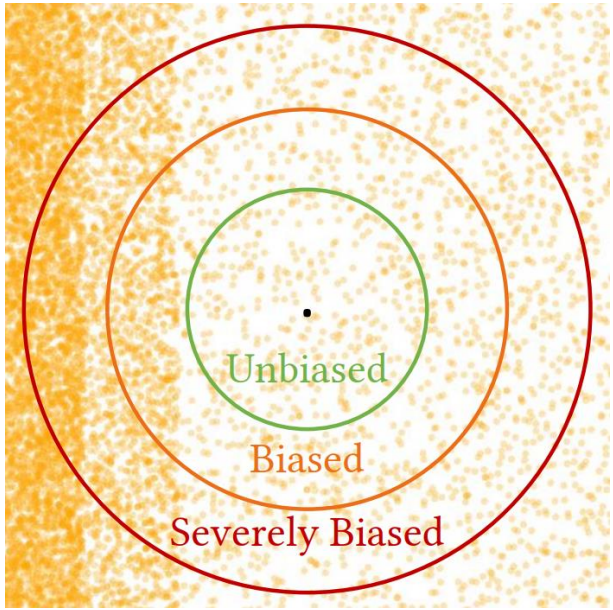
Challenge

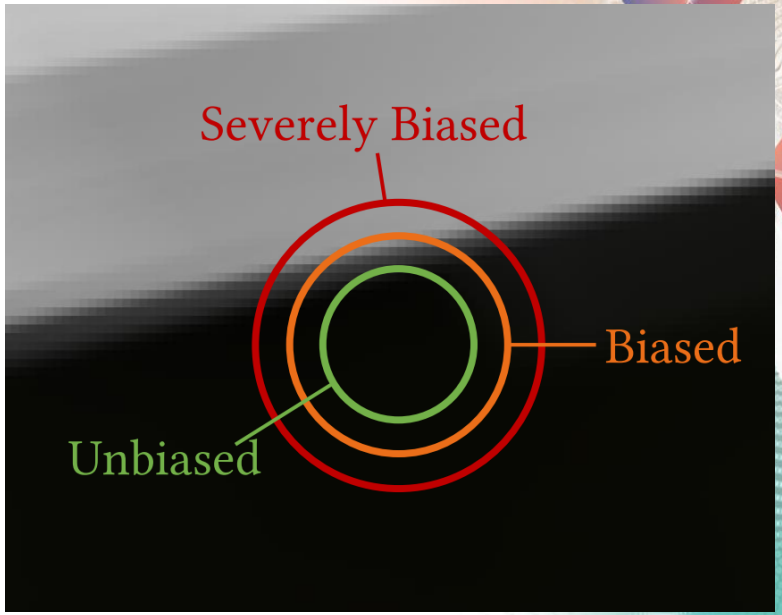
- Challenge: Blur and Noise
  - Blur:
    - Caused by bias
    - Bandwidth too large
  - Noise:
    - Caused by variance
    - Bandwidth too small
- Difficult to eliminate them at the same time



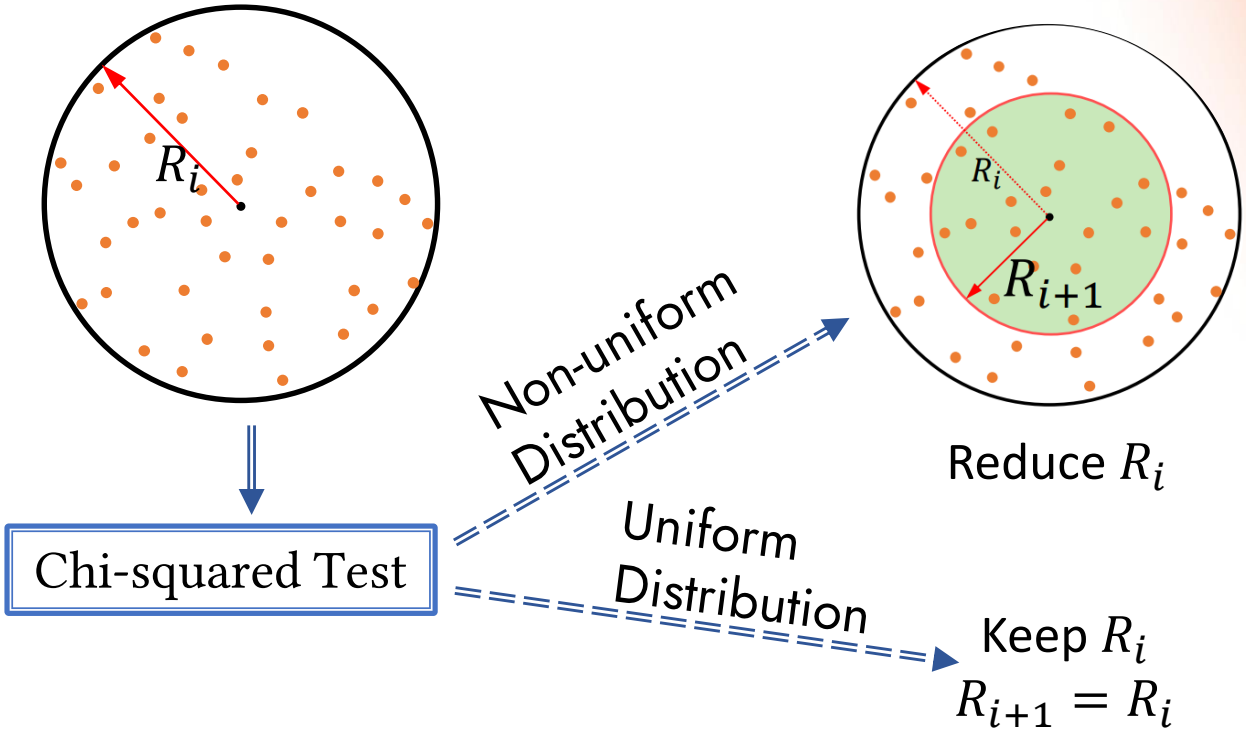
Key Idea: Benefits of Uniform Distribution

- Uniformly distributed photons have advantages
- Unnecessary to use a smaller bandwidth



