Focal stacks and lightfields

http://graphics.cs.cmu.edu/courses/15-463

15-463, 15-663, 15-862
Computational Photography
Fall 2023, Lecture 10
Course announcements

• Homework assignment 4 is out.
  - Due November 3rd
  - Any questions?

• Mid-semester grades will be posted by 4 pm today.

• Competition results for homework assignment 2 posted.
Overview of today’s lecture

• Quick reminder: pinhole vs lens cameras.
• Focal stack.
• Confocal stereo.
• Lightfield.
• Measuring lightfields.
• Plenoptic camera.
• Images from lightfields.
• Some notes on (auto-)focusing.
Many of these slides were adapted from:

- Fredo Durand (MIT).
- Gordon Wetzstein (Stanford).
- Marc Levoy (Stanford).
Pinhole vs lens cameras
Pinhole camera

- Everything is in focus.
- Very light inefficient.
• Only one plane is in focus.
• Very light efficient.

How can we get an all in-focus image?
Focal stack
Focal stack

- Capture images focused at multiple planes.
- Merge them into a single all in-focus image.

Analogous to what we did in HDR
- Focal stack instead of exposure stack.
Homework 4: focal stack imaging
Focal stack imaging

1. Capture a focal stack

2. Merge into an all in-focus image
Focal stack imaging

1. Capture a focal stack

2. Merge into an all in-focus image
How do you capture a focal stack?

Which of these parameters would you change (and how)?

- lens-sensor distance $D'$
- lens-object distance $D$
- lens aperture $f/#$
- lens focal length $f$
How do you capture a focal stack?

Which of these parameters would you change (and how would you achieve that)?

- lens-sensor distance $D'$
- lens-object distance $D$
- lens aperture $f/$
- lens focal length $f$
- rotate lens focus ring (not zoom!)
How do you capture a focal stack?

Which of these parameters would you change (and how would you achieve that)?
Capturing a focal stack

In-focus plane in each stack image
Focal stack imaging

1. Capture a focal stack

2. Merge into an all in-focus image
How do you merge a focal stack?
How do you merge a focal stack?

1. Align images
2. Assign per-pixel weights representing “in-focus”-ness
3. Compute image average
How do you merge a focal stack?

1. Align images
2. Assign per-pixel weights representing “in-focus”-ness
3. Compute image average
Image alignment

Why do we need to align the images?
Image alignment

Why do we need to align the images?
• When we change focus distance, we also change field of view (magnification).
• Also, scene may not be static (but we will be ignoring this for now).
Image alignment
Image alignment
Why do we need to align the images?
• When we change focus distance, we also change field of view (magnification).
• Also, scene may not be static (but we will be ignoring this for now).

- Assume we know $f$ and all $D'$ values.
- How do we can we align the images?
Image alignment

Why do we need to align the images?
• When we change focus distance, we also change field of view (magnification).
• Also, scene may not be static (but we will be ignoring this for now).

\[
m = \frac{f}{D' - f}
\]

\[
\frac{1}{D'} + \frac{1}{D} = \frac{1}{f}
\]

- Assume we know \( f \) and all \( D' \) values.
- How do we align the images?

change in magnification

change in focus

resize using these equations
How can we avoid having to do alignment?
Use a telecentric lens

Place a pinhole at focal length, so that only rays parallel to primary ray pass through.

Magnification independent of object depth.

Magnification depends only on sensor-lens distance $S'$.

[Watanabe and Nayar, PAMI 1997]
How do you merge a focal stack?

1. Align images
2. Assign per-pixel weights representing “in-focus”-ness
3. Compute image average
Weight assignment

How do we measure how much “in-focus” each pixel is?
Weight assignment

How do we measure how much “in-focus” each pixel is?
• Measure local sharpness.

