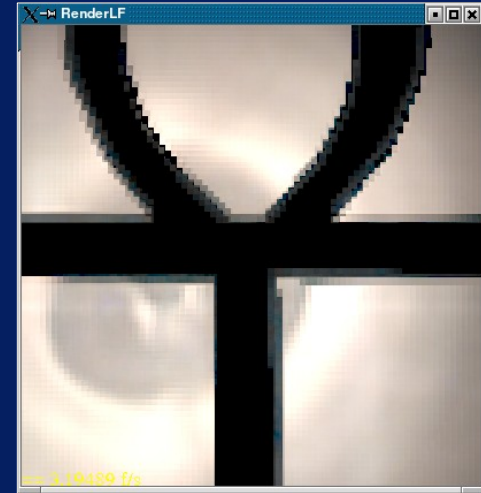
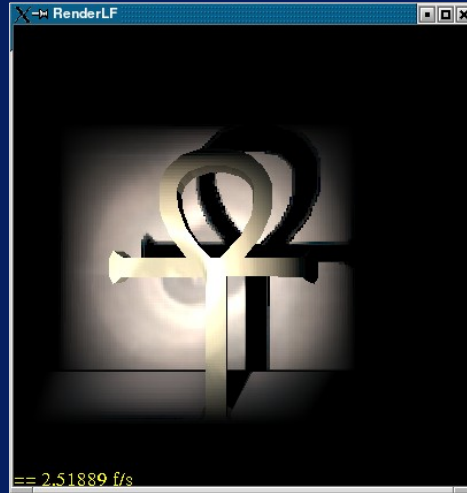


Interactive Visualization of Complex Real-World Light Sources



Xavier Granier⁽¹⁻²⁾ – Michael Goesele⁽³⁾
Wolfgang Heidrich⁽²⁾ – Hans-Peter Seidel⁽³⁾

(1)  **IPARLA/INRIA – LaBRI (Bordeaux)**

(2)  **University of British Columbia**

(3)  **MPI Informatik**

Overview

Real Light-Sources Acquisition

Needed for realistic rendering and simulation

Recent Approach

Optically prefiltered light field

Integration to global illumination solution

Overview

Real Light-Sources Acquisition

Needed for realistic rendering and simulation

Recent Approach

Optically prefiltered light field

Integration to global illumination solution

Goal : Interactive visualization of this models

Overview

Real Light-Sources Acquisition

Needed for realistic rendering and simulation

Recent Approach

Optically prefiltered light field

Integration to global illumination solution

Goal : Interactive visualization of this models

Approximations

Shift Invariant / Direct Lighting

Hardware Implementation

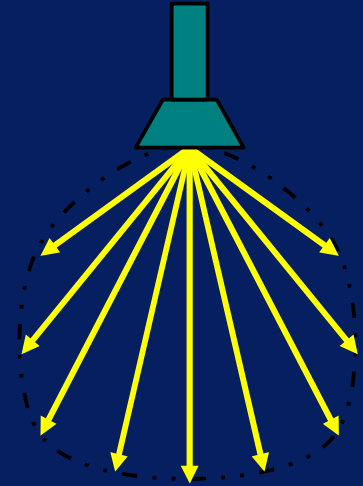
Light Source Models (1/2)

Goniometric diagrams

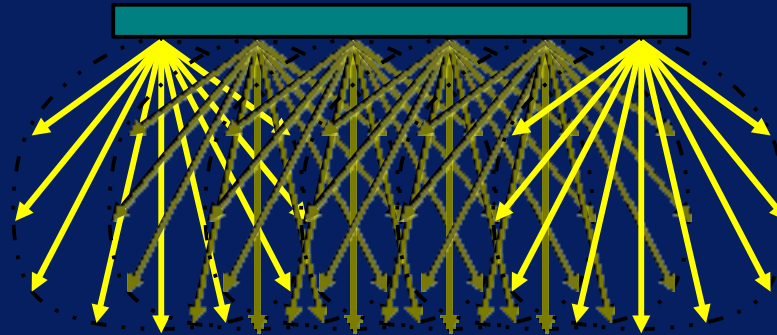
Point light source

Only far field

Directional information only



Multiple goniometric diagrams



Light Source Models (2/2)

Image based approaches

Near Field [*Ashdown93-97,...*] and Light-Field

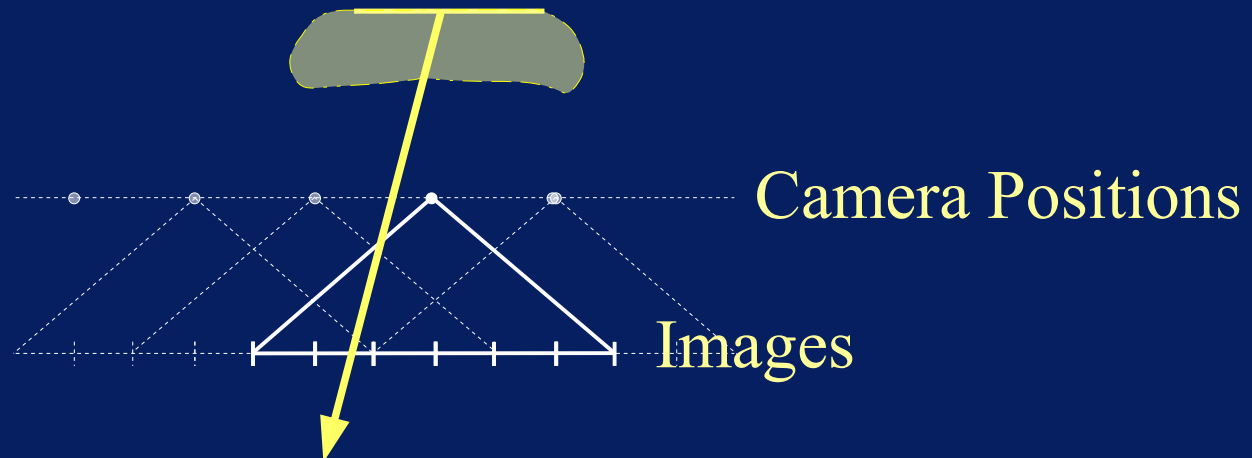
Light represented as an array of images

Reconstruction by interpolation between coefficients

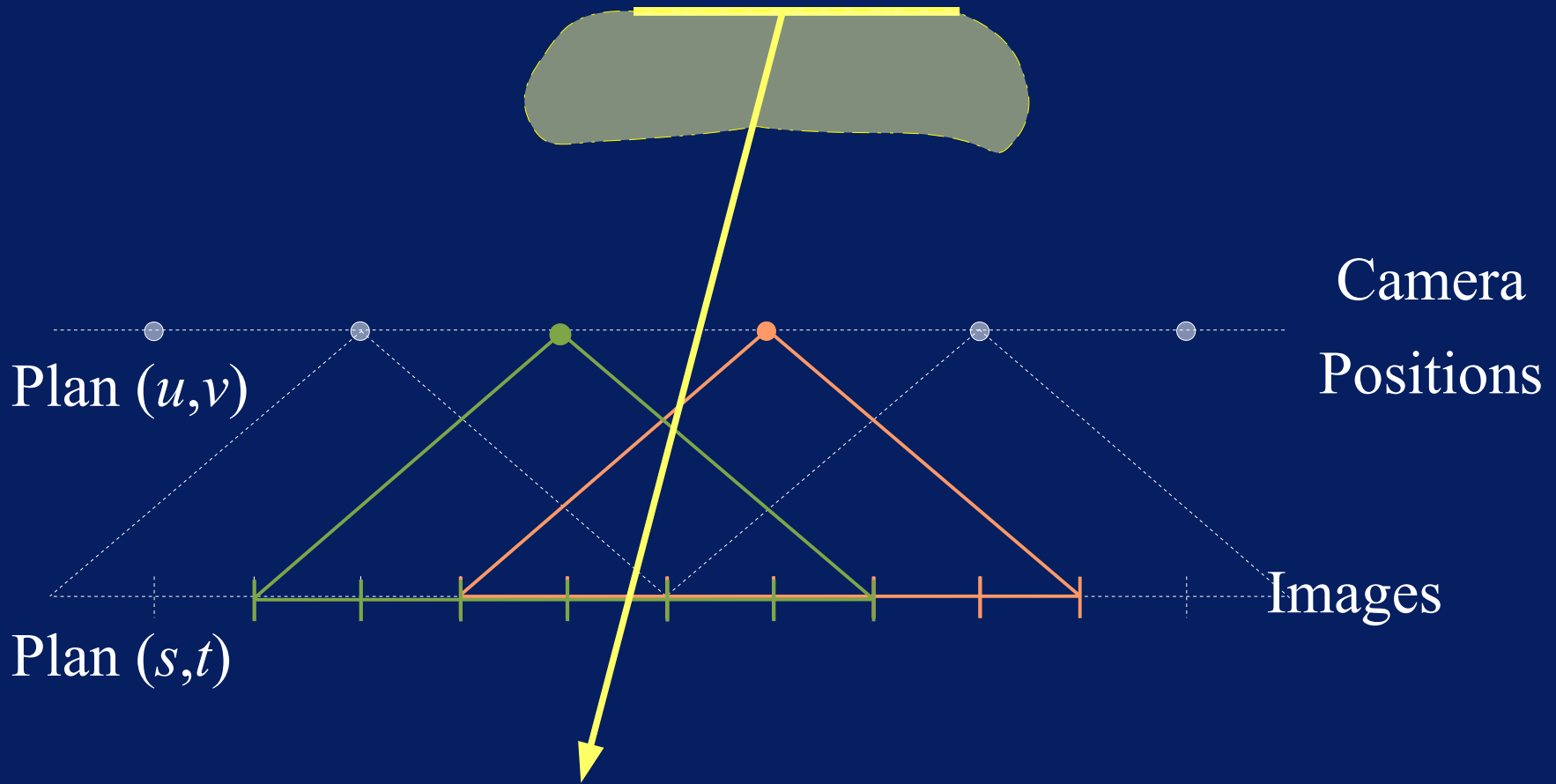
Aliasing and filtering problem [*Levoy96,...*]

