WWW.IARIA.ORG

PANEL MMEDIA
Challenges in Multimedia

Prof. Dr. Petre DINI

Concordia University, Canada

China Space Agency Center, China

IARIA Organization

petre@iaria.org

Panel

- **Moderator**
  Petre Dini, Concordia University, Canada || China Space Agency Center, China
- **Panelists**
  Jong Pil Park, Samsung Electronics, South Korea
  Jaeseon Song, Chungbuk National University, South Korea
  Petre Dini, Concordia University, Canada || China Space Agency Center, China

Topics

- **Jong Pil Park, Samsung Electronics, South Korea**
  - **> Realistic rendering and real-time rendering in embedded hardware**
- **Jaeseon Song, Chungbuk National University, South Korea**
  - **> Effect of depth information on illusion in daily life**
- **Petre Dini, Concordia University, Canada || China Space Agency Center, China**
  - **> Big Data, Small Data, Linked Data for information retrieval**

Big Data, Small Data, Linked Data

- **Challenges**
  - **Complexity**
  - **Distribution**
  - **Ownership**
  - **Big vs. Small Data**

- **Current approaches**
  - **Ontology**
  - **Semantic models**
  - **Social networks**
  - **Collaborative research**

Selective Ray Tracing Framework for Real-time Global Illumination on Mobile Platform

Jong Pil Park, Jung Bum Kim, Sang Jun Ahn
Samsung Electronics

1. Motivation

Global Illumination is expensive

- Global Illumination Methods (Ray Tracing, Radiosity, Light Transport)
  . Computational cost : **(Ray Number x Object Number^2 x Light Number)**
- Offline Rendering / Pseudo Realistic Method in PC GPU

Ray Tracing HW is expensive

- Dedicated HW for Ray tracing can accelerate the Global Illumination
  . Minimized size of HW is essential

Ray Tracing embedded HW very expensive

- Additional constraints in memory bandwidth, gate count and GPU performance

→ **Minimize Ray Tracing Region**

2. Restriction

HW Ray Tracing Pipeline is Fixed

- Ray Tile Structure
- Fixed Pipeline

Use Conventional GPU in Embedded Environment

- Cannot use GPGPU Tech.
- Conventional GPU Operation can be used only.