How do we measure local sharpness?
Weight assignment

How do we measure how much “in-focus” each pixel is?
• Measure local sharpness.

run Laplacian operator
do some Gaussian blurring (why?)
Weight assignment

How do we measure how much “in-focus” each pixel is?
• Measure local sharpness.

run Laplacian operator
do some Gaussian blurring (so that nearby pixels have similar weights)

Just one example, many alternatives possible.
How do you merge a focal stack?

1. Align images
2. Assign per-pixel weights representing “in-focus”-ness
3. Compute image average
Focal stack merging

and divide by sum of weights
Some results

example image from stack

all in-focus image
Another example

Focal stacking is very useful in macrophotography, where depths of field are very shallow
Another example

middle image from stack

all in-focus image
Another look at the mixing weights

What do the mixing weights look like?
Another look at the mixing weights

Depth from focus = determining sharpest pixel in focal stack
Depth from focus on a mobile phone

Use focal stack from autofocus

[Suwajanakorn et al., CVPR 2015]
Another look at the mixing weights

What is a problem of these depth maps?
Another look at the mixing weights

What is a problem of these depth maps?
- Blurry because we need to process entire neighborhoods to compute sharpness.
- Can we use any extra information to get per-pixel depth?
Confocal stereo
Confocal stereo

- Capture a 2D stack by varying both focus and aperture
- Analyze each pixel’s focus-aperture image to find true depth

[Hassinof and Kutulakos, ECCV 2006]
Aperture-Focus Image of pixel p

(aperture $\alpha_i$, focus $f_j$)

aperture $\alpha$

focus $f$

pixel $p$
AFI model fitting

Key idea: When a point is in focus, its color and intensity remain the same for all apertures.

- This property is called *confocal constancy*.
- We can find correct depth by checking intensity variation across apertures for each focus setting.
Depth inference techniques

Confocal stereo:
• Requires focus-aperture stack.
• Gives per-pixel depth estimates.

Depth from focus:
• Requires focus stack.
• Gives per-patch depth estimates.

What is a downside of these two approaches?
Reminder: Circle of confusion

Size of circle of confusion depends on distance from in-focus plane.

\[
\frac{y}{D/2} = \frac{|O - S|}{O}
\]

\[
\frac{y}{c/2} = \frac{1}{m}
\]

\[c = mD \frac{|O - S|}{O}\]
Depth from defocus

Use as few as two images (two-frame depth from defocus):
- Assume circle of confusion can be modeled as a (typically Gaussian) blur kernel with varying $\sigma$.
- Use pair of images to estimate how blur size at each pixel changes from one depth to another.
- Relate this blur size to depth.
Requires very elaborate modeling and priors to be robust.

[Tang et al., CVPR 2017]
Depth inference techniques

Confocal stereo:
• Requires focus-aperture stack.
• Gives per-pixel depth estimates.

Depth from focus:
• Requires focus stack.
• Gives per-patch depth estimates.

Depth from defocus:
• Requires only two images.
• Gives per-patch depth estimates.

What is a downside of these three approaches?
Focal flow

Use a dense focal stack (e.g., from autofocus):

• Assume circle of confusion can be modeled as a (typically Gaussian) blur kernel with varying σ.
• Estimate optical flow between successive frames in focal stack.
• Relate optical flow change to depth.

Requires very little computation (no convolutions) but only works on textured patches (why?).

\[
\begin{bmatrix}
I_x & I_y & (xI_x + yI_y) & (I_{xx} + I_{yy})
\end{bmatrix} \tilde{u}_4 + I_t = 0
\]

https://www.youtube.com/watch?v=vHdjMgo47BQ

[Alexander et al., ECCV 2016; Guo et al., ICCV 2017]
Focal flow

Use a dense focal stack (e.g., from autofocus):

• Assume circle of confusion can be modeled as a (typically Gaussian) blur kernel with varying σ.
• Estimate optical flow between successive frames in focal stack.
• Relate optical flow change to depth.

Requires very little computation (no convolutions) but only works on textured patches (why?).