Can not capture high variations

Light Source



Light-Field / Reconstruction



$$L(u, v, s, t) = \sum_{ijkl} L_{ijkl} \Psi_{ijkl}(u, v, s, t)$$

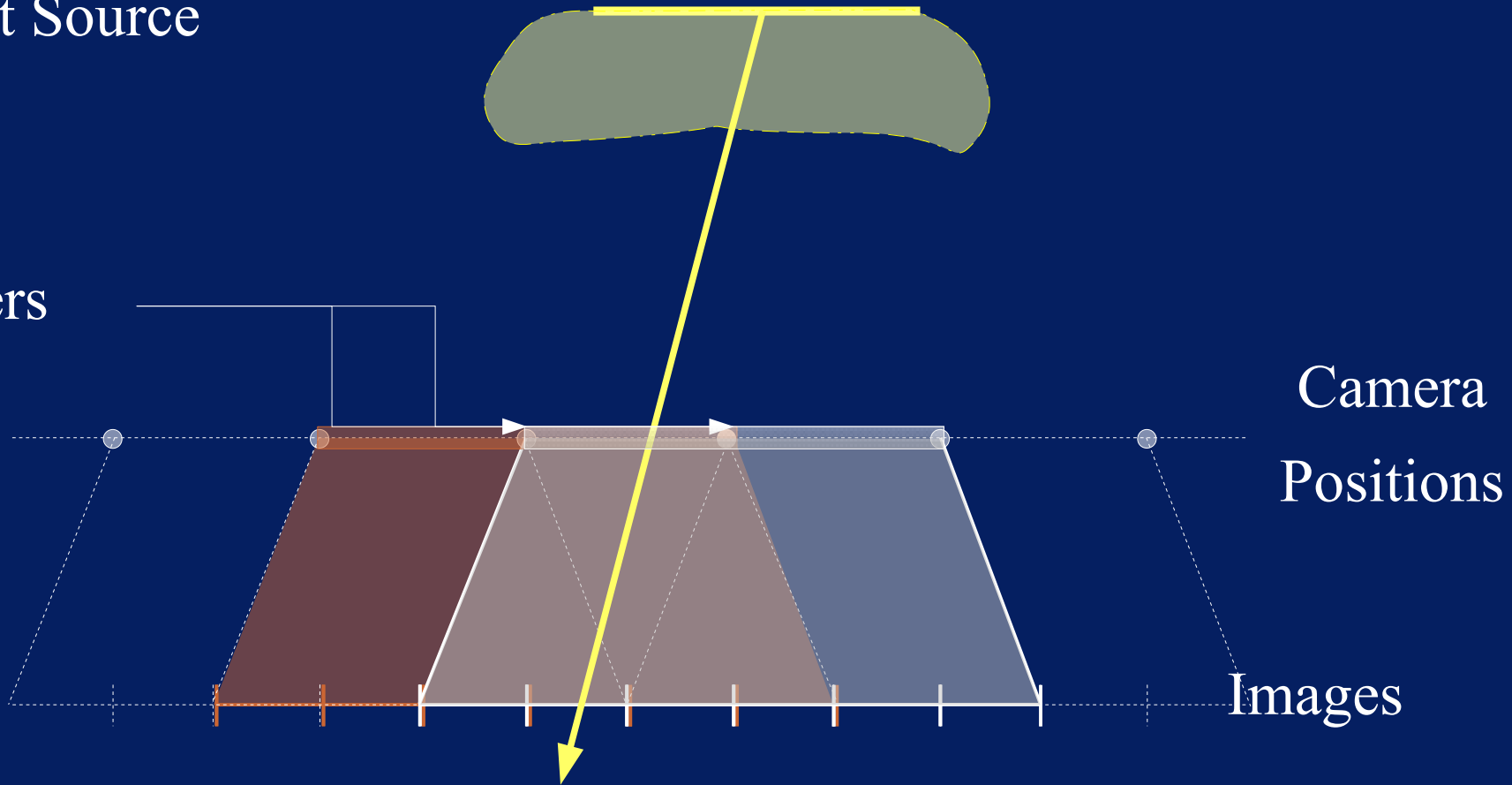
Our Approach : Filtering

Light Source

Filters

Camera
Positions

Images

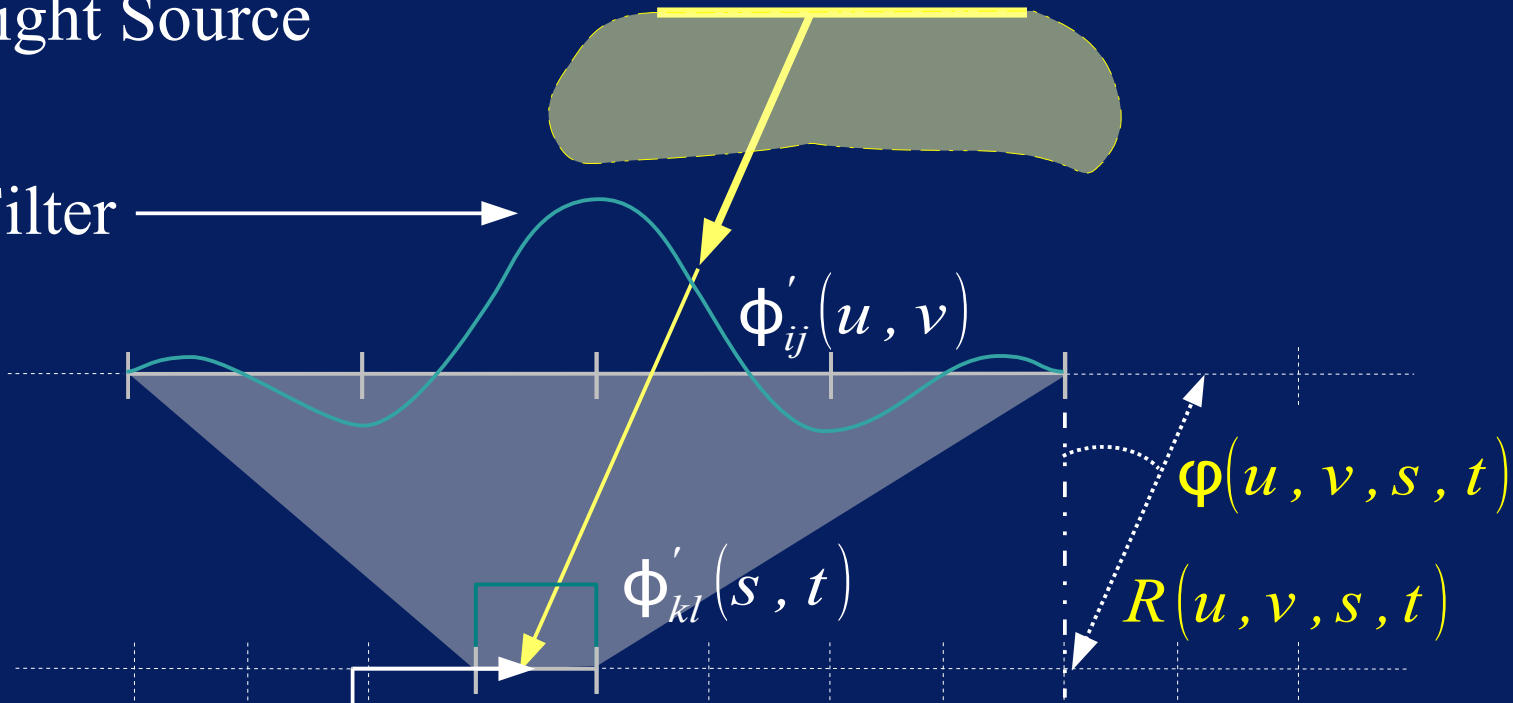


Array of optically filtered images

Configuration

Light Source

Filter



$$E_{ijkl} = \int_{uvst} \frac{\cos^2 \varphi}{R^2} \phi'_{ij}(u, v) \phi'_{kl}(s, t) L(u, v, s, t)$$

General Approach

$$L(u, v, s, t) = \sum_{ijkl} L_{ijkl} \Psi_{ijkl}(u, v, s, t)$$

Base 4D

$$\Psi_{ijkl}(u, v, s, t) = \frac{R^2}{\cos^2 \varphi} \phi_{ij}(u, v) \phi_{kl}(s, t)$$

