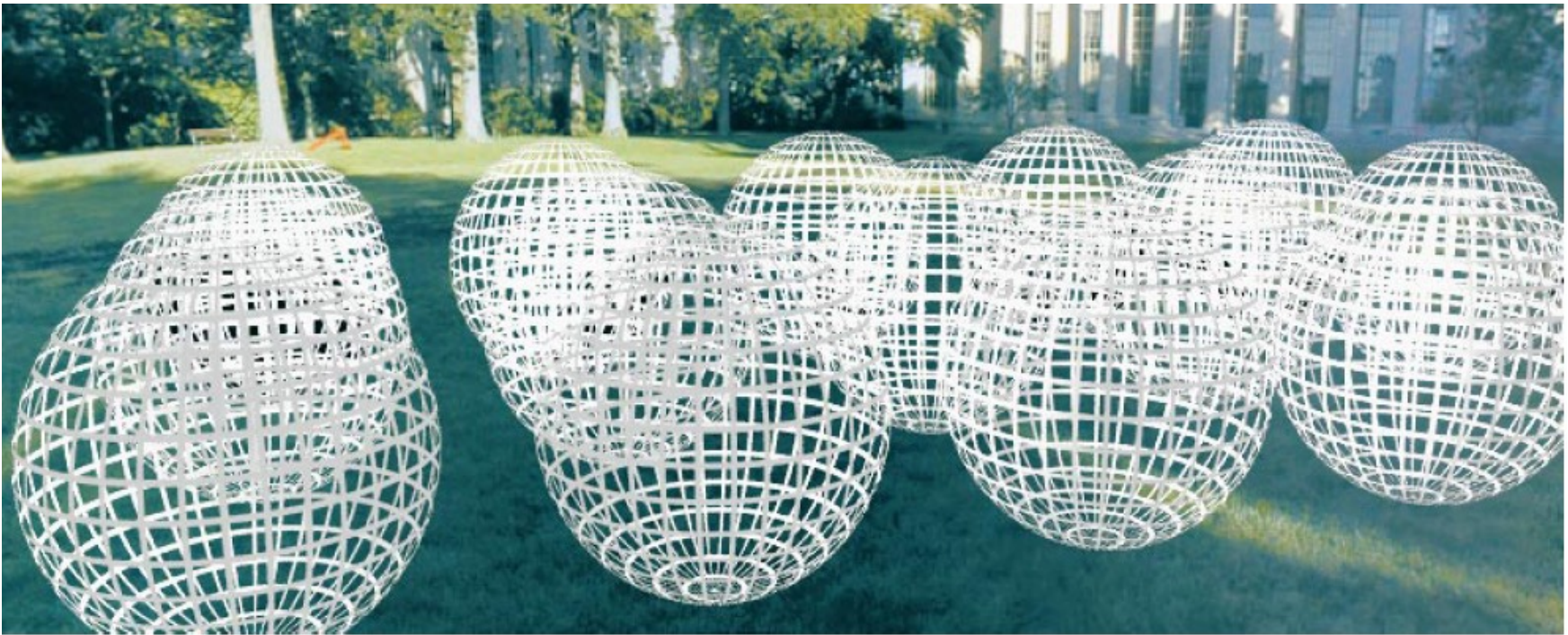


Modeling the plenoptic function



Adopted from: CS194: Intro to Comp. Vision, and Comp. Photo
Alexei Efros & Angjoo Kanazawa, UC Berkeley

Outline

- The plenoptic function
- Two-plane light fields
- Plenoptic camera
- Neural methods: NeRFs, 3D Gaussian splatting (next time)

Goal: Novel view rendering

- Given several images of the same object or scene from known viewpoints, how can we generate a rendering of the same scene from a novel viewpoint?
- Multiview stereo answer: create a textured 3D model from the images, use traditional graphics to render

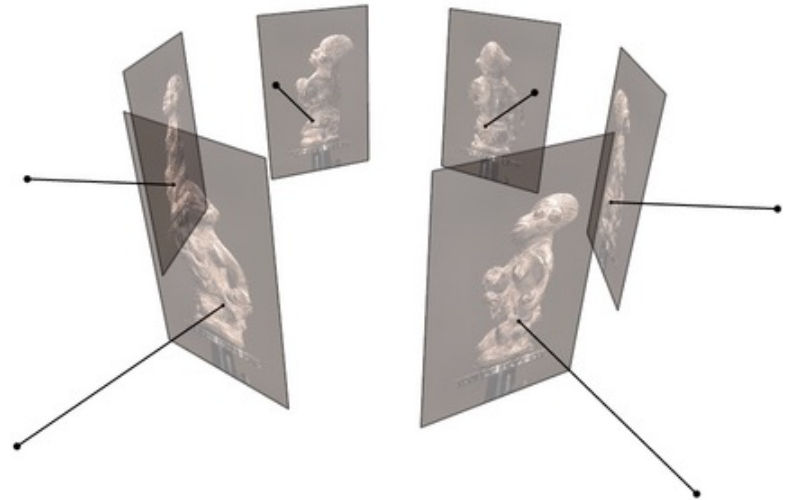
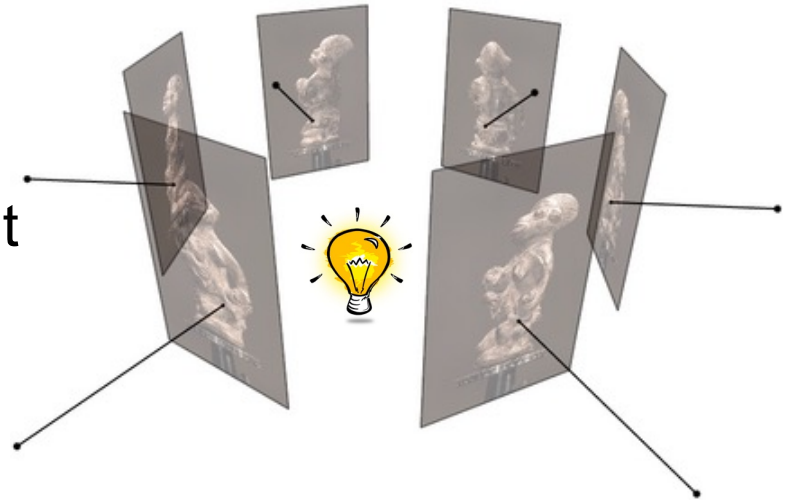


Figure source: C. Hernandez, N. Snavely

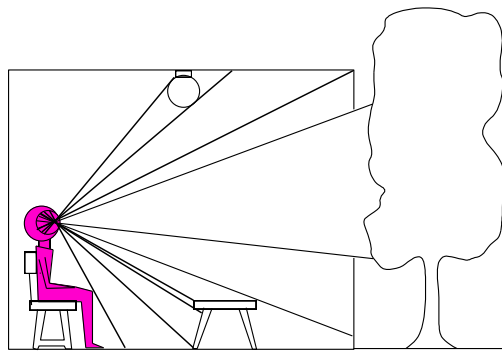
Goal: Novel view rendering

- Given several images of the same object or scene from known viewpoints, how can we generate a rendering of the same scene from a novel viewpoint?
- Multiview stereo answer: create a textured 3D model from the images, use traditional graphics to render
- Alternate answer: model the *light field* of the scene, sample new views from it



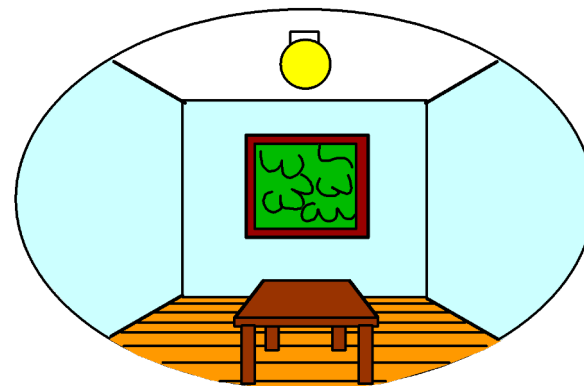
Modeling the light field

3D world



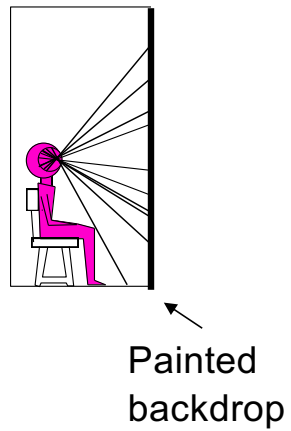
Point of observation

2D image

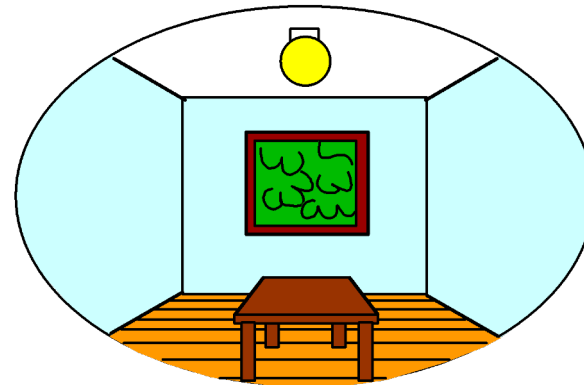


Modeling the light field

3D world



2D image



The light field, or plenoptic function



Figure by Leonard McMillan

Q: What is the set of all things that we can ever see?

