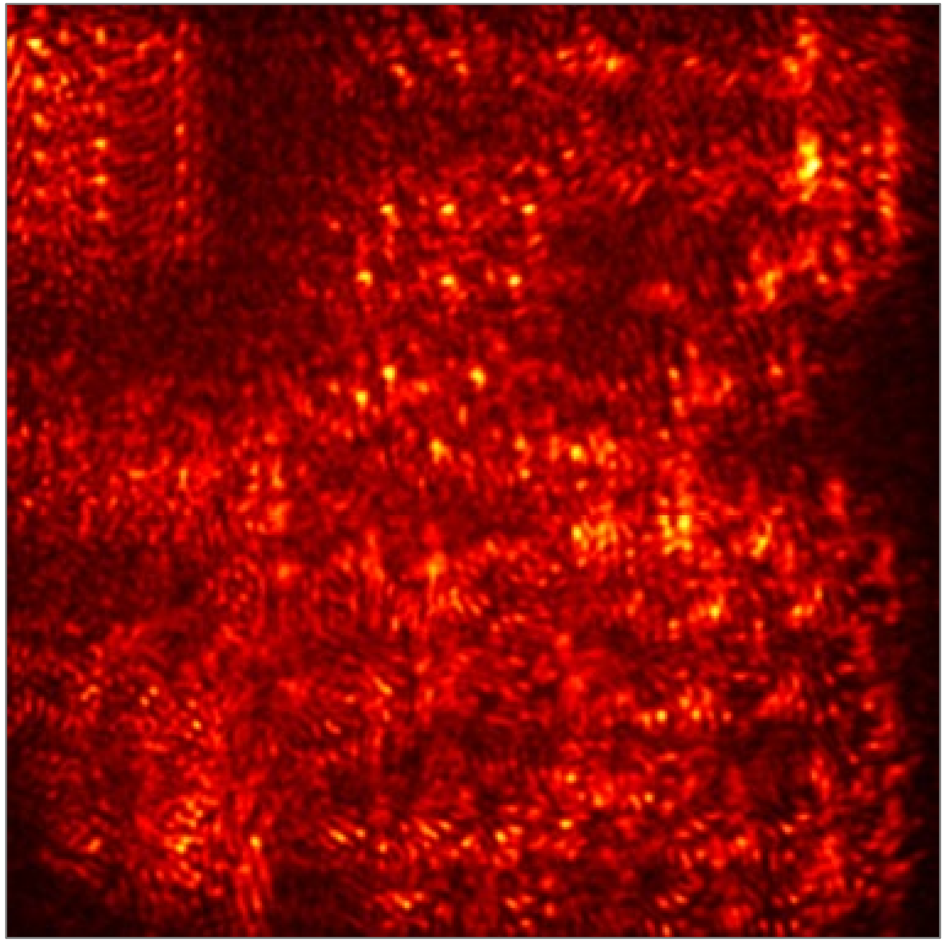
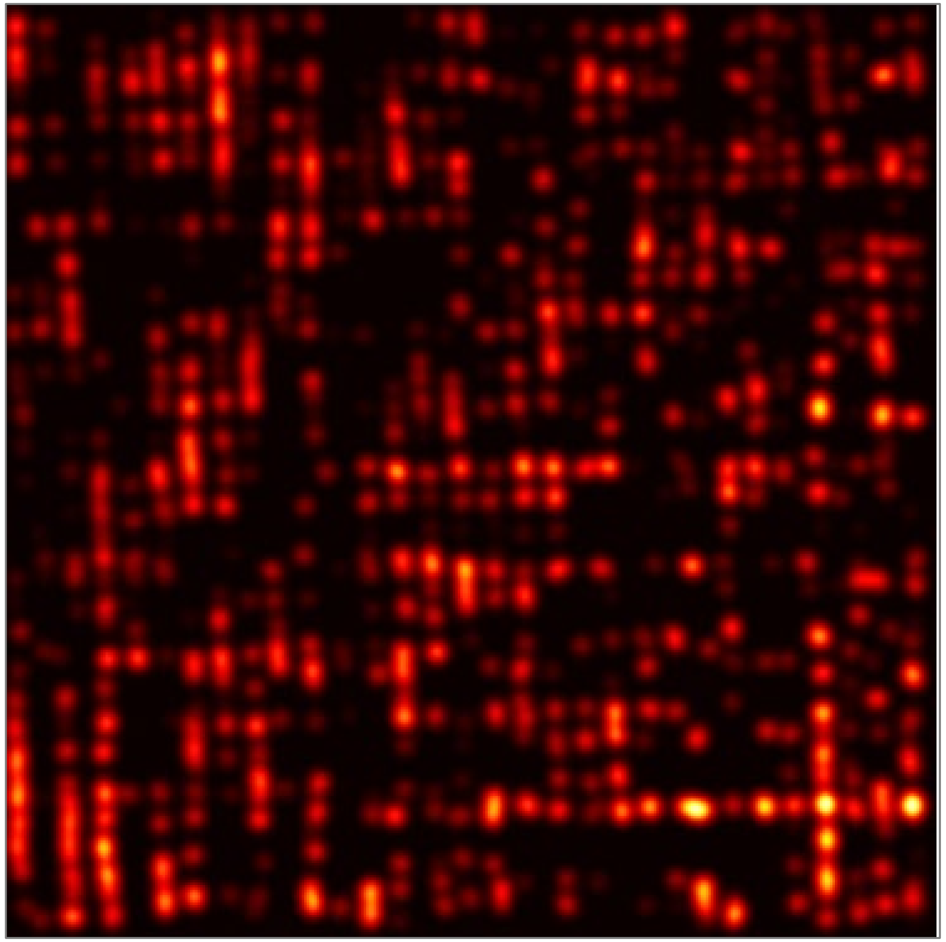
Rendering wave effects





15-468, 15-668, 15-868
Physics-based Rendering
Spring 2022, Lecture 16

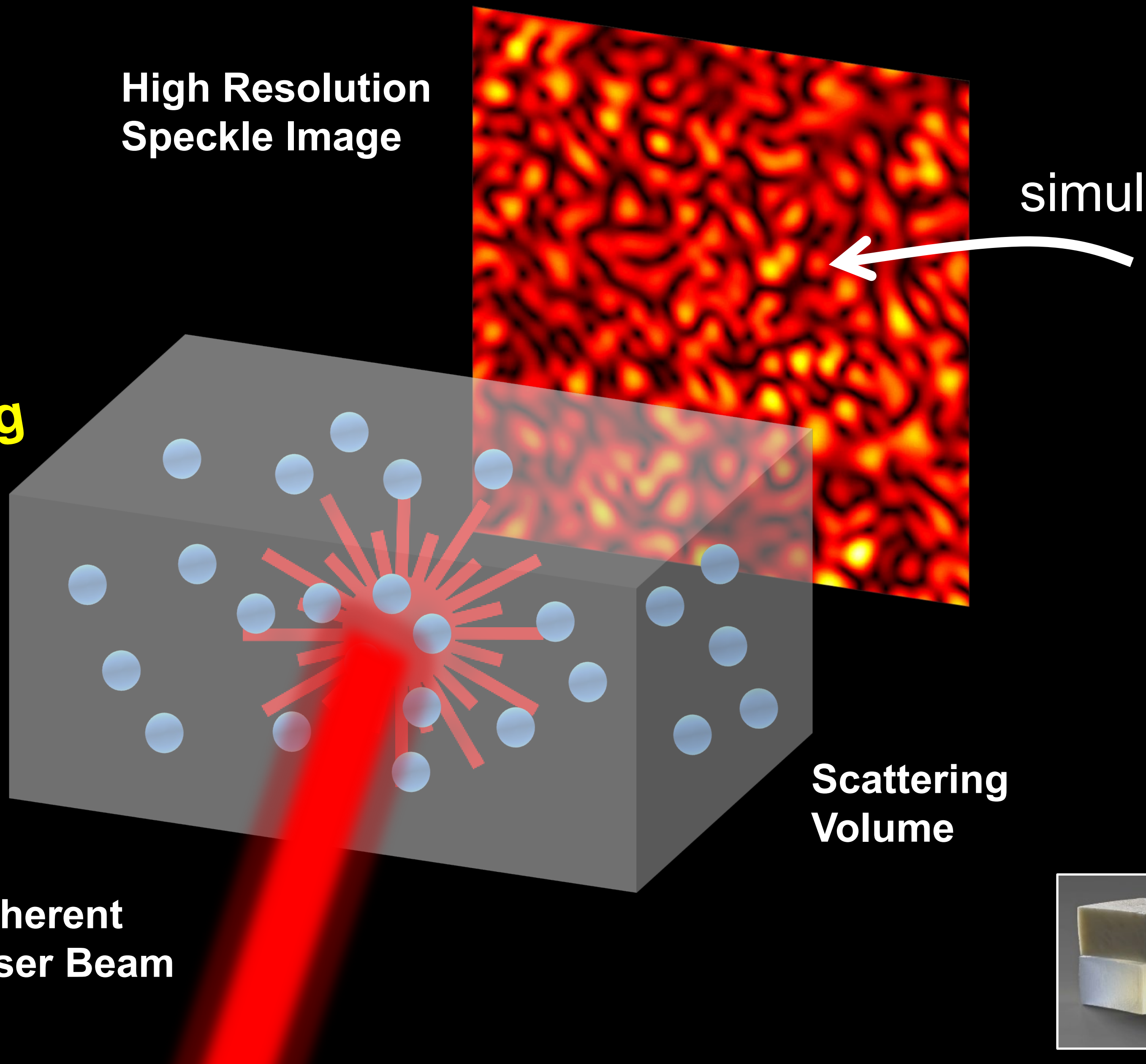
http://graphics.cs.cmu.edu/courses/15-468

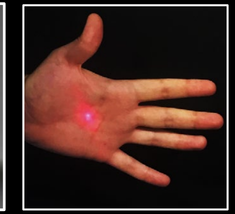
Course announcements

- How many of you attended Thomas Mueller’s talk?
- Take-home quiz 9-10 posted, due 4/26.
- Nobody was around for yesterday’s recitation :-(.
- Will try to go over final project proposals tonight.

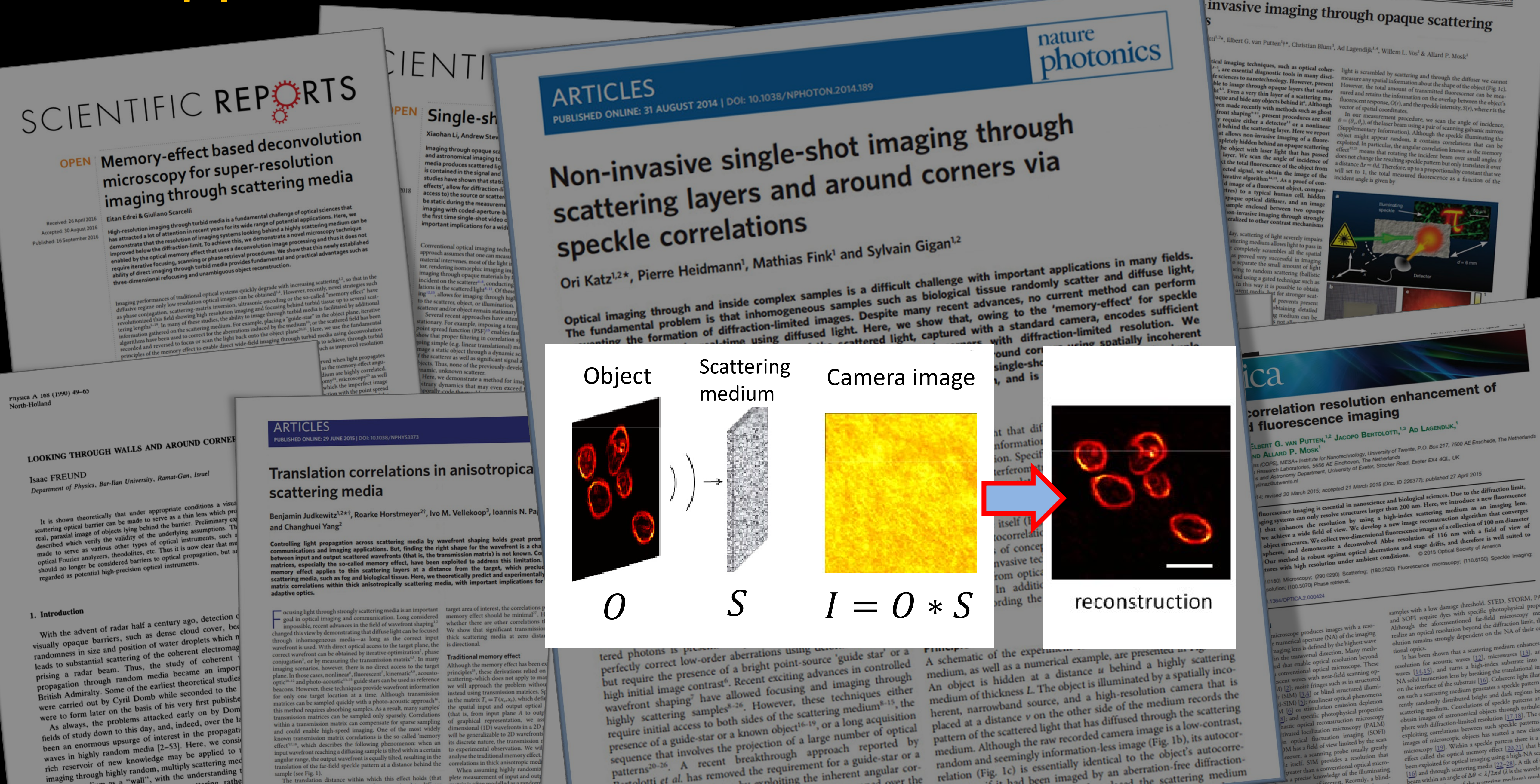
Coherent Scattering and Memory Effect

The Memory Effect is very useful in many computational imaging applications





Applications and Related Work

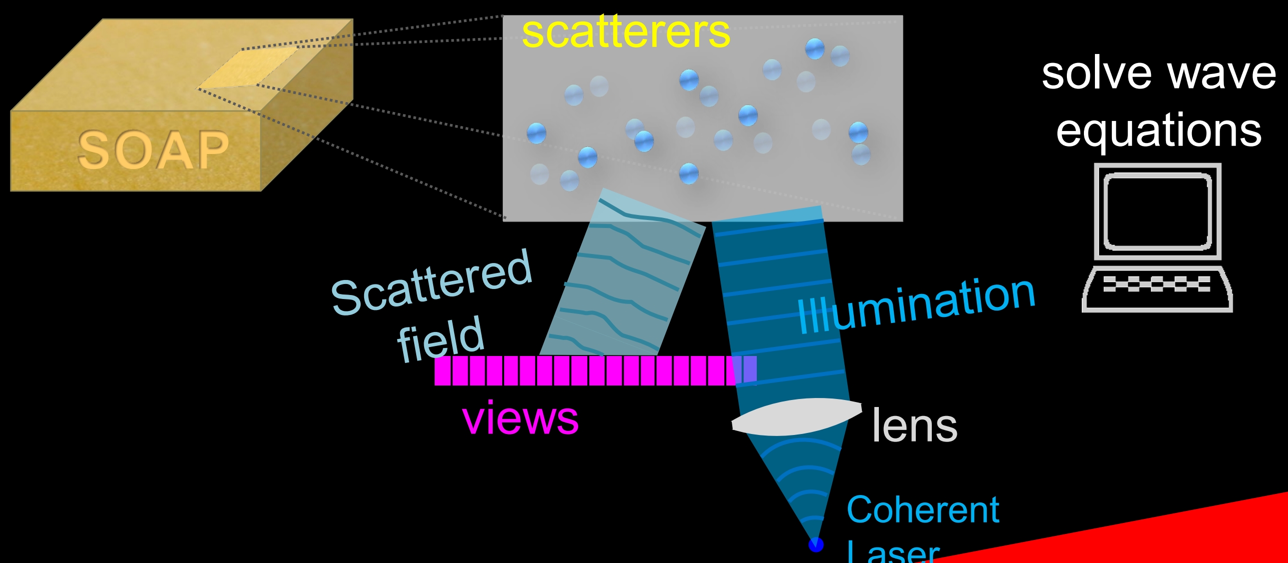


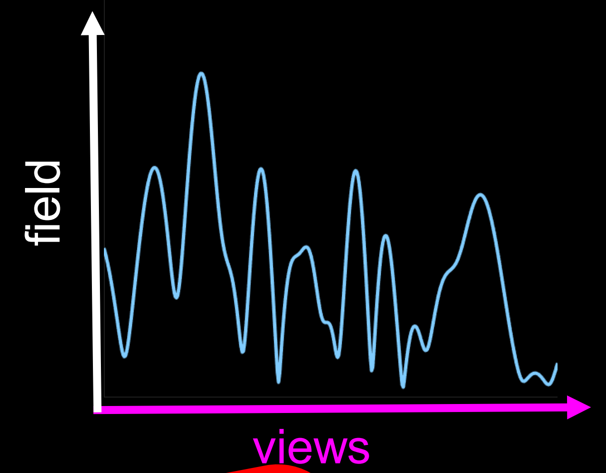
Simulating speckles

inefficient

Specify exact (sub-wavelength) position of scatterers

In graphics we describe materials by ***statistical*** bulk parameters, as the ***density*** of scatterers





Wave equation solvers

- Differential equation F
- Integral equation (e.g.,

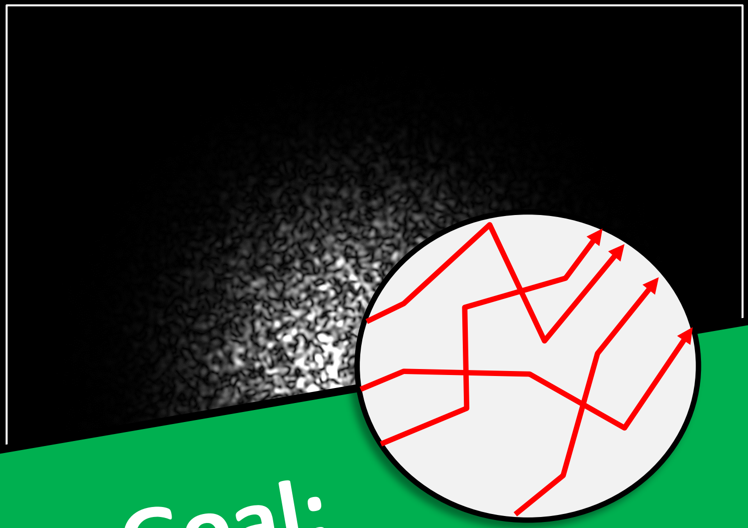
Slow

Practical only for tiny or optically thin media

For simplicity: **Flatland**
Scattering medium is **2D**
Sensor is **1D**
Speckle pattern is **1D**

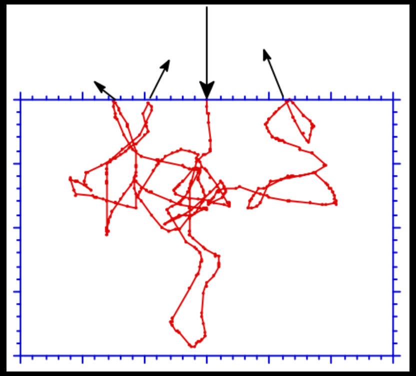
Monte Carlo (MC) Simulation of Speckles

MC Advantage:
1. **Fast**
2. input is scatterer **density** rather than exact scatterer locations



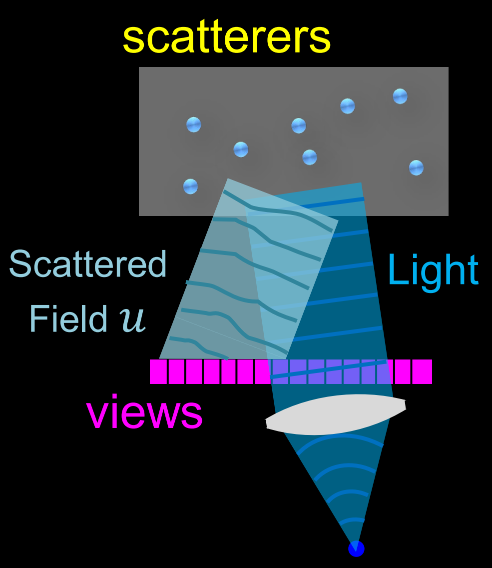
Our Goal:
Extend efficient MC tools to evaluate speckles and their coherent statistics

Standard intensity MC

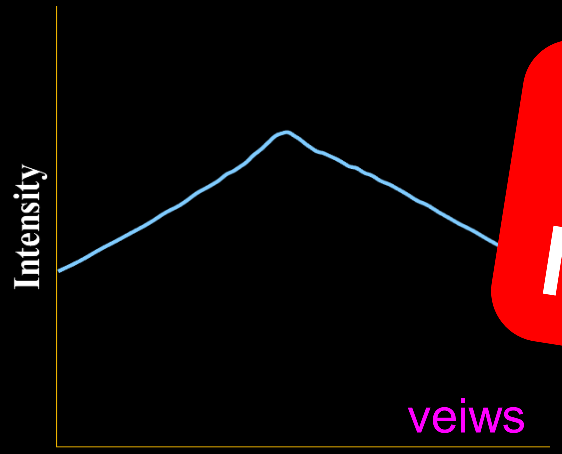


Monte Carlo Modeling of Light Transport in Multi-layered Tissues, **Wang & Jacques, 1992**

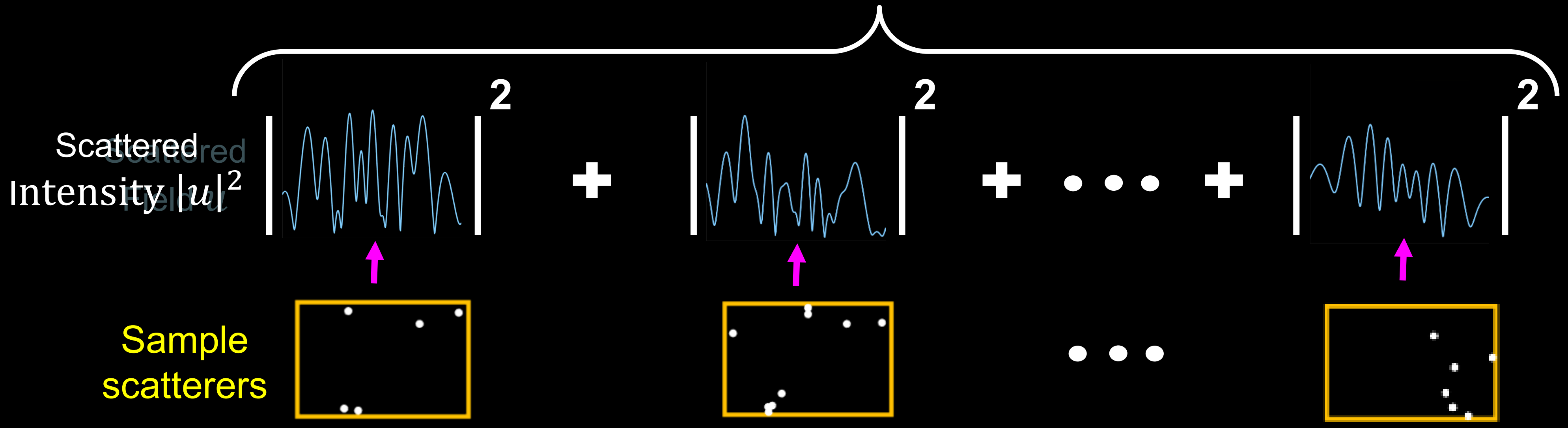
Wave Solution v.s. Monte Carlo



Intensity $= E_{\text{scatterers}}\left[\left|u^{\text{scatterers}}\right|^2\right]$



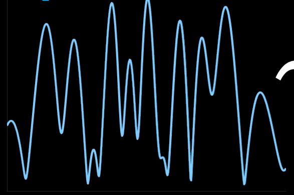
No Speckles
No Memory Effect



MC requires the scatterers density – no need for exact positions

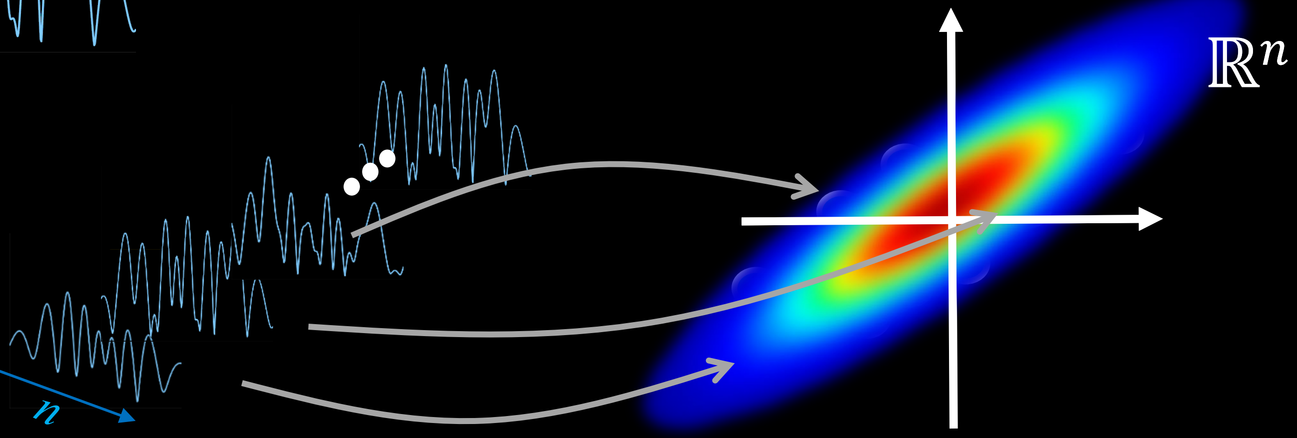
Speckle Statistics

Speckles



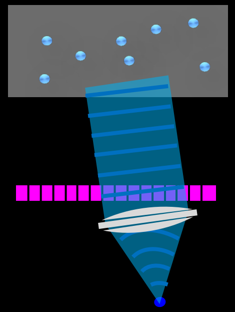
$\sim \mathcal{N}\,(\text{Mean}, \text{Covariance})$

Sufficient Statistics



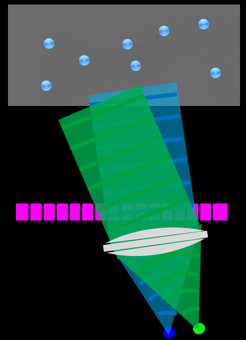
1st moment

$\cancel{\text{Intensity Mean} = \mathrm{E}_{\text{scatterers}}\left[\left|u^{\text{scatterers}}\right|^2\right]}$ $\Rightarrow$ Incoherent Summation

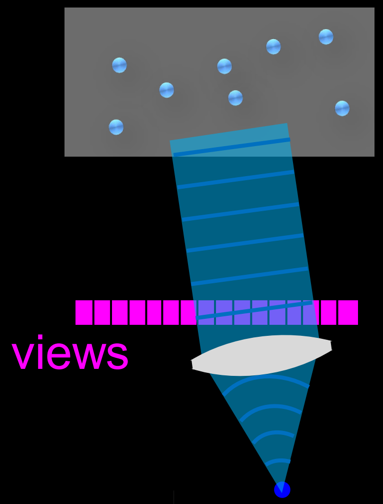


Cross-Illumination

$$\text{Field Covariance} = \mathrm{E}_{\text{scatterers}}\left[u_{\text{view}_i}^{\text{light}_1,\text{scatterers}} \cdot u_{\text{view}_j}^{\text{light}_2,\text{scatterers}\,*}\right]$$