Use dual basis as a filter

$$\langle \phi'_{mn} | \phi_{ij} \rangle = \delta_m^i \delta_n^j$$

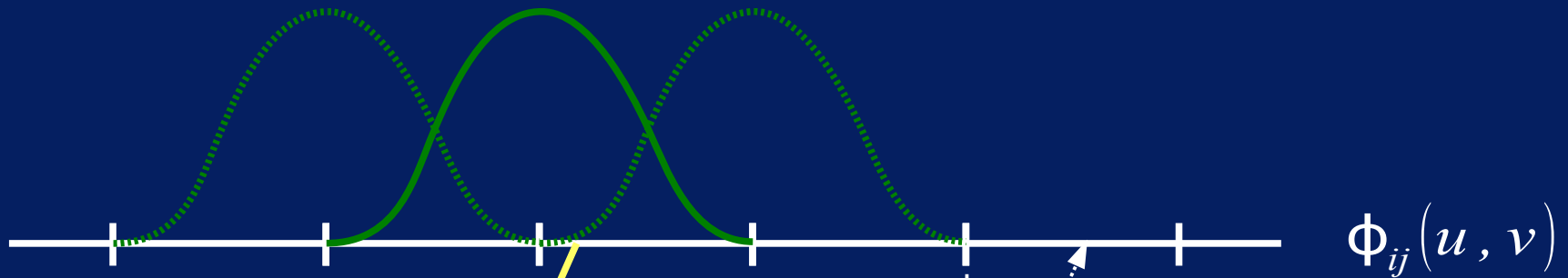
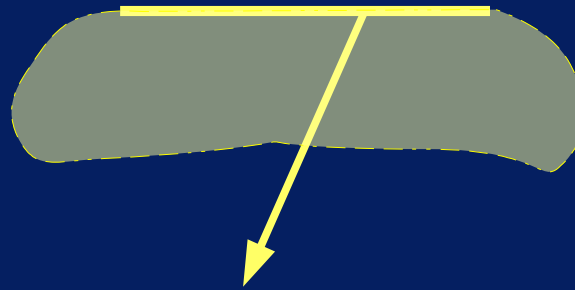
Projection

$$E_{ijkl} = L_{ijkl}$$

Quadratic Basis

Light Source

$$L(x, y, \theta, \phi)$$

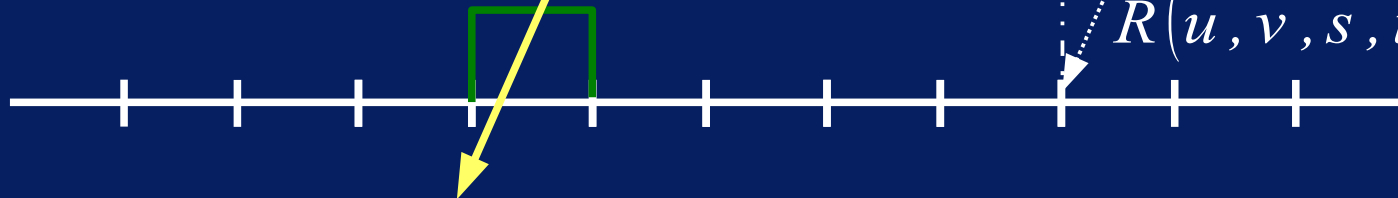


Filter Support

$$\varphi(u, v, s, t)$$

$$R(u, v, s, t)$$

$$\Phi_{kl}(s, t)$$



Measure/Image Support

Approximations

Faster rendering

Texture based

Shift Invariant

Remove geometric term in the reconstruction function

Direct lighting

Use each measure as a texture

One lighting evaluation for each measure

Shift-Invariant Approximation

Constant geometric term (error $\leq 8\%$)

$$E_{ijkl} = \left(\frac{\cos^2 \varphi}{R^2} \right)_{ijkl} \int_{uvst} L(u, v, s, t) \phi'_{ij}(u, v) \phi'_{kl}(s, t)$$

$$E'_{ijkl} = \int_{uvst} L(u, v, s, t) \phi'_{ij}(u, v) \phi'_{kl}(s, t)$$

New reconstruction function

$$\Psi^a_{ijkl}(u, v, s, t) = \phi_{ij}(u, v) \phi_{kl}(s, t)$$

$$L(u, v, s, t) = \sum_{ij} \phi_{ij}(u, v) \sum_{kl} \phi_{kl}(s, t)$$

Lighting Approximation

Constant value over a basis support for :

Geometric term

Visibility

BRDF

$$I(x, \mathbf{o}) = \sum_{ij} \Gamma_{ij}(x, \mathbf{o}, \mathbf{i}) V_{ij}(x) G_{ij}(x) \overline{T}_{ij}(x)$$

Average energy through a basis

Similar to a mip-map

Texturing + Multi-Pass

Hardware Implementation (1/2)

For each measure

Shadow map (ie Visibility) $V_{ij}(x)$

Lighting $\Gamma_{ij}(x, \mathbf{o}, \mathbf{i}) G_{ij}(x) \overline{T}_{ij}(x)$

2N rendering passes

Requirement

Floating-point textures

Floating-point buffer for the sum

Hardware Implementation (2/2)

Combine 3 lighting computation in 1 pass

- Reduce to $3N/4$ rendering passes

- Using current hardware programmability

Reducing data transfert

- Save 3 depth maps as a 8bit RGB texture

- Save current solution as a 8bit RGBA texture

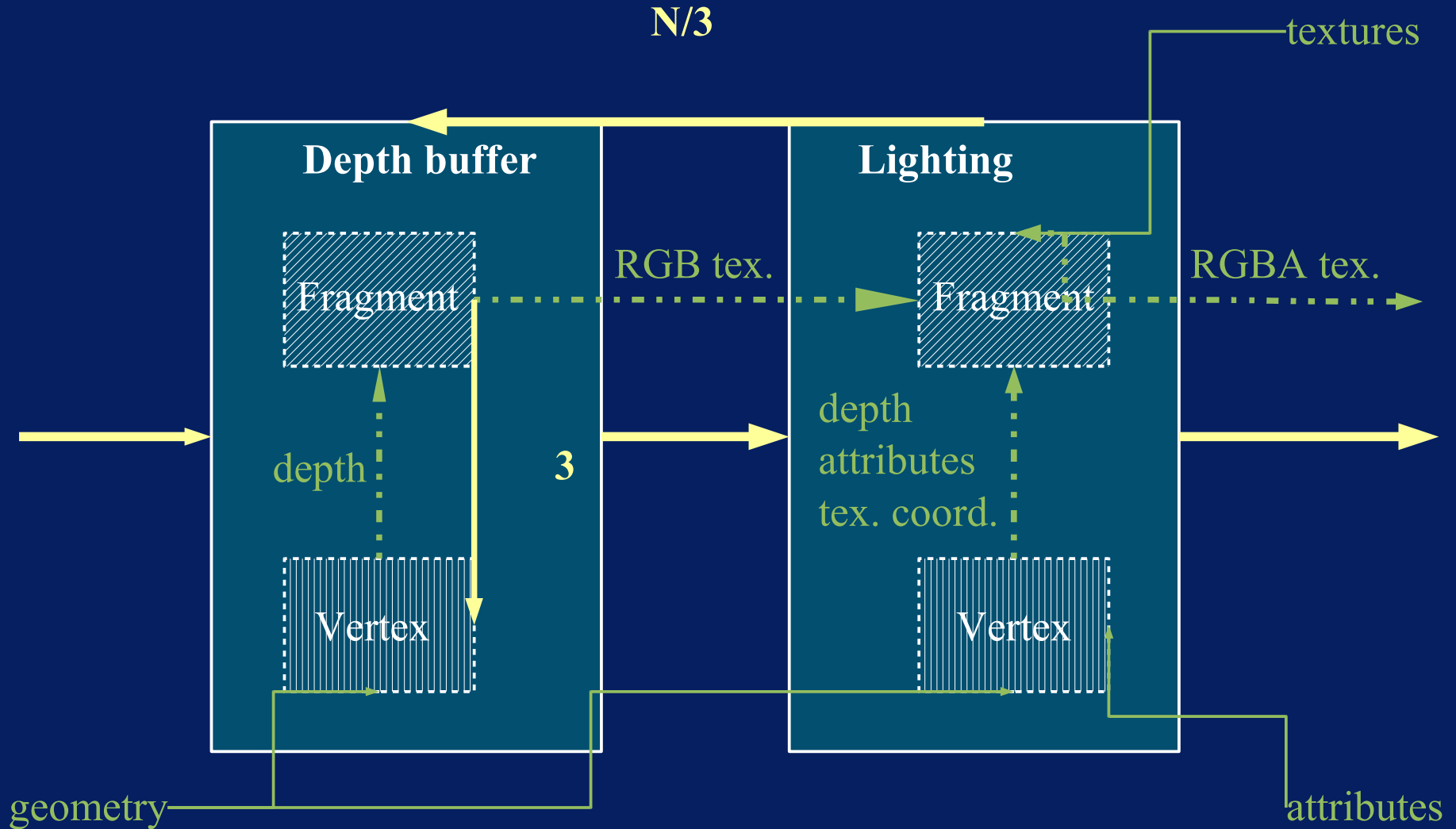
sRGBE format

- Similar to RGBE

- Store negative values

- Adapted to hardware limitations

Data Flow & Rendering Path



sRGBE

Similar to RGBE

Compression

Exponent on the absolute values

$$e_b = \log_2 \left(\max(|R_f|, |G_f|, |B_f|) \right) + 2$$

Scale the coefficient to $[0,1]$

$$[R, G, B] = 2^{-e_b} [R_f, G_f, B_f] + 0.5$$

Scale the exponent to $[0,1]$

$$E = \frac{e + 126}{255}$$

Decompression

$$[R_f, G_f, B_f] = 2^{255 \cdot E - 126} \left([R, G, B] - 0.5 \right)$$

Tests Configurations

2 tests scenes (different polygon complexity)

500/ 8,000 polygons

3 Light sources

5x5 / 7x7 / 9x7 images (300x300 pixels)



Results

one iteration	Size	40	500	1000	5000	10000
	5x5	1.7 (0.3)	1.7 (0.3)	1.7 (0.3)	1.7 (0.3)	1.7 (0.3)
Maglite	9x7	1.2 (1.4)	1.2 (1.4)	1.2 (1.4)	1.2 (1.4)	1.2 (1.4)