A: The *plenoptic function*

E. Adelson and J. Bergen. [The plenoptic function and the elements of early vision.](#)
Computational models of visual processing, MIT Press, 1991

The light field, or plenoptic function



Figure by Leonard McMillan

Q: What is the set of all things that we can ever see?

A: The *plenoptic function*

Let's start with a stationary person and try to parameterize everything that they can see...

Color snapshot



$$L(\theta, \phi, \lambda)$$

- Intensity of light
 - Seen from a single view point
 - At a single time
 - As a function of wavelength

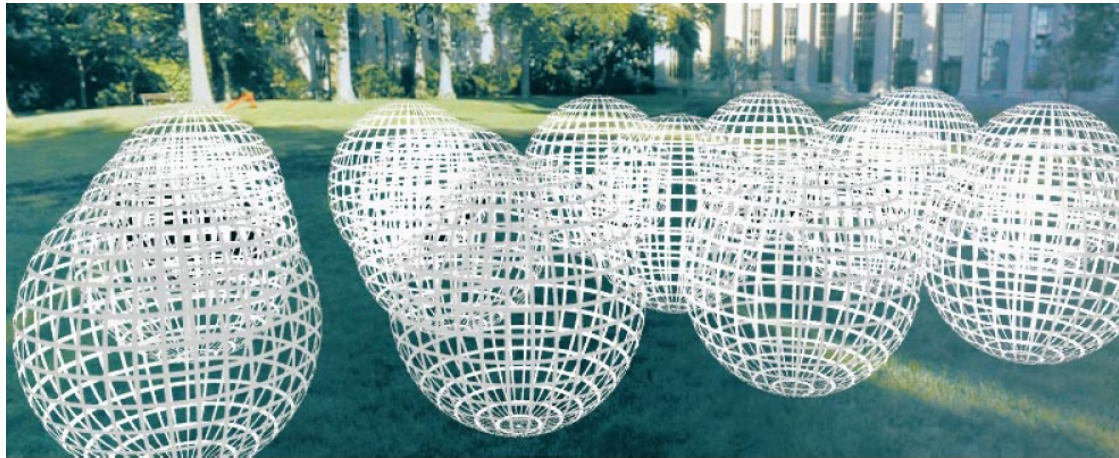
A movie



$$L(\theta, \phi, \lambda, t)$$

- Intensity of light
 - Seen from a single view point
 - Over time
 - As a function of wavelength

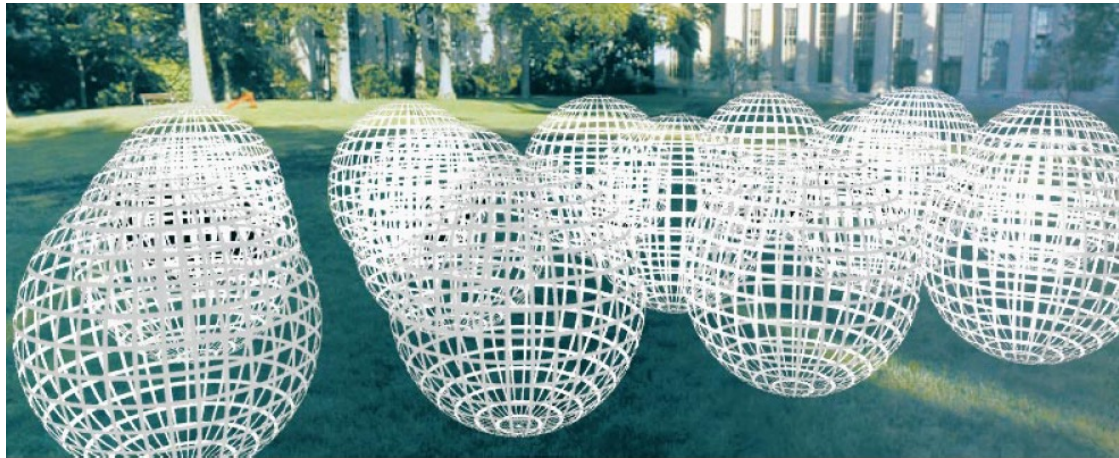
Holographic movie



$$L(\theta, \phi, \lambda, t, x, y, z)$$

- Intensity of light
 - Seen from ANY viewpoint
 - Over time
 - As a function of wavelength

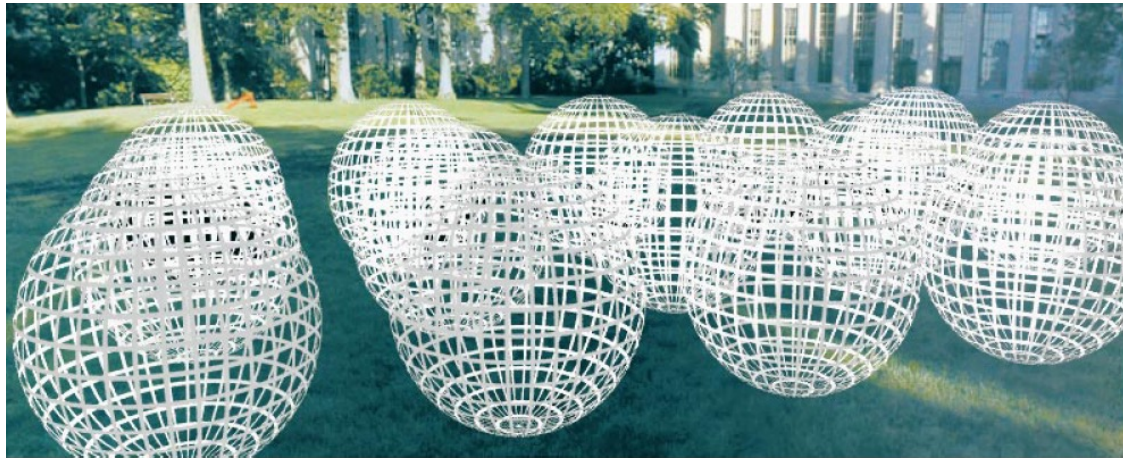
The plenoptic function



$$L(\theta, \phi, \lambda, t, x, y, z)$$

- Can reconstruct every possible view, at every moment, from every position, at every wavelength
- Contains every photograph, every movie, everything that anyone has ever seen! it completely captures our visual reality!
- Not bad for a function...

The plenoptic function: More practical version



$$L(\theta, \phi, x, y, z) = (r, g, b)$$

- Other simplifications/variants are possible, as we will see

Modeling the plenoptic function

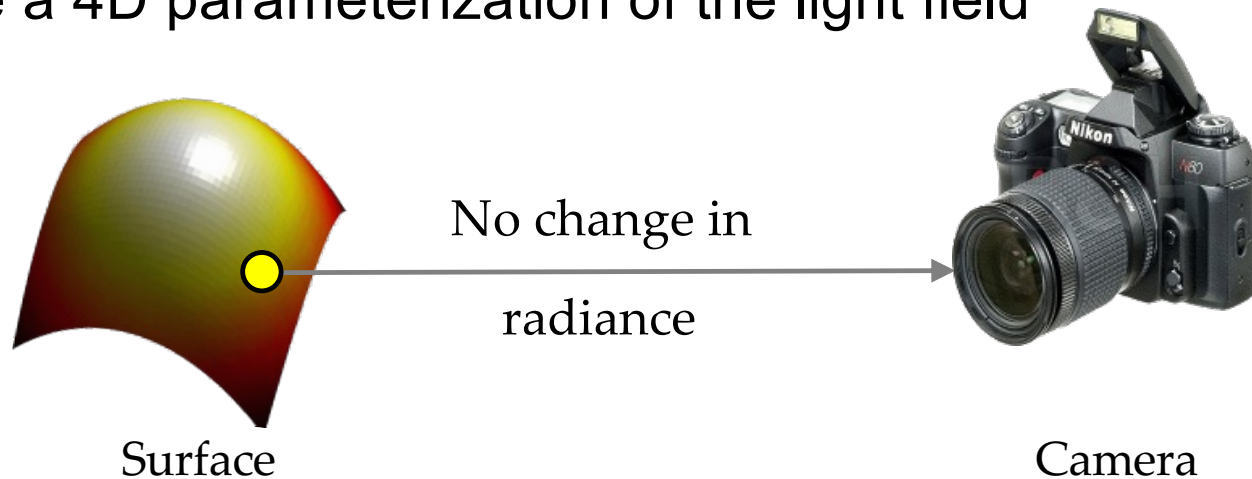
- Capture
 - Create a special camera setup to capture a slice of the plenoptic function
 - Combine captured rays for novel view synthesis, defocus, and other effects
- Fitting
 - Given a set of multi-view calibrated images, optimize a parametric representation of the plenoptic function of the scene