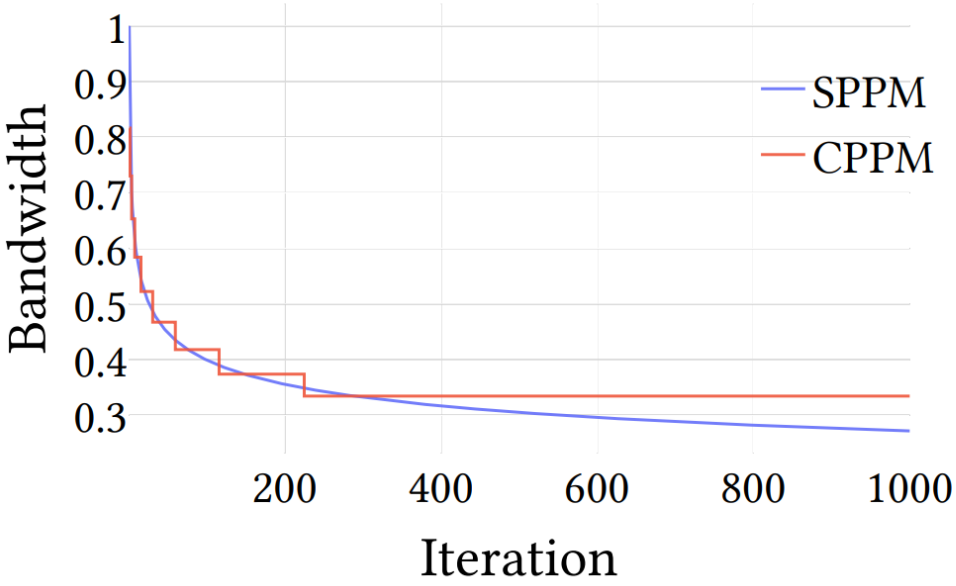


Key Idea: Chi-squared Test on Photons

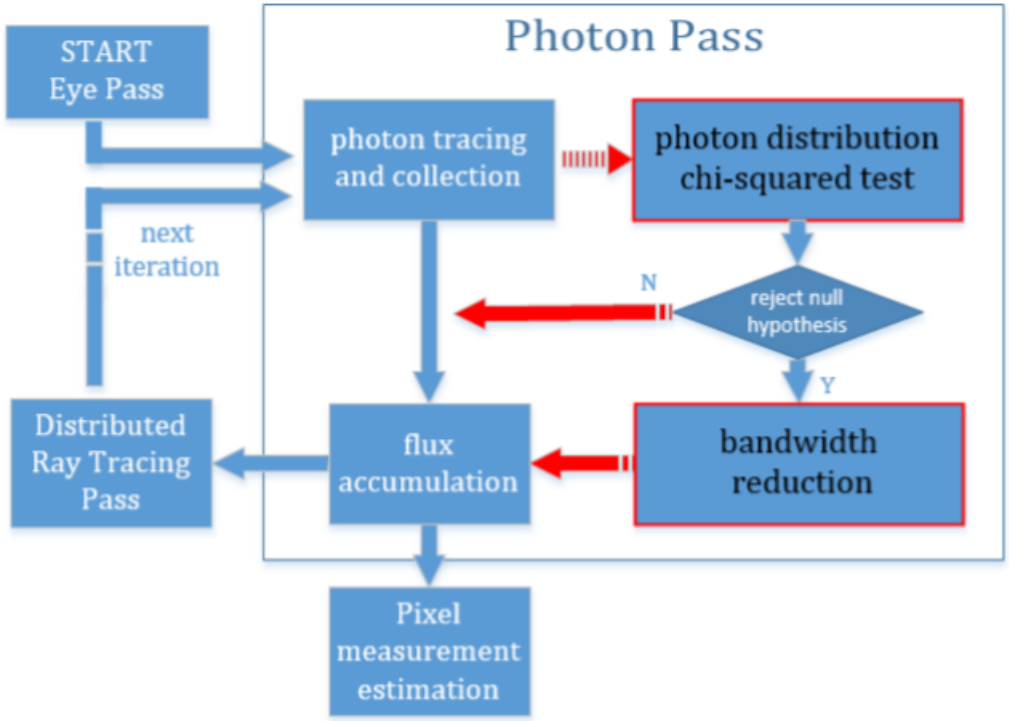


Key Idea: Bandwidth Reduction Scheme

- A novel bandwidth reduction scheme to work with chi-squared test

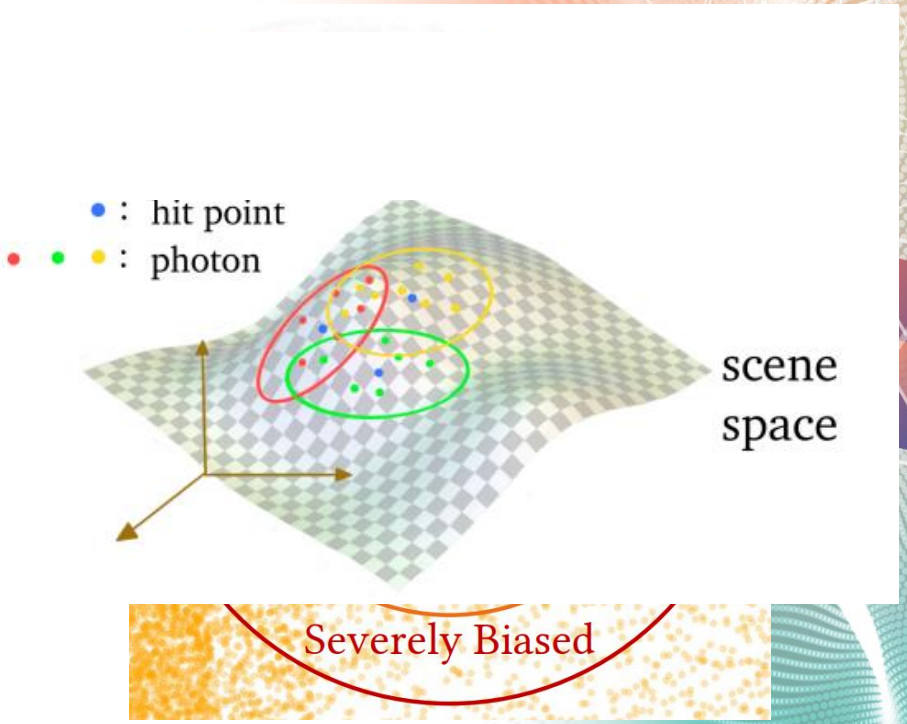


Algorithm: Pipeline



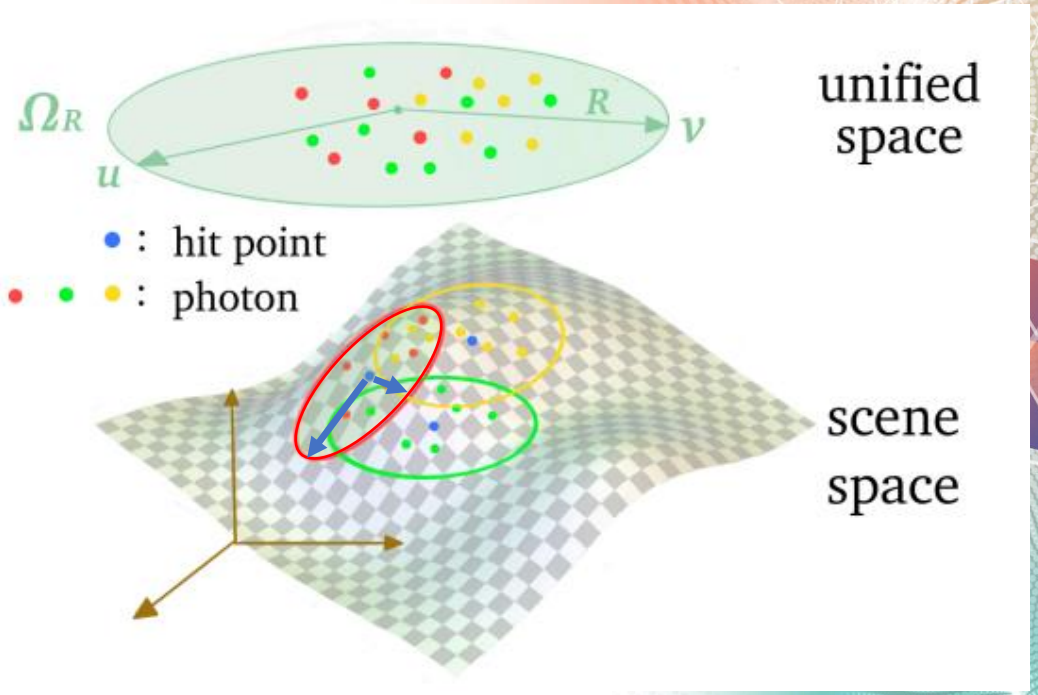
Algorithm: Theoretical Foundation

- How to define a uniform distribution on multiple searching areas?
- A unified space can be a solution



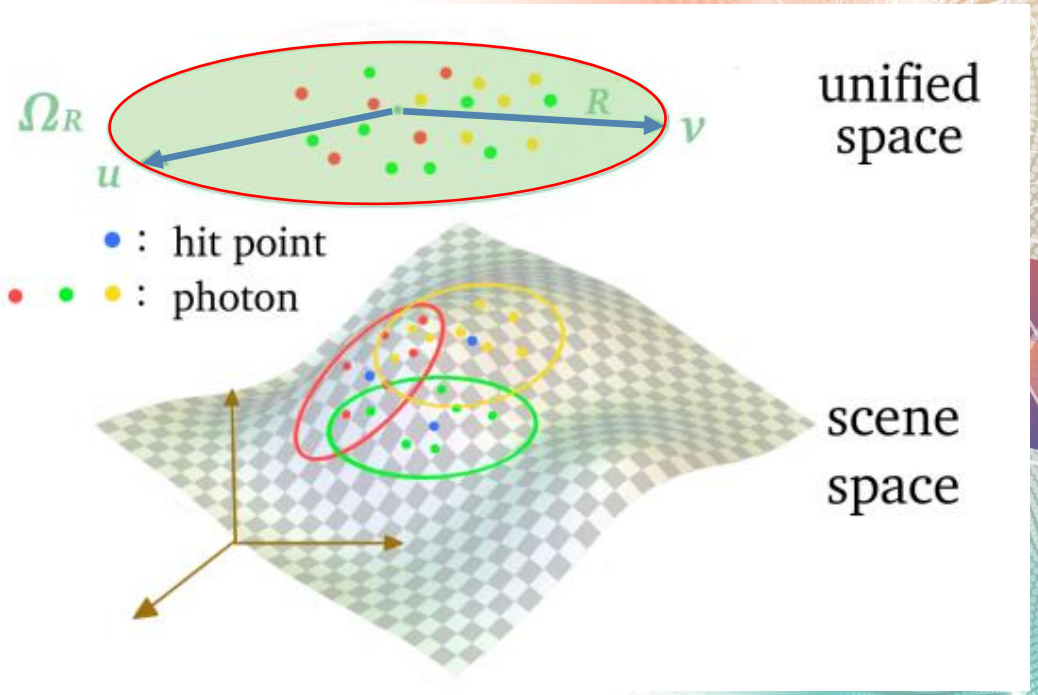
Algorithm: Theoretical Foundation

- How to define a uniform distribution on multiple searching areas?
- A unified space can be a solution
- Align the searching areas



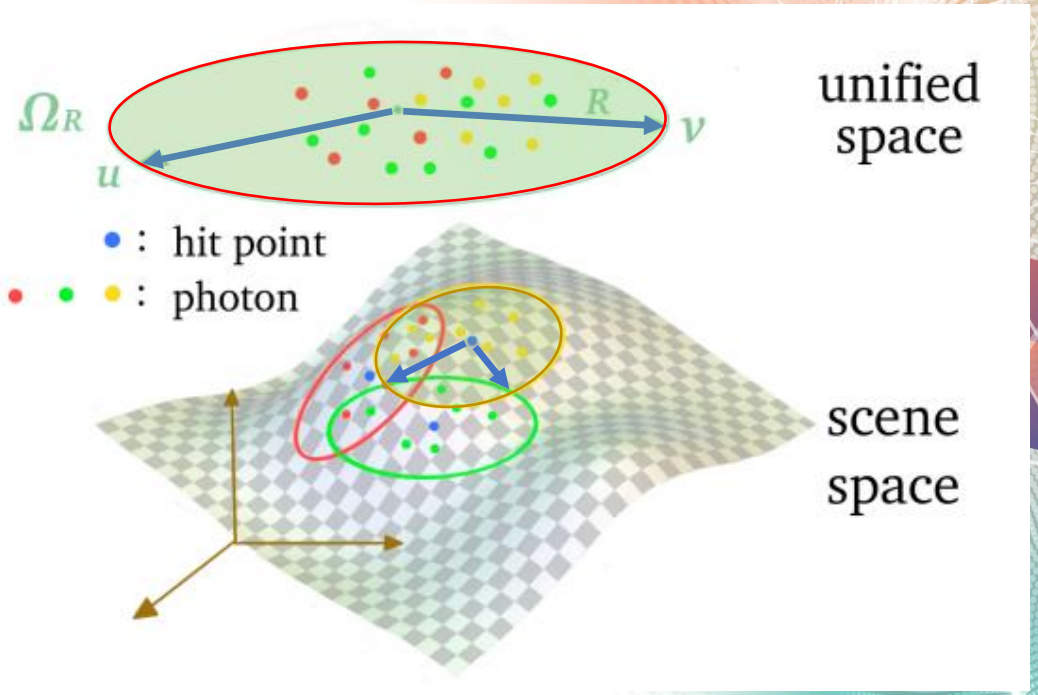
Algorithm: Theoretical Foundation

- How to define a uniform distribution on multiple searching areas?
- A unified space can be a solution
- Align the searching areas