2. Restriction & Opportunity

HW Ray Tracing Pipeline is Fixed

→**Using Tile Structure**

Use Conventional GPU in Embedded Environment

→ **Parallel Processing of GPU and RT HW**

3. Selective Ray Tracing Framework

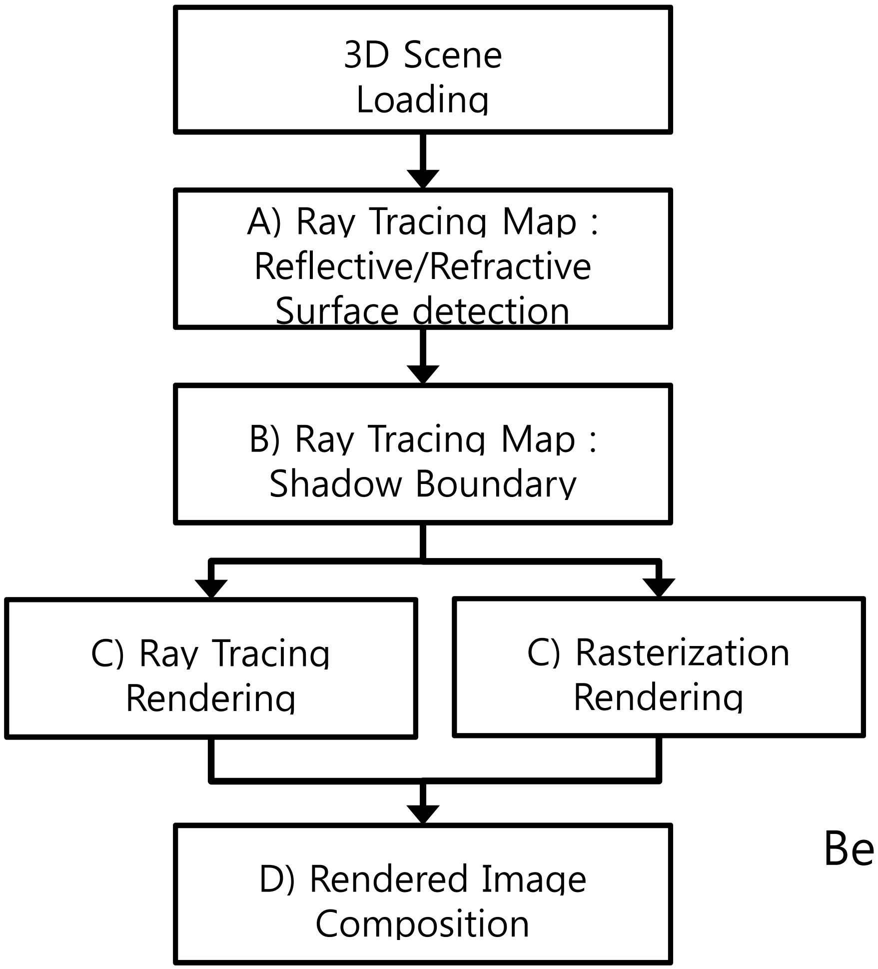


Fig.1 Selective ray tracing framework

Algorithm

A. **Ray Tracing Map generation** :
find the region of to be rendered by **reflective/ refractive surface** and set the region to be ray traced

B. **Ray Tracing Map generation :**
find **the shadow boundary region** and set the region to be ray traced

C. **Rendering of rasterization / ray tracing** due to calculated ray selection map
   a. Rasterization rendering use GPU operation
   b. Ray tracing rendering use dedicated HW rendering

D. **Rendered image composition using framebuffer**

Benefit

- **Reduction of ray tracing region**
- **Performance enhancement for the parallel Processing**

4. Boundary Test with Early Termination

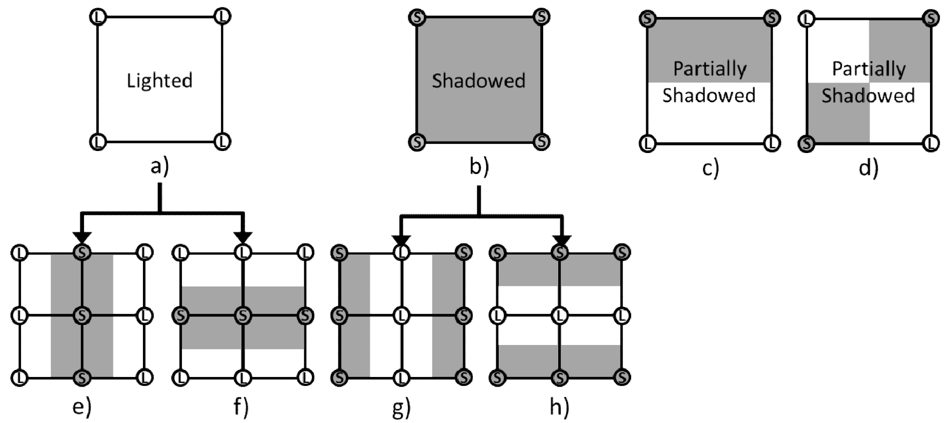


Fig.2 Algorithm diagram of shadow boundary detection

- Goal
  - Detect the shadow boundary region to be ray traced

- Algorithm
  1. Boundary test of the **tile at corner point**(same as Sen's approach[1])
  2. If we found current tile is tested as boundary, then **terminate (c, d)**
  3. In other case, **split the current tile into 4 subtile**
  4. Boundary test for the subtiles
     a. If current tile is lighted (a) and if any subtile is shadowed, then terminate (e,f)
     b. If current tile is shadowed(b) and if any subtile is lighted, then terminate (g,h)
  5. If not terminated, iteratively repeat 3, 4 until all the pixel is tested
  6. If all not terminated until 5, current pixel is regarded as "Not Boundary" region

[1] P. Sen, M. Cammarano, and P. Hanrahan, "Shadow silhouette maps," in ACM Transactions on Graphics (TOG), vol. 22, no. 3. ACM, 2003, pp. 521–526.