Outline

- The plenoptic function
- Two-plane light fields

Two-plane light fields

- Key idea: assuming light is constant along rays, we can create a 4D parameterization of the light field



If there is no occlusion or fog

S. Gortler, R. Grzeszczuk, S. Szaliski, M. Cohen. [The Lumigraph](#). Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, 1996

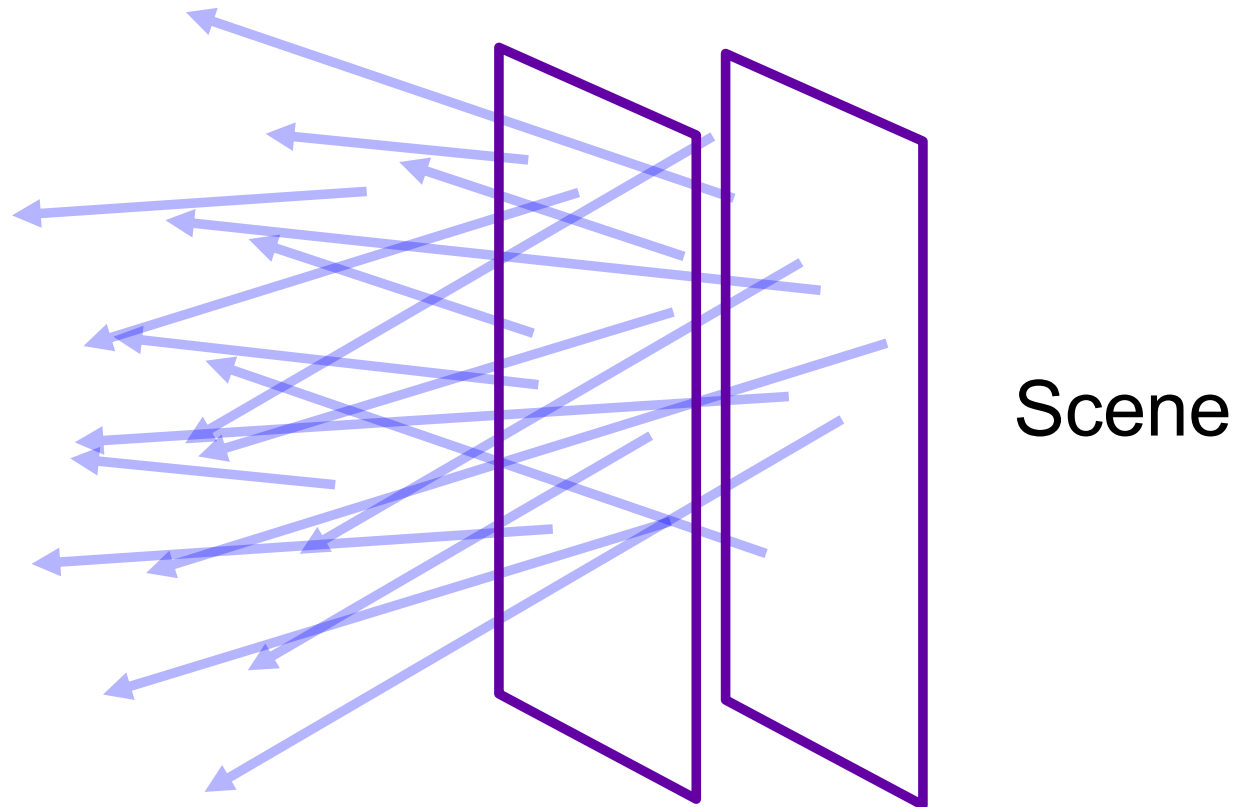
M. Levoy and P. Hanrahan. [Light field rendering](#). SIGGRAPH 1996

Two-plane light fields

- Two-plane parameterization:



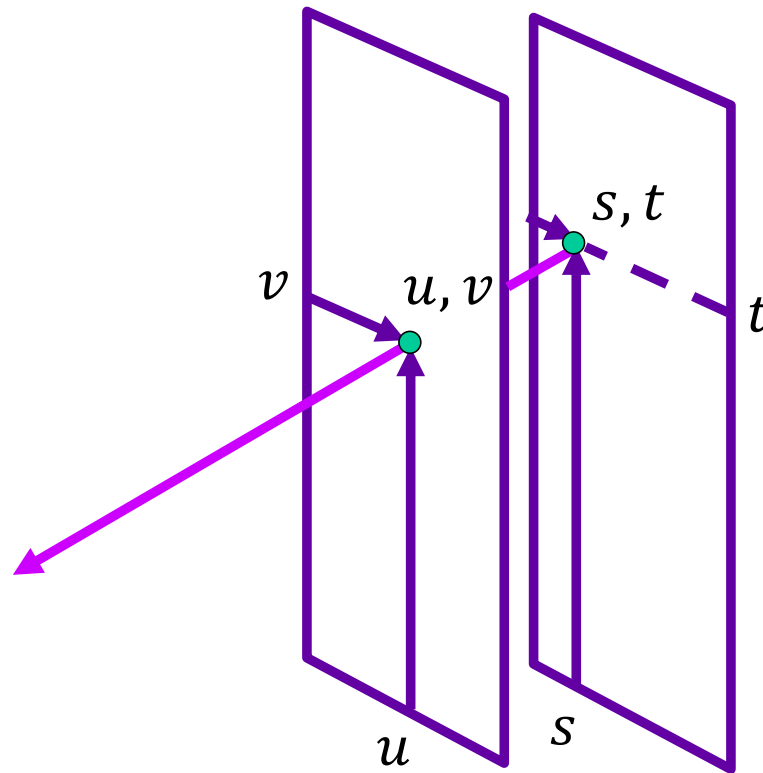
Observer



Two-plane light fields

- Two-plane parameterization:

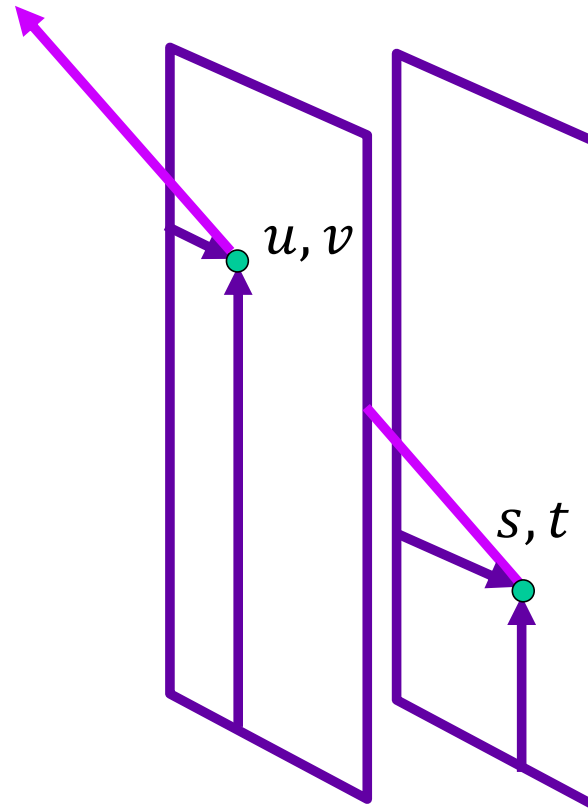
$$L(s, t, u, v) = (r, g, b)$$



Two-plane light fields

- Two-plane parameterization:

$$L(s, t, u, v) = (r, g, b)$$

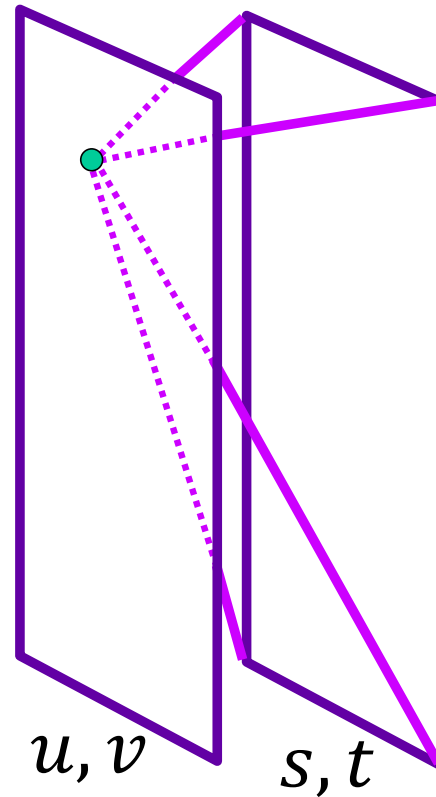


Two-plane light fields

- What do we get if we hold u, v constant and let s, t vary?
- An image!