[Alexander et al., ECCV 2016; Guo et al., ICCV 2017]

Biologically-inspired from jumping spiders!

https://www.youtube.com/watch?v=vHdjMgo47BQ
Depth inference techniques

Confocal stereo:
• Requires focus-aperture stack.
• Gives per-pixel depth estimates.

Depth from focus:
• Requires focus stack.
• Gives per-patch depth estimates.

Focal flow:
• Requires dense focus stack.
• Gives efficient per-patch depth estimates.

Depth from defocus:
• Requires only two images.
• Gives per-patch depth estimates.
Lightfield
A lens measures all rays radiated from the object (up to aperture size).
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
We can capture the same set of rays by using a pinhole camera from multiple viewpoints.

- How would you merge these images into a lens-based, defocused image?
Lightfield: all rays in a scene

What is the dimension of the lightfield?
Parameterize every ray based on its intersections with two planes.
Lightfield: all rays in a scene

4-dimensional function $L(u, v, s, t)$

Parameterize every ray based on its intersections with two planes.

- How else can we parameterize the lightfield?
- Where else have we seen it?
Reminder from lecture 4: paraxial optics

\[ x_i = s_i \theta_i \]
\[ x_o = s_o \theta_o \]
\[ \frac{1}{s_o} + \frac{1}{s_i} = \frac{1}{f} \]
\[ d \approx 0 \Rightarrow x_i = x_o \]

Putting it all together, we can write:

\[
\begin{bmatrix}
  x_o \\
  \theta_o
\end{bmatrix}
= \begin{bmatrix}
  1 & 0 \\
  \frac{1}{f} & 1
\end{bmatrix}
\begin{bmatrix}
  x_i \\
  \theta_i
\end{bmatrix}
\]

Ray transfer matrix: relates each incoming ray to an outgoing ray.
Reminder from lecture 4: ray transfer matrix analysis

\[
\begin{bmatrix}
    x_o \\
    \theta_o \\
\end{bmatrix}
= \begin{bmatrix}
    f(x_i, \theta_i) \\
    g(x_i, \theta_i) \\
\end{bmatrix}
\approx \begin{bmatrix}
    A & B \\
    C & D \\
\end{bmatrix}
\begin{bmatrix}
    x_i \\
    \theta_i \\
\end{bmatrix}
\]

Under paraxial approximation:

\[
A = \frac{\partial f}{\partial x_i}igg|_{x_i=\theta_i=0} \\
B = \frac{\partial f}{\partial \theta_i}igg|_{x_i=\theta_i=0} \\
C = \frac{\partial g}{\partial x_i}igg|_{x_i=\theta_i=0} \\
D = \frac{\partial g}{\partial \theta_i}igg|_{x_i=\theta_i=0}
\]

where

definition of ray transfer matrix, a.k.a. ABCD matrix

why at \(x_i = \theta_i = 0\)?
Lightfield: all rays in a scene

Parameterize every ray based on its intersections with two planes.

• How else can we parameterize the lightfield?
• Where else have we seen it?
Lightfield: all rays in a scene

Parameterize every ray based on its intersections with two planes.
Lightfield slices

aperture plane \((u, v)\)
sensor plane \((s, t)\)

What does \(L(u = u_o, v = v_o, s, t)\) look like?

4-dimensional function \(L(u, v, s, t)\)  
(conjugate of scene-based function)
Lightfield slices

aperture plane \((u, v)\)  
sensor plane \((s, t)\)

What does \(L(u = u_o, v = v_o, s, t)\) look like?
- a pinhole image from a certain viewpoint

4-dimensional function \(L(u, v, s, t)\)  
(conjugate of scene-based function)
Lightfield slices

reference plane \((s, t)\) \hspace{1cm} \text{aperture plane} \((u, v)\) \hspace{1cm} \text{sensor plane} \((s, t)\)

Lightfield slice \(L(u = u_0, v = v_0, s, t)\)
Lightfield slices

aperture plane \((u, v)\)  
sensor plane \((s, t)\)

What does \(L(u = u_o, v = v_o, s, t)\) look like?
- a pinhole image from a certain viewpoint

What does \(L(u, v, s = s_o, t = t_o)\) look like?