4. Boundary Test with Early Termination

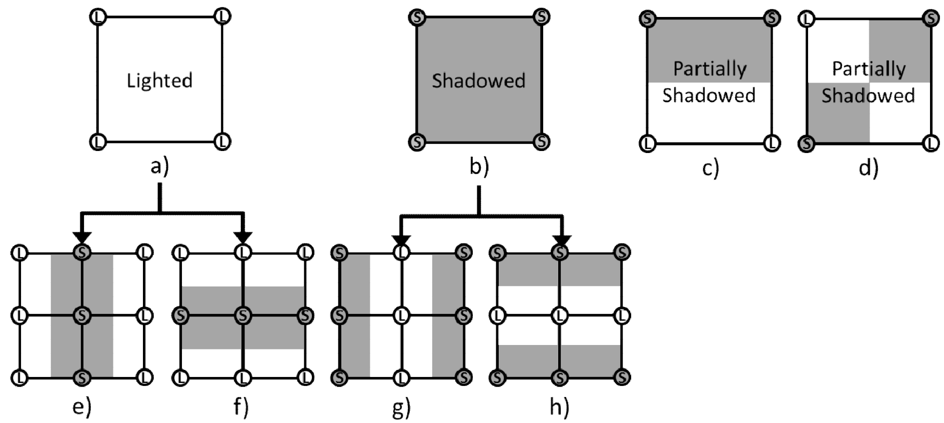


Fig.2 Algorithm diagram of shadow boundary detection

- Benefit
  - Performance :

[Brute force Test]

Cost = $8 * n^2 * k_1$

[Our Approach]

- **Min.**

Cost = $(n/m+1)^2 * k_2$

- **Max.**

Cost = $(n+1)^2 * k_2$

n : total Pixel number

m : tile pixel length

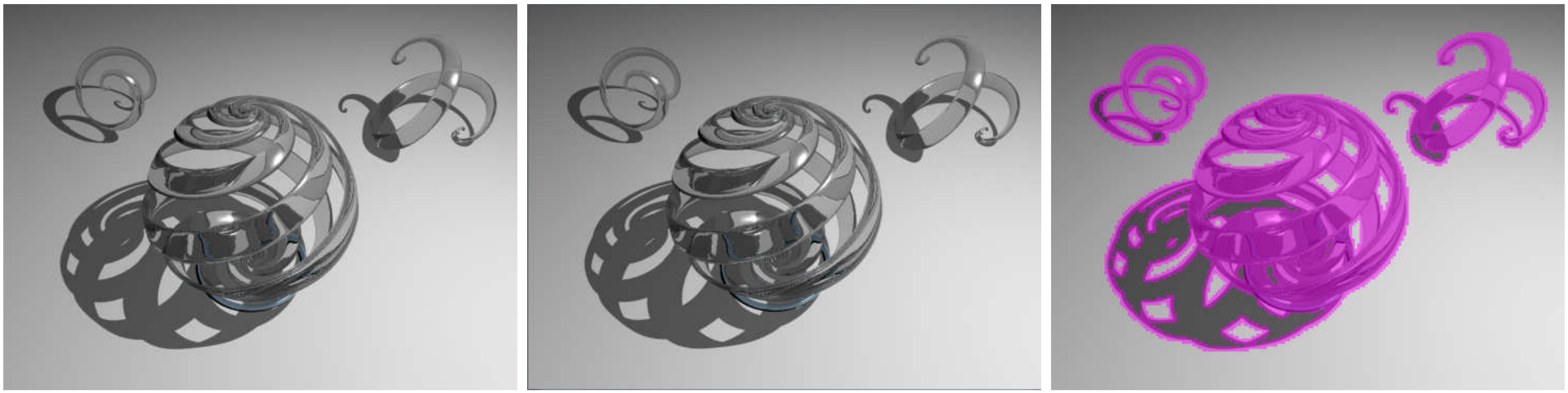
k_1 : computational cost for adjacent pixel test

k_2 : computational cost for shadow test of a single pixel

- Structure
  - **. Tile based Rendering**
  - **. Early Terminate**

[1] P. Sen, M. Cammarano, and P. Hanrahan, "Shadow silhouette maps," in ACM Transactions on Graphics (TOG), vol. 22, no. 3. ACM, 2003, pp. 521–526.