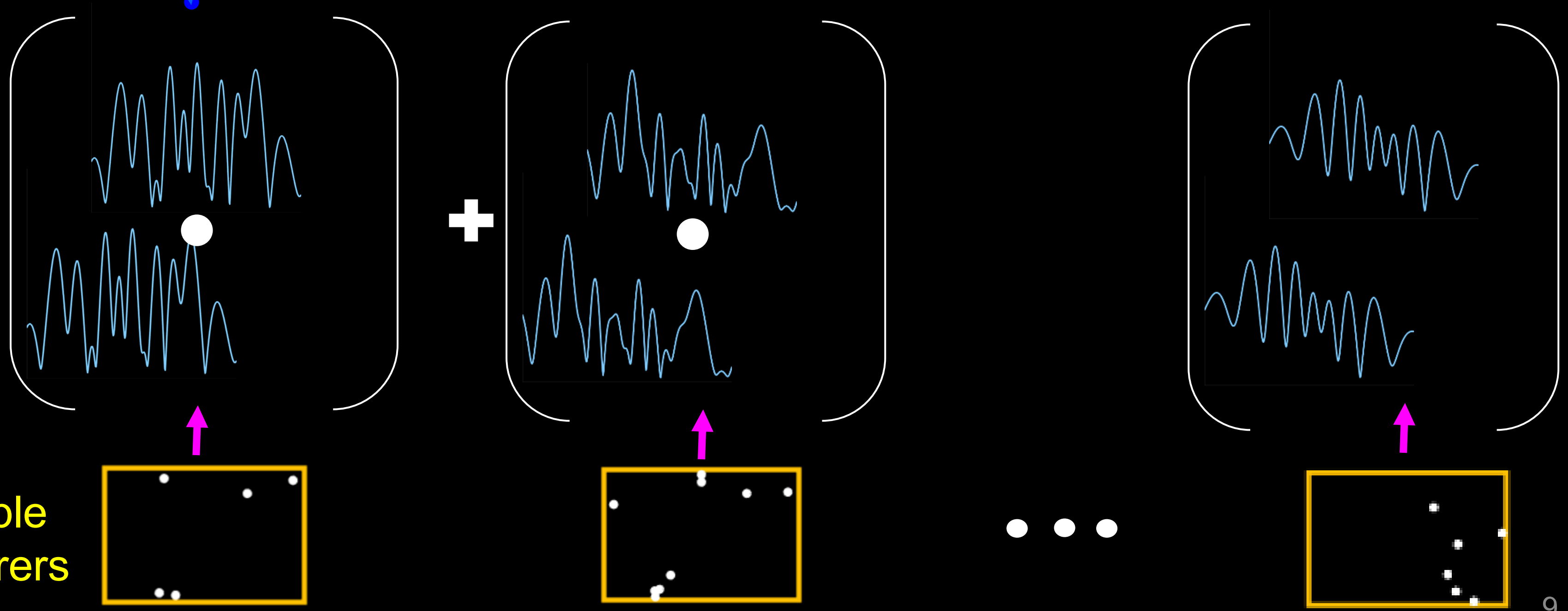


2^{nd} Moment - Covariance

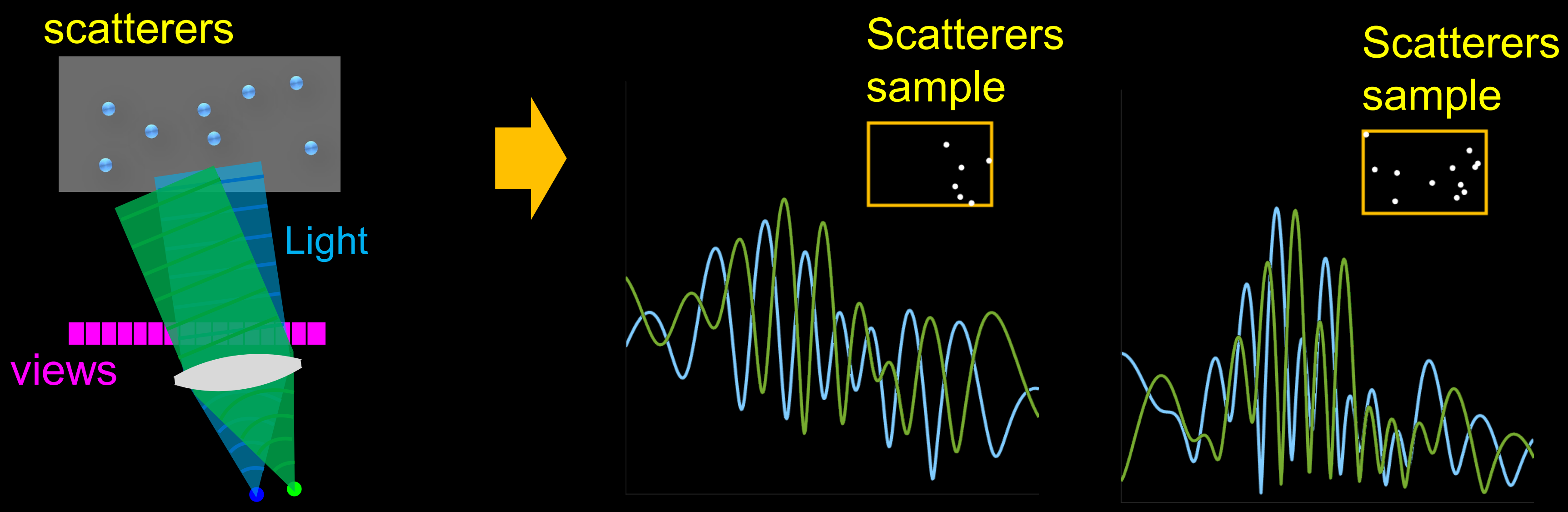


Covariance $= \mathrm{E}_{\text{scatterers}}\left[u_{\text{view}_i}^{\text{scatterers}} \cdot u_{\text{view}_j}^{\text{scatterers}^*}\right]$

There is correlation between pixels across illuminations → Memory Effect



Cross –illumination statistics



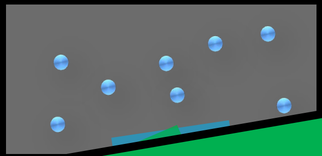
Memory Effect:

tilting illumination results in highly correlated shifted speckles

Next: Cross Illumination Covariance

Cross –illumination statistics

scatterers



$$\text{Covariance} = \mathrm{E}_{\text{scatterers}}\left[u_{\text{view}_i}^{\text{light}_1,\text{scatterers}} \cdot u_{\text{view}_j}^{\text{light}_2,\text{scatterers}\,*}\right]$$

nal shift =

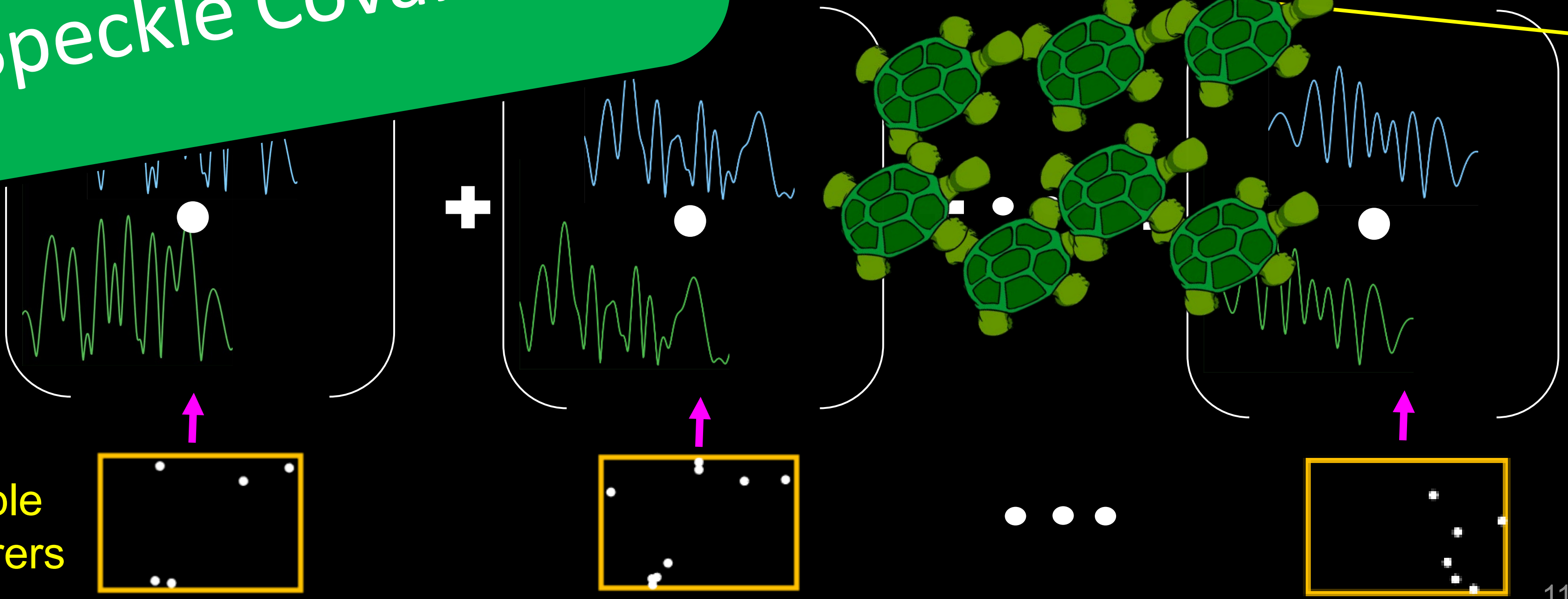
ift

We developed:
Efficient MC
for Speckle Covariance

Speckles are Gaussian:
Mean + Covariance
are sufficient statistics

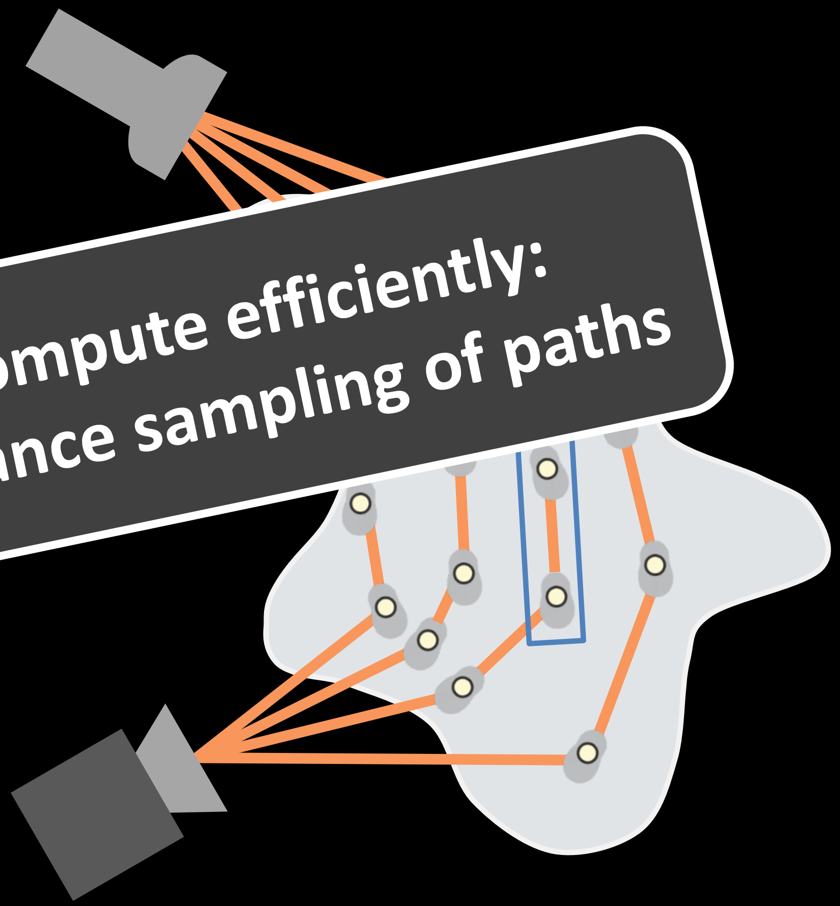
This is all we need to describe speckles

Scattered Field u^{light_2}



Sample scatterers

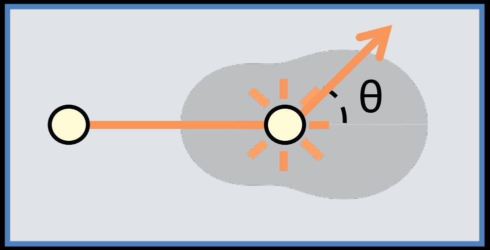
Monte Carlo Rendering 101



To compute efficiently: importance sampling of paths

$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

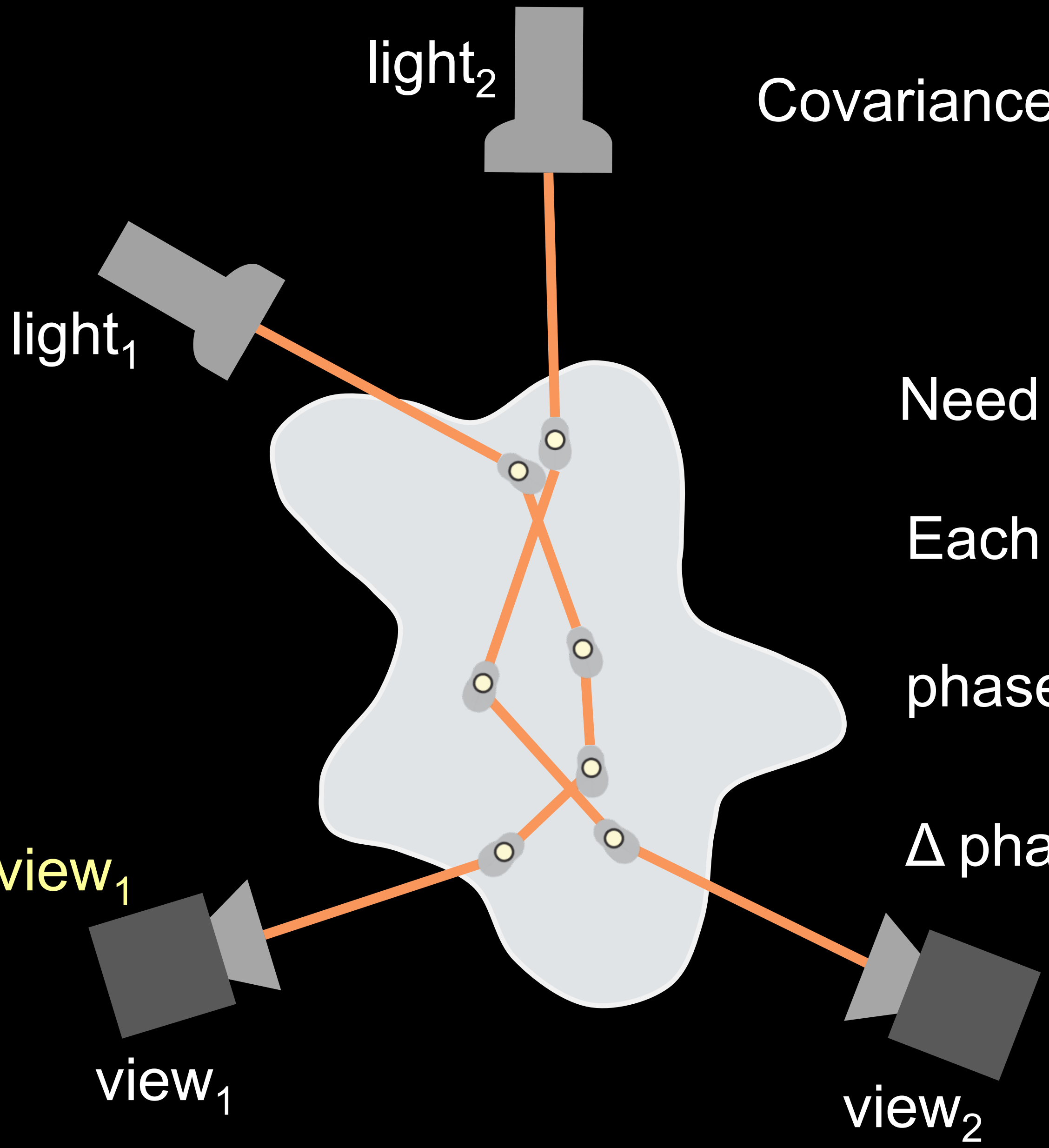
Throughput that acts on each path, depends on the scattering material



material =

$$\begin{array}{r}\text{volumetric density} \\ \text{(extinction coefficient)} \\ \text{scattering albedo} \\ \text{phase function}\end{array} \begin{bmatrix} \sigma \\ a \\ p_\theta \end{bmatrix}$$

Covariance Rendering



path$_1$: light$_1$ → view$_1$

path$_2$: light$_2$ → view$_2$

$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1)\cdot u^*(\text{path}_2)$$

$u = |u|\, e^{i\cdot \text{phase}}$

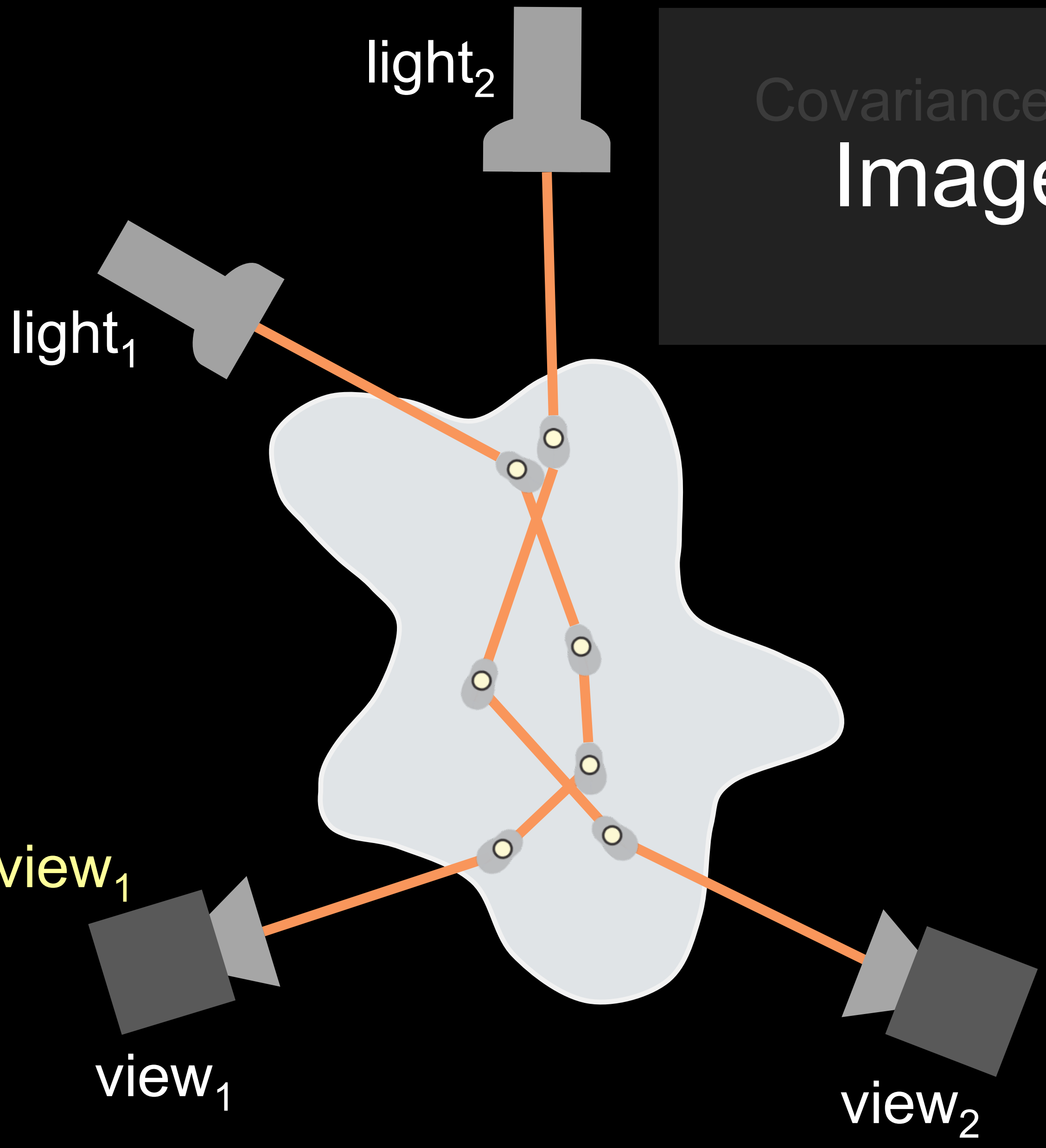
Need to consider products of ***pairs*** of paths

Each path contributes a complex number u

phase ∝ Length (path)

Δ phase ∝ Length (path$_1$) - Length (path$_2$)

Covariance Rendering



$$\text{Image} = \int_{\text{paths}} f(\text{path})$$

path$_1$:
light$_1$ → view$_1$

$$\text{path}_1 = \text{path}_2$$

⇩

Same complex contribution

$$u(\text{path}_1) = u(\text{path}_2)$$

⇩

$$\Delta\ \text{phase} = 0$$

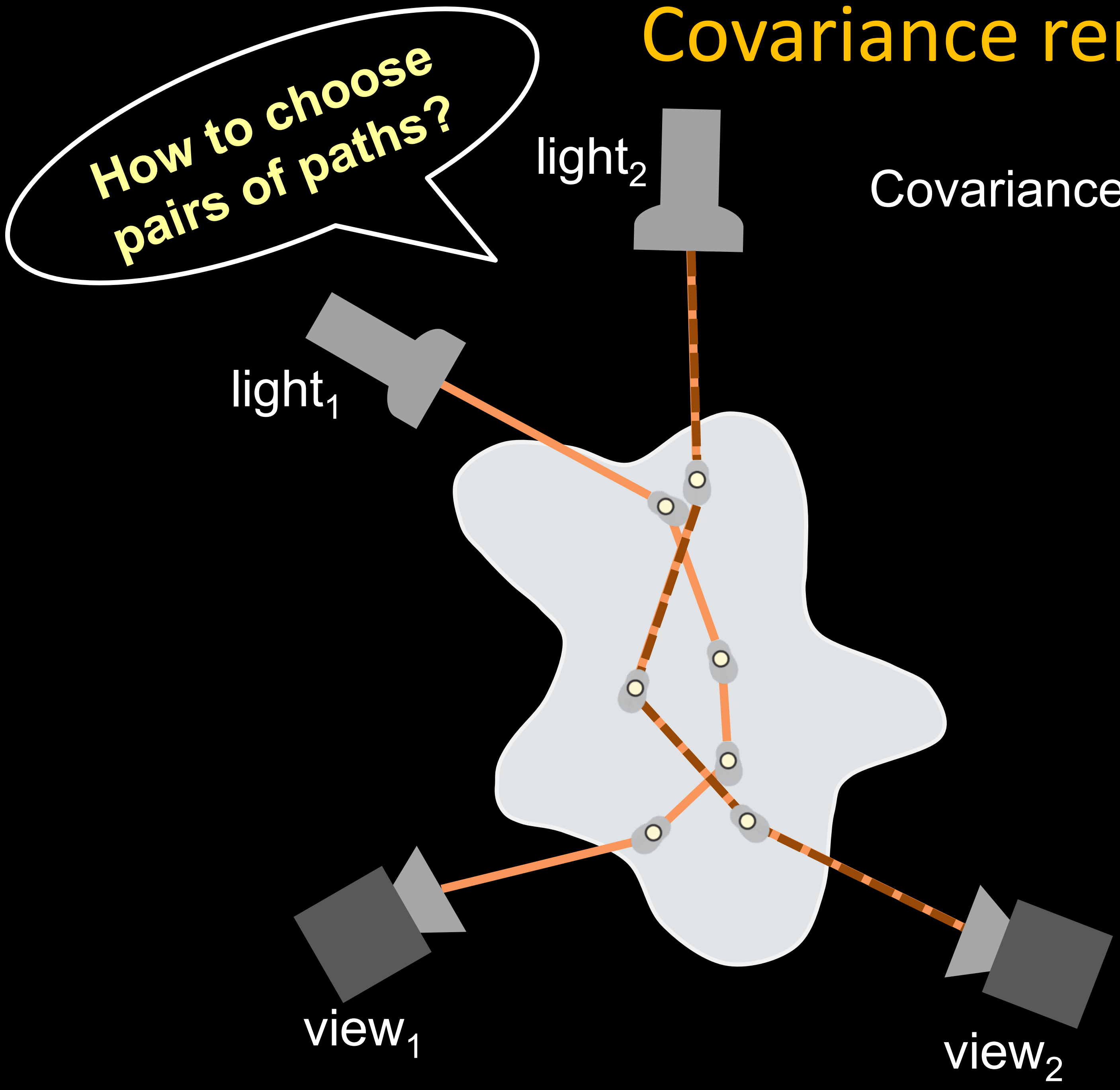
⇩

Real

$$u(\text{path}) \cdot u^*(\text{path}) = f\,(\text{path})$$

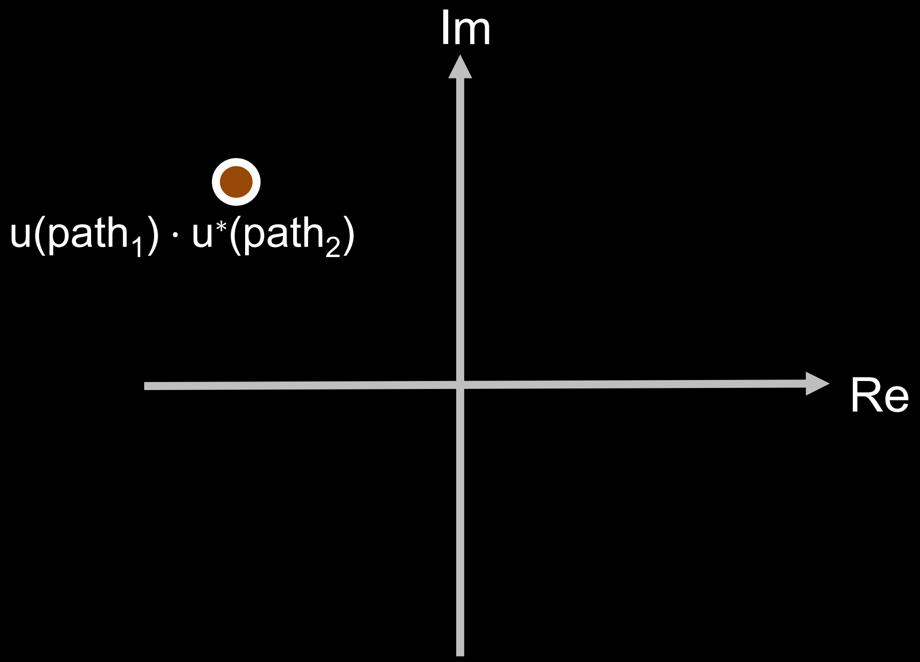
path$_2$:
light$_2$ → view$_2$

Covariance rendering



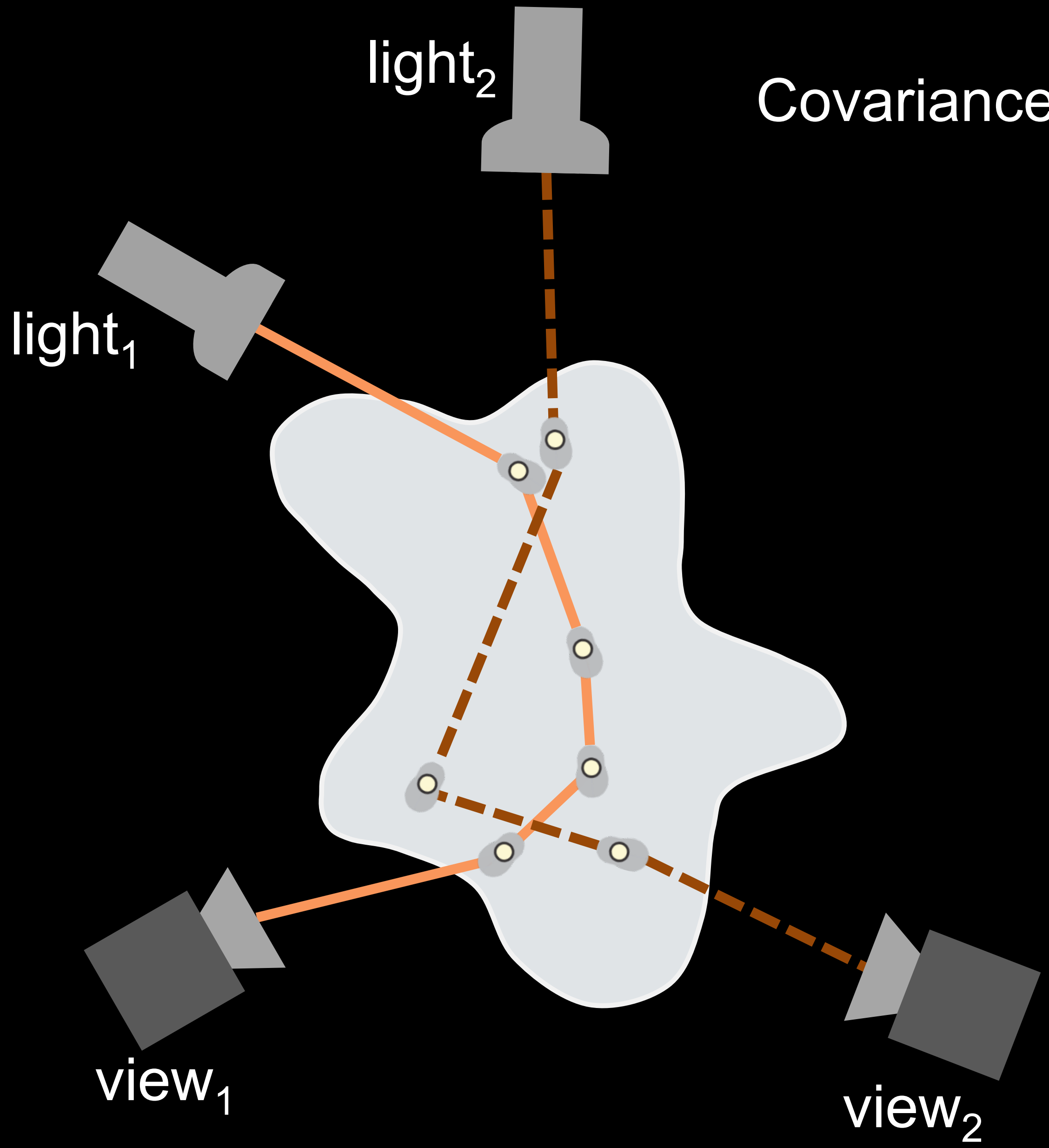
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