4-dimensional function \(L(u, v, s, t)\)  
(conjugate of scene-based function)
What does $L(u = u_o, v = v_o, s, t)$ look like?
• a pinhole image from a certain viewpoint

What does $L(u, v, s = s_o, t = t_o)$ look like?
• radiance emitted by a certain (in-focus) point at various directions

4-dimensional function $L(u, v, s, t)$ (conjugate of scene-based function)
Lightfield slices

reference plane \((s, t)\)  
aperture plane \((u, v)\)  
sensor plane \((s, t)\)

reference/sensor coordinates \(s = s_0, t = t_0\)

Lightfield slice \(L(u, v, s = s_0, t = t_0)\)
Lightfield slices

aperture plane \((u, v)\)
sensor plane \((s, t)\)

What does \(L(u = u_o, v = v_o, s, t)\) look like?
- a pinhole image from a certain viewpoint

What does \(L(u, v, s = s_o, t = t_o)\) look like?
- radiance emitted by a certain (in-focus) point in various directions

4-dimensional function \(L(u, v, s, t)\)
(conjugate of scene-based function)
Lightfield visualization

$L(u, v, s = s_0, t = t_0)$ is the radiance emitted by a certain (in-focus) point at various directions.

$L(u = u_0, v = v_0, s, t)$ is a pinhole image from a certain viewpoint.

Demo: [http://lightfield.stanford.edu/lfs.html](http://lightfield.stanford.edu/lfs.html)
Lightfield: all rays in a scene

How can you capture the lightfield of a scene?
Measuring lightfields
How to capture a lightfield?

How can you do this?
Option 1: use multiple cameras

Stanford camera array

(“synthetic aperture”)

[Willburn et al., SIGGRAPH 2005]
Option 1: use multiple cameras

Stanford camera array

What kind of lens would you use for this?

(“synthetic aperture”)

[Willburn et al., SIGGRAPH 2005]
Compound Eye vs Light Field

Image: National Geographic

Mosquito eye, image: Raija Peura, University of Oulu
Option 2: take multiple images with one camera


How would you move the camera?
Plenoptic camera
Option 3: use a plenoptic camera

plenoptic = plenus (Latin for “full”) + optic (Greek for “seeing”, in this case)
Making a plenoptic camera

reference plane \((s, t)\)

aperture plane \((u, v)\)

generator plane \((s, t)\)

Lightfield slice \(L(u, v, s = s_0, t = t_0)\)

reference/sensor coordinates
\(s = s_0, t = t_0\)

reference/sensor coordinates
\(s = s_0, t = t_0\)
Making a plenoptic camera

reference plane \((s, t)\)  
aperture plane \((u, v)\)  
sensor plane \((s, t)\)

Lightfield slice \(L(u, v, s = s_0, t = t_0)\)

each pixel measures ray \(L(u, v, s = s_0, t = t_0)\) for different \((u, v)\)

reference/sensor coordinates \(s = s_0, t = t_0\)
Making a plenoptic camera

Reference plane \((s, t)\)  
Aperture plane \((u, v)\)  
Sensor plane \((s, t)\)

Lightfield \(L(u, v, s, t)\)

Each pixel measures a ray \(L(u, v, s, t)\)

Each pinhole corresponds to a slice \(L(u, v, s = s_o, t = t_o)\)

How can we make this more light efficient?
Making a plenoptic camera

Lightfield $L(u, v, s, t)$

How can we make this more light efficient?
- replace pinholes with lenslets
First conceptualized by Gabriel Lippmann, who called it integral photography.

- Original article appeared in French journal three years earlier, what is shown here is an American re-print.
History of the plenoptic camera

First conceptualized by Gabriel Lippmann, who called it integral photography.

- Original article appeared in French journal three years earlier, what is shown here is an American re-print.