5. Selective Ray Tracing Results



a) Global Illumination b) Selective Ray Tracing c) The region which is ray traced in the selective ray tracing

Fig.3 Selective ray tracing rendering result on "*Almost Sphere*" scene

a) Global Illumination b) Selective Ray Tracing c) The region which is ray traced in the selective ray tracing

Fig.3 Selective ray tracing rendering result on "*Chess*" scene

5. Selective Ray Tracing Results

a) Global Illumination

b) Selective Ray Tracing

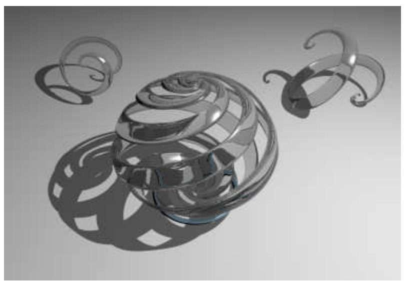
c) The region which is ray traced in the selective ray tracing

Fig.3 Selective ray tracing rendering result on "*Wedding Ring*" scene

6. Selective Ray Tracing Statistics

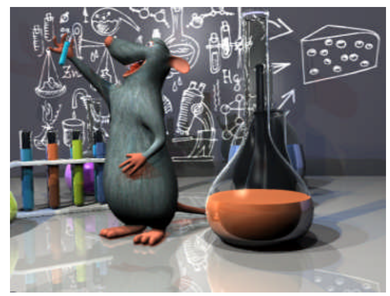
	Almost Sphere	*Amplifier*	*Chemical Lab.*	*Chess*	*Fiat*	*Stained Glass*	*Wedding Ring*
Number of primitives	53762	314582	98024	105766	52982	491961	115476
Number of rays in full ray tracing (A)	1442367	10317256	3369251	1306236	2876165	6057722	1594023
Number of rays in selective ray Tracing (B)	766527	8726526	2645273	466414	2003045	4473405	321115
Ratio B/A	0.531436867	0.845818501	0.785121975	0.357067176	0.696429099	0.738463238	0.201449414

Table 1. The number of rays in global illumination and hybrid rendering on the test scenes



Almost Sphere

Amplifier

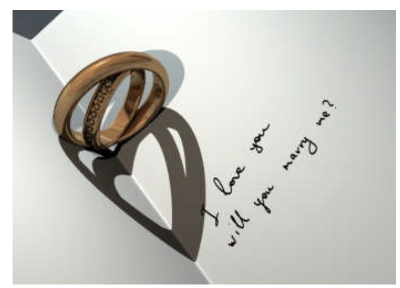


Chemical Lab.