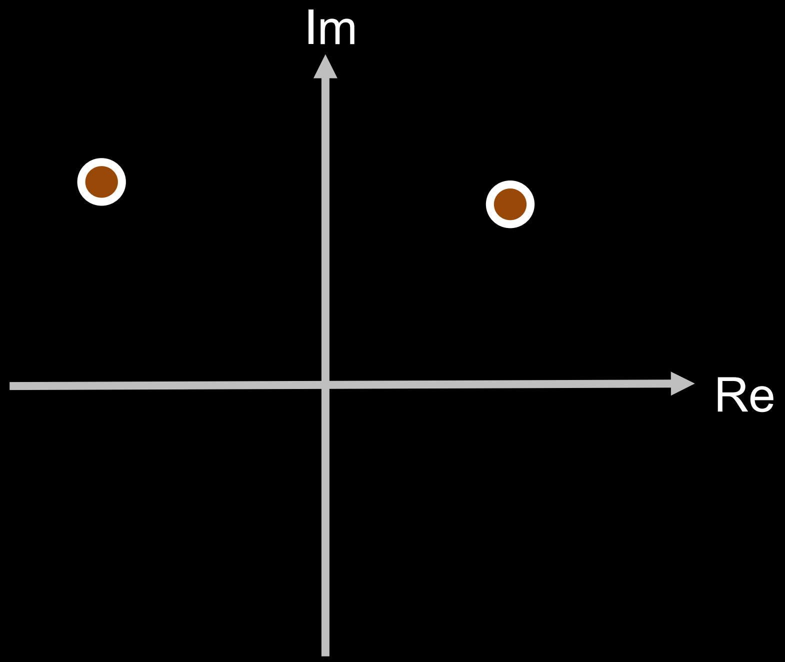
$$u = |u|\, e^{i \cdot \text{phase}}$$



Covariance rendering

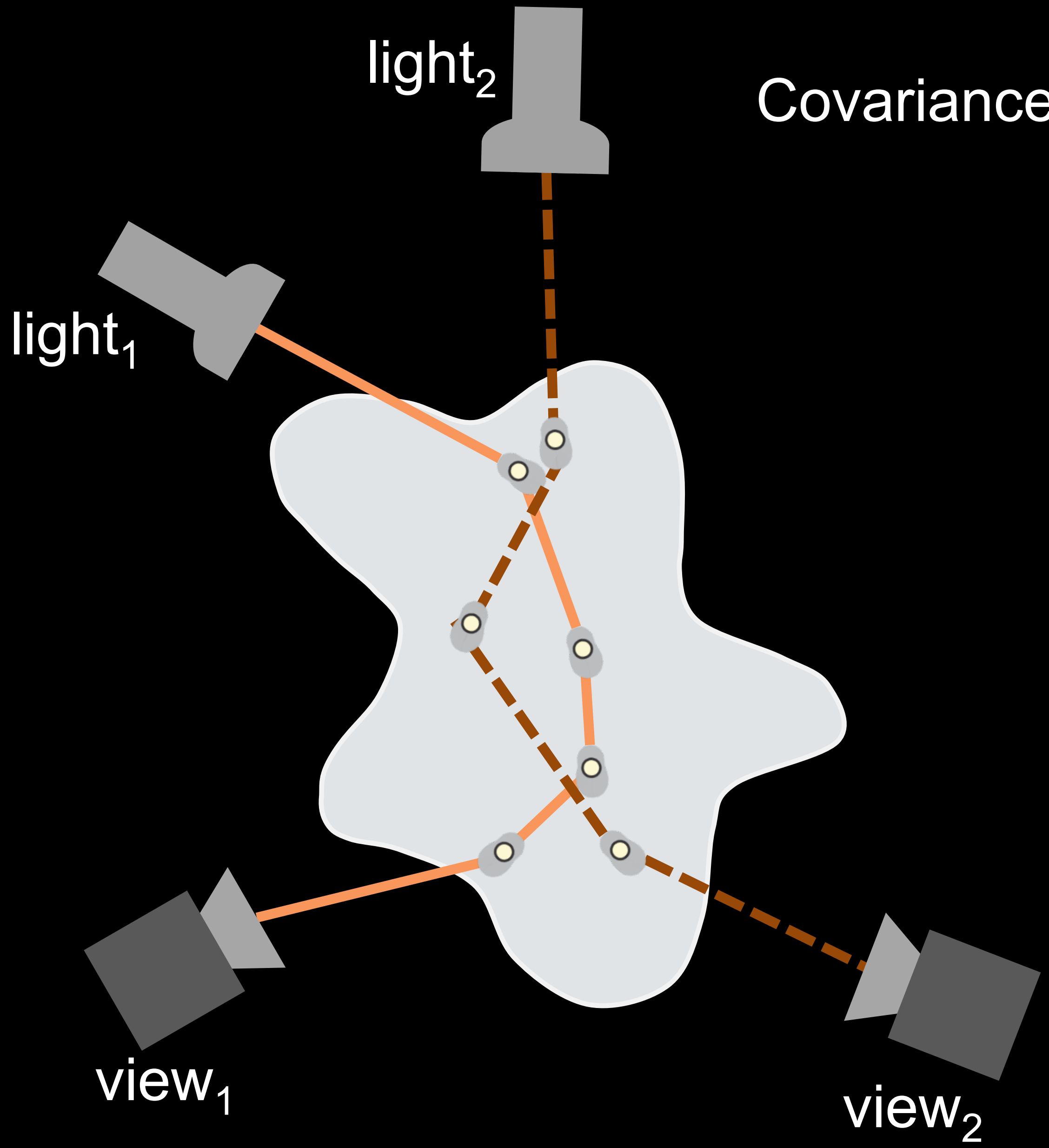
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

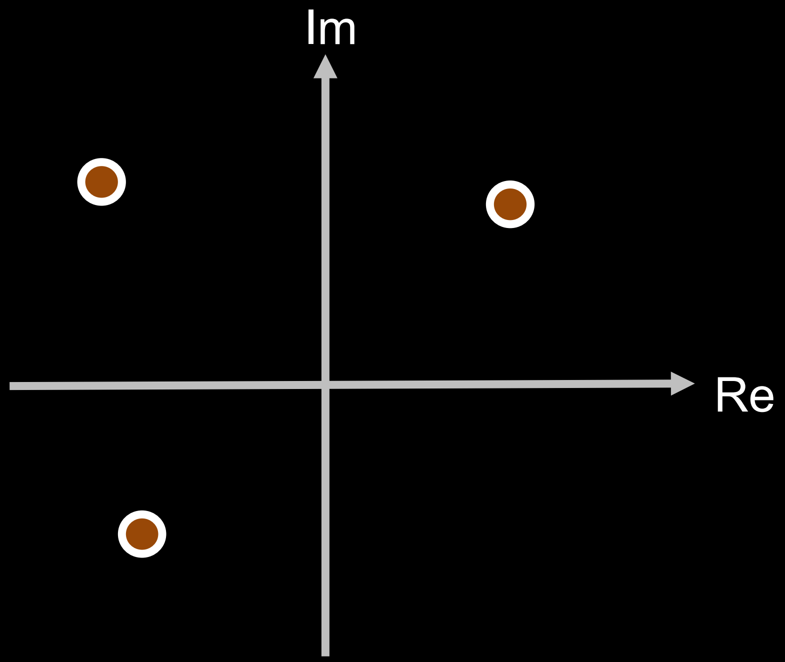




Covariance rendering

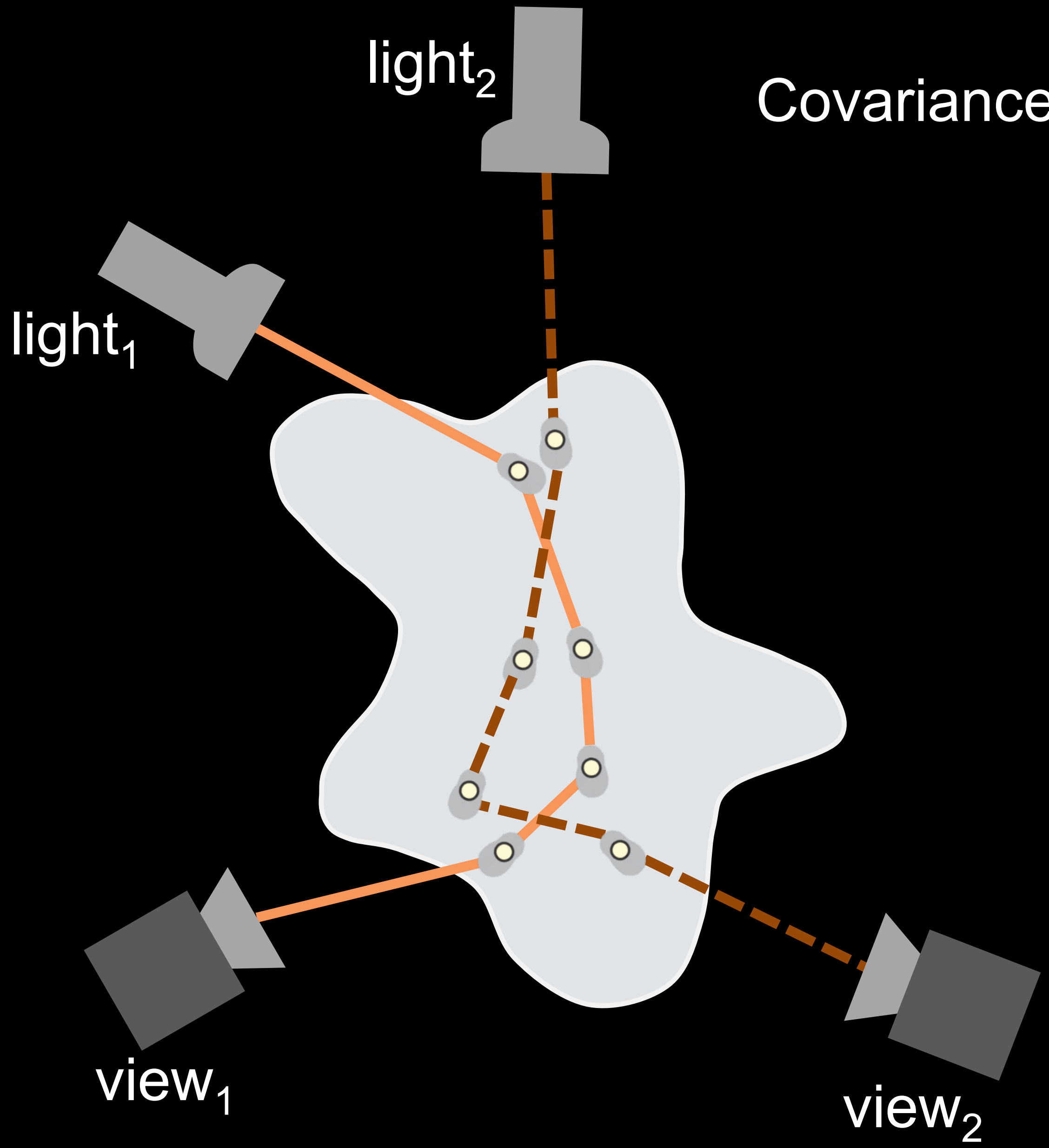
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

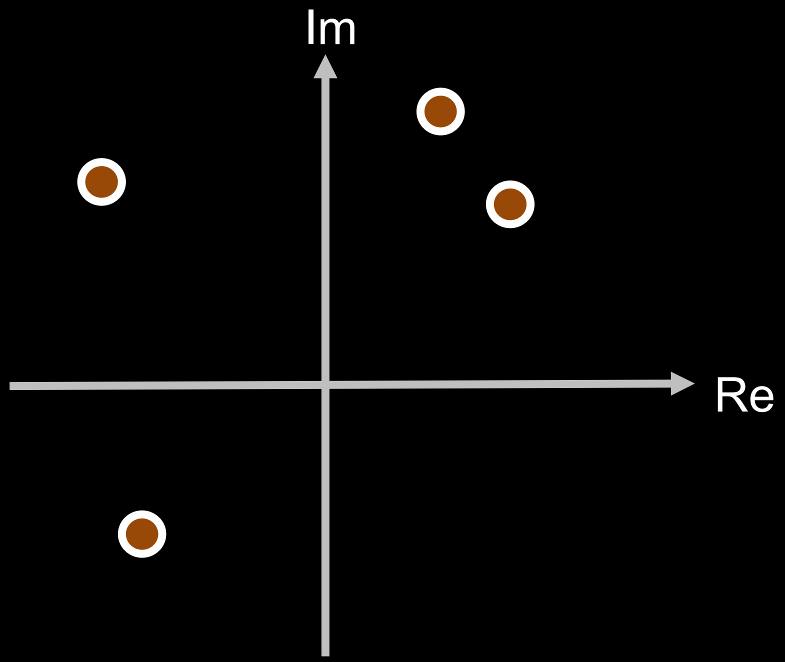




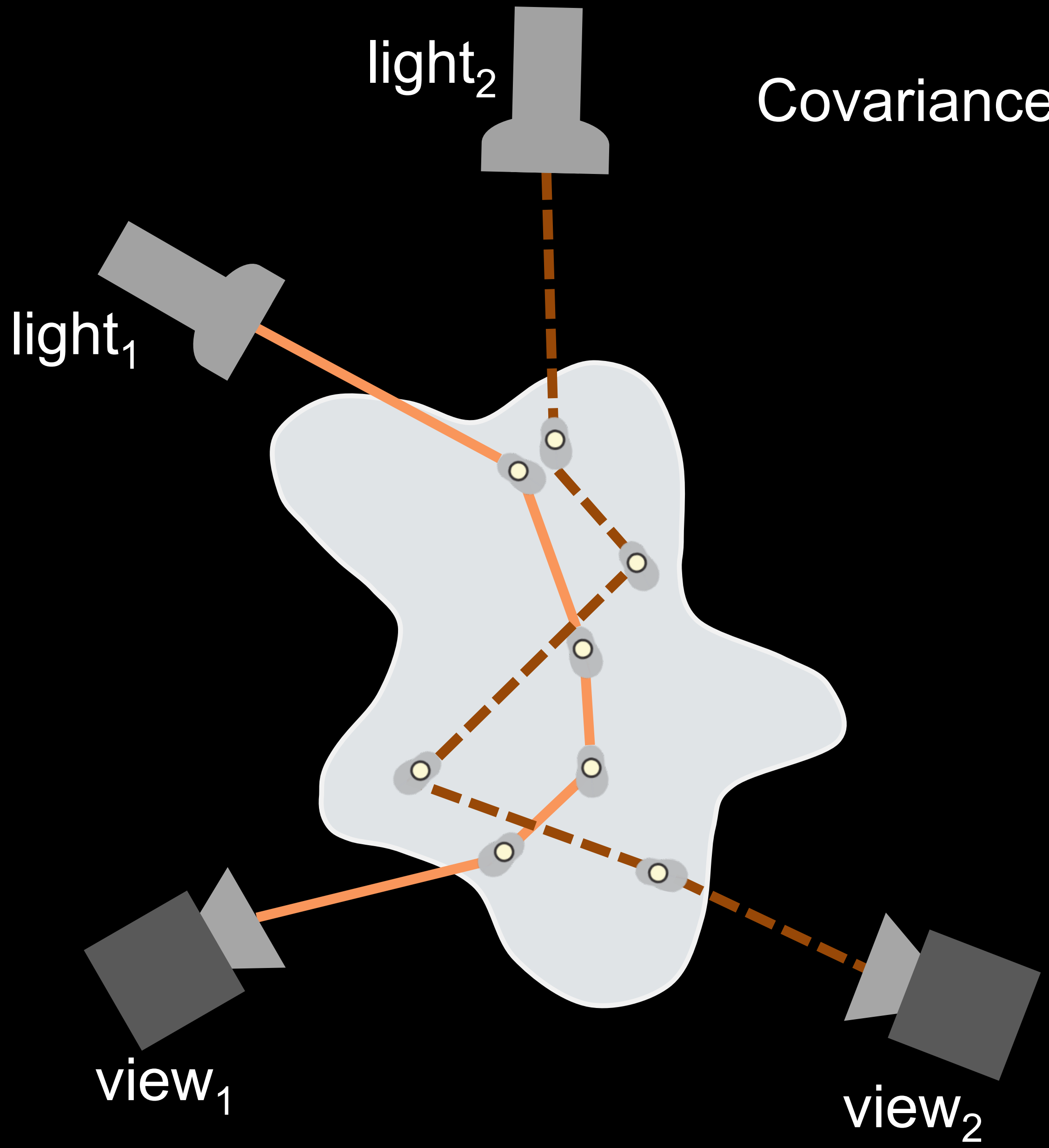
Covariance rendering

$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

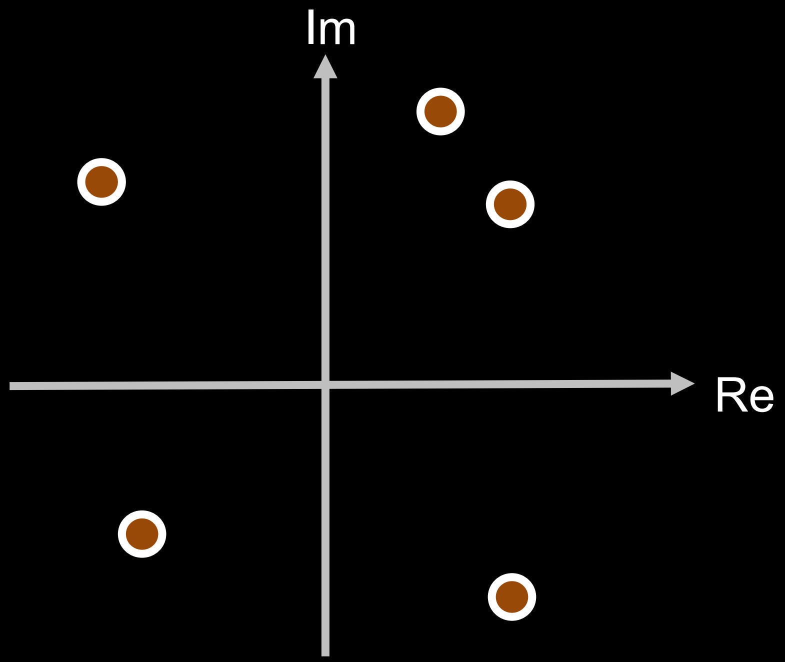




Covariance rendering

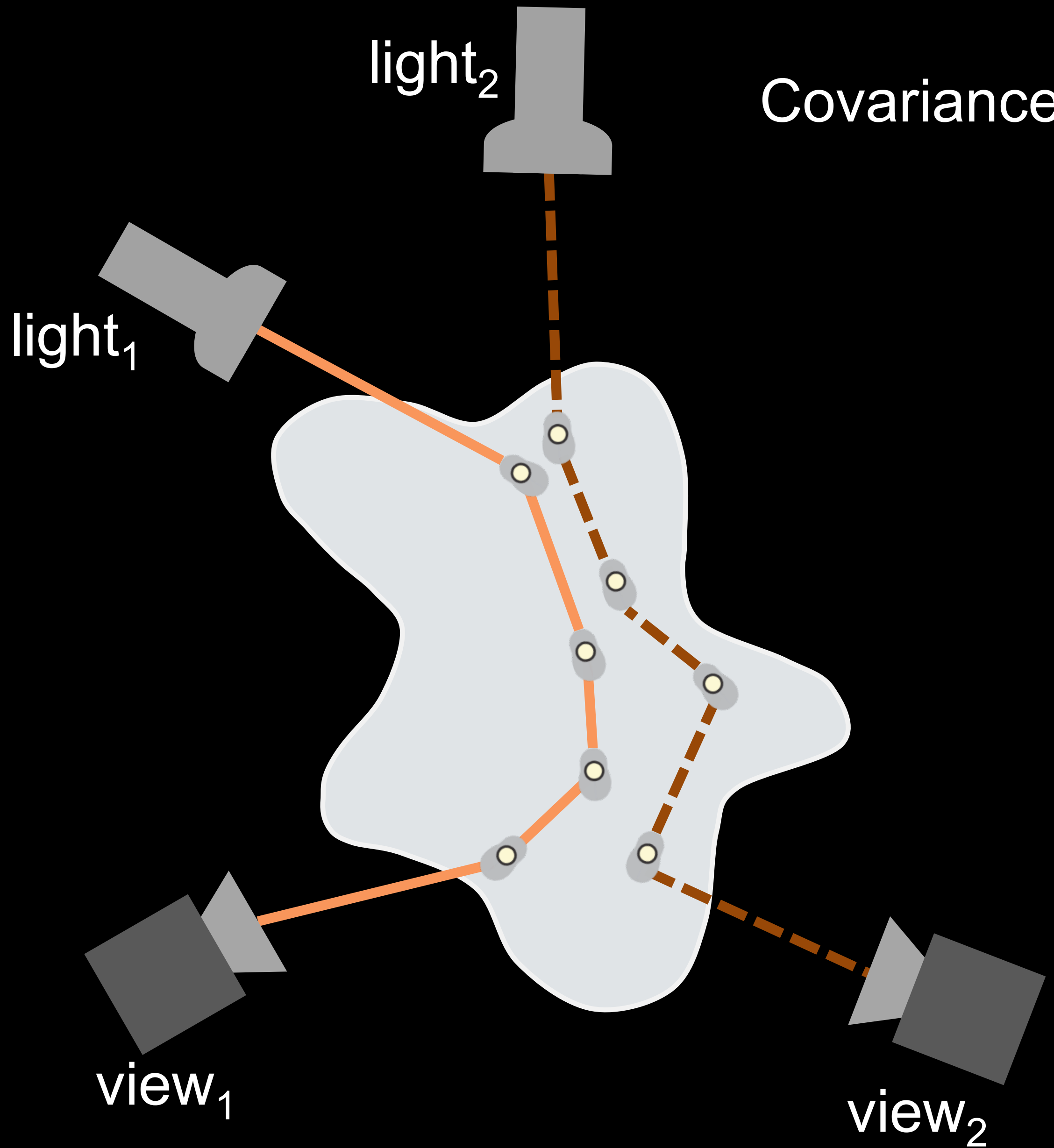


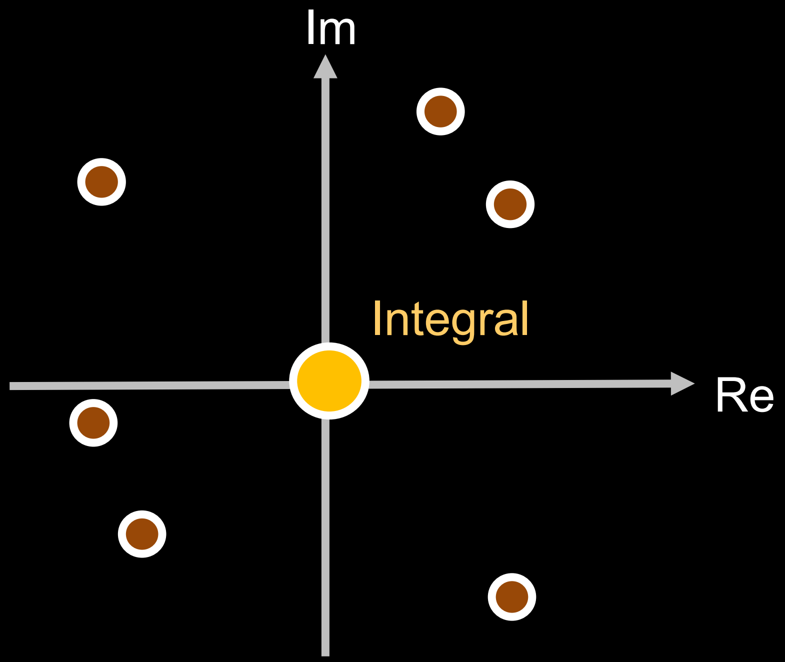
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$



Covariance rendering

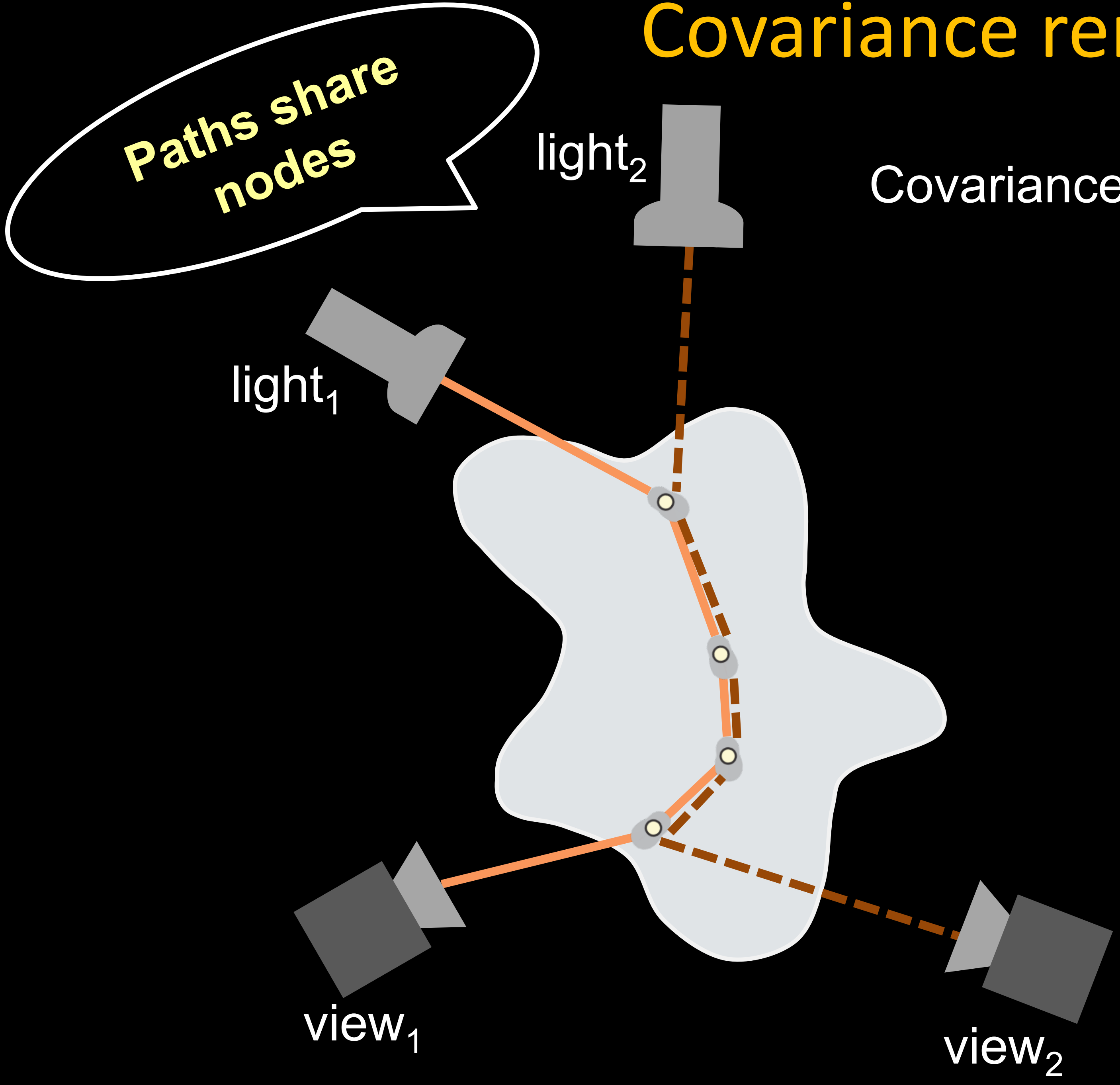
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