Notice the date: more than a century ago.
History of the plenoptic camera

Reappeared under different forms and names throughout the century.

• The left paper is from 1930, the right one from 1970.
History of the plenoptic camera

Single Lens Stereo with a Plenoptic Camera
Edward H. Adelson and John Y.A. Wang

Abstract—Ordinary cameras gather light across the area of their lens aperture, and the light striking a given subregion of the aperture is structured somewhat differently than the light striking an adjacent subregion. By analyzing this optical structure, one can infer the depths of objects in the scene, i.e., one can achieve “single lens stereo.” We describe a novel camera for performing this analysis. It incorporates a single main lens along with a lenticular array placed at the sensor plane. The resulting “plenoptic camera” provides information about how the scene would look when viewed from a continuum of possible viewpoints bounded by the main lens aperture. Deriving depth information is simpler than in a binocular stereo system because the correspondence problem is minimized. The camera extracts information about both horizontal and vertical parallax, which improves the reliability of the depth estimates.

I. INTRODUCTION

“E\textsc{very body} in the light and shade fills the surrounding air with infinite images of itself; and these, by infinite pyramids diffused in the air, represent this body throughout space and on every side.” Leonardo da Vinci [1] wrote.

Fig. 1. Diagram from Leonardo’s notebooks illustrating the fact that the light rays leaving an object’s surface may be considered to form a collection of cones (which Leonardo calls “pyramids”), each cone constituting an image that would be seen by a pinhole camera at a given location.

Re-introduced to computer vision and graphics in 1992, which was also when the term \textit{plenoptic camera} was coined.
History of the plenoptic camera

Figure from the 1992 paper, which shows the version of the plenoptic camera with pinholes instead of lenslets on the image plane.

Fig. 5. Array of miniature pinhole cameras placed at the image plane can be used to analyze the structure of the light striking each macropixel.
History of the plenoptic camera

Re-re-introduced to computer vision and graphics in 2005, when the term light field photography was also coined.

Light Field Photography with a Hand-held Plenoptic Camera

Ren Ng\textsuperscript{*}, Marc Levoy\textsuperscript{*}, Mathieu Brédif\textsuperscript{*}, Gene Duval\textsuperscript{\dagger}, Mark Horowitz\textsuperscript{*}, Pat Hanrahan\textsuperscript{*}

\textsuperscript{*}Stanford University \hspace{1cm} \textsuperscript{\dagger}Duval Design

Abstract

This paper presents a camera that samples the 4D light field on its sensor in a single photographic exposure. This is achieved by inserting a microlens array between the sensor and main lens, creating a plenoptic camera. Each microlens measures not just the total amount of light deposited at that location, but how much light arrives along each ray. By re-sorting the measured rays of light to where they would have terminated in slightly different, synthetic cameras, we can compute sharp photographs focused at different depths. We show that a linear increase in the resolution of images under each microlens results in a linear increase in the sharpness of the refocused photographs. This property allows us to extend the depth of field of the camera without reducing the aperture, enabling shorter exposures and lower image noise. Especially in the macrophotography regime, we demonstrate that we can also compute synthetic photographs from a range of different viewpoints. These capabilities argue for a different strategy in designing photographic imaging systems.

To the photographer, the plenoptic camera operates exactly like an ordinary hand-held camera. We have used our prototype to take hundreds of light field photographs, and we present examples of portraits, high-speed action and macro close-ups.

Keywords: Digital photography, light field, microlens array, synthetic photography, refocusing.

Externally, our hand-held light field camera looks and operates exactly like a conventional camera: the viewfinder, focusing mechanism, length of exposure, etc. are identical. Internally, we augment the 2D photosensor by placing a microlens array in front of it, as proposed by Adelson and Wang [1992] in their work on the “plenoptic camera” (They did not build this device, but prototyped a non-portable version containing a relay lens.) Each microlens forms a tiny sharp image of the lens aperture, measuring the directional distribution of light at that microlens.