Chess

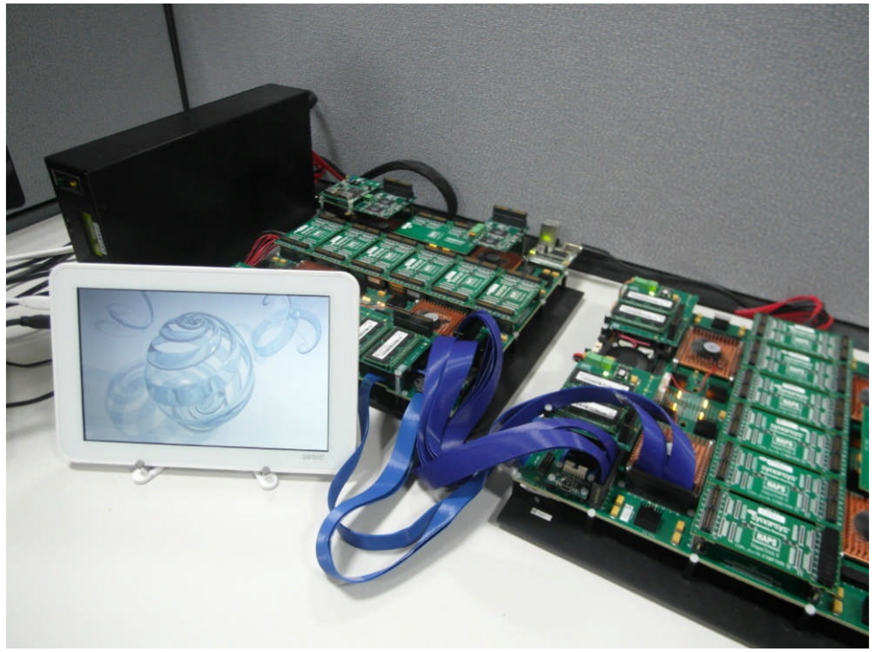
Fiat

Stained Glass



Wedding Ring

7. Hardware Implementation



- Hardware Implementation :
  - Hardware ray tracer framework with FPGA board. Dedicated HW and GPU is used
  - The platform renders same image as in PC simulator and it could extended to global illumination rendering in mobile platform.

8. Conclusion

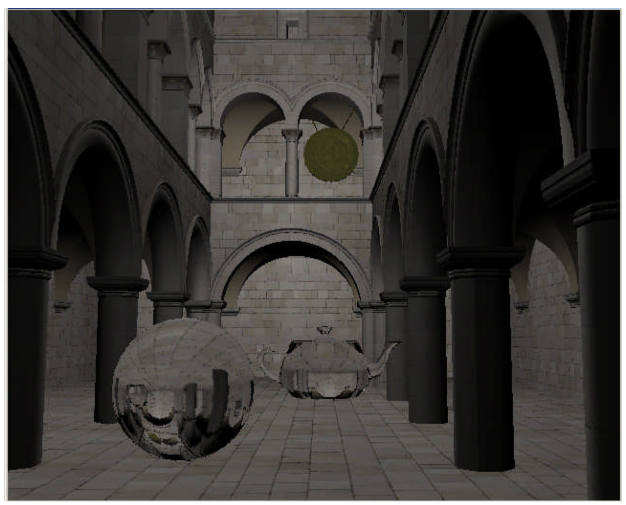
- Efficient Ray Tracing Framework :
  - We develop efficient selective ray tracing framework for real-time global illumination. Our hybrid framework is tested in software framework and hardware platform both.

- Global Illumination Rendering :
  - The selective ray tracing rendering of this work produces same image as full ray tracing using rasterization and ray tracing.
  - The selective ray generation of shadow processing removes aliasing artifact of rasterized shadow. We devise new shadow boundary detection algorithm and it help to increase performance of rendering.

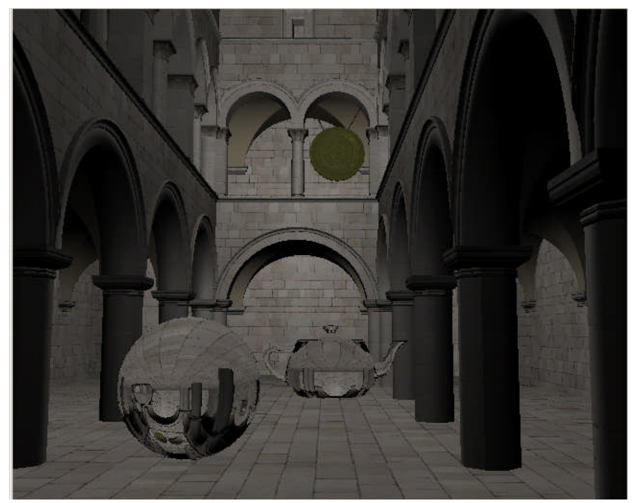
8. Future Work

- LoD Concept for Reduction of Rays
  - The ray number of distance area is reduced and interpolate the adjacent pixels

- Hardware Implementation on Mobile SoC Chip :
  - The FPGA implementation is to be applied to real-time global illumination on mobile environment.