GeForce FX 5800 Ultra

Analysis

Effects reproduced

Soft shadows / Near field

Blocky appearance

No interpolation for floating point textures

No mipmap for floating point textures

Classical depth map error

Influence on the frame rate

Main : size of the datas

Geometric complexity :

20 x more polygons => 2 x slower

Conclusion

Visualization of Complex Light Sources

Based on our light-source representation

Approximations

Shift Invariant

Texture-based lighting

Multi-pass

Combine 3 passes in 1

Efficient data representation

Up to 3.2 fps

Futur Work

Better quality

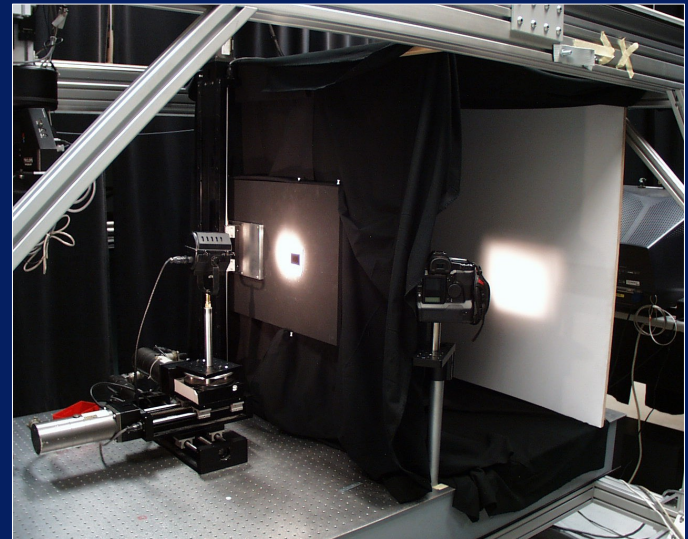
Better shadow algorithms

Summed-area tables (mipmap)

Better frame rate

Integration into acquisition setup

UBC Active Measurement
facility (ACME) [*Pai 2001*]



Thanks to

UBC's Institute of Applied Mathematics

This work was funded by

PIMS Post-doctoral Fellowship program

BC Advanced Systems Institute

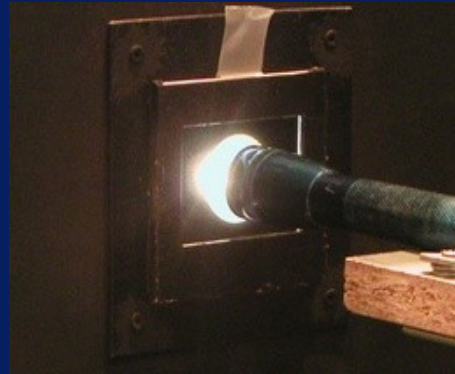
DFG Schwerpunktprogramm V3D2

All the people from the IMAGER/UBC lab !!!

Light Source Acquisition

Goals:

capture near field of a light source
sampling with correct pre-filtering
enable efficient rendering



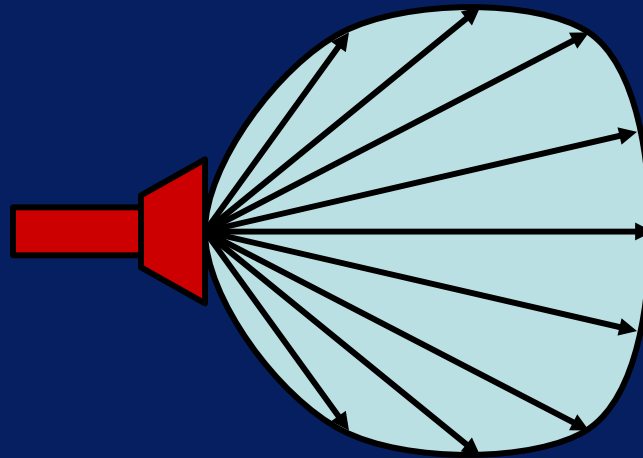
Near Field vs. Far Field

far field assumption

only light direction

all light is emitted from a single point

intensity approximation valid for
distances $> 5x-20x$ emitting diameter



Near Field vs. Far Field

near field data

origin and direction of light
(4D data)

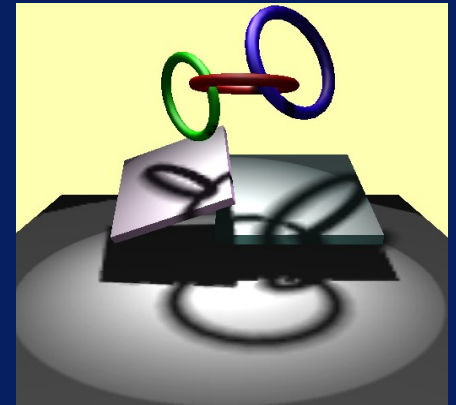
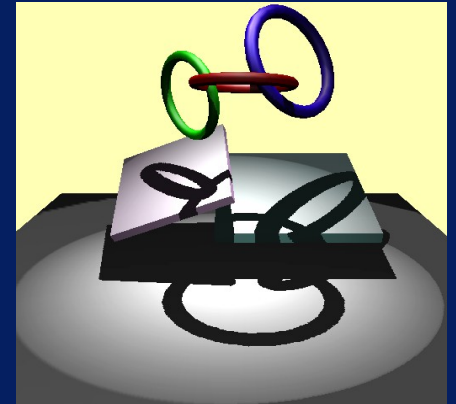
soft shadows

distance effects

change of light pattern

slide projector in

focus/out of focus







synthetic slide projector data set focused at 3m
acquired at 90 cm



Sampling Issues

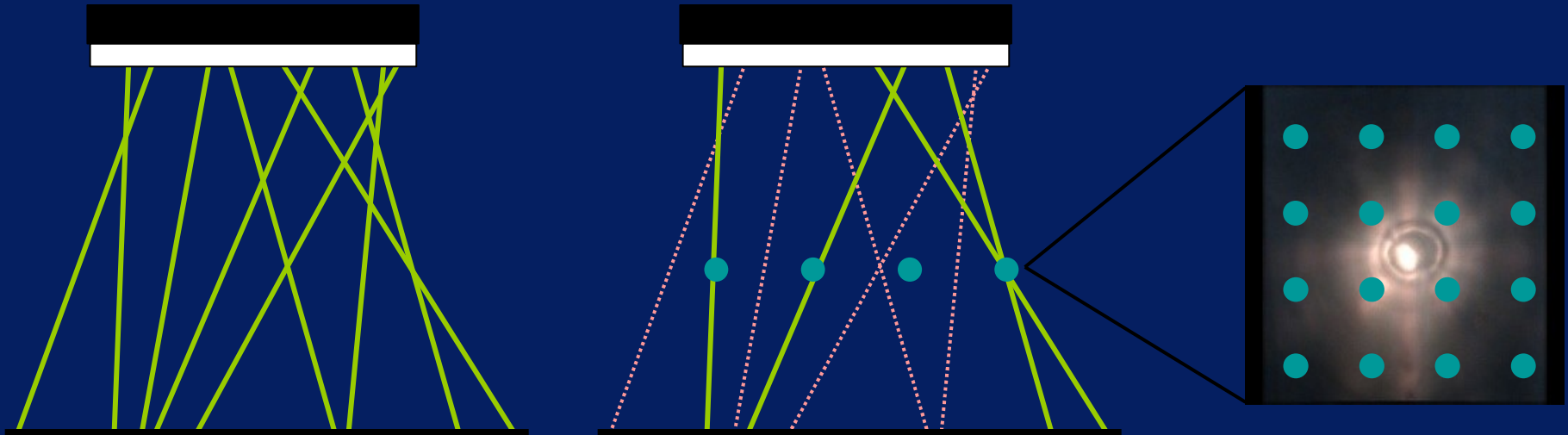
How to sample a light source?

point sampling

miss a lot of rays

potential of aliasing

see also [Levoy and Hanrahan 1996]



Sampling Issues

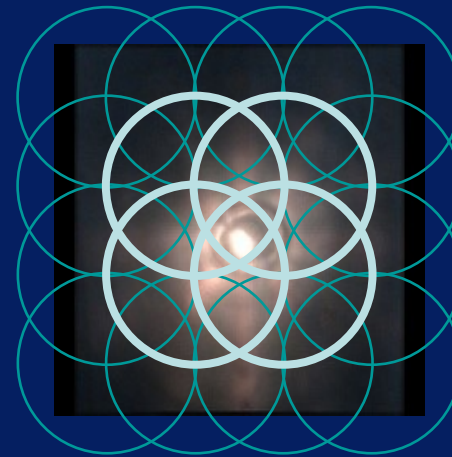
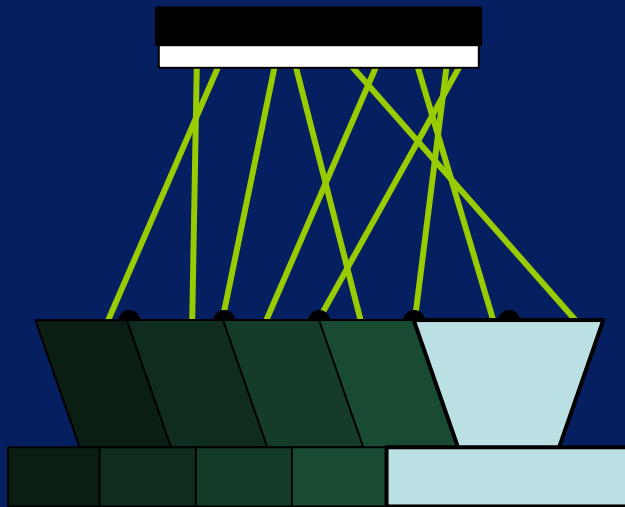
camera and lens system

aperture 2 time sample spacing

[Halle 1994]

get all rays multiple times

sampling behavior not well defined



Light Source Models

“light source light fields”

can capture near and far field

near field photometry

[Ashdown 1993, 1995]

canned light sources

[Heidrich et al. 1998]

capturing incident light field

[Unger et al. 2003]