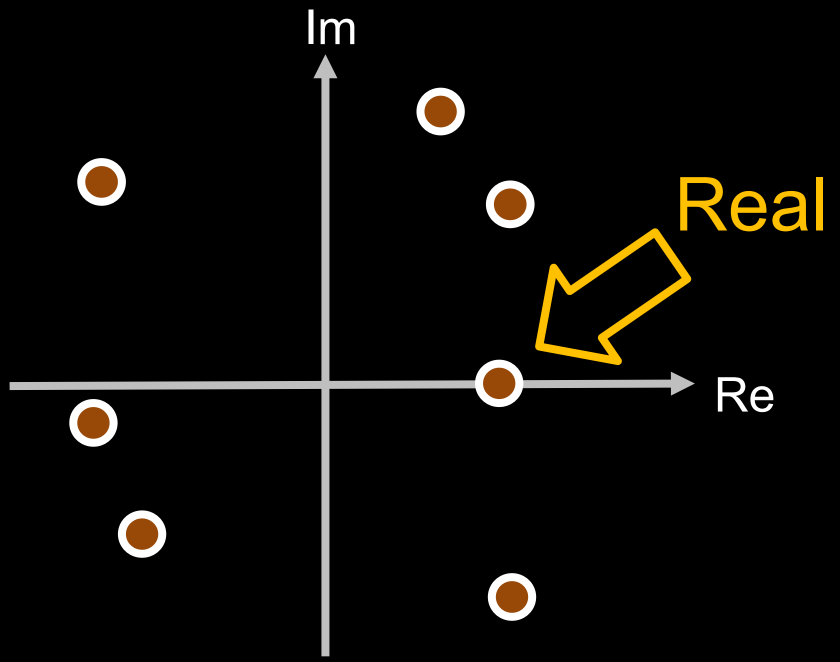




Covariance rendering

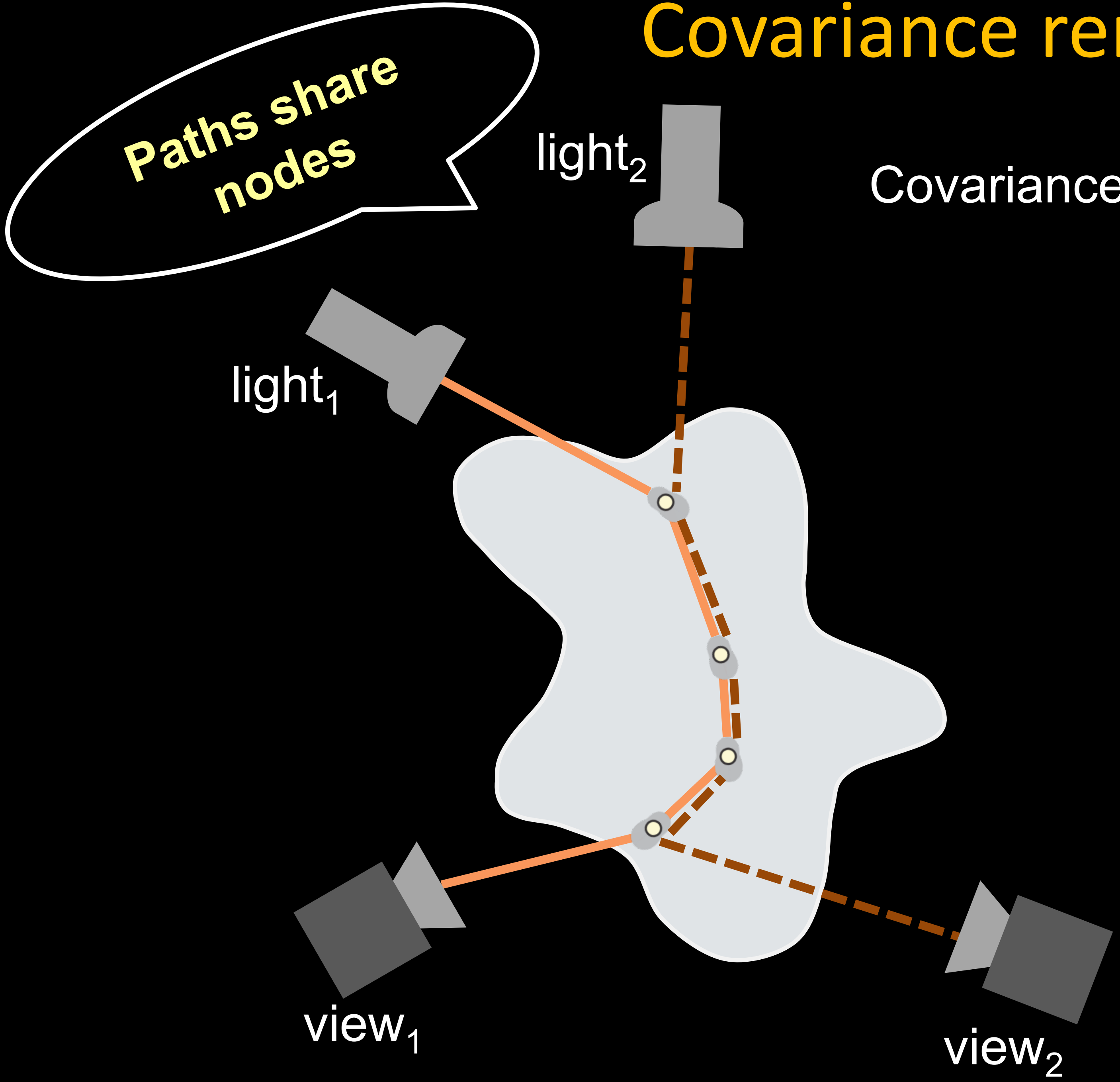
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

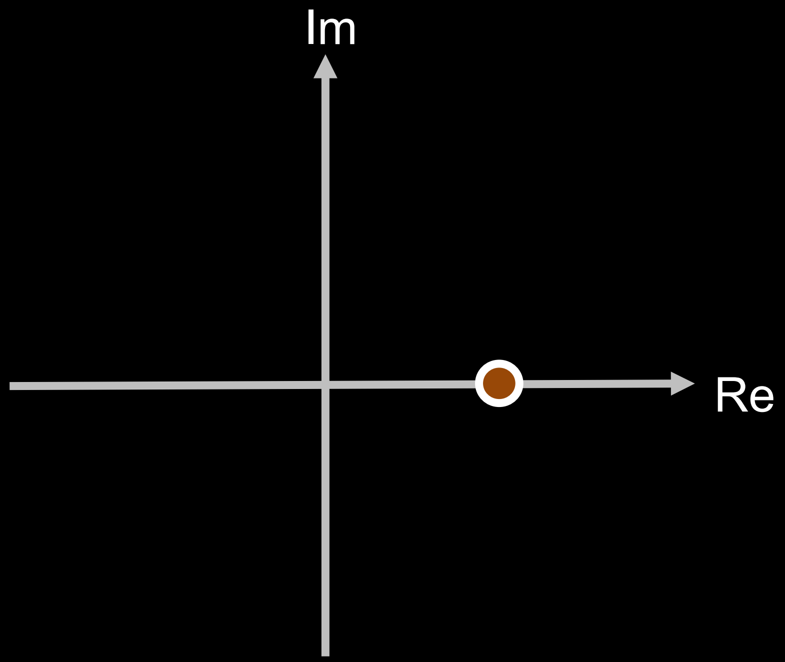




Covariance rendering

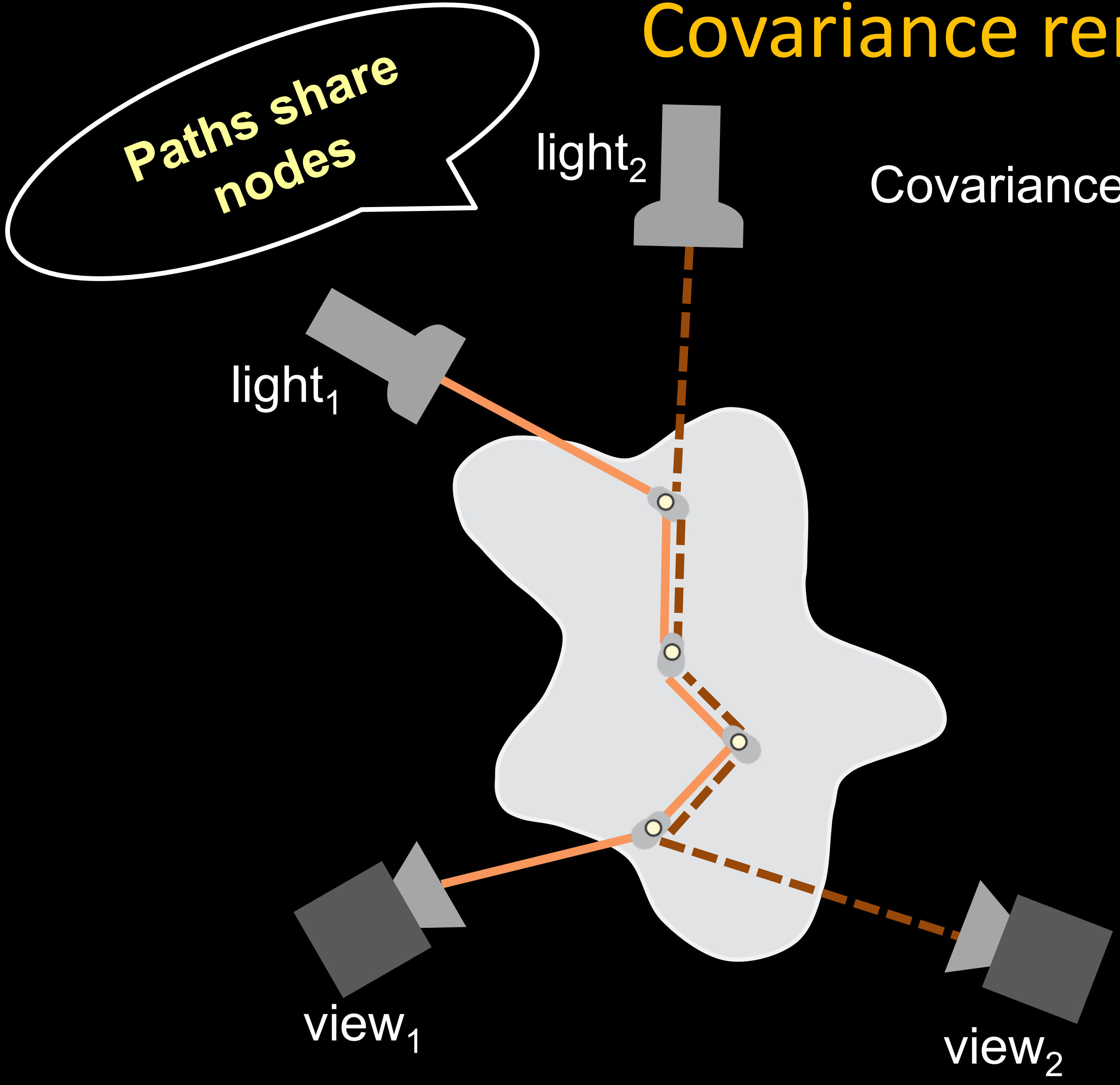
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

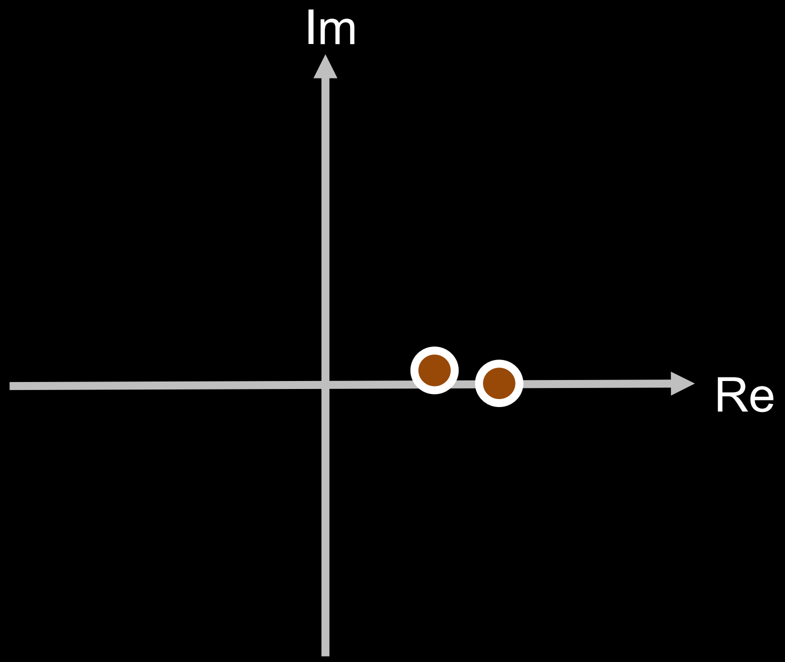




Covariance rendering

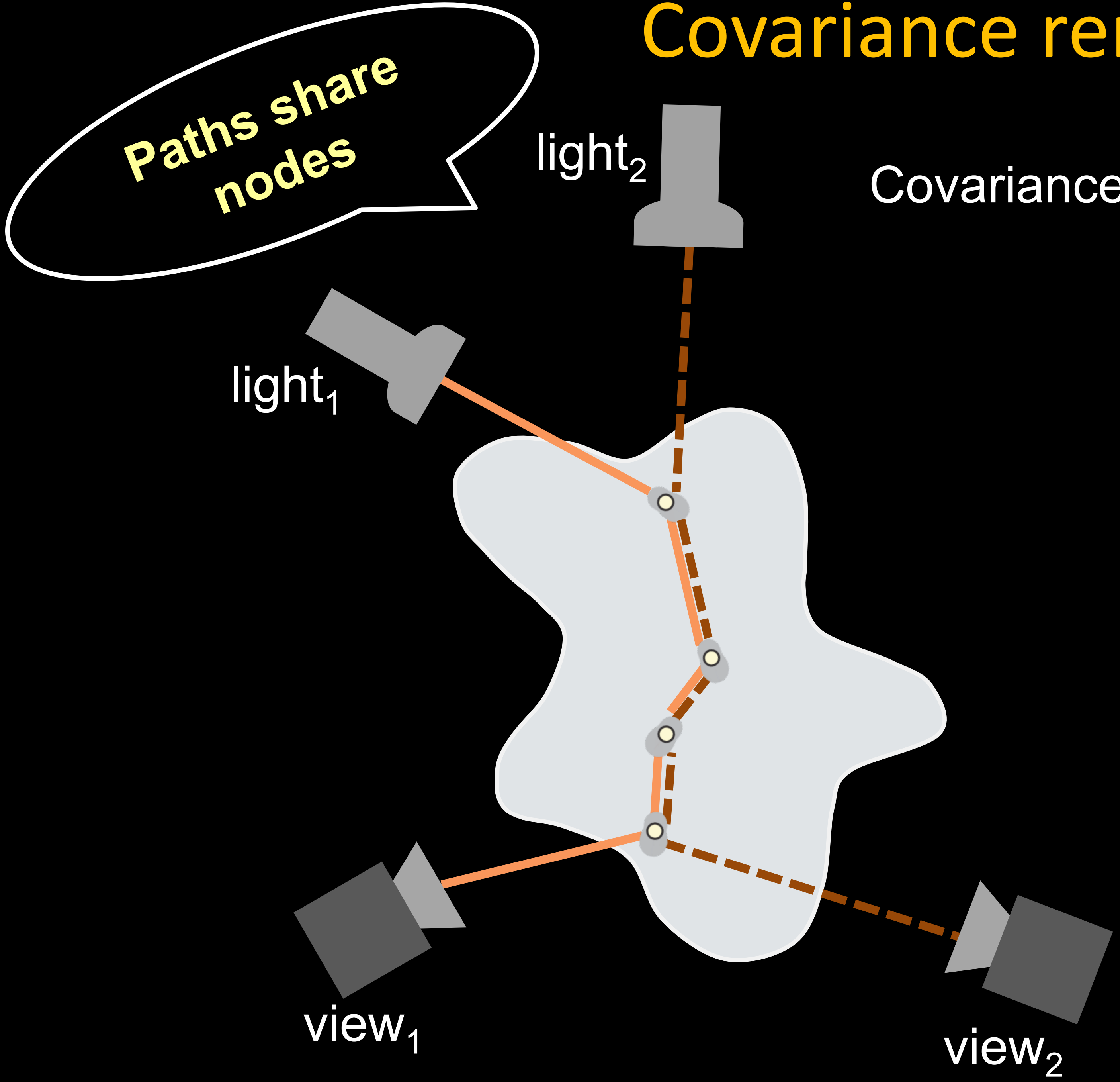
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

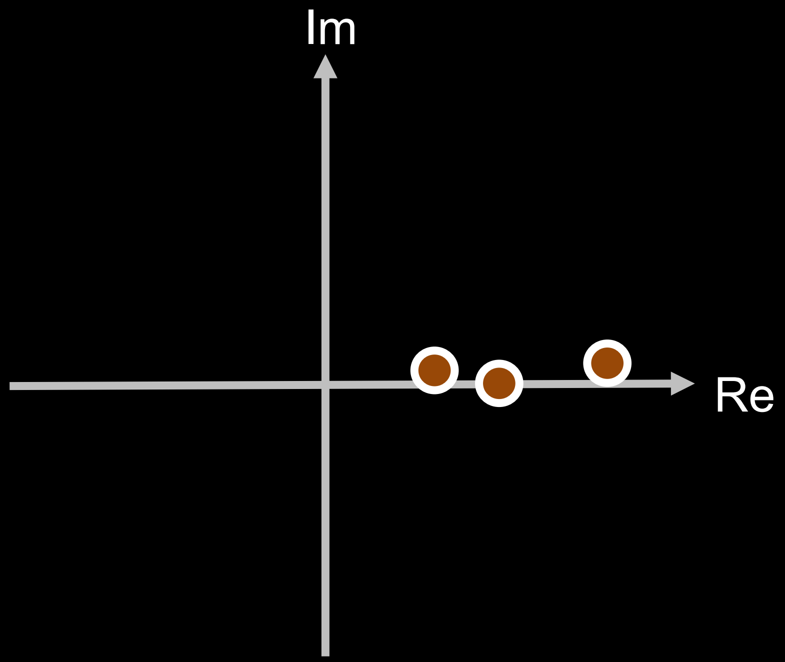




Covariance rendering

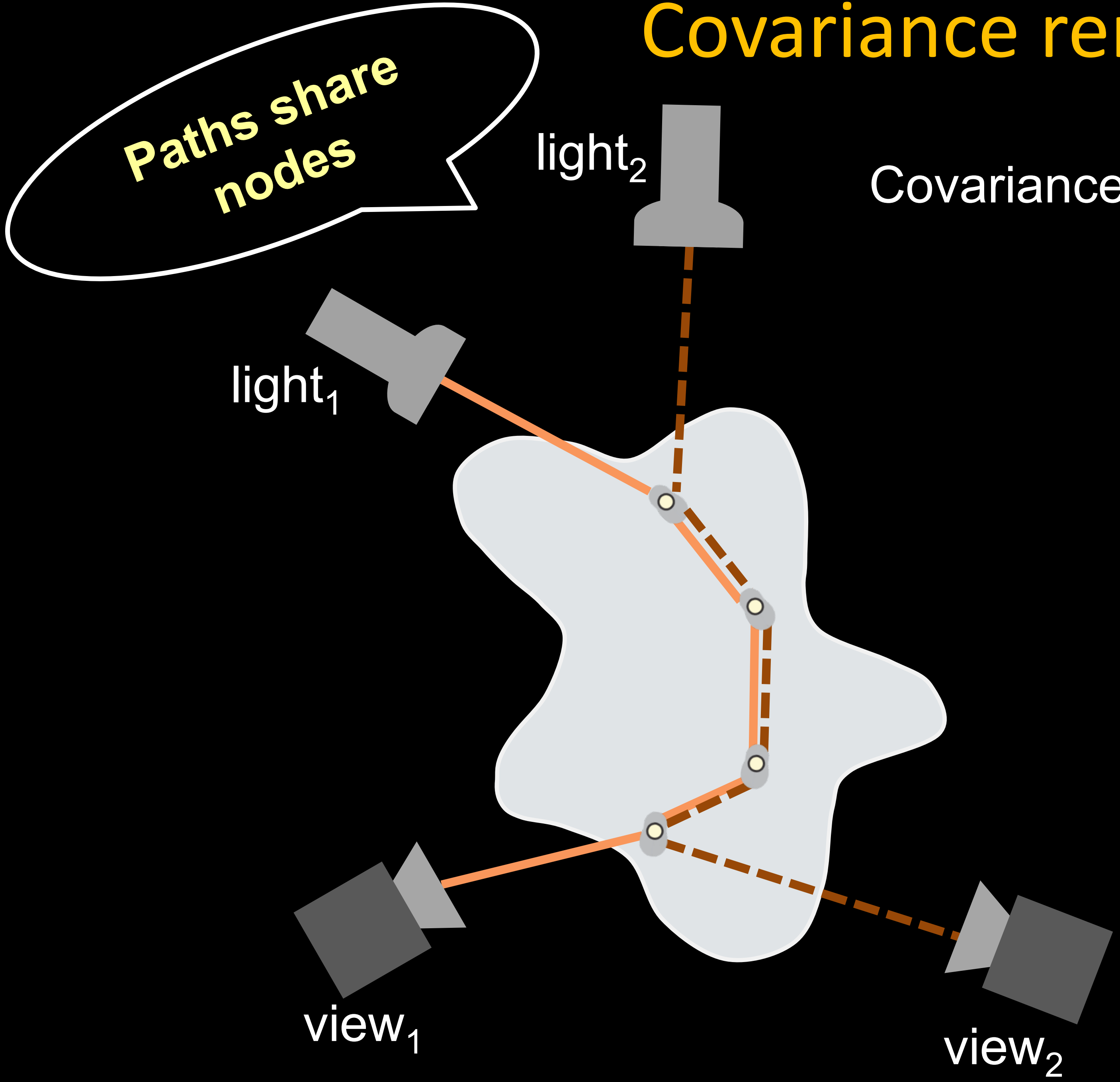
$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

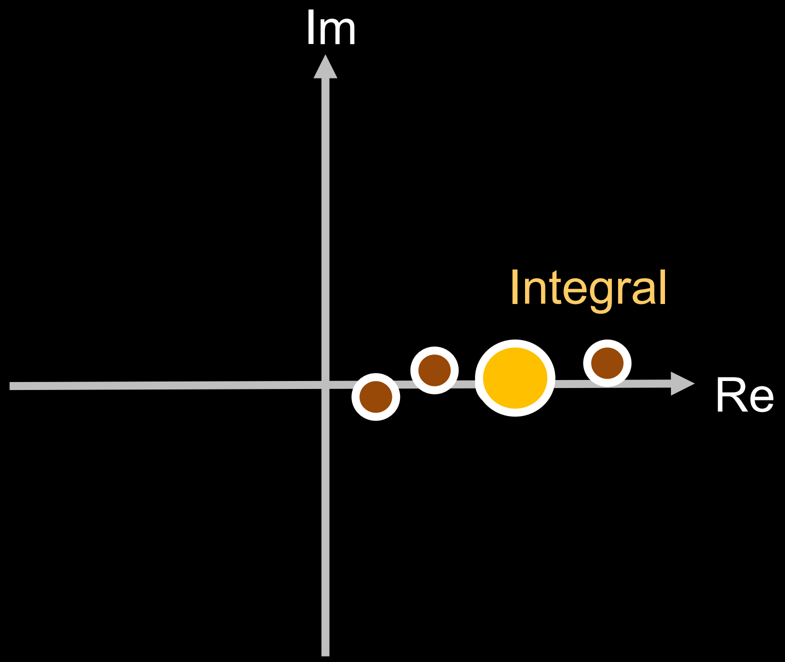




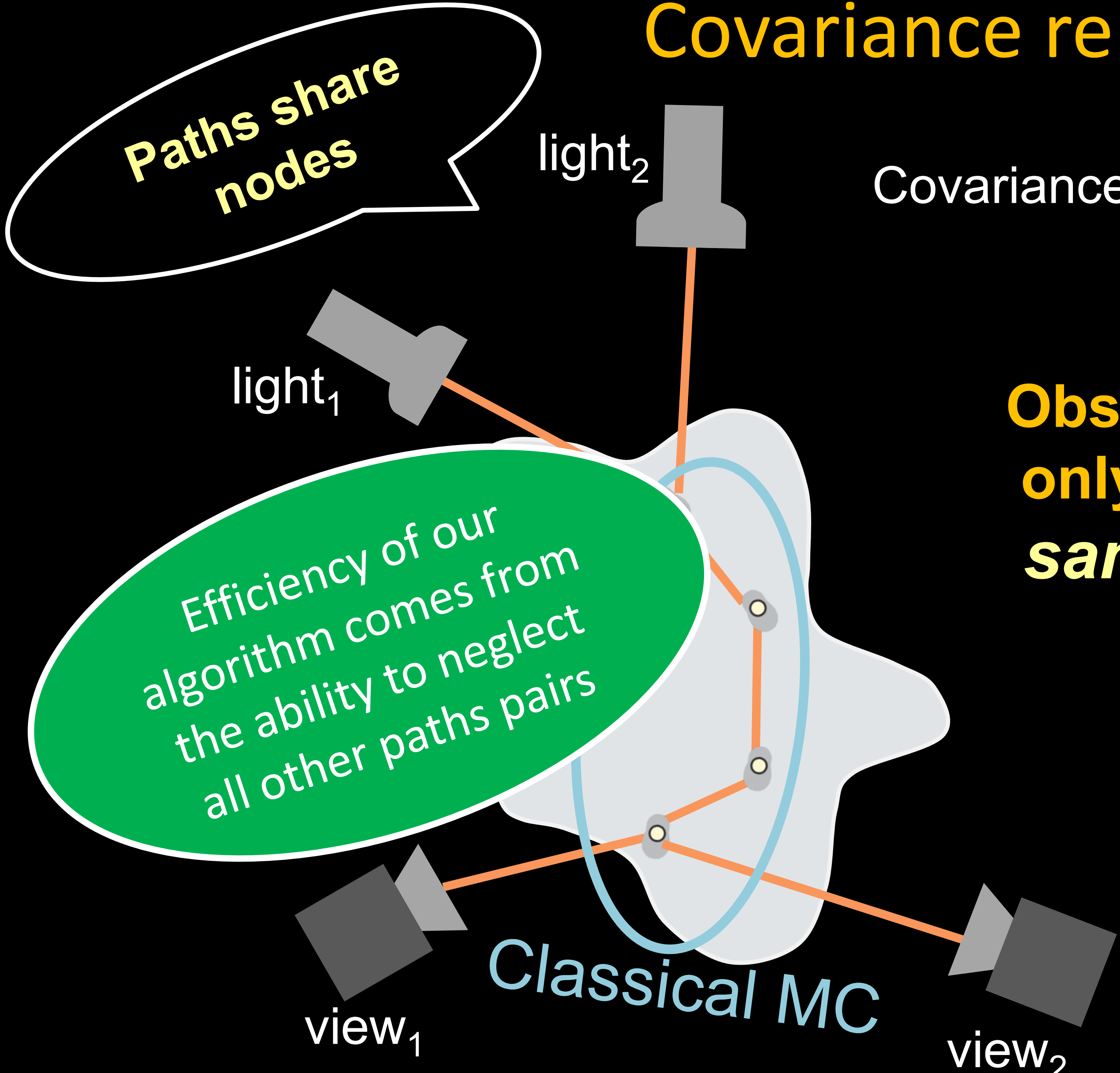
Covariance rendering

$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^{*}(\text{path}_2)$$



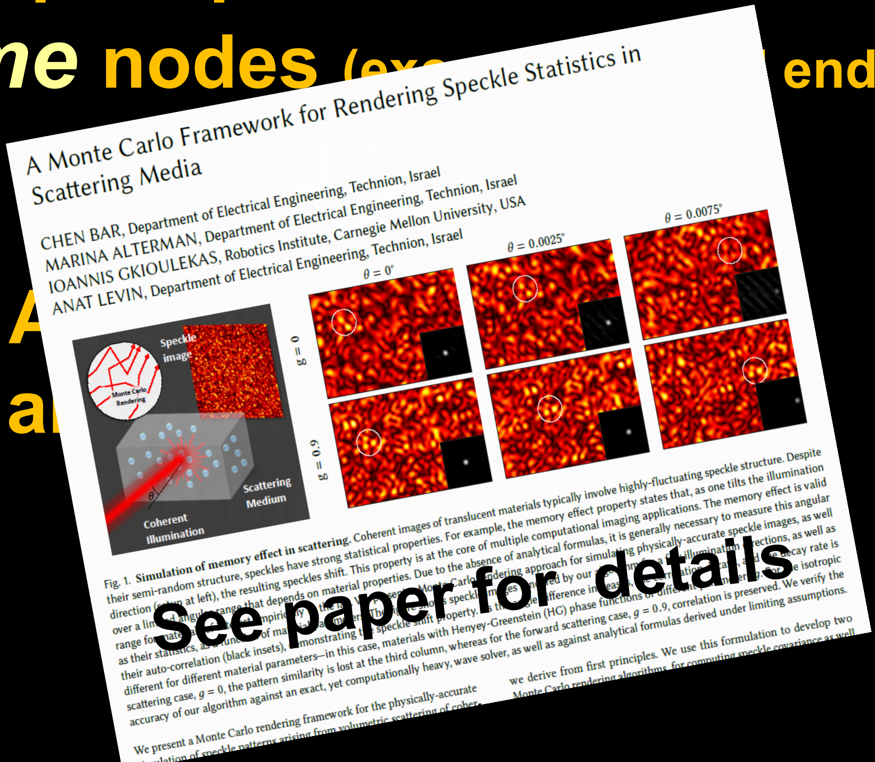


Covariance rendering



$$\text{Covariance} = \int_{\text{path}_1,\text{path}_2} u(\text{path}_1) \cdot u^*(\text{path}_2)$$

Observation: need to consider only path pairs that share the *same* nodes (exc... end)



A Monte Carlo Framework for Rendering Speckle Statistics in Scattering Media

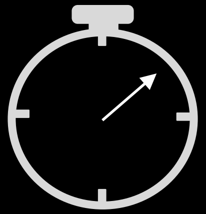
CHEN BAR, Department of Electrical Engineering, Technion, Israel
MARINA ALTERMAN, Department of Electrical Engineering, Technion, Israel
IOANNIS GKIOULEKAS, Robotics Institute, Carnegie Mellon University, USA
ANAT LEVIN, Department of Electrical Engineering, Technion, Israel

