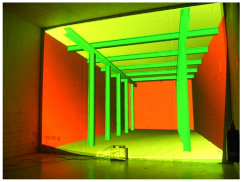
Multi-Projector Techniques for Real-Time Visualizations in

Outline

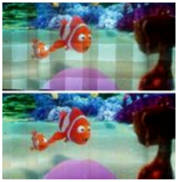
Outline

these slides:
www.uni-weimar.de/medien/AR



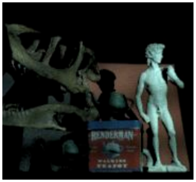
Introduction
Motivations and Applications

Geometric Correction
Planar, Non-Trivial, Complex Surfaces



Radiometric Compensation
Local and Global Light Effects

Advanced Techniques
View-Dependence, Multi-Focal Projection, Light Transport

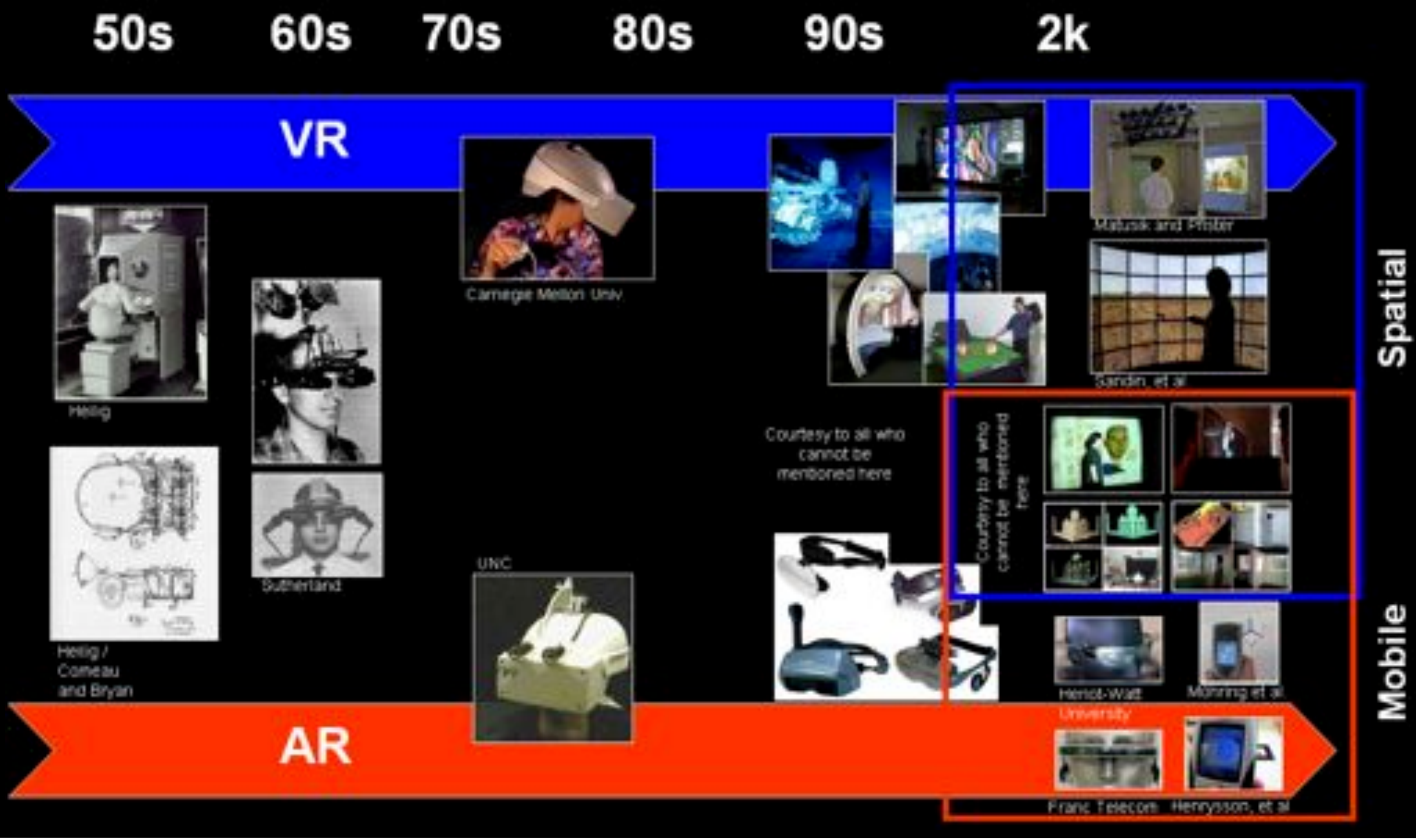


Outlook
Limitations and Future Work

Introduction

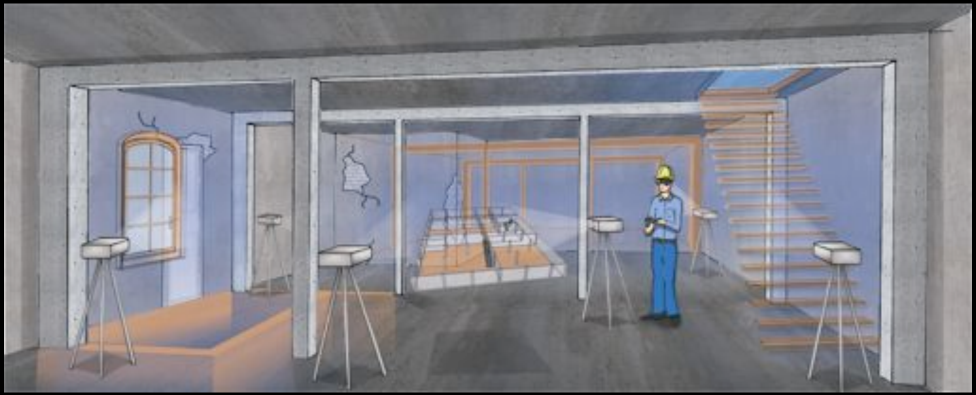
Evolving Evolution

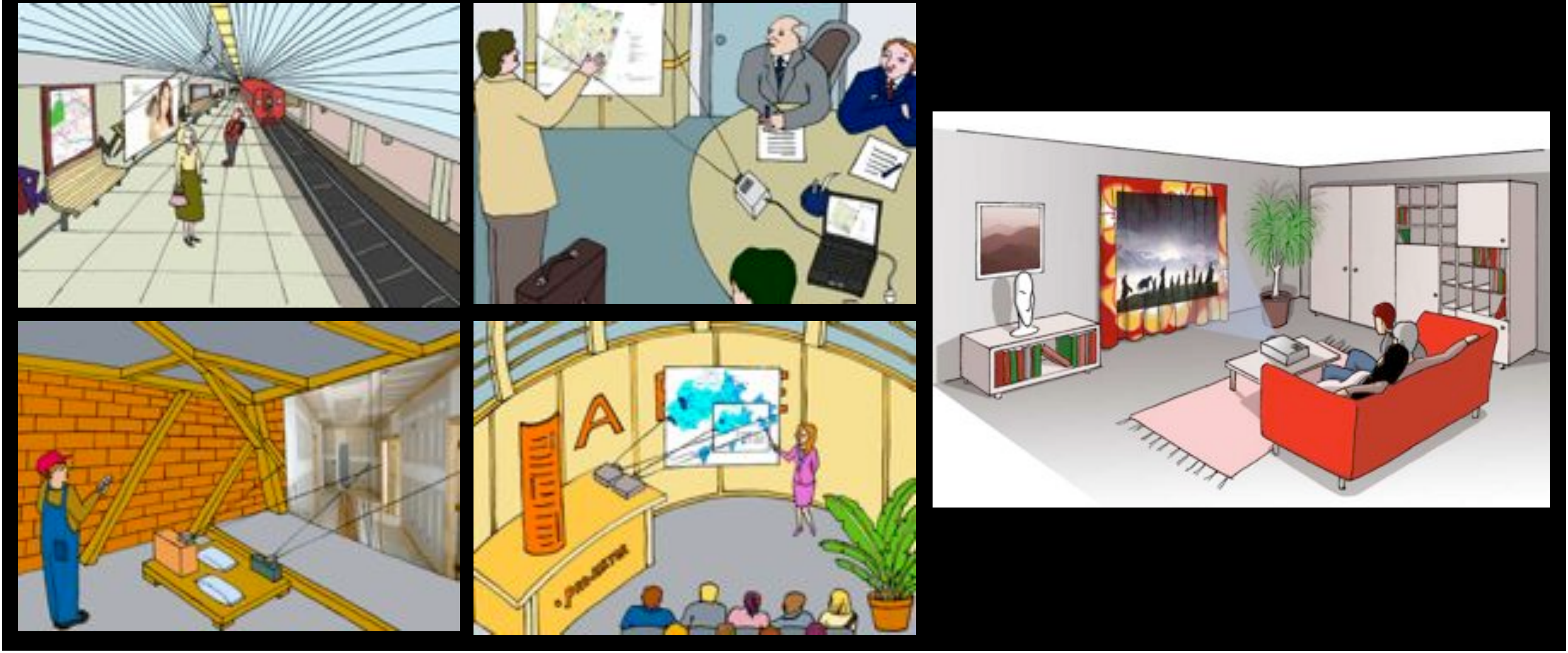
Evolving Evolution



Motivation: Projection

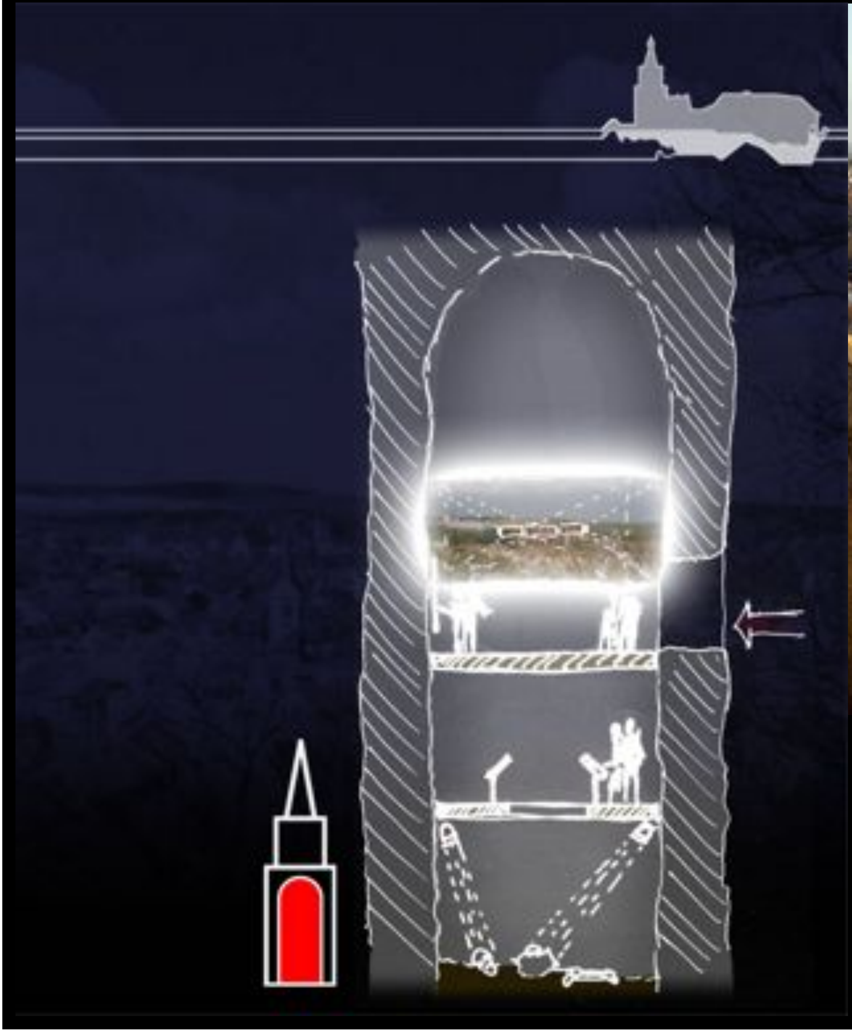
Motivation: Projection





Application: Historic Sites and Museums

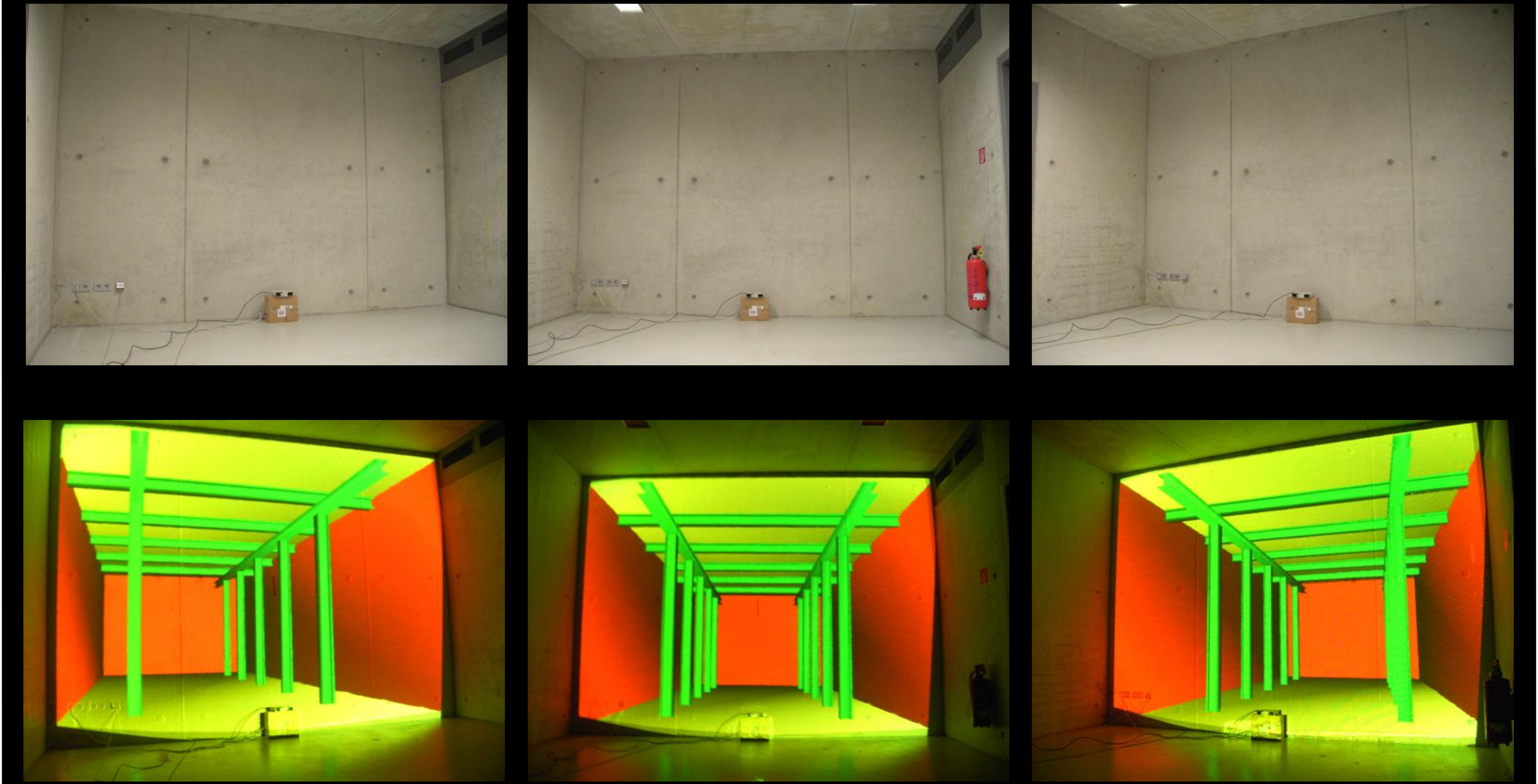
Application: Historic Sites and Museums



360° Surround Projection in Castel Tower (Running project in coop. with Bennert Group)

Application: Architectural Visualization

Application: Architectural Visualization



Bimber et al, IEEE/ACM
ISMAR 2005

On-Site Architectural Visualizations
(Running project in coop. with Architecture Faculty, BUW)