Our Contributions

Acquisition:

optical filtering before sampling

projection of light field into 4D function basis

low pass filtering in spatial domain

avoids aliasing

Rendering:

importance sampling of light field

constant time particle emission

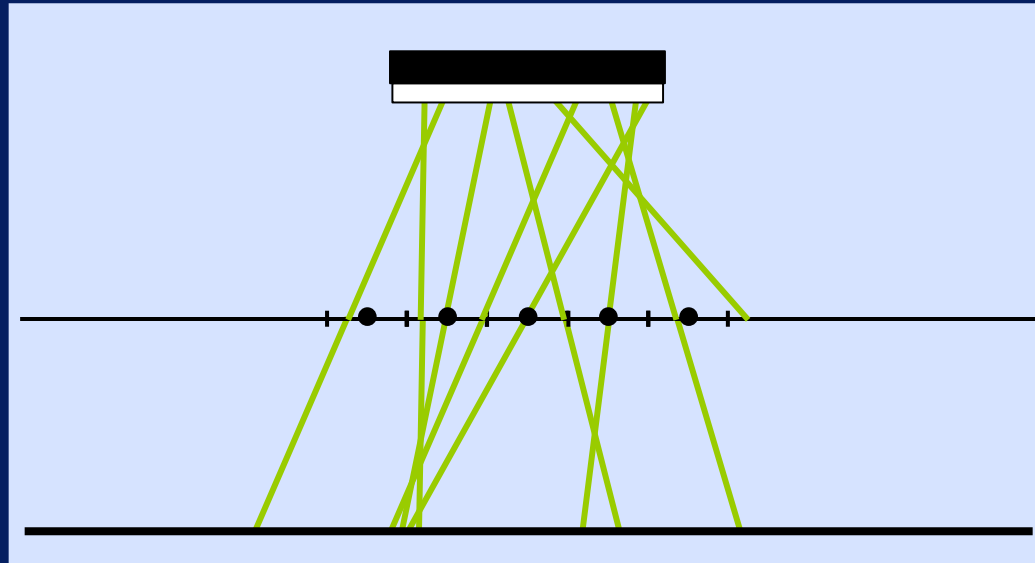
Sampling Issues

box filtering

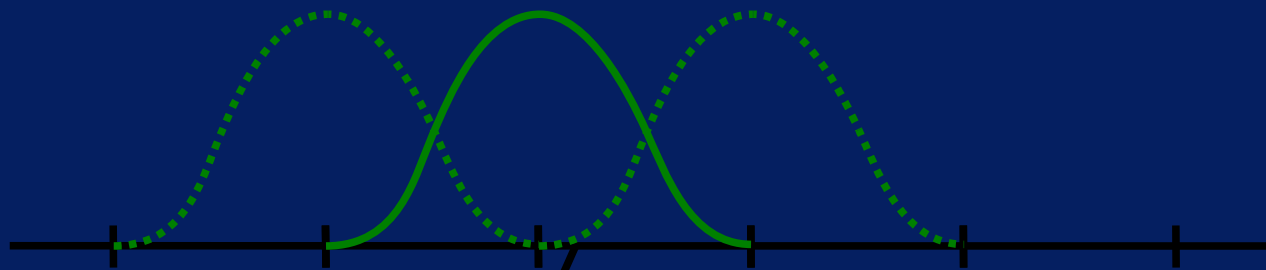
everything exactly once

reduces aliasing

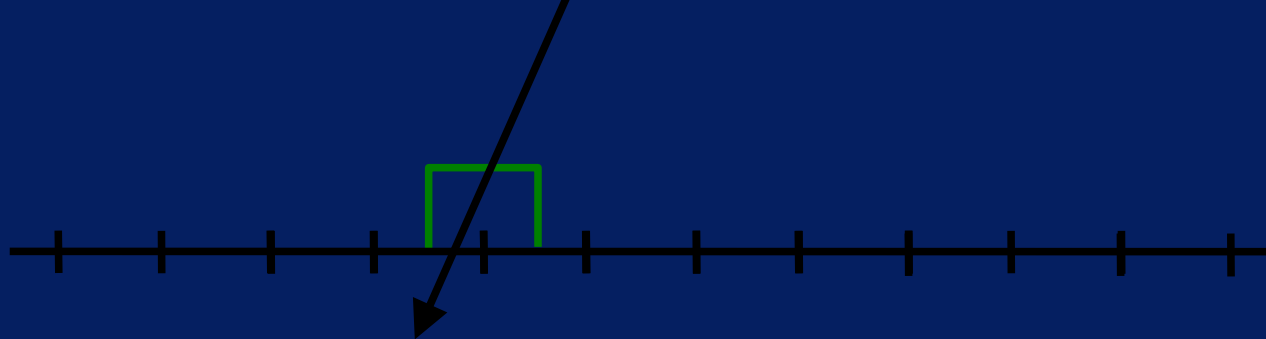
non-ideal basis (piecewise constant)



Projection Into a 4D Basis



Sampling
plane S



measurement
plane M

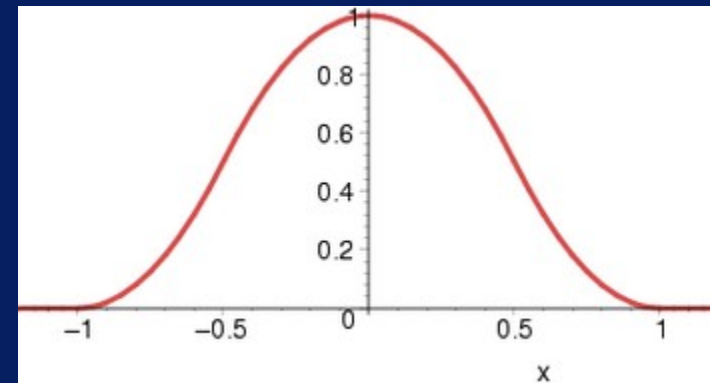
Projection into 4D Basis

sampling plane S

coarse sampling

e.g. piecewise quadratic basis functions

$$\Phi \phi_{i,bellq} = \begin{cases} 1 - 2x^2 & |x| \leq 1/2 \\ 2(|x| - 1)^2 & 1/2 < |x| \leq 1 \\ 0 & \text{else} \end{cases}$$



Projection into a 4D Basis

measurement plane M

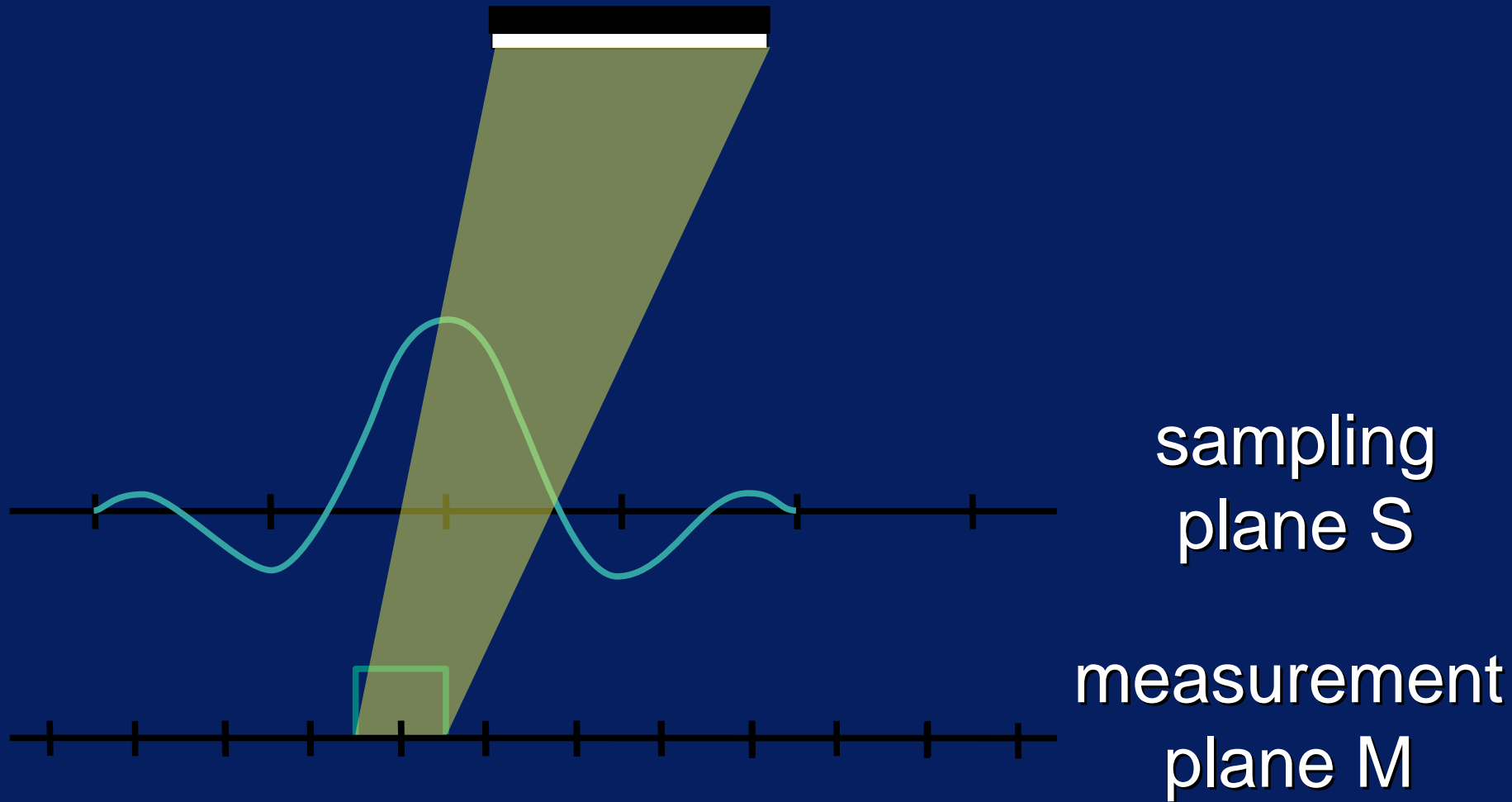
dense sampling

piecewise constant basis

tensor product construction of
4D basis

$$\begin{aligned}\Phi \Phi_{ijkl}(u, v, s, t) \\ &= \Phi \phi_{ij}(u, v) \cdot \phi_{kl}(s, t) \\ &= \Phi \phi_{i, bellq}(u) \cdot \phi_{j, bellq}(v) \cdot \phi_{k, const}(s) \cdot \phi_{l, const}(t)\end{aligned}$$