Observer

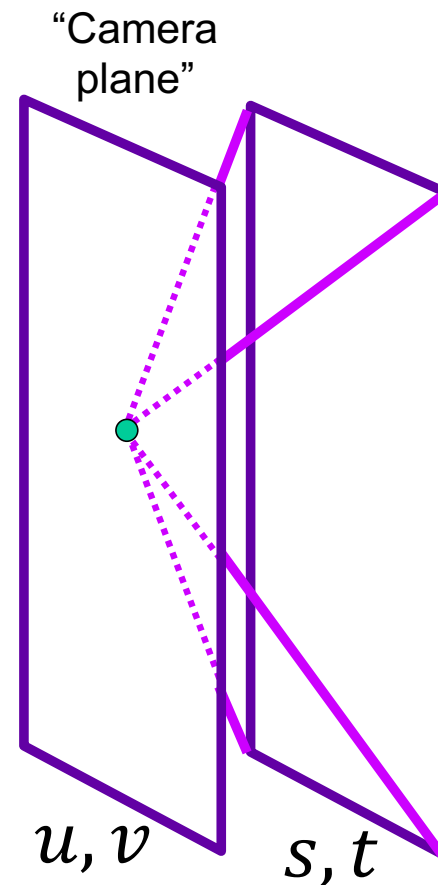


Two-plane light fields

- What do we get if we hold u, v constant and let s, t vary?
- An image!



Observer

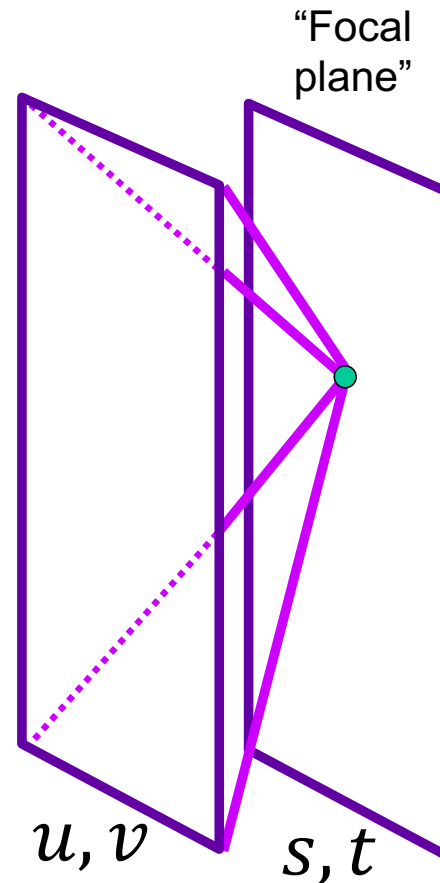


Two-plane light fields

- What do we get if we hold s, t constant and let u, v vary?
- A set of rays leaving a point in the scene in a bundle of directions towards the observer



Observer



Light field visualization

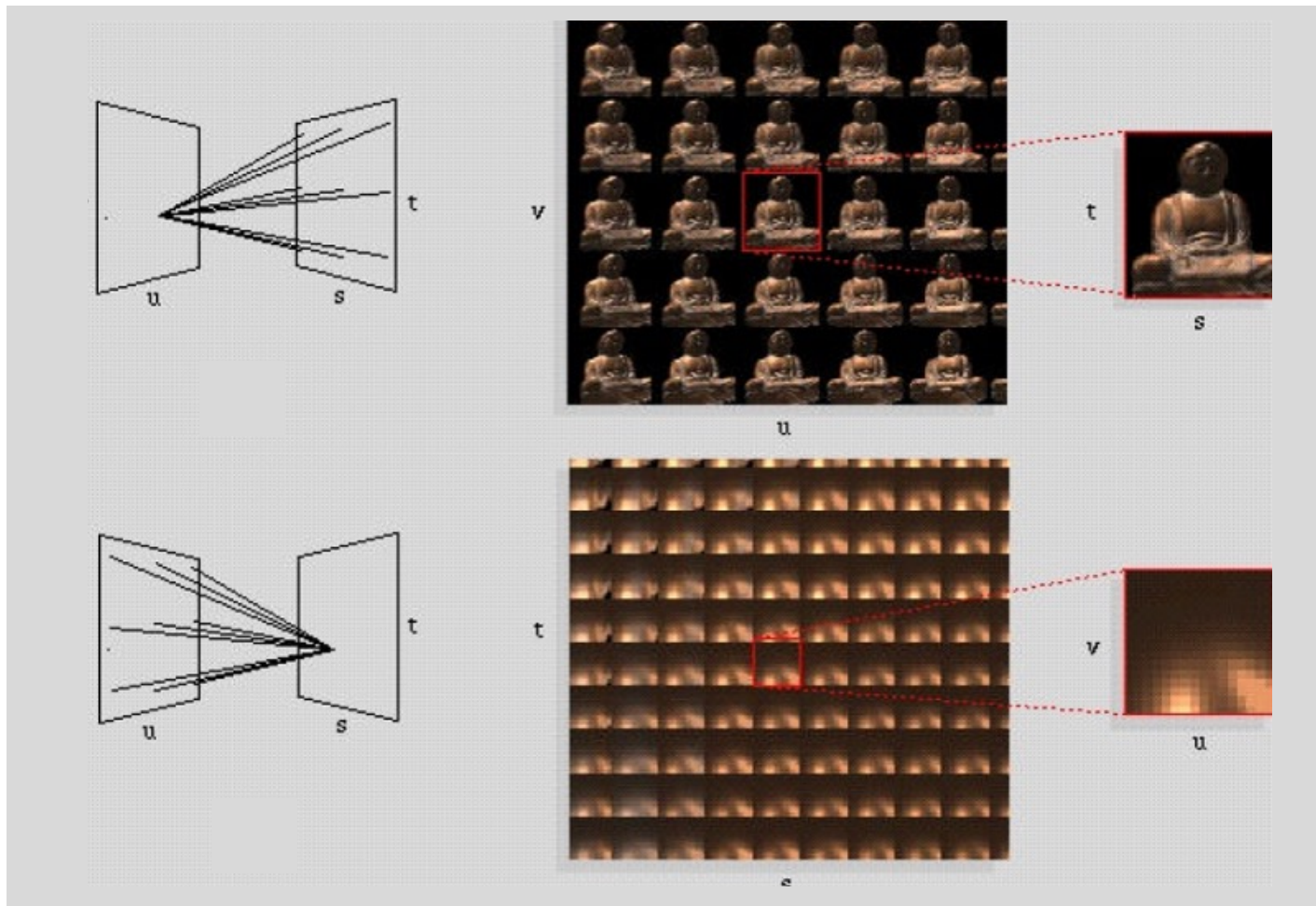
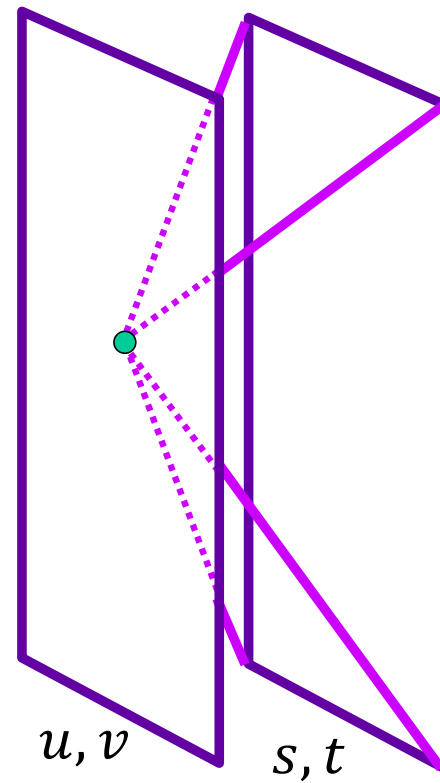
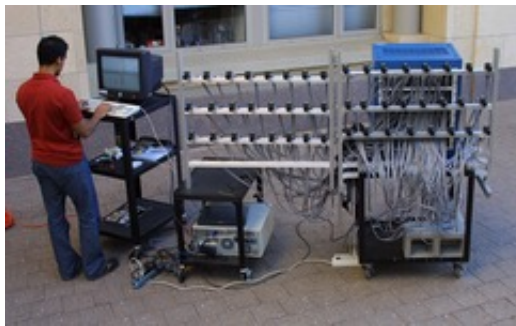