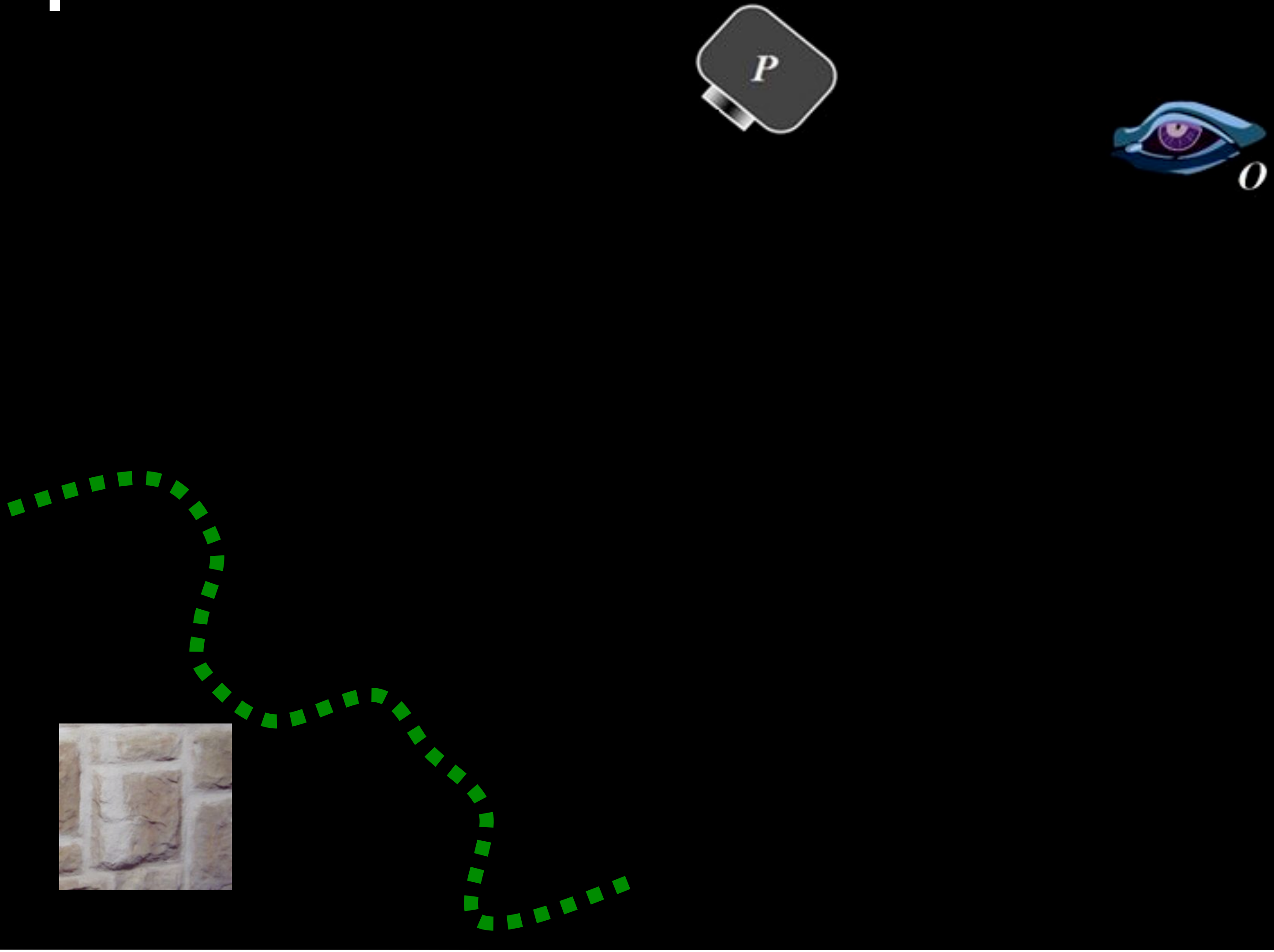
Application: Pocket Projectors

Application: Pocket Projectors

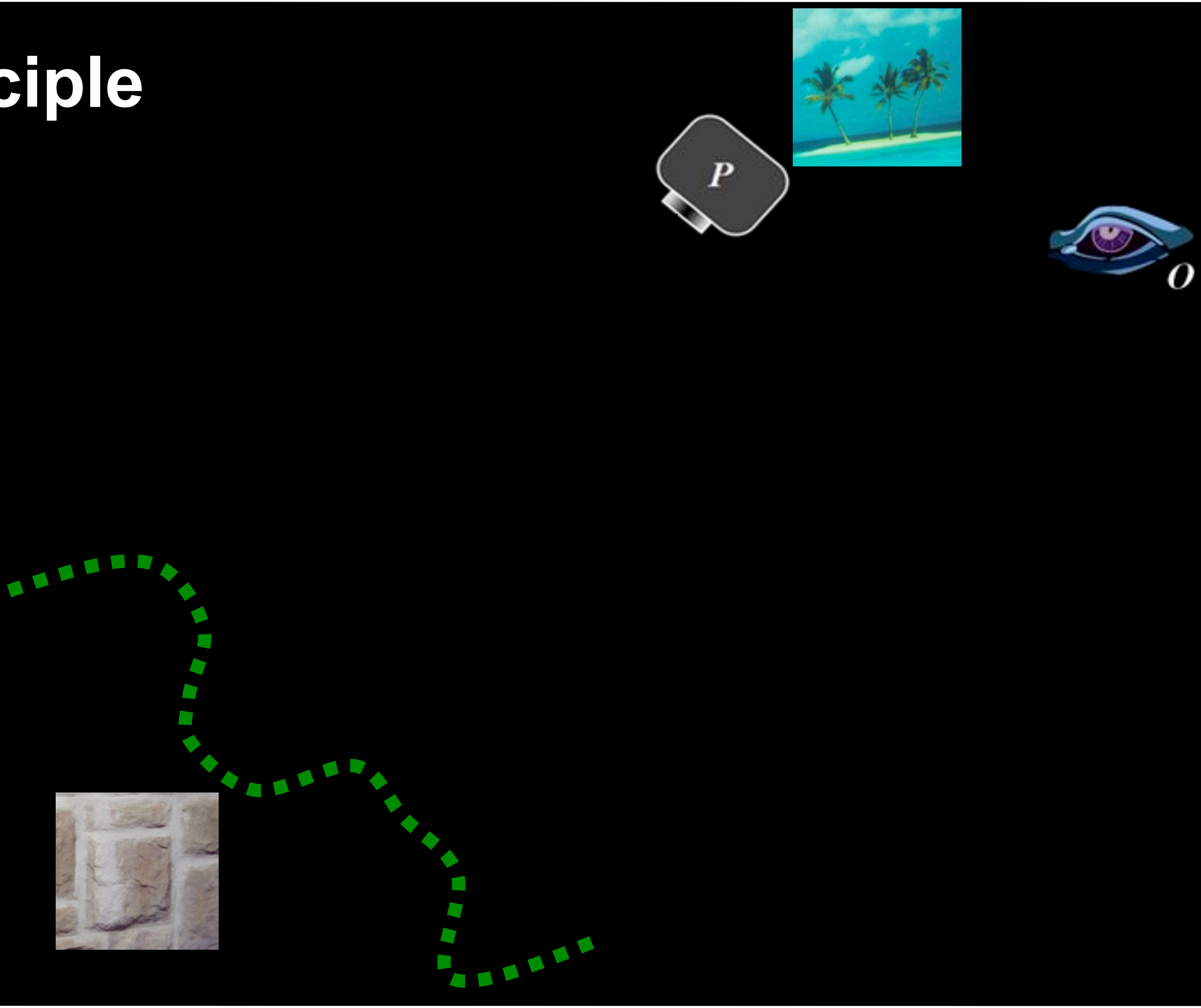
Courtesy: InFocus

Courtesy: Siemens

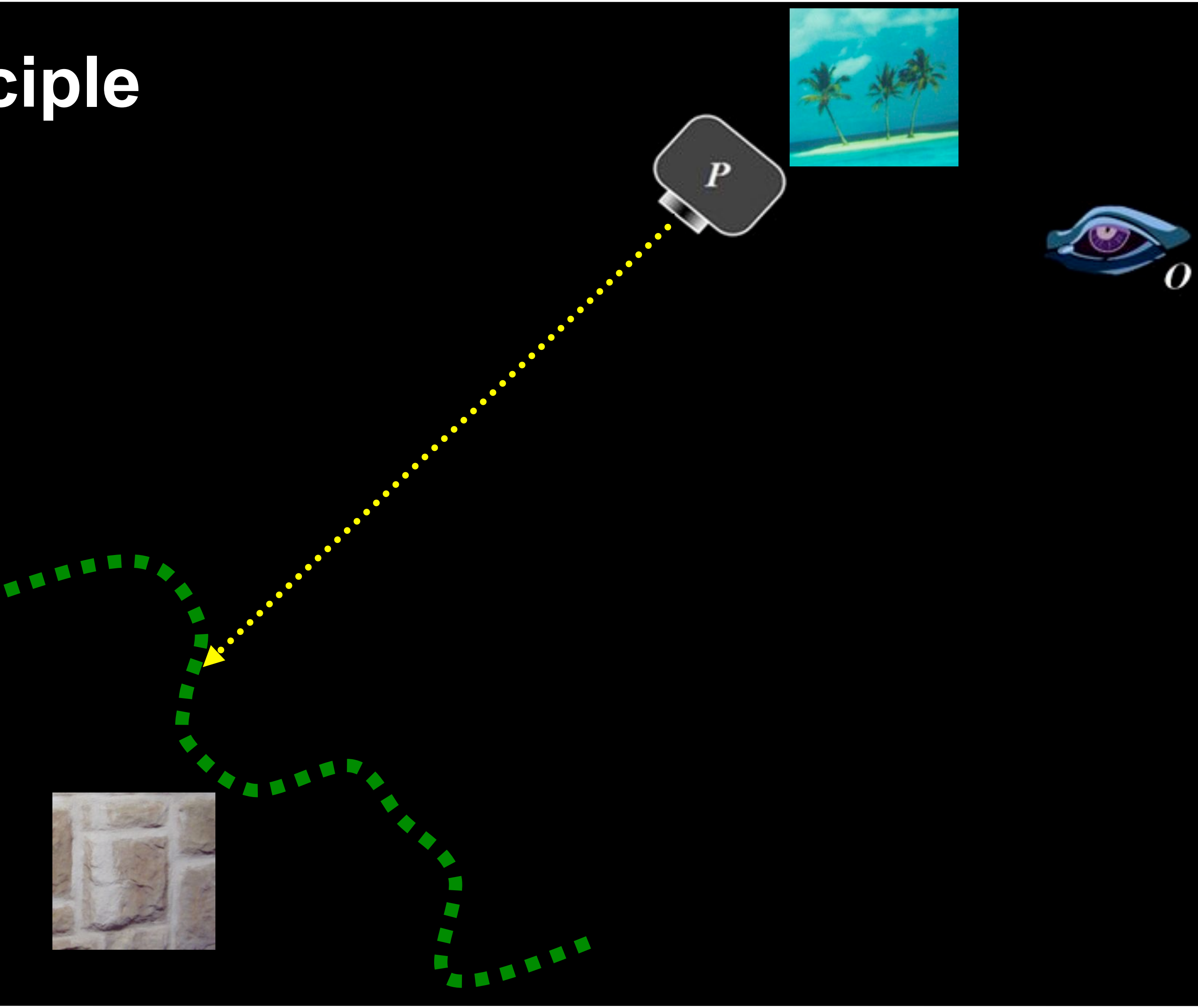
Principle



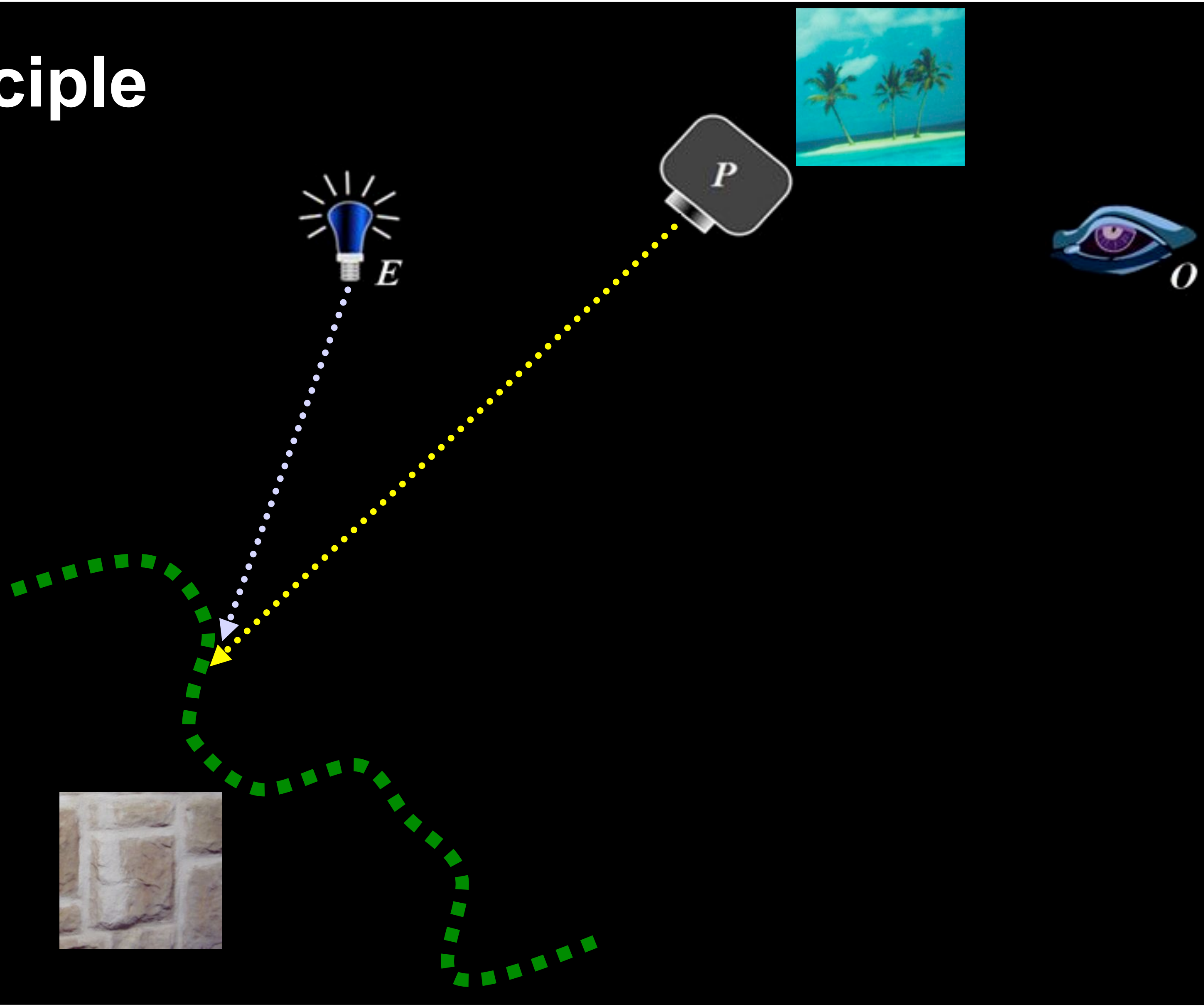
Principle



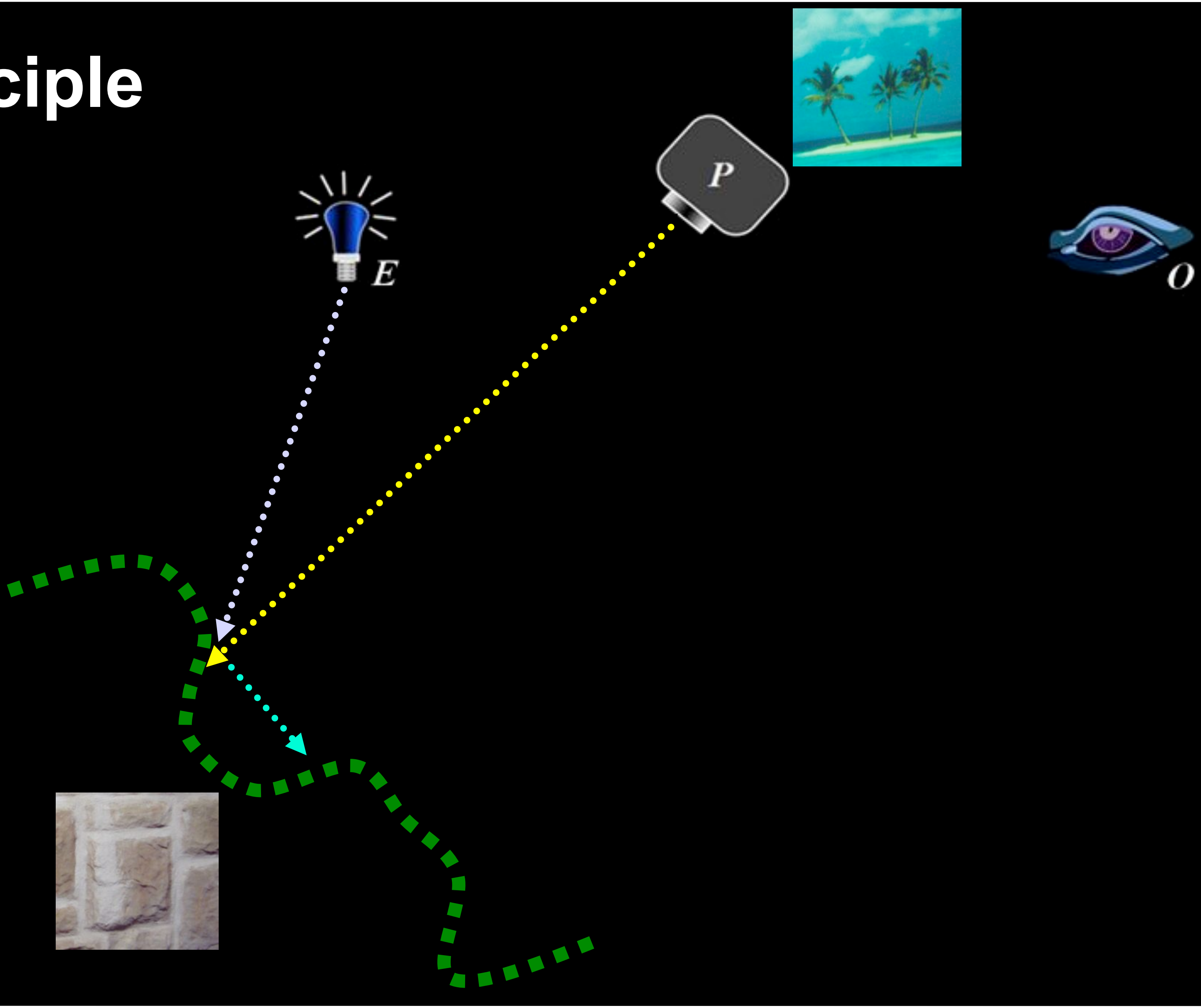
Principle



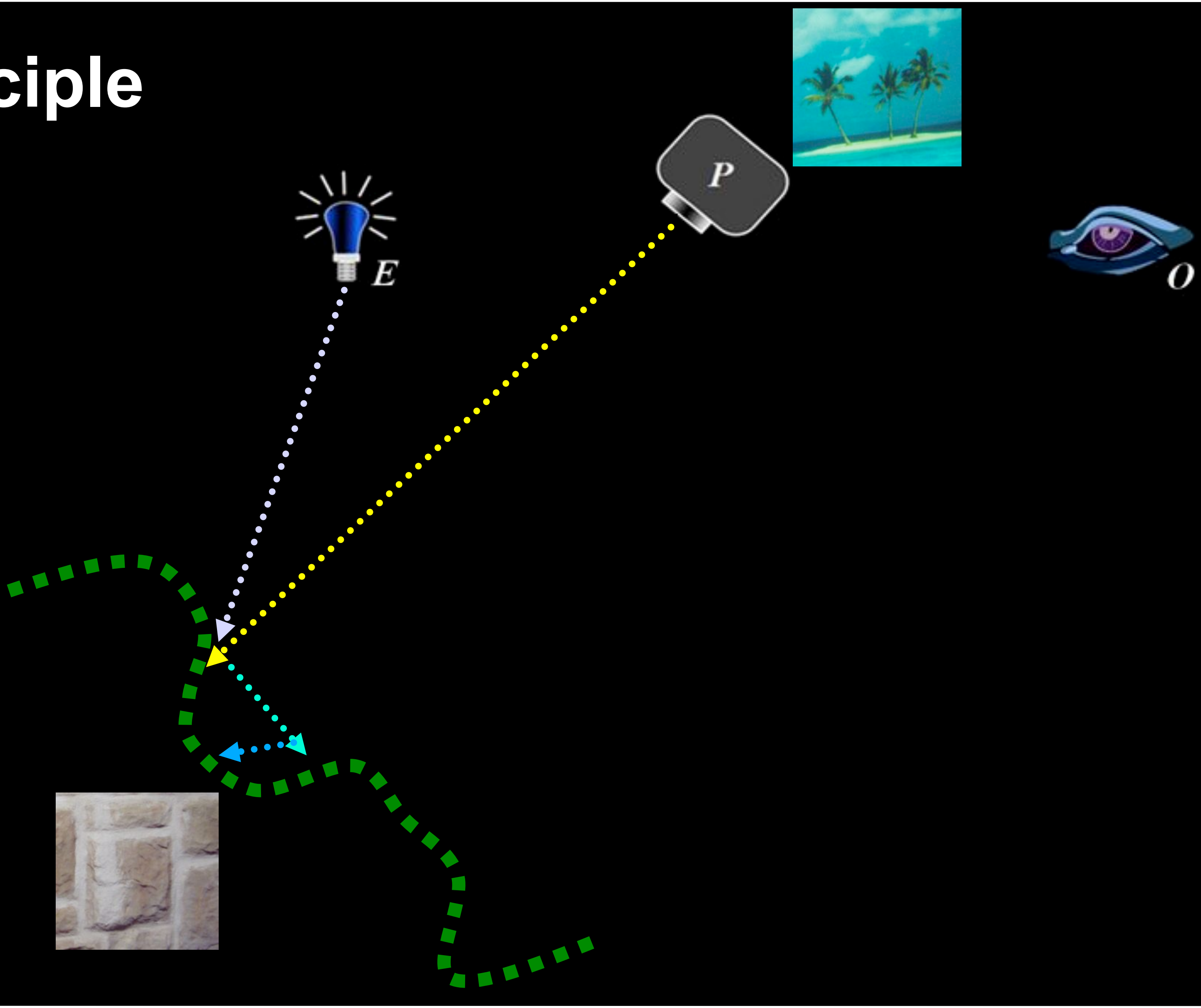
Principle



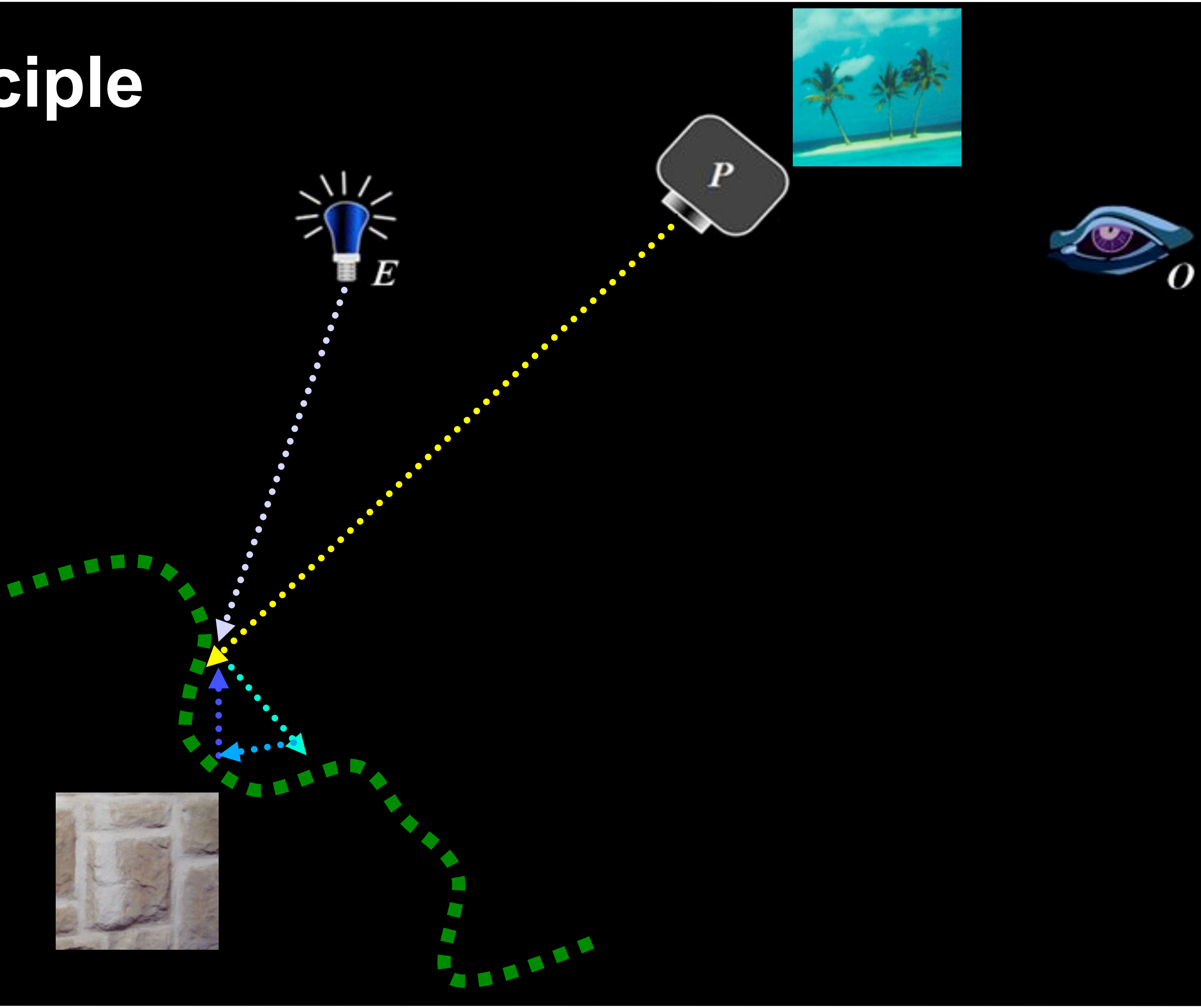
Principle



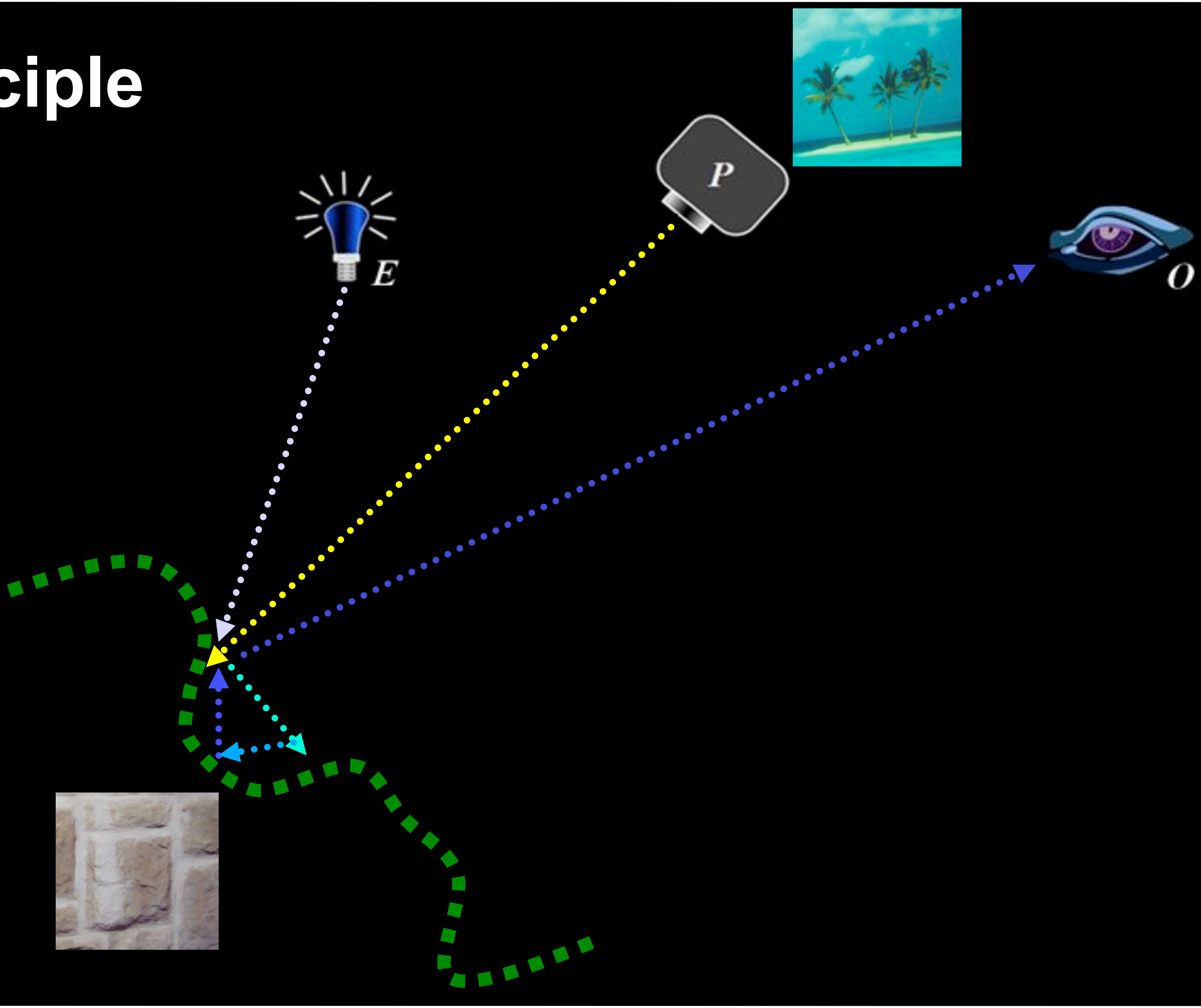
Principle



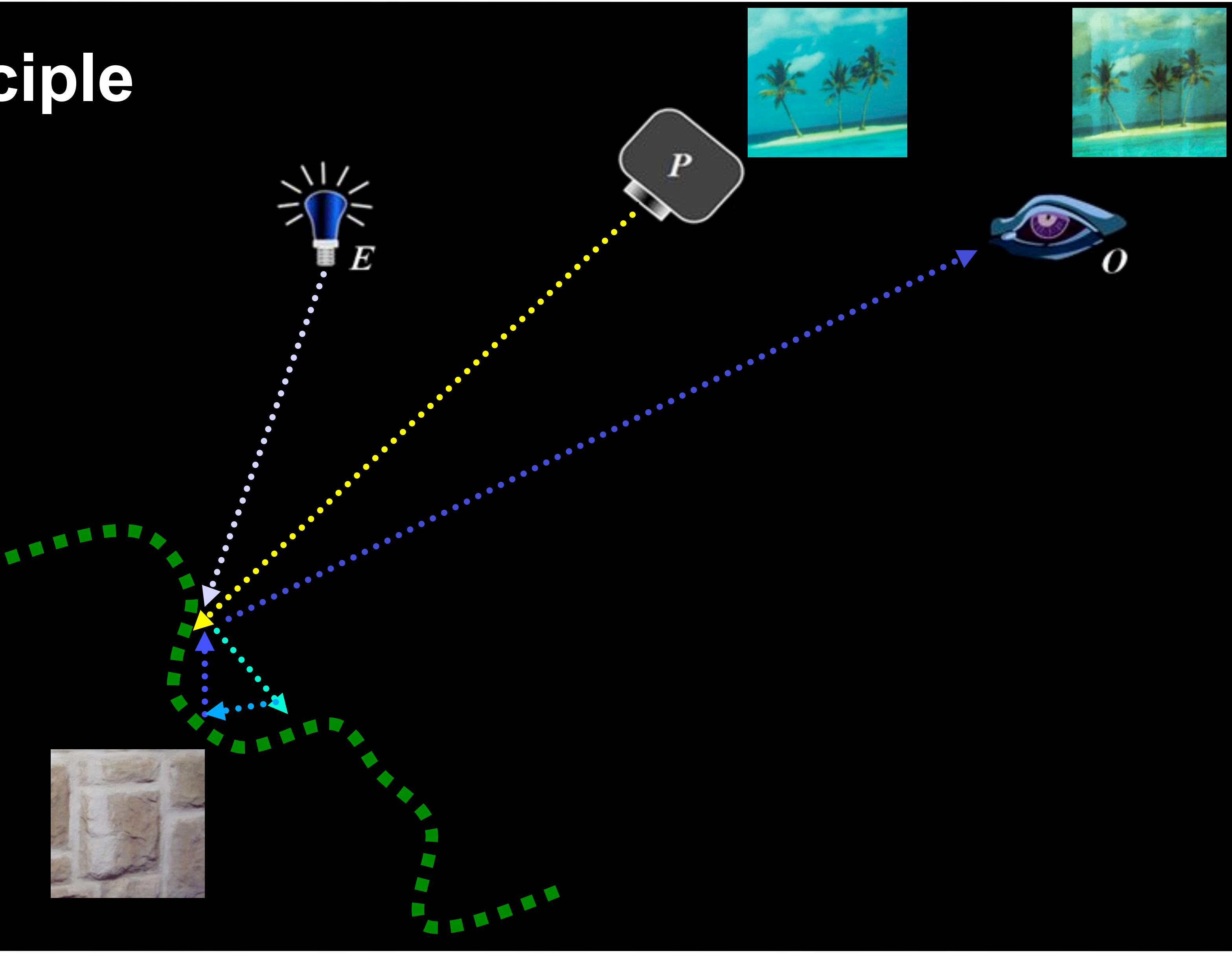
Principle



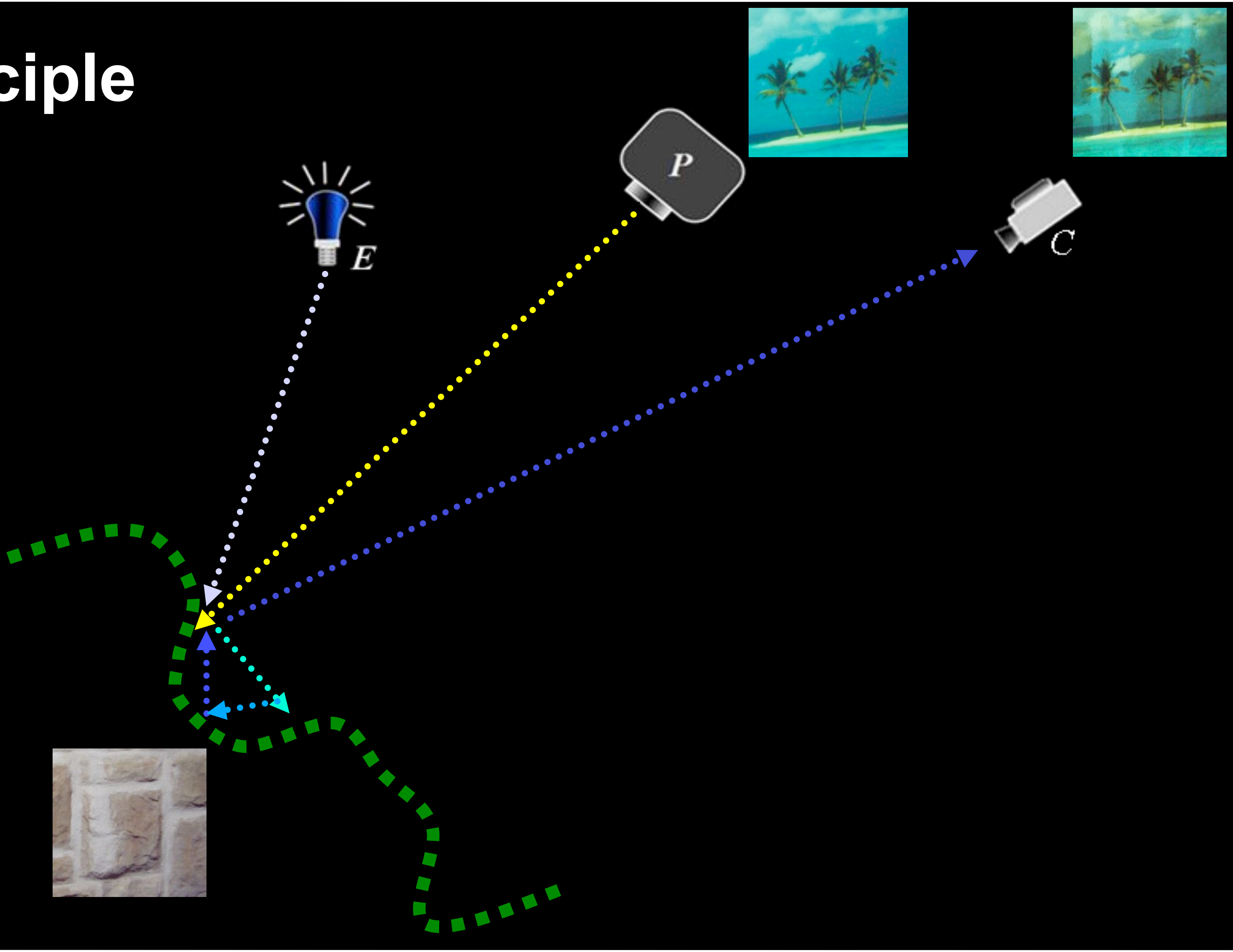
Principle



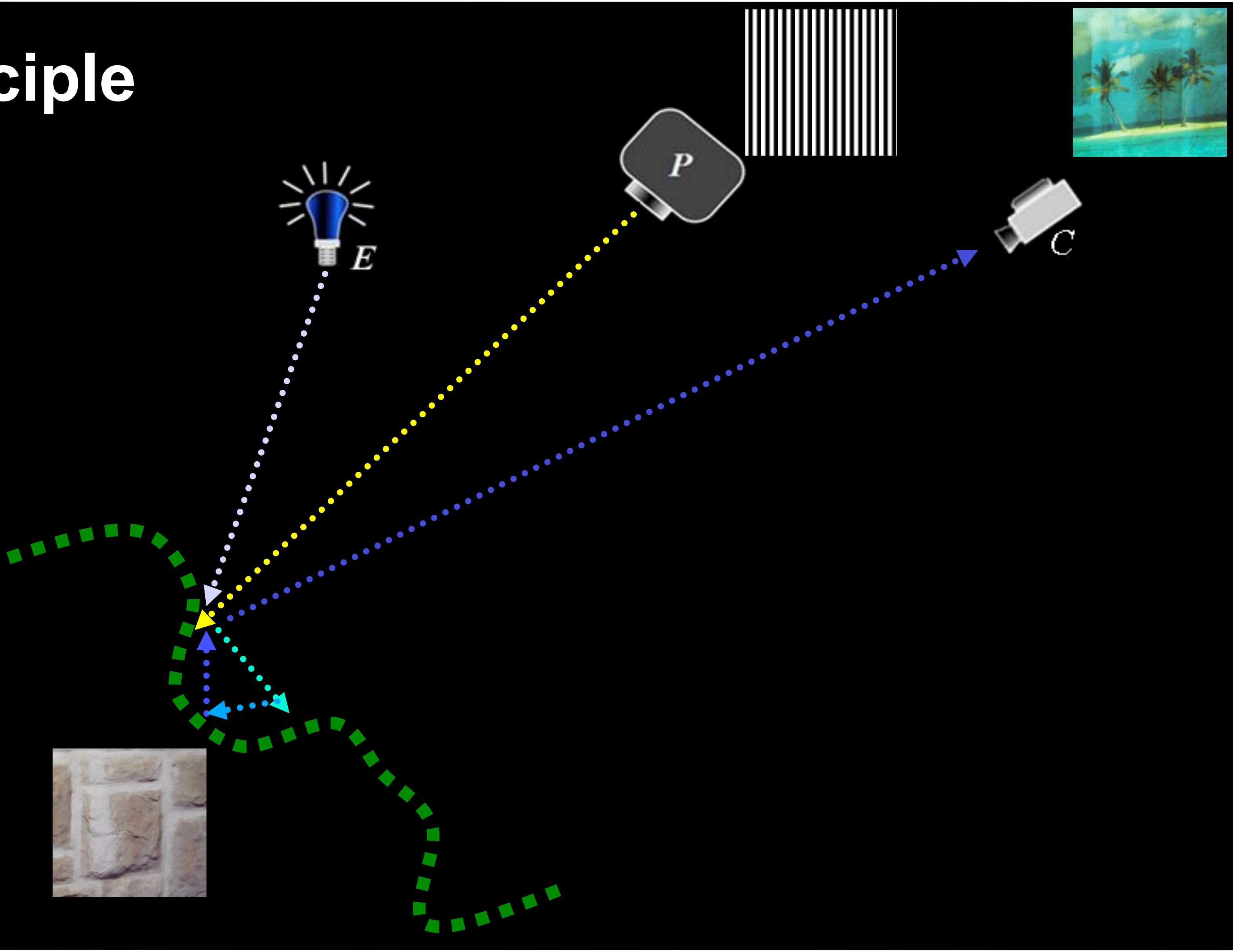
Principle



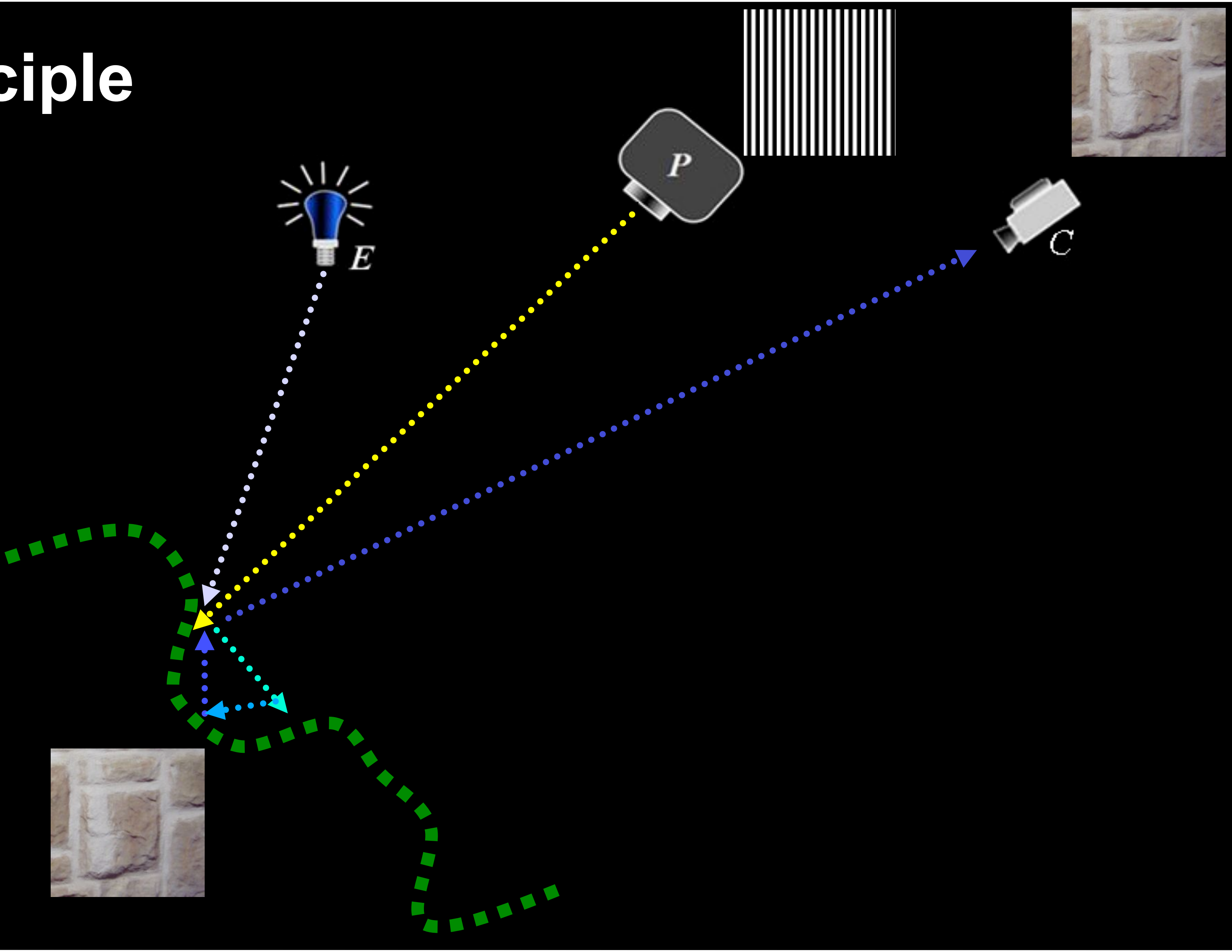
Principle



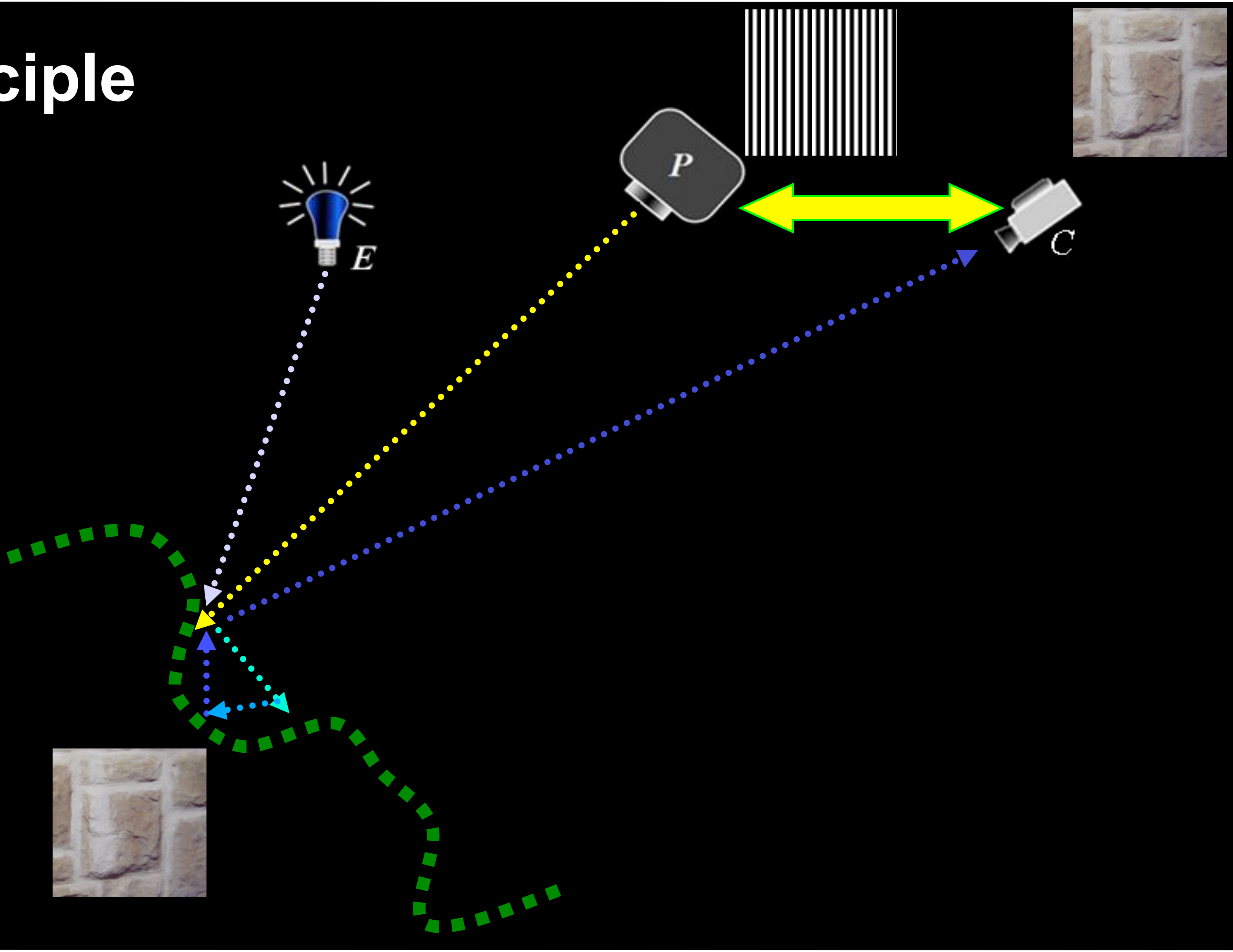
Principle



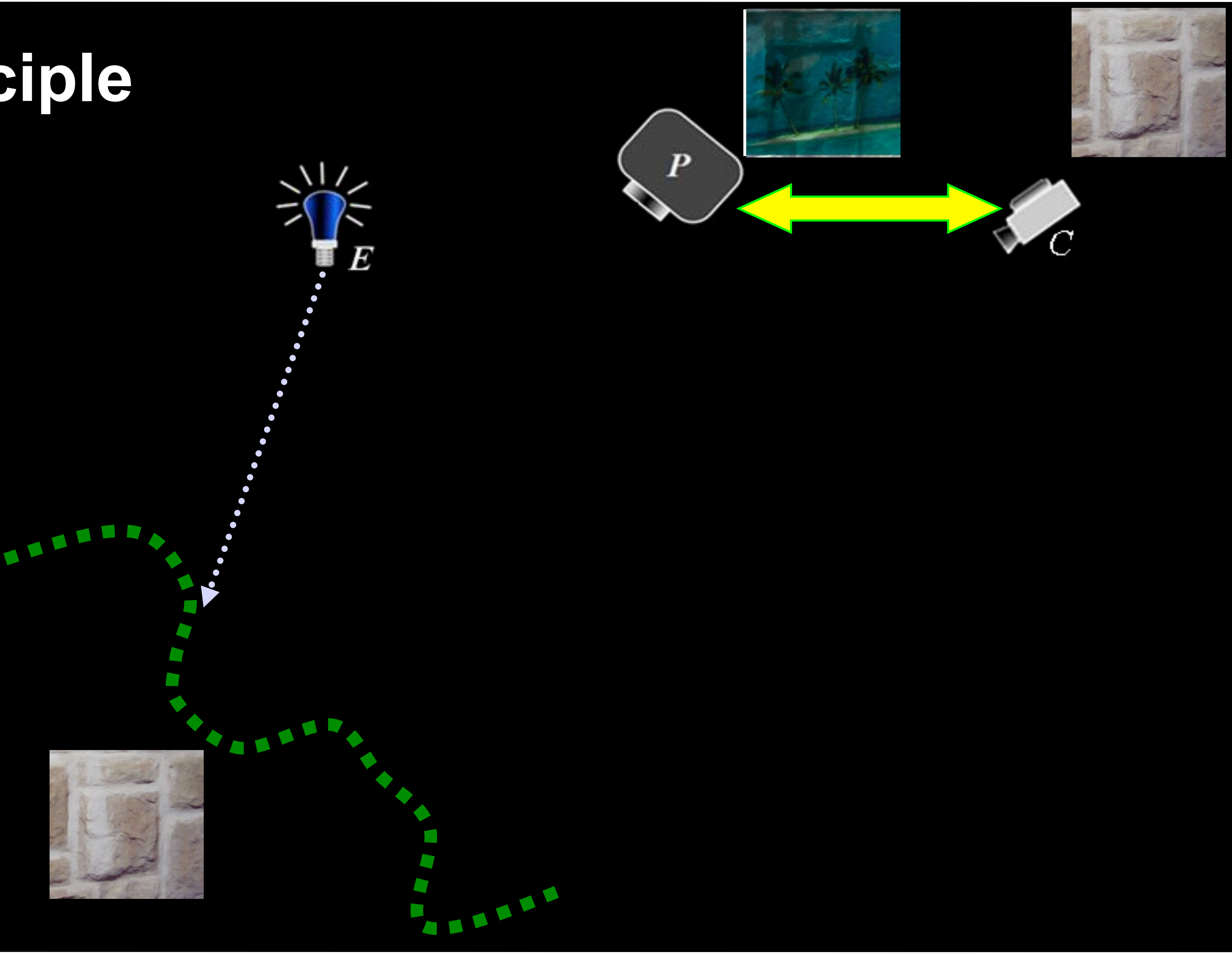
Principle



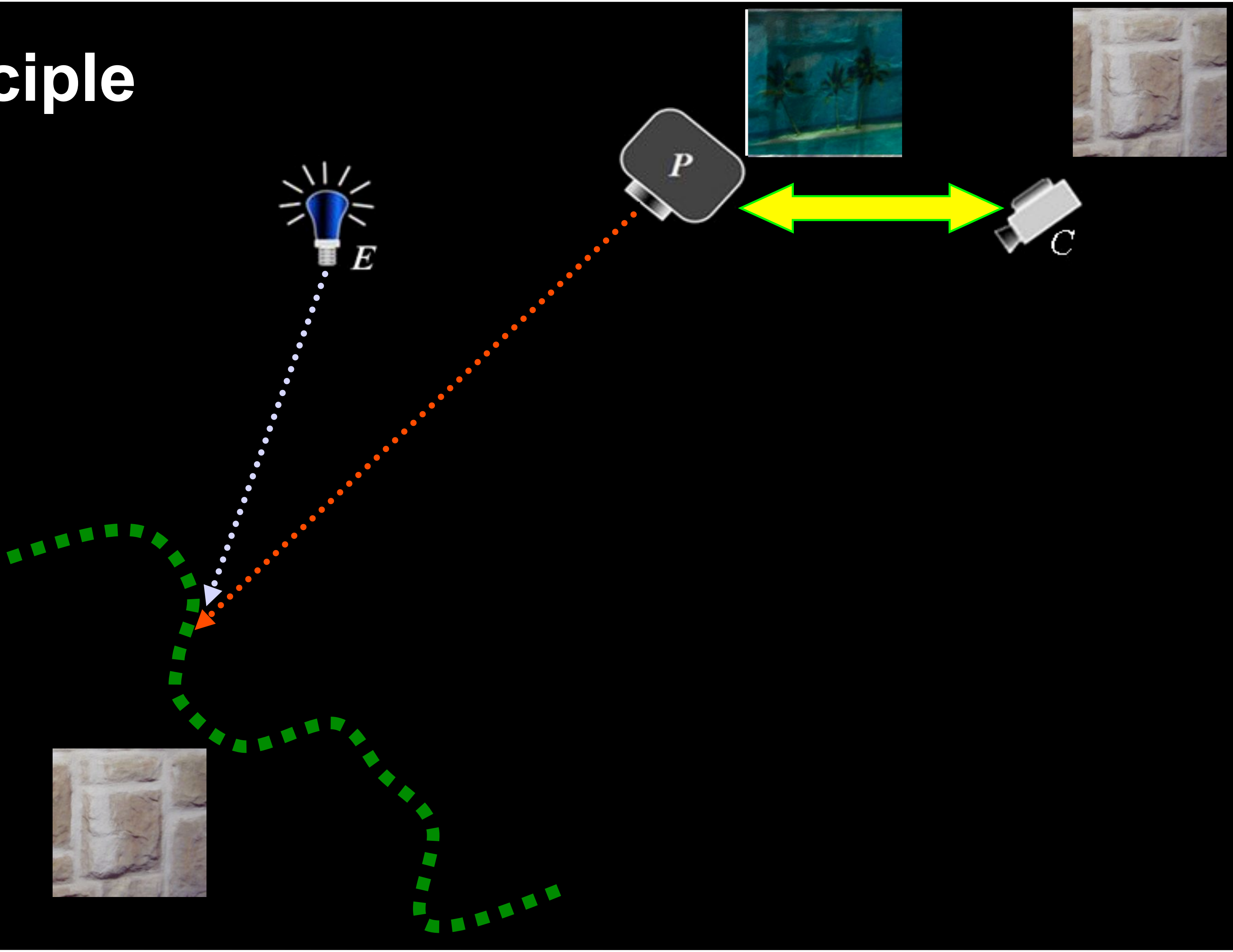
Principle



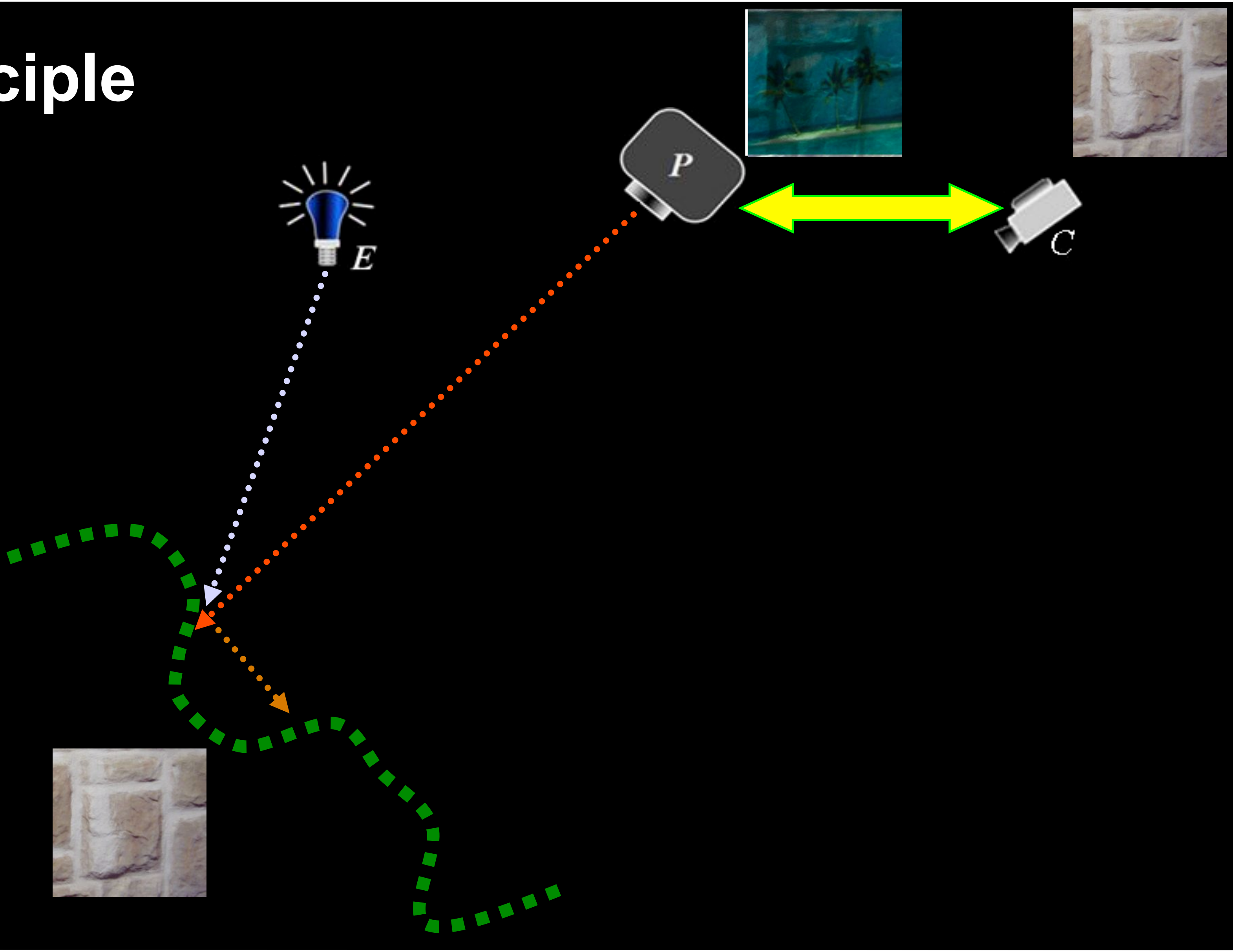
Principle



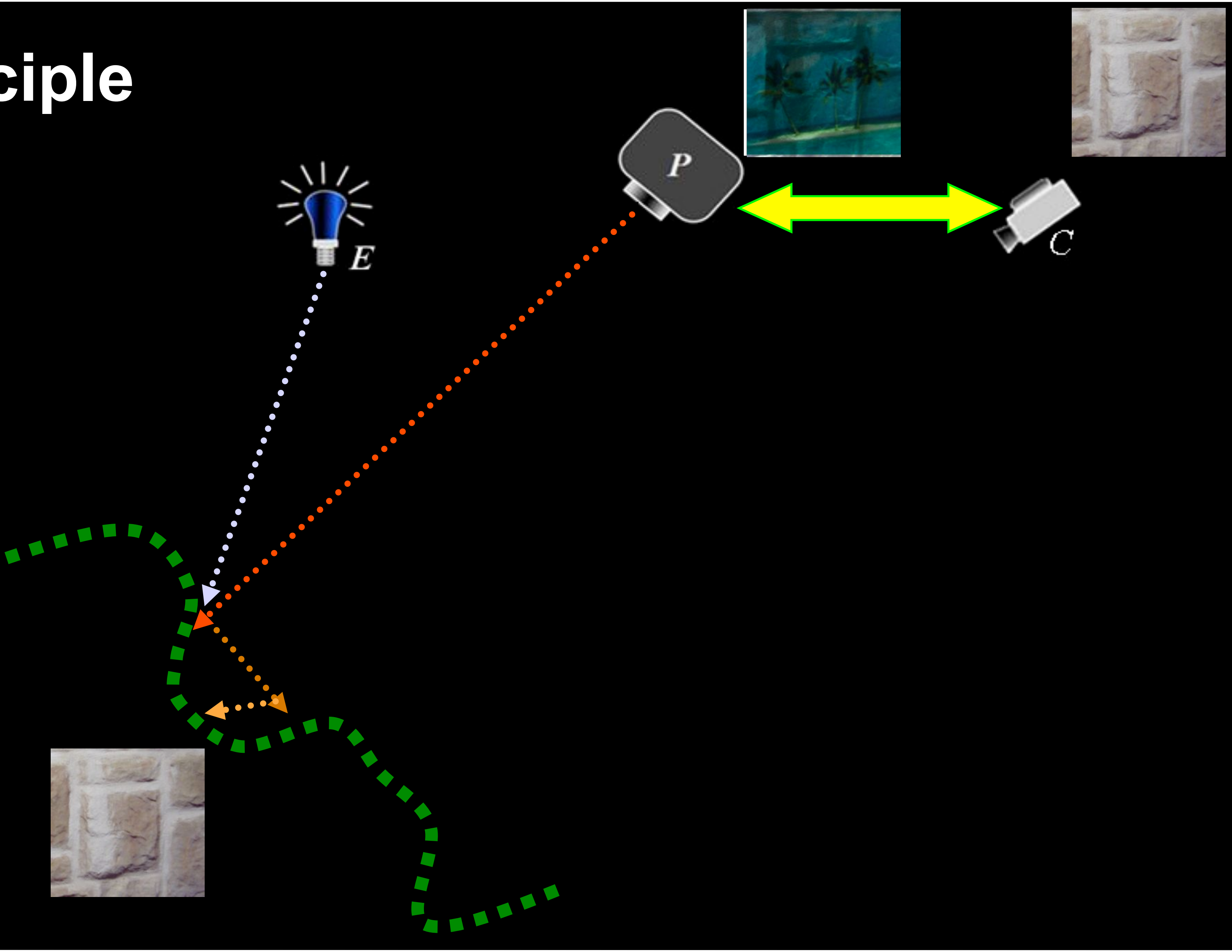
Principle



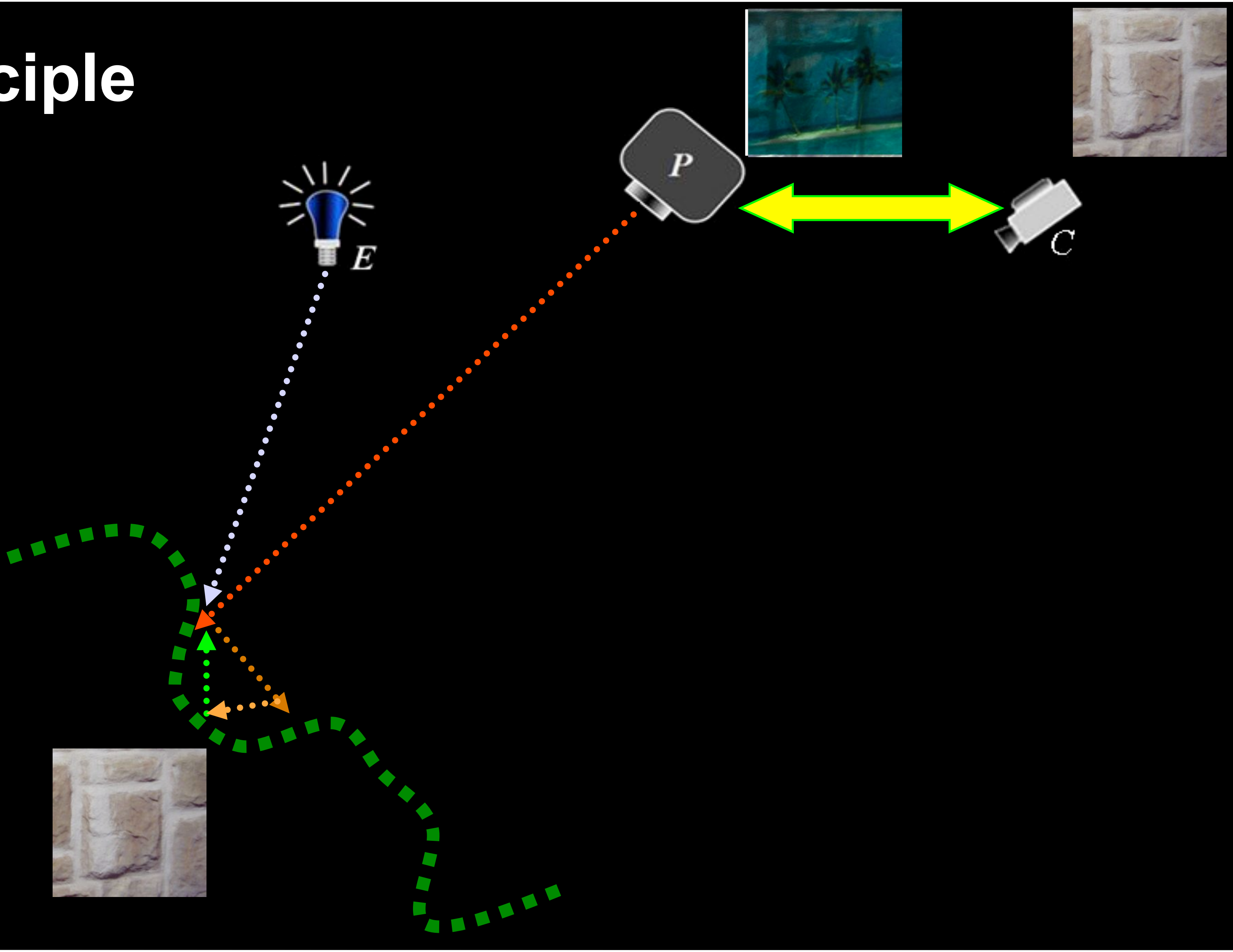
Principle



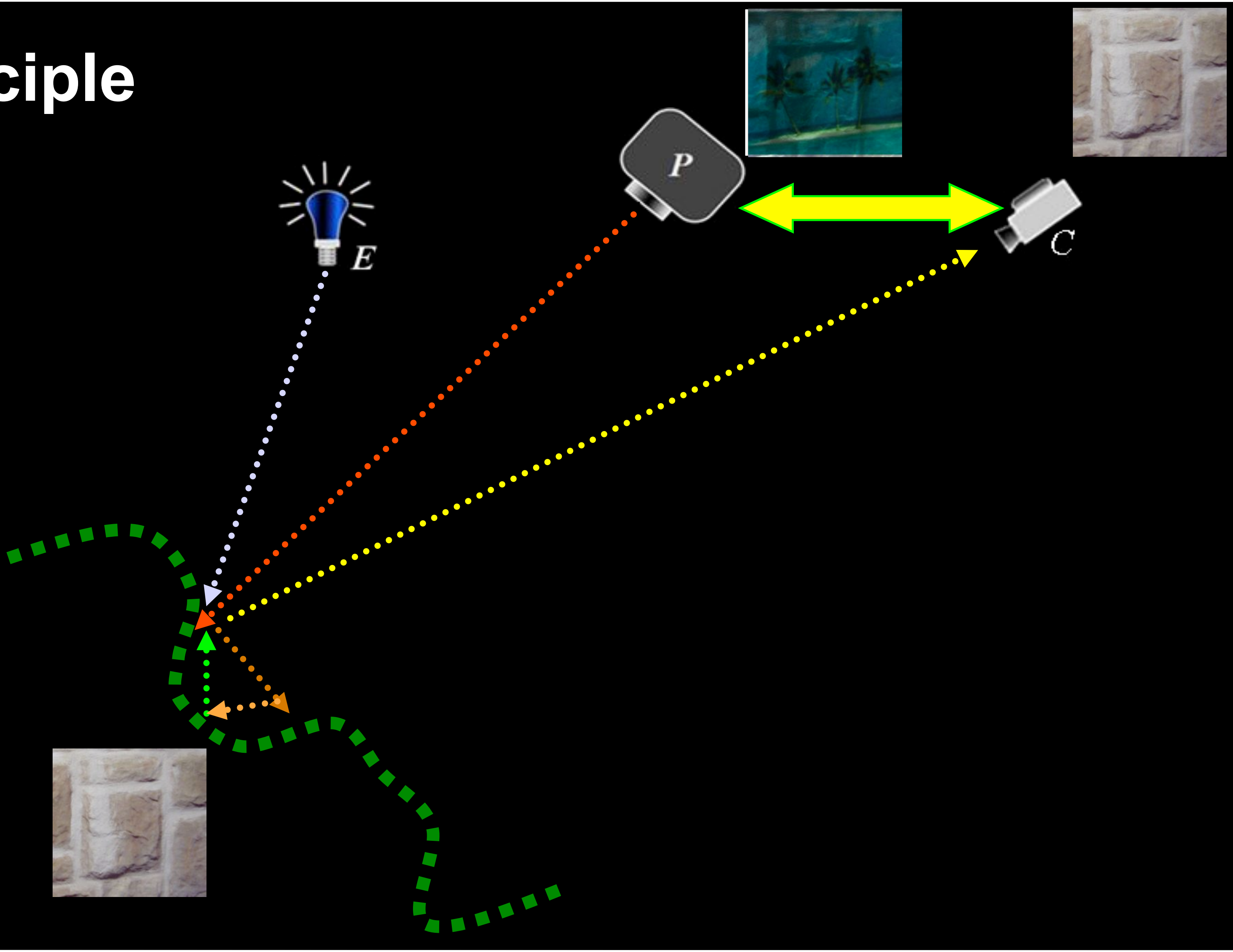
Principle



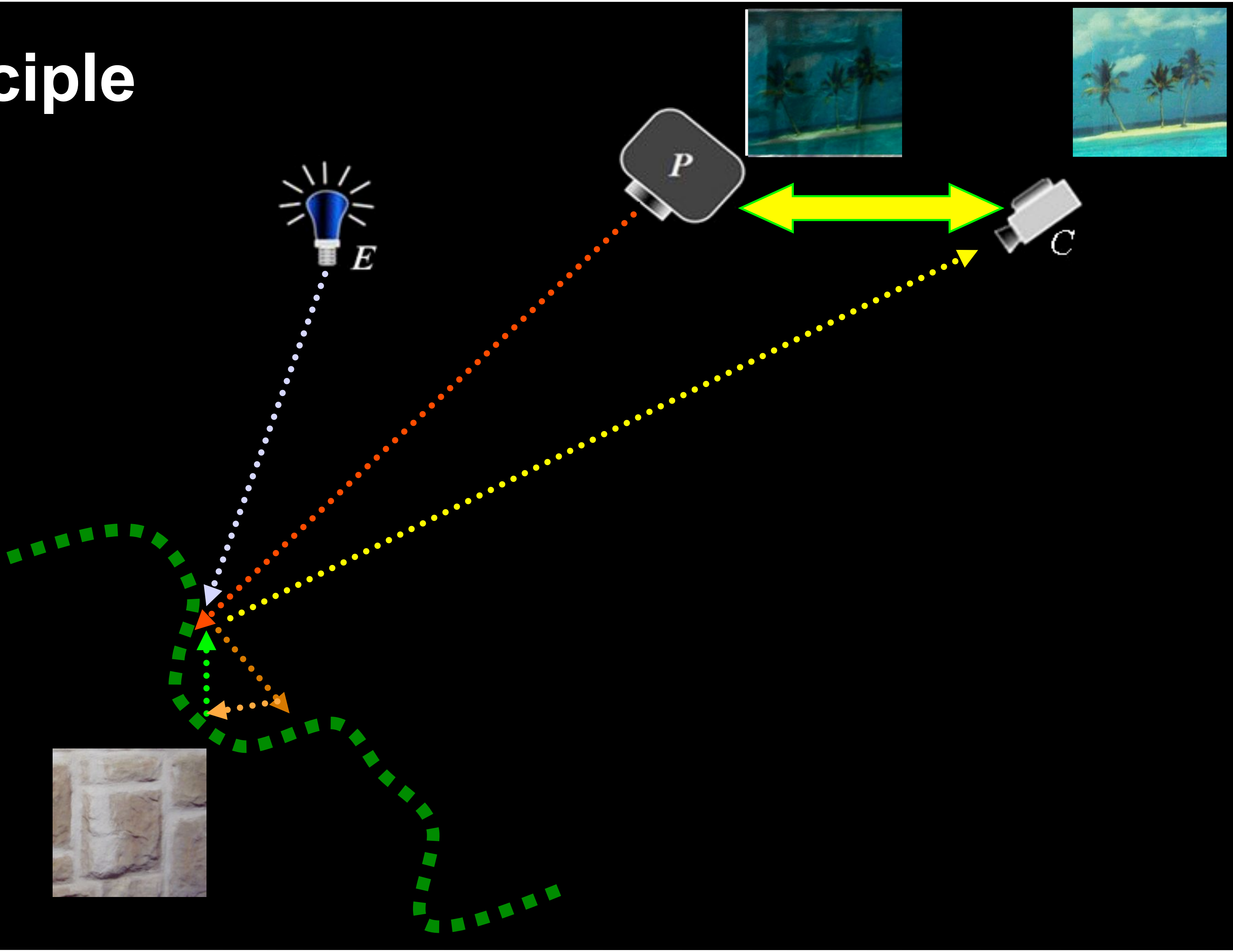
Principle



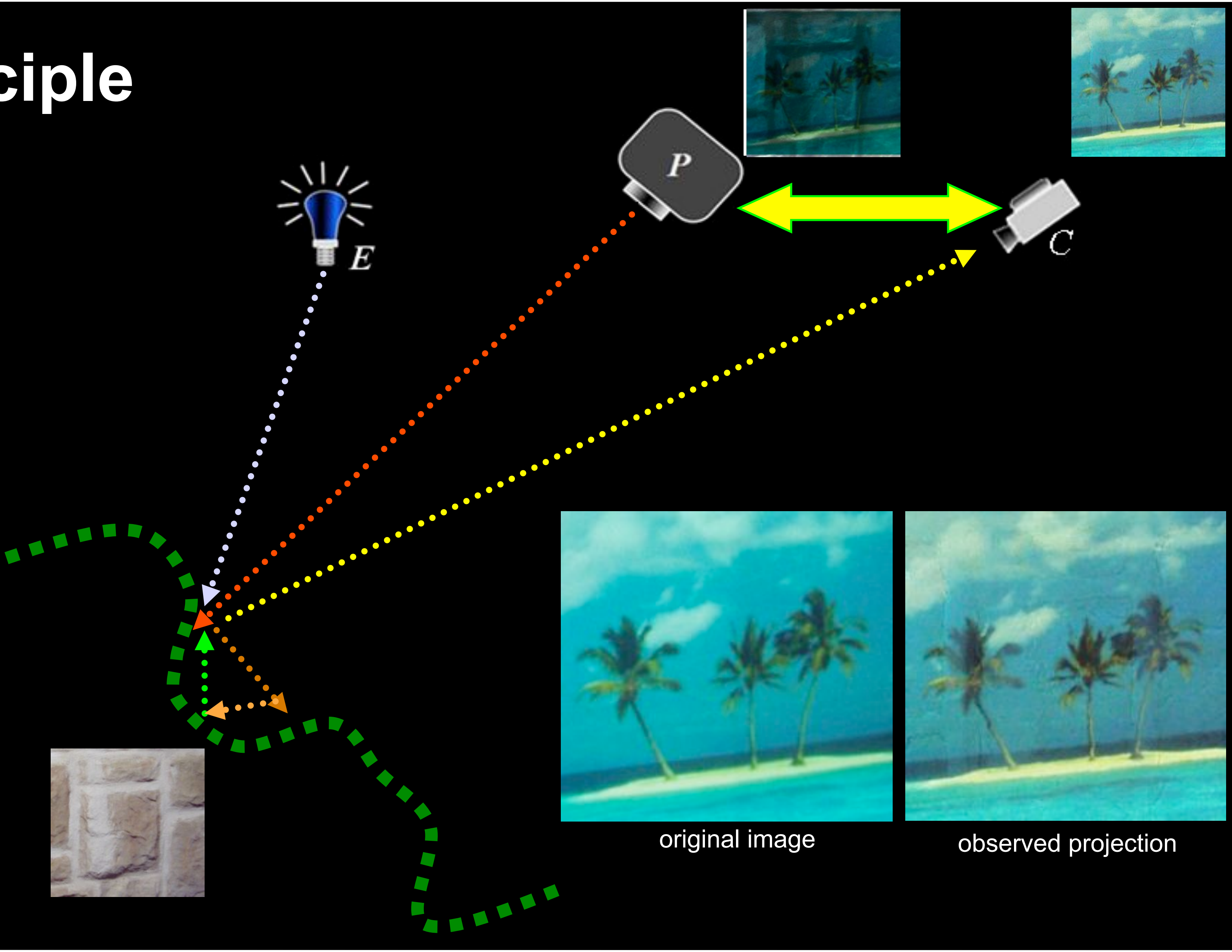
Principle



Principle



Principle



original image

observed projection

Some Challenges

Some Challenges

color blending

Some Challenges

color blending

geometric warping

Some Challenges


color blending


geometric warping

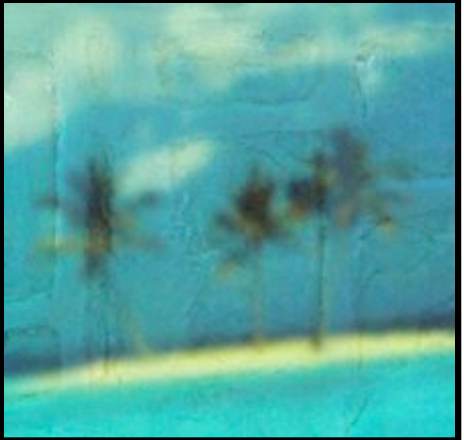

misregistration

Some Challenges

color blending

geometric warping

misregistration



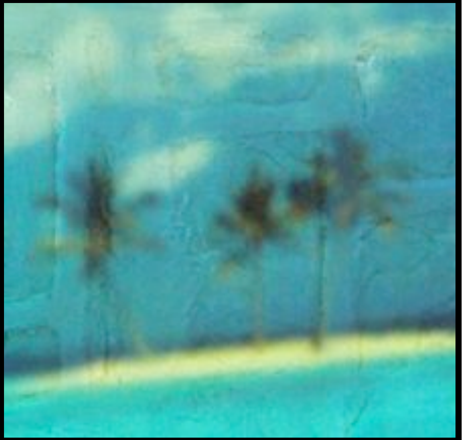
regional defocus

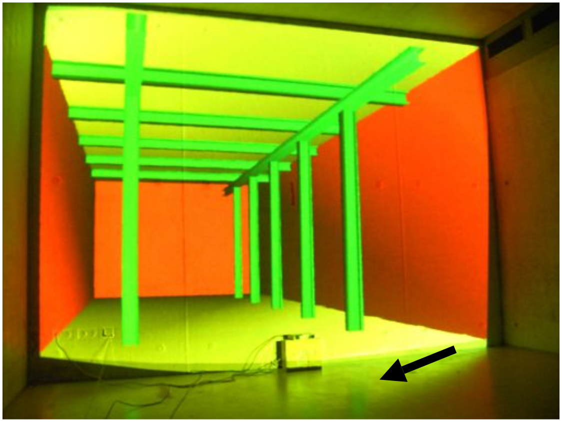
Some Challenges


color blending


geometric warping


misregistration


regional defocus

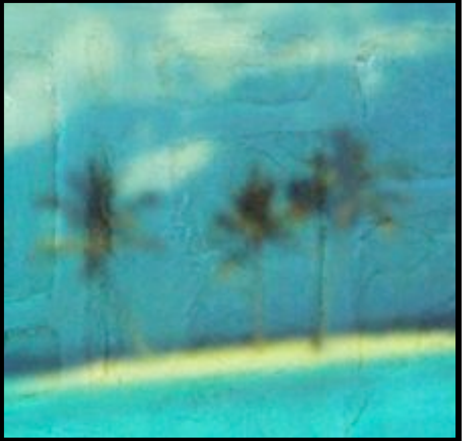

scattering

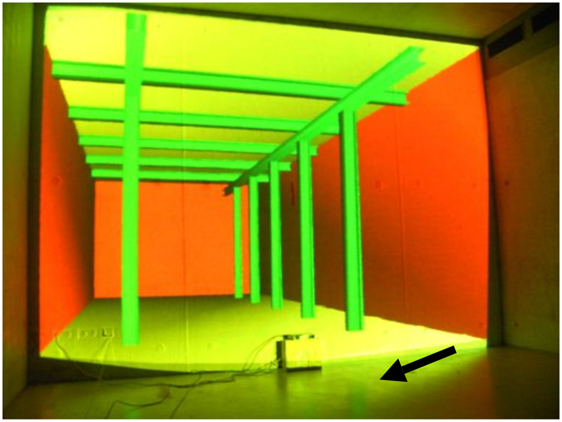
Some Challenges


color blending


geometric warping

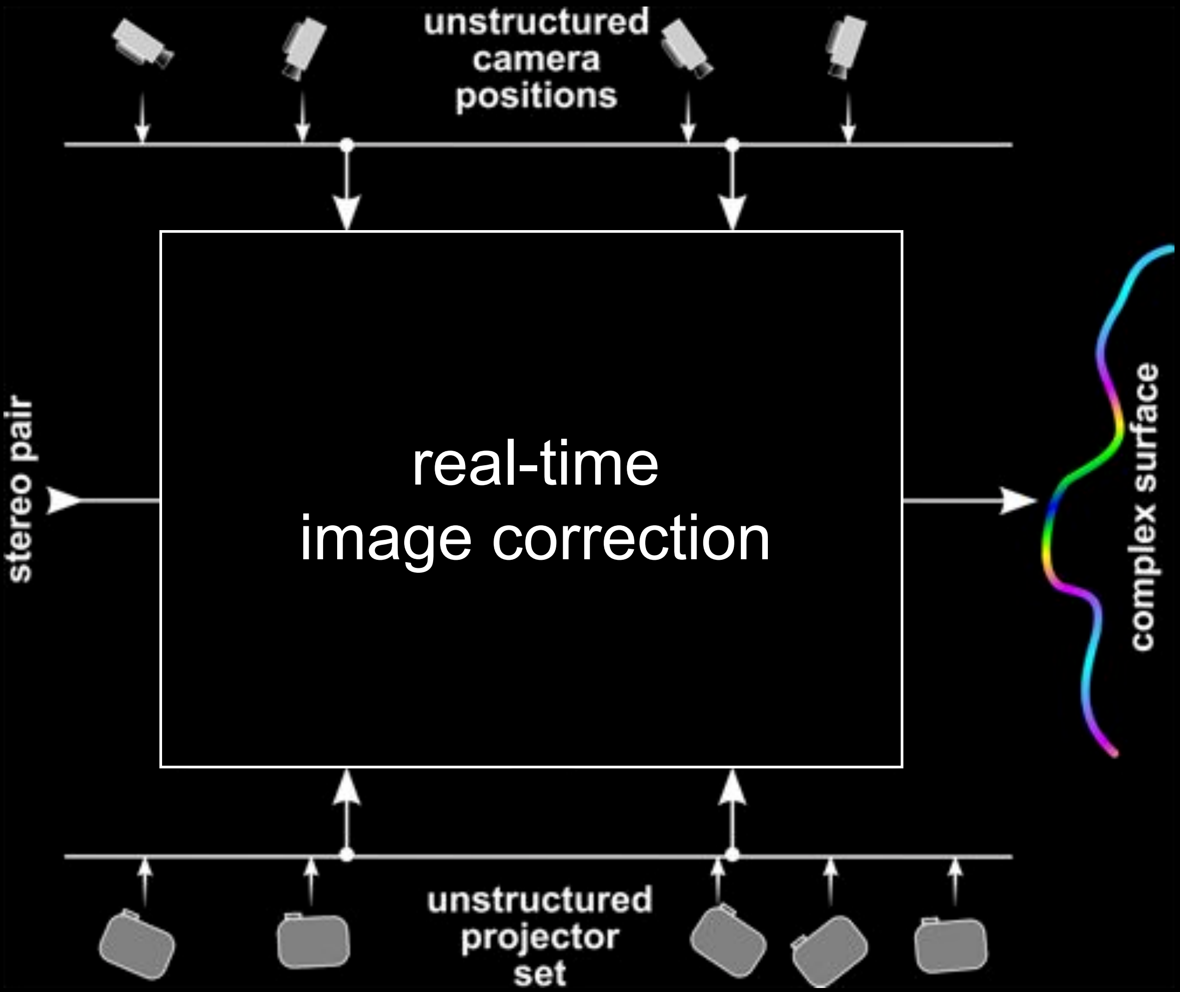

misregistration


regional defocus


scattering

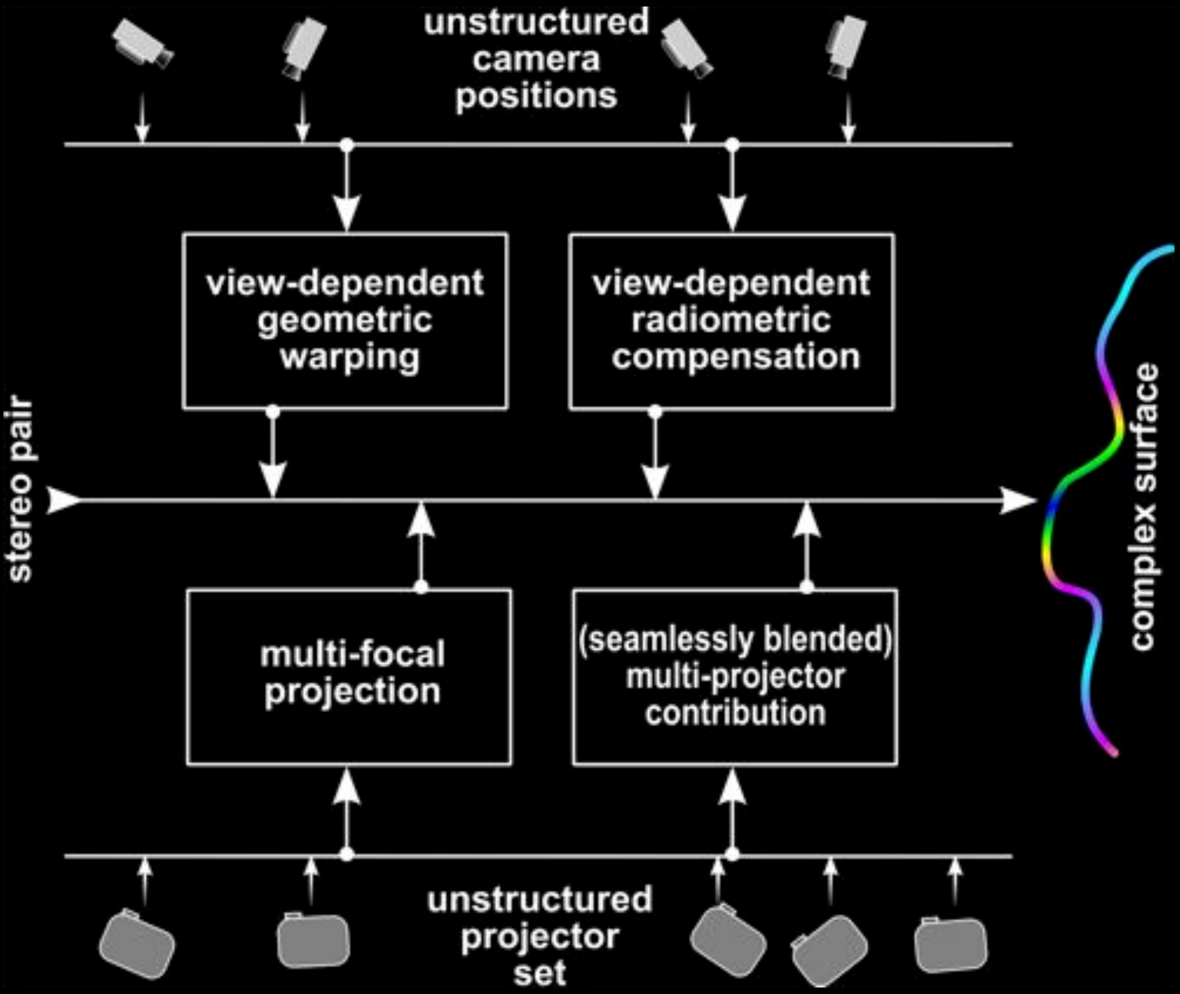

specular reflection, refraction, sub-surface scattering, inter-reflections, dispersion, diffraction, etc.

A Multi-Projector-Camera Approach



real-time
image correction

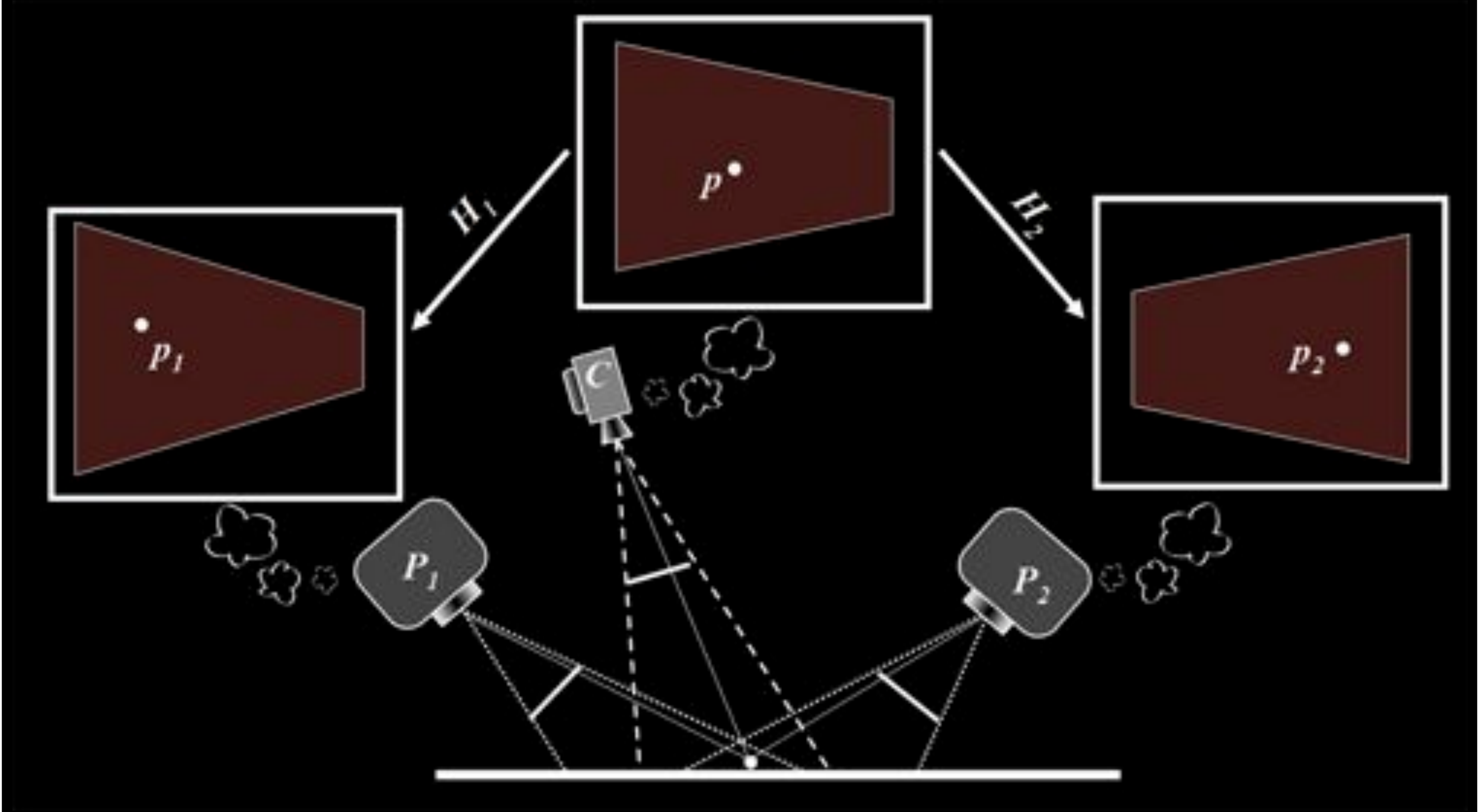
A Multi-Projector-Camera Approach



Geometric Correction

Planar Surfaces

Planar Surfaces



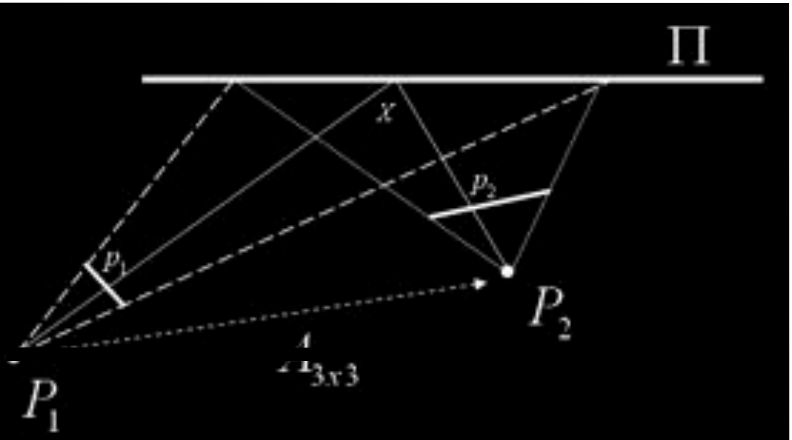
Homography

Homography

- homography is a mapping between two projections over a plane
- can map pixel coordinates from one perspective to another
- equation system has to be solved to determine 8 parameters of matrix A
- can be used directly in transformation pipeline by multiplying the following matrix after projection (without perspective division):

$$A_{4x4} = \begin{bmatrix} a_{11} & a_{12} & 0 & a_{13} \\ a_{21} & a_{22} & 0 & a_{23} \\ 0 & 0 & 1 & 0 \\ a_{31} & a_{32} & 0 & 1 \end{bmatrix}$$

- ensure intact depth values with (approximately)



$$p_2 \cong A_{3x3} p_1$$

$$\begin{bmatrix} p_{2x} \\ p_{2y} \\ 1 \end{bmatrix} \cong \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{bmatrix} \begin{bmatrix} p_{1x} \\ p_{1y} \\ 1 \end{bmatrix}$$

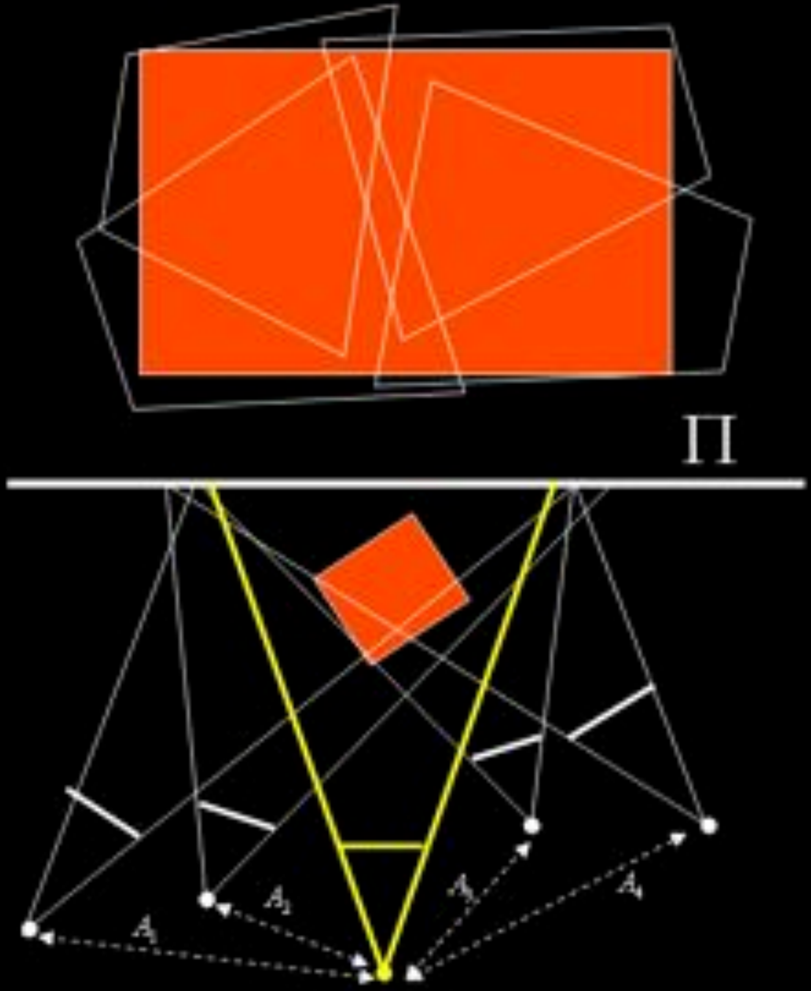