a) Selective Ray Tracing
Ray Number : 27424



a) Selective Ray Tracing w/ LoD
Ray Number : 19710

Effects of added lines and orientations of image on angle illusion in road scene

Song, Jae Seon **Jung, Woo Hyun†**

Department of Psychology, Chungbuk National University, Korea

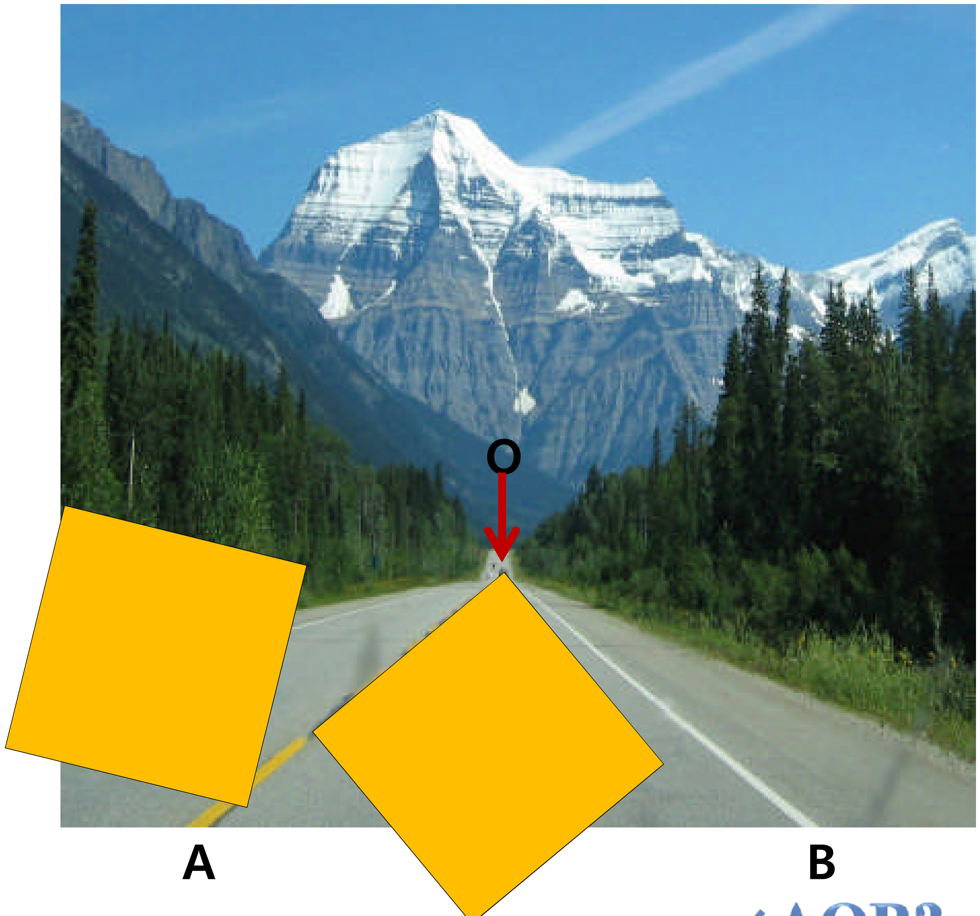
†com4man@gmail.com

Same type of car, Different size?

Same size!