Figure source: M. Levoy and P. Hanrahan

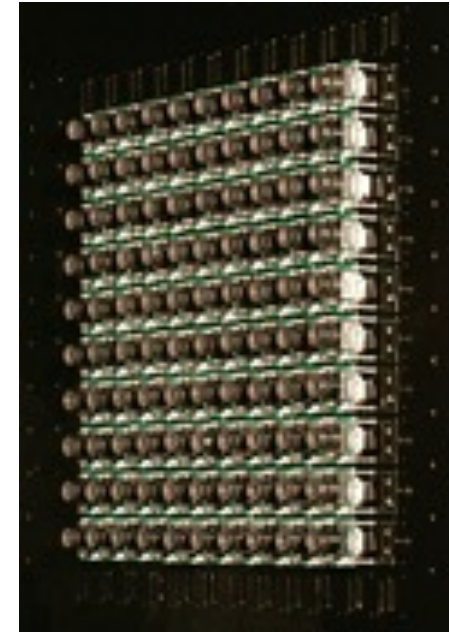
Light field capture

- Idea 1: move camera carefully over u, v plane



Stanford multi-camera array

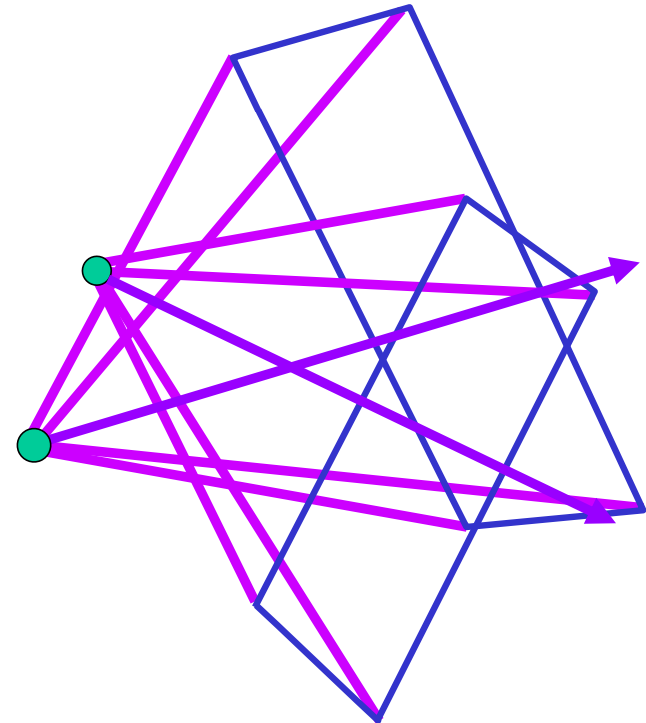
- 640 × 480 pixels ×
30 fps × 128 cameras
- Synchronized timing
- Continuous streaming
- Flexible arrangement



<http://graphics.stanford.edu/projects/array/>

Light field capture

- Idea 2: move camera anywhere, use rebinning or resampling



Light field capture

- Idea 2: move camera anywhere, use rebinning or resampling

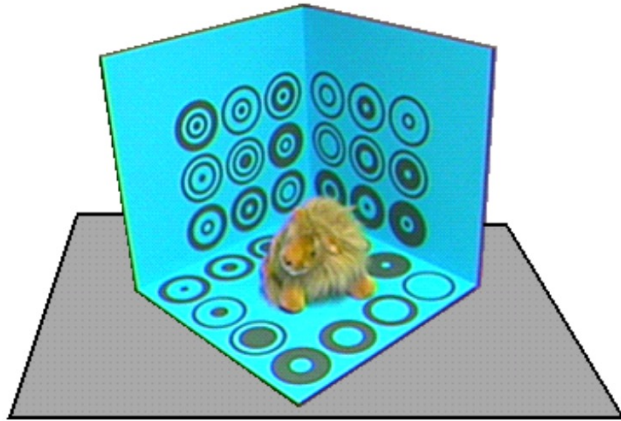


Figure 10: The capture stage

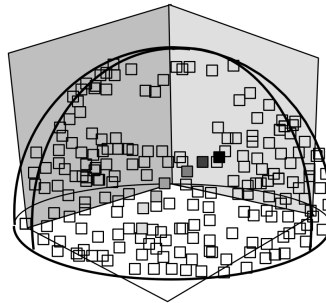
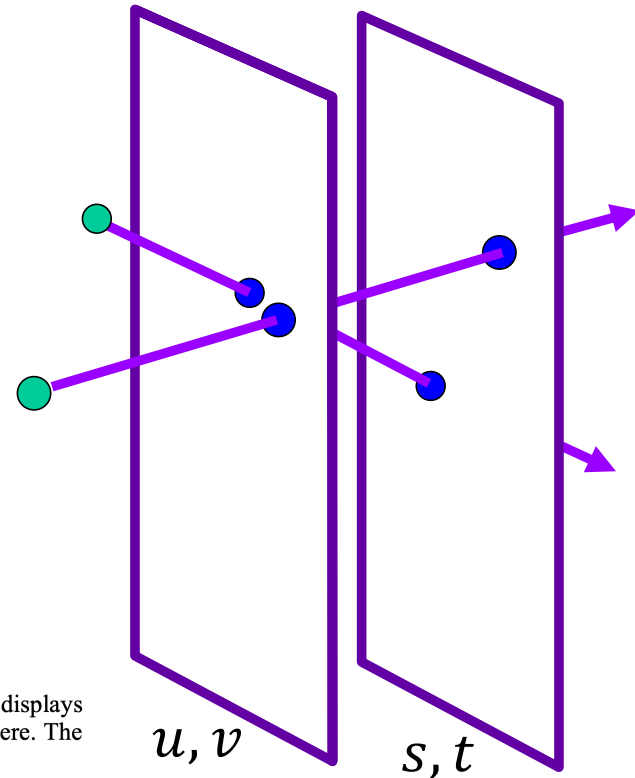
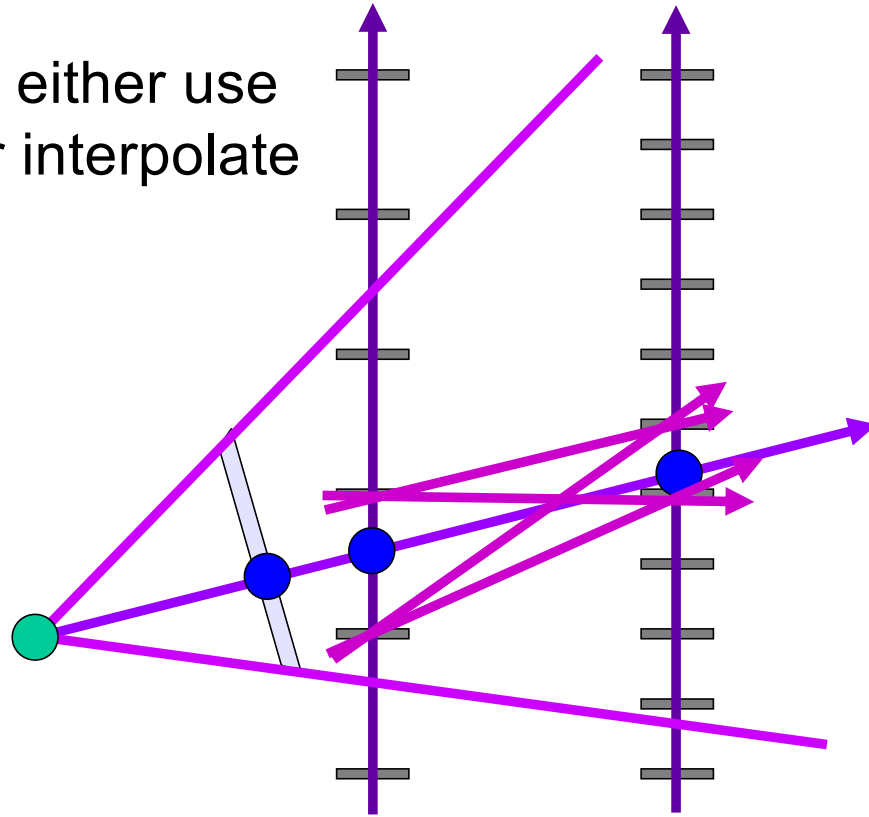


Figure 11: The user interface for the image capture stage displays the current and previous camera positions on a viewing sphere. The goal of the user is to “paint” the sphere.