$$p_{2z} = \frac{p_{1z}}{(a_{31} p_{1x} + a_{32} p_{1y} + p_{1w})} \in [-1,1]$$

$$A_{4x4} = \begin{bmatrix} a_{11} & a_{12} & 0 & a_{13} \\ a_{21} & a_{22} & 0 & a_{23} \\ 0 & 0 & 1-|a_{31}|-|a_{32}| & 0 \\ a_{31} & a_{32} & 0 & 1 \end{bmatrix}$$

Multi-Projector Registration

Multi-Projector Registration

- registering multiple projectors onto a common planar surface
- map all perspective into a single target perspective via homographies
- target perspective can be camera perspective
    - automatic determination of matrix parameters via structured light
- rendering
    - render image for target perspective (if target perspective is **orthogonal** to plane, then it can be done with an off-axis projection of an observer!)
    - map pixels into individual projector views (i.e., multiply 4x4 version of homography matrix onto matrix stack [after projection] and ensure that depth values remain intact!)



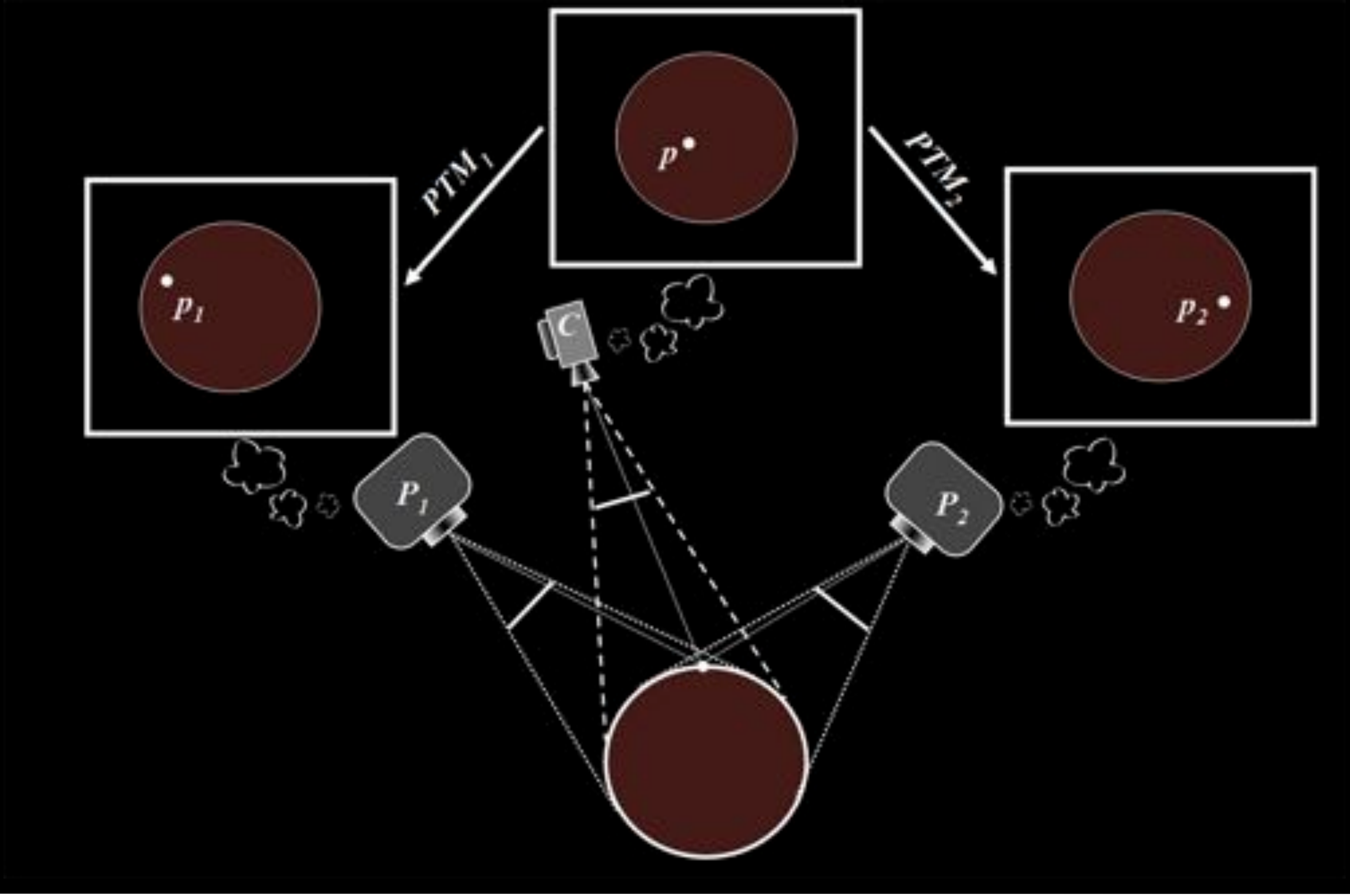
Example: Tiled Projection Screens

Example: Tiled Projection Screens

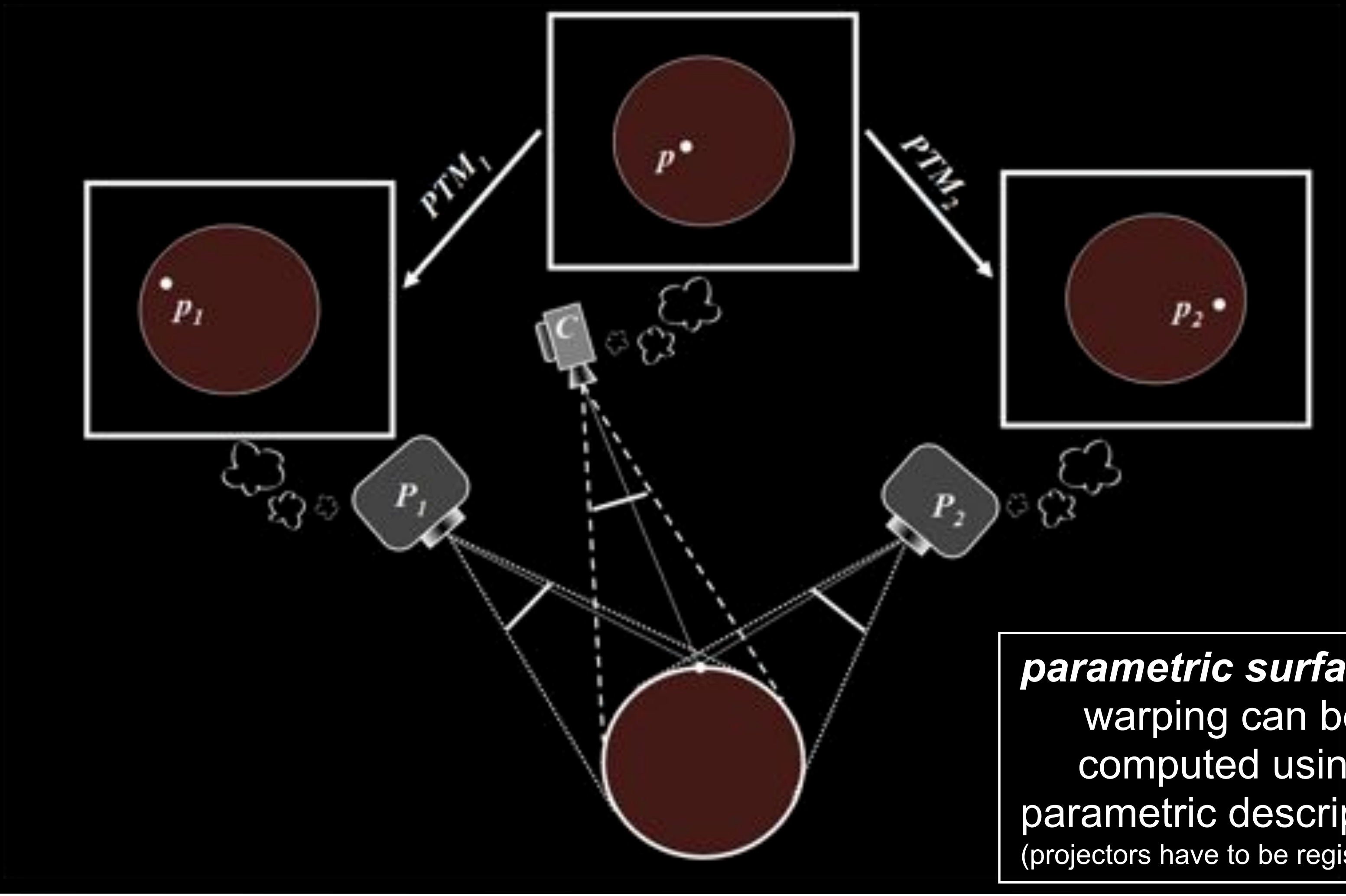
Courtesy: Brown, et al.,
IEEE TVCG, 2005

Non-Trivial Surfaces

Non-Trivial Surfaces



Non-Trivial Surfaces

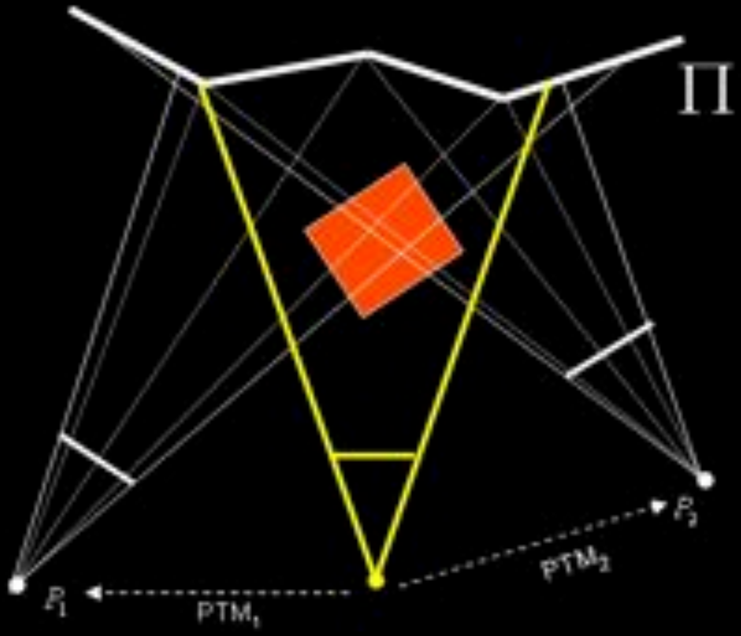


parametric surfaces:
warping can be
computed using
parametric description
(projectors have to be registered)

Projective Texture Mapping

Projective Texture Mapping

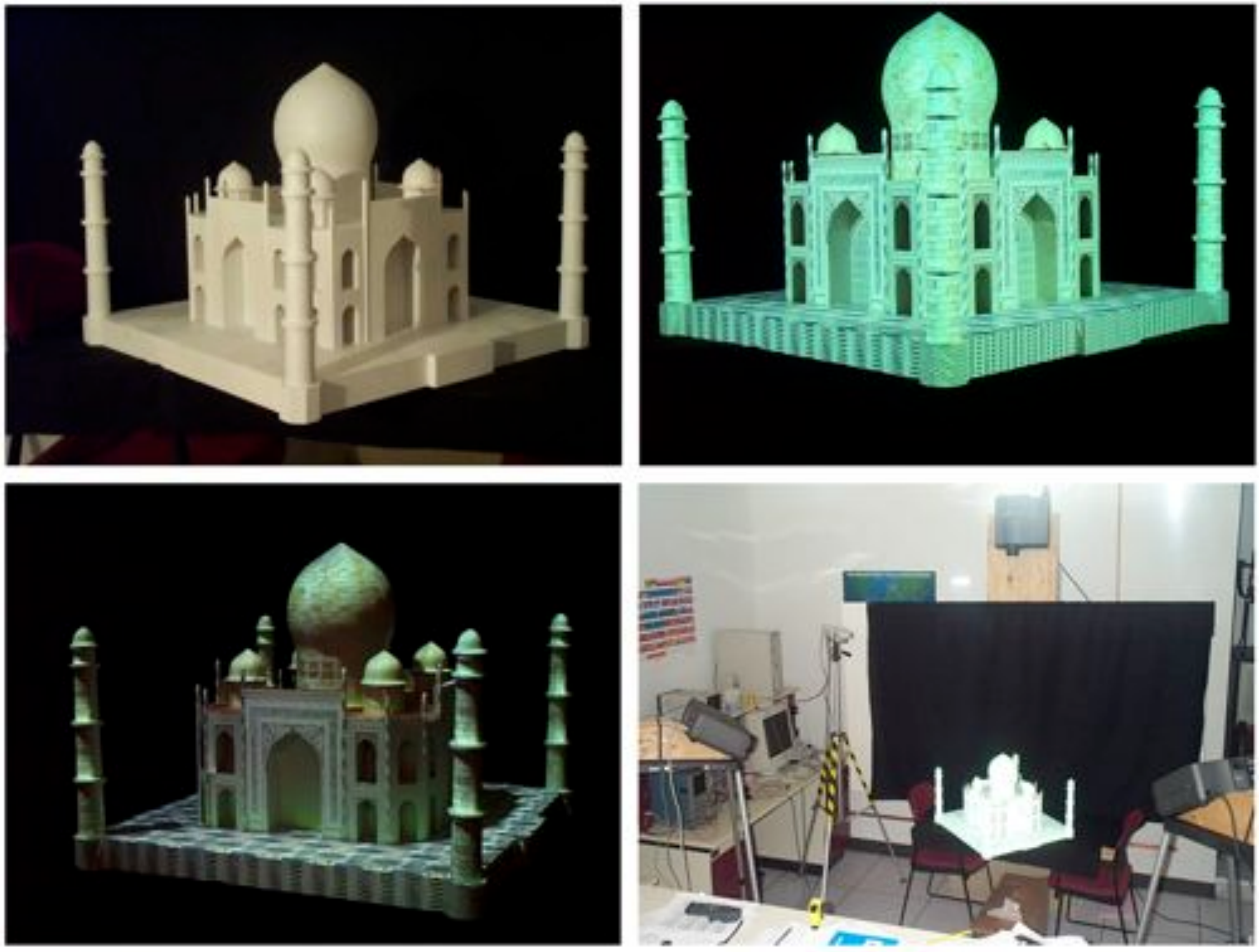
- given a geometric definition of the surface
    - scan or model
- determine intrinsic and extrinsic of projectors with respect to surface
    - measure projections of known 3D surface points on image plane of projector and solve equation system to determine parameters of matrix
- define virtual camera with same parameter for each projector
- render 3D model of surface, textured with images, from perspective of projectors/ virtual cameras
- texture coordinates can be automatically generated from target perspective via projective texture mapping



$$\begin{pmatrix} wx \\ wy \\ wz \\ w \end{pmatrix} = \begin{bmatrix} f & \cdot & \cdot & \cdot \\ \cdot & f & \cdot & \cdot \\ \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & 1 & \cdot \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_x \\ R_{21} & R_{22} & R_{23} & t_y \\ R_{31} & R_{32} & R_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Example: Shader Lamps

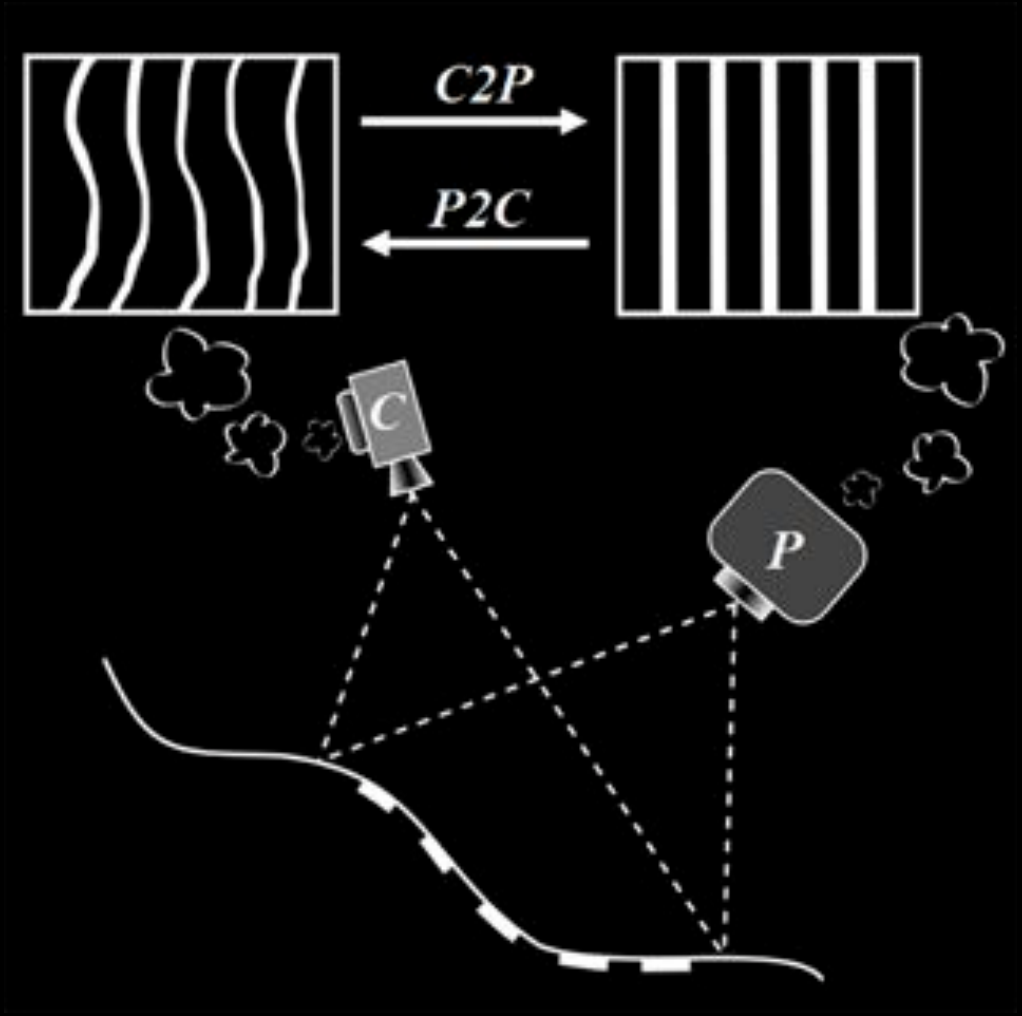
Example: Shader Lamps



Courtesy: Raskar, et al., EGRW 2001

Complex Surfaces

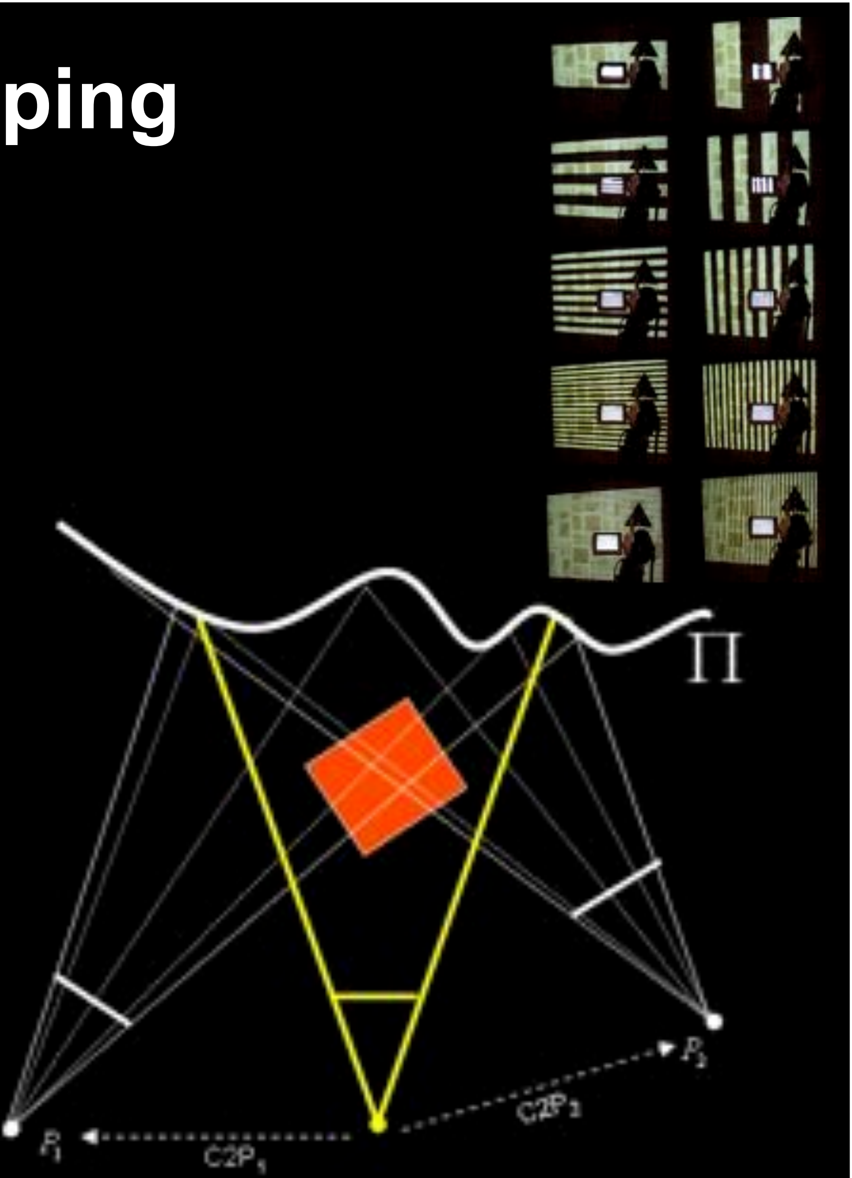
Complex Surfaces



Pixel Displacement Mapping

Pixel Displacement Mapping

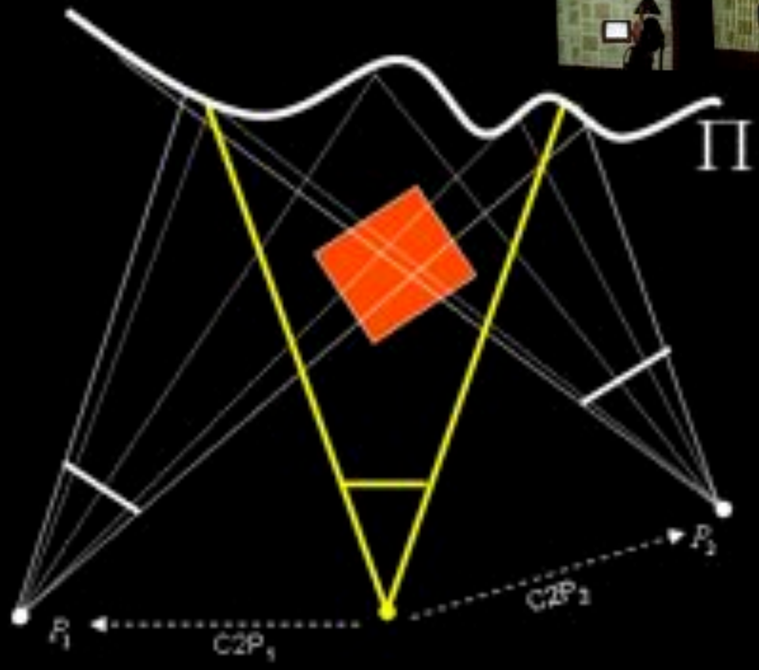
- registering projections to such a surface by determining their intrinsic and extrinsic is too imprecise
  - non-linear lens distortion
  - errors in measuring fiducials
- rendering of 3D surface representation from perspective of projector might be to slow
  - high geometric complexity of model
  - many triangles to render
    - project, raster, texture
- measure per-pixel mapping between projector perspectives and target perspective (e.g., camera)
- render image from target perspective and map it (look-up) into perspective of projectors (e.g., pixel-shading)