Introduction

Angle illusion in straight road



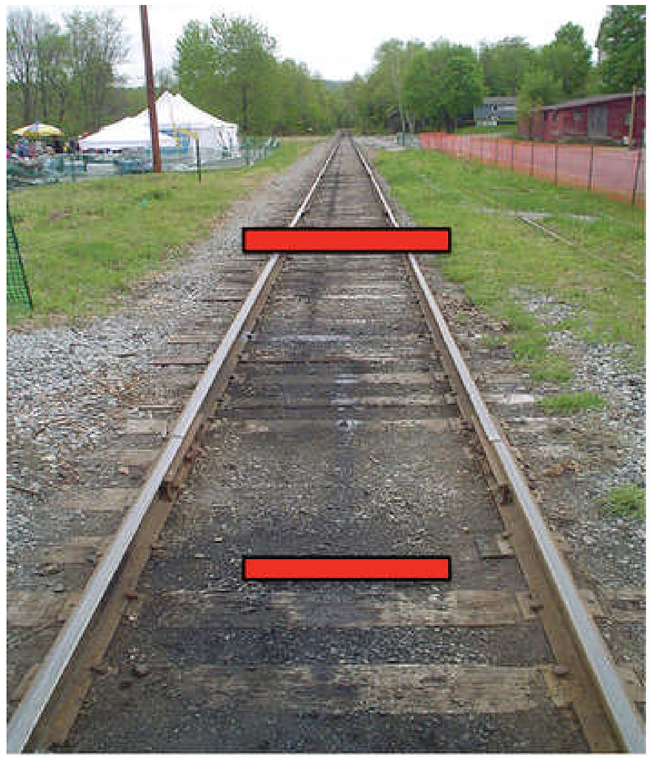
A　　　　B

∠AOB?