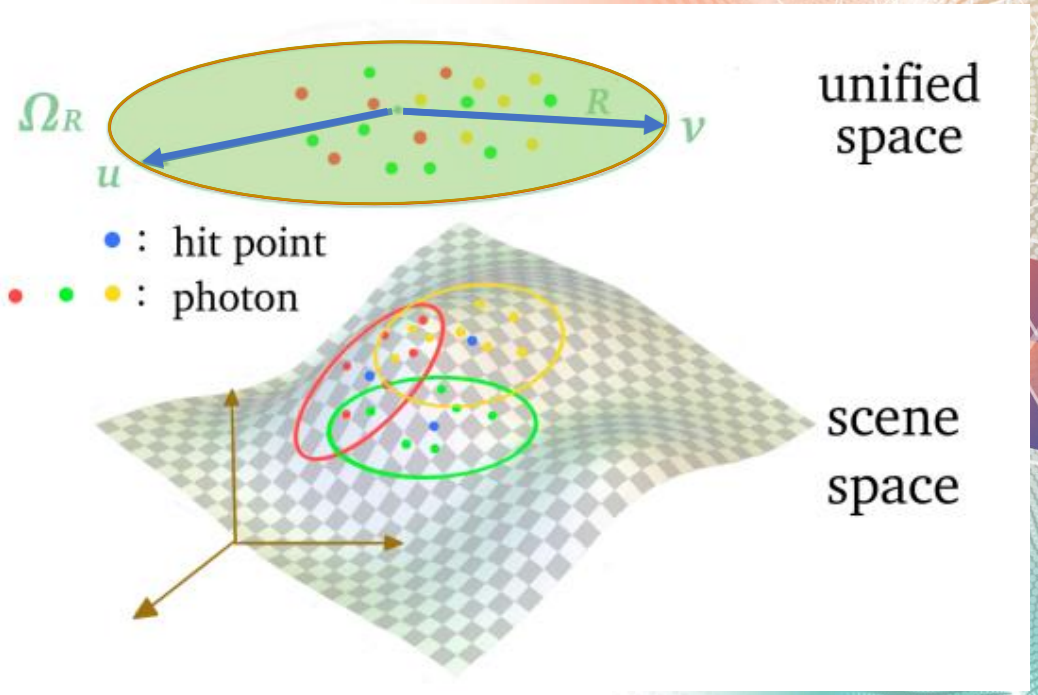
Algorithm: Theoretical Foundation

- How to define a uniform distribution on multiple searching areas?
- A unified space can be a solution
- Align the searching areas



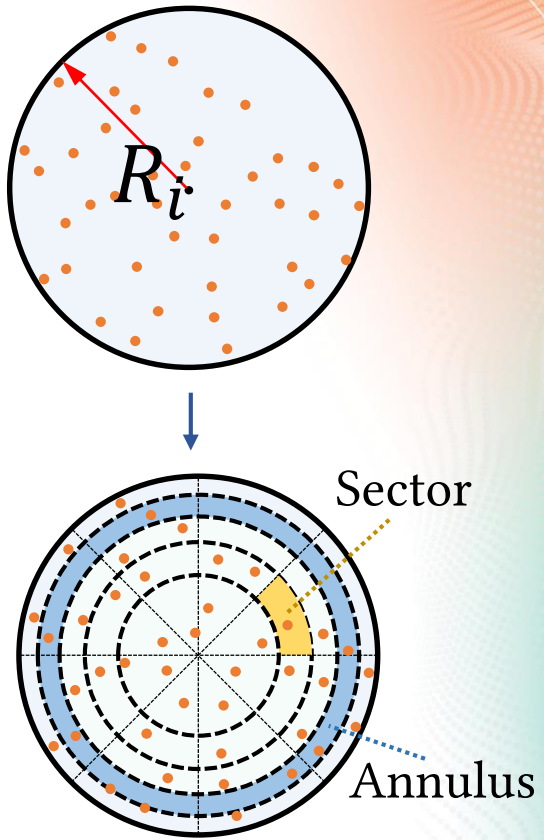
Algorithm: Theoretical Foundation

- How to define a uniform distribution on multiple searching areas?
- A unified space can be a solution
- Align the searching areas



Algorithm: Chi-squared Test on Photons

- Partition the disc
- Count photons in sectors
- Calculate chi-squared statistic to identify uniform distribution



Algorithm: Conditional Bandwidth Reduction

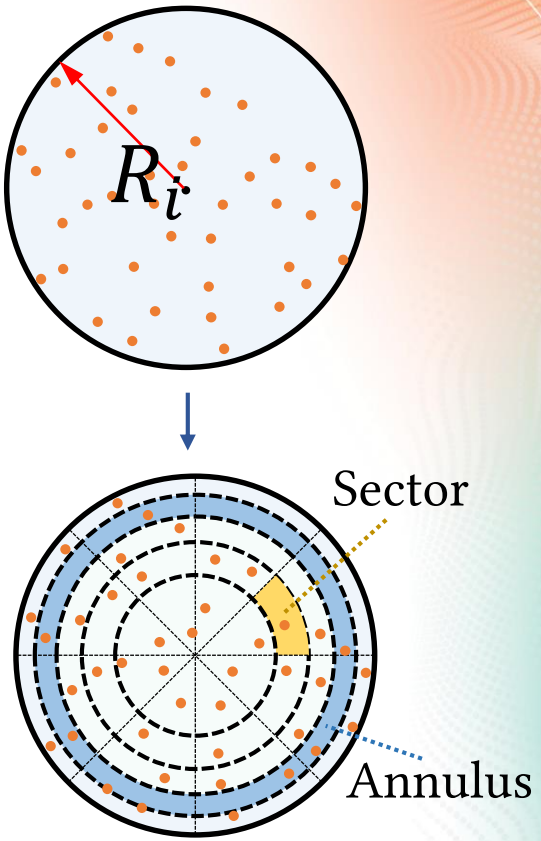
Photons

Uniformly distributed

$R_{i+1} = R_i$

Not uniformly distributed

$R_{i+1} < R_i$



Algorithm: Conditional Bandwidth Reduction

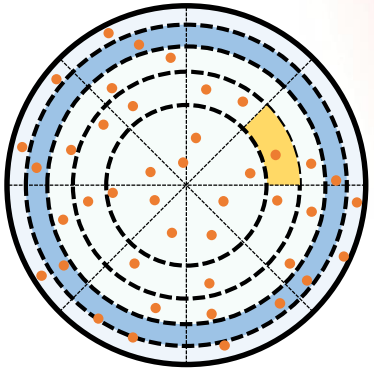
Photons

Uniformly distributed

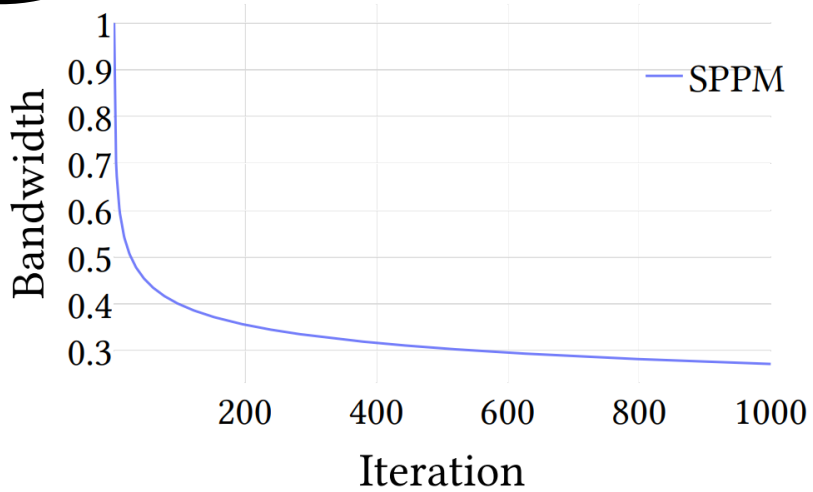
$R_{i+1} = R_i$

Not uniformly distributed

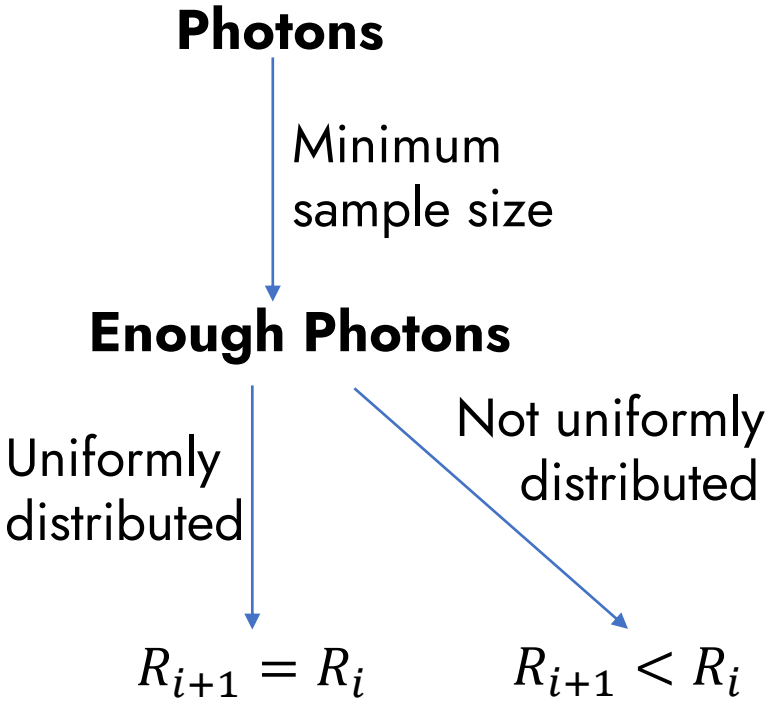
$R_{i+1} < R_i$

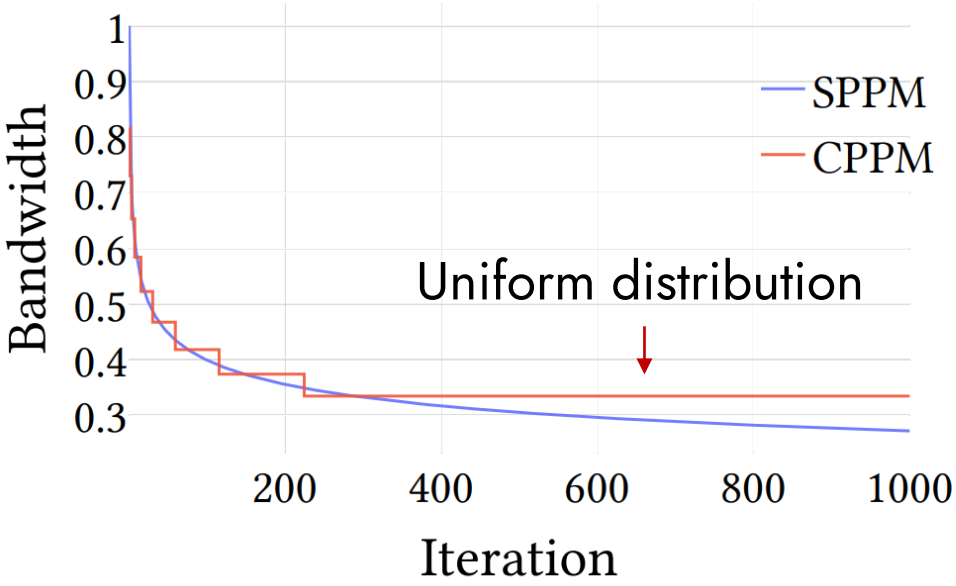


- How to get enough samples for the chi-squared test?