Fig. 1. **Simulation of memory effect in scattering.**

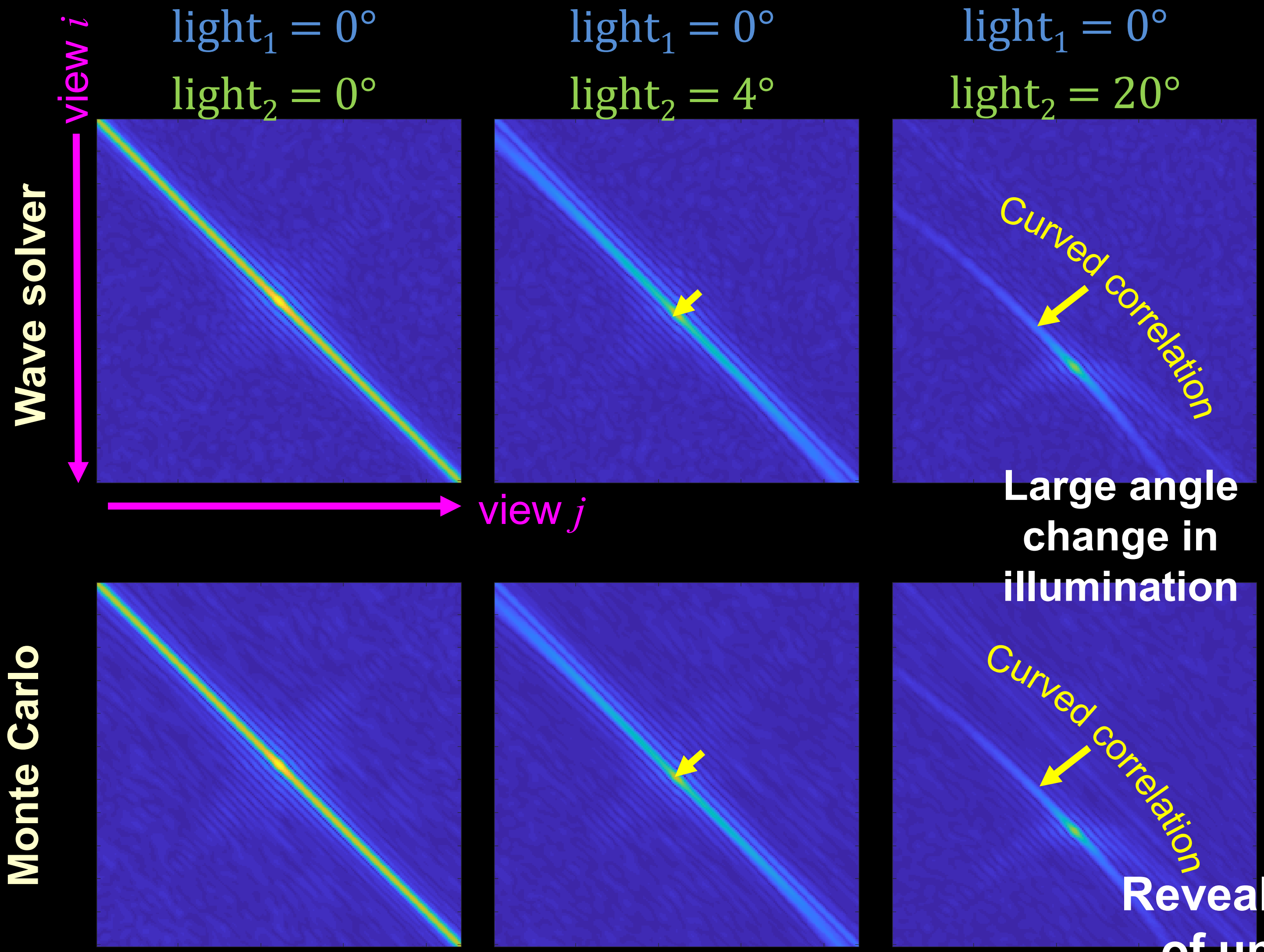
See paper for details

Validation: Wave Equation Covariances v.s. MC

Computation takes days



Several minutes

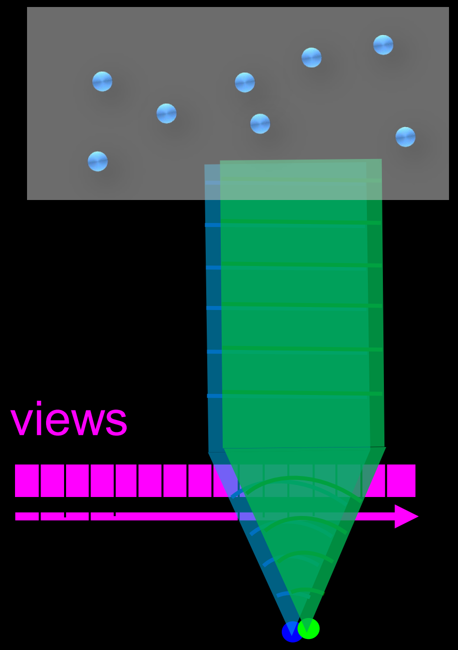


Large angle change in illumination

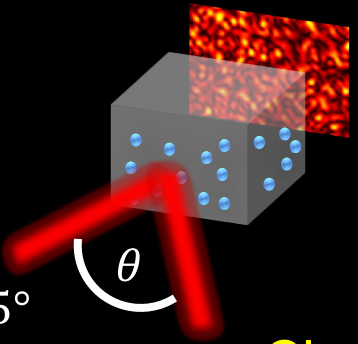
Reveal new types of unexplored correlations

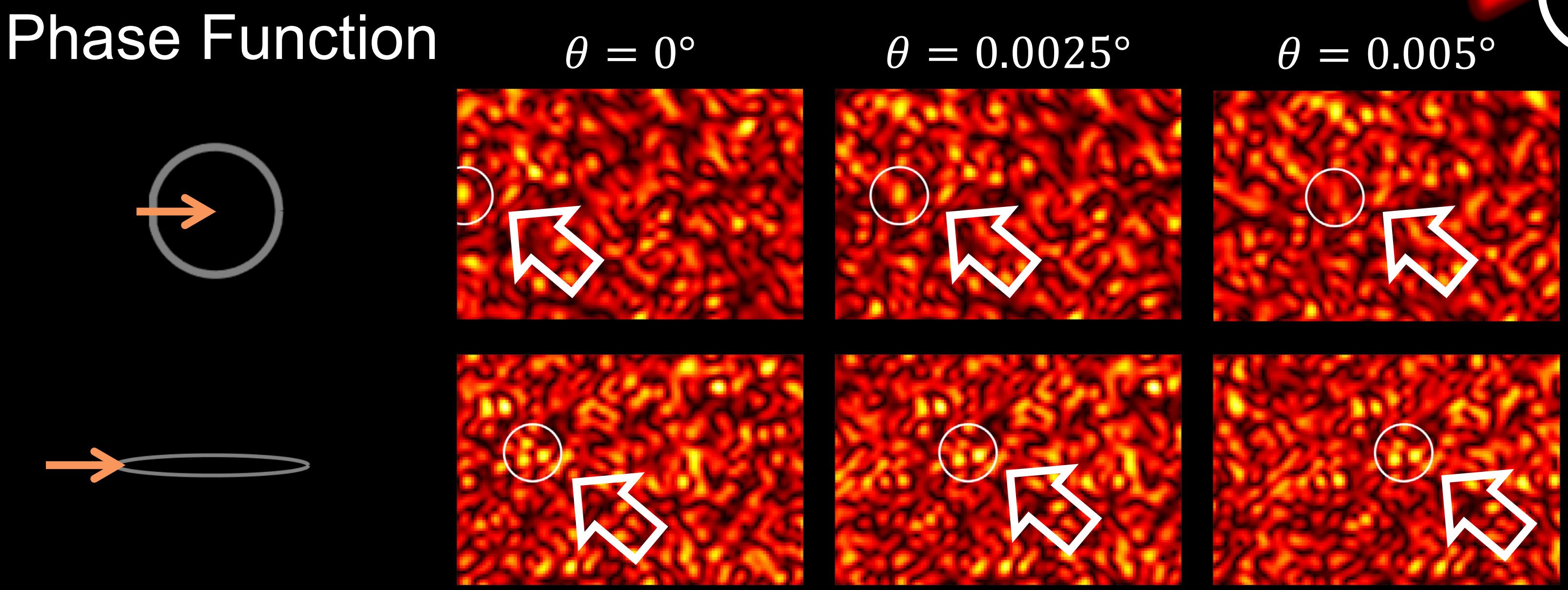
Classical ME holds for relatively small angles

Setup



Rendering Speckles

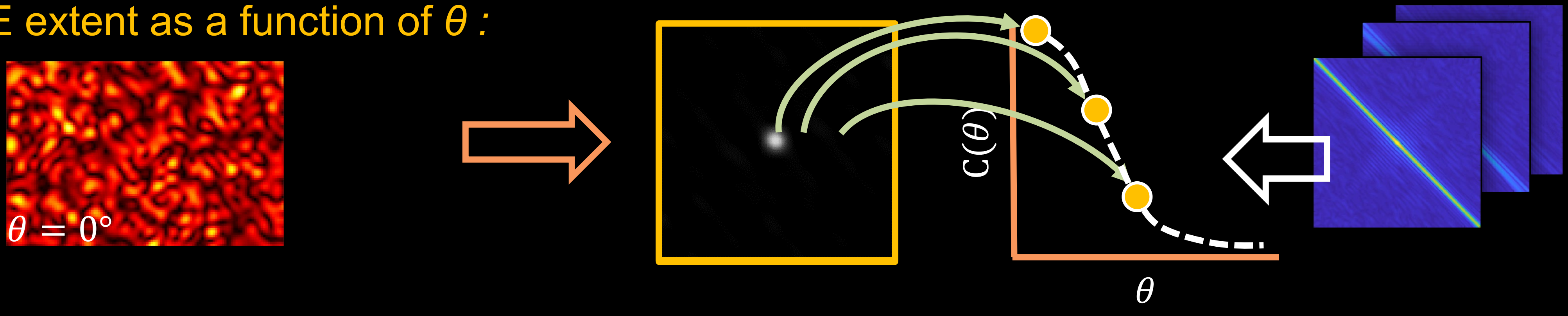




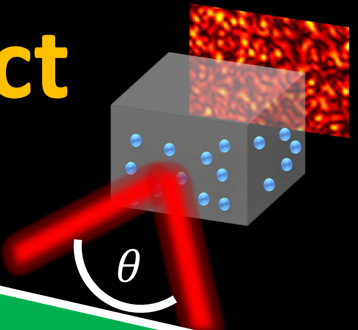
Classical ME holds for relatively small angles

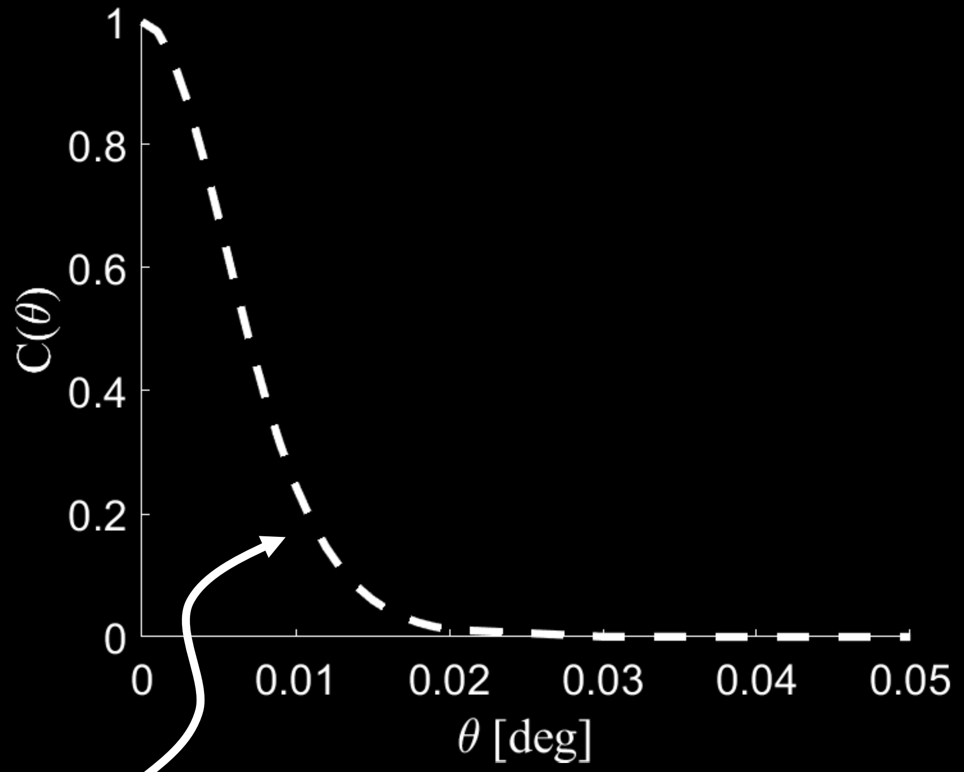
Exact ME extent is different for different materials.

Computing ME extent as a function of θ :



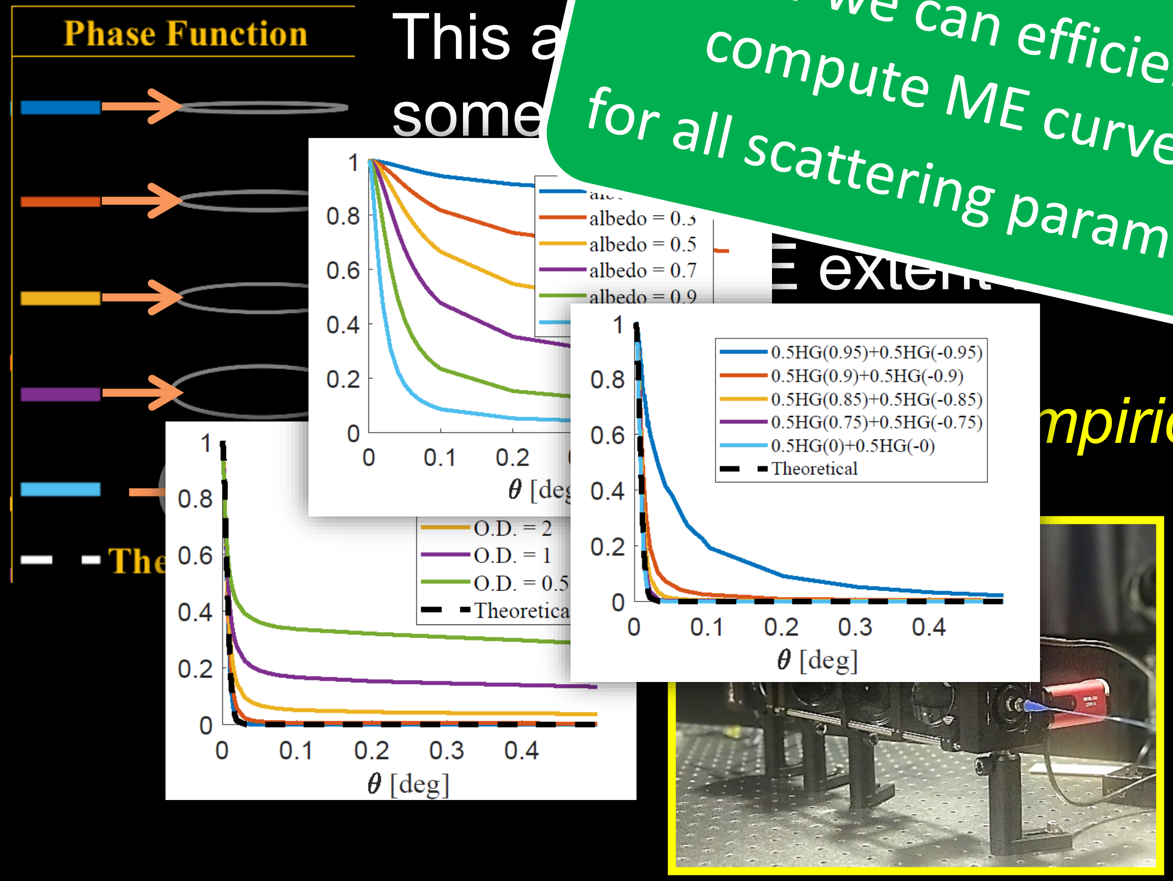
Evaluating the Memory Effect





Analytical solution based on diffusion approximation

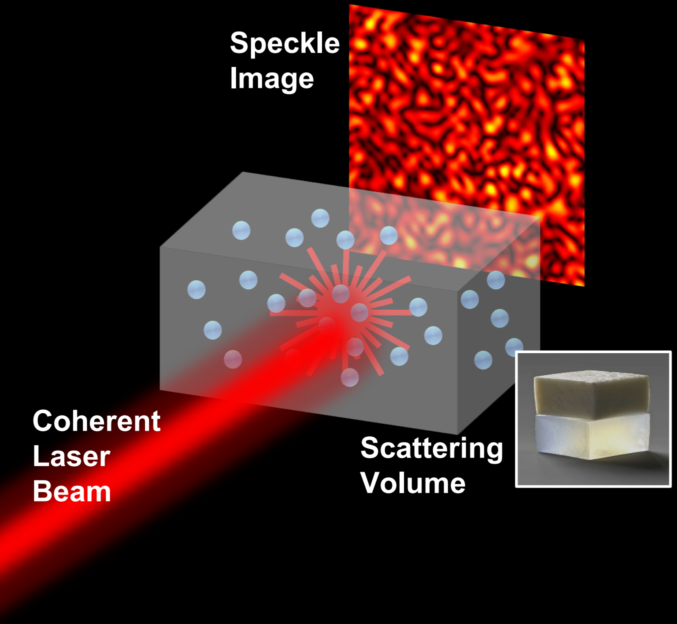
This a…
some…
…E extent…
…er
…mpirically in



Now we can efficiently compute ME curves for all scattering parameters

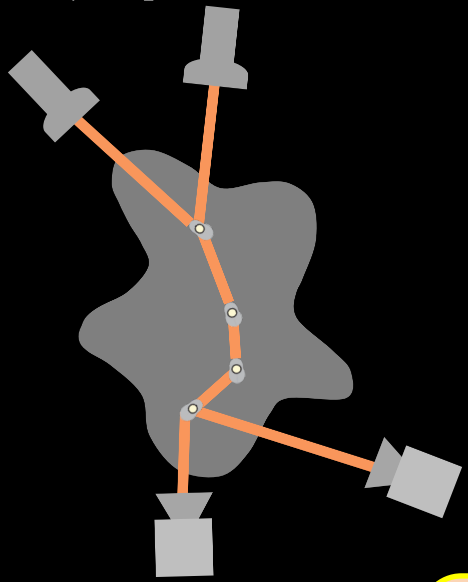
Summary

Problem:
Coherent Scattering

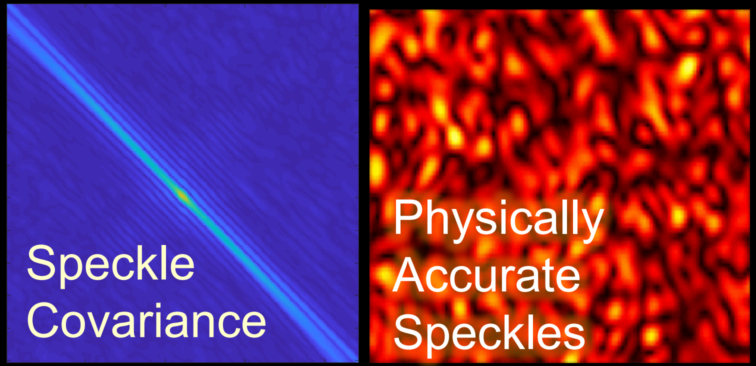


Path-integral formulation for speckle covariance

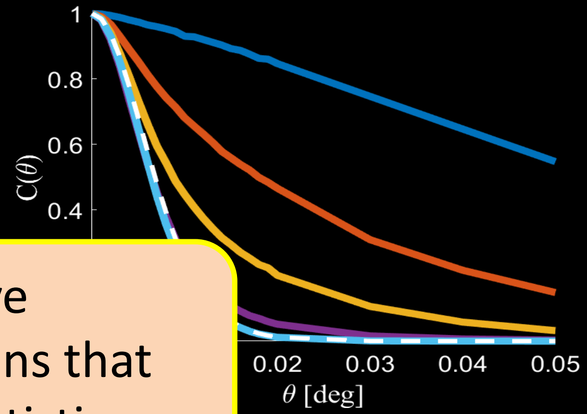
$$\text{Covariance} = \int_{path_1, path_2} u(path_1) \cdot u^{*}(path_2)$$



Efficient MC Rendering



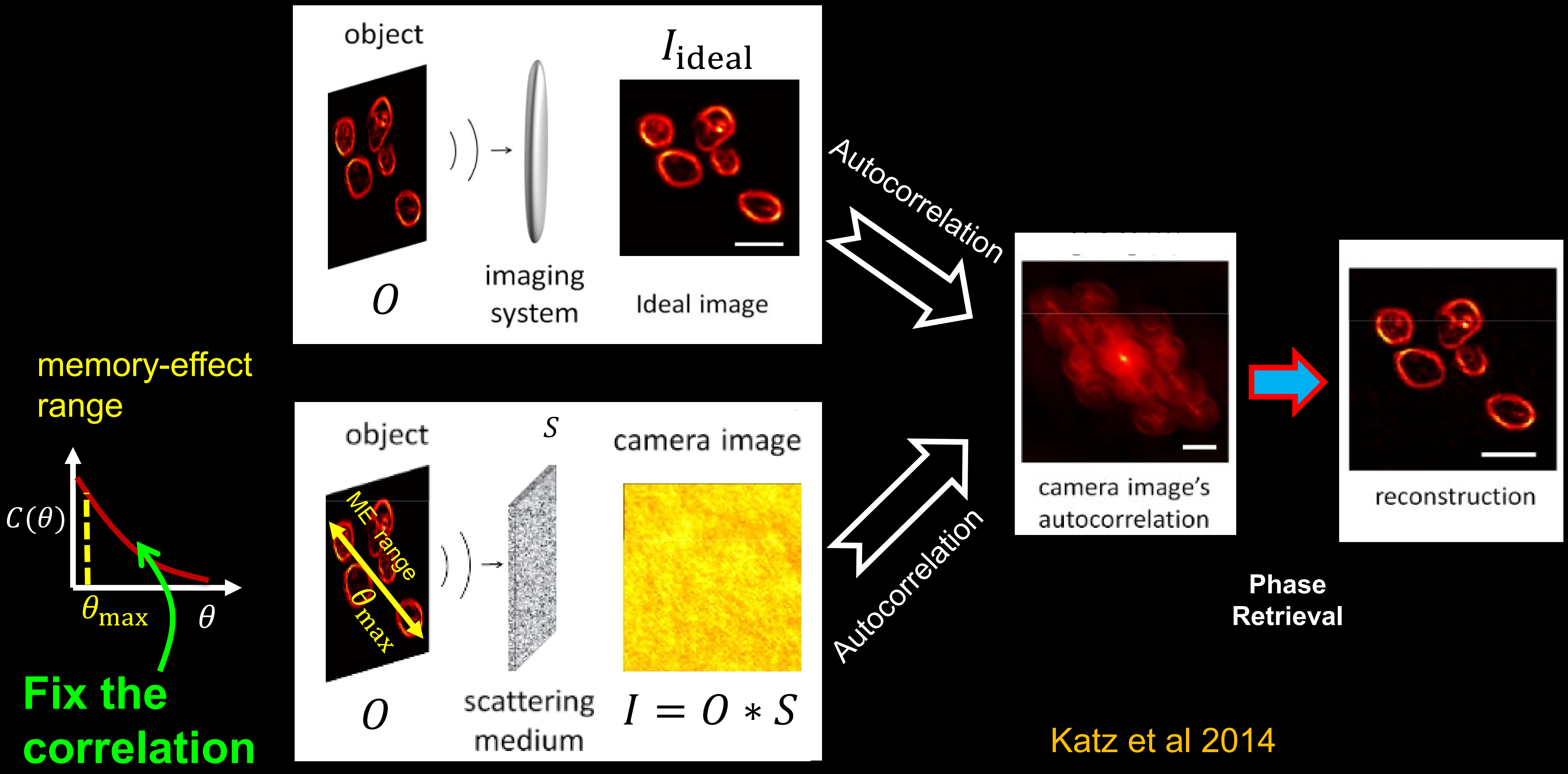
Memory Effect Evaluation



Potentially improve imaging applications that rely on speckle statistics

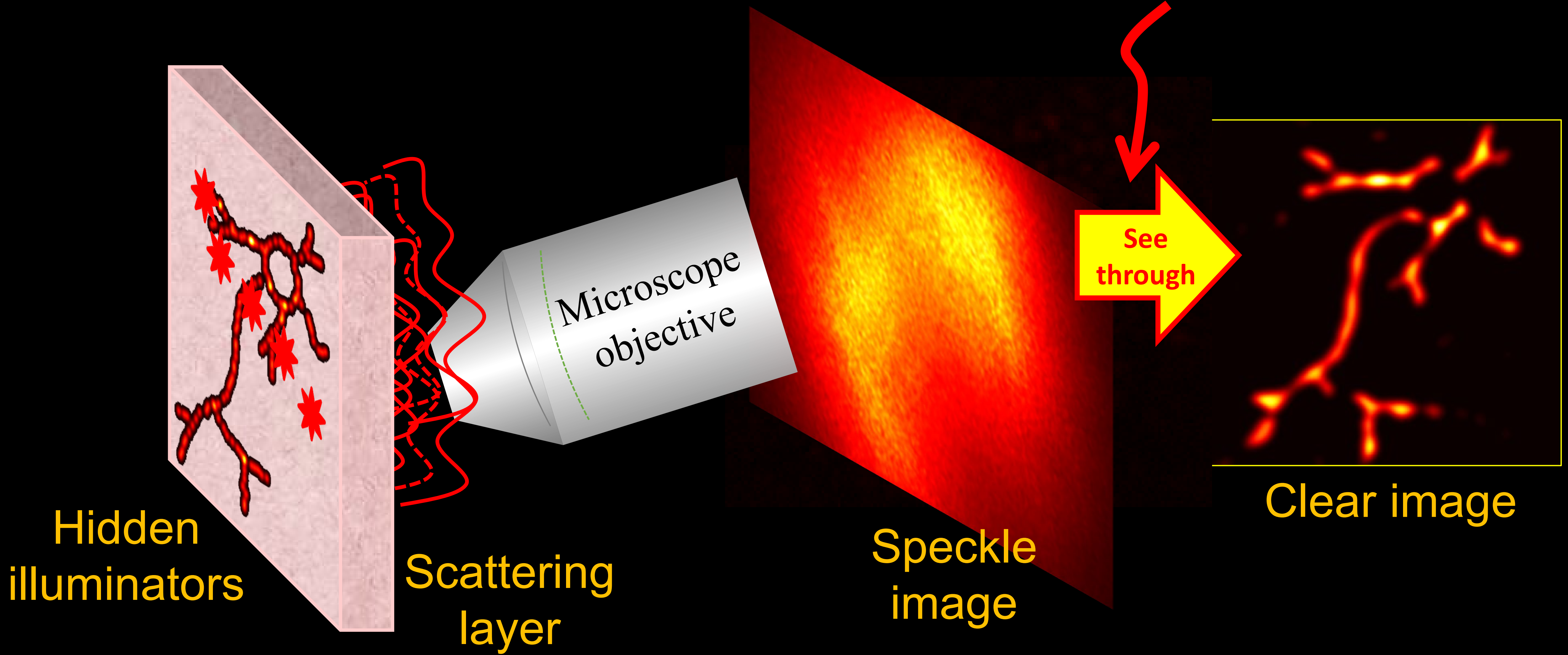
Seeing Through Scattering Layers

Cool Application by Ori Katz et al 2014

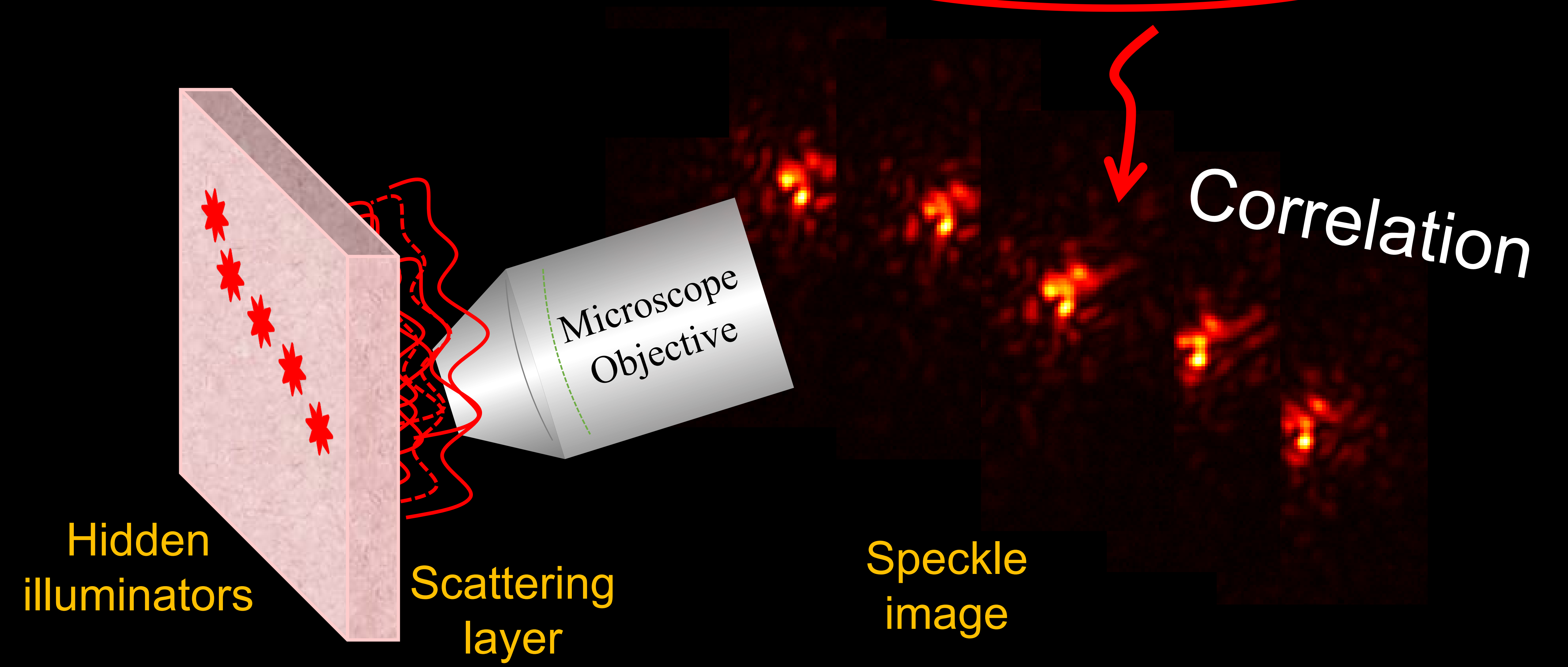


Katz et al 2014

Coherent scattering and memory effect (ME)