Pixel Displacement Mapping

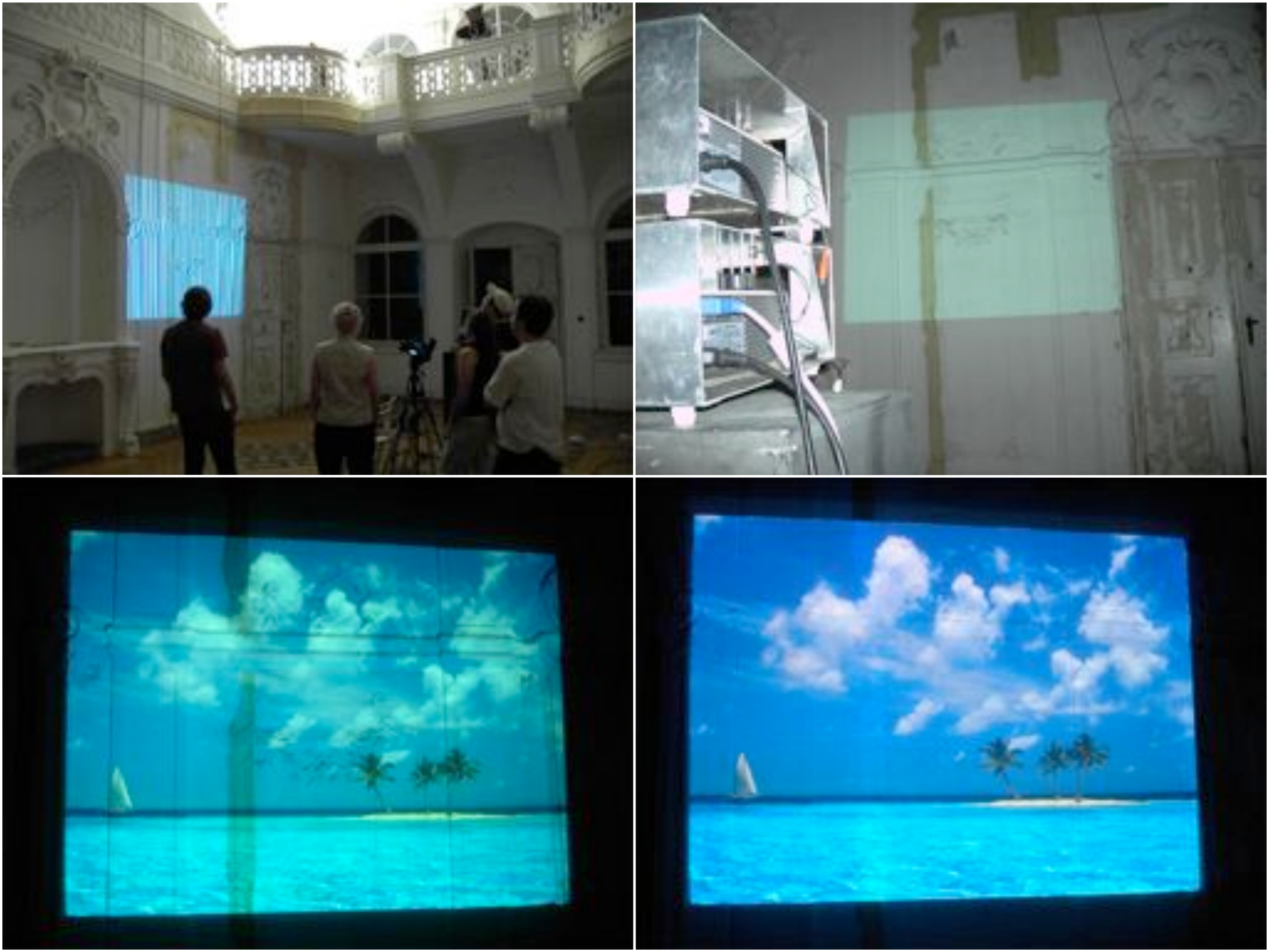
- registering projections to such a surface by determining their intrinsic and extrinsic is too imprecise
  - non-linear lens distortion
  - errors in measuring fiducials
- rendering of 3D surface representation from perspective of projector might be to slow
  - high geometric complexity of model
  - many triangles to render
    - project, raster, texture
- measure per-pixel mapping between projector perspectives and target perspective (e.g., camera)
- render image from target perspective and map it (look-up) into perspective of projectors (e.g., pixel-shading)

problem:
works only for static target perspective!
(but image-based rendering approaches exist)



Example: Stucco Wall

Example: Stucco Wall



In coop. with castle Ettersburg

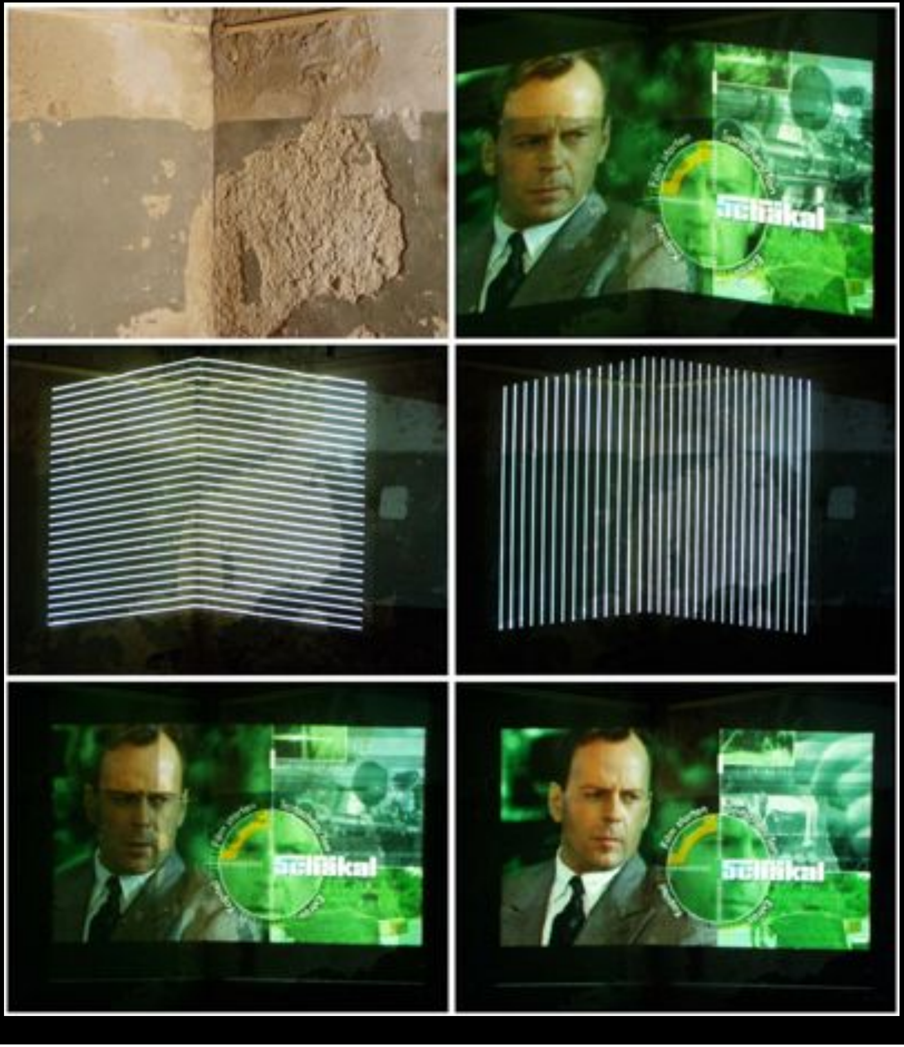
Example: Fossil Cast

Example: Fossil Cast

In coop. with Senckenberg Museum

Example: Scruffy Room Corner

Example: Scruffy Room Corner



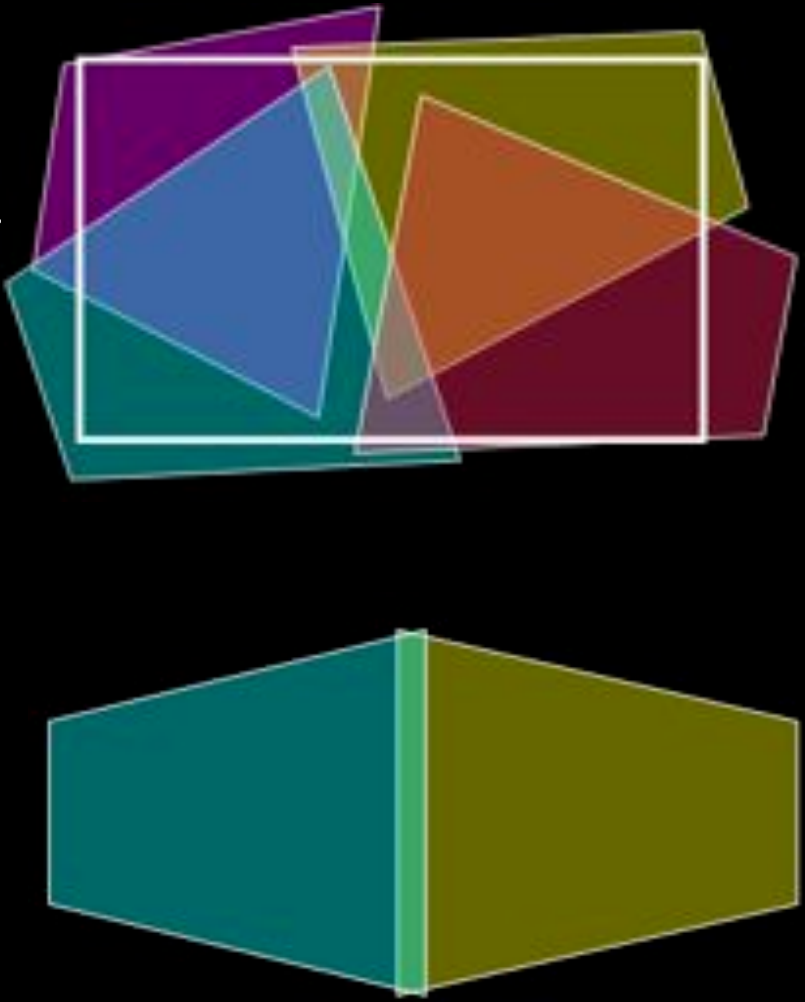
Bimber et al.,
IEEE Computer
2005

Radiometric Compensation

Photometric Calibration

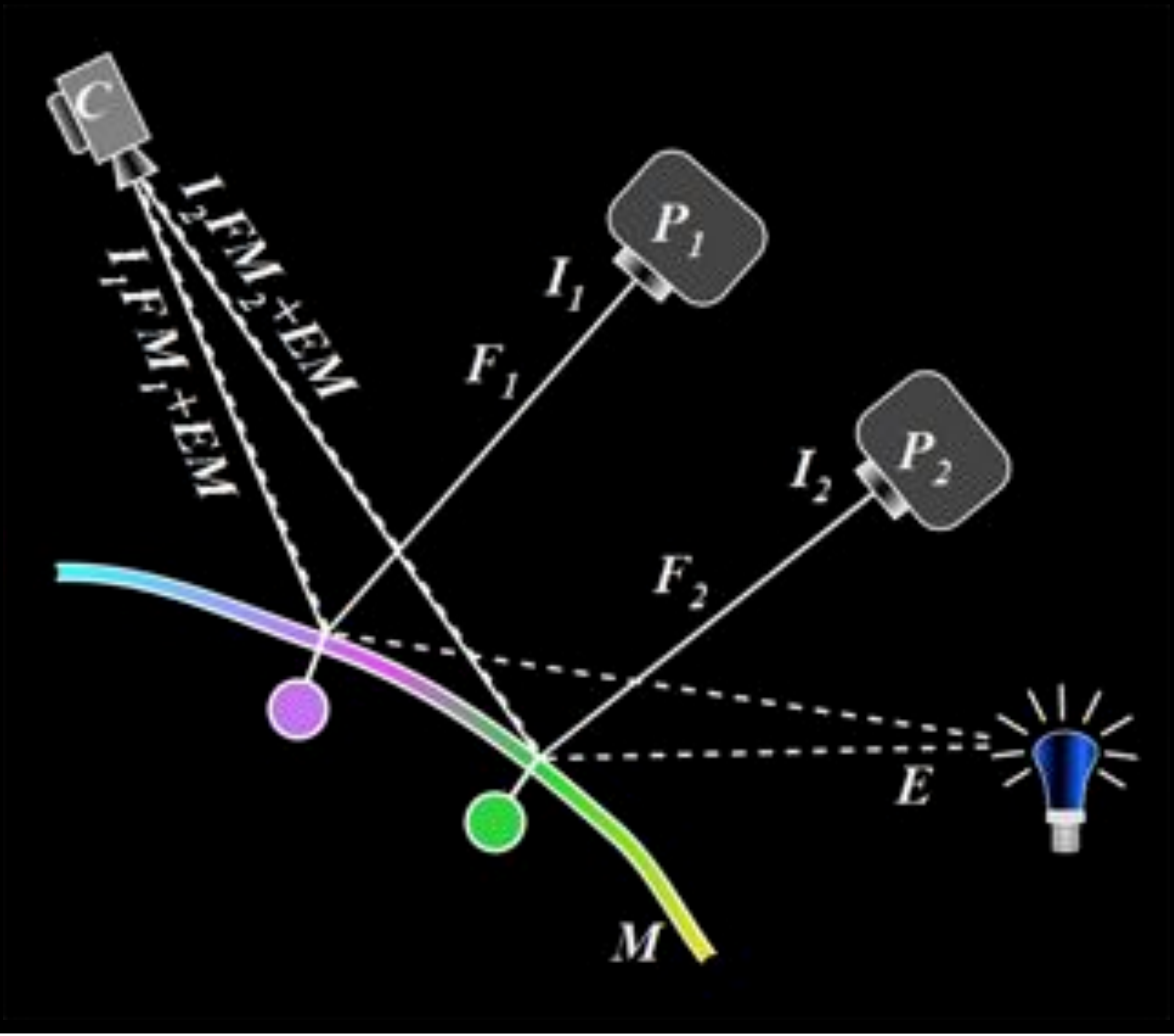
Photometric Calibration

- regions of display surfaces that are illuminated by multiple projectors simultaneously appear brighter
- projectors can have different brightness and can cover a different color space
- result: inconsistent image (intensity and color)
- humans can perceive 2% difference in brightness and a color variation of 2*nm*
- variations in brightness is more critical than variation in color
- solutions: intensity blending and color space mapping
- **these techniques are not explained here!**
- **we assume that projectors and cameras are linearized and color mapped**



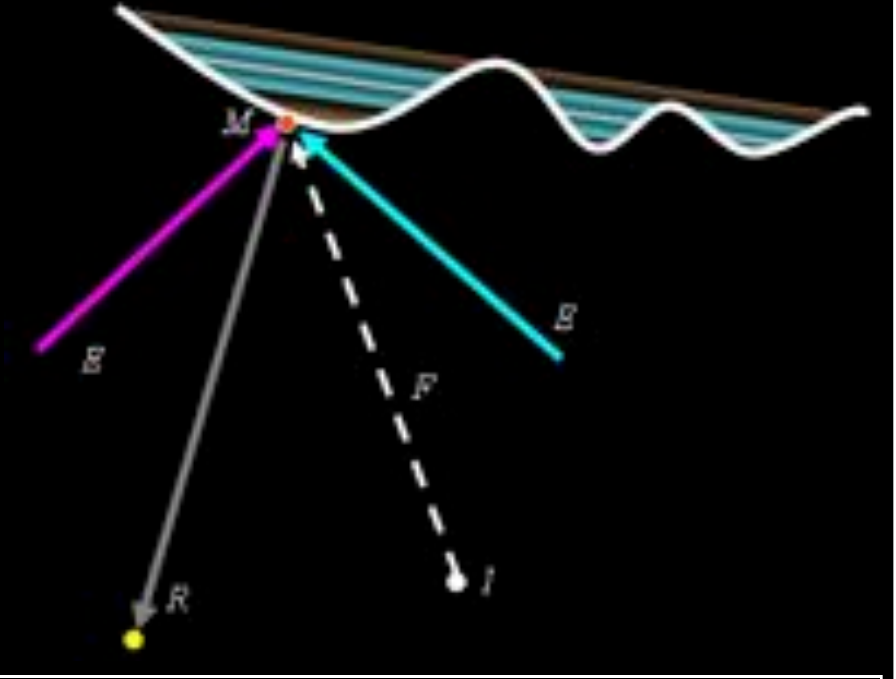
Compensating Local Light Effects

Compensating Local Light Effects



Single Projector

Single Projector

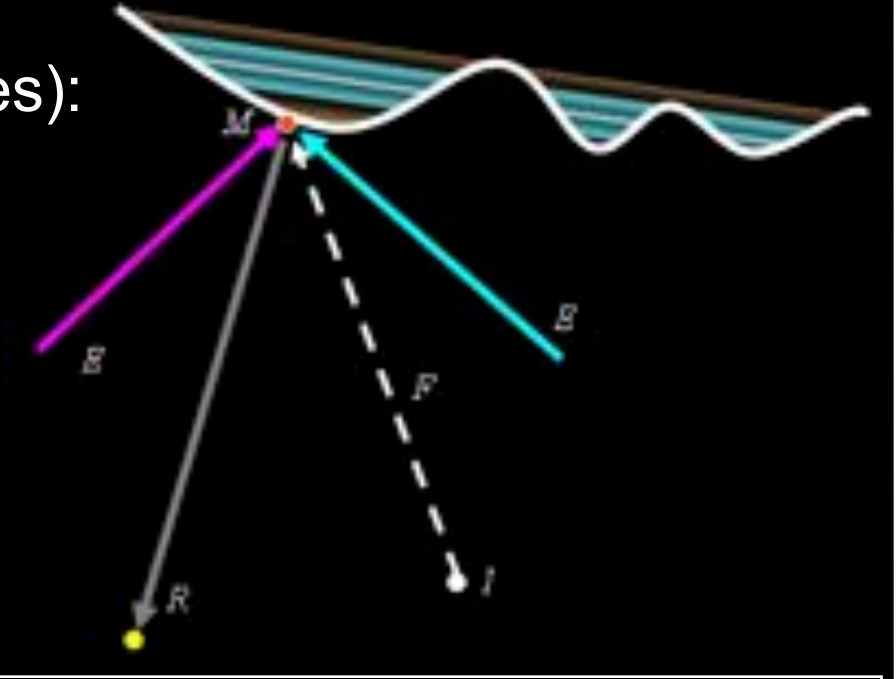


R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):



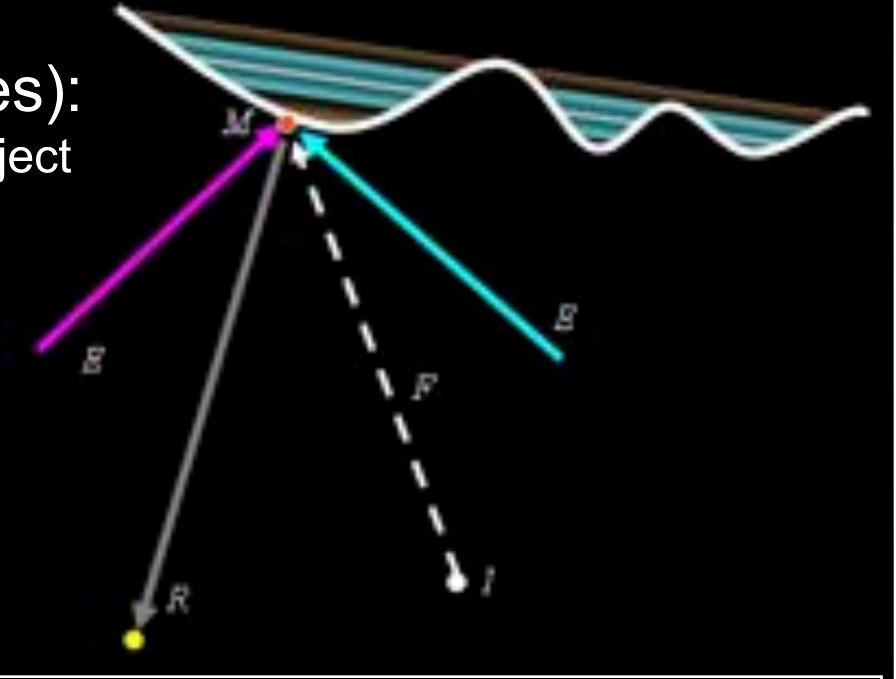
R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image



R=IFM+EM

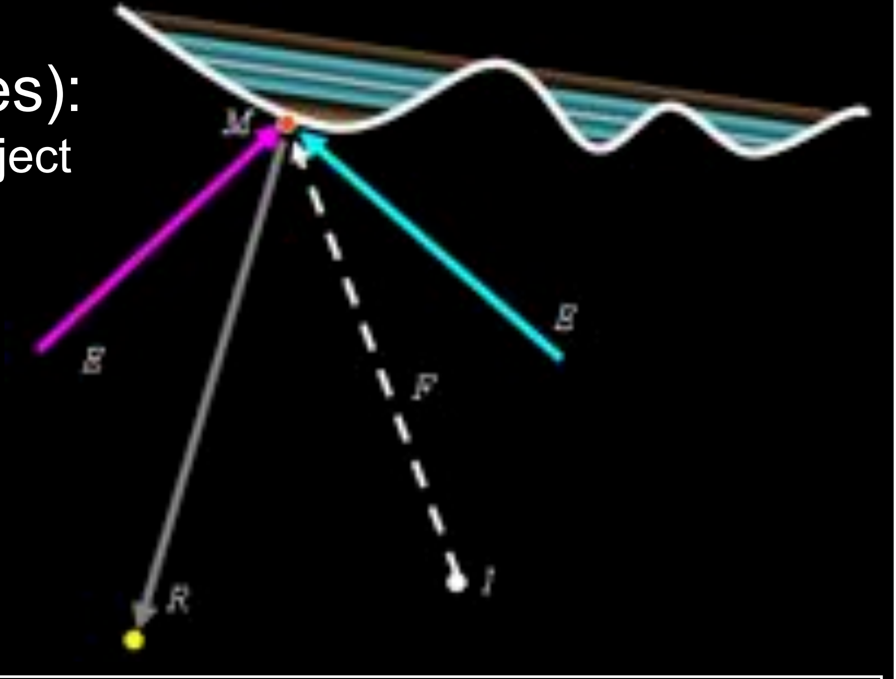
I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

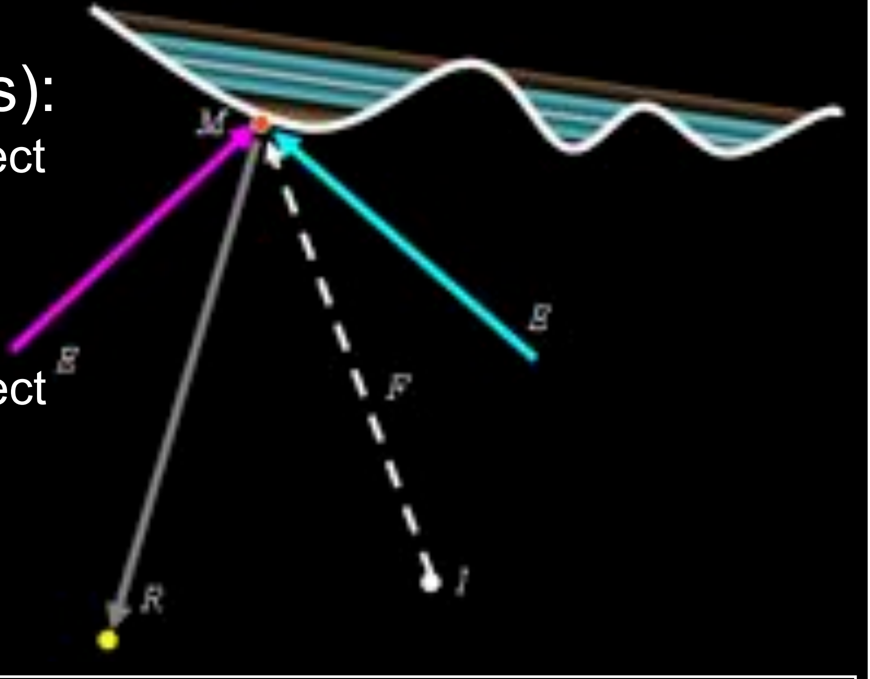
Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM

(2) turn on environment light and project black flood image



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

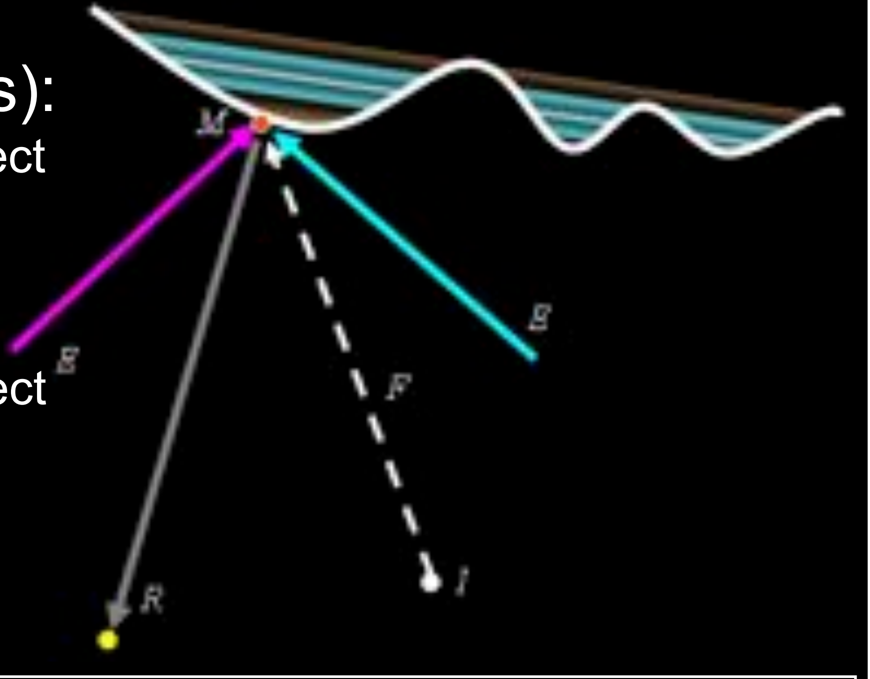
determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM

(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

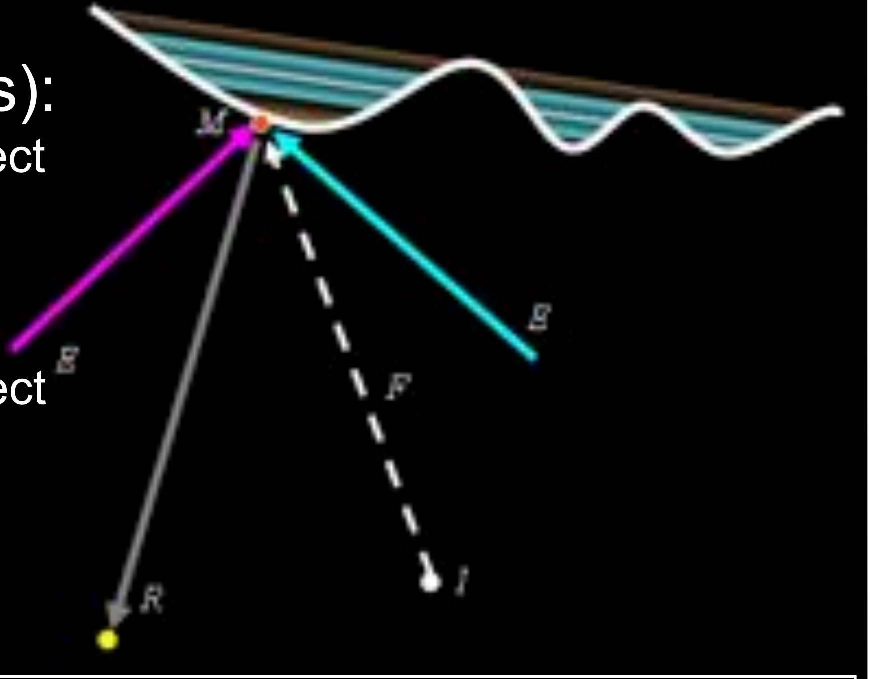
(1) turn off environment light and project black flood image

I=0,E=0 → BFM

(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

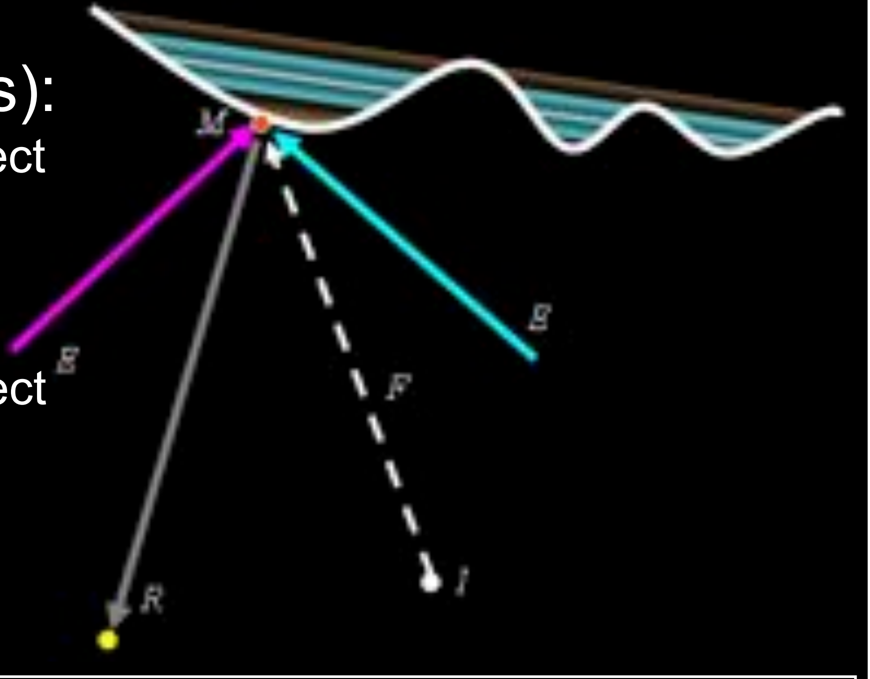
(1) turn off environment light and project black flood image

I=0,E=0 → BFM

(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and project white flood image



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

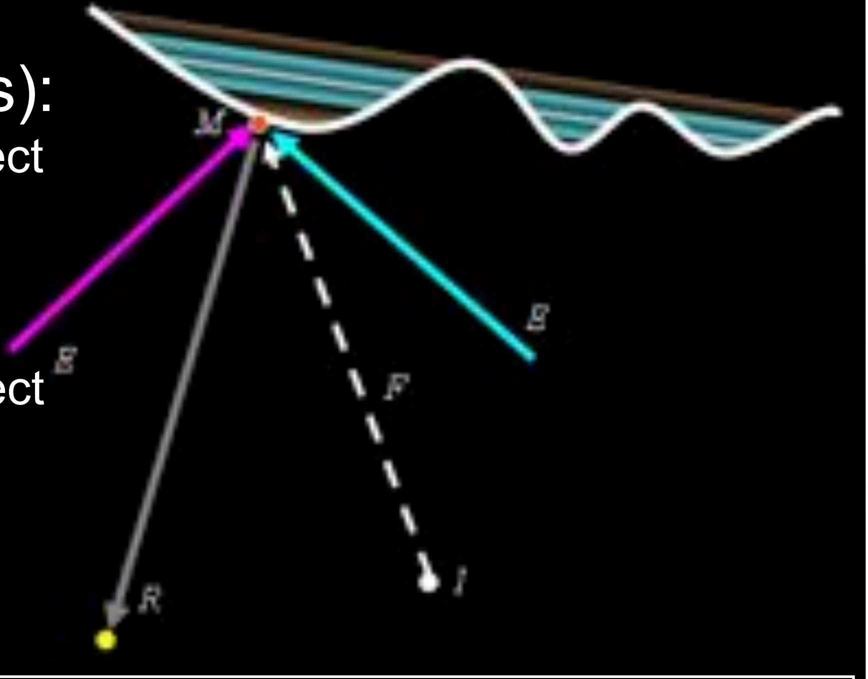
I=0,E=0 → BFM

(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and project white flood image

I=1,E=0 → FM (incl. **BFM !**)



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM

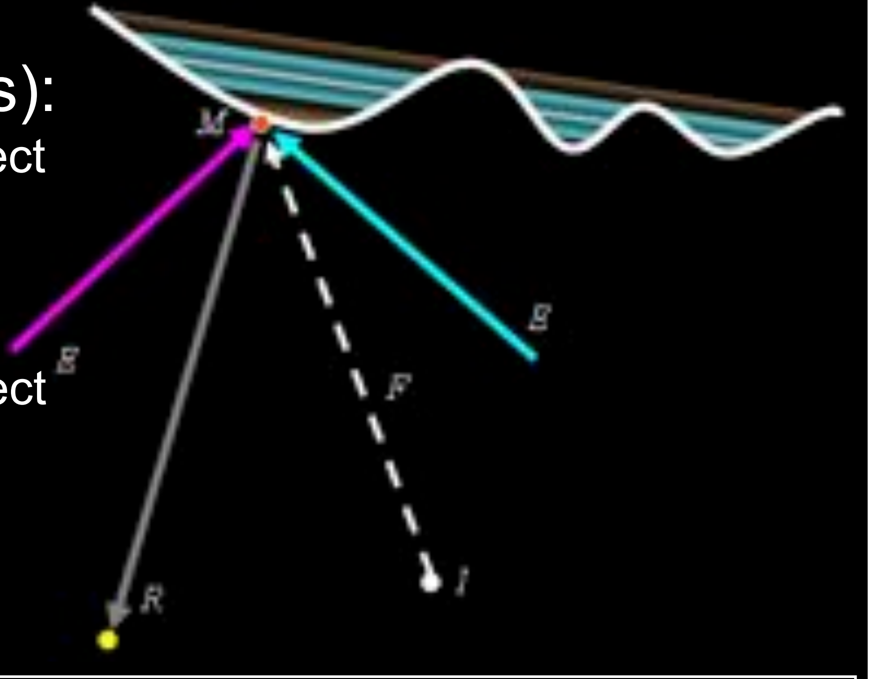
(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and project white flood image

I=1,E=0 → FM (incl. **BFM !**)

→ FM=FM-BFM



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM

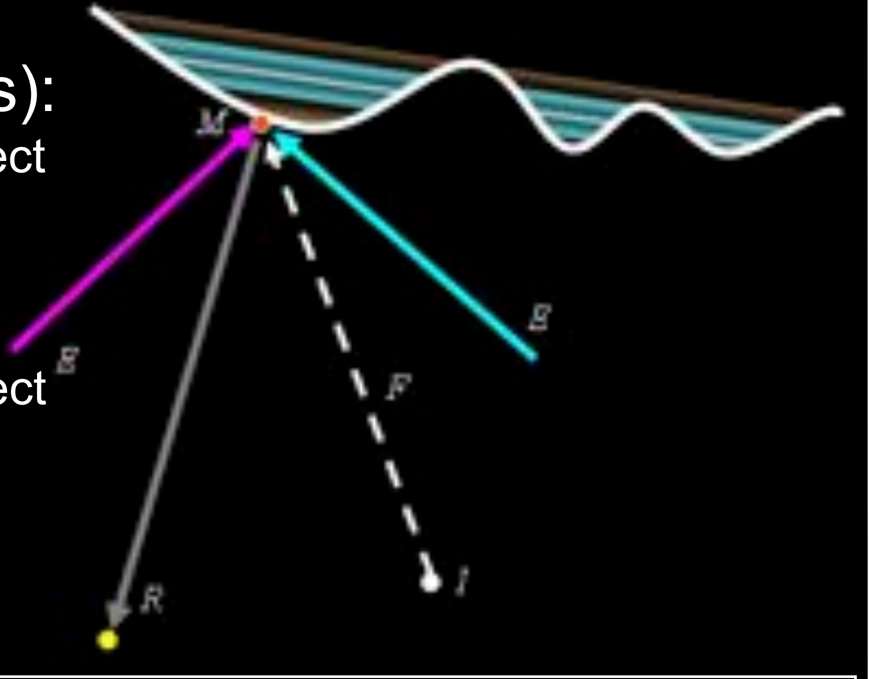
(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and project white flood image

I=1,E=0 → FM (incl. **BFM !**)

→ FM=FM-BFM



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

compensation (per pixel):

Bimber et al, IEEE Computer 2005

Single Projector

determining parameters (textures):

(1) turn off environment light and project black flood image

I=0,E=0 → BFM

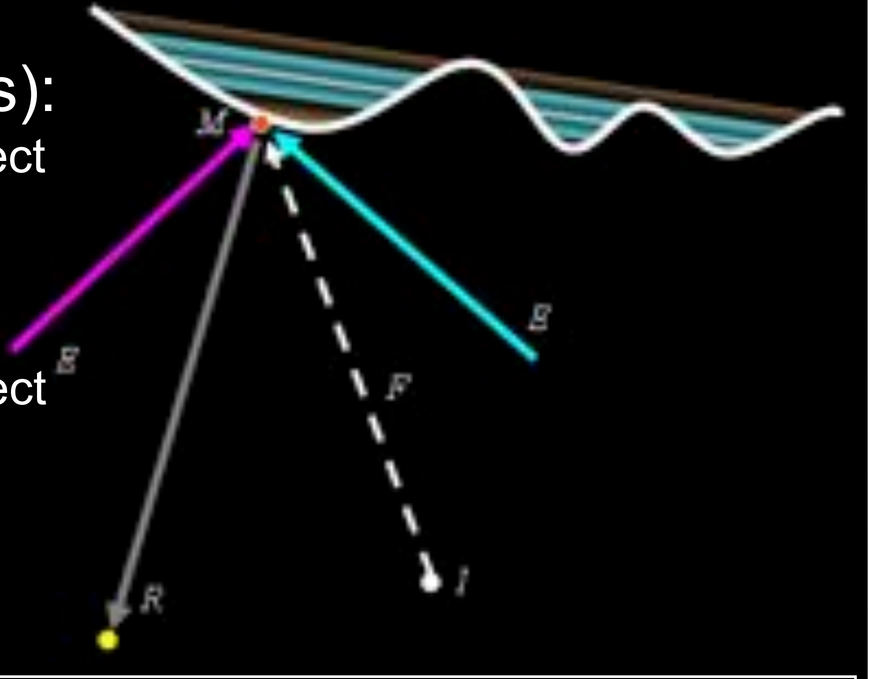
(2) turn on environment light and project black flood image

I=0,E=1 → EM (incl. **BFM !**)

(3) turn off environment light and project white flood image

I=1,E=0 → FM (incl. **BFM !**)

→ FM=FM-BFM



R=IFM+EM

I → projected image
B → black-level
F → projector-2-surface form factor
E → environment light
M → surface reflectance (diffuse)

compensation (per pixel):

I=(R-EM)/(FM)

Color Mixing

Color Mixing

Nayar et al, CVPR 2004

$$\begin{array}{cc} & \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ \downarrow & \downarrow & \downarrow \end{array} \\ \underset{\text{(per pixel)}}{V} = & \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix} \end{array}$$

R=V*I

I → projected image
V → color mixing matrix
(projector/camera/reflectance)

Color Mixing

Nayar et al, CVPR 2004

$$V = \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ \downarrow & \downarrow & \downarrow \end{array}$$

$$\underset{\text{(per pixel)}}{V} = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

→ FM (in un-normalized matrix)

R=V*I

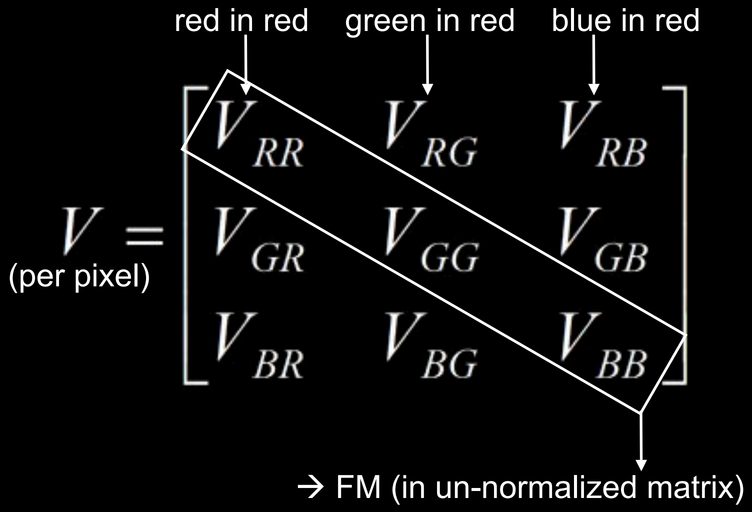
I → projected image

V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:



$$V = \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{array}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

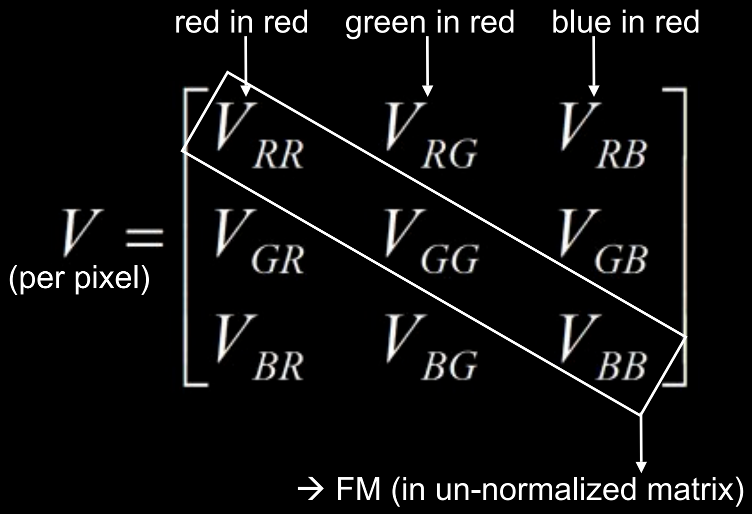
I → projected image
V → color mixing matrix
(projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):



red in red green in red blue in red

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image
V → color mixing matrix (projector/camera/reflectance)

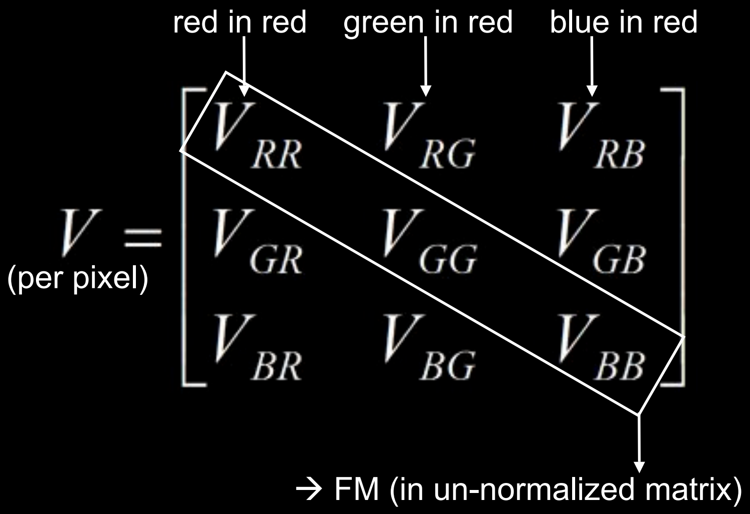
Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares



$$V = \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ \end{array}$$

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image
V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