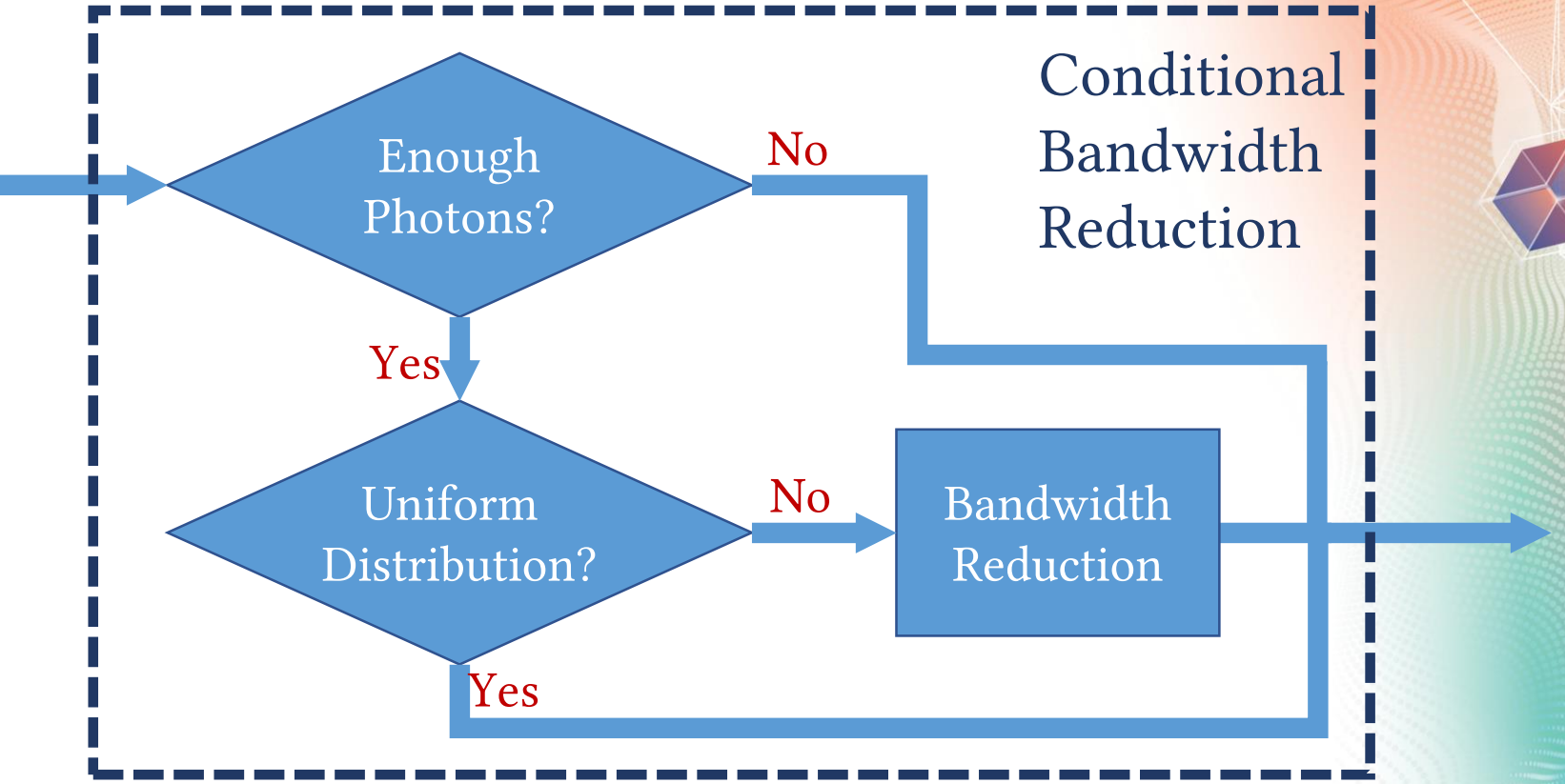


Algorithm: Bandwidth Reduction Scheme





CPPM Pipeline



Algorithm: Convergence

- Worst case:

$$R_N = O\left(N^{-\frac{1}{2}\log_{\frac{\beta}{k}}\frac{1}{k}}\right)$$

  - equivalent to SPPM
- Best case:

  Bias $= 0$

  Variance $= O(N^{-1})$

Exceptional Cases of the Chi-squared Test

- The chi-squared test may get wrong results:

1. Reject a uniform distribution

   Acceptable 

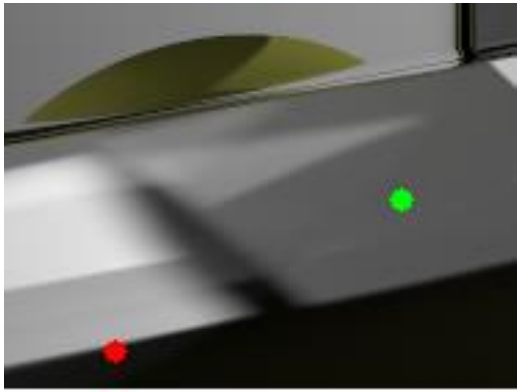
2. Not reject a non-uniform distribution