This paper explains the optical recipe of this camera in detail, and develops its theory of operation. We describe an implementation using a medium format digital camera and microlens array. Using this prototype, we have performed resolution experiments that corroborate the limits of refocusing predicted by the theory. Finally, we demonstrate examples of refocusing and view-point manipulation involving close-up macro subjects, human portraits, and high-speed action.

2 Related Work

The optical design of our camera is very similar to that of Adelson and Wang’s plenoptic camera [1992]. Compared to Adelson and Wang, our prototype contains two fewer lenses, which significantly shortens the optical path, resulting in a portable camera. These differences are explained in more detail Section 3.1 once sufficient technical background has been introduced. The other main differ-
Prototype plenoptic camera

- predecessor of Lytro
- resolution: 292x292px, 14x14 light field views

Kodak 16-megapixel sensor

125μ square-sided microlenses

[Ng et al., Stanford Technical Report 2005]
Commercial plenoptic camera

Lens
The Lytro Light Field Camera starts with an 8X optical zoom, f/2 aperture lens. The aperture is constant across the zoom range allowing for unheard of light capture.

Light Field Engine 1.0
The Light Field Engine replaces the supercomputer from the lab and processes the light ray data captured by the sensor.

Light Field Sensor
From a roomful of cameras to a micro-lens array specially adhered to a standard sensor, the Lytro’s Light Field Sensor captures 11 million light rays.
Commercial plenoptic camera

**Lens**
The Lytro Light Field Camera starts with an 8X optical zoom, 1/2 aperture lens. The aperture is constant across the zoom range allowing for unheard of light capture.

**Light Field Sensor**
From a roomful of cameras to a micro-lens array specially adhered to a standard sensor, the Lytro’s Light Field Sensor captures 11 million light rays.

**Light Field Engine 1.0**
The Light Field Engine replaces the supercomputer from the lab and processes the light ray data captured by the sensor.

The Light Field Engine travels with every living picture as it is shared, letting you refocus pictures right on the camera, on your desktop and online.

**lenslet array**
Commercial plenoptic camera

newer version with higher resolution
Industrial plenoptic cameras

Plenoptic cameras have become quite popular in lab and industrial settings.

- Much higher resolution, both spatial and angular, than commercial cameras.
- Support interchangeable lenses.
- Can do video.
- Very expensive.
Shack-Hartmann wavefront sensors

- Completely different use: measuring how close a wavefront is to being planar.
- Exactly the same optics as a plenoptic camera.
- Common instrument in optics labs and medical imaging.
Question

Given that the plenoptic camera design was known since 1908, why did it take a century for such a camera to be made commercially available?
Question

Given that the plenoptic camera design was known since 1908, why did it take a century for such a camera to be made commercially available?

• Difficult to manufacture good lenslet arrays at high resolution (unless attached to sensors).

• Digital sensors did not have sufficiently high resolution.

• No known good use for them besides depth sensing.
  - The introduction of the handheld plenoptic camera was strongly influenced by the 1996 lightfield papers, which demonstrated all of the useful photographic operations one could replicate by having access to the entire lightfield.

• And of course, nobody thought of commercializing them.
Is it possible to make a plenoptic camera using an array of pinholes?
Is it possible to make a plenoptic camera using an array of pinholes?
• Yes, under certain conditions on the images being captured.

[Georgiev, ECCV 2008]
Images from lightfields
A plenoptic “image”

What are these circles?
A plenoptic camera

reference plane \((s, t)\)

aperture plane \((u, v)\)

sensor plane \((s, t)\)

Lightfield \(L(u, v, s, t)\)

Each lenslet corresponds to a slice \(L(u, v, s = s_o, t = t_o)\)
The plenoptic image

Which coordinates do I change when I move from one circle to another?

Which coordinates do I change when I move within each circle?
The plenoptic image

Which coordinates do I change when I move from one circle to another?
• I change s, t (sensor plane) coordinates.