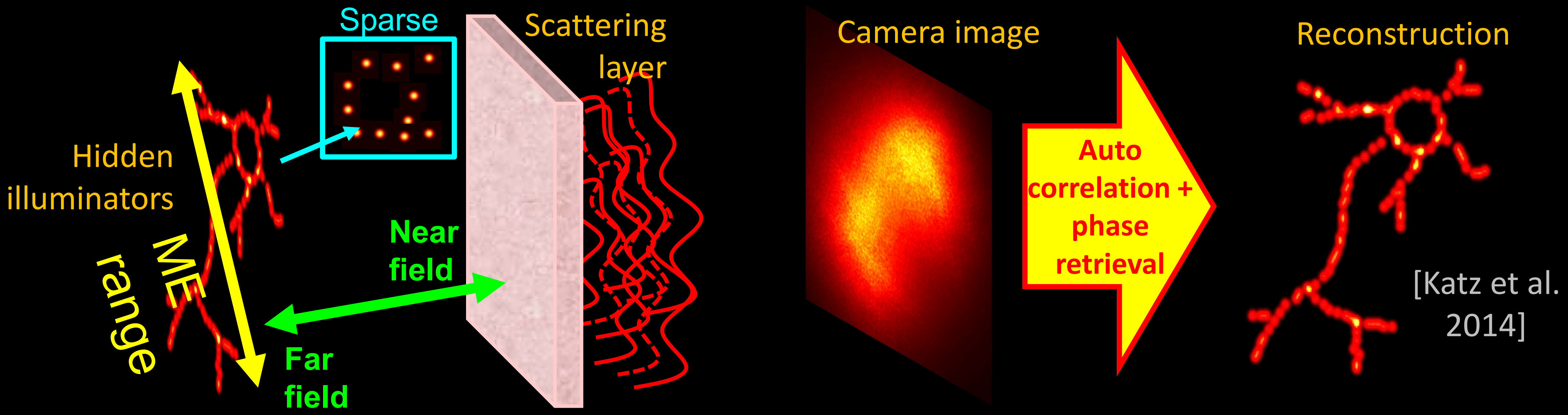


Coherent scattering and memory effect (ME)

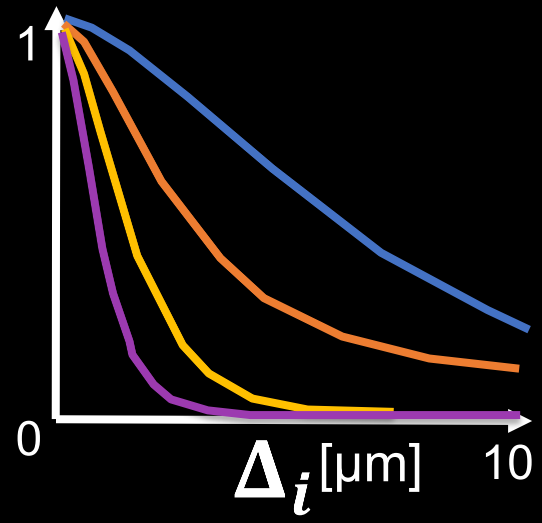


Problems with classical see-through approach

- **Limited range: Only illuminators within the ME range can be recovered.**
- **Limited density: Only a small number of illuminators can be recovered.**
- **Unrealistic setup: Far-field imaging conditions do not apply to tissue imaging.**



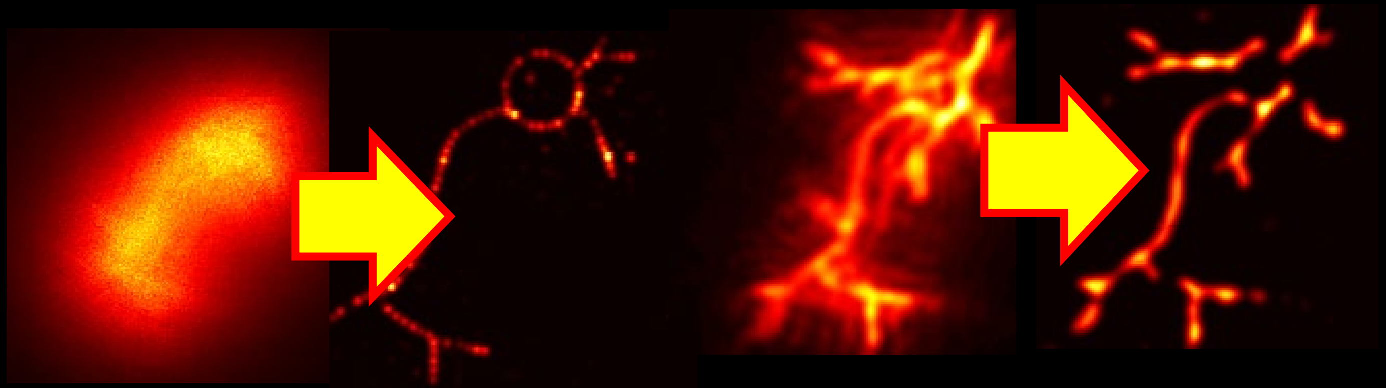
Our contributions



(i) Theoretical and rendering based analysis of ME in **near** and **far**-field settings

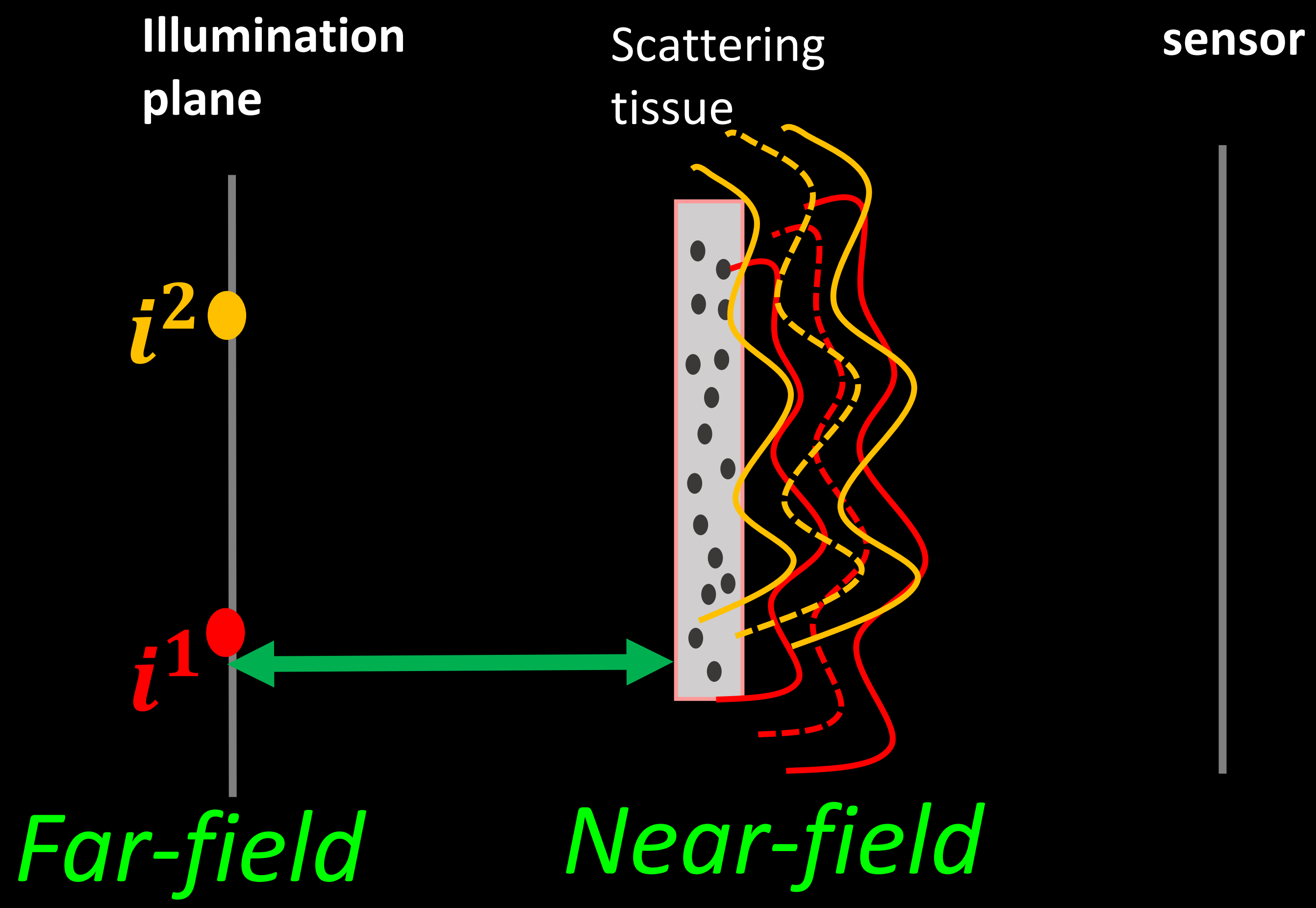
(ii) Develop a better algorithm for imaging though scattering: 1. higher density 2. wider range 3. near sources

(iii) Real Lab experiments in **near** and **far** fields



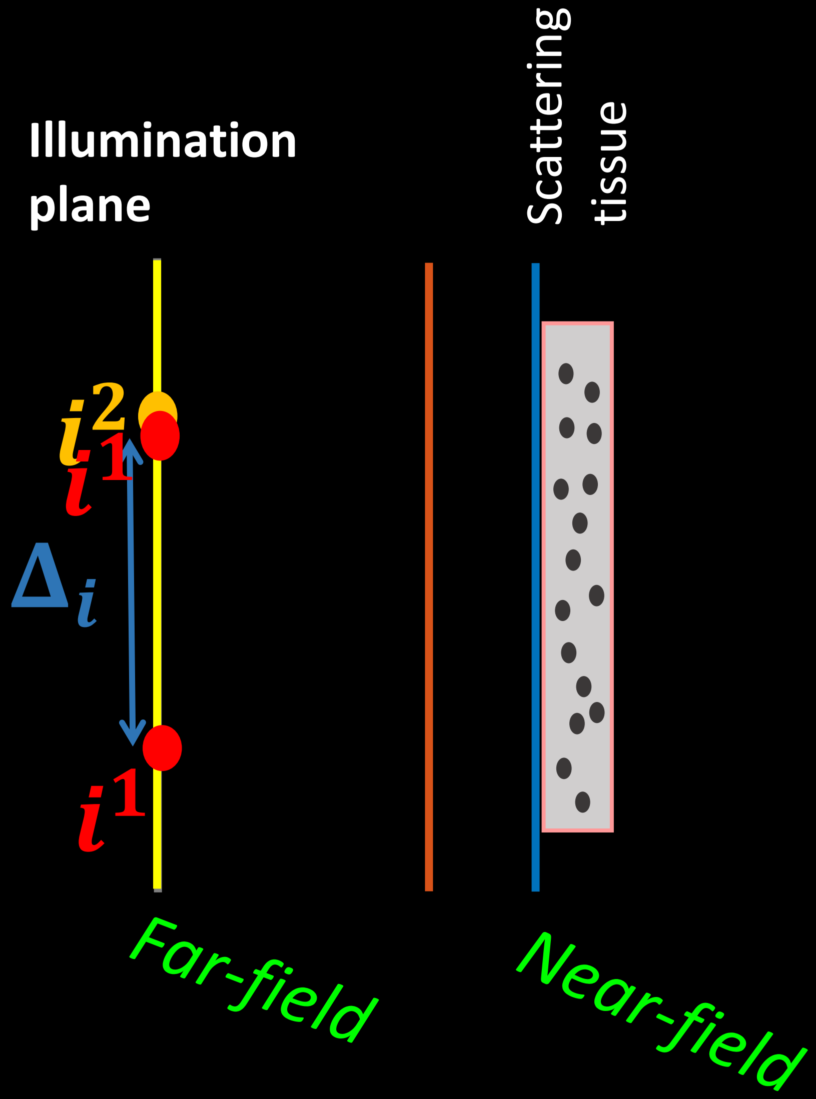
Theory and simulation analysis

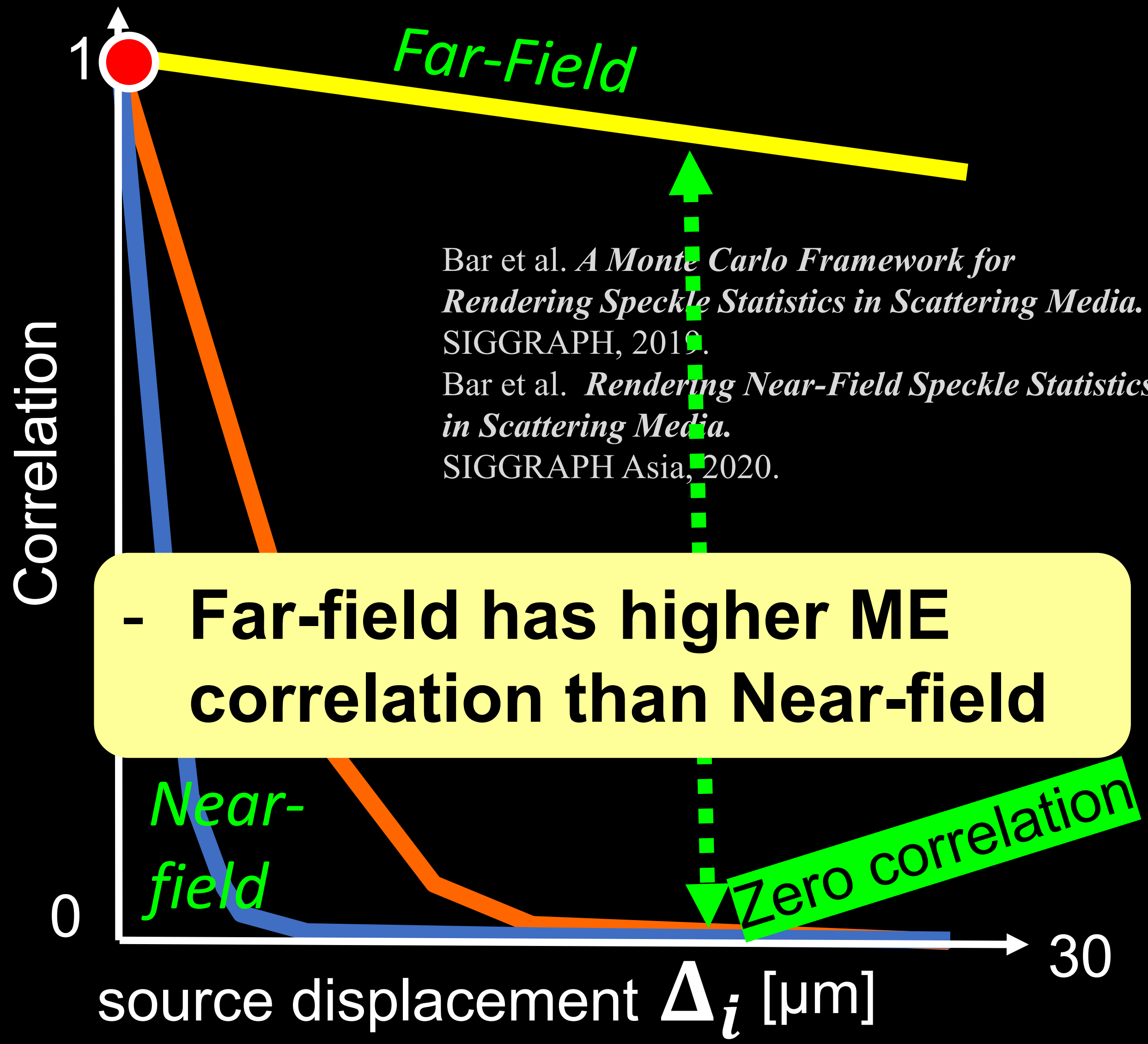
Problem setting and memory effect



Change **illumination** plane

ME correlation in near-field vs. far-field

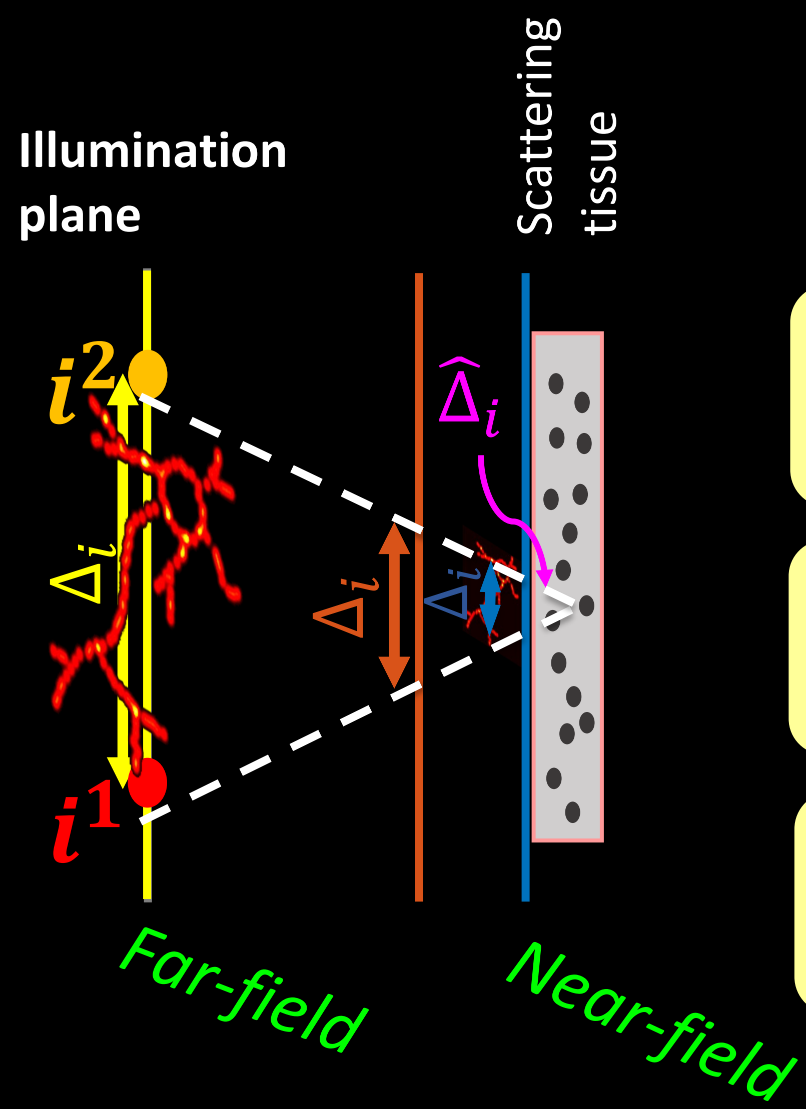


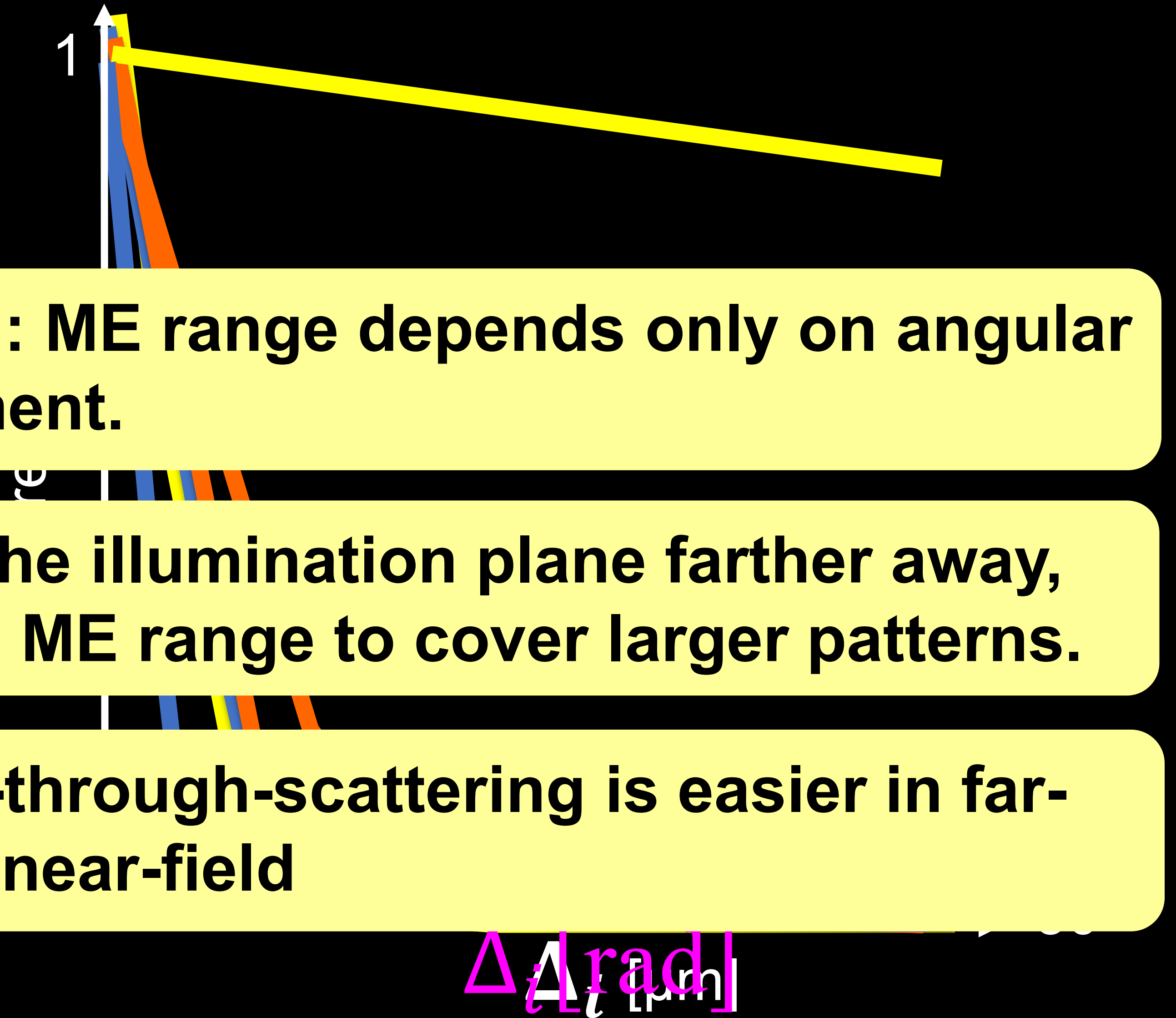


Bar et al. ***A Monte Carlo Framework for Rendering Speckle Statistics in Scattering Media.*** SIGGRAPH, 2019.
Bar et al. ***Rendering Near-Field Speckle Statistics in Scattering Media.*** SIGGRAPH Asia, 2020.

- **Far-field has higher ME correlation than Near-field**

Aligning ME correlation in the near and far fields



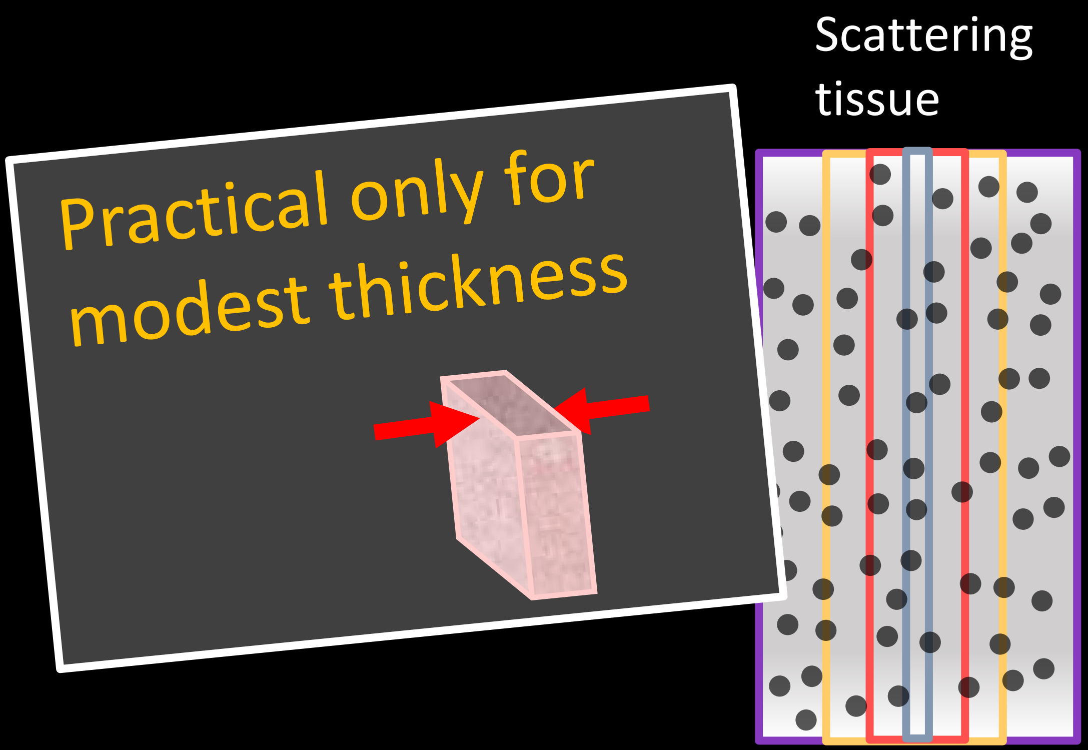


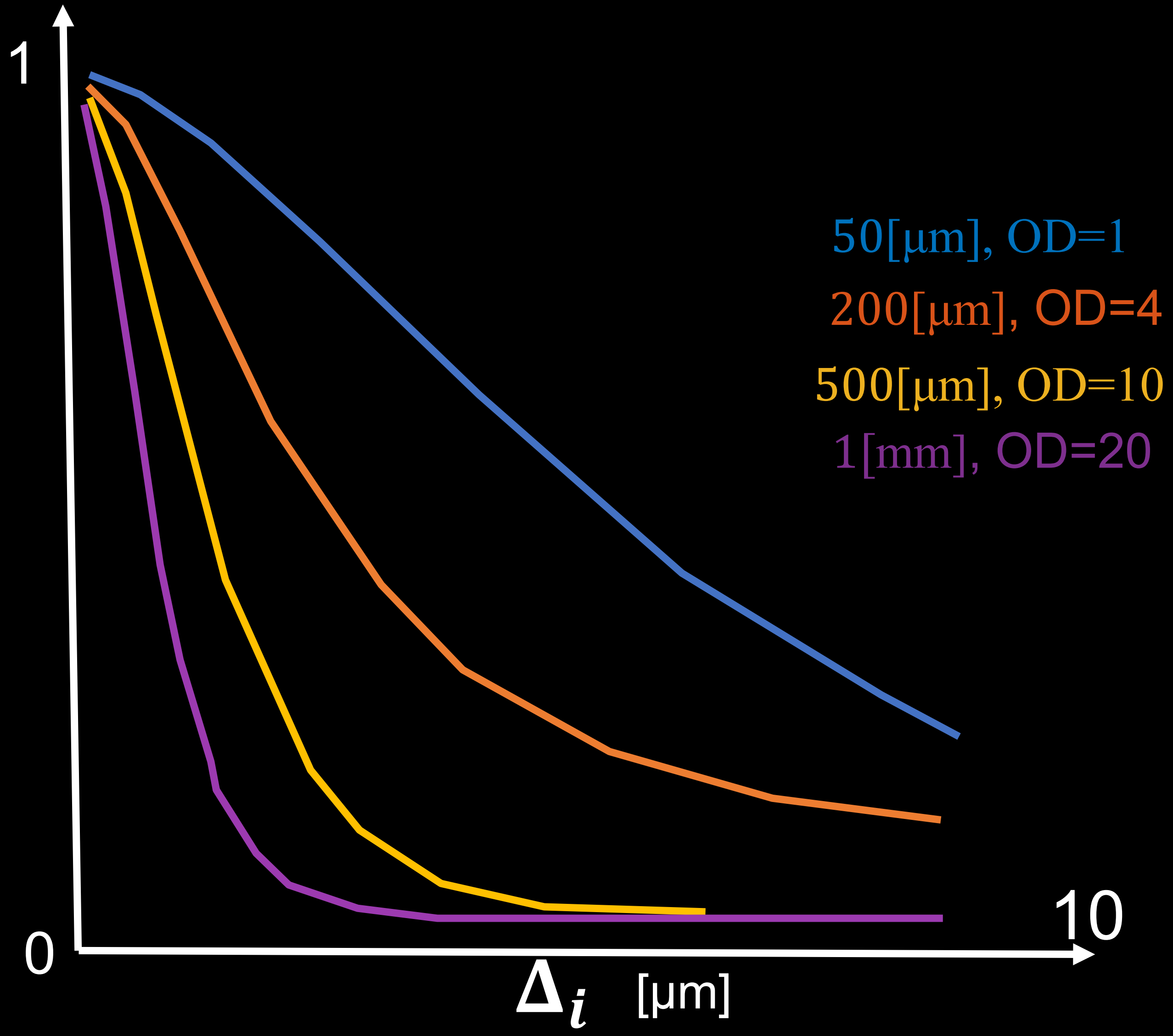
- Theorem: ME range depends only on angular displacement.