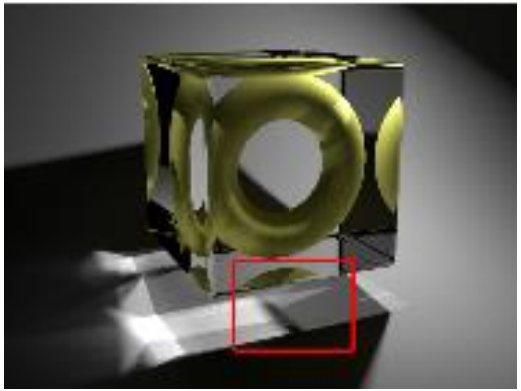
   Critical 

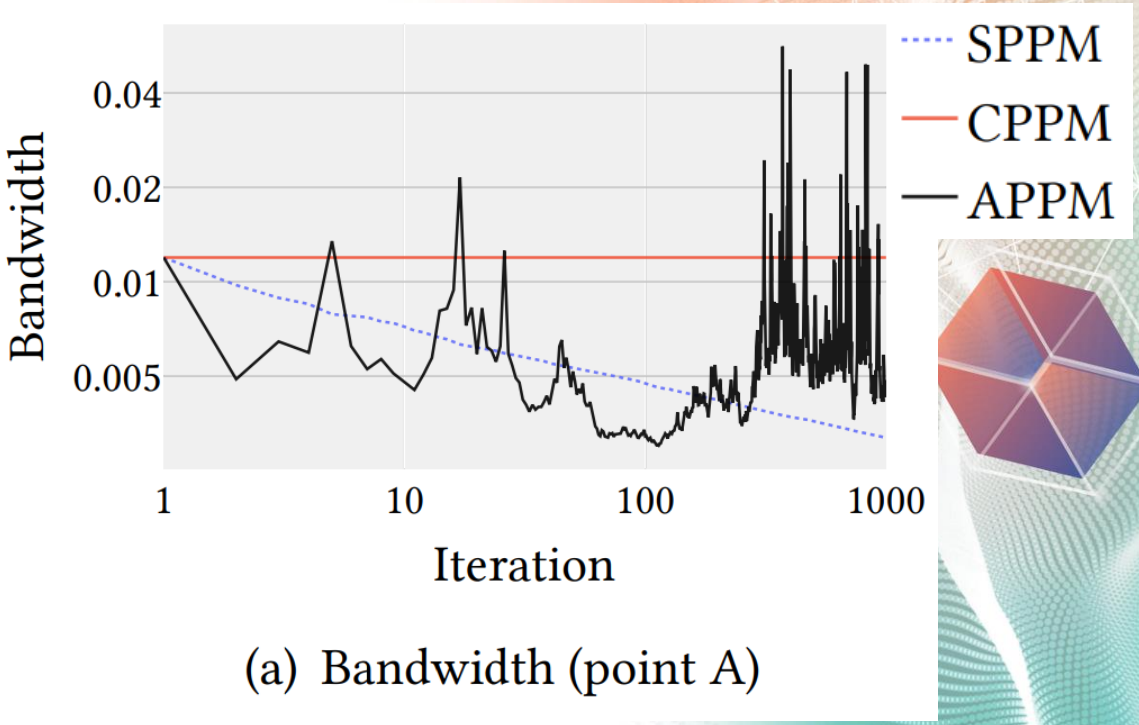
   Solution: keep testing 

SIGGRAPH ASIA 2020 VIRTUAL

Results

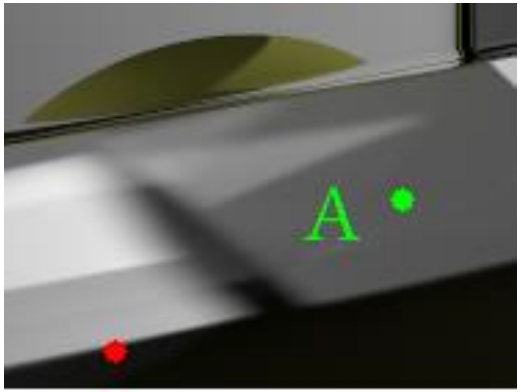


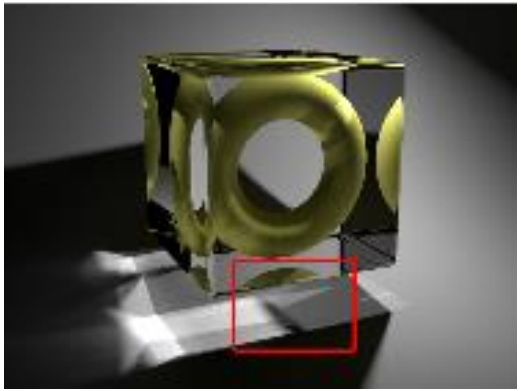


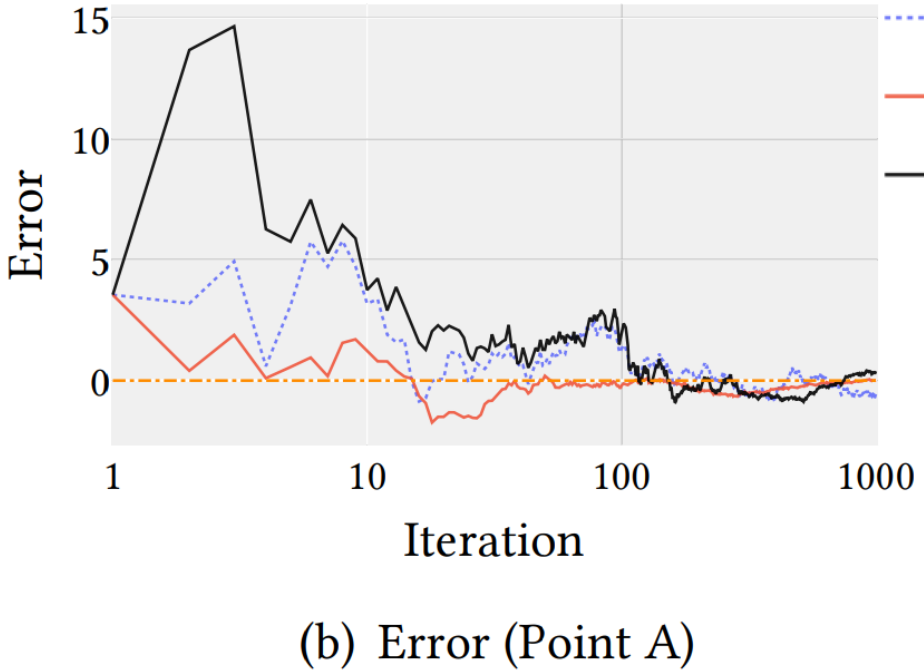


(a) Bandwidth (point A)

Results

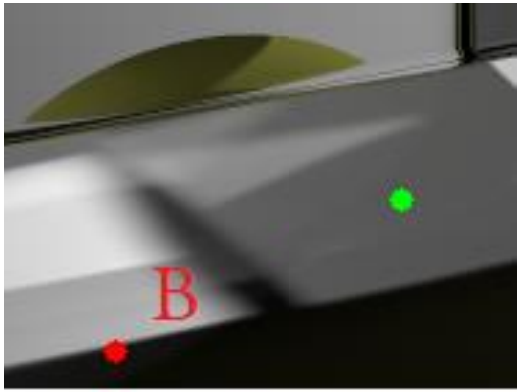


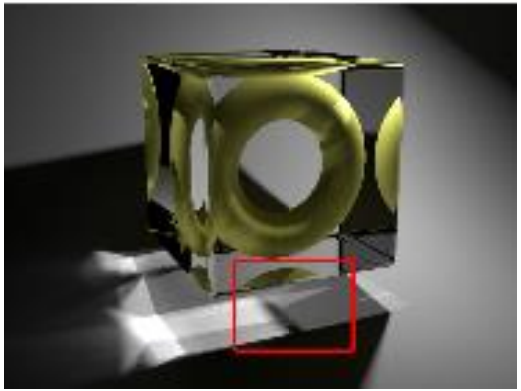




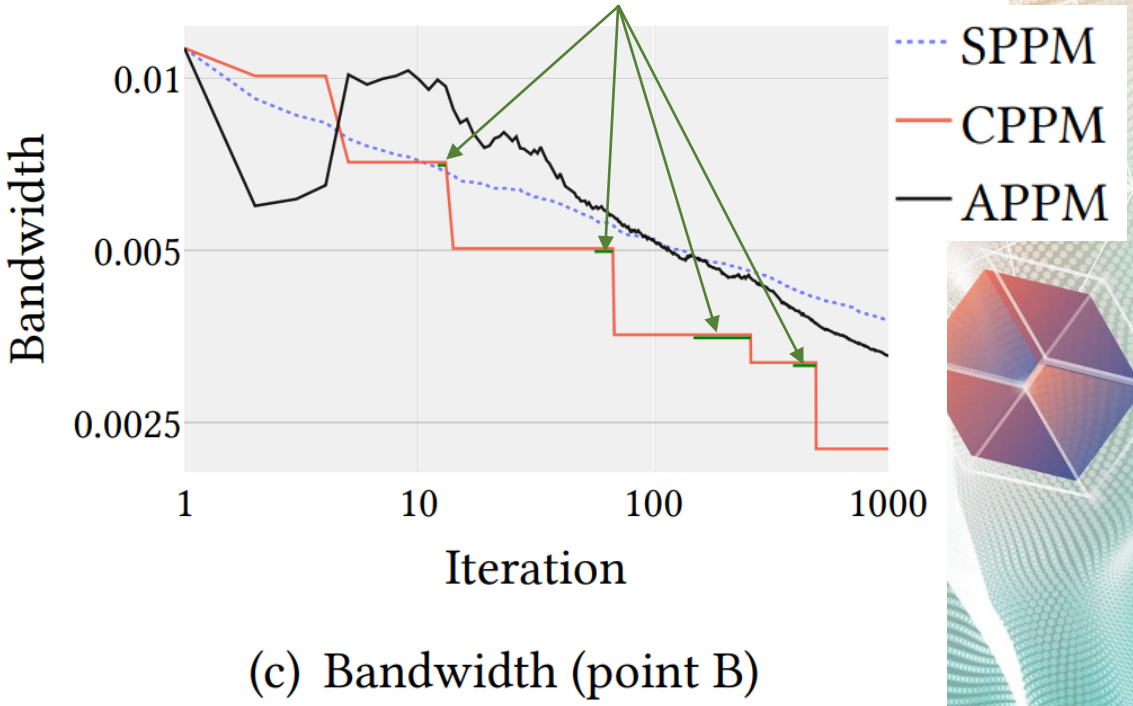
(b) Error (Point A)

Results



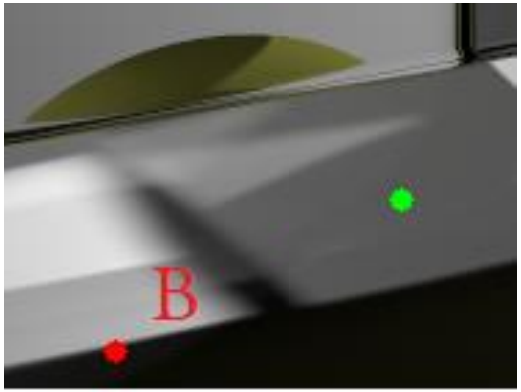


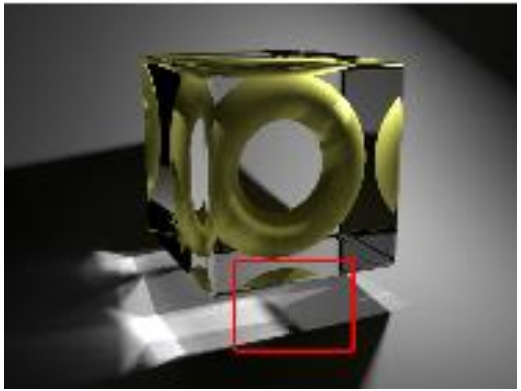
Chi-squared test fails to reject a non-uniform distribution

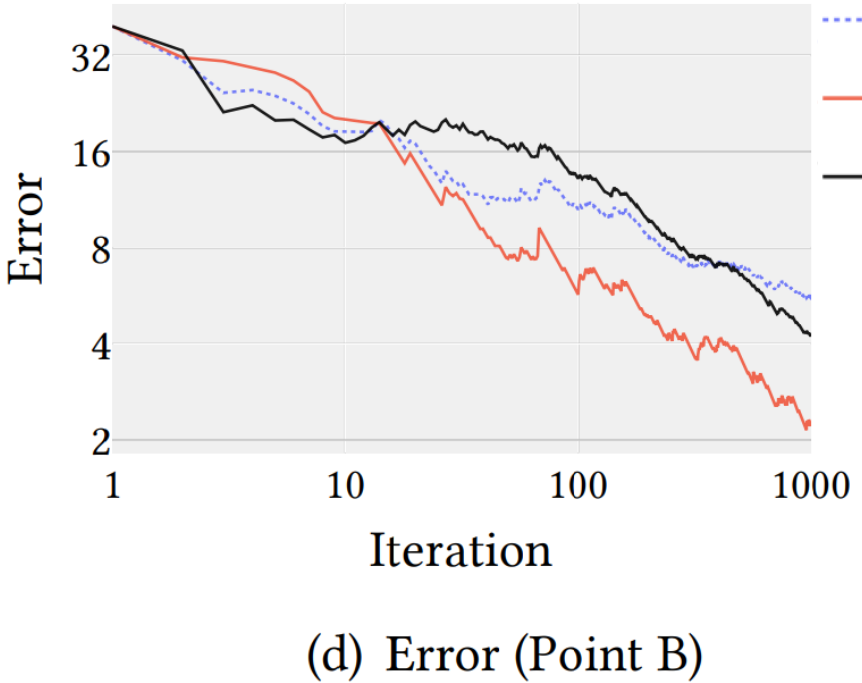


(c) Bandwidth (point B)

Results







(d) Error (Point B)