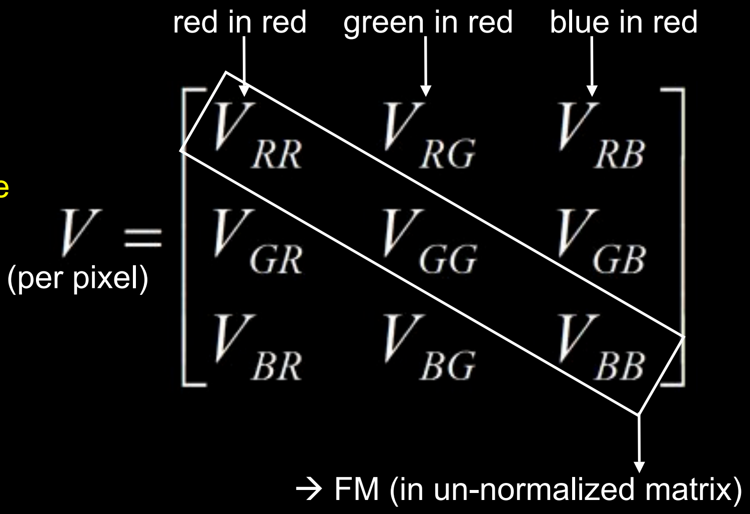
Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

for normalized matrix (camera response must be known, projector response can be unknown):



$$\begin{array}{cccc} & \text{red in red} & \text{green in red} & \text{blue in red} \\ V = & V_{RR} & V_{RG} & V_{RB} \\ & V_{GR} & V_{GG} & V_{GB} \\ & V_{BR} & V_{BG} & V_{BB} \end{array}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image
V → color mixing matrix
(projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

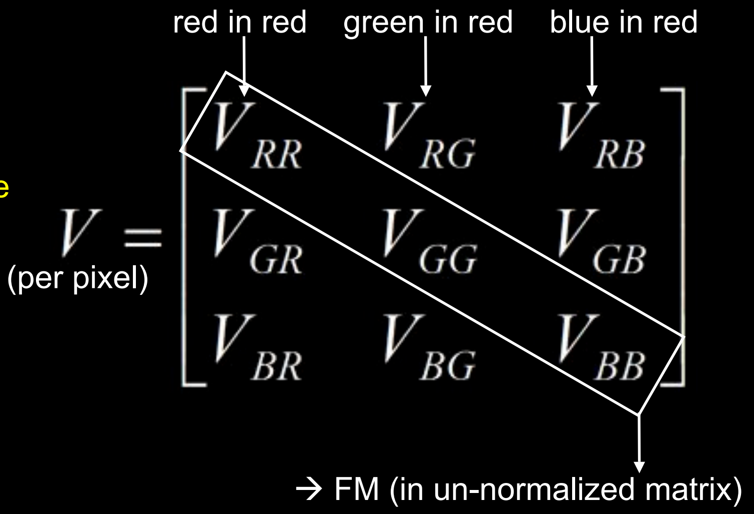
determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)



$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

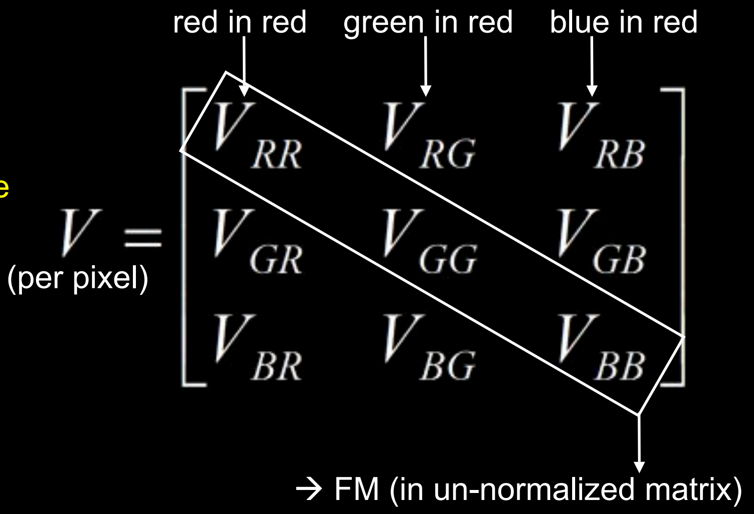
I → projected image
V → color mixing matrix
(projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

- for un-normalized matrix: (camera and projector response must be known and linearized):
  - capture 9+ images → least squares
- for normalized matrix (camera response must be known, projector response can be unknown):
  - diagonals are 1 (unknown scaling)
  - off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\mathbf{\Delta C_j/\Delta R_i}$



$$V = \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ \downarrow & \downarrow & \downarrow \end{array}$$

$$\underset{\text{(per pixel)}}{V} = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

→ FM (in un-normalized matrix)

R=V*I

I → projected image
V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

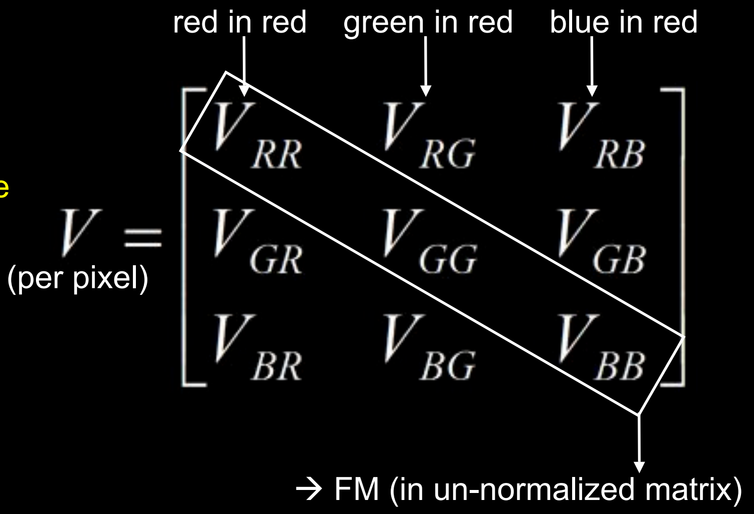
capture 9+ images → least squares

for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)

off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\Delta \mathbf{C_j}/\Delta \mathbf{R_i}$

(since $\mathbf{V_{ii}}=1$, $\Delta \mathbf{I_i}=\Delta \mathbf{C_i}$)



red in red green in red blue in red

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image
V → color mixing matrix (projector/camera/reflectance)

Color Mixing

Nayar et al, CVPR 2004

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

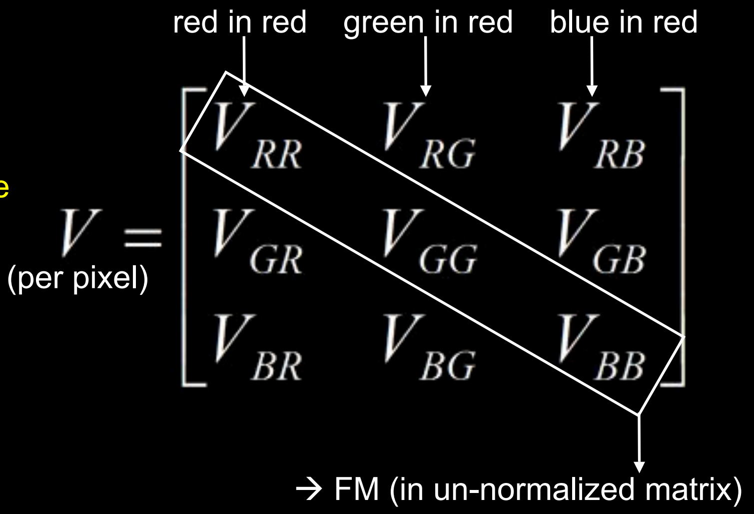
for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)

off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\Delta C_j/\Delta R_i$

(since $\mathbf{V_{ii}}=1$, $\Delta\mathbf{I_i}=\Delta\mathbf{C_i}$)

capture 6 images **C** (2 per color channel



$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

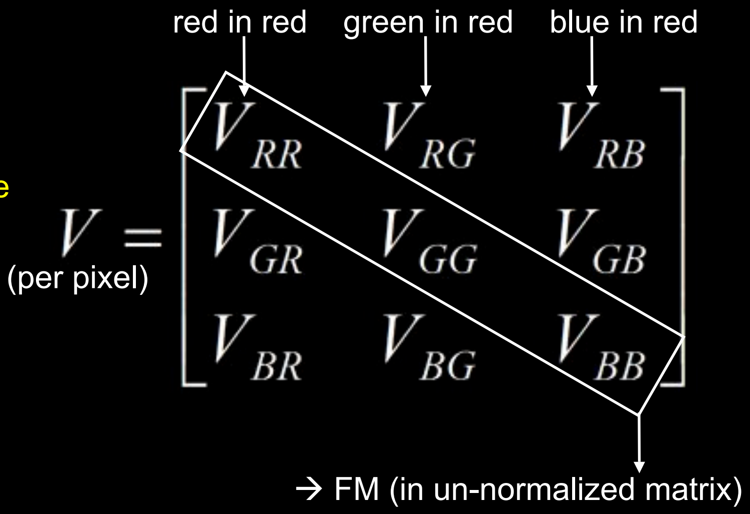
I → projected image
V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

- for un-normalized matrix: (camera and projector response must be known and linearized):
  - capture 9+ images → least squares
- for normalized matrix (camera response must be known, projector response can be unknown):
  - diagonals are 1 (unknown scaling)
  - off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\mathbf{\Delta C_j/\Delta R_i}$
    (since $\mathbf{V_{ii}}=1$, $\mathbf{\Delta I_i=\Delta C_i}$)
  - capture 6 images **C** (2 per color channel to determine deltas)



$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

R=V*I

I → projected image

V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

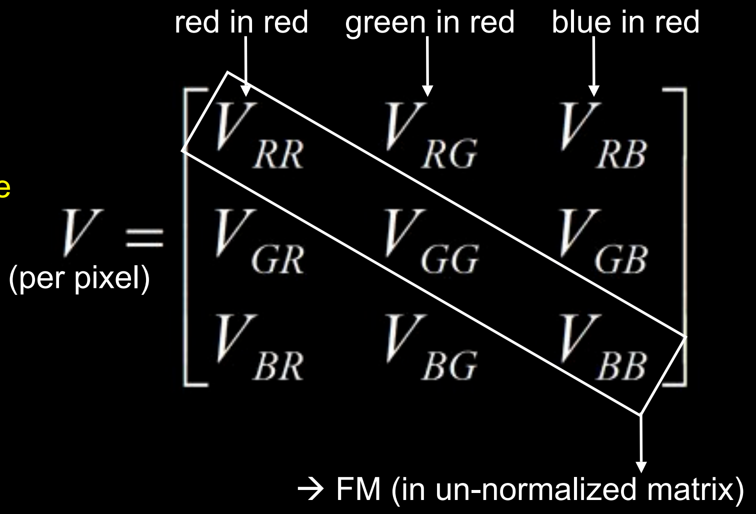
for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)

off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\Delta \mathbf{C_j/\Delta R_i}$

(since $\mathbf{V_{ii}}=1$, $\Delta \mathbf{I_i}=\Delta \mathbf{C_i}$)

capture 6 images **C** (2 per color channel to determine deltas)



red in red
green in red
blue in red

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

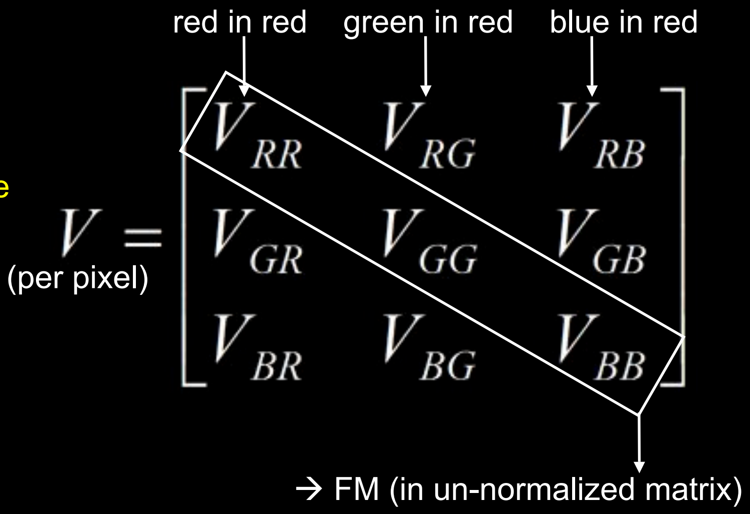
I → projected image
V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

- for un-normalized matrix: (camera and projector response must be known and linearized):
  - capture 9+ images → least squares
- for normalized matrix (camera response must be known, projector response can be unknown):
  - diagonals are 1 (unknown scaling)
  - off-diagonals are $\mathbf{V}_{ij}=\Delta\mathbf{C}_j/\Delta\mathbf{I}_i=\Delta\mathbf{C}_j/\Delta\mathbf{R}_i$ (since $\mathbf{V}_{ii}=1$, $\Delta\mathbf{I}_i=\Delta\mathbf{C}_i$)
  - capture 6 images **C** (2 per color channel to determine deltas)



red in red green in red blue in red

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

compensation (per pixel):

R=V*I

I → projected image
V → color mixing matrix (projector/camera/reflectance)

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

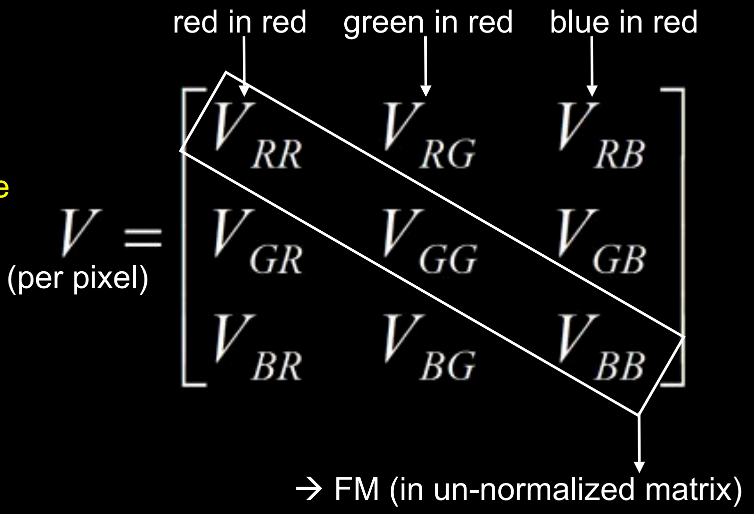
for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)

off-diagonals are $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}=\Delta C_j/\Delta R_i$

(since $\mathbf{V_{ii}}=1$, $\Delta \mathbf{I_i}=\Delta \mathbf{C_i}$)

capture 6 images **C** (2 per color channel to determine deltas)



red in red

green in red

blue in red

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image

V → color mixing matrix (projector/camera/reflectance)

compensation (per pixel):

$\mathbf{I=V^{-1}*R}$ (does not consider

Nayar et al, CVPR 2004

Color Mixing

determining color mixing matrix V:

for un-normalized matrix: (camera and projector response must be known and linearized):

capture 9+ images → least squares

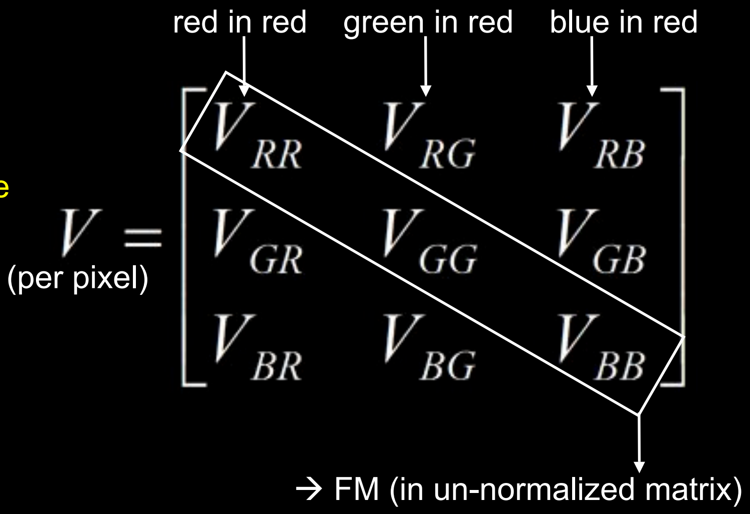
for normalized matrix (camera response must be known, projector response can be unknown):

diagonals are 1 (unknown scaling)

off-diagonals are $\mathbf{V_{ij}}=\Delta \mathbf{C_j}/\Delta \mathbf{I_i}=\Delta \mathbf{C_j}/\Delta \mathbf{R_i}$

(since $\mathbf{V_{ii}}=1$, $\Delta \mathbf{I_i}=\Delta \mathbf{C_i}$)

capture 6 images **C** (2 per color channel to determine deltas)



$$V = \begin{array}{ccc} \text{red in red} & \text{green in red} & \text{blue in red} \\ \left[\begin{array}{c} V_{RR} \\ V_{GR} \\ V_{BR} \end{array} \right. & \begin{array}{c} V_{RG} \\ V_{GG} \\ V_{BG} \end{array} & \left. \begin{array}{c} V_{RB} \\ V_{GB} \\ V_{BB} \end{array} \right] \end{array}$$

(per pixel)

→ FM (in un-normalized matrix)

R=V*I

I → projected image

V → color mixing matrix (projector/camera/reflectance)

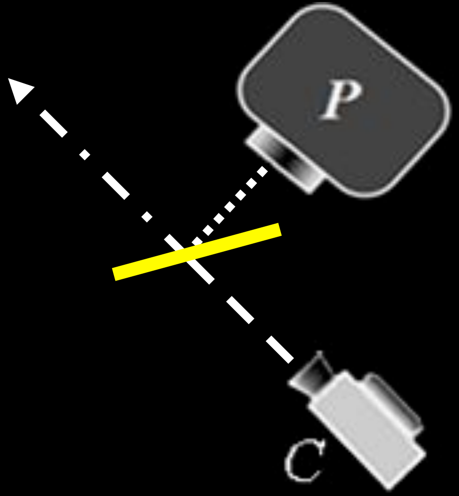
compensation (per pixel):

I=V⁻¹*R (does not consider environment light!)

Dynamic Adaptation

Dynamic Adaptation

Fujii et al, CVPR 2005



$R_t=M_t/M_0*(E_t*M_0+V_0*I_t)$

t → time index

I_t → projected image at t

V_0 → un-normalized color mixing matrix at t=0 (const.)

M_t → material at t

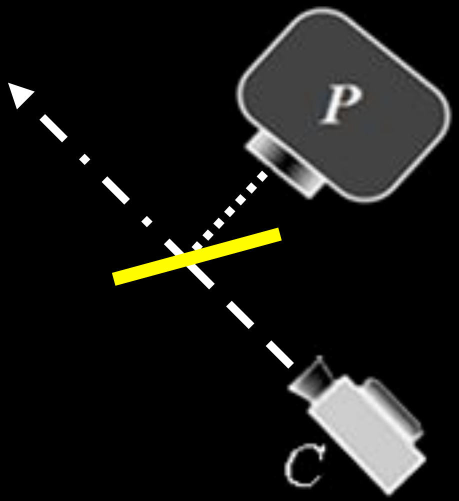
M_0 → material at t=0

E_t → environment light at t=0

Fujii et al, CVPR 2005

Dynamic Adaptation

determining color mixing matrix V_0:



$$R_t=M_t/M_0*(E_t*M_0+V_0*I_t)$$

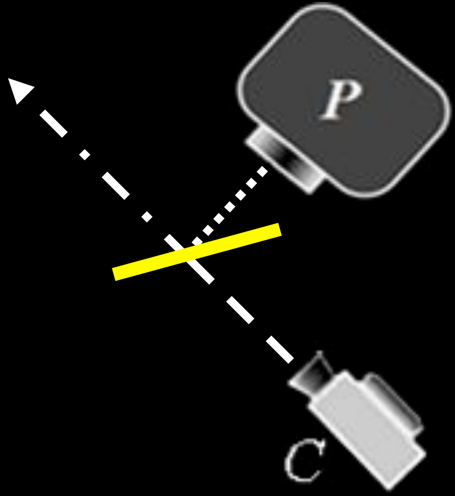
t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Fujii et al, CVPR 2005

Dynamic Adaptation

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$



$$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$$

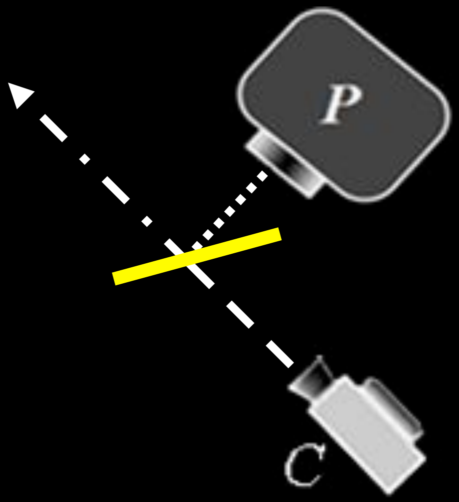
t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Dynamic Adaptation

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$

(un-normalized!)



Fujii et al, CVPR 2005

$$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$$

t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

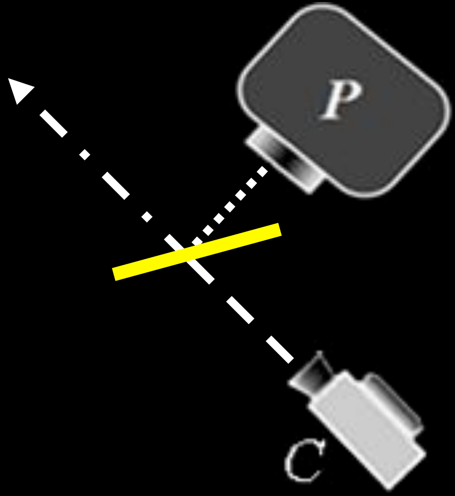
Fujii et al, CVPR 2005

Dynamic Adaptation

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$

(un-normalized!)



determine reflected environment

$$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$$

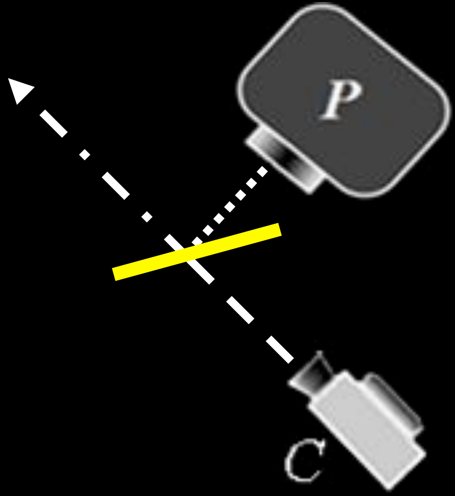
t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Fujii et al, CVPR 2005

Dynamic Adaptation

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$ (un-normalized!)



determine reflected environment light $E_0 * M_0$ at t=0:

$$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$$

t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

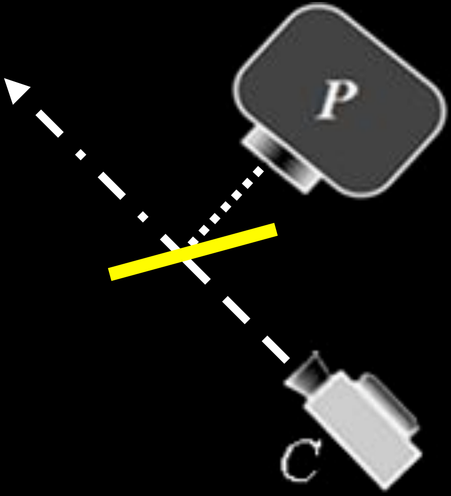
Fujii et al, CVPR 2005

Dynamic Adaptation

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$

(un-normalized!)



determine reflected environment light E_0*M_0 at t=0:

$\mathbf{E_0 * M_0 = C - V_0 * I}$ (project

$\mathbf{R_t = M_t / M_0 * (E_t * M_0 + V_0 * I_t)}$

t → time index

I_t → projected image at t

V_0 → un-normalized color mixing matrix at t=0 (const.)

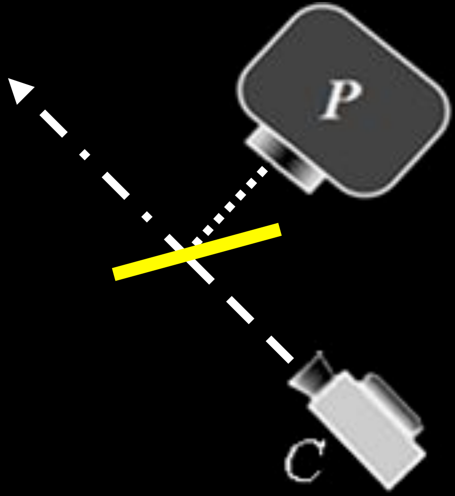
M_t → material at t

M_0 → material at t=0

E_t → environment light at t=0

Fujii et al, CVPR 2005

Dynamic Adaptation



determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$ (un-normalized!)

determine reflected environment light E_0*M_0 at t=0:

$\mathbf{E_0*M_0=C-V_0*I}$ (project arbitrary I and capture C)

$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$

t → time index

I_t → projected image at t

V_0 → un-normalized color mixing matrix at t=0 (const.)

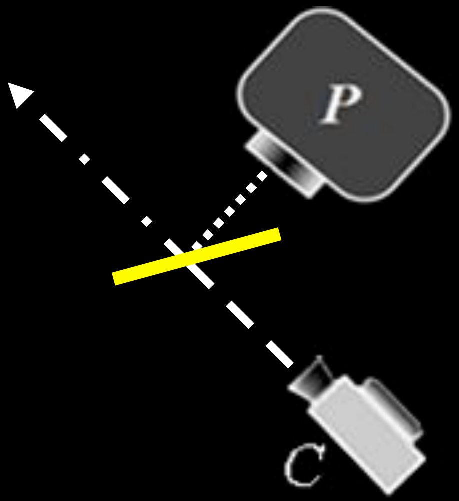
M_t → material at t

M_0 → material at t=0

E_t → environment light at t=0

Dynamic Adaptation

Fujii et al, CVPR 2005



determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$ (un-normalized!)

determine reflected environment light $E_0{}^*M_0$ at t=0:

$\mathbf{E_0{}^*M_0=C-V_0{}^*I}$ (project arbitrary I and capture C)

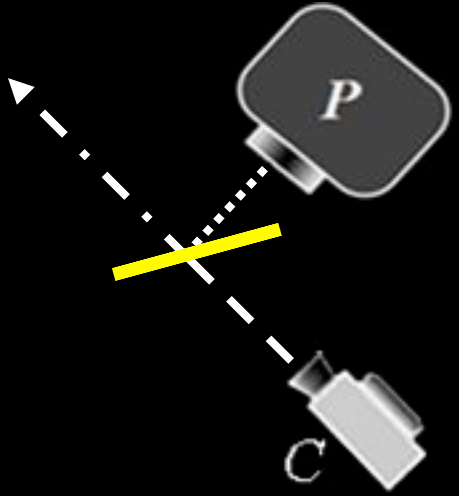
compensation (per pixel at t):

$$\mathbf{R_t=M_t/M_0{}^*(E_t{}^*M_0+V_0{}^*I_t)}$$

t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Fujii et al, CVPR 2005

Dynamic Adaptation



determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$ (un-normalized!)

determine reflected environment light $E_0{*}M_0$ at t=0:

$\mathbf{E_0{*}M_0=C-V_0{*}I}$ (project arbitrary I and capture C)

compensation (per pixel at t):

$\mathbf{I_t=V_0^{-1}{*}(R{*}M_0/M_{t-1}-E_{t-1}{*}M_0)}$

$\mathbf{R_t=M_t/M_0{*}(E_t{*}M_0+V_0{*}I_t)}$

t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

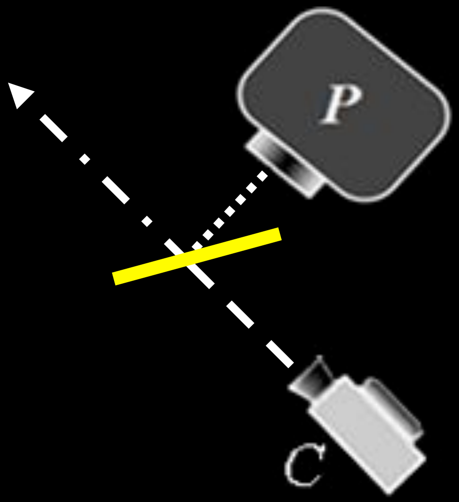
Dynamic Adaptation

Fujii et al, CVPR 2005

determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$

(un-normalized!)



determine reflected environment light E_0*M_0 at t=0:

$\mathbf{E_0*M_0=C-V_0*I}$ (project arbitrary I and capture C)

compensation (per pixel at t):

$\mathbf{I_t=V_0^{-1}*(R*M_0/M_{t-1}-E_{t-1}*M_0)}$

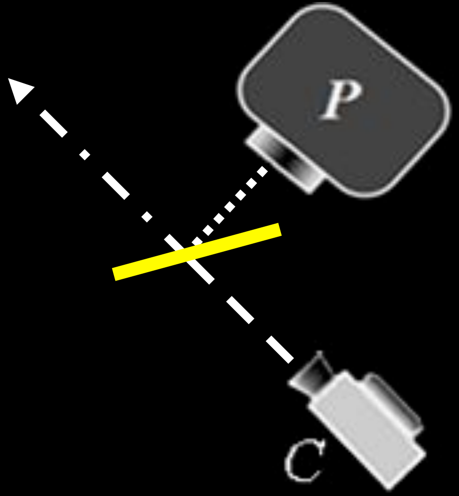
→ $\mathbf{E_{t-1}*M_0}$ **approx.** $\mathbf{E_0*M_0}$

$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$

t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Dynamic Adaptation

Fujii et al, CVPR 2005



determining color mixing matrix V_0:

similar as before: $\mathbf{V_{ij}=\Delta C_j/\Delta I_i}$

(un-normalized!)

determine reflected environment light E_0*M_0 at t=0:

$\mathbf{E_0*M_0=C-V_0*I}$ (project arbitrary I and capture C)

compensation (per pixel at t):

$\mathbf{I_t=V_0^{-1}*(R*M_0/M_{t-1}-E_{t-1}*M_0)}$

→$\mathbf{E_{t-1}*M_0}$ **approx.** $\mathbf{E_0*M_0}$

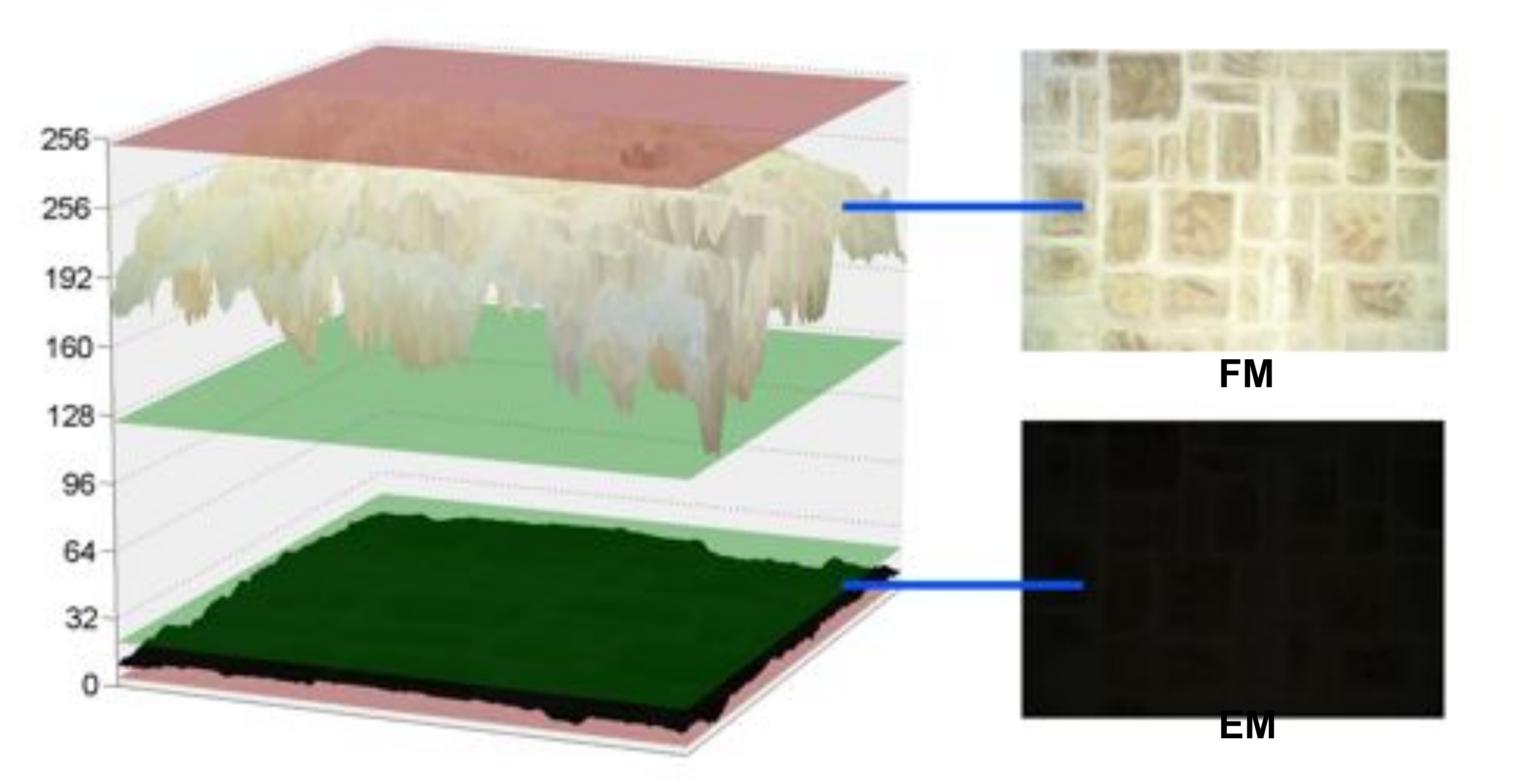
→$\mathbf{M_0/M_{t-1}=C_0/C_{t-1}}$

$\mathbf{R_t=M_t/M_0*(E_t*M_0+V_0*I_t)}$

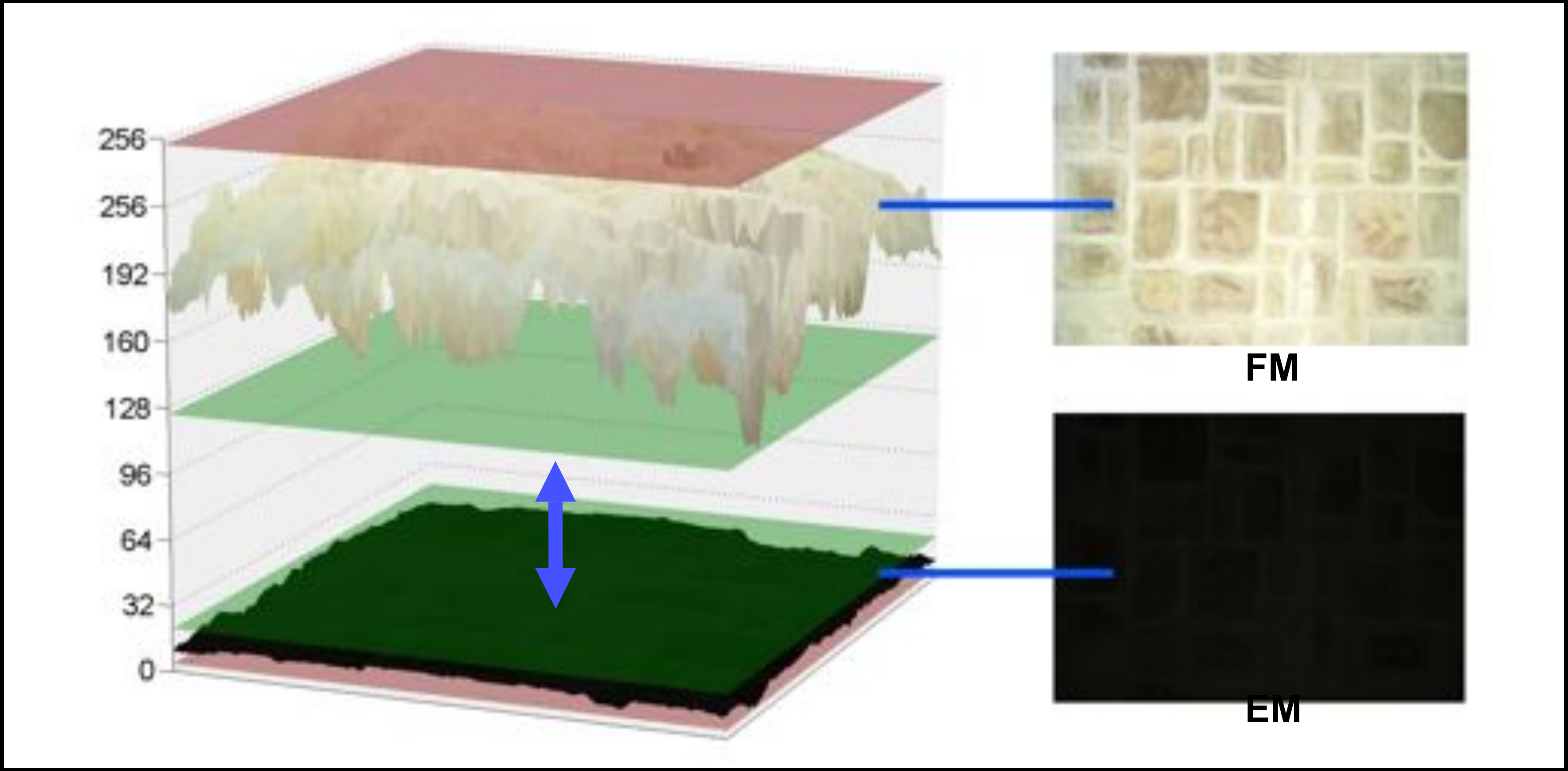
t → time index
I_t → projected image at t
V_0 → un-normalized color mixing matrix at t=0 (const.)
M_t → material at t
M_0 → material at t=0
E_t → environment light at t=0

Limited Dynamic Range and Brightness

Limited Dynamic Range and Brightness

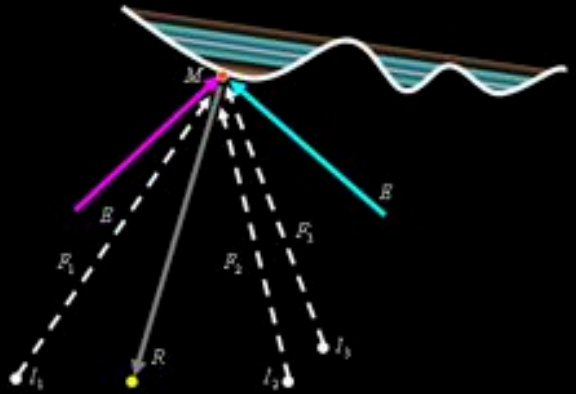


Limited Dynamic Range and Brightness



Multiple Projectors

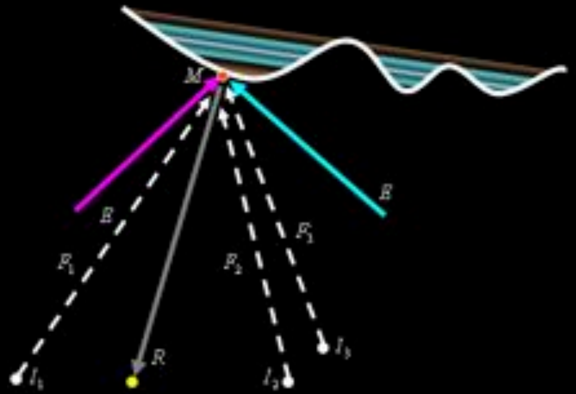
Multiple Projectors



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$

Multiple Projectors

strategy: balance intensity load



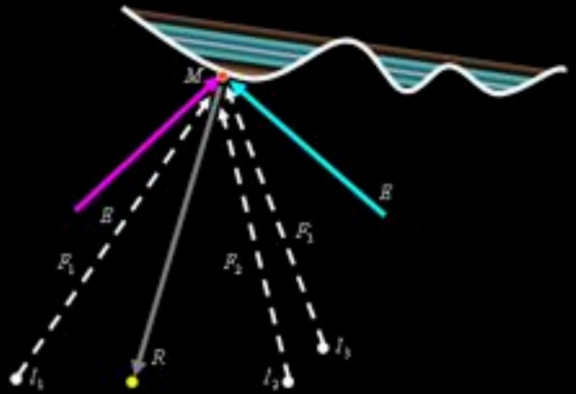
$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$

Multiple Projectors

strategy: balance intensity load

- assume: total intensity is equally balanced among multiple low-capacity units

  $I_i = I_1 = I_2 = \ldots = I_N$



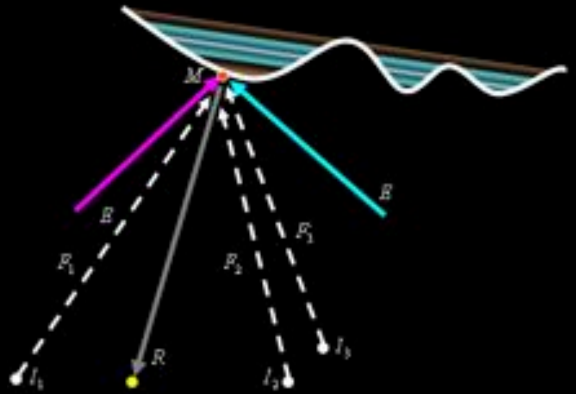
$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$

Multiple Projectors

strategy: balance intensity load

- assume: total intensity is equally balanced among multiple low-capacity units

$$I_i = I_1 = I_2 = \ldots = I_N$$



$$\boxed{R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM}$$

$$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

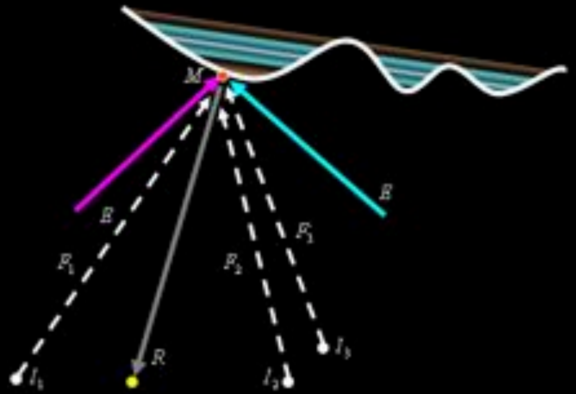
$$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$