Novel view synthesis

- For each output pixel, determine s, t, u, v , then either use closest discrete RGB or interpolate several nearby values



Outline

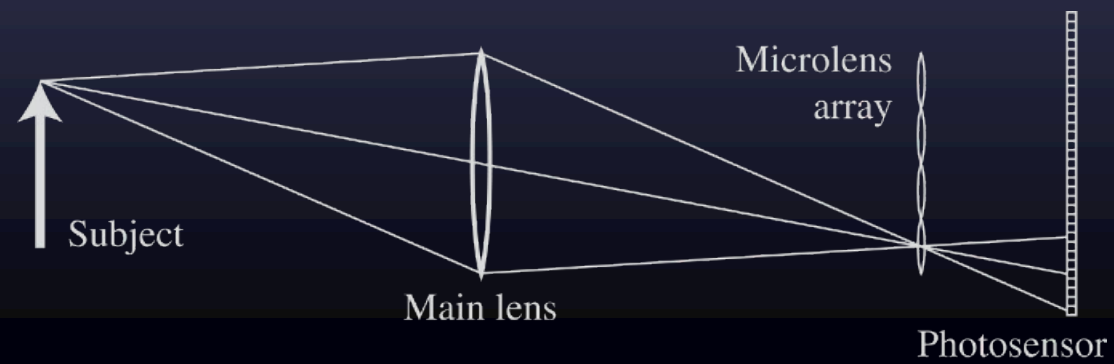
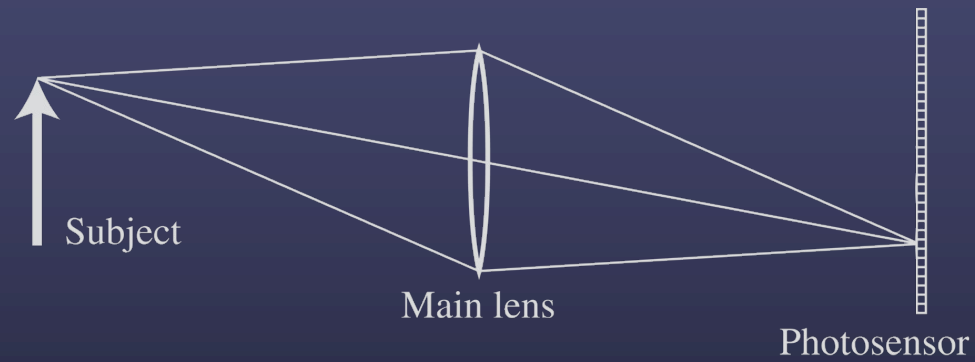
- The plenoptic function
- Two-plane light fields
- Plenoptic camera

Plenoptic camera

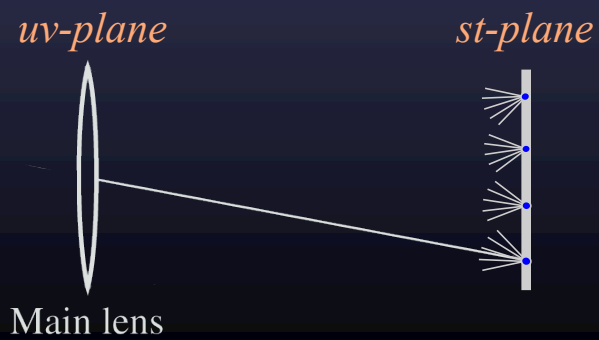
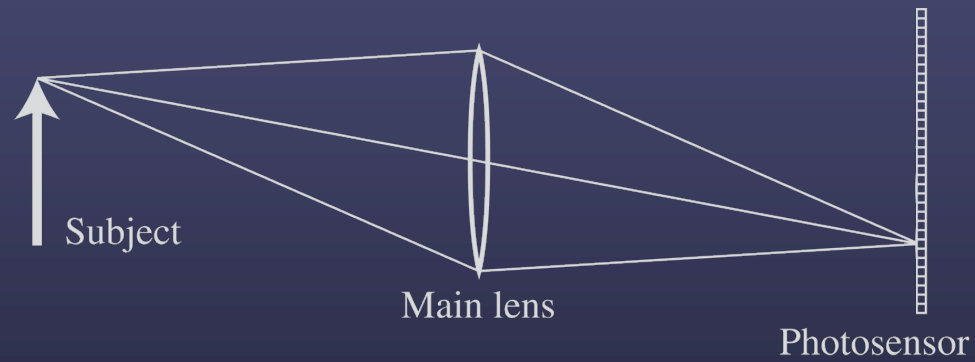