SIGGRAPH ASIA 2020 VIRTUAL

Results

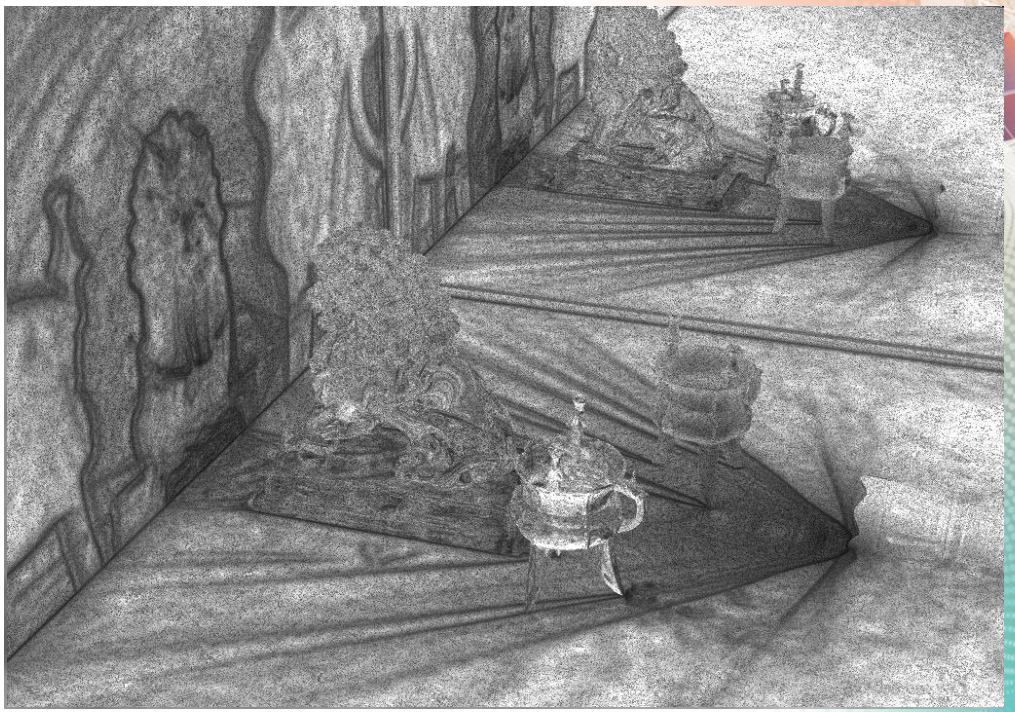
- How many pixels can find a uniform distribution?

Results

- How many pixels can find a uniform distribution?
- Bandwidth of SPPM:

Results

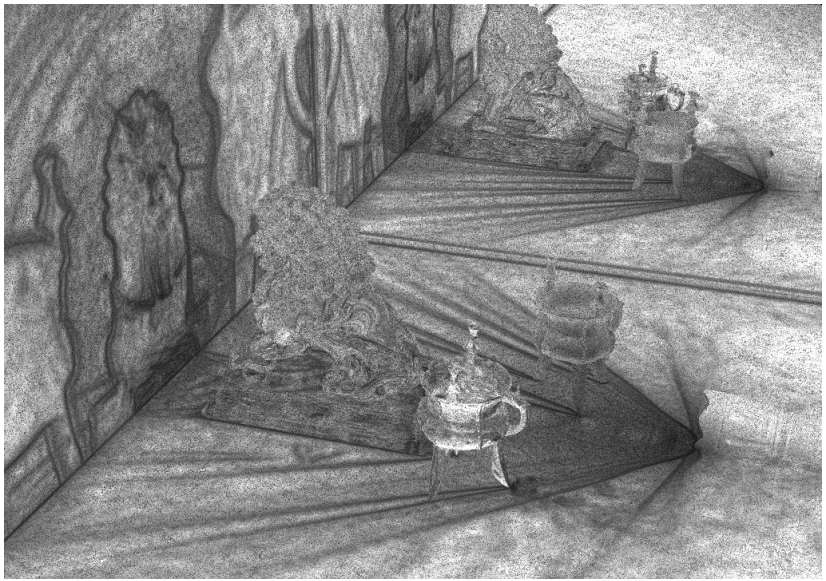
- How many pixels can find a uniform distribution?
- Bandwidth of CPPM:



Results

- Most pixels can find a uniform distribution

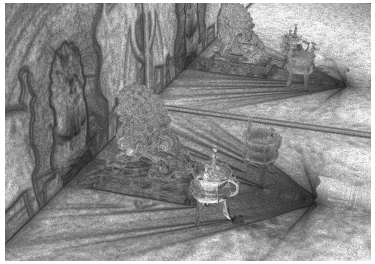
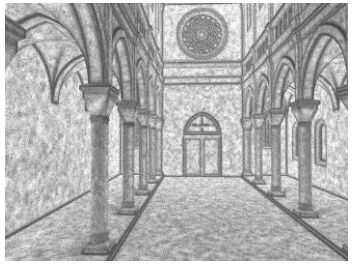
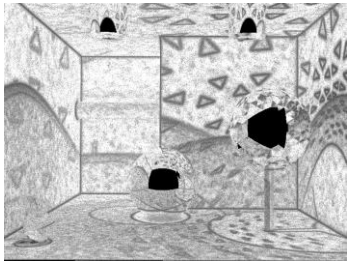
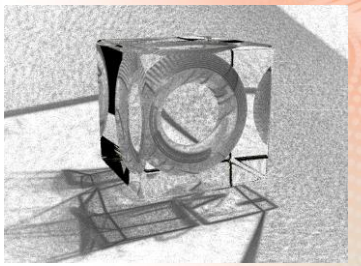
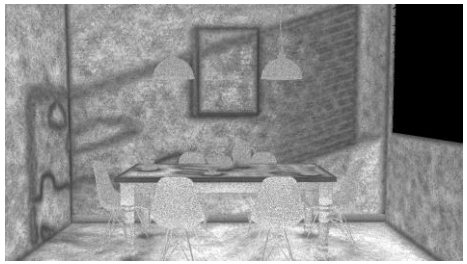
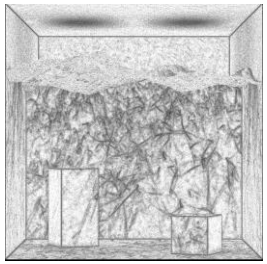
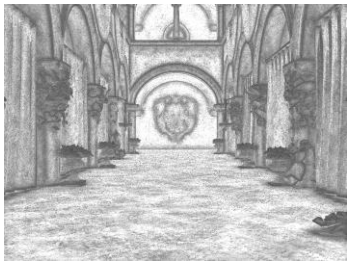
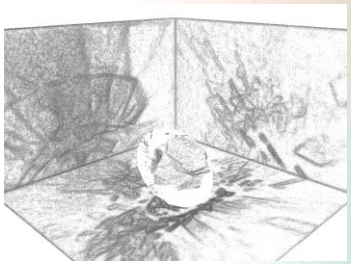
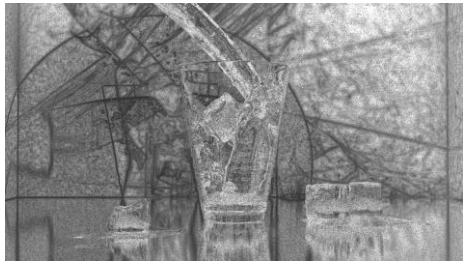
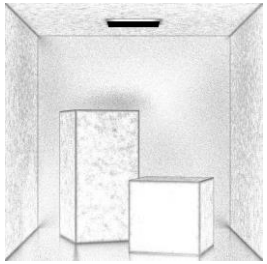
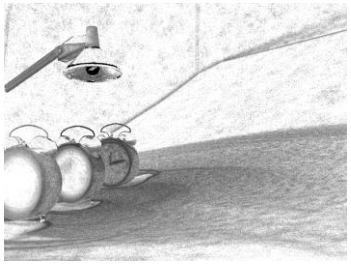
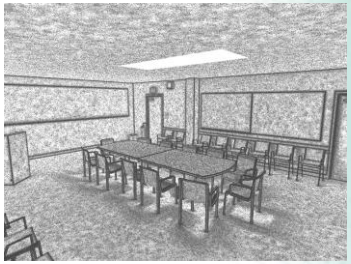
CPPM



SPPM

SIGGRAPH ASIA 2020 VIRTUAL

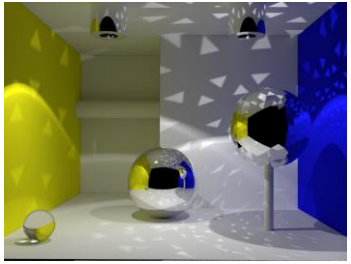
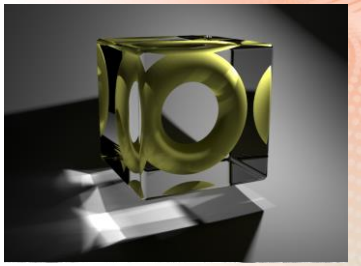
Results