Which coordinates do I change when I move within each circle?
• I change u, v (aperture plane) coordinates.
How do I...

Simulate different viewpoints?
Simulate different viewpoints?
• Pick same pixel within each lenslet view
Changing the viewpoint

Viewpoint change is limited by the aperture of each of the lenslets.
How do I...

Simulate different viewpoints?
• Pick same pixel within each lenslet view

Simulate different aperture sizes?
How do I...

Simulate different viewpoints?
- Pick same pixel within each lenslet view

Simulate different aperture sizes?
- Sum more than one pixels within each lenslet view
How do I...

Simulate different viewpoints?
• Pick same pixel within each lenslet view

Simulate different aperture sizes?
• Sum more than one pixels within each lenslet view

Simulate lens at current focus setting?
How do I...

Simulate different viewpoints?
• Pick same pixel within each lenslet view

Simulate different aperture sizes?
• Sum more than one pixels within each lenslet view

Simulate lens at current focus setting?
• Same as above. Sum all pixels for max aperture setting.

How do I change focus setting?
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints.
Measuring rays

We can capture the same set of rays by using a pinhole camera from multiple viewpoints

• How would you merge these images into a lens, defocused image?
Measuring rays

Sum all pixels in each lenslet view.
Form lens image

reference plane \((s, t)\)

aperture plane \((u, v)\)

sensor plane \((s, t)\)

How do I refocus?

Sum all pixels in each lenslet view.
Form lens image

How do I refocus?

• Need to move sensor plane to a different location.
Understanding Refocus

- consider light field inside camera
- synthesize image on sensor $i_{d=0}(x) = \int_\Omega l(x,\nu) d\nu$

\[
i_d(x) = \int_\Omega l(x + d\nu, \nu) d\nu
\]
Understanding Refocus – Fourier Slicing

- Fourier slice theorem: projection in primal is slicing in Fourier space

\[ i_d(x) = \int_{\Omega} l(x + d\nu, \nu) d\nu \quad \leftrightarrow \quad \hat{i}_d(f_x) = \hat{l}(f_x, f_\nu - df_x) \]
Refocusing example
Refocusing example
Refocusing example
Synthesizing images from a lightfield

Many more examples with demo: http://lightfields.stanford.edu/
Three ways to measure a lightfield

1) Use a plenoptic camera

2) Use a camera array

3) Use one camera multiple times

What are the pros and cons of each?
Some notes on (auto-)focusing
Different cameras have different focusing processes
Manual focus in rangefinder cameras

- Focusing based on triangulation: when the image is in focus, you will see the two copies aligned.
- Very accurate but very painstaking.
- Different perspective than that of the main lens.

standard in Leica cameras
Manual focus in (D)SLR cameras

• Same view as main lens.
• Just rotate the focusing ring until you are satisfied by the sharpness.
• Viewfinder indicators can help this process.

These arrows will tell you if the focus is in/out and the green dot lets you know that you are in focus.

grid of points where sharpness is evaluated
Manual focus in (D)SLR cameras

- Same view as main lens.
- Just rotate the focusing ring until you are satisfied by the sharpness.
- Viewfinder indicators can help this process.

Instead of a grid, you can also focus based on a single point.
Active auto-focus: time-of-flight sensors

- Basically how SONAR works (we’ll also see time-of-flight sensors later in class.
- Method used in Polaroid cameras, which used ultrasound waves.
- Energy inefficient.
- Limited range.
- Multi-path interference (e.g., glass surfaces back-reflected the waves).
Passive auto-focus: phase detection

• As the lens moves, ray bundles from an object converge to a different point in the camera and change in angle.
• This change in angle causes them to refocus through two lenslets to different positions on a separate AF sensor.
• A certain spacing between these double images indicates that the object is “in focus”.