Dual Basis as a Filter



Sampling Issues

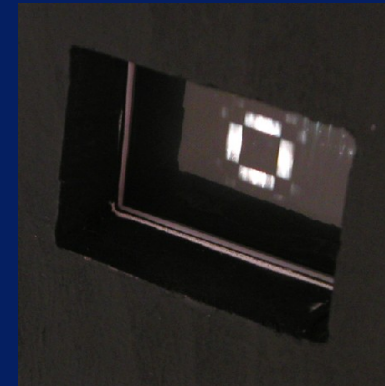
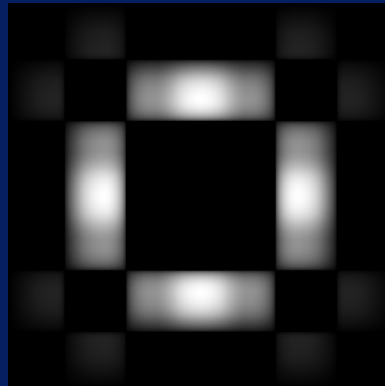
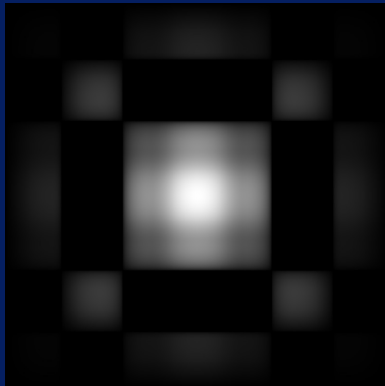
advanced filtering

use grayscale printed slide

arbitrary filter kernel

adapted to reconstruction algorithm

negative coefficients possible

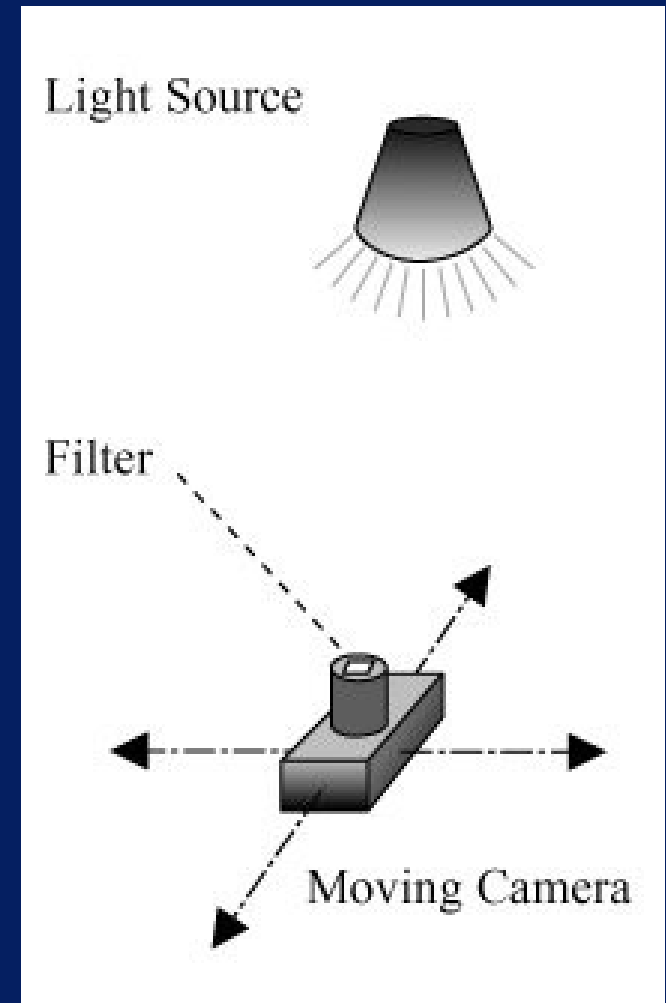


Measurement Setup A

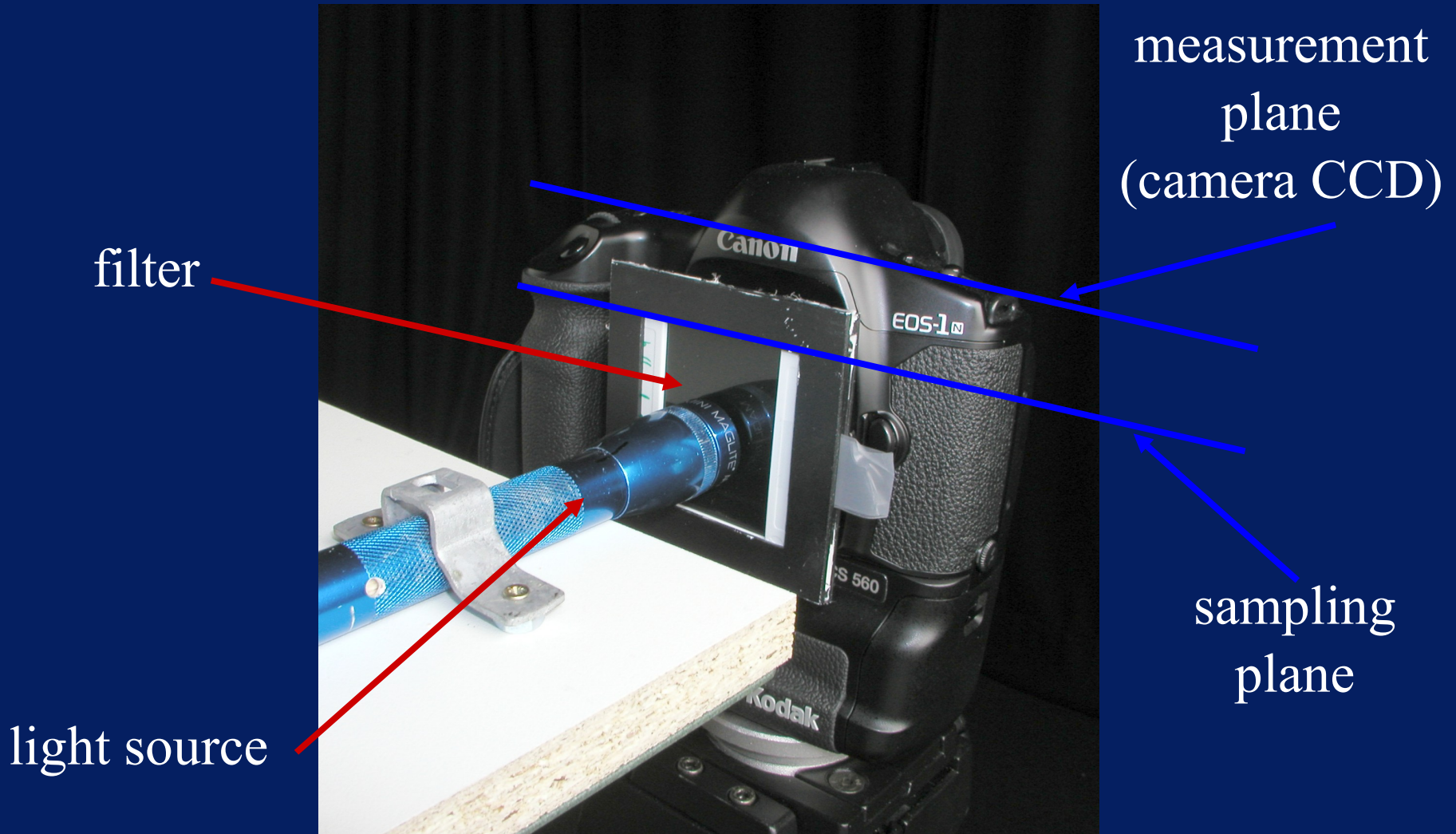
replace camera lens
system with filter
pinhole camera with
filter as “pinhole”