- Moving the illumination plane farther away, scales the ME range to cover larger patterns.

- Imaging-through-scattering is easier in far-field than near-field

ME correlation in the near-field

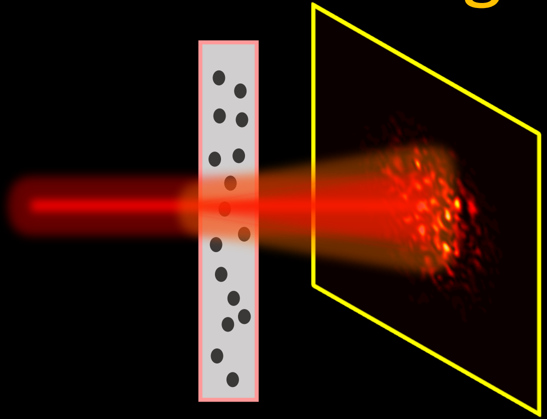




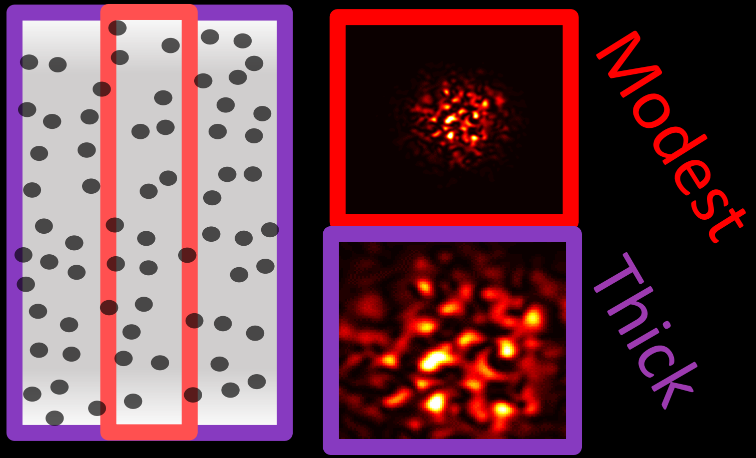
- As the thickness increases, ME correlation decreases.

Speckle Local Support

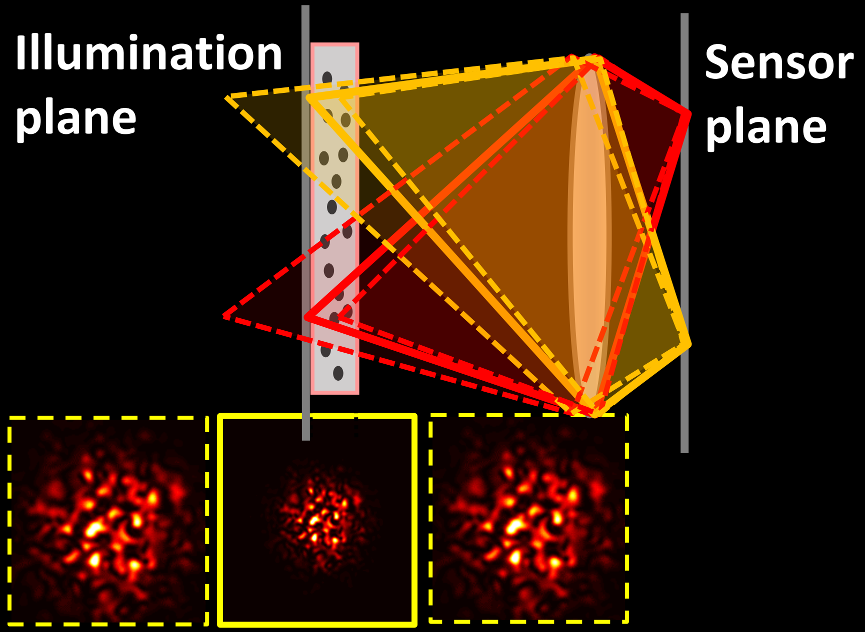
Tissue is forward scattering

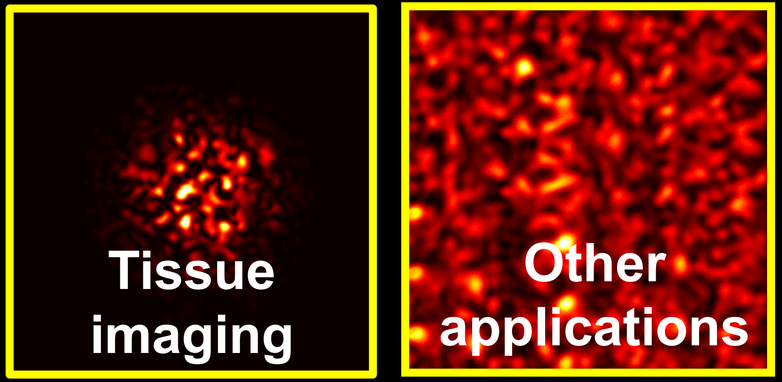


Modest thickness



Focusing on illumination

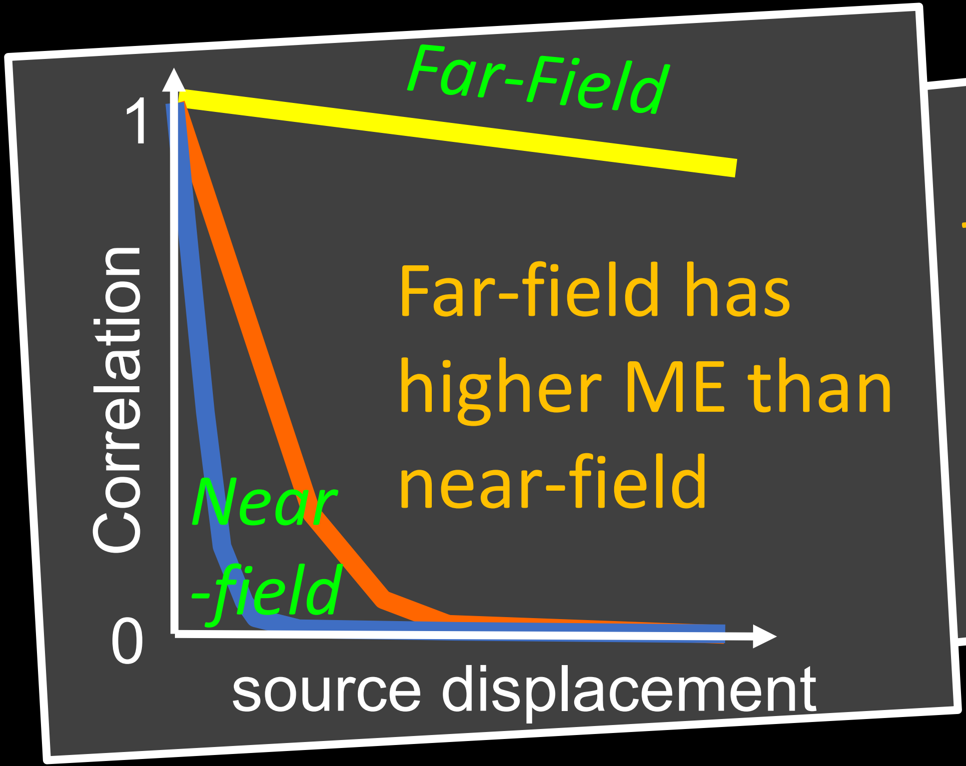


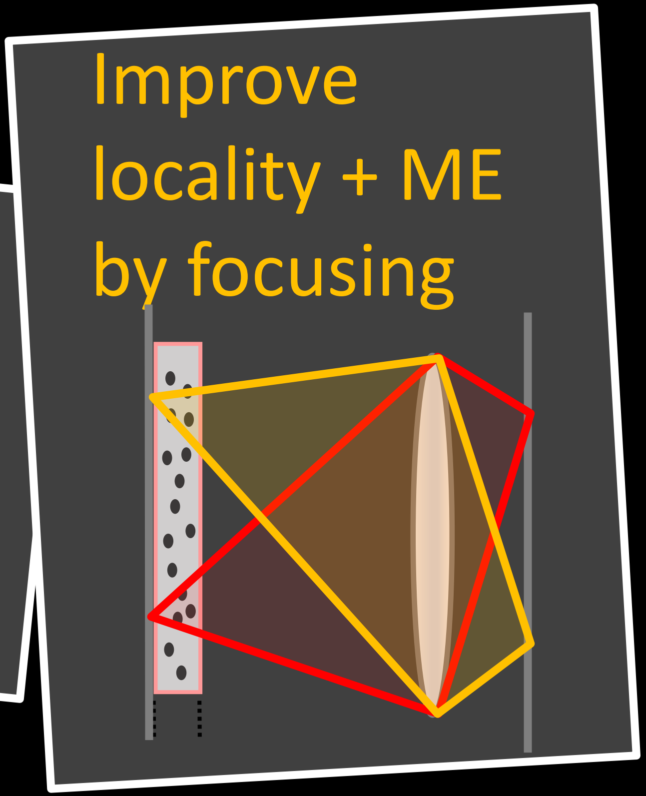


Local support is key for enhancement of seeing-through-scattering algorithms.

Summary of theoretical contributions

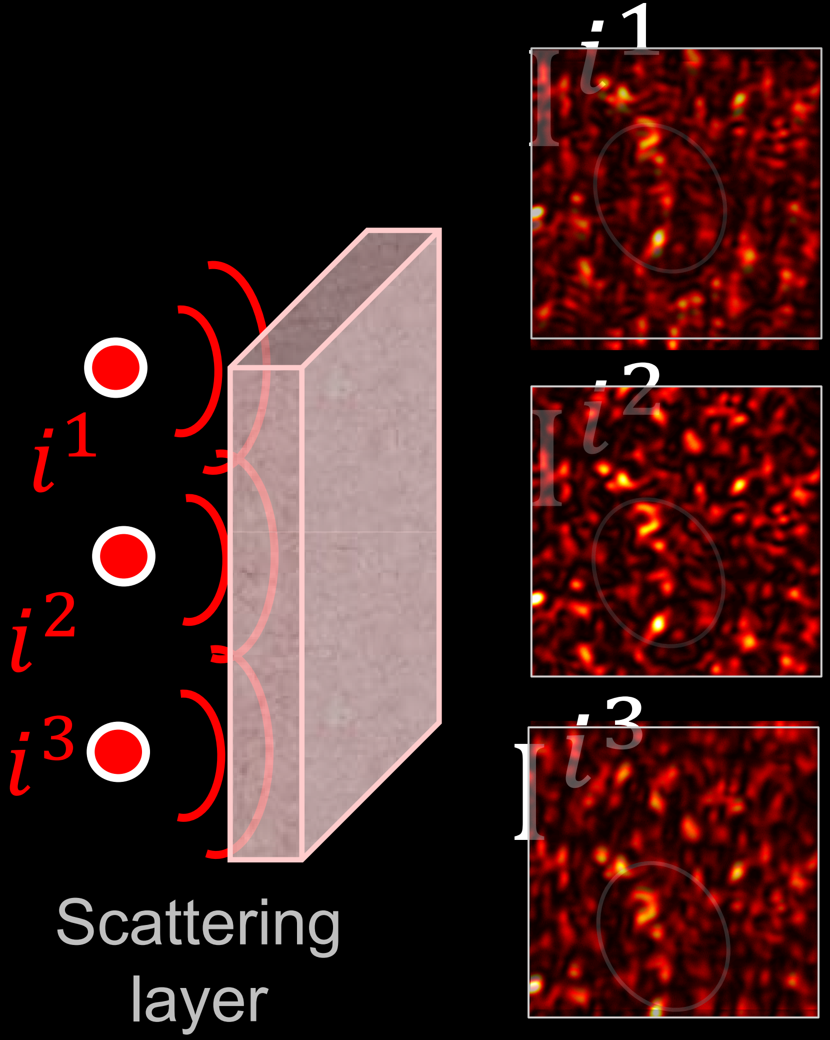
(i) Theoretical and rendering based analysis of ME in **near** and **far**-field settings





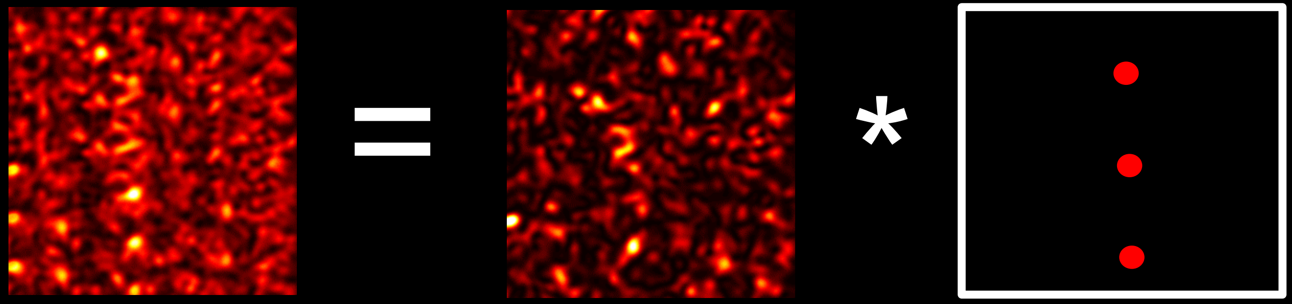
Our algorithm

Seeing though scattering using ME



Scatter free image

$$I = \sum I^{i^k} \qquad S \qquad O$$



$$= \qquad * $$

$$I \star I = S \star S \star O \star O \approx O \star O$$

Overlay

$$S \star S \approx \delta$$

Full-frame

$$\boxed{O = \text{Phase Retrieval}(I \star I)}$$

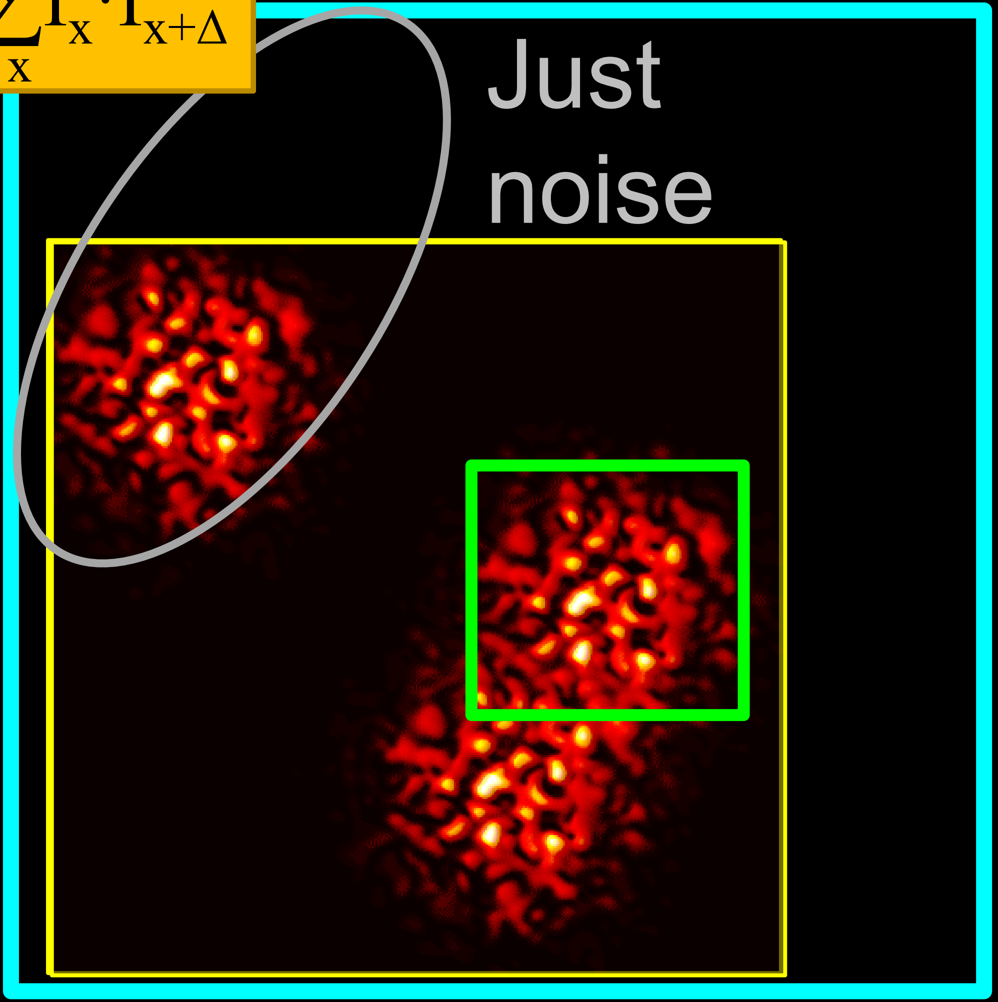
Reconstruction using local speckle correlation

$$\text{Correlation}(\Delta) = \sum_{x} I_x \cdot I_{x+\Delta}$$

What illuminators produce this image?

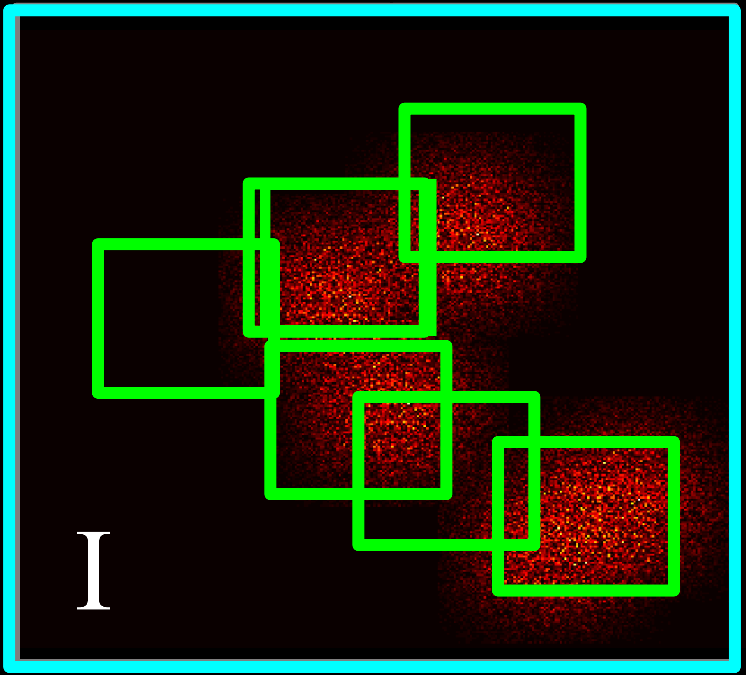
Previously: sum over full frame ⇨ **Noisy**

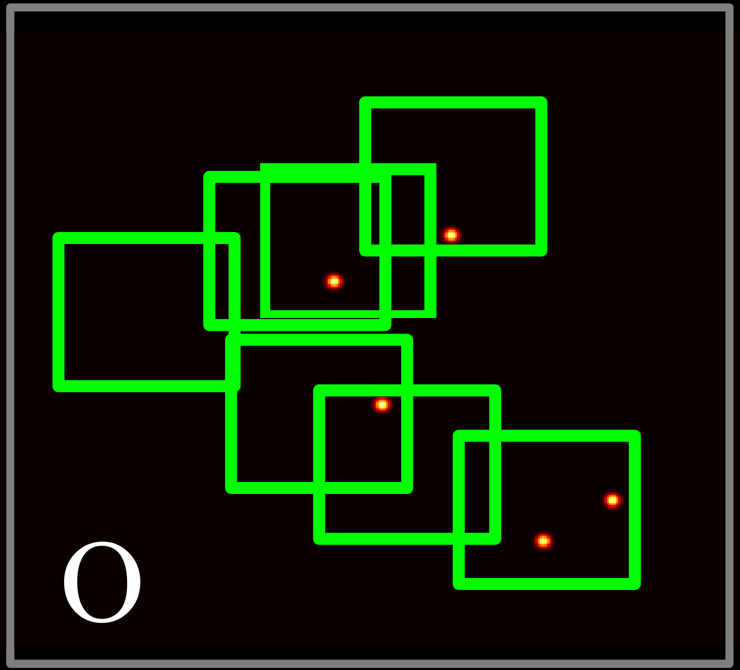
Ours: sum over local window ⇨ **Theorem: order of magnitude increase in SNR**



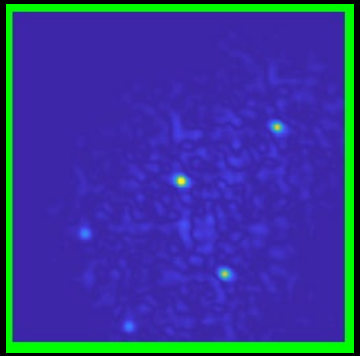
Optimization with local support

Full-frame algorithm

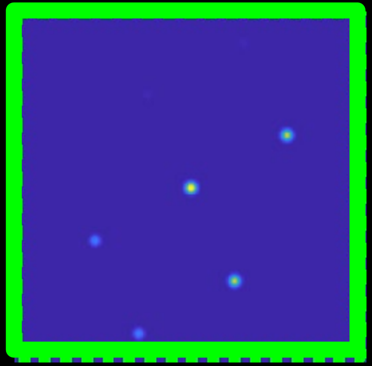




$$\min_{O} \sum \Big\| \Big($$



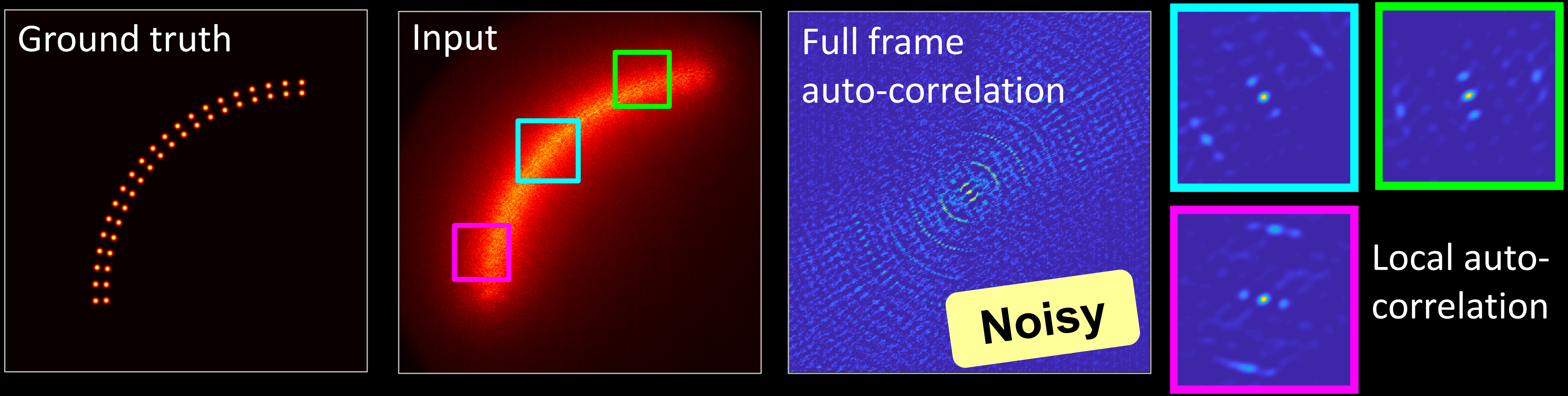
$$\Big) - \Big($$



$$\Big) \Big\|^2$$

All local windows

Local vs. global correlation



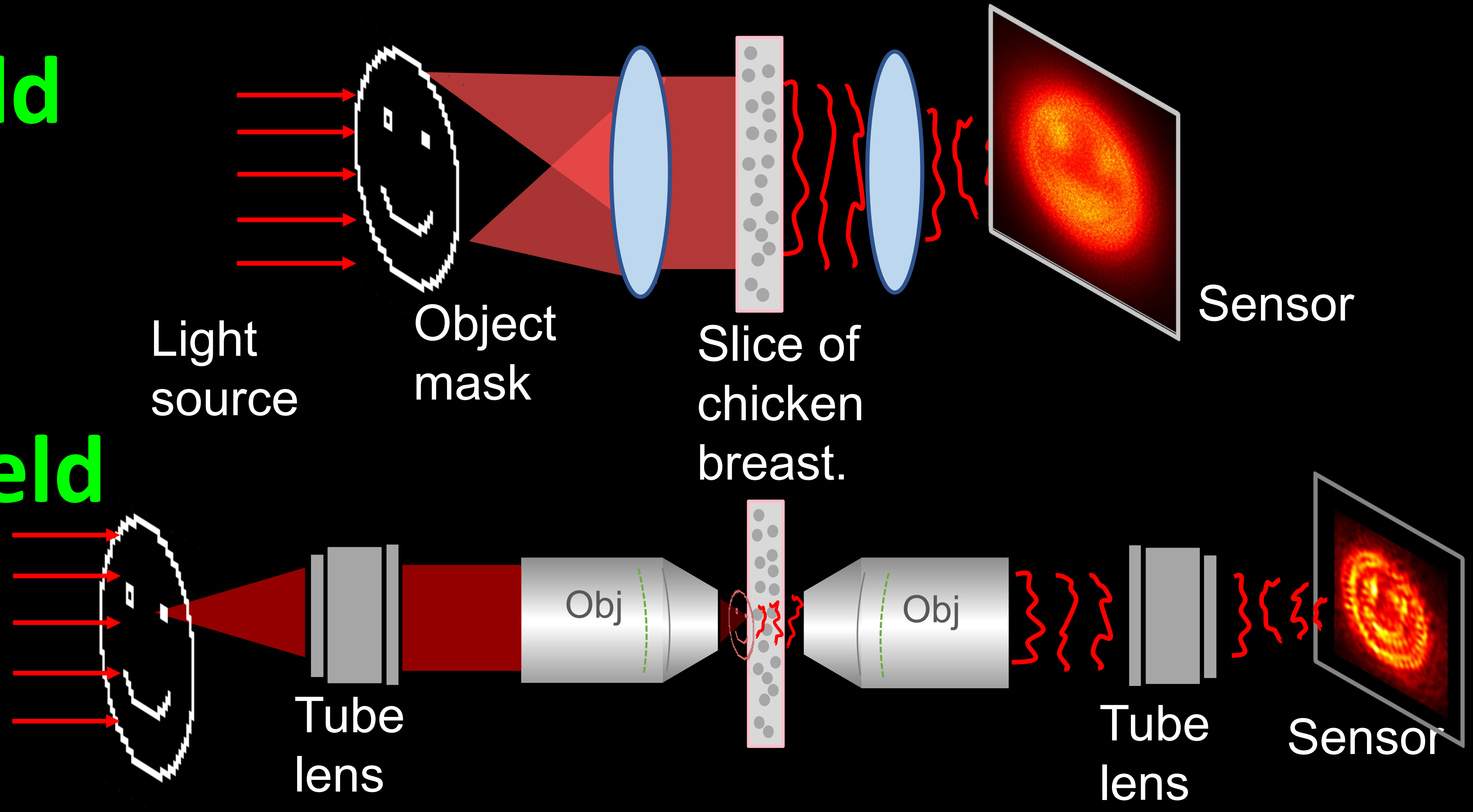
Local correlation: **less noisy and reveals some information about the desired shape**