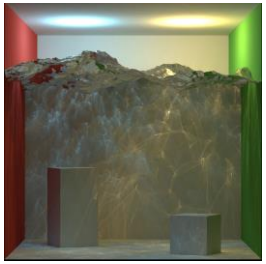













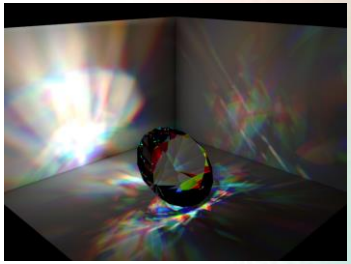
SIGGRAPH ASIA 2020 VIRTUAL

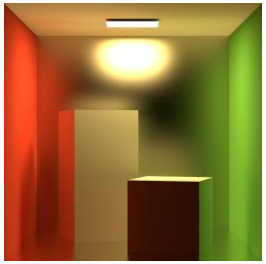
Results



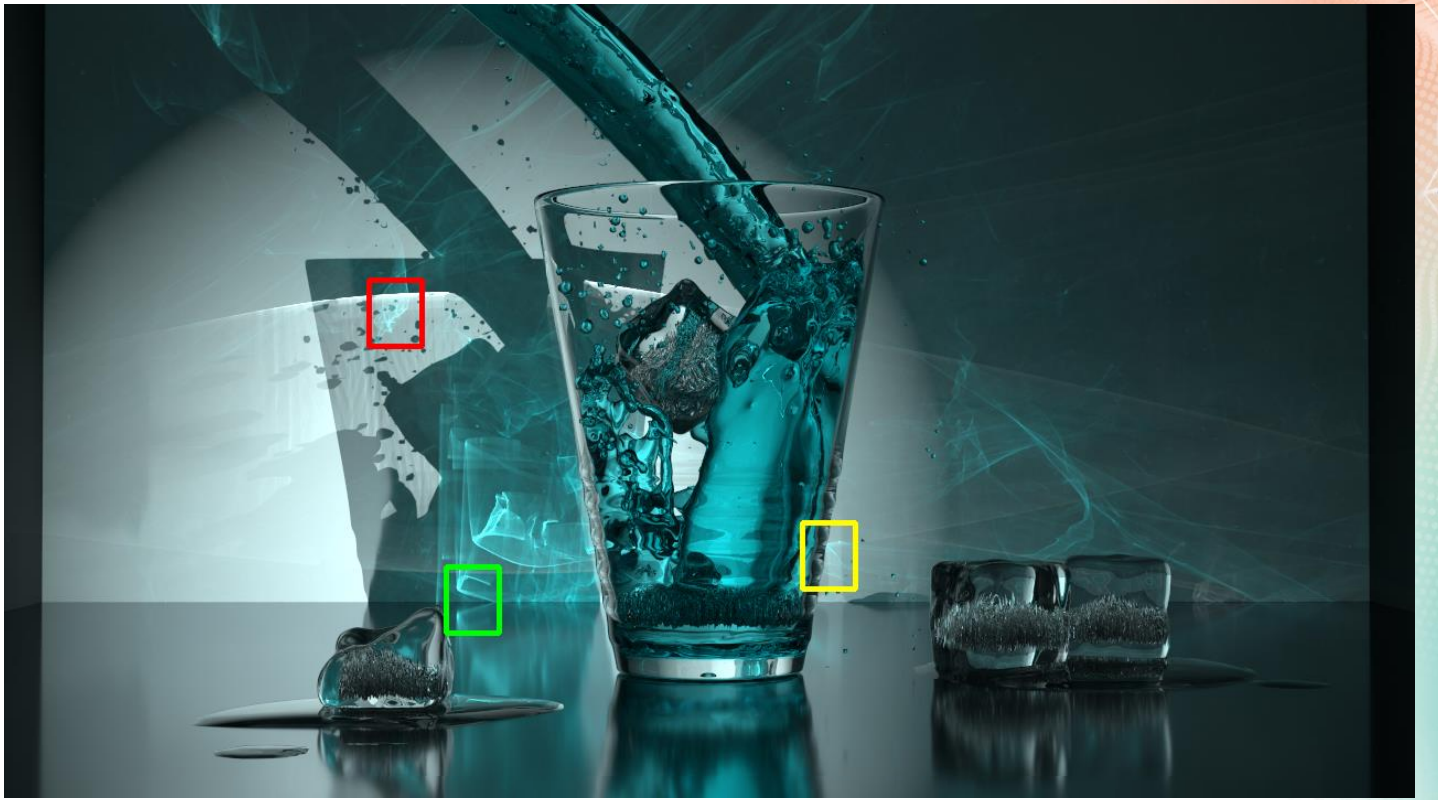






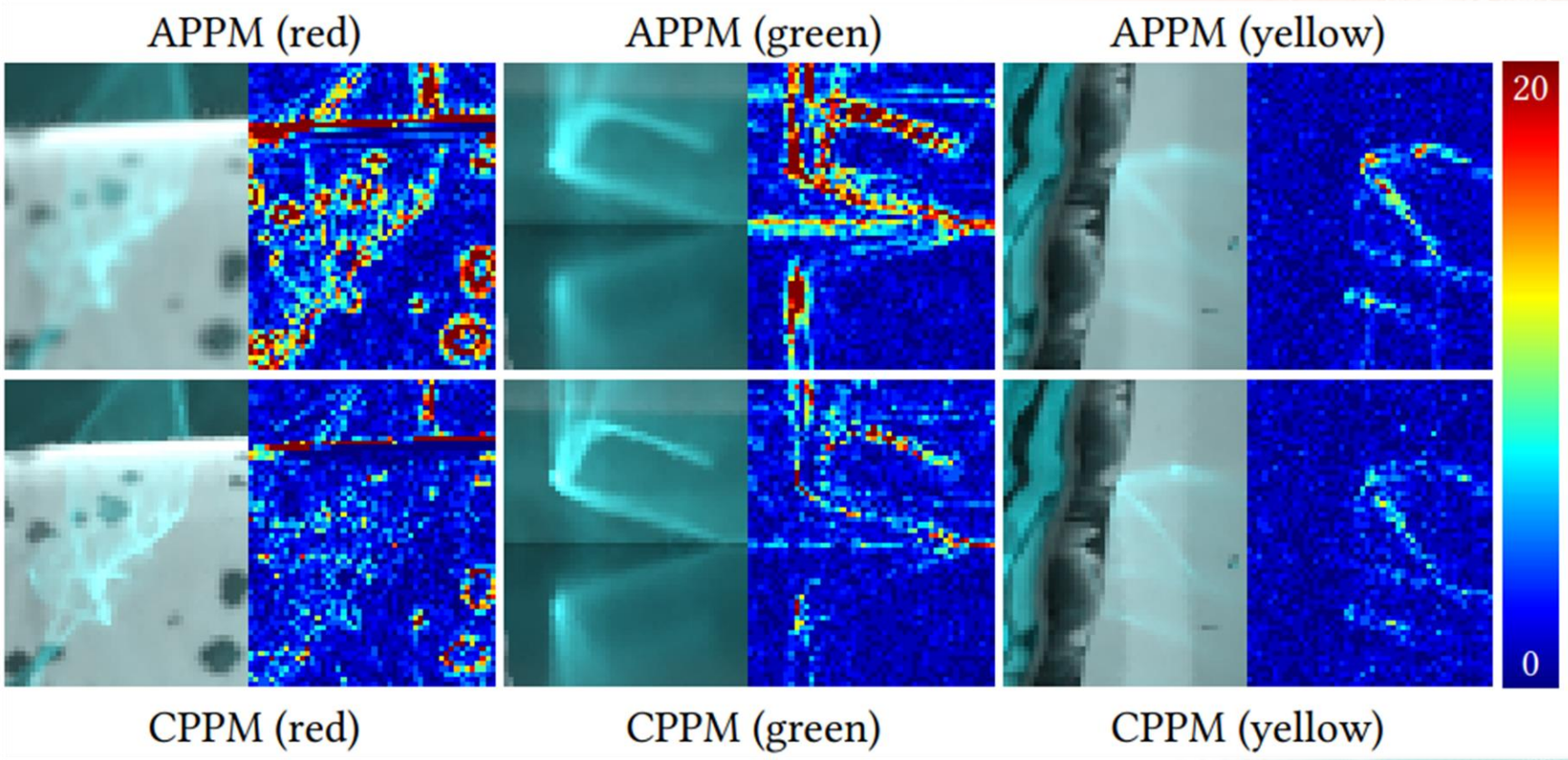
SIGGRAPH ASIA 2020 VIRTUAL

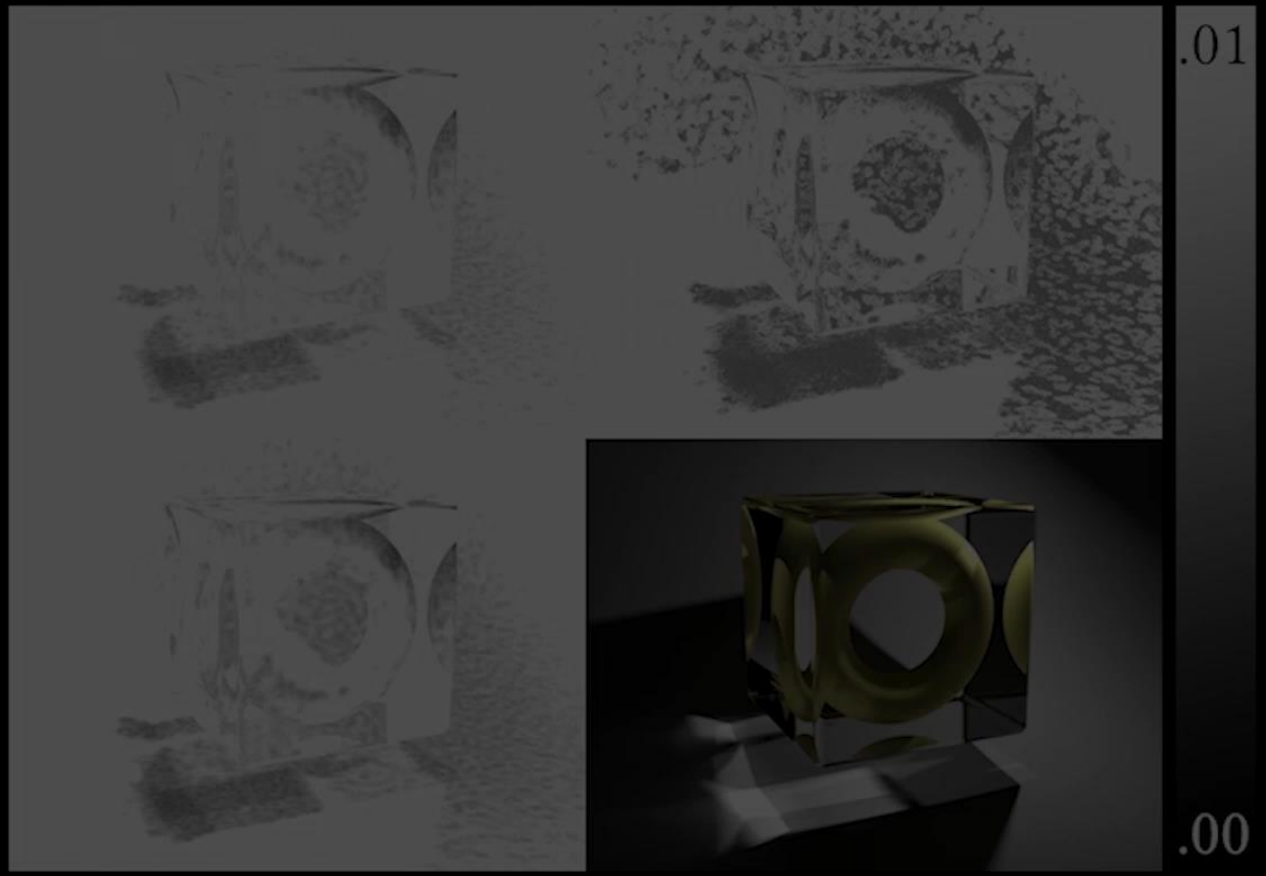
Results



SIGGRAPH ASIA 2020 VIRTUAL

Results

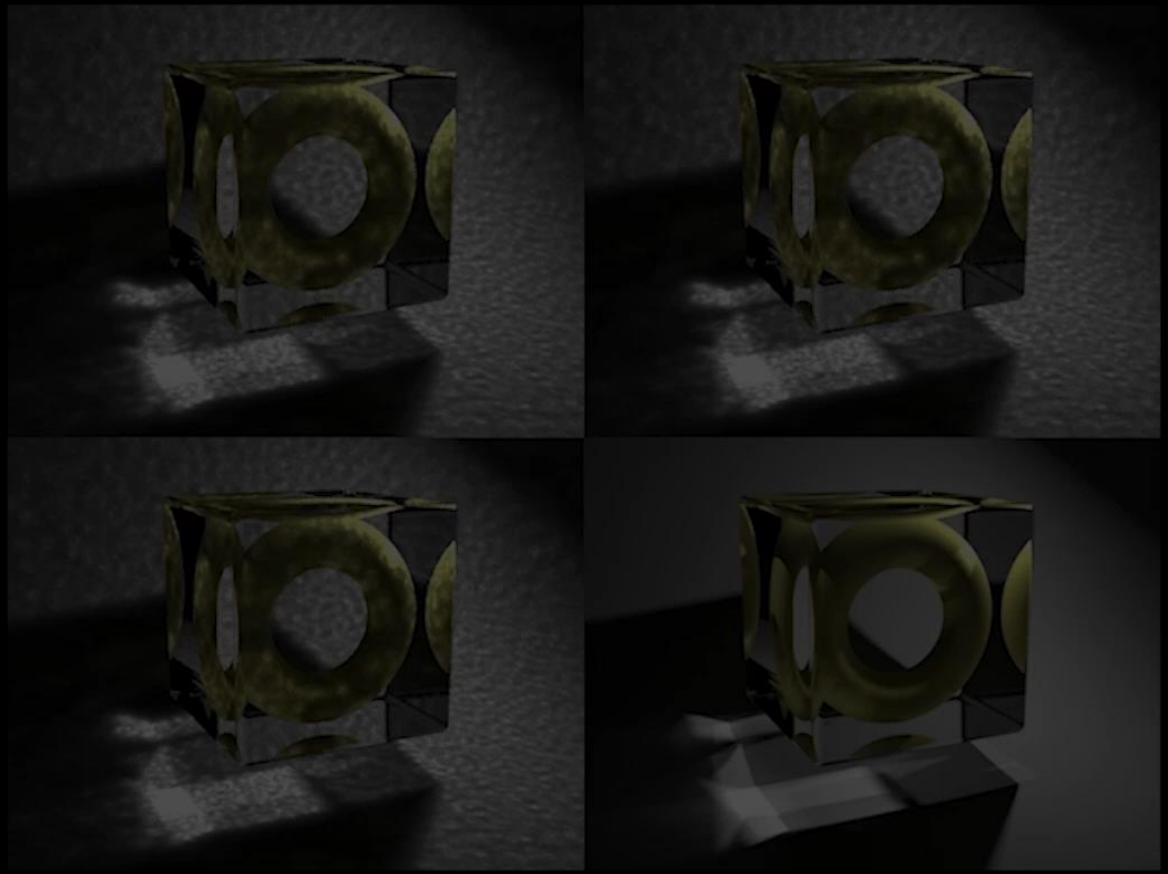



CPPM
Torus Bandwidth Visualization
APPM
.01
.00
SPPM
Iteration: 1
Photon: 0.1M
Reference

CPPM

Torus

APPM

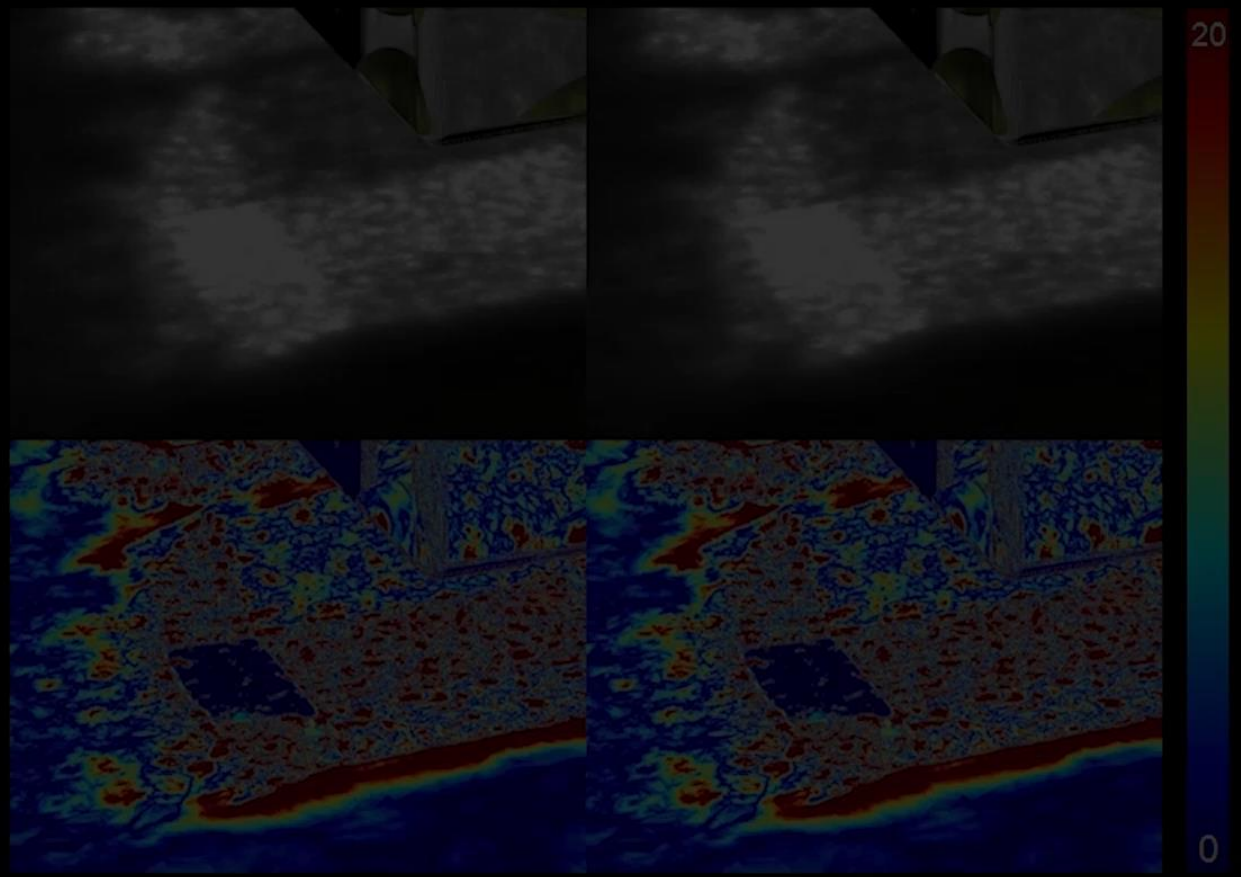


SPPM

Iteration: 1

Photon: 0.1M

Reference


CPPM
Torus
SPPM
20
0
Iteration: 1
Photon: 0.1M

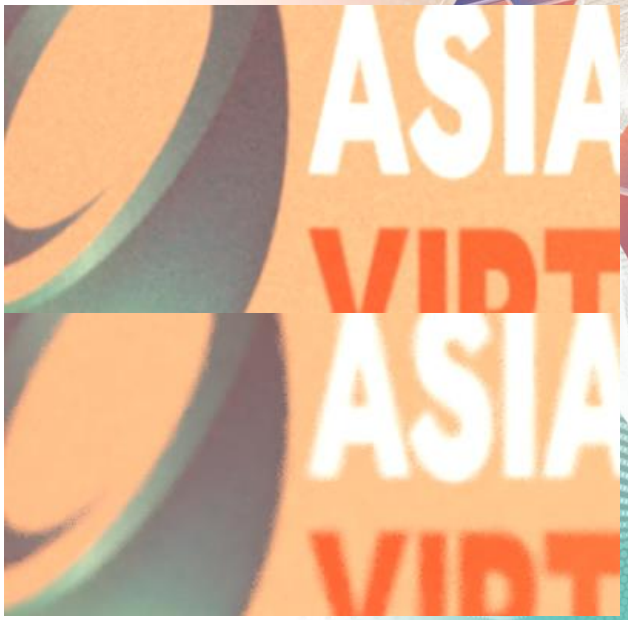
Limitation

- Uniformly distributed photons are not always good
- If the contribution of the photons depends on their location

Textured Spotlight:

SPPM

CPPM



Limitation

- If the contribution of the photons is related to their location
- Emit the photons proportional to the luminance:

SPPM

CPPM

CPPM+

Reference

SIGGRAPH ASIA 2020 VIRTUAL

Summary

Contributions:

+ Unnecessary to converge the bandwidth to infinitesimal
+ Reveal the benefits of uniformly distributed photons
+ Introduce the chi-squared test to check the photons
+ Propose a pipeline to robustly find a desired bandwidth

Future Work:

* Take contribution of photons into account
* Make bandwidth expansion possible
* Combine APPM and CPPM in a hybrid manner
* Integrate CPPM with other sampling-based techniques

Q & A

Contact: zehui@pku.edu.cn
Project Page: https://bactlink.github.io/CPPM

Thank you!

Contact: zehui@pku.edu.cn
Project Page: https://bactlink.github.io/CPPM