CCD chip as
measurement plane

move light source or
camera



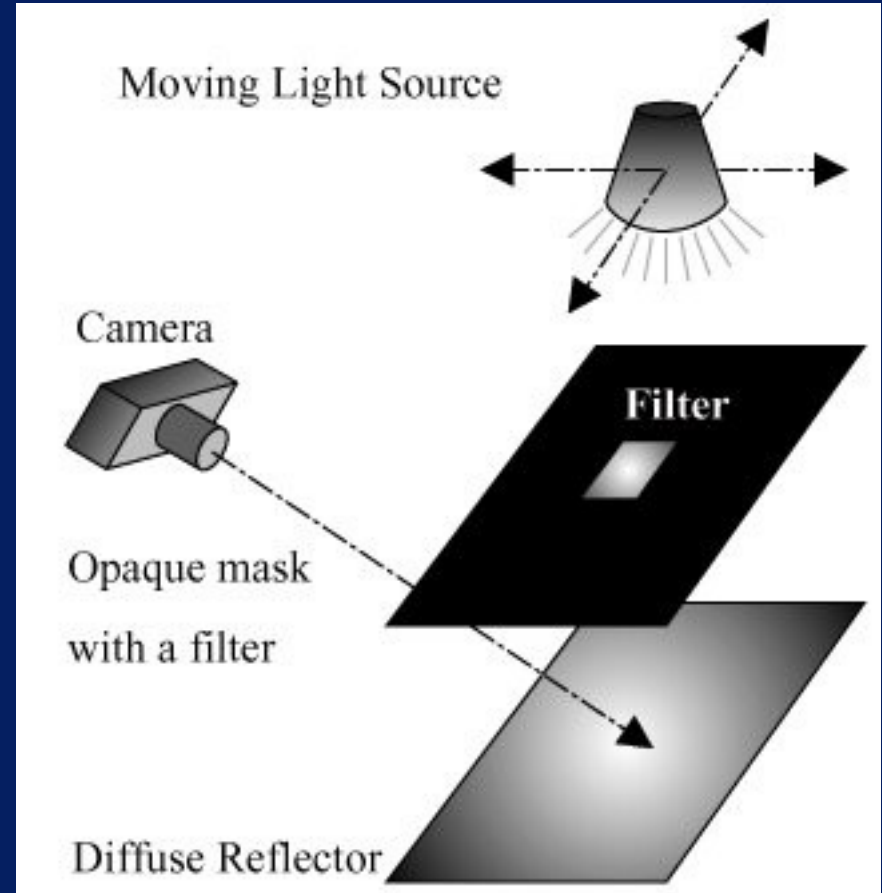
Measurement Setup A



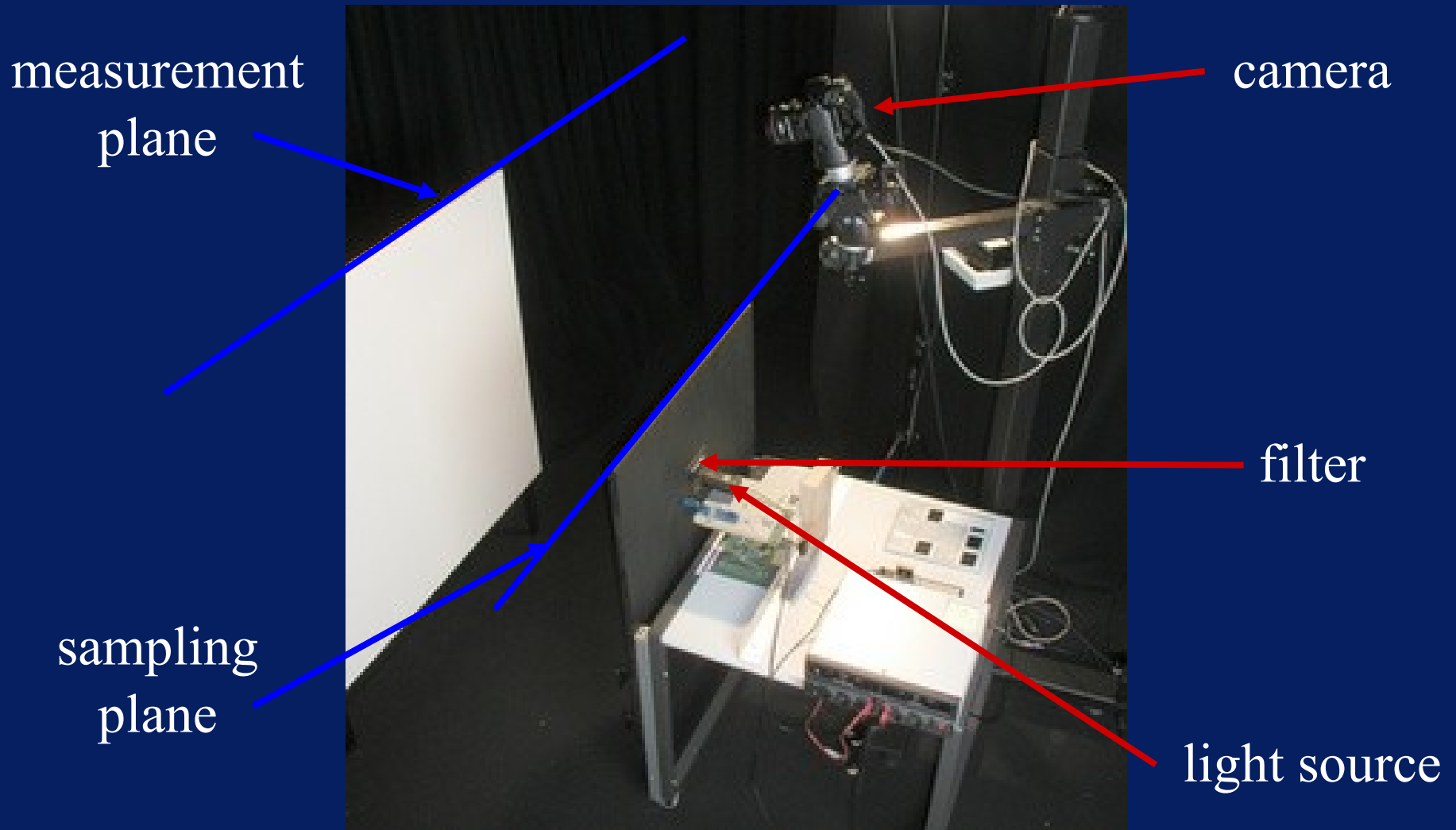
Measurement Setup B

filter projects light
source on projection
screen

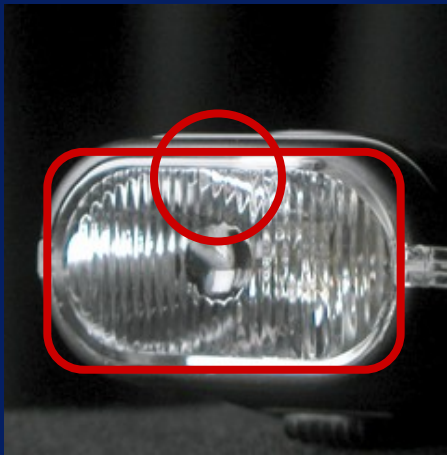
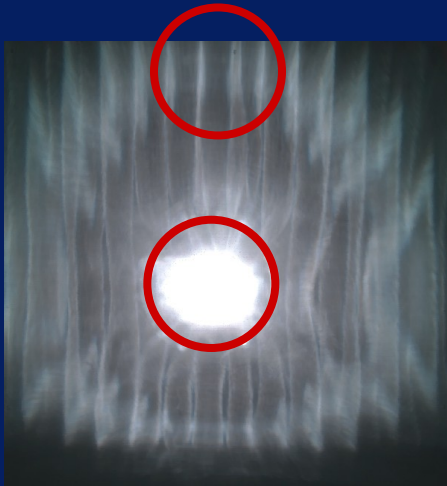
take pictures using
standard
photographic
techniques



Measurement Setup B



Acquired Data



“directional information sorted by spatial origin”

Rendering with Global Illumination

particle emission

(particle tracing, photon map)