R. Ng et al. [Light Field Photography with a Hand-held Plenoptic Camera](#). 2005

Conventional vs. light field camera



Conventional vs. light field camera



Prototype camera



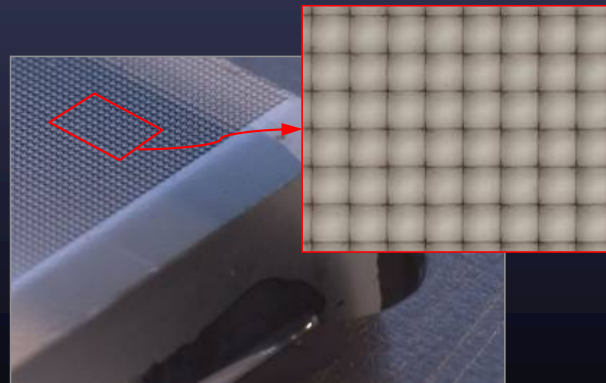
Contax medium format camera



Kodak 16-megapixel sensor



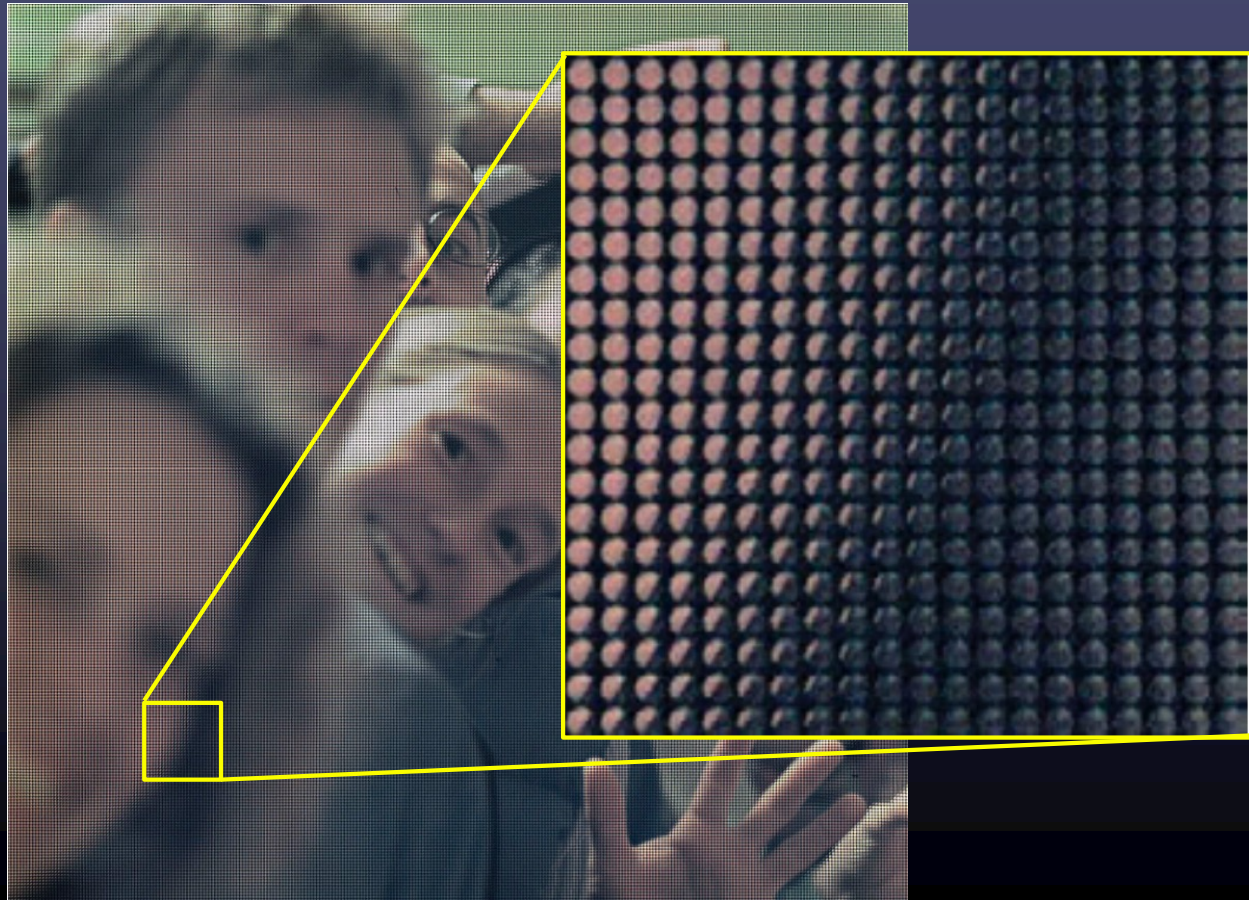
Adaptive Optics microlens array



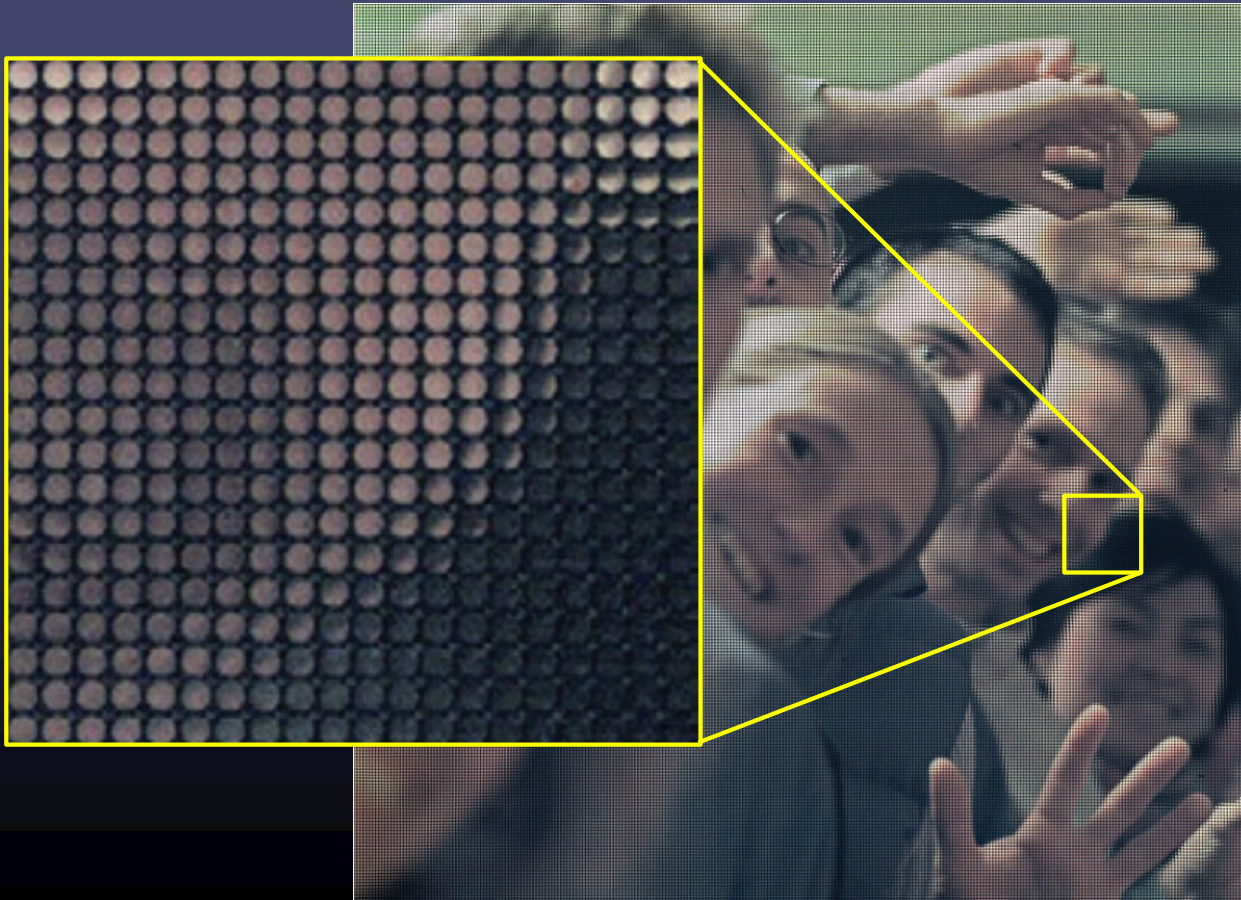
125 μ square-sided microlenses

$$\begin{aligned} &4000 \times 4000 \text{ pixels} / \\ &292 \times 292 \text{ lenses} = \\ &14 \times 14 \text{ pixels per lens} \end{aligned}$$