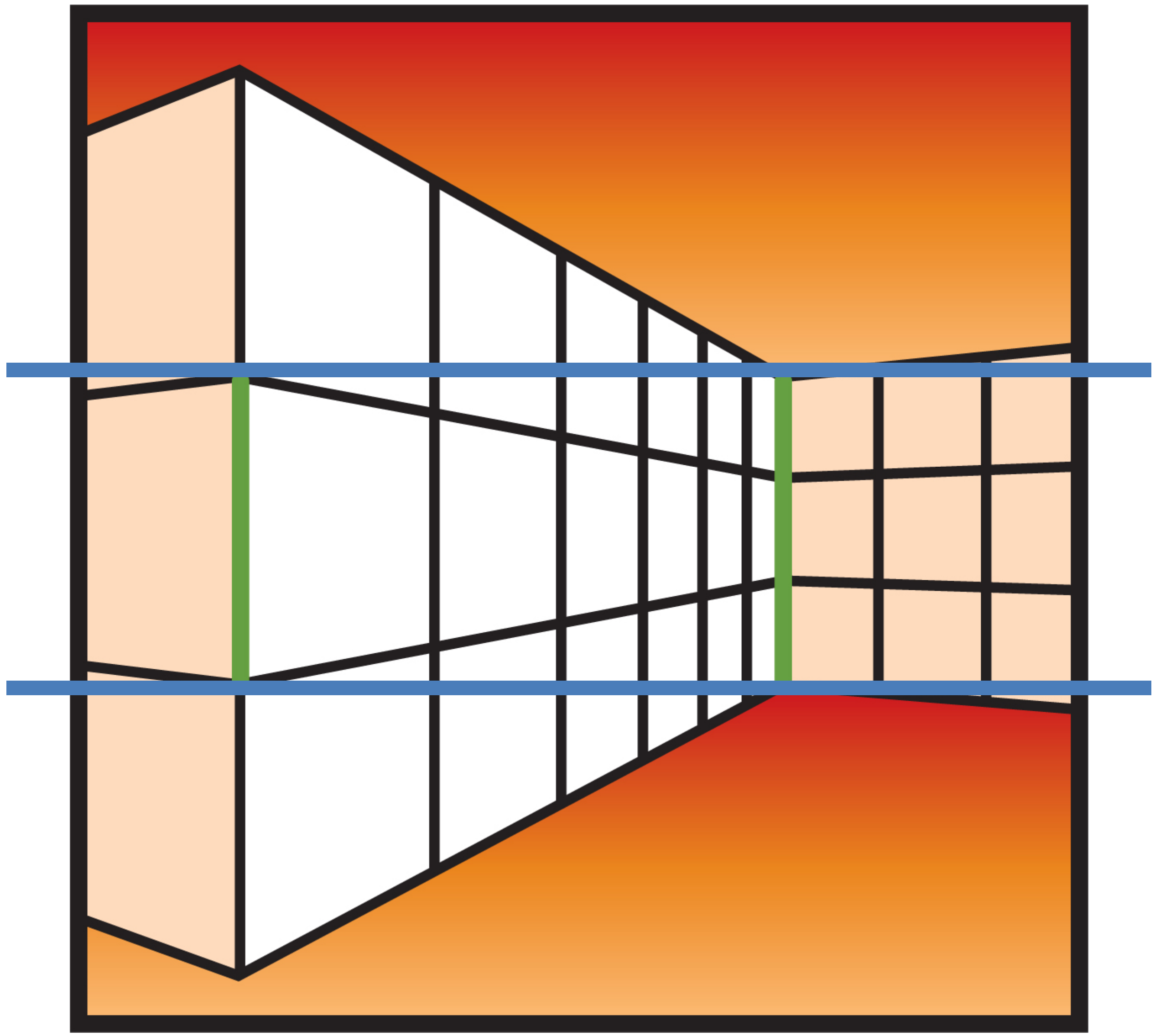
It's 99°

Introduction

Ponzo illusion



Another type of Ponzo illusion



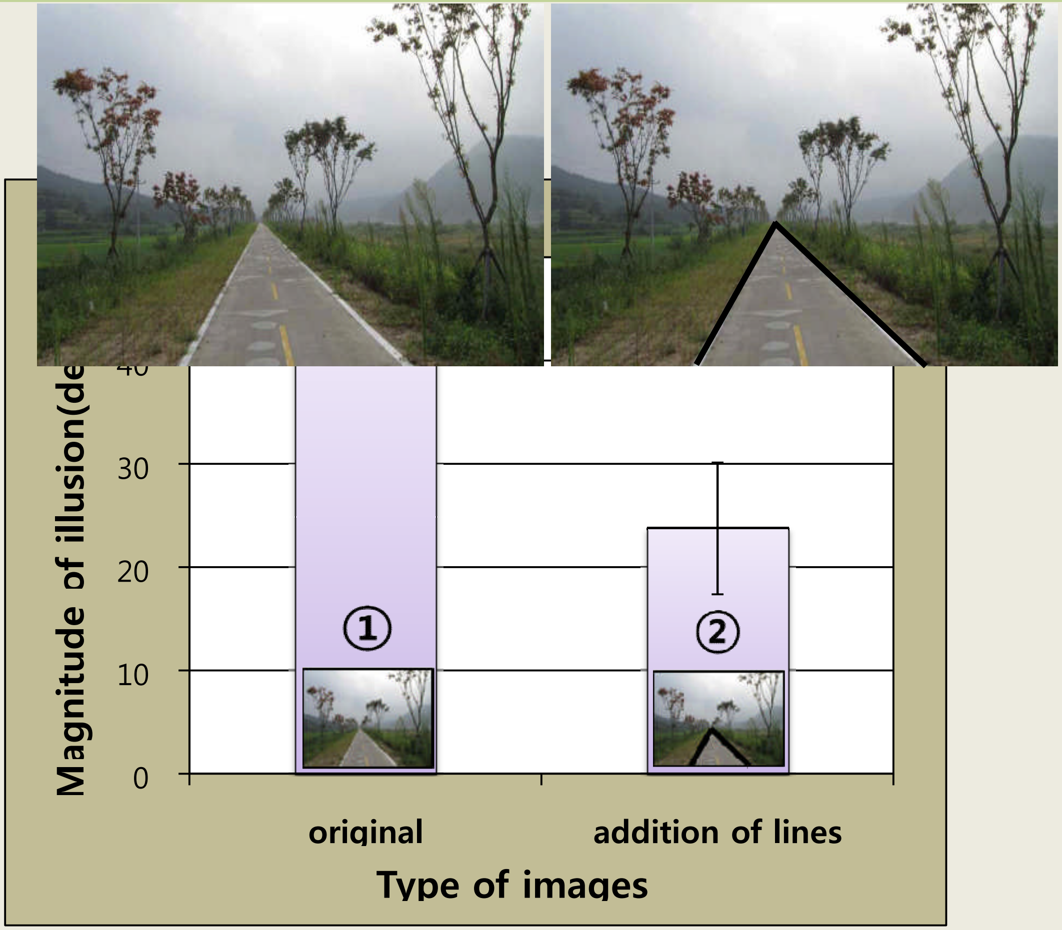
Experiment 1

IV: original image(①)
addition of salient lines to original image(②)
DV: perceived size of angle between two converging lines

①

②

Experiment 1



Magnitude of illusion =
actual angle size - perceived angle size

Experiment 2