treat intensities in light field as importance
function



Rendering with Global Illumination

constant-time particle emission

independent of data set size

inverted cumulative density function

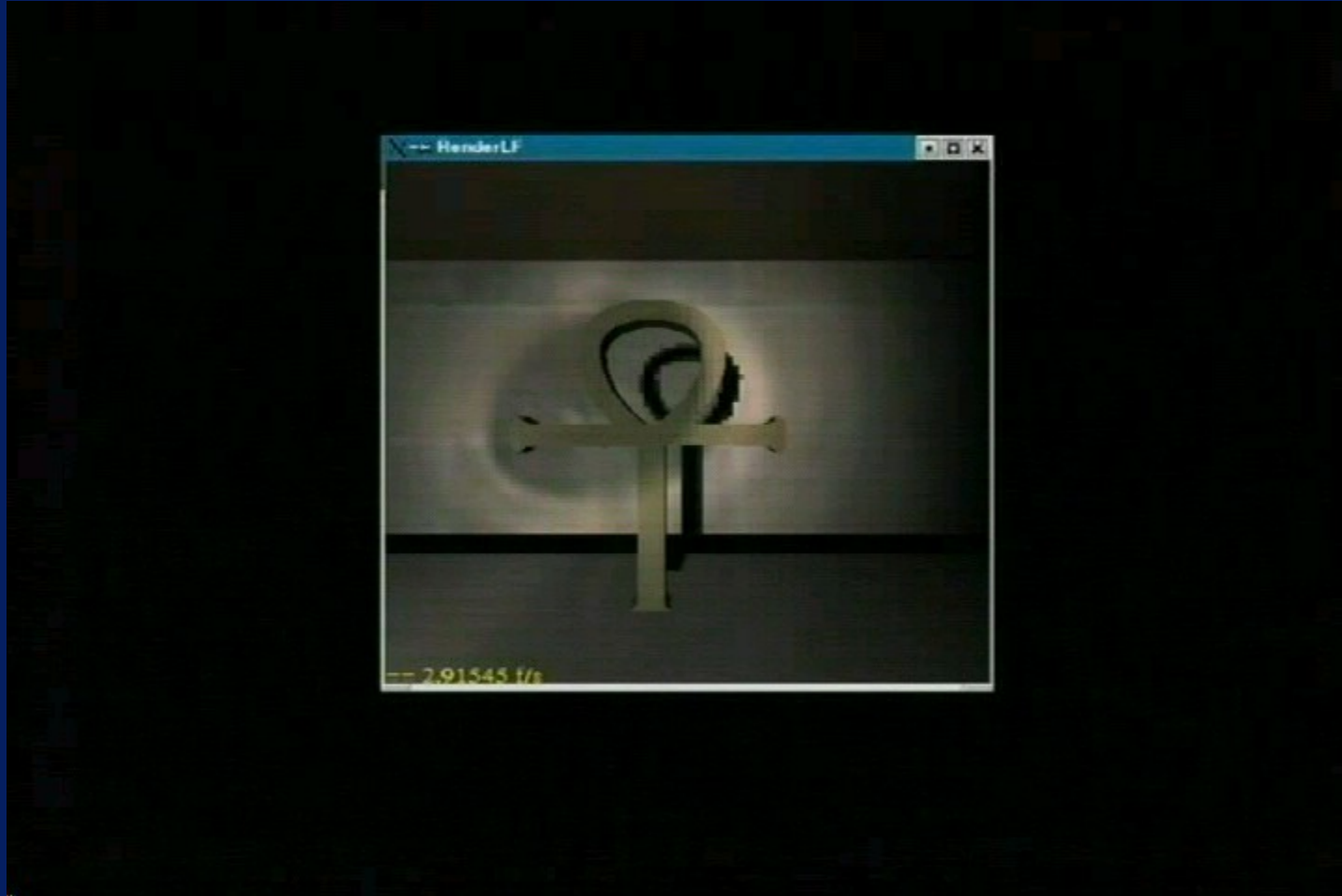
complex due to basis functions

2 step approach

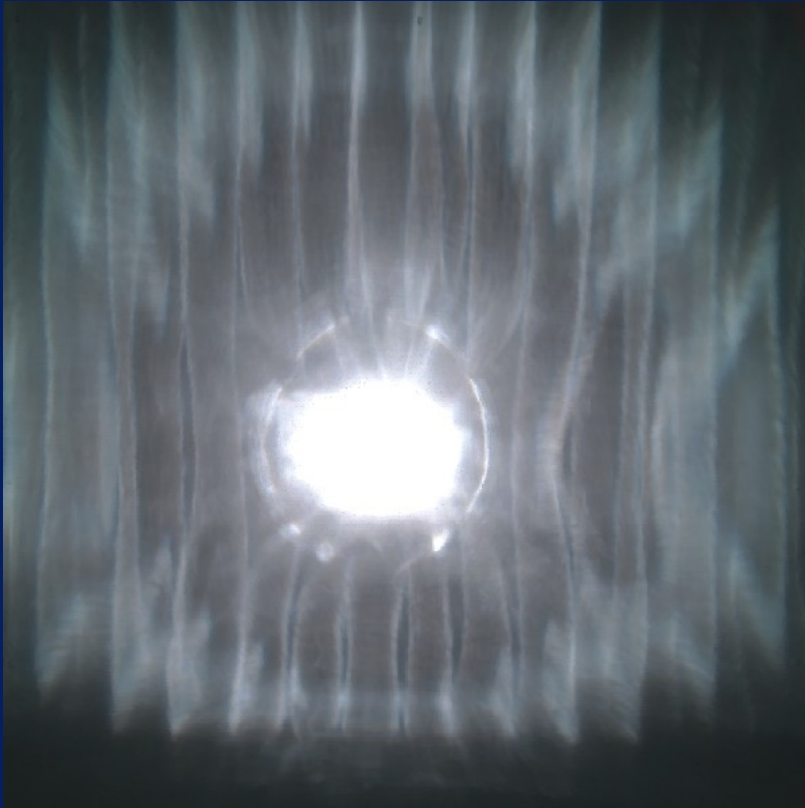
used for redistribution of random or quasi-random 4D samples

(e.g. Halton sequence, jittered sampling)

Hardware Accelerated Rendering



Results

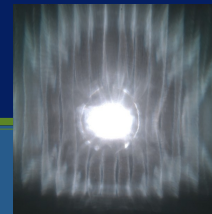


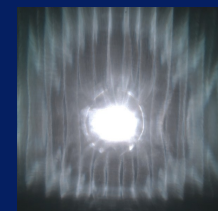
measured bike light data set
(9x7 images)



Measured bike light data set
(9x7 images)

PG 2003





measured bike light data set (9x7 images)

Conclusion

light source acquisition algorithm for near field
and far field

correct pre-filtering implemented

efficient rendering



Future Work

sampling issues

- choice of basis function

- sampling density

- replace printed filters with LCD panel

 - filters instantly exchangeable

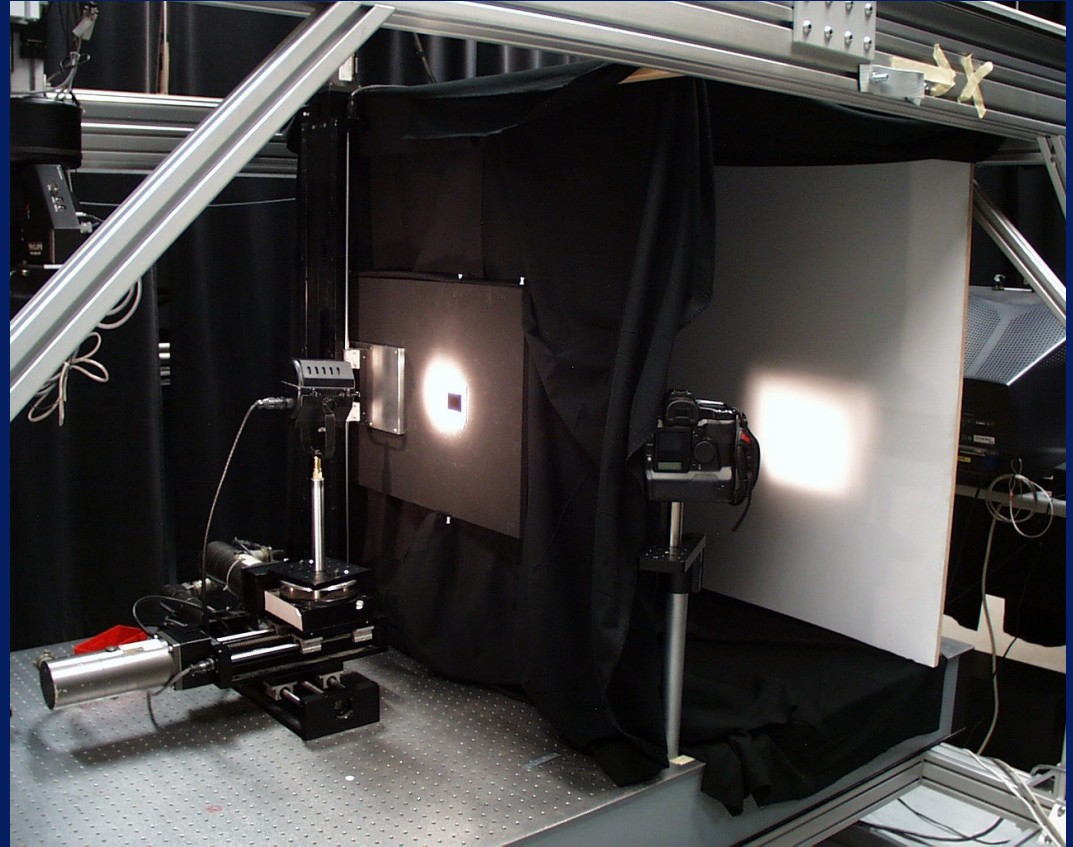
 - hierarchical acquisition (wavelets)

Future Work

different sampling surfaces

automated setup

UBC Active
Measurement
facility (ACME)
[Pai 2001]



Thanks to ...

Ian Ashdown

Oliver Ashoff

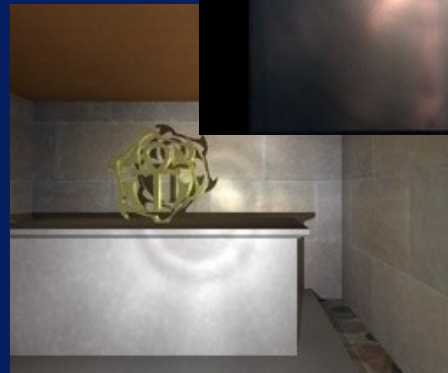
Gerhard Heisler

Michael Laise

John Lloyd

Axel Koepfel and

the anonymous SIGGRAPH
reviewers



Thanks to ...

REVES/INRIA Sophia-Antipolis
(GIS Global Illumination platform)

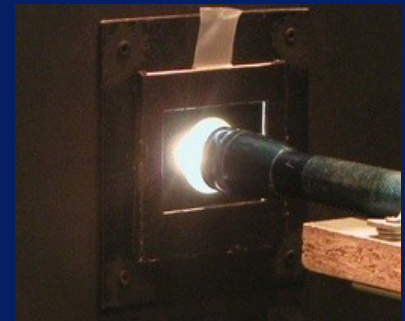
UBC's Institute of Applied Mathematics

This work was funded by

PIMS Post-doctoral Fellowship program

BC Advanced Systems Institute

DFG Schwerpunktprogramm V3D2



Questions?