Lab Results

Lab setup

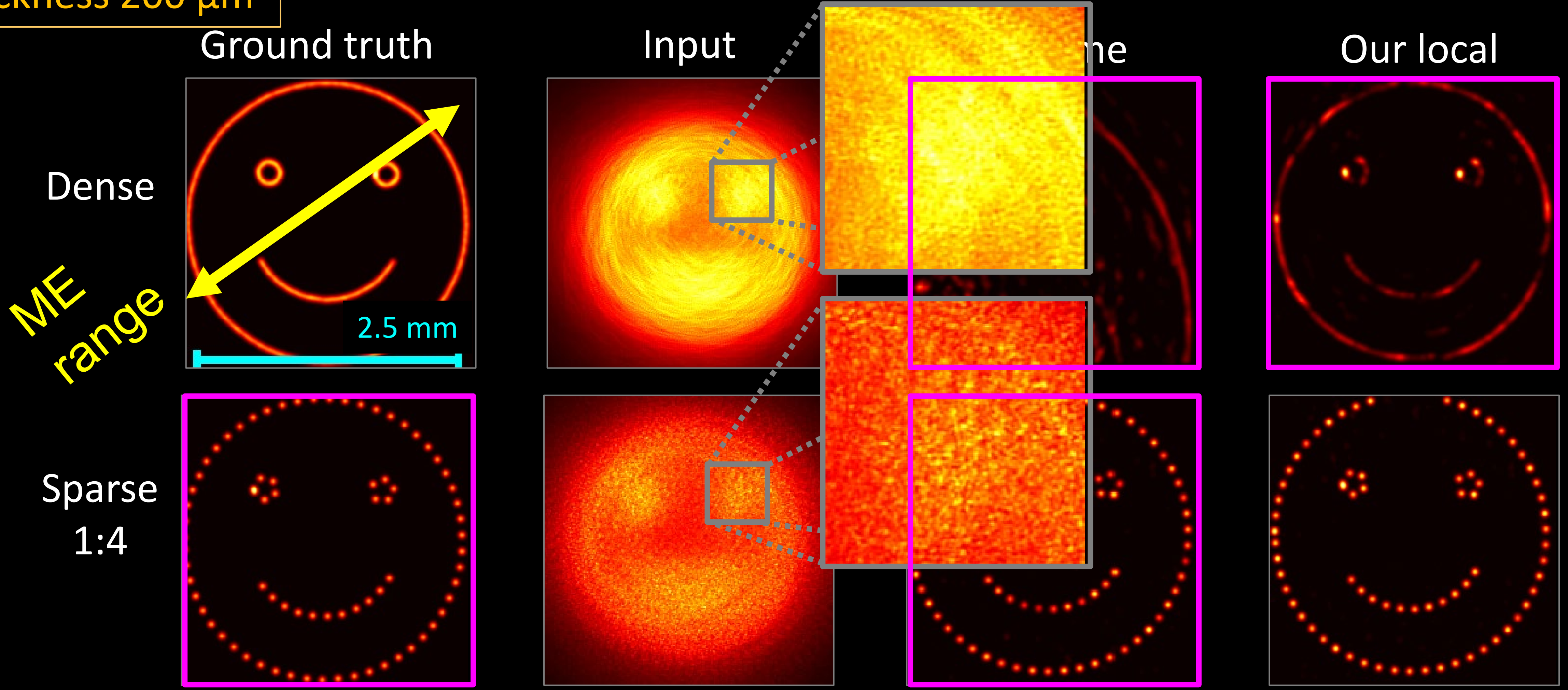
Far-field

Near-field



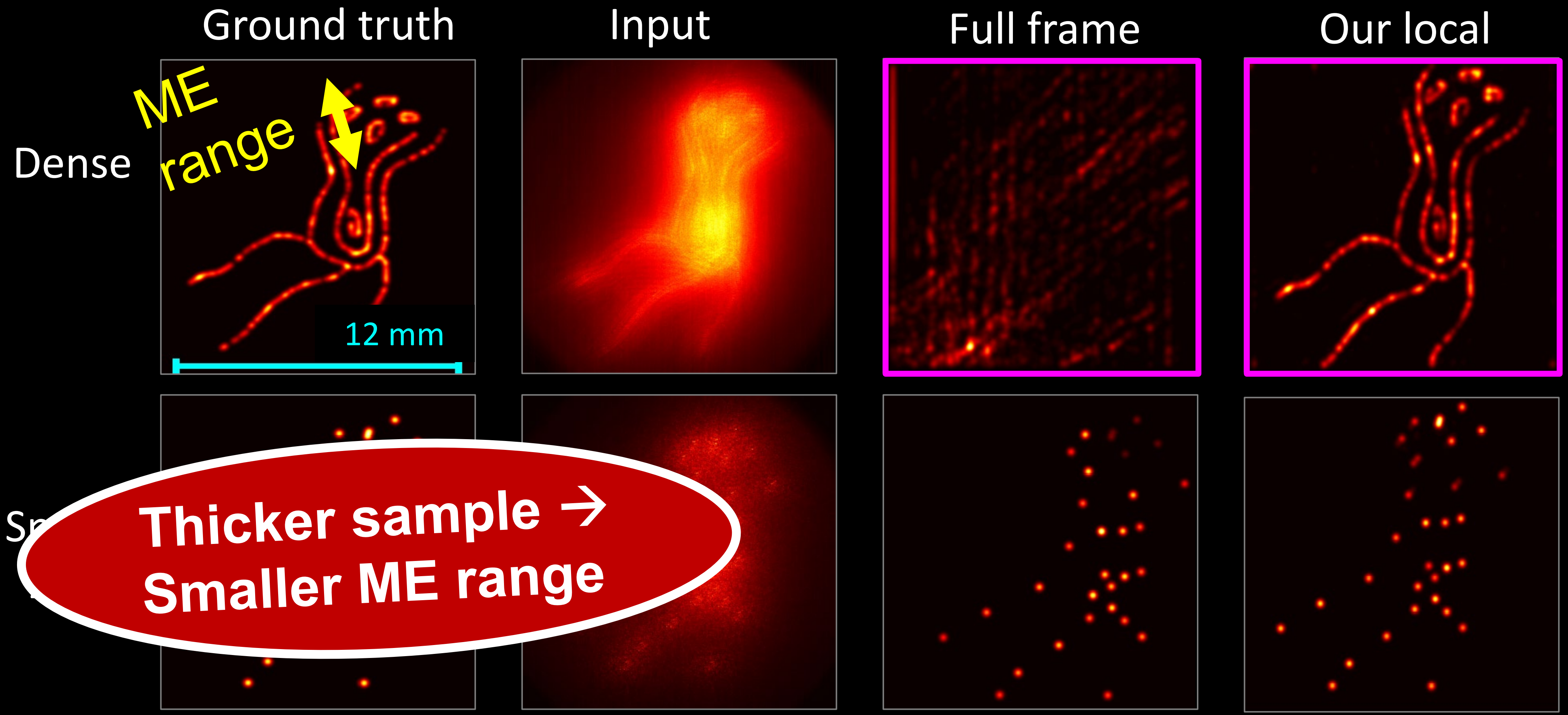
Lab results: far-field

Chicken breast
thickness 200 µm

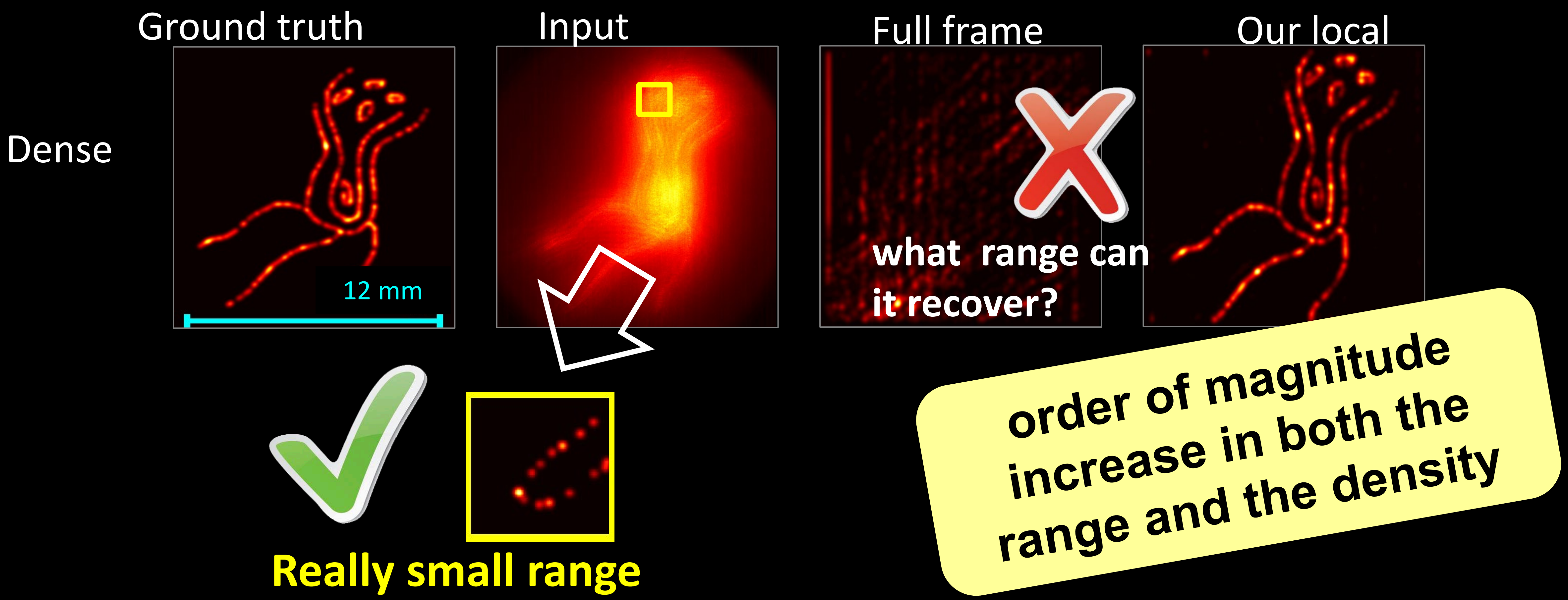


Chicken breast
thickness 340 µm

Lab results: far-field

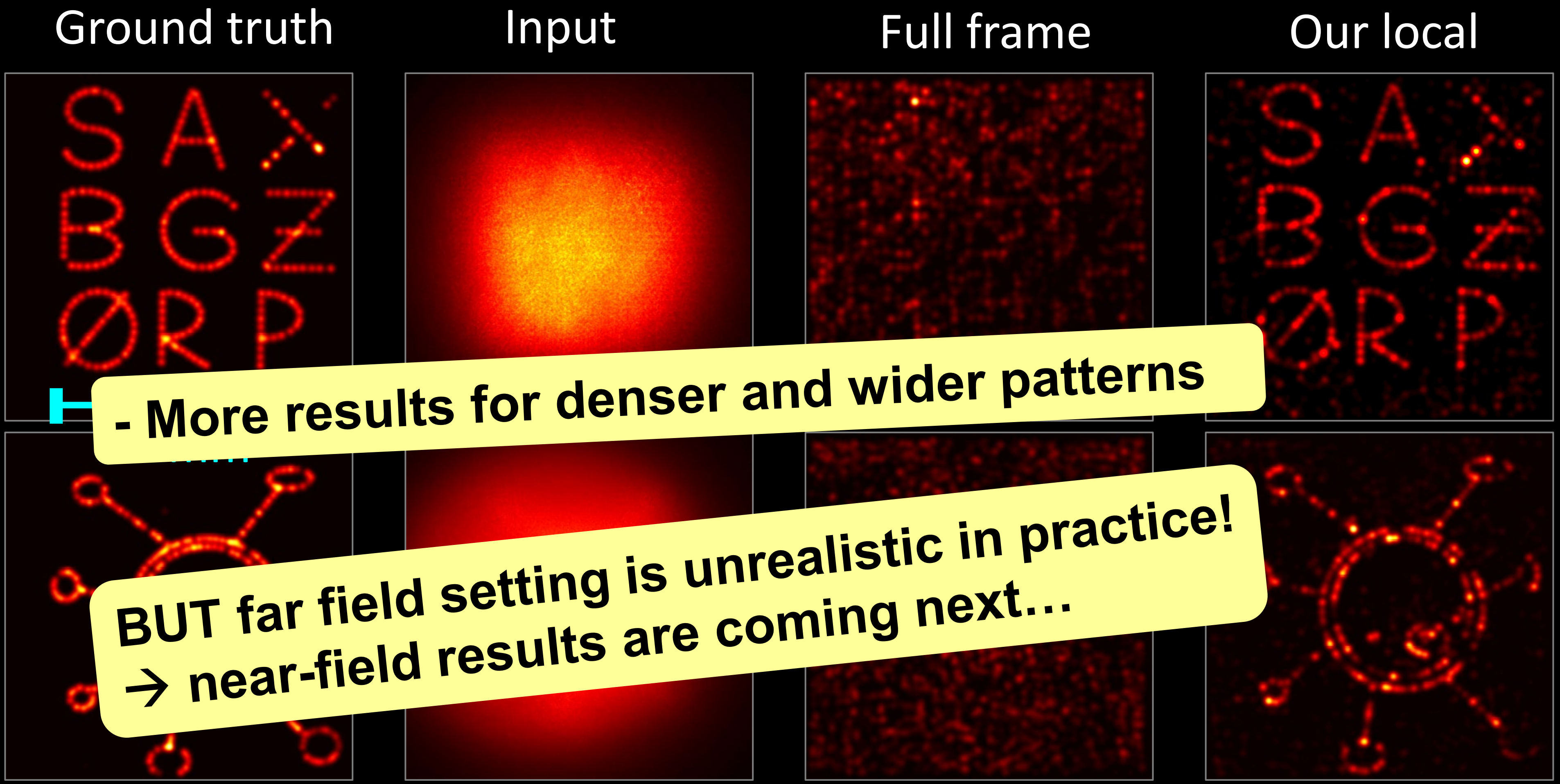


Lab results: far-field



Chicken breast
thickness 170 µm

Lab results: far-field

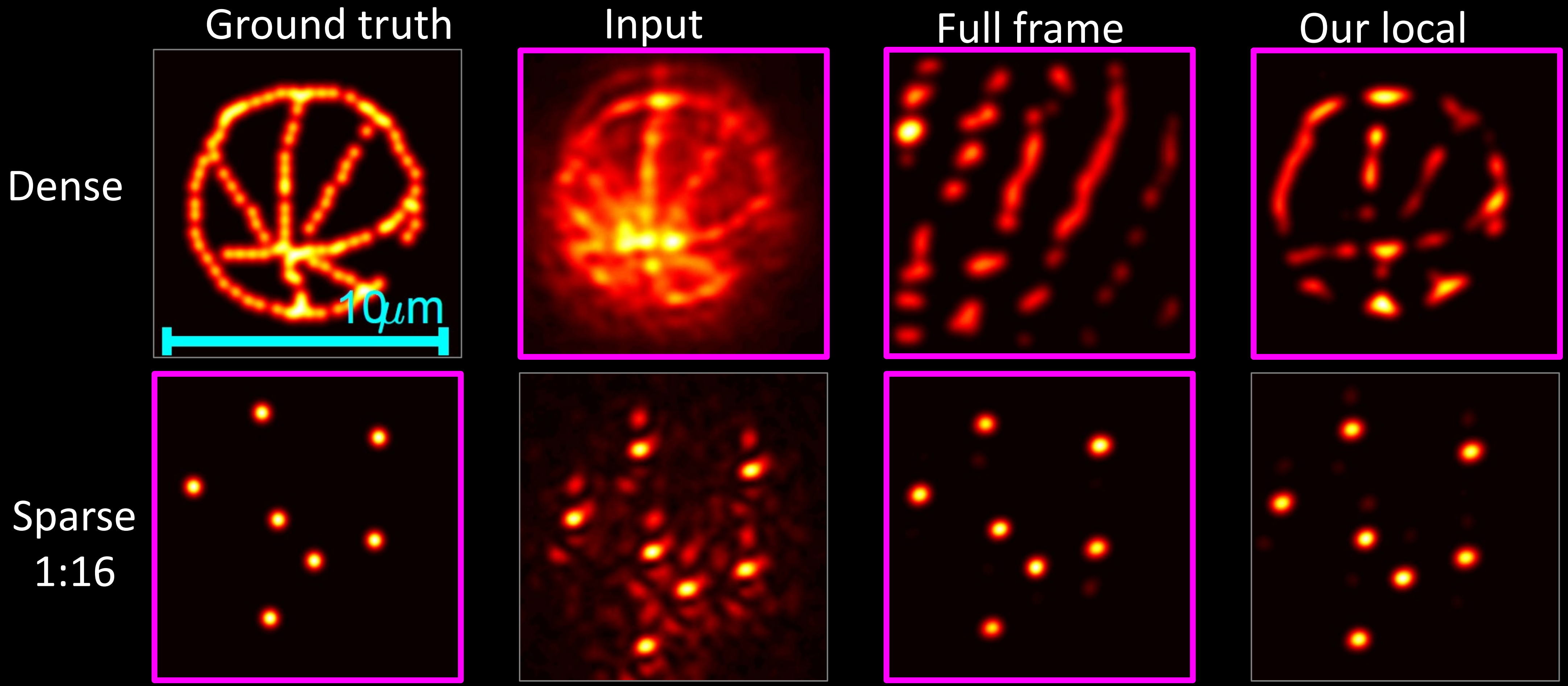


- More results for denser and wider patterns

BUT far field setting is unrealistic in practice!
→ near-field results are coming next…

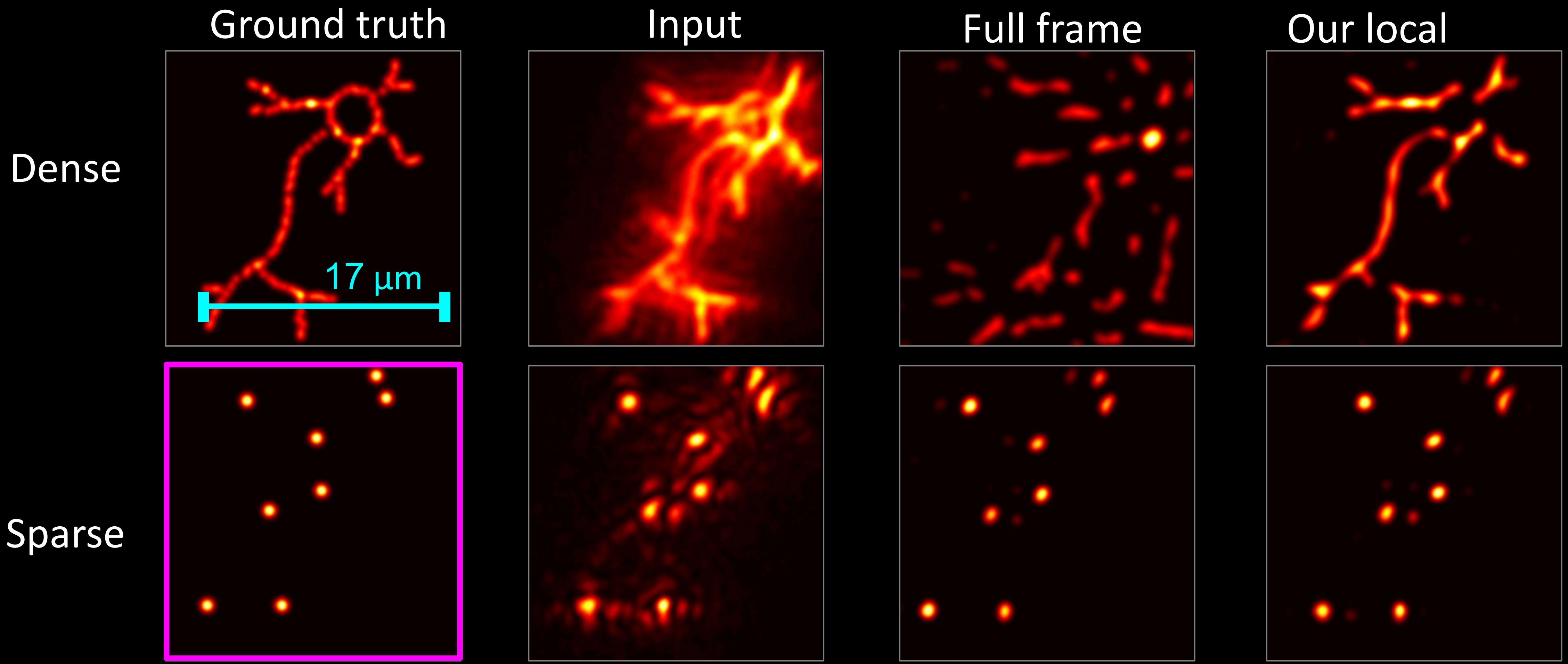
Chicken breast
thickness 140 µm

Lab results: near-field



Chicken breast
thickness 140 µm

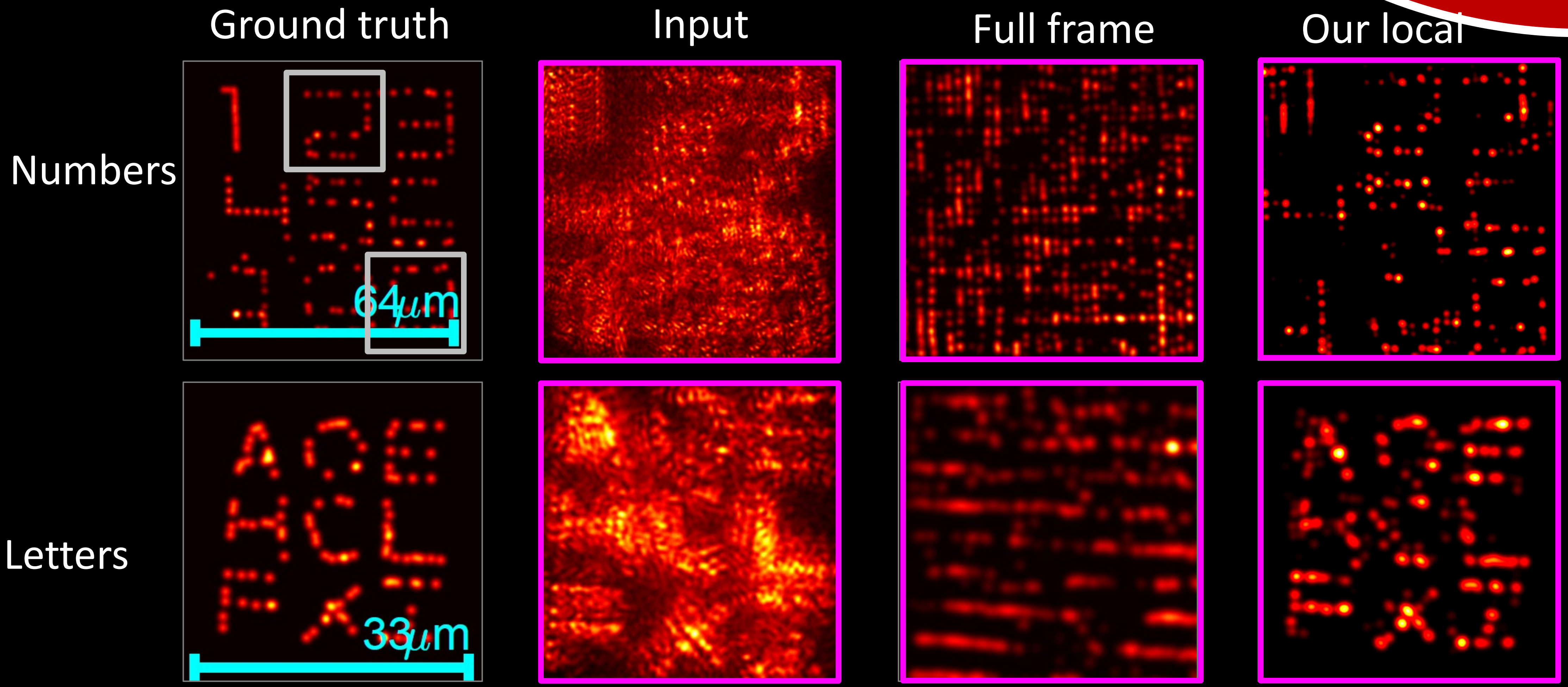
Lab results: near-field



Chicken breast
thickness 200 μm

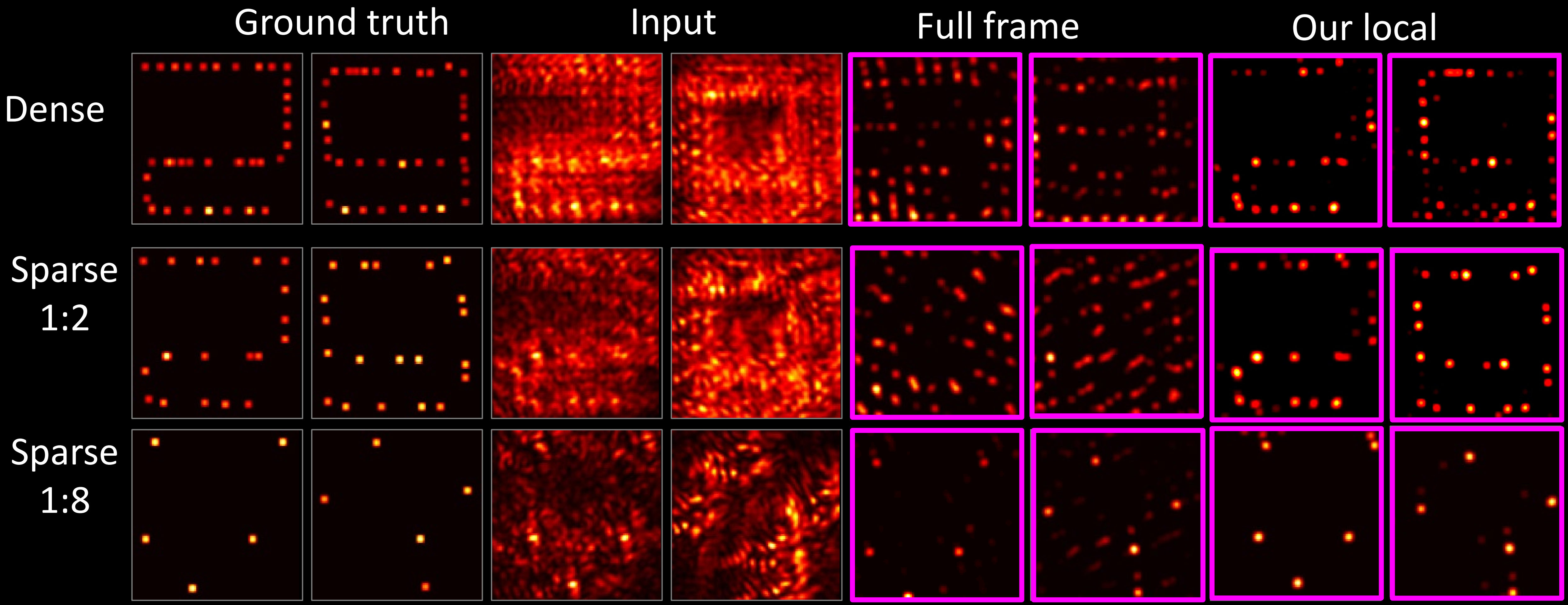
Lab results: near-field

Larger degradation



Chicken breast
thickness 200 µm

Lab results: near-field

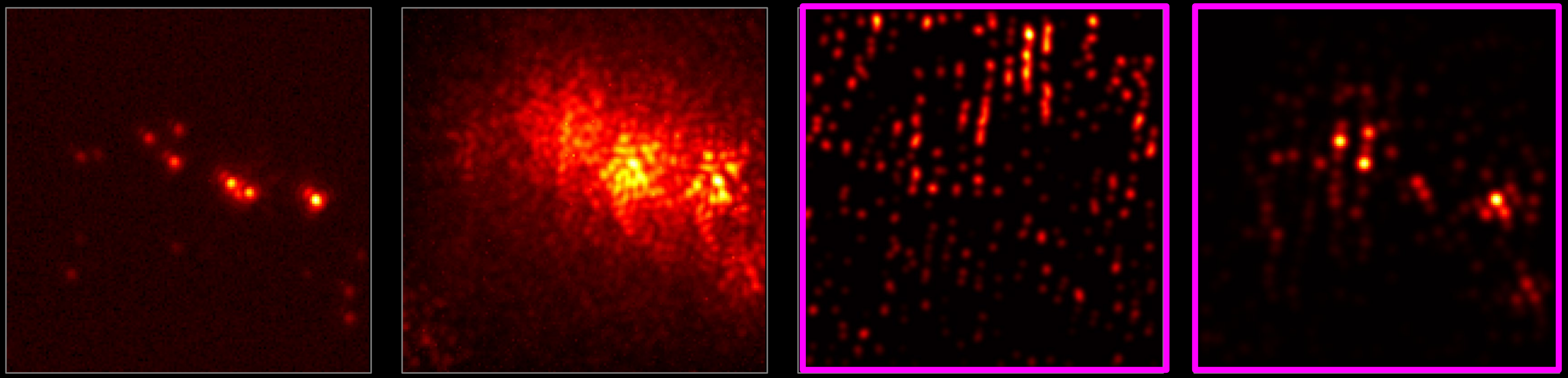


Chicken breast
thickness 100 µm

Lab results: near-field

Ground truth | Input | Full frame | Our local

Beads



Reconstructing
fluorescent beads

Thanks to **Dr. Lucien Weiss** from Schechtman lab
for preparing the fluorescent samples