Multiple Projectors

strategy: balance intensity load

- assume: total intensity is equally balanced among multiple low-capacity units

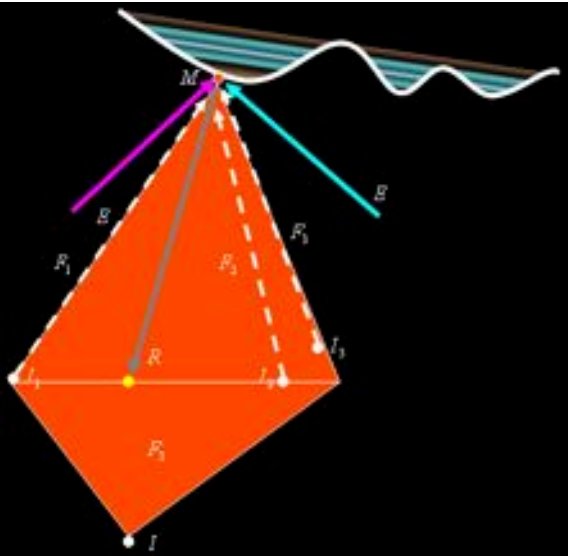
$$I_i = I_1 = I_2 = \ldots = I_N$$



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$

$$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

$$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$



Multiple Projectors

strategy: balance intensity load

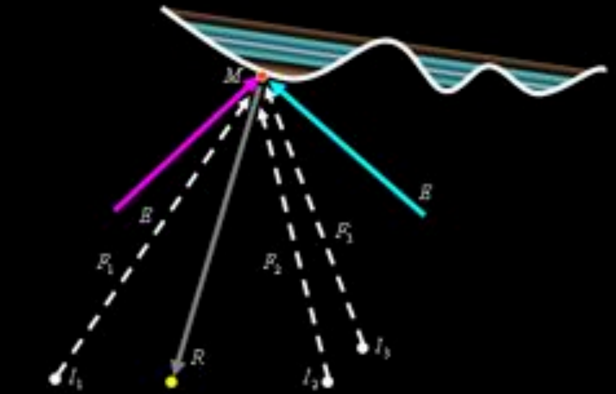
- assume: total intensity is equally balanced among multiple low-capacity units

  $$I_i = I_1 = I_2 = \ldots = I_N$$

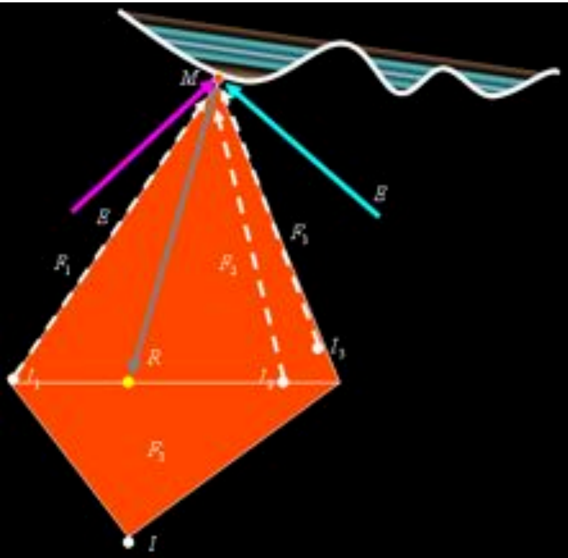
- this is equivalent to the assumption that a single high capacity projector produces the total intensity arriving on the surface virtually

  $$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

  $$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$



Multiple Projectors

strategy: balance intensity load

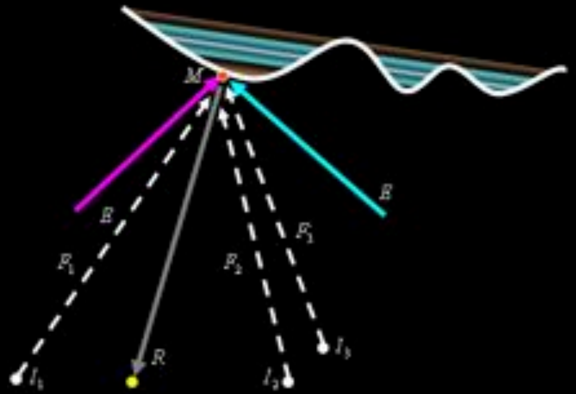
- assume: total intensity is equally balanced among multiple low-capacity units

  $$I_i = I_1 = I_2 = \ldots = I_N$$

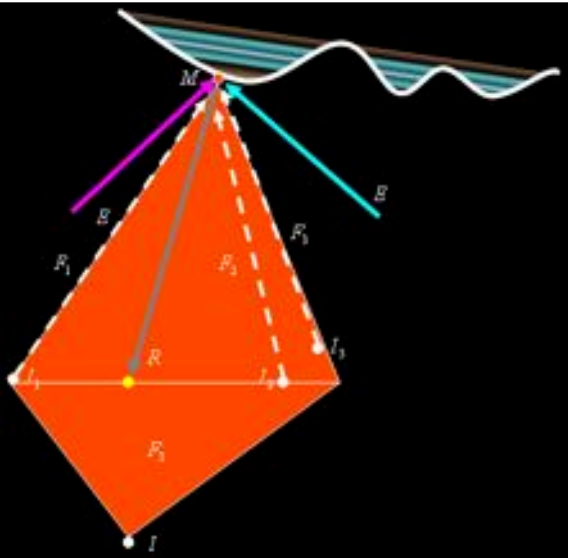
- this is equivalent to the assumption that a single high capacity projector produces the total intensity arriving on the surface virtually

  $$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

  $$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$



Multiple Projectors

strategy: balance intensity load

- assume: total intensity is equally balanced among multiple low-capacity units

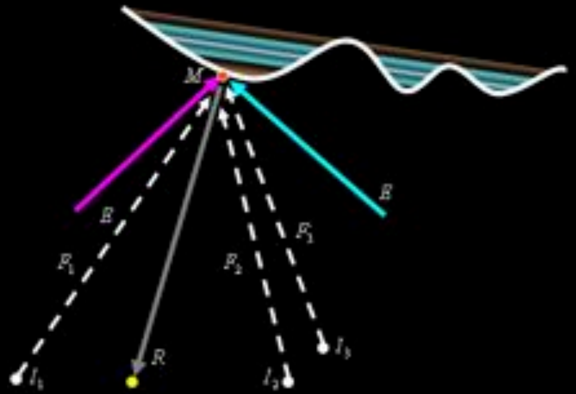
$$I_i = I_1 = I_2 = \ldots = I_N$$

- this is equivalent to the assumption that a single high capacity projector produces the total intensity arriving on the surface virtually

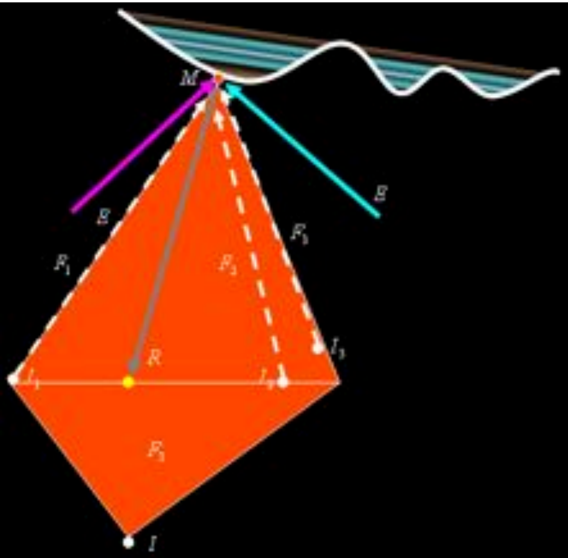
$$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

$$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$

compensation (per pixel):



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$



Multiple Projectors

strategy: balance intensity load

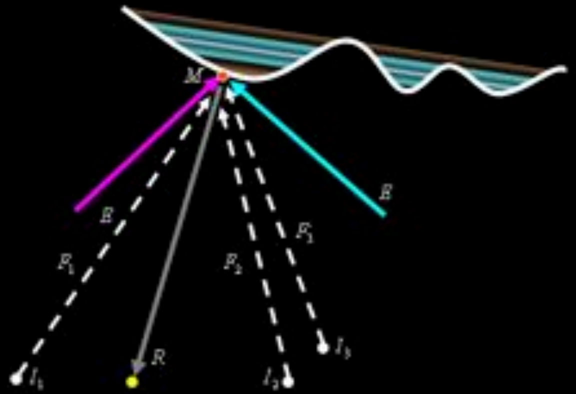
- assume: total intensity is equally balanced among multiple low-capacity units

$$I_i = I_1 = I_2 = \ldots = I_N$$

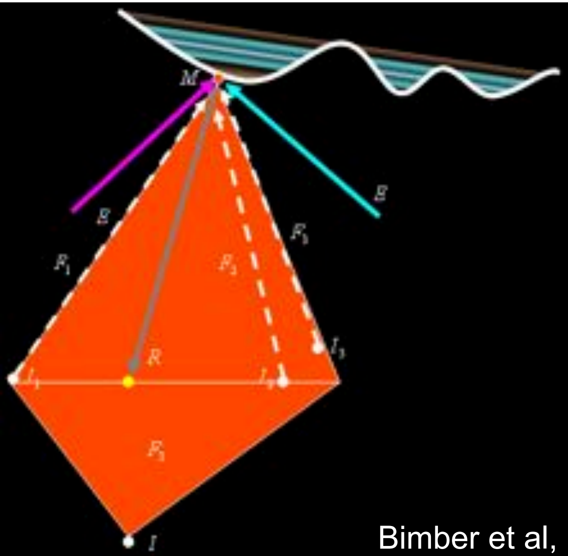
- this is equivalent to the assumption that a single high capacity projector produces the total intensity arriving on the surface virtually

$$R = EM + I_i(F_1M + F_2M + \ldots + F_NM)$$

$$\rightarrow EM+I(F_1+F_2+\ldots F_N)M$$



$$R = EM + I_1F_1M + I_2F_2M + \ldots + I_NF_NM$$



compensation (per pixel):

$$I_i = (R-EM)/(F_1M + F_2M + \ldots + F_NM)$$

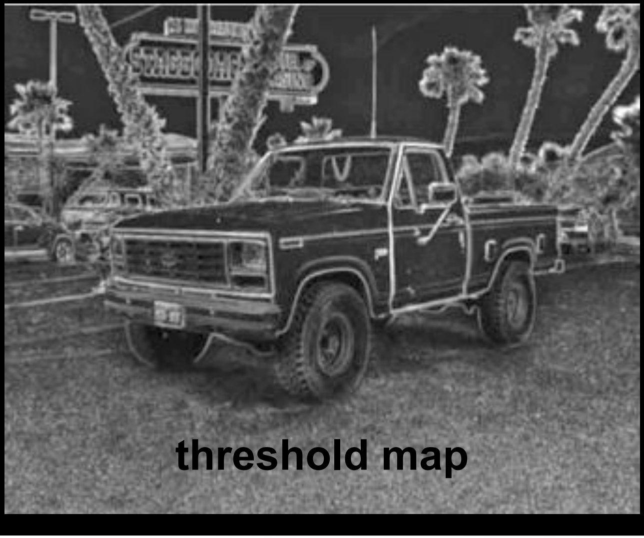
remember: $F_iM=F_iM-B_iF_iM$!

or $BFM=B_1F_1M+\ldots+B_iF_iM$

Bimber et al, IEEE Computer 2005

Considering Human Visual Perception

Considering Human Visual Perception



threshold map (Ramasubramanian et al. Siggraph'99)

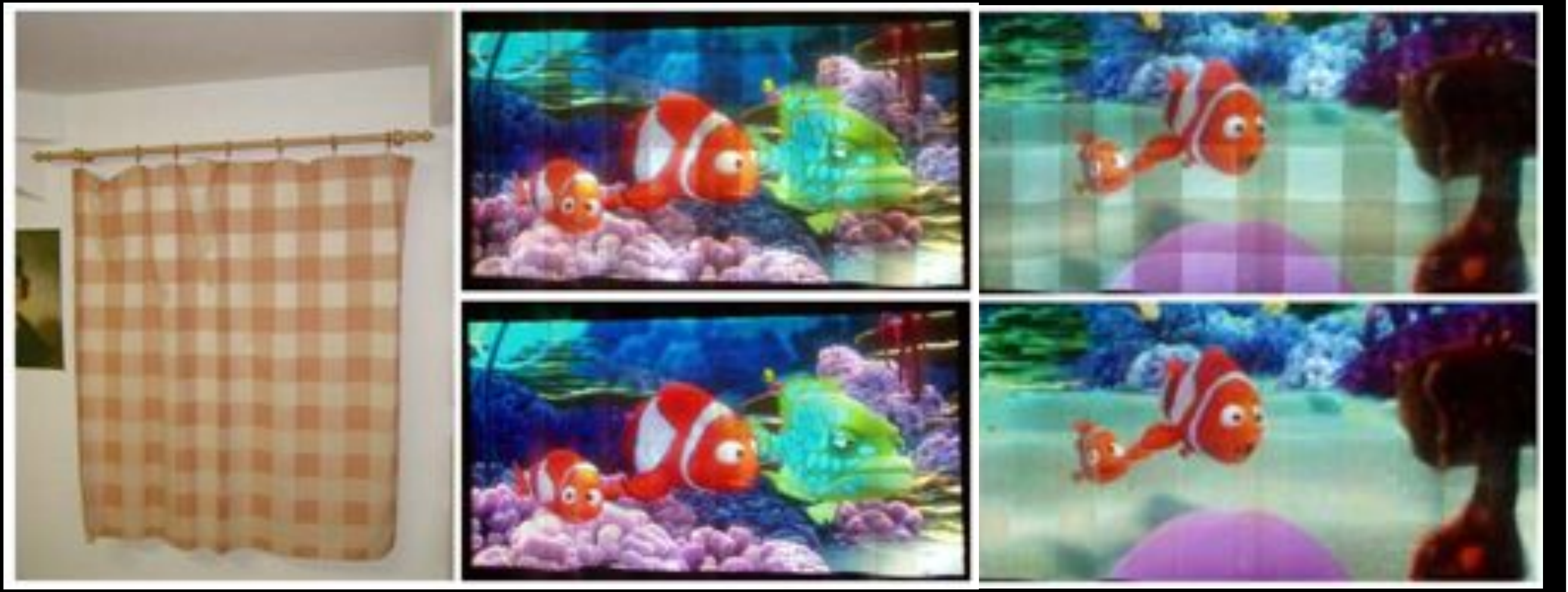
- computes for every pixel of an image **R** the amount of luminance difference that is imperceptible
- considers ***contrast***, ***luminance*** and ***spatial frequency*** in local neighborhood

adaptation of un-compensated (original) image R:

- compute and apply a single (global) scaling factor **R'=R***α that minimizes the perceived error (Wang, et al. 2005, only monochrome, not real-time, single projector)
- coming soon: color, real-time, global and local adaptation, potentially multiple projectors

Example: Curtain

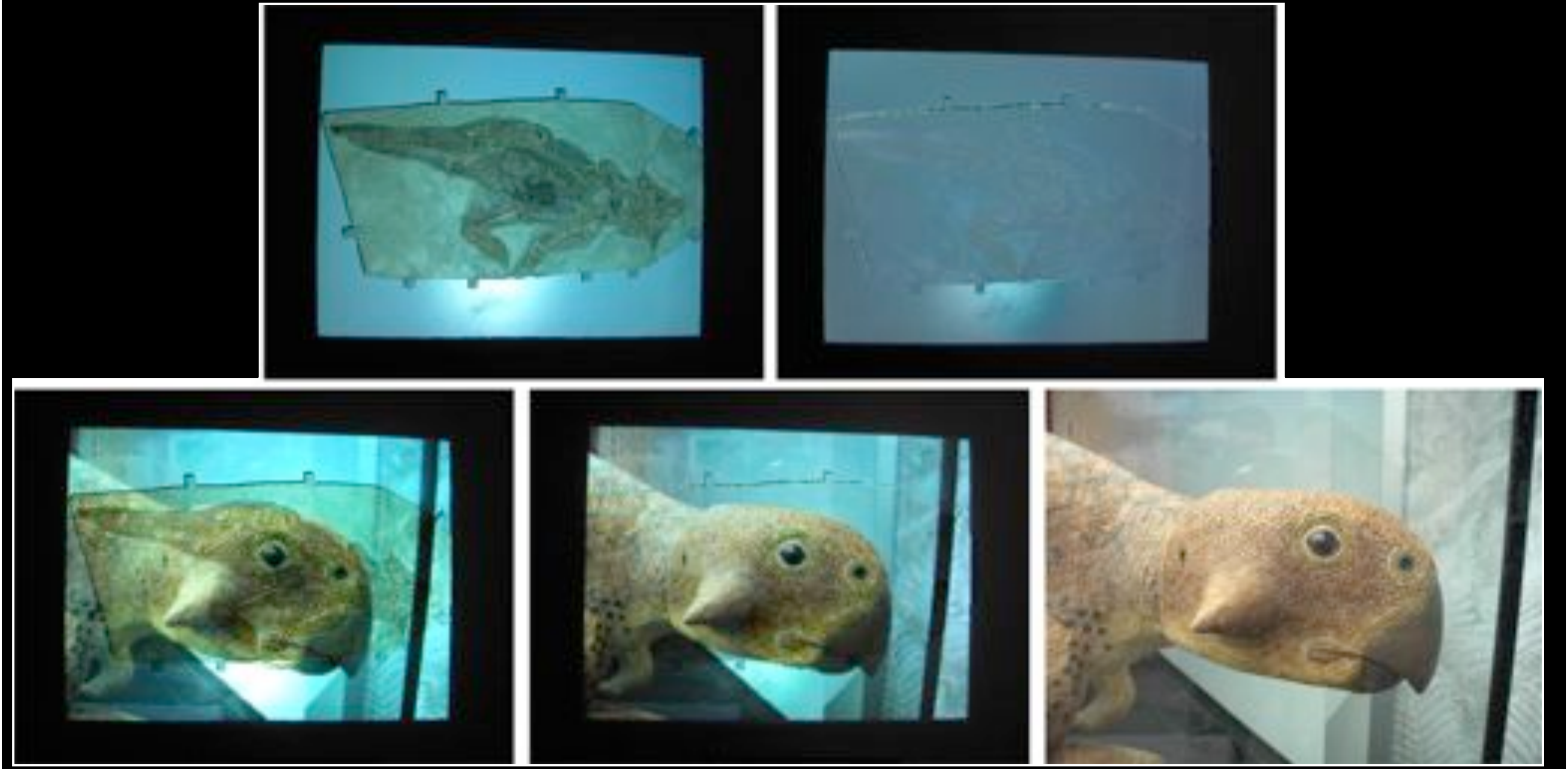
Example: Curtain



Bimber et al, IEEE
Computer 2005

Example: Fossil

Example: Fossil



In coop. with Senckenberg Museum

Example: Natural Stone Wall

Example: Natural Stone Wall

Bimber et al, IEEE Computer 2005
In coop. with Bennert Group

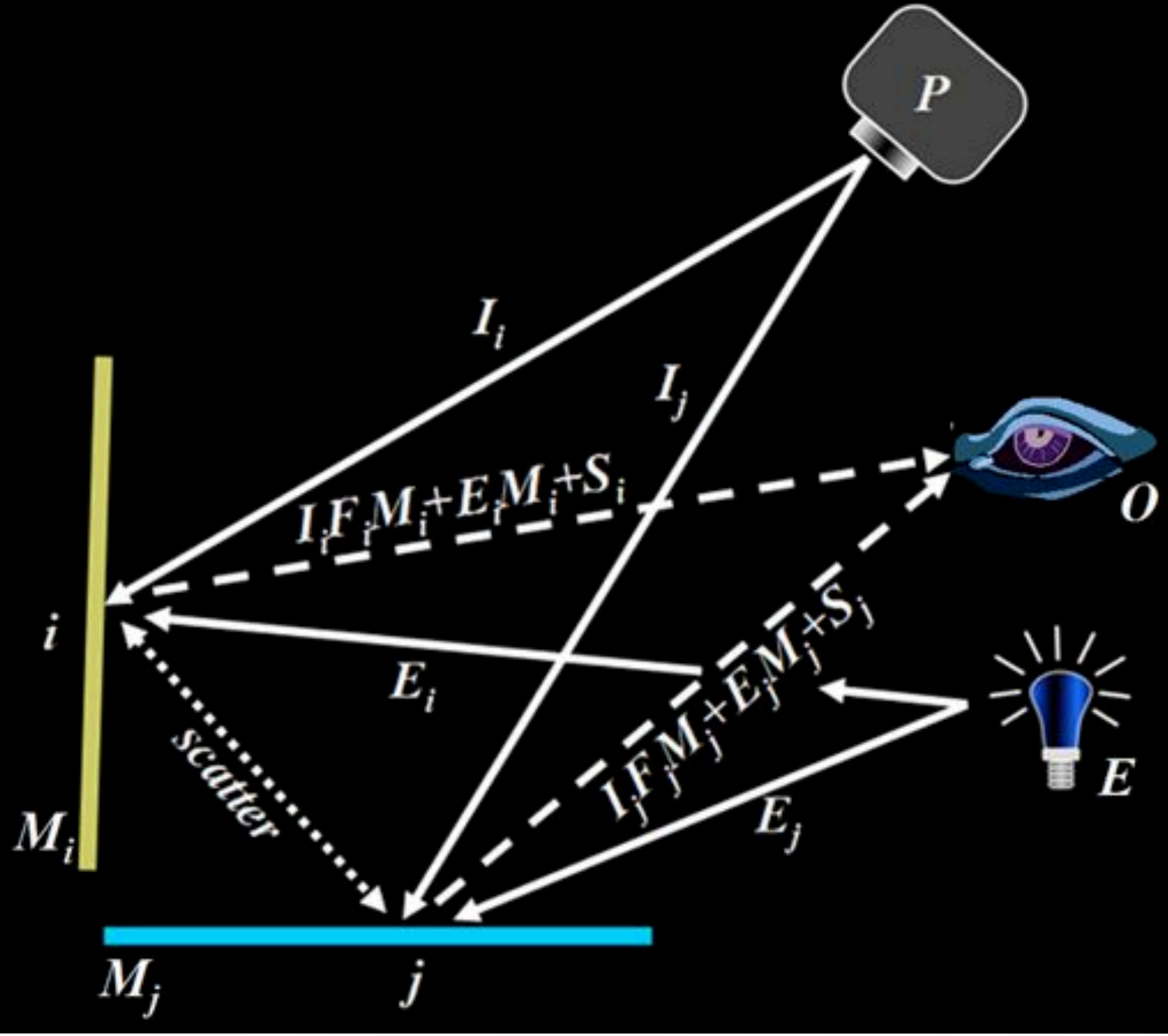
Example: Wallpaper

Example: Wallpaper

Bimber et al,
IEEE Computer
2005

Compensating Global Light Effects

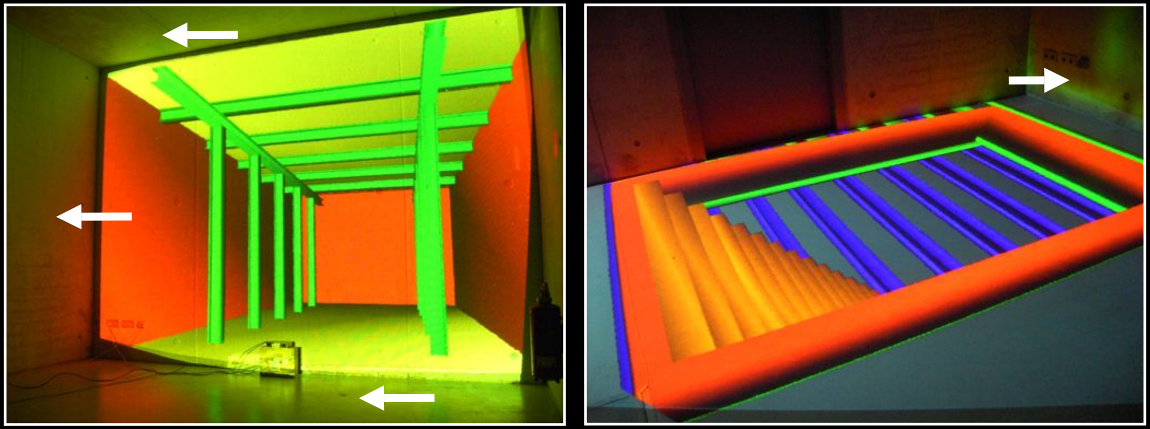
Compensating Global Light Effects



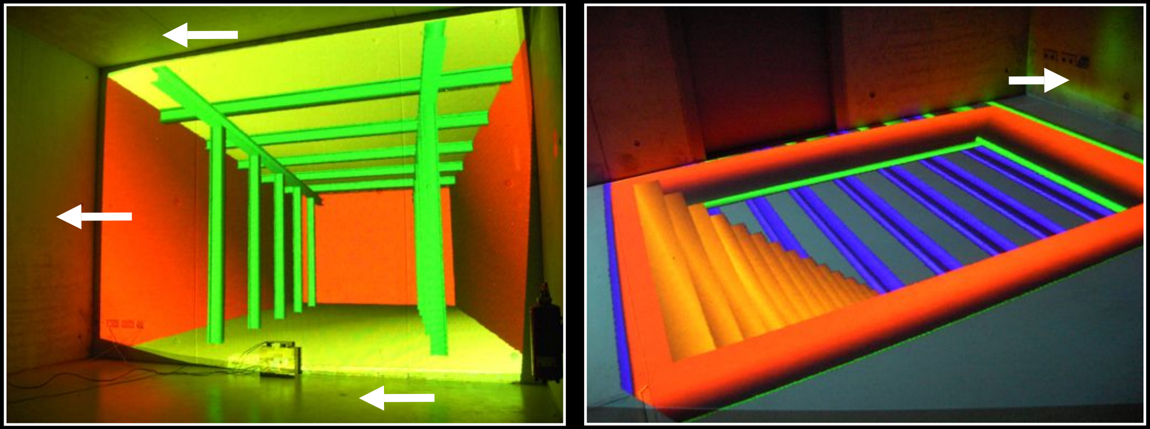
Compensating Diffuse Scattering

Compensating Diffuse Scattering

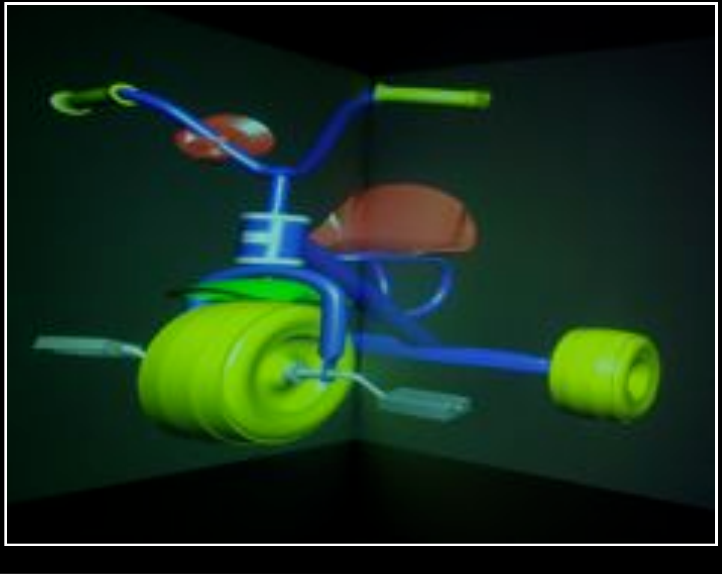
Bimber et al,
IEEE/ACM
ISMAR 2005



Compensating Diffuse Scattering



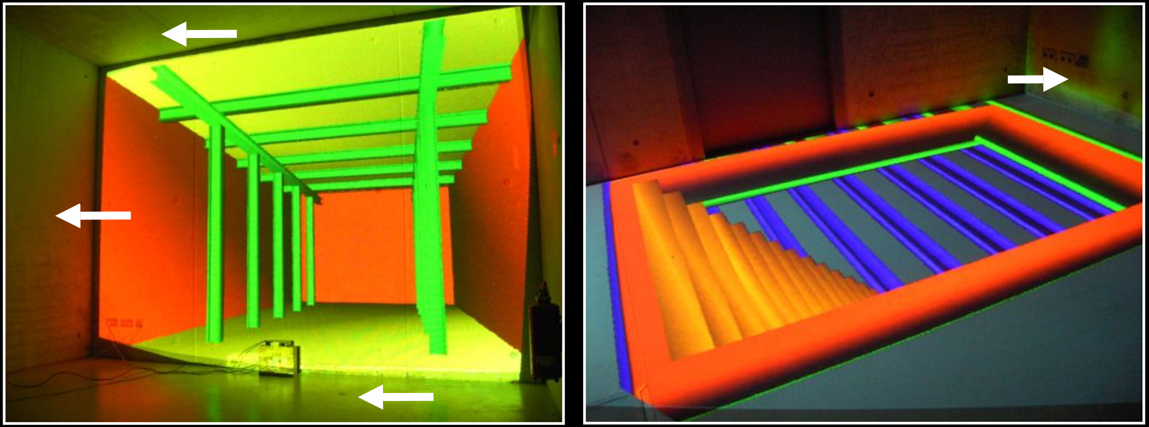
Bimber et al,
IEEE/ACM
ISMAR 2005



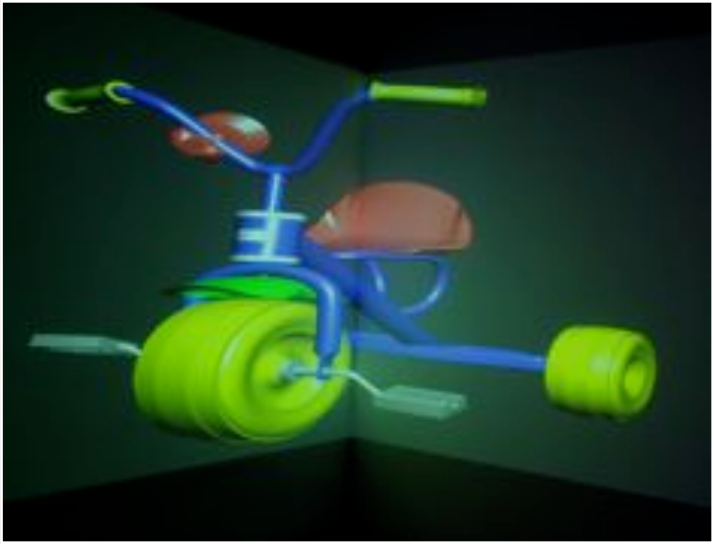
Bimber et al,
IEEE VR, 2006

Compensating Diffuse Scattering

Bimber et al,
IEEE/ACM
ISMAR 2005

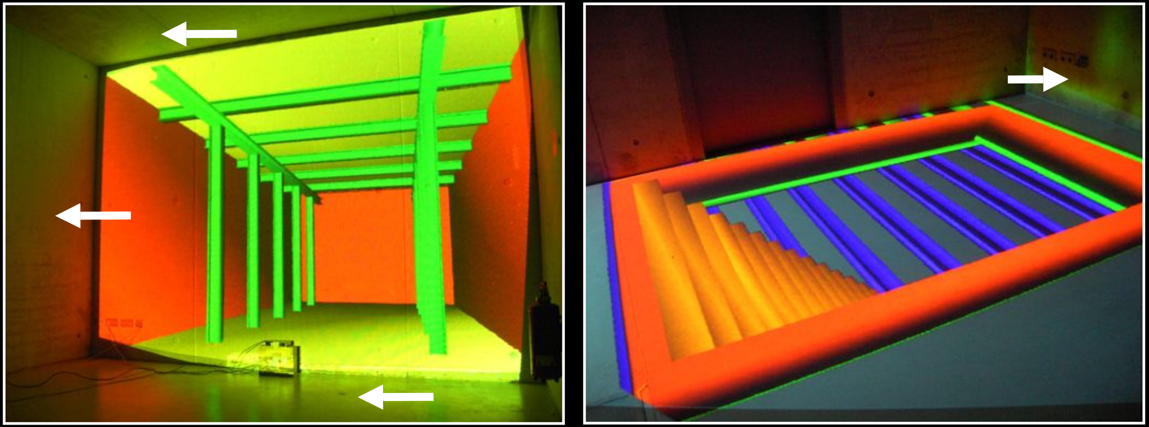


Bimber et al,
IEEE VR, 2006

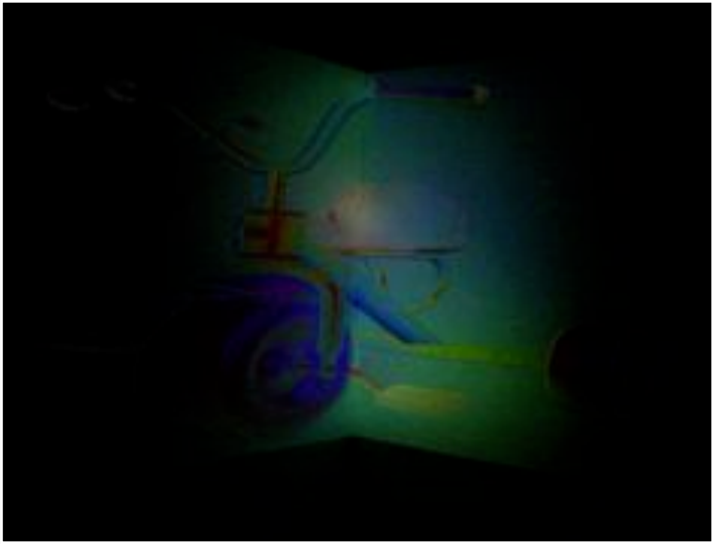


Compensating Diffuse Scattering

Bimber et al,
IEEE/ACM
ISMAR 2005



Bimber et al,
IEEE VR, 2006



details:
IEEE VR talk on Wednesday morning (8:30am), session on tracking and projection displays

see demo!

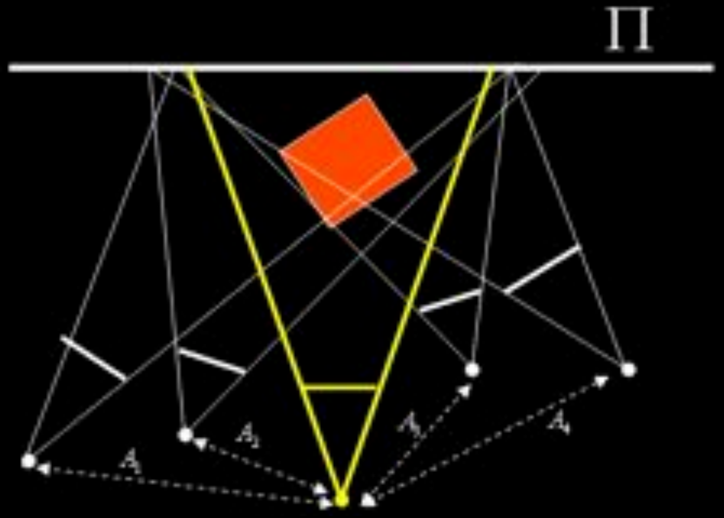
Advanced Techniques

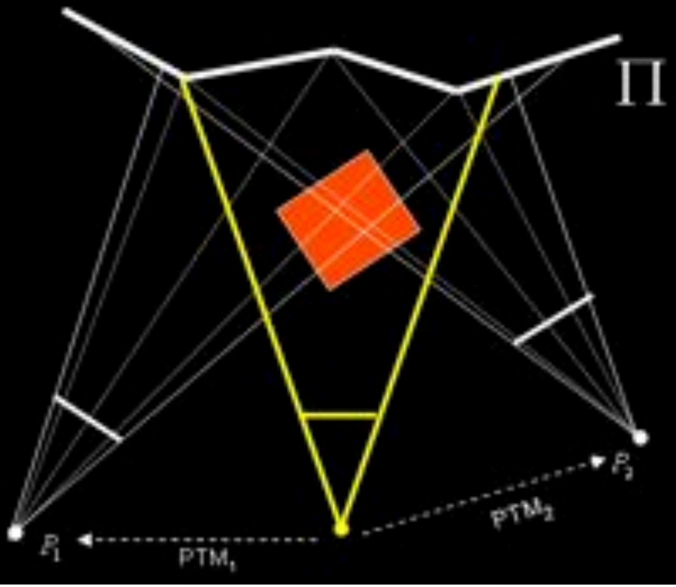
View-Dependence

Non-Complex Surfaces

Non-Complex Surfaces

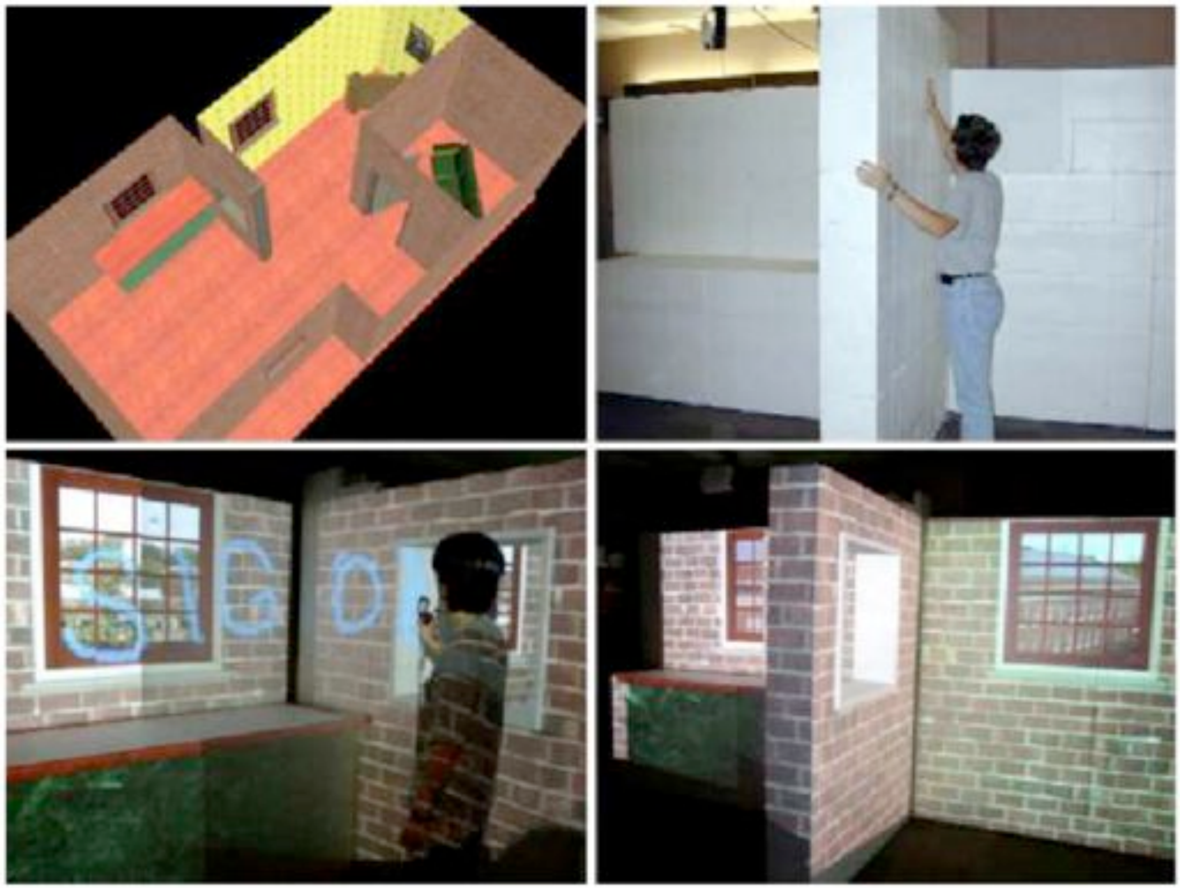
- view-dependent geometry correction can be compute if geometry is known
- for example:
  - **planar/multi-plane:** off-axis projection
  - **parametric:** warping via parametric description
  - **scanned/modelled:** projective texture mapping





Example: Life-Sized Projector-Based Dioramas

Example: Life-Sized Projector-Based Dioramas



Courtesy: Low, et al., 2001

Complex Surfaces

Complex Surfaces

- if geometry is unknown image-based rendering helps
- sample geometric and radiometric parameters from multiple (source) camera (perspective)
- for novel (destination) camera
  - compute weighted penalties:

$$p_j = \alpha a_j + (1-\alpha) b_j$$

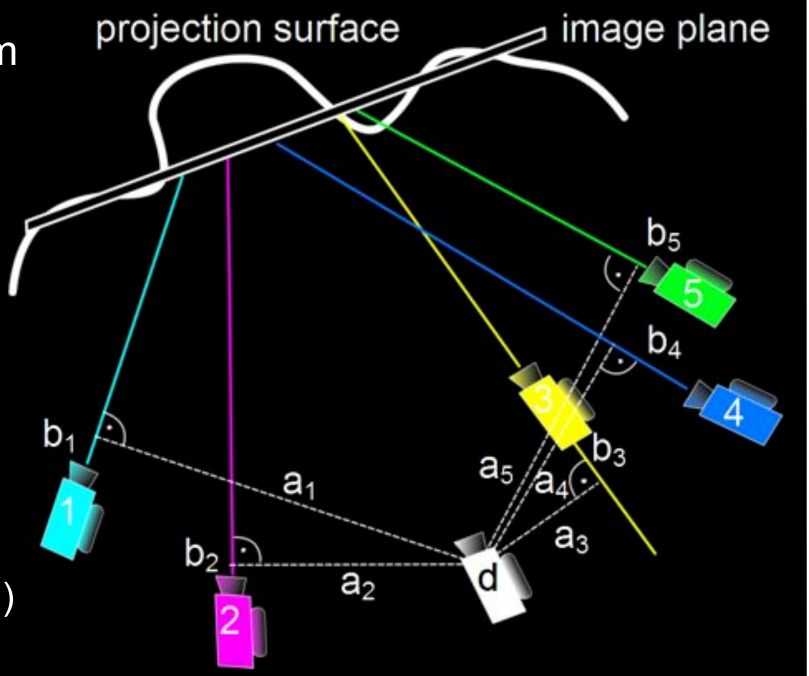
  - select k best perspectives (lowest penalties) and normalize them:

$$w_j = \left(1 - \frac{p_j}{\max_{pk}}\right)\frac{1}{p_j}$$

  - interpolate new parameter textures (P_i2C_j, $F_{ij}M$, $E_{ij}M$) and direction vector for destination perspective to render new IP:

$$t_d = \sum_j^k w_j t_j$$

  - lookups in $F_{ij}M$, $E_{ij}M$ with non-interpolated P_i2C_j
  - lookups in IP with interpolated P_i2C_j