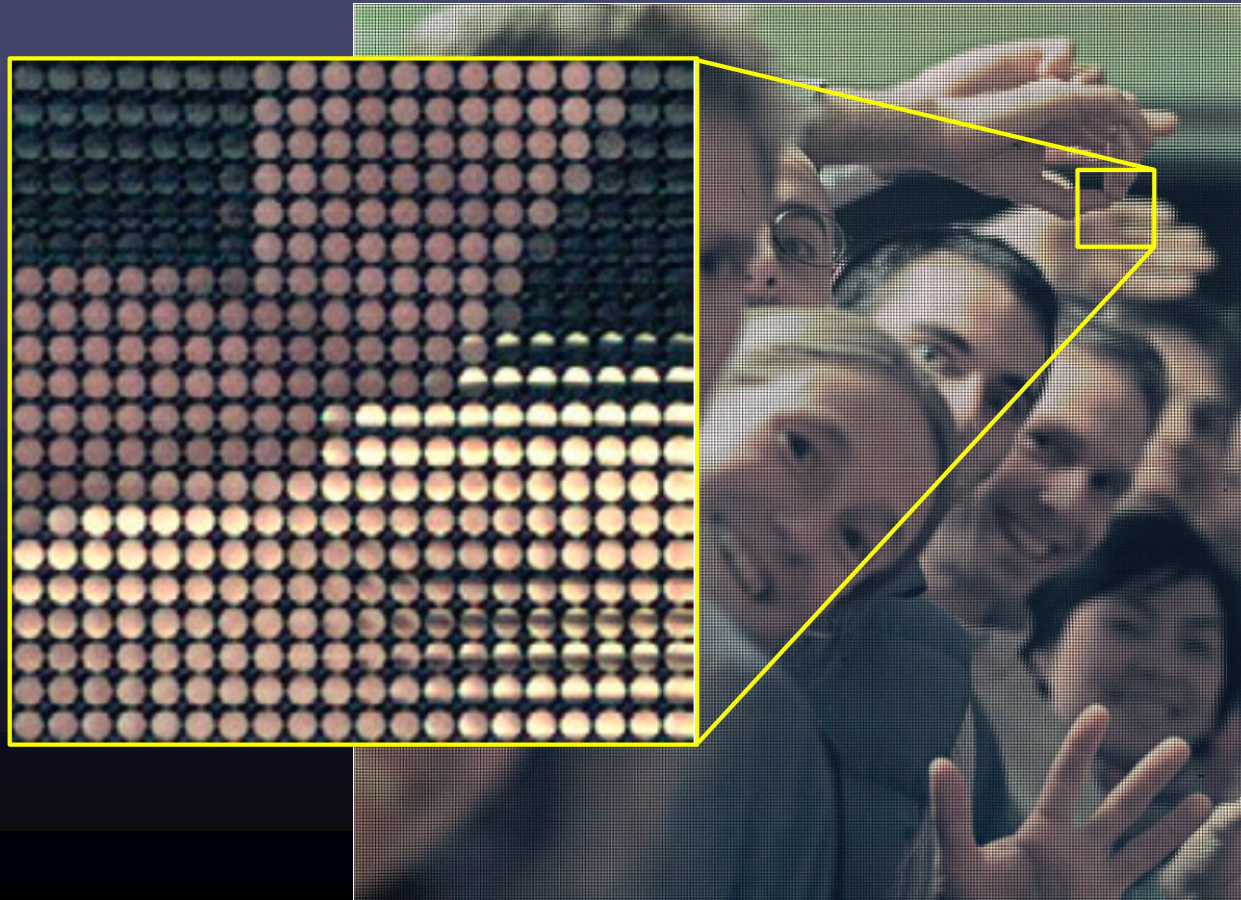
Captured light field



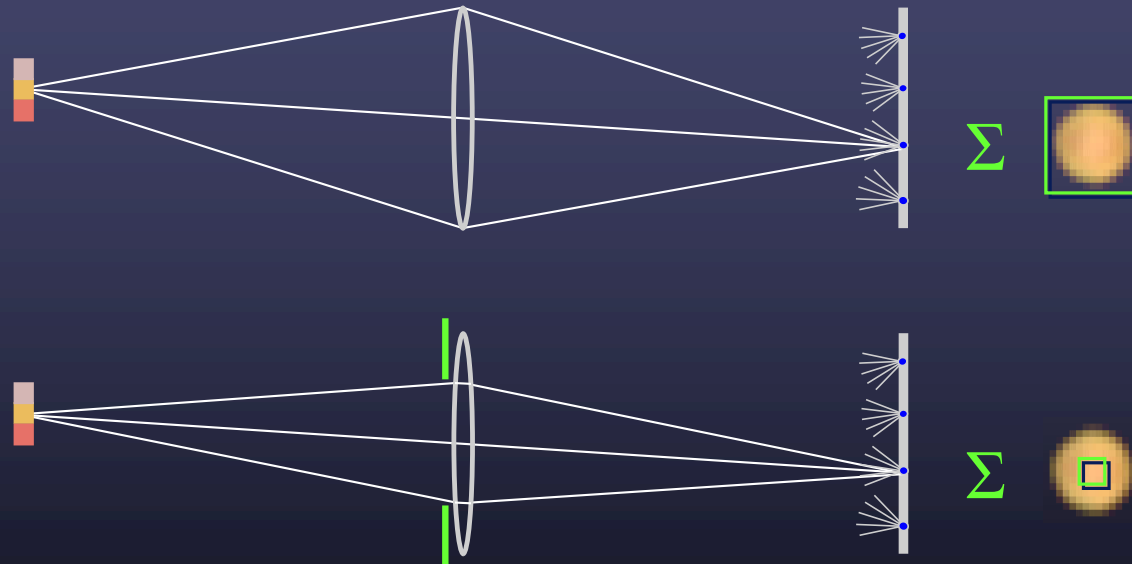
Captured light field



Captured light field

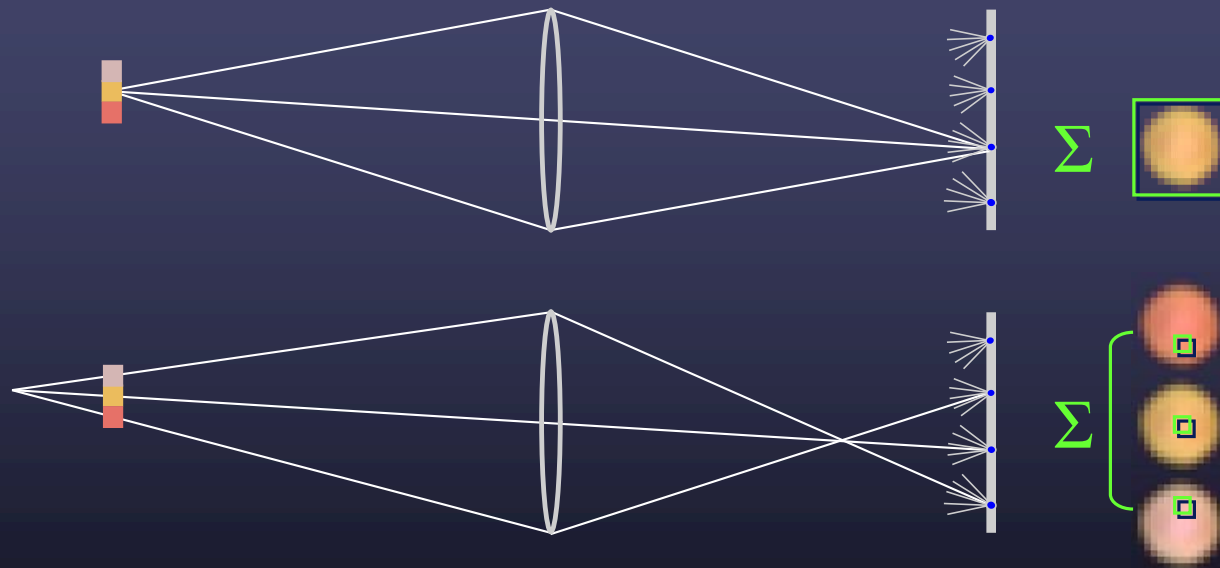


Digitally stopping down (reducing the aperture)



- stopping down = summing only the central portion of each microlens

Digital refocusing

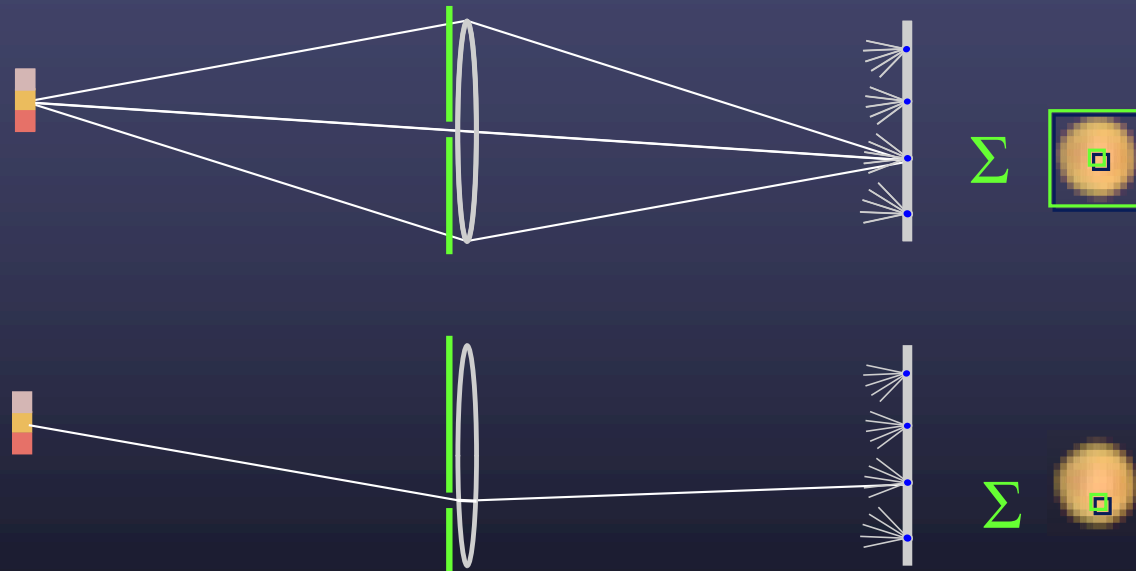


- refocusing = summing windows extracted from several microlenses

Digital refocusing

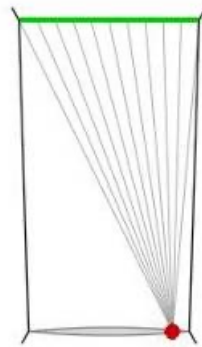
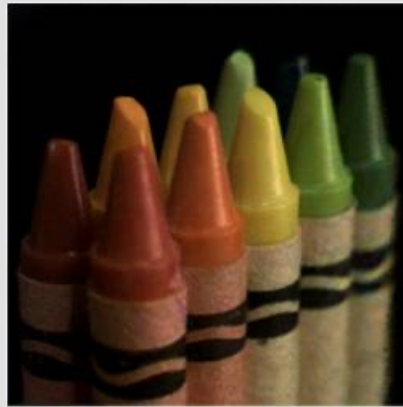


Digitally moving the observer

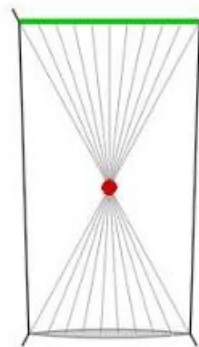


- moving the observer = moving the window we extract from the microlenses

Digitally moving the observer



Digitally moving the observer



Lytro (RIP)



<https://en.wikipedia.org/wiki/Lytro>

What happened to Lytro?