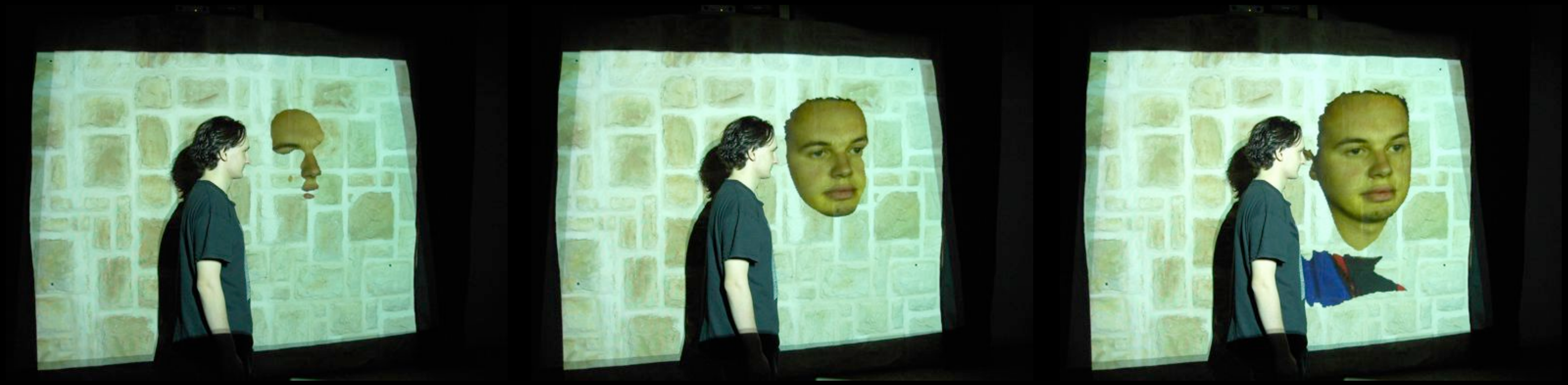
Example: Tracking and Stereo

Example: Tracking and Stereo

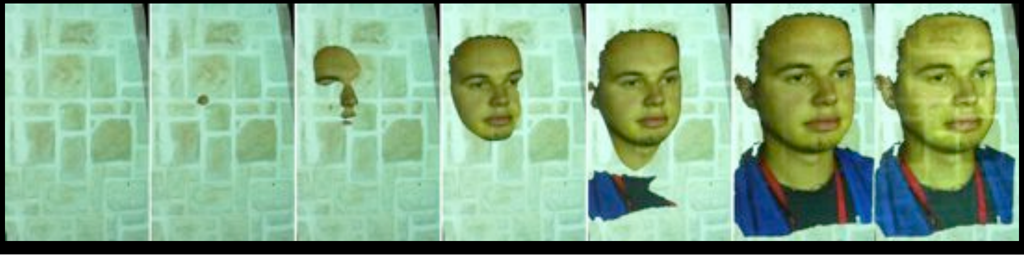
Bimber et al, IEEE/ACM ISMAR 2005

Depth and Occlusion

Depth and Occlusion

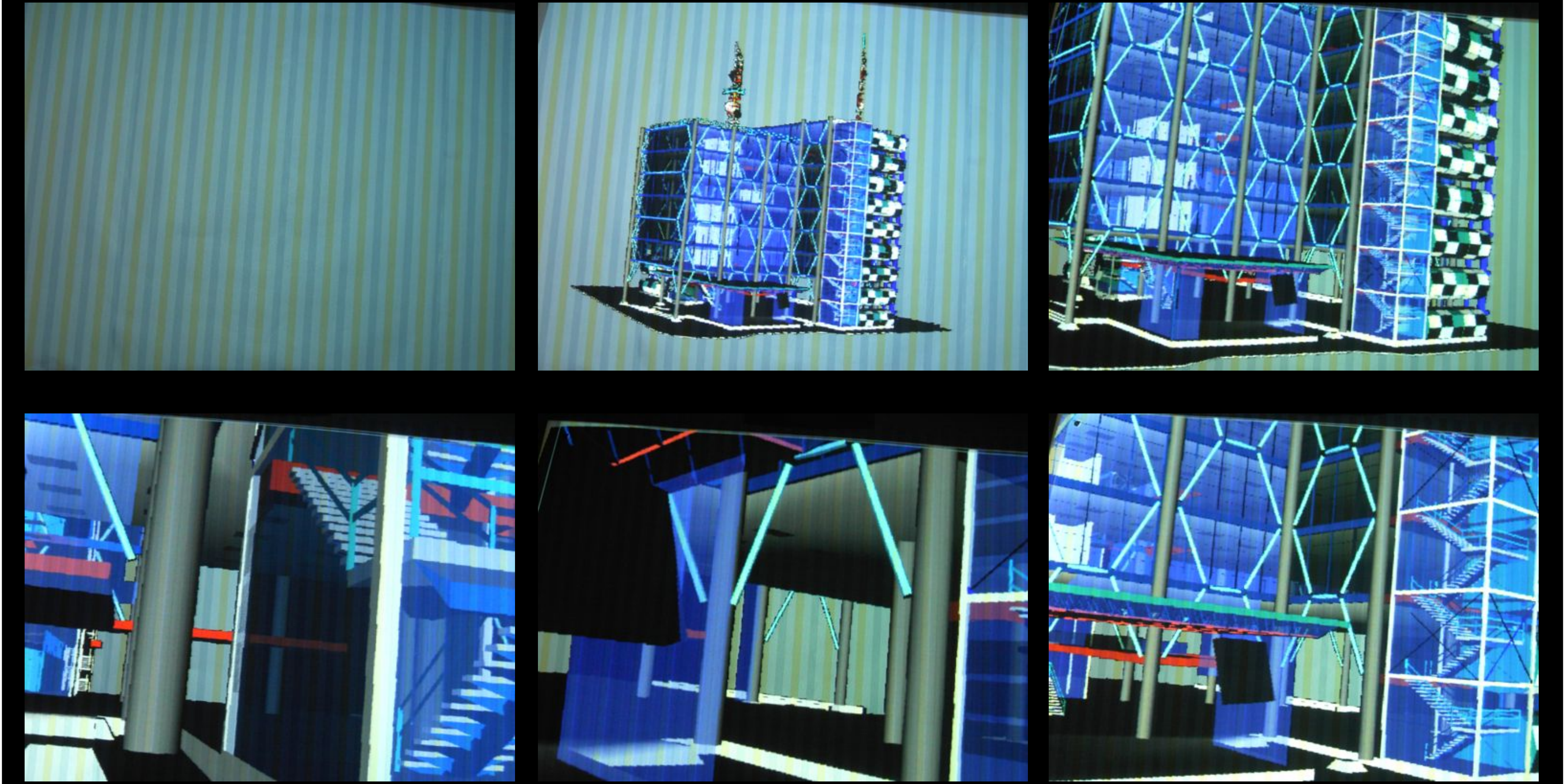


Bimber et al,
IEEE/ACM
ISMAR 2005



Example: Stereo on Wallpaper

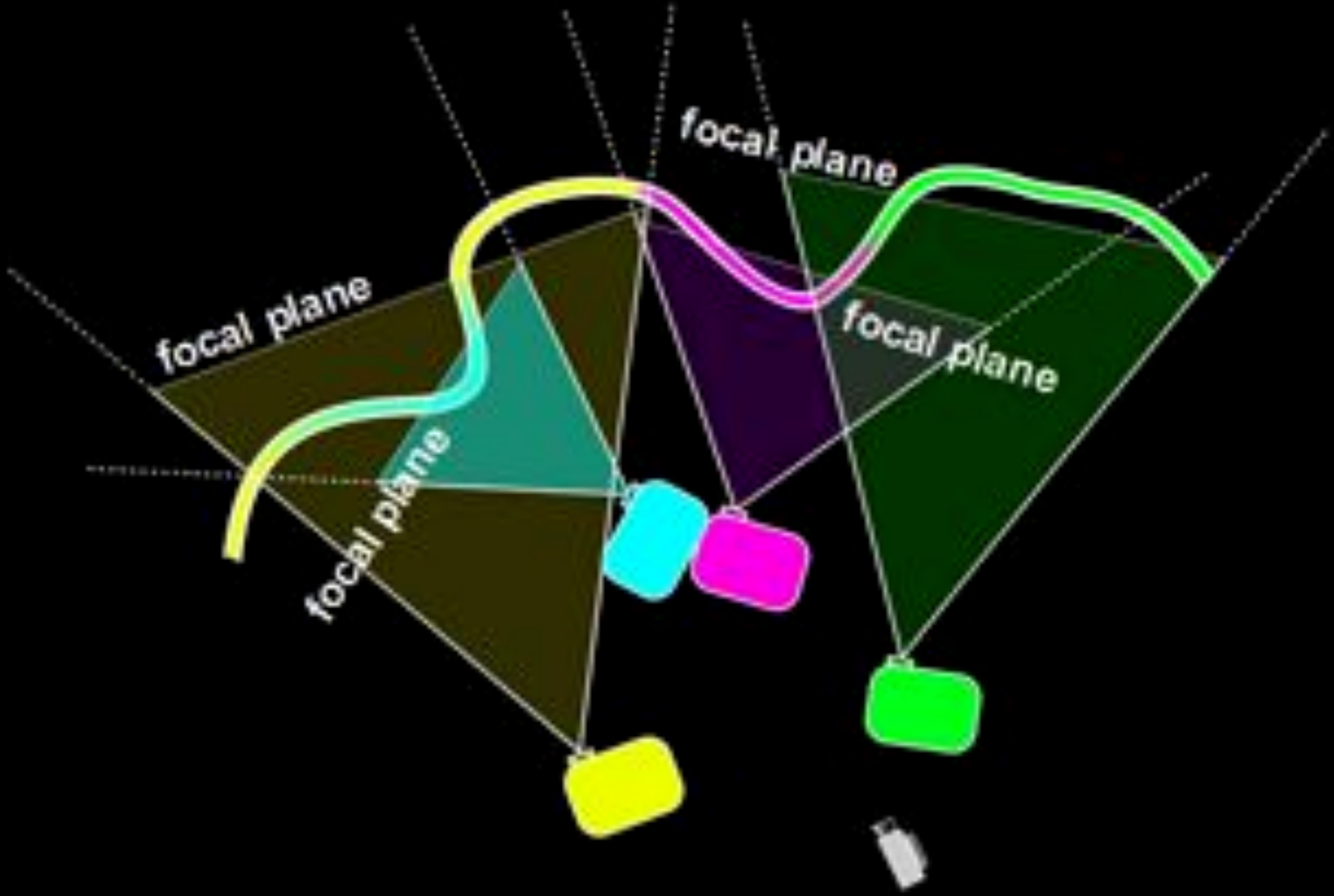
Example: Stereo on Wallpaper



Bimber et al, IEEE/ACM
ISMAR 2005

Advanced Techniques
Multi-Focal Projection

Multi-Projector-Camera Technique that Increases Focal Depth



Determining Defocus

Determining Defocus

- structured light projection of grid point samples (2- dimensional phase shift)
    - **pre-correction:** geometric and radiometric correction (corrected grid points must be observed in camera)

$$I_{x,y} = (R_{x,y} - EM_{x,y}) / FM_{x,y}$$

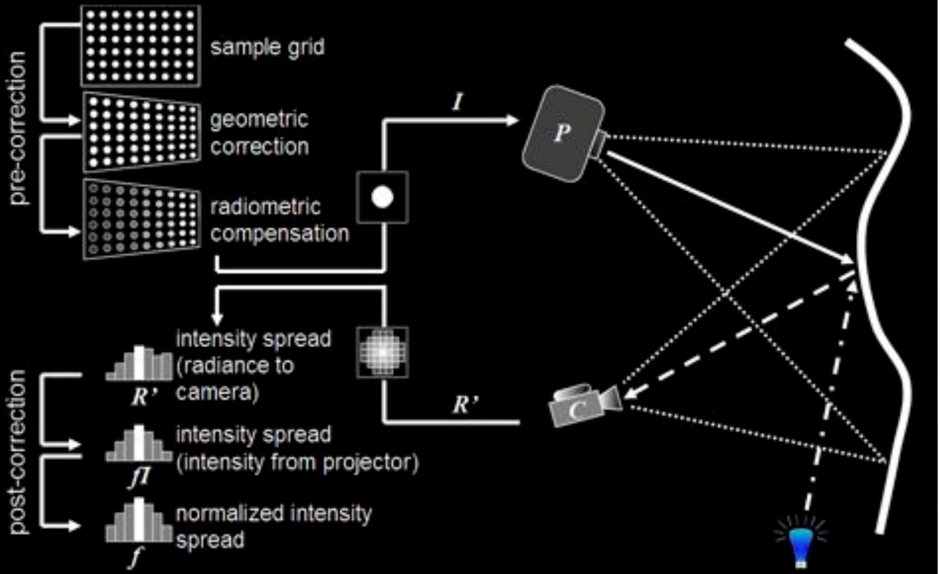
    - **post-correction:**

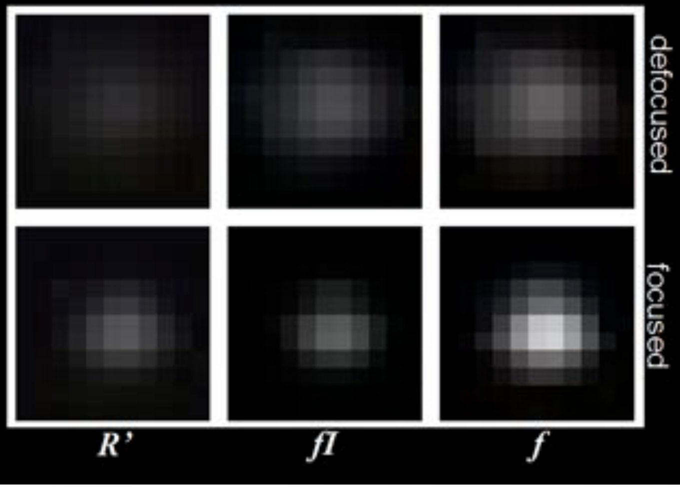
$$R'_{x',y'} = fI_{x,y} FM_{x',y'} + EM_{x',y'}$$

$$fI_{x,y} = \frac{(R'_{x',y'} - EM_{x',y'})}{FM_{x',y'}}$$

$$f = \frac{fI_{x,y}}{I_{x,y}}$$

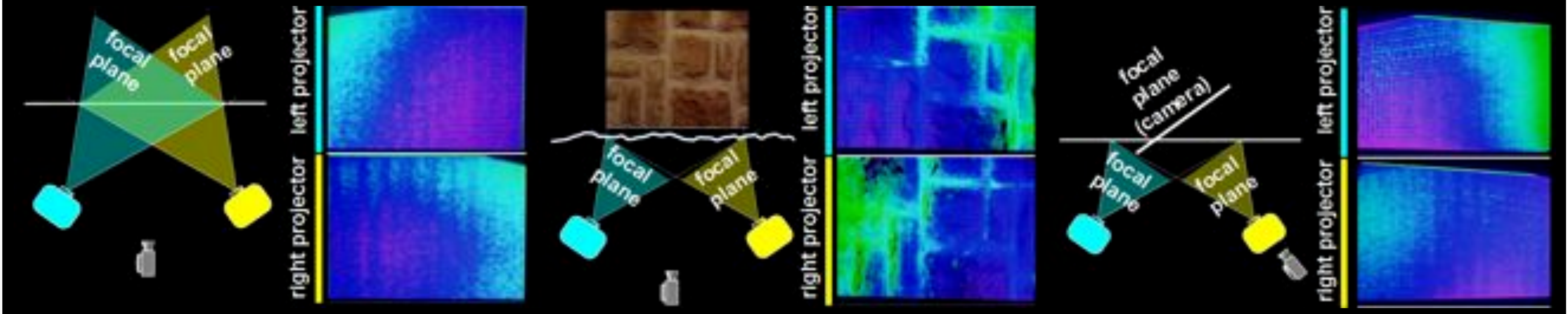
    - the normalized intensity spread texture f serves as basis to estimate focus measures (e.g., via FFT/DCT, intensity loss, point spread, etc.)





Example: Different Configurations

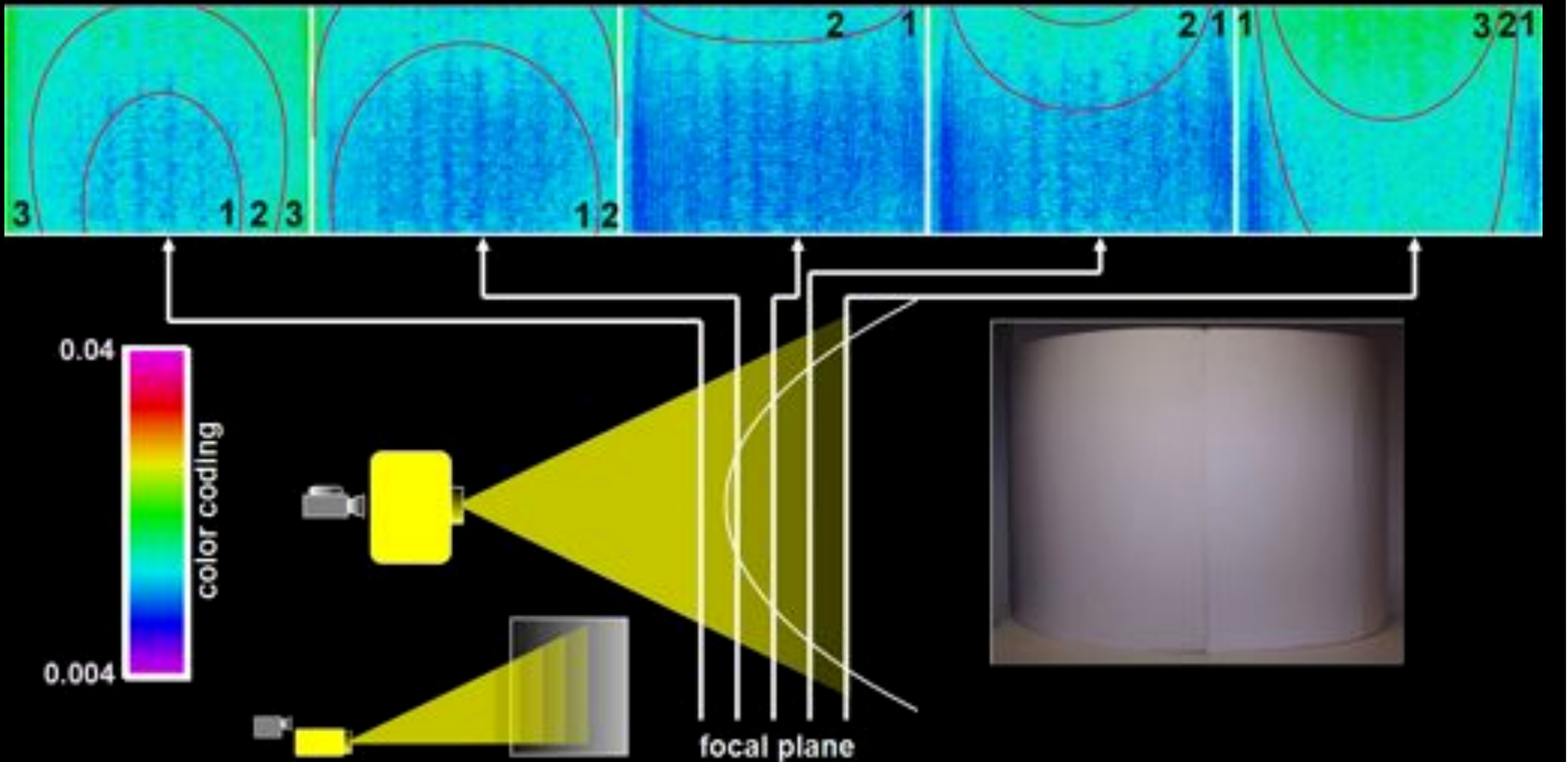
Example: Different Configurations



Bimber et al, IEEE TVCG 2006

Example: Shifting Focal Plane

Example: Shifting Focal Plane



Bimber et al, IEEE TVCG 2006

Image Composition

Image Composition

- using the focus values of each projector's pixels ($\Phi_{i,x,y}$), compose an image with minimal total defocus
  - **excusive composition:** surface point is covered by a single projector pixel (the one with highest $\Phi_{i,x,y}$)

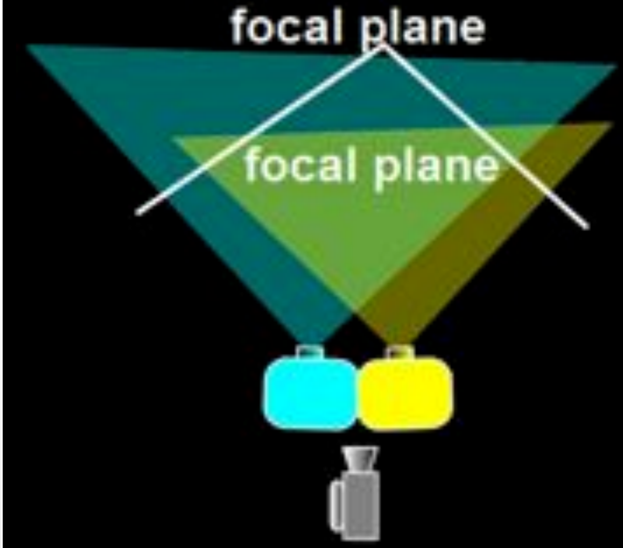
$$I_i = \frac{w_i(R-EM)}{\sum_j^N w_j FM_j} \quad w_{i,x,y} = \frac{\Phi_{i,x,y}}{\sum_j^N \Phi_{j,x,y}}$$

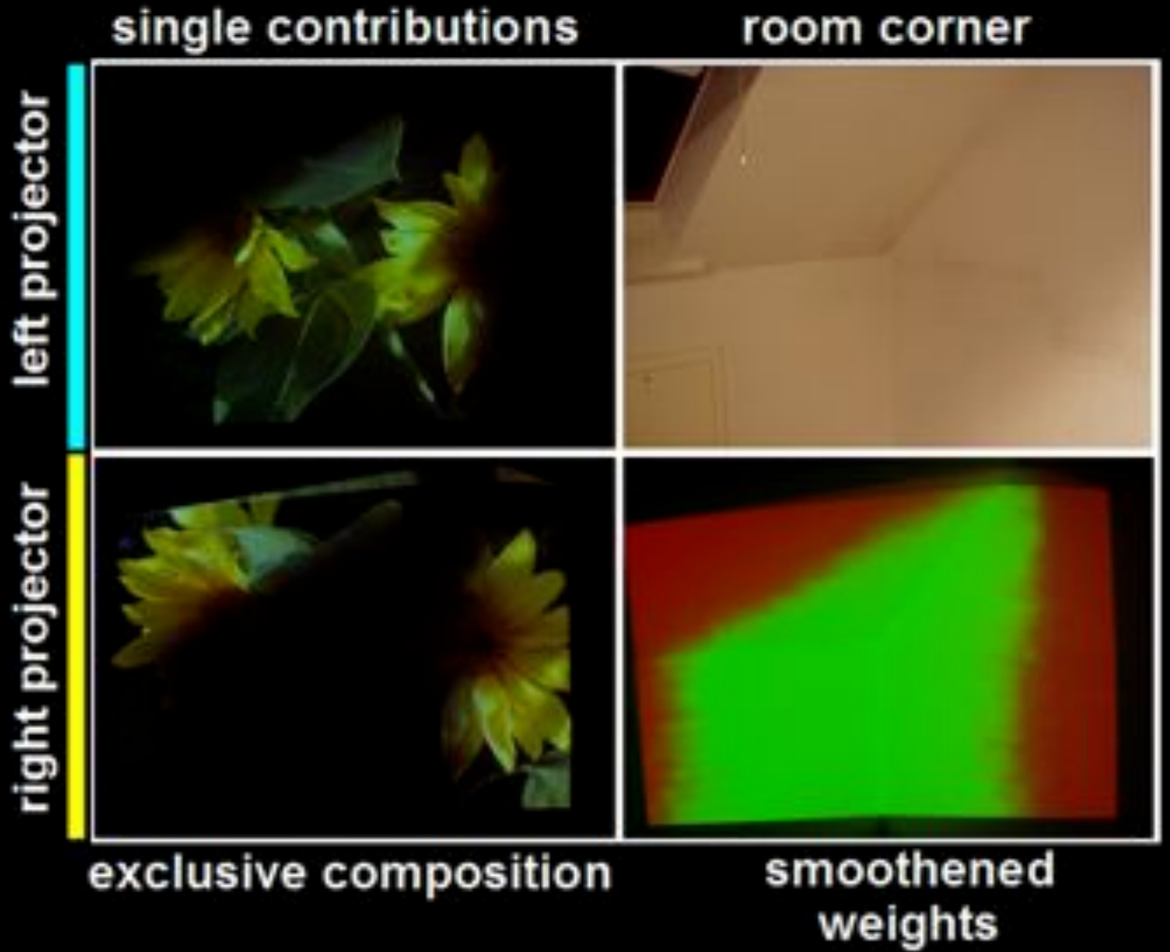
  - **weighted composition:** compute normalized weight and multiply it with *FM* and *I*

$$I_i = w_i(R\text{-}EM)/FM_i, \quad w_i = \begin{cases} 1 & \Phi_{i,x,y} \geq \Phi_{j,x,y} \\ 0 & else \end{cases}$$

Example: Room Corner

Example: Room Corner

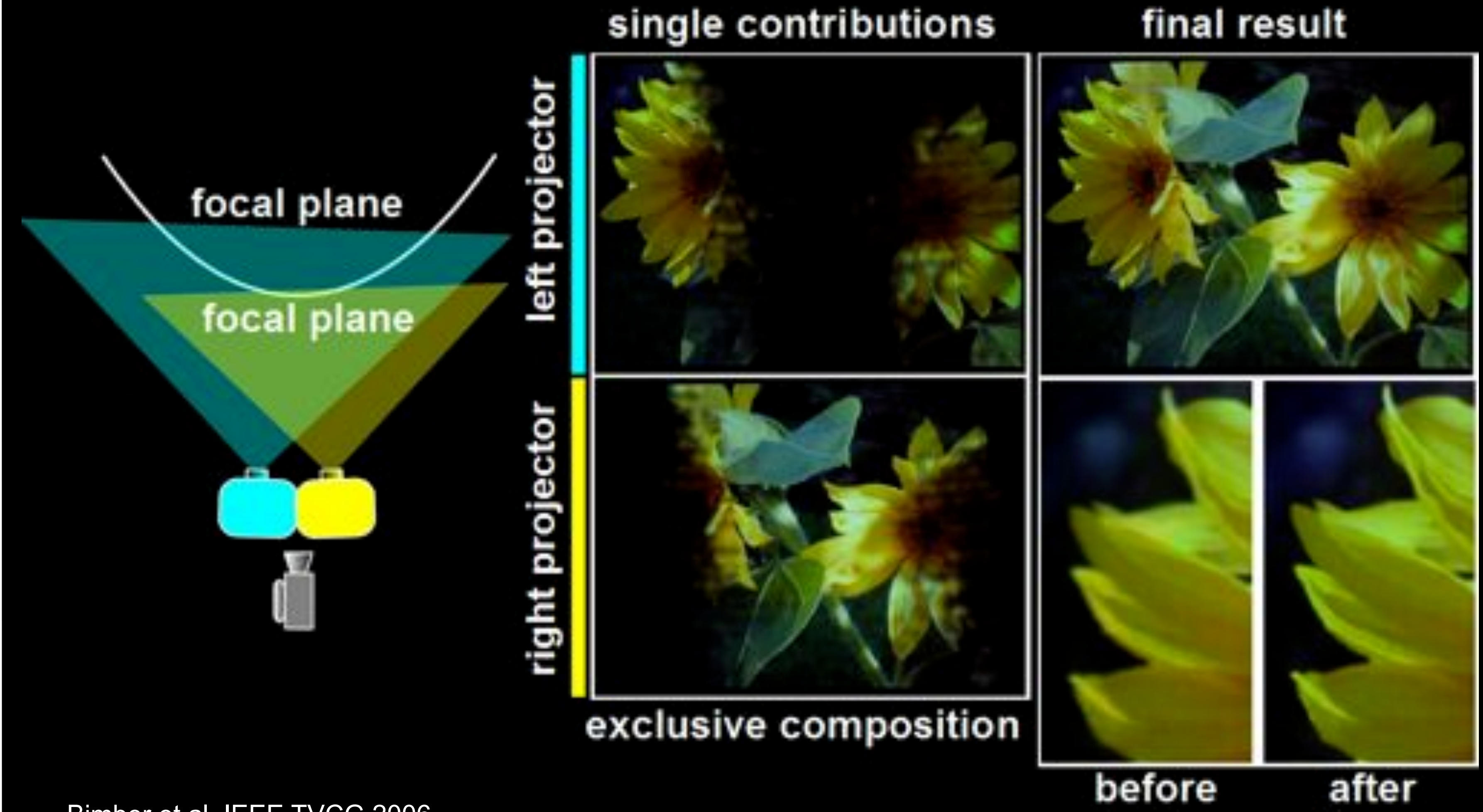




Bimber et al, IEEE TVCG 2006

Example: Cylindrical Surface

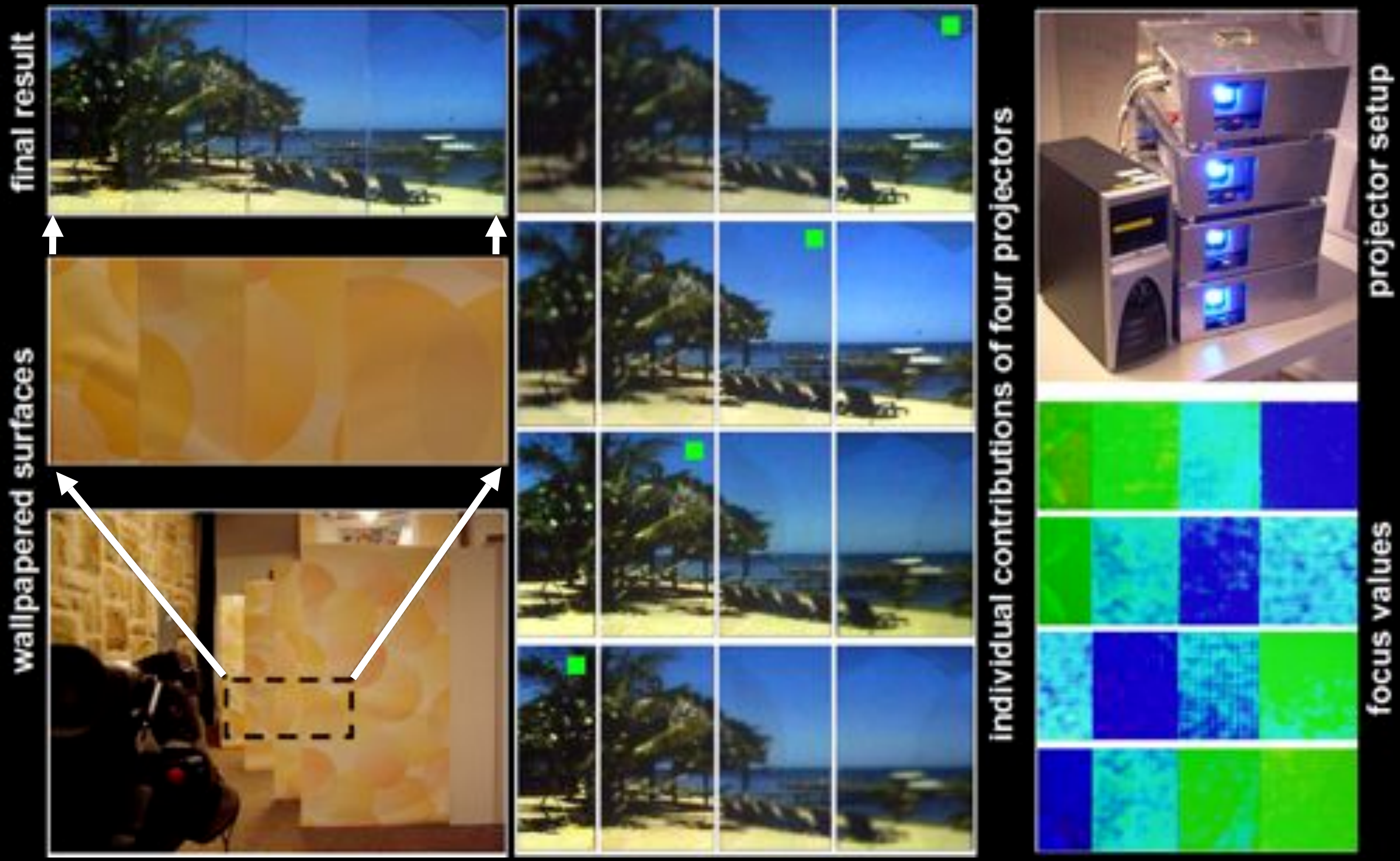
Example: Cylindrical Surface



Bimber et al, IEEE TVCG 2006

Example: Large Focal Depth

Example: Large Focal Depth



Bimber et al, IEEE TVCG 2006

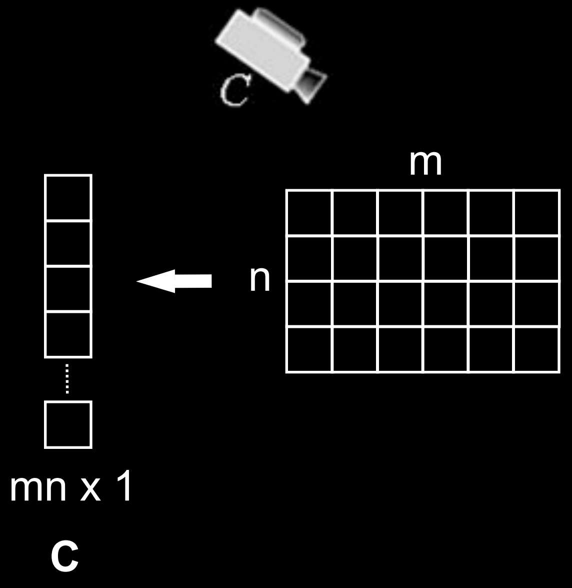
Advanced Techniques

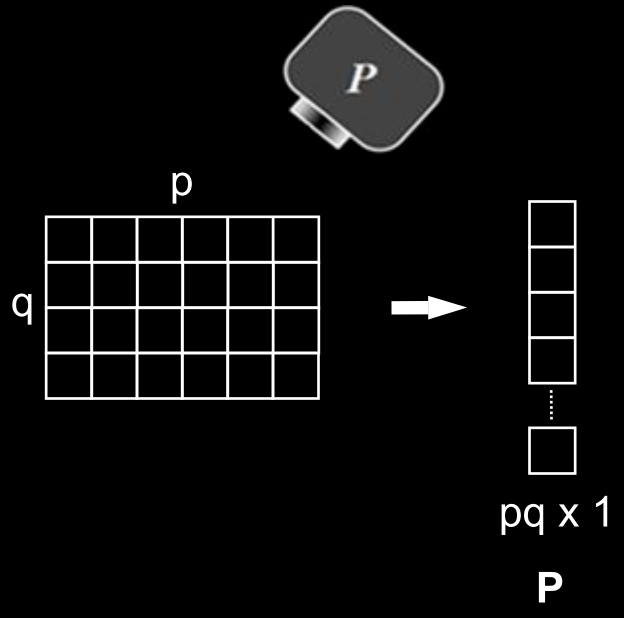
Light Transport

Acquisition

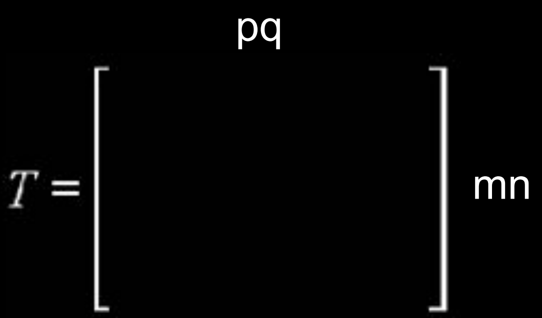
Acquisition

single camera & projector capture 4D slice of 8D reflectance field

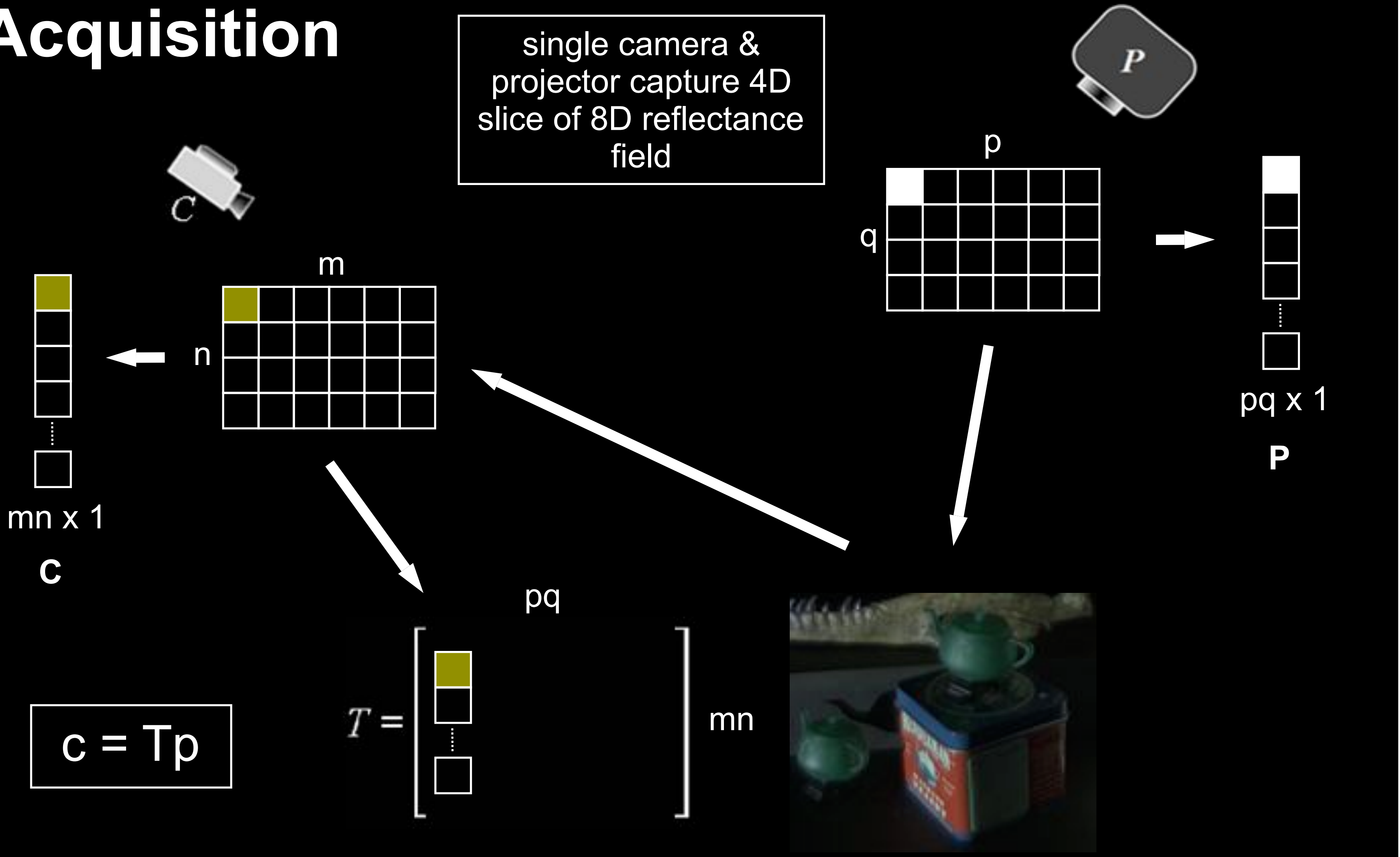




$$\boxed{c = Tp}$$



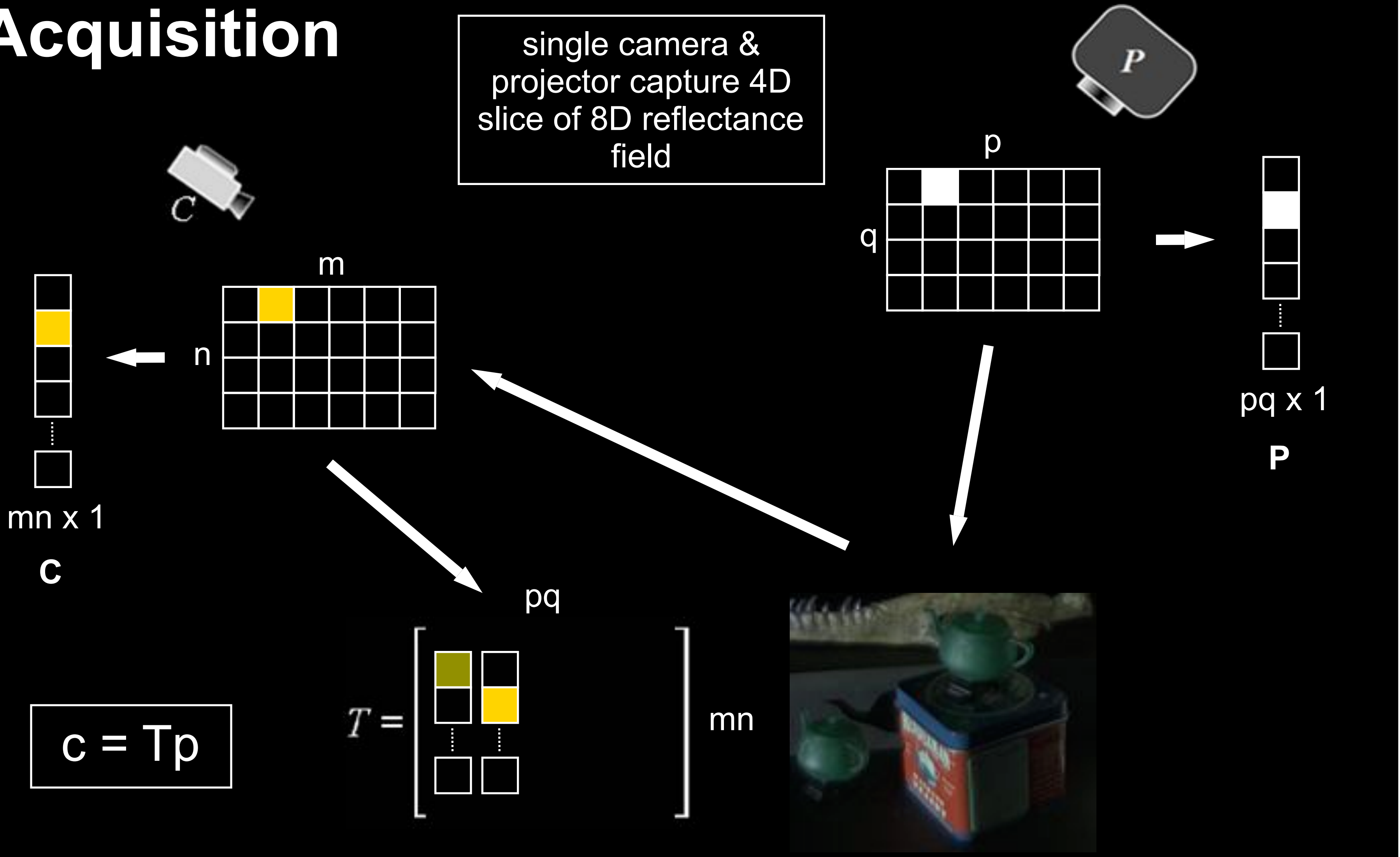
Acquisition



single camera & projector capture 4D slice of 8D reflectance field

$$\boxed{c = Tp}$$

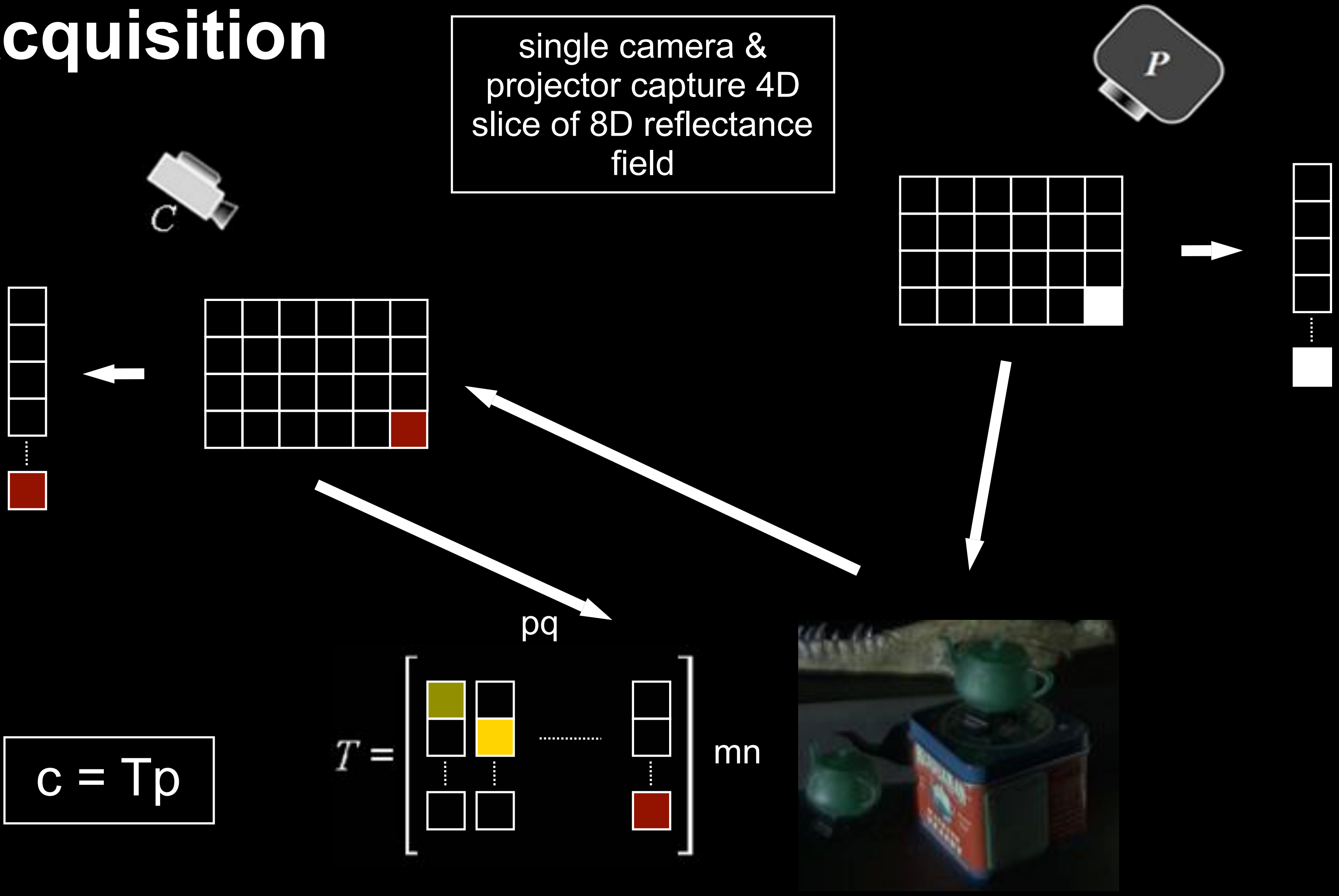
Acquisition



single camera & projector capture 4D slice of 8D reflectance field

$c = Tp$

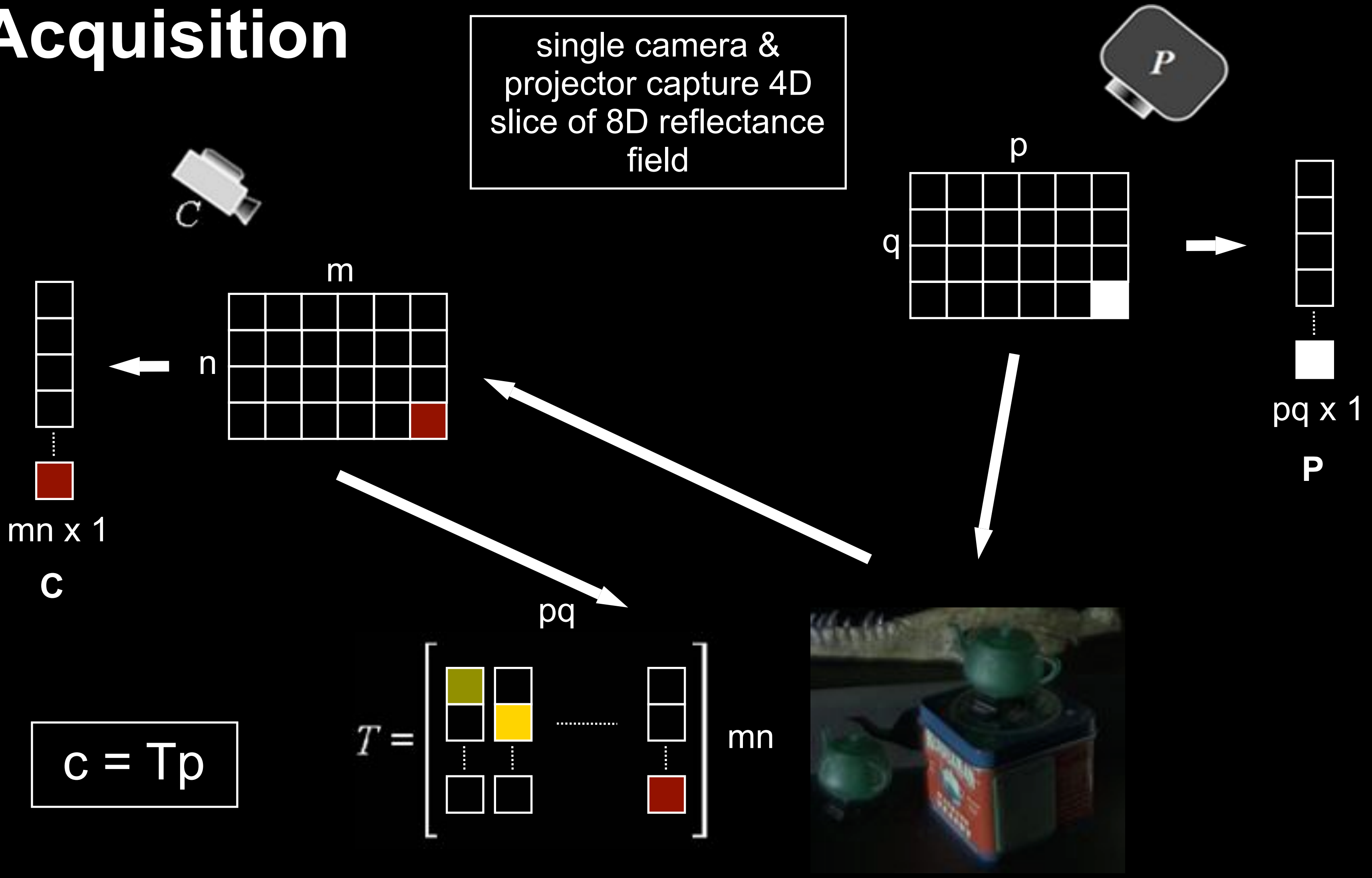
Acquisition



single camera & projector capture 4D slice of 8D reflectance field

c = Tp

Acquisition

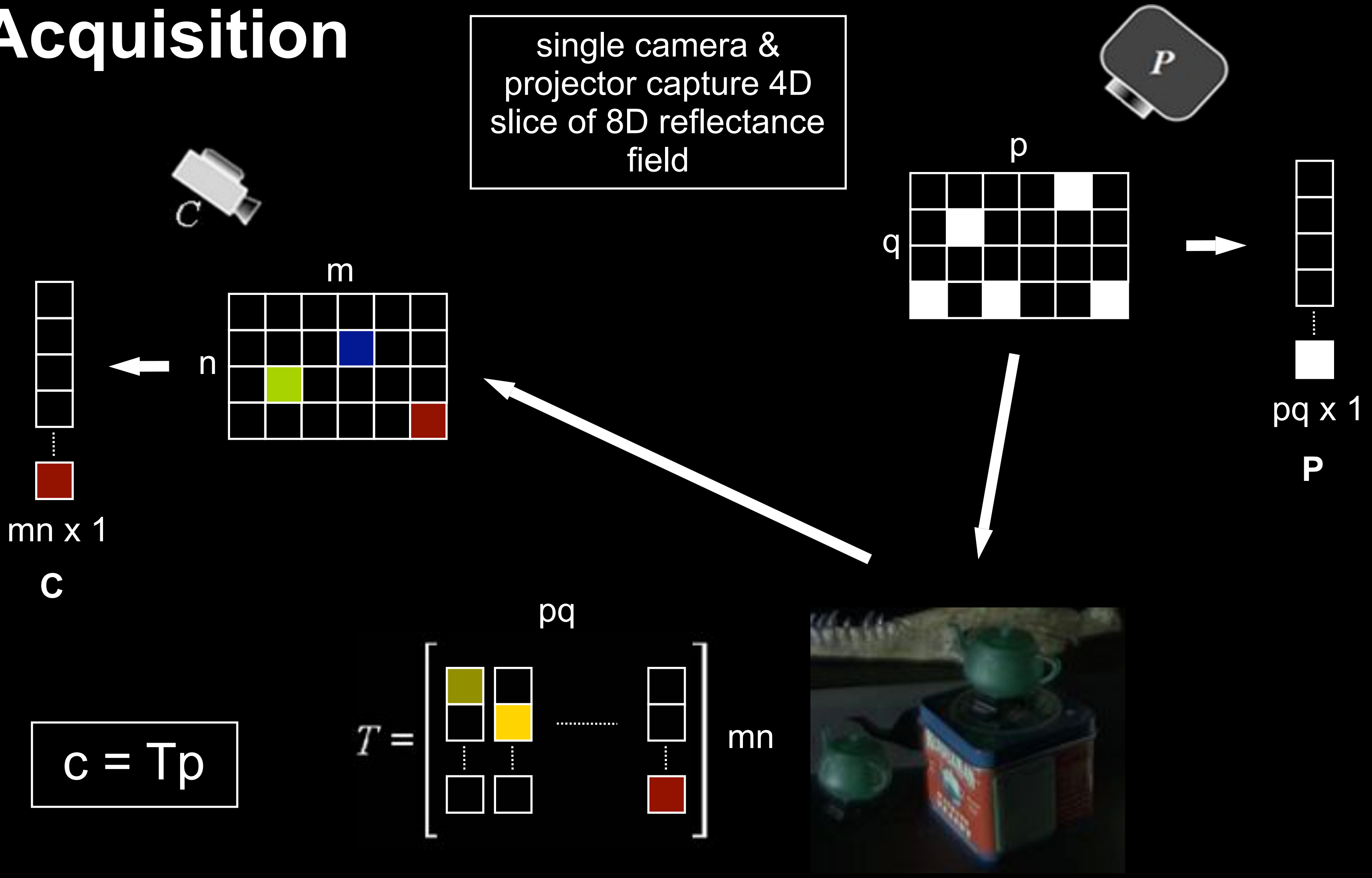


single camera & projector capture 4D slice of 8D reflectance field

$c = Tp$

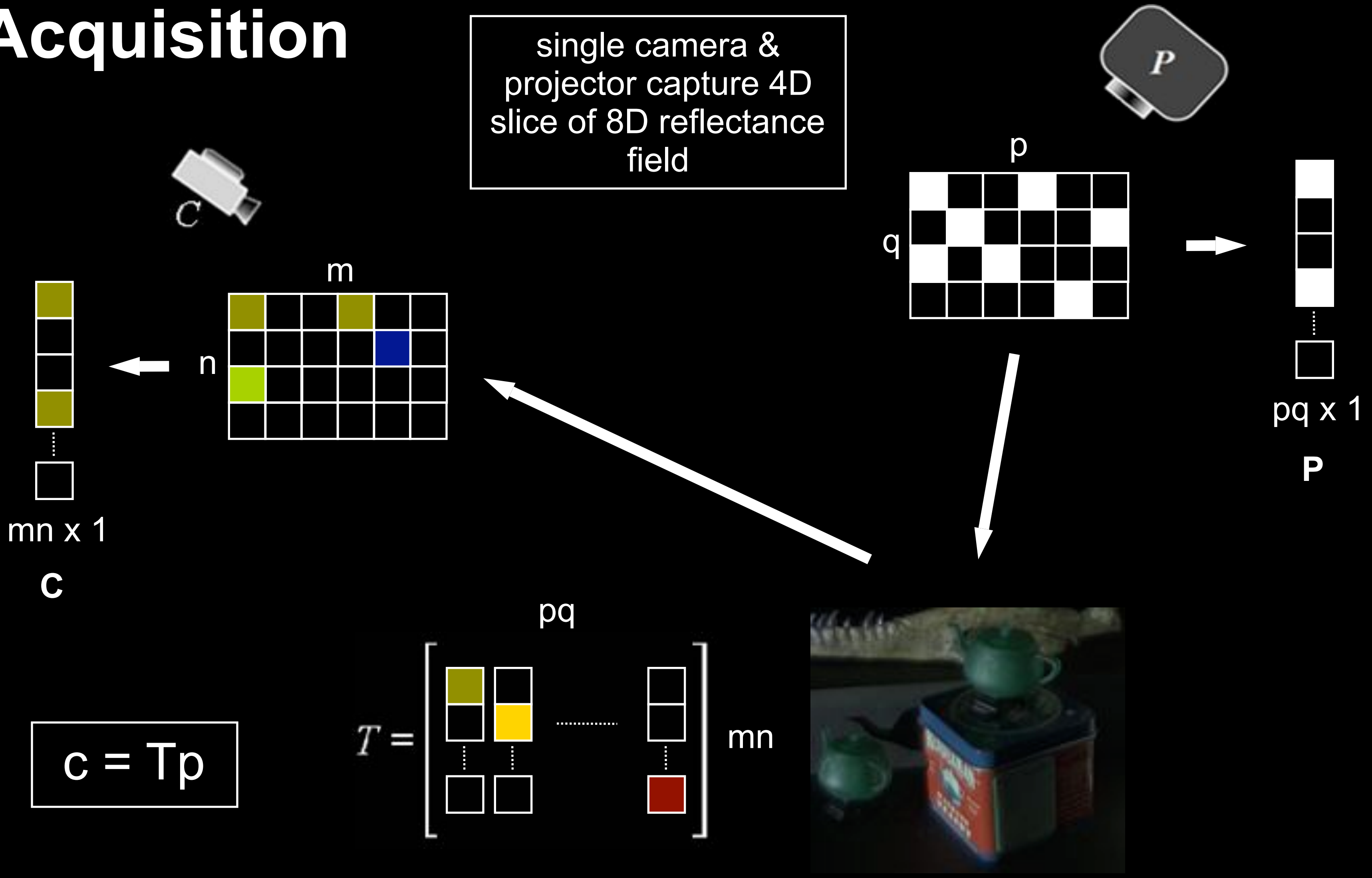
Acquisition

single camera & projector capture 4D slice of 8D reflectance field



c = Tp

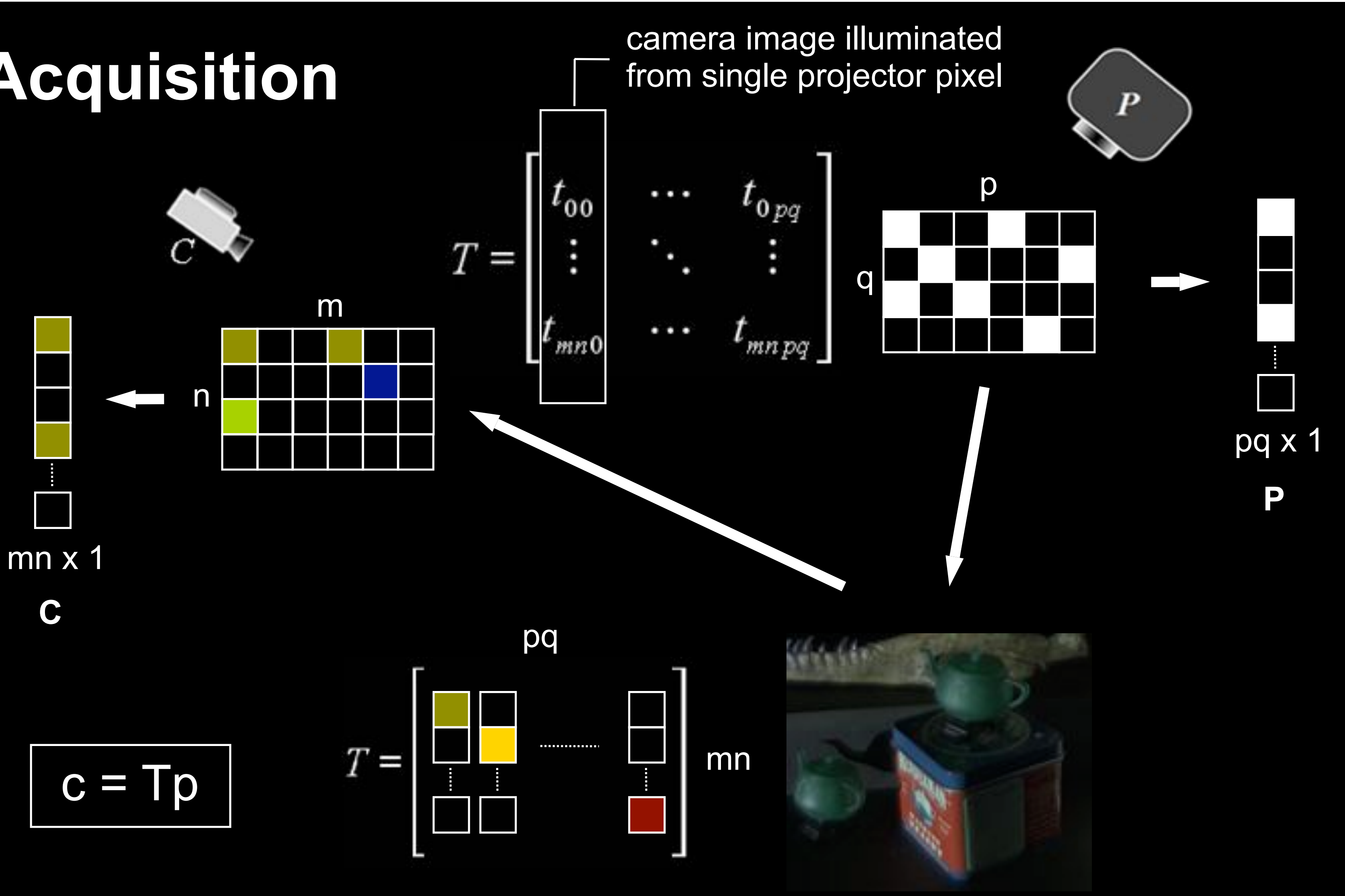
Acquisition



single camera & projector capture 4D slice of 8D reflectance field

$$c = Tp$$

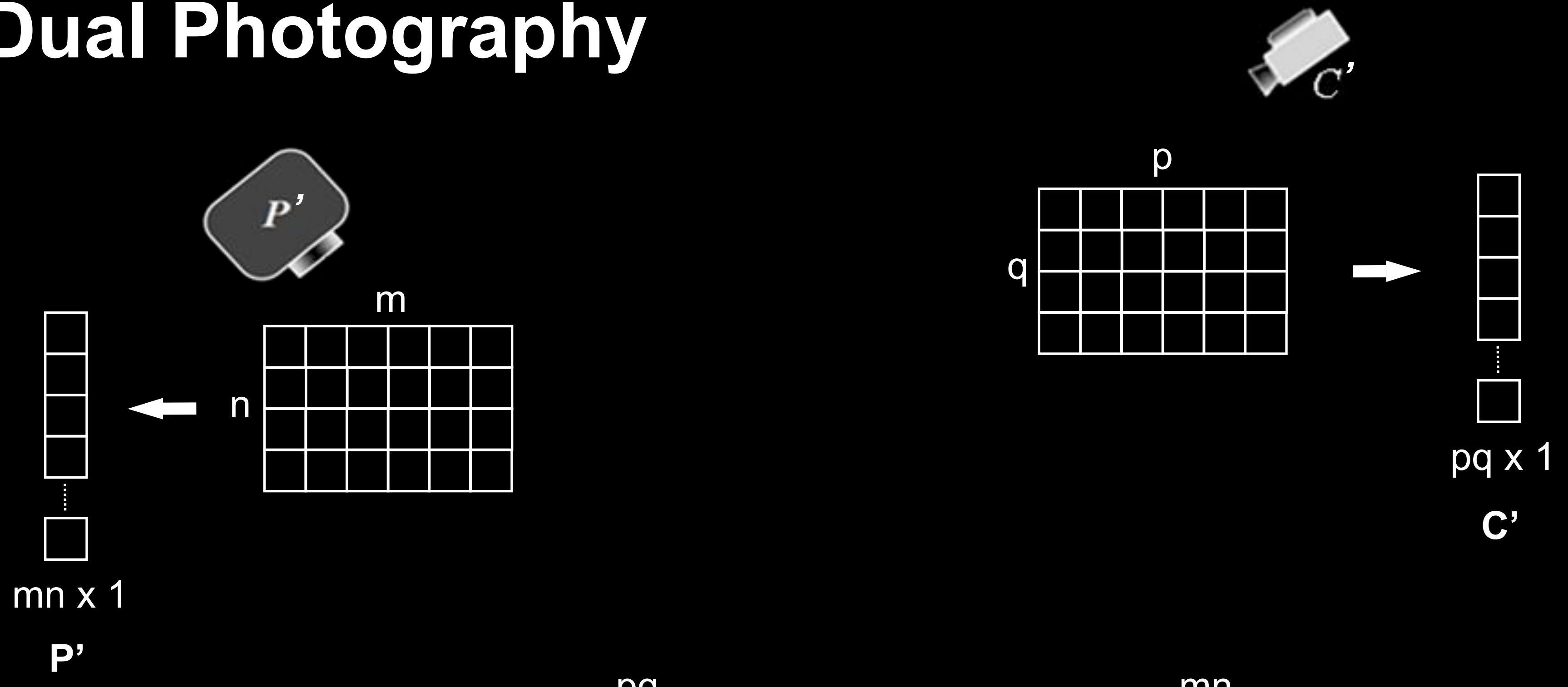
Acquisition

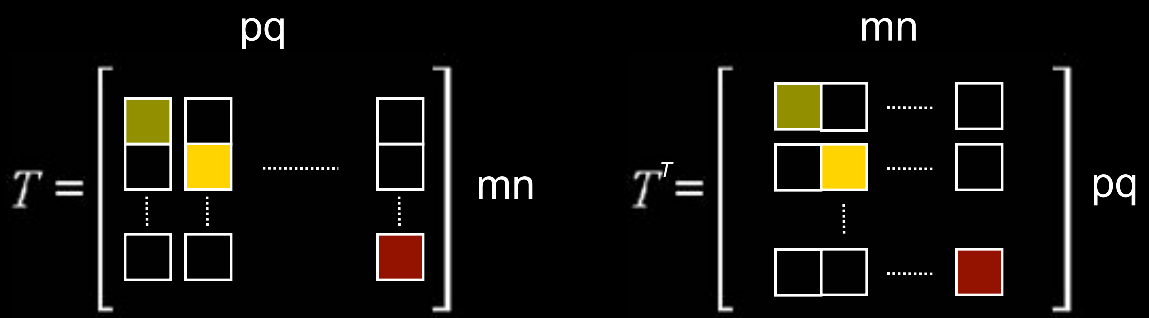


$$T = \begin{bmatrix} t_{00} & \cdots & t_{0pq} \\ \vdots & \ddots & \vdots \\ t_{mn0} & \cdots & t_{mnpq} \end{bmatrix}$$

$$\boxed{c = Tp}$$

Dual Photography





$c = Tp$

$c' = T^T p'$

Dual Photography

Dual Photography

floodlight camera image

Dual Photography

floodlight camera image

projected structured light

Dual Photography

dual image

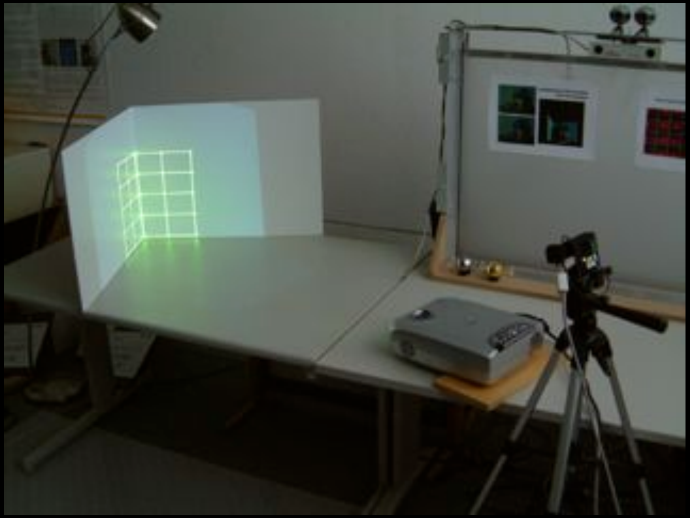
floodlight camera image

projected structured light

more information on
dual photography:
Sen,et al.,
Siggraph'05

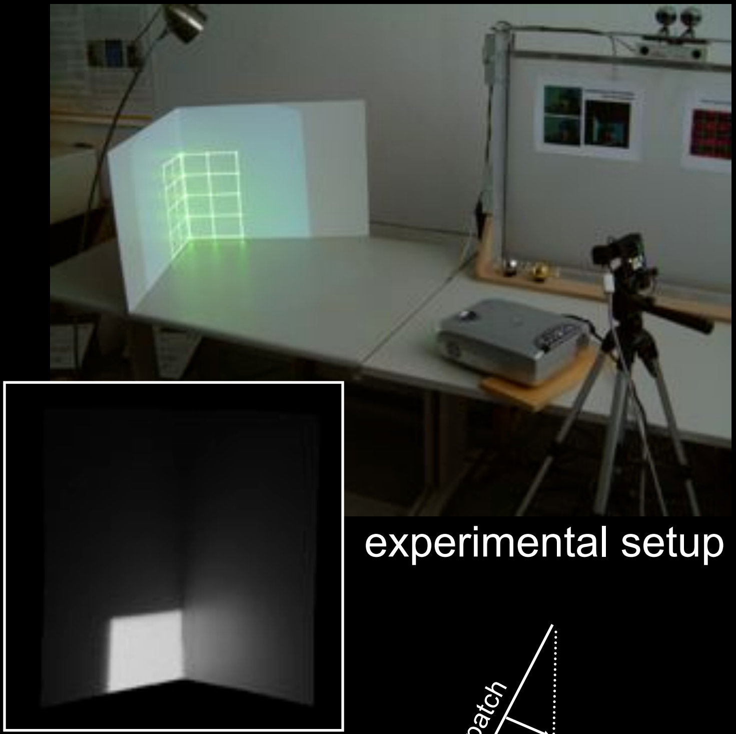
Form-Factors from Light Transport Matrix

Form-Factors from Light Transport Matrix

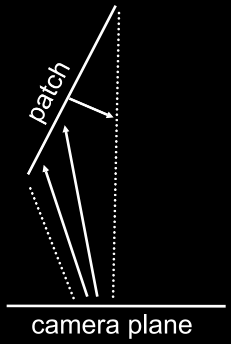


experimental setup

Form-Factors from Light Transport Matrix

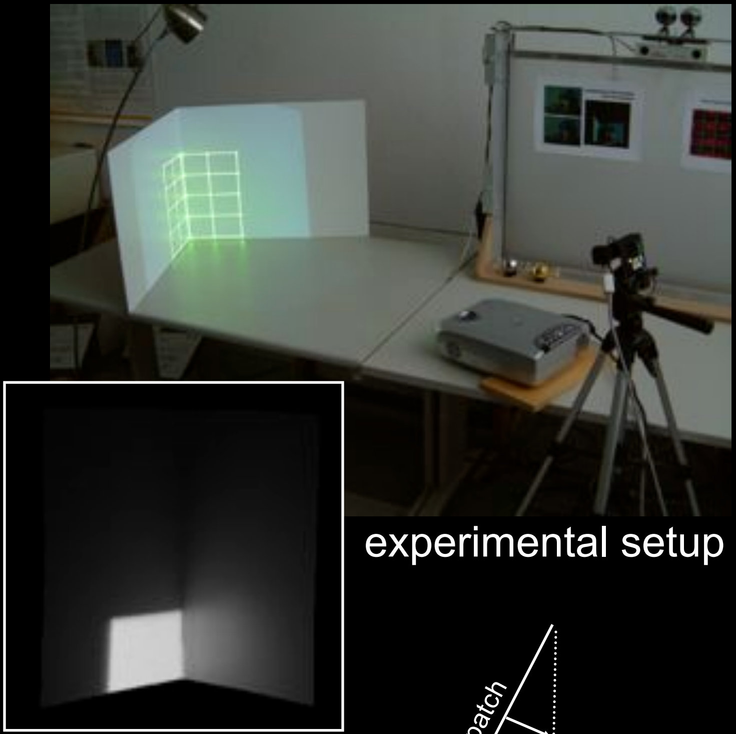


experimental setup



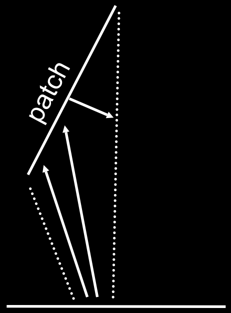
$$M_e = \frac{d^2}{\cos\alpha} L_e$$

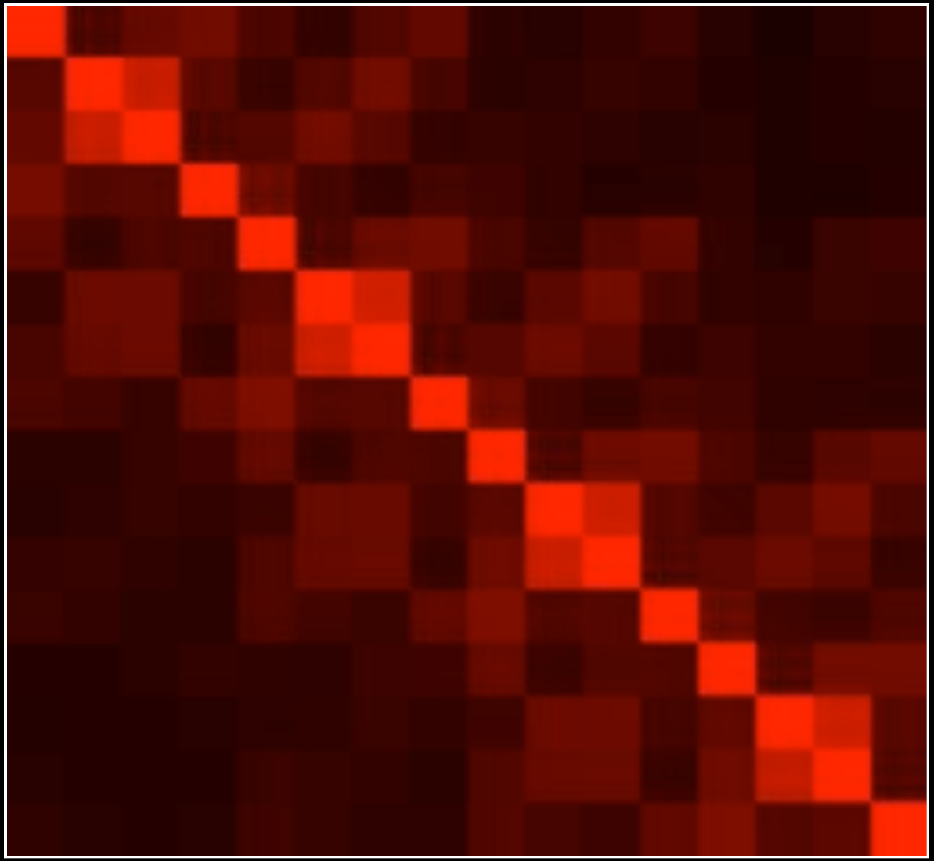
Form-Factors from Light Transport Matrix



experimental setup

$$M_e = \frac{d^2}{\cos\alpha} L_e$$





measured 16x16 form-factor matrix
(computed from light transport matrix)

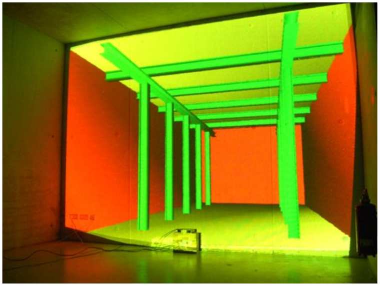
Global Radiometric Compensation

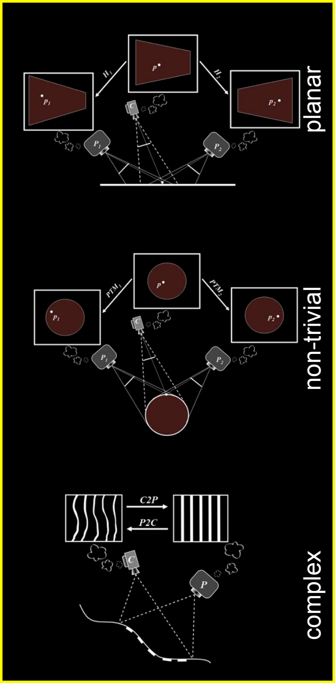
Global Radiometric Compensation

- traditional radiometric compensation requires direct projector-camera pixel correspondence
- include arbitrary global illumination effects using T
- apply inverse light transport $T^{-1}C=P$
- since T is huge, decompose it into clusters and solve in real-time on GPU

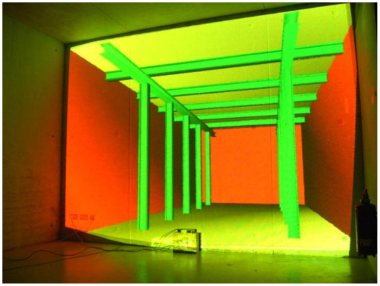
$$\overset{C}{\begin{bmatrix} c_{10} \\ c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \end{bmatrix}} = \overset{T}{\begin{bmatrix} t^4_{10} & t^3_{10} \\ t^4_{11} & t^3_{11} \\ t^4_{12} & t^3_{12} \\ t^4_{13} & t^3_{13} \\ t^4_{14} & t^3_{14} \end{bmatrix}} \overset{P}{\begin{bmatrix} p_4 \\ p_3 \end{bmatrix}} \rightarrow \overset{T^{-1}}{\begin{bmatrix} t^4_{10} & t^4_{11} & t^4_{12} & t^4_{13} & t^4_{14} \\ t^3_{10} & t^3_{11} & t^3_{12} & t^3_{13} & t^3_{14} \end{bmatrix}^{-1}} \overset{C}{\begin{bmatrix} c_{10} \\ c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \end{bmatrix}} = \overset{P}{\begin{bmatrix} p_4 \\ p_3 \end{bmatrix}}$$

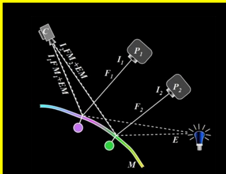
Outlook





geometric warping

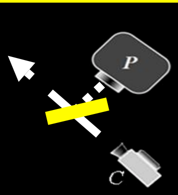




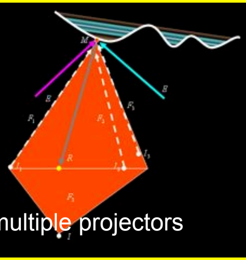
single projector

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

color mixing

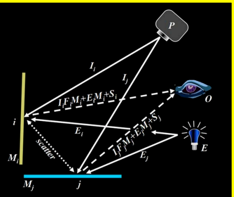


dynamic adaptation



multiple projectors

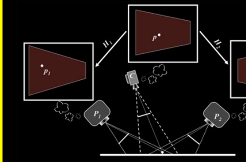
perception



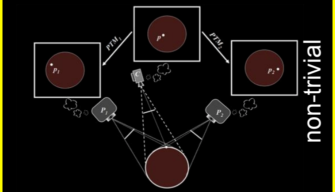
global effects

radiometric compensation

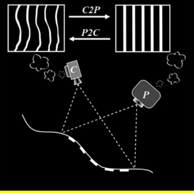
geometric warping



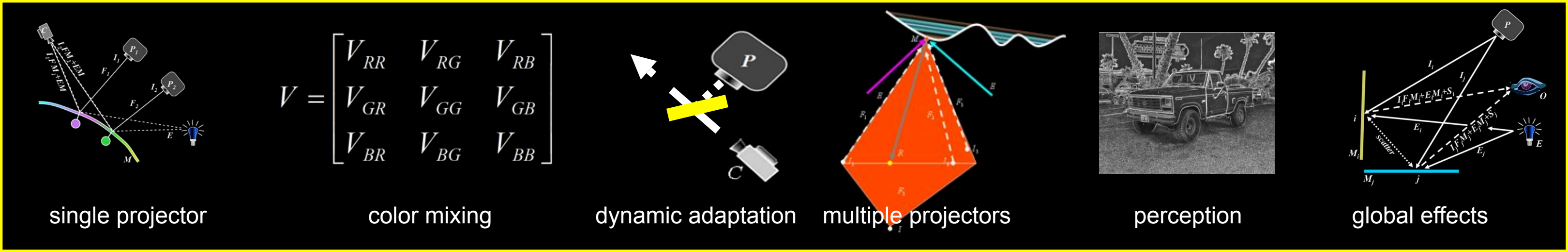
planar



non-trivial



complex



single projector

color mixing

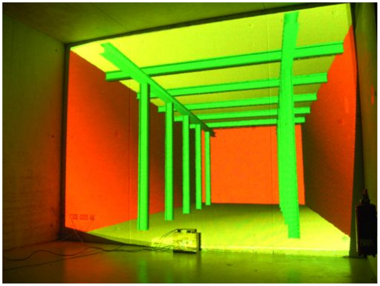
dynamic adaptation

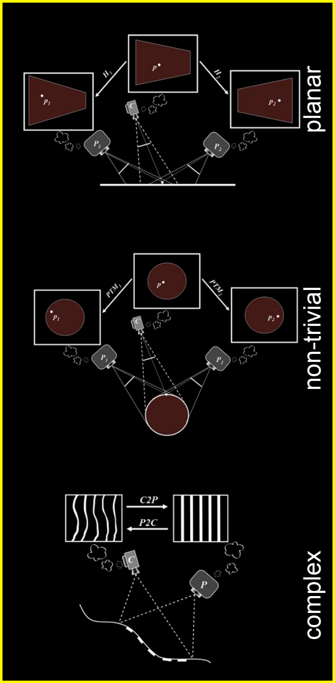
multiple projectors

perception

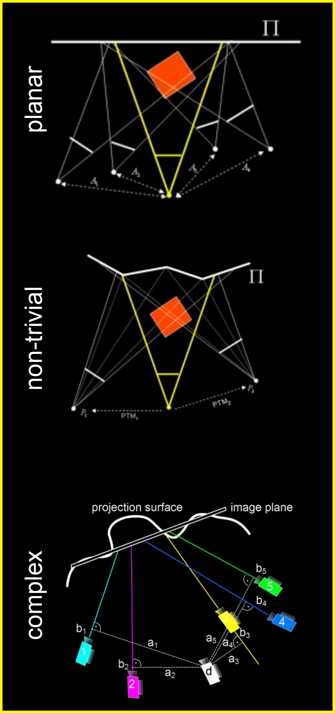
global effects

radiometric compensation

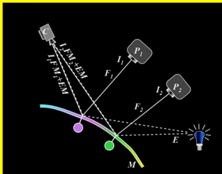




geometric warping



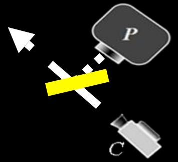
view-dependence



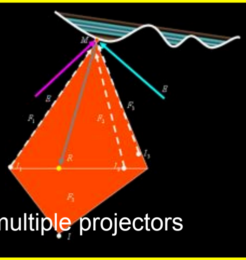
single projector

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

color mixing

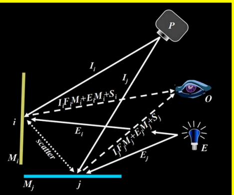


dynamic adaptation



multiple projectors

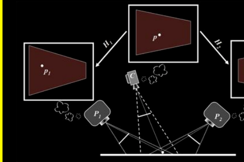
perception



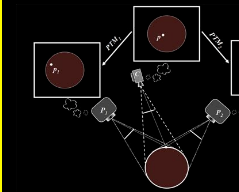
global effects

radiometric compensation

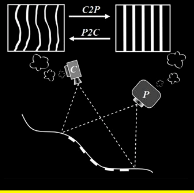
geometric warping



planar

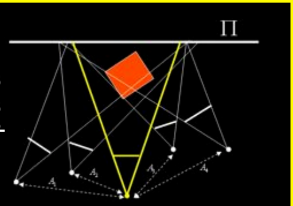


non-trivial

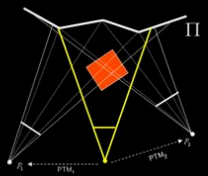


complex

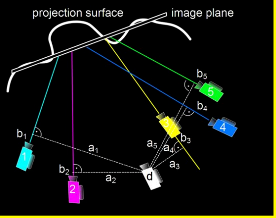
view-dependence



planar

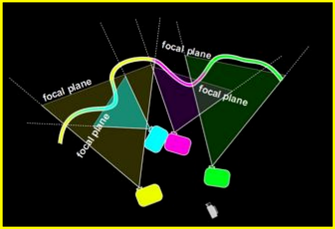


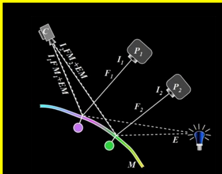
non-trivial



complex

multi-focal projection

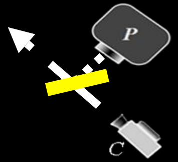




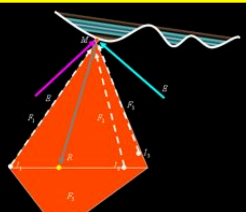
single projector

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix}$$

color mixing

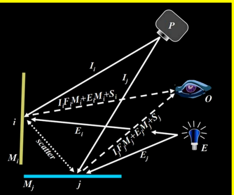


dynamic adaptation



multiple projectors

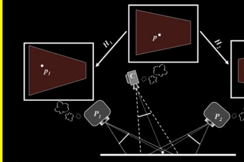
perception



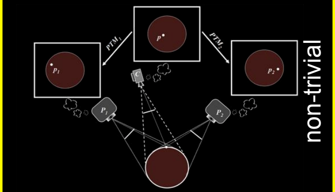
global effects

radiometric compensation

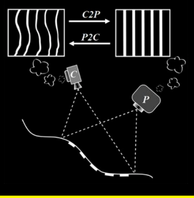
geometric warping



planar

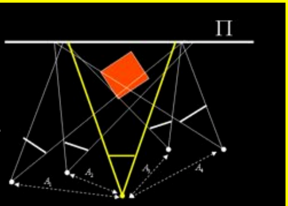


non-trivial

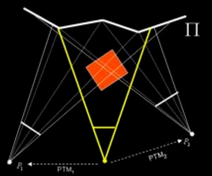


complex

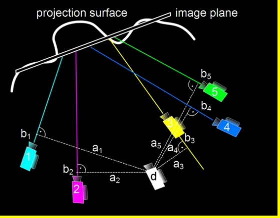
view-dependence



planar

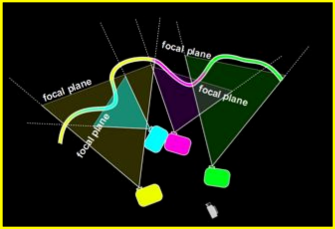


non-trivial



complex

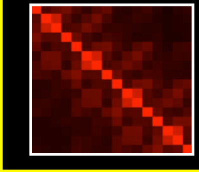
multi-focal projection



light transport

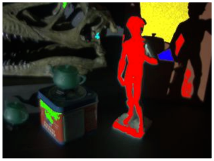
$$T = \begin{bmatrix} t_{00} & \cdots & t_{0pq} \\ \vdots & \ddots & \vdots \\ t_{mn0} & \cdots & t_{mnpq} \end{bmatrix}$$

global parameters



dual image

global rad. comp.



Limitations

Limitations

- technological limitations of projectors:
  - brightness, resolution, focal depth

Limitations

- technological limitations of projectors:
  - brightness, resolution, focal depth → can be solved by using multiple projectors (or wait for better ones)
  - black-level and dynamic range

Limitations

- technological limitations of projectors:
  - brightness, resolution, focal depth → can be solved by using multiple projectors (or wait for better ones)
  - black-level and dynamic range → wait for HDR light-valve or laser projectors
  - size, cost, portability

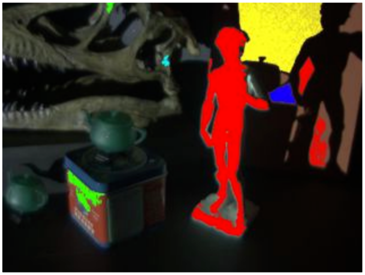
Limitations

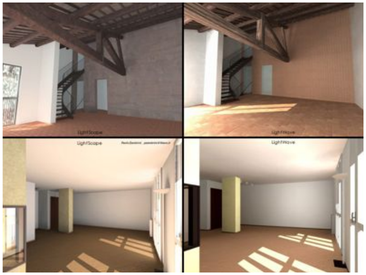
- technological limitations of projectors:
    - brightness, resolution, focal depth → can be solved by using multiple projectors (or wait for better ones)
    - black-level and dynamic range → wait for HDR light-valve or laser projectors
    - size, cost, portability → wait for (good enough) pocket projectors
- technological limitations of cameras:

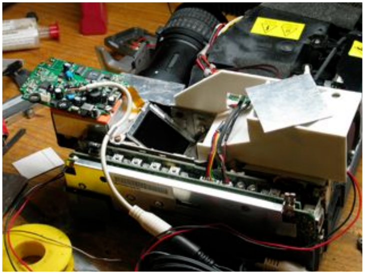
Future Work

Future Work

- new techniques:
    - consider human visual perception
        - spent computational power only on overcoming limitations that can actually be perceived
    - consider global effects
        - inter-reflections, scattering, etc.







Selected Papers on Geometric Correction

Selected Papers on Geometric Correction

Bimber, O., Wetzstein, G., Emmerling, A., & Nitschke, C. (2005). Enabling View-Dependent Stereoscopic Projection in Real Environments. *Proc. of IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR'05)*, 14-23.

Low, K-L., Welch, G., Lastra, A., & Fuchs, H. (2001). Life-Sized Projector-Based Dioramas, *Proc. Symp. Virtual Reality Software and Technology (VRST'01)*, 93-101.

Raskar, R. (1999). Oblique Projector Rendering on Planar Surfaces for a Tracked User. *Proc. of ACM Siggraph'99*, sketch.

Raskar, R., Brown, M.S., Yang, R., Chen, W., Welch, G., Towles, H., Seales, B., & Fuchs, H. (1999b). Multi-projector displays using camera-based registration, *Proc. of IEEE Visualization* (IEEE Viz'99), 161-168.

Raskar, R., Welch, G., Low, K.L. & Bandyopadhyay, D. (2001). Shader Lamps: Animating real objects with image-based illumination. *Proc. of Eurographics Rendering Workshop*, 89-102.

Selected Papers on Radiometric Compensation

Selected Papers on Radiometric Compensation

Bell, I.E. (2003). Neutralizing Paintings with a Projector. *Proc. of SPIE/IS&T*, 5008, 560-568.

Bimber, O., Coriand, F., Kleppe, A., Bruns, E., Zollmann, S., & Langlotz, T. (2005). Superimposing Pictorial Artwork with Projected Imagery. *IEEE MultiMedia*. 12(1), 16-26.

Bimber, O., Grundhöfer, A., Zeidler, T., Danch, D., & Kapakos, P. (2006). Compensating Indirect Scattering for Immersive and Semi-Immersive Projection Displays. *Proc. of IEEE Virtual Reality (IEEE VR'06)*.

Fujii, K., Grossberg, M.D., & Nayar, S.K. (2005). A projector-camera system with real-time photometric adaptation for dynamic environments. *Proc. of Computer Vision and Pattern Recognition (CVPR'05)*, 2, 20-25.

Grossberg, M.D., Peri, H., Nayar, S.K., & Bulhumeur, P. (2004). Making One Object Look Like Another: Controlling Appearance Using a Projector-Camera System. *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04)*, 1, 452-459.

Nayar, S.K., Peri, H., Grossberg, M.D., & Belhumeur, P.N. (2003). A Projection System with Radiometric Compensation for Screen Imperfections. *Proc. of International Workshop on Projector-Camera Systems (ProCams'03)*.

Wang, D., Sato, I., Okabe, T., & Sato, Y. (2005). Radiometric Compensation in a Projector-Camera System Based on the Properties of Human Vision System. *In Proc. of IEEE International Workshop on Projector-Camera Systems (ProCams'05)*.

Yoshida, T., Horii, C. & Sato, K. (2003). A Virtual Color reconstruction System for Real Heritage with Light Projection. *Proc. of Virtual Systems and Multimedia*, 158-164.

Selected Papers Other and Related Techniques

Selected Papers Other and Related Techniques

Bimber, O. & Emmerling, A. (2006). Multi-Focal Projection: A Multi-Projector Technique for Increasing Focal Depth. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*.

Brown, M., Majumder, A., and Yang, R. (2005). Camera-Based Calibration Techniques for Seamless Multi-Projector Displays. I*EEE Transactions on Visualization and Computer Graphics (TVCG)*, 11(2), 193-206.

Cotting, D., Naef, M., Gross, M., & Fuchs, H. (2004). Embedding Imperceptible Patterns into Projected Images for Simultaneous Acquisition and Display. *Proc. of IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)*, 100-109.

Ehnes, J., Hirota, K., & Hirose, M. (2004). Projected Augmentation – Augmented Reality using Rotatable Video Projectors. *Proc. of IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)*, 26-35.

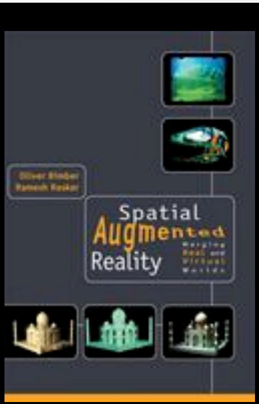
Underkoffler, J., Ullmer, B. & Ishii, H. (1999). Emancipated pixels: real-world graphics in the luminous room. *Proc. of ACM Siggraph*, 385-392.

Levoy, M., Chen, B., Vaish, V., Horowitz, M., McDowall, I., and Bolas, M. (2004) Synthetic Aperture Confocal Imagining, Proc. of ACM Siggraph'04, pp. 825-834.

Sen, P., Chen, B., Garg, G., Marschner, S.R., Horowitz, M., Levoy, M., and Lensch, H.P.A (2005)., Dual Photography, *Proc. of ACM Siggraph*, pp. 745-755

Thank you!

www.uni.weimar.de/medien/AR



Bimber, O. & Raskar, R. *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A K Peters LTD (publisher), ISBN: 1-56881-230-2.

Thank you!

www.uni.weimar.de/medien/AR

This project is supported by the Deutsche Forschungsgemeinschaft (DFG) under contract number PE 1183/1-1.