Demo: http://graphics.stanford.edu/courses/cs178/applets/autofocuspd.html
Passive auto-focus: phase detection

Each yellow box indicates *two* sensors, each measuring light from different parts of the aperture.
- Which one is correct focusing?
- How do you need to move the lens or sensor to get correct focusing?
Passive auto-focus: contrast detection

- Sensors at different image distances will see the same object at high-contrast if it’s in focus, or low-contrast if it’s not.
- Move the lens until the high-contrast sub-image falls on the middle sensor, which is conjugate to the camera’s main sensor.
- Compute contrast using local differences of pixel values. Effectively the same as depth from focus.

High-end DSLRs use phase detection

- Distance between sub-images allows lens to move directly into focus, without hunting
- Many AF points corresponding to different points on imaging sensor, complicated algorithms for choosing among them: generally use closest point, but also consider position in FOV.
Low-end cameras (and phones) use contrast detection

- Nowadays it is mostly done using main camera sensor instead of dedicated sensors.
- Requires repeated measurements as lens moves, which are captured using the main sensor (an “autofocus stack”).
- Slow, requires hunting, suffers from overshooting.

But

- People have come up with creative uses for the autofocus stack (depth-from-focus on a cell phone, HDR+ on Android).

[Suwajanakorn et al., CVPR 2015; Hasinoff et al., SIGGRAPH Asia 2016]
Low-end cameras (and phones) use contrast detection

- Nowadays it is mostly done using main camera sensor instead of dedicated sensors.
- Requires repeated measurements as lens moves, which are captured using the main sensor (an “autofocus stack”).
- Slow, requires hunting, suffers from overshooting.

But

- People have come up with creative uses for the autofocus stack (depth-from-focus on a cell phone, HDR+ on Android).

[Suwajanakorn et al., CVPR 2015; Hasinoff et al., SIGGRAPH Asia 2016]
Modern mirror-less cameras use phase detection

• Dedicate a small number of pixels on the imaging sensor to work for phase detection.
• Do this at different parts of the sensor to be able to autofocus at different parts of the image.

Any downsides?
Dual-pixel phase detection autofocus

- Split each pixel into two independent photodiodes.
- Use different pixels for phase detection.
- Many other interesting opportunities (more on this during the stereo lecture).
Should you use autofocus?
Should you use autofocus?

Quick answer: Yes.
Should you use autofocus?

Quick answer: Yes.

More detailed answer: Yes, except for certain special circumstances.

• You are using a lens that does not have an autofocus motor (e.g., vintage or otherwise old lenses, high-end lenses, industrial and machine vision lenses).

• You are trying to capture an image under conditions where autofocus is prone to fail (e.g., macrophotography, poorly-lit scenes, imaging through glass or occluders).

• You intentionally want some part of the scene to be out of focus (e.g., for artistic effect, or because you want a face or other featured to be obscured).

• You are in an once-in-a-lifetime opportunity to photograph something, and you cannot afford to risk autofocus failing. This additionally assumes that:
  - Your scene is static enough that you can take the time to focus manually.
  - You are experienced enough so that the probability of manual focus failing is smaller than the probability of autofocus failing.
References

Basic reading:
• Szeliski textbook, Section 12.1.3, 13.3.

Additional reading:
  One of the (relatively) early papers on depth from defocus.
• Suwajanakorn et al., “Depth from Focus with Your Mobile Phone,” CVPR 2015.
  Implementing depth from defocus on a mobile phone using the autofocus focal stack.
  The paper on high resolution depth from a focus and aperture stack.
  Continuously change focus within one exposure, without stopping to capture a stack.
• Levoy and Hanrahan, “Light Field Rendering,” SIGGRAPH 1996.
  The two papers introducing the light field.
  The paper (re)-introducing the plenoptic camera to computer vision and graphics.
• Ng et al., “Light field photography with a hand-held plenoptic camera,” Stanford TR 2005.
  The paper (re)-(re)-introducing the plenoptic camera, and the precursor to Lytro.
• Ng, “Fourier Slice Photography,” SIGGRAPH 2005.
  The paper on frequency-space analysis of refocusing and lightfield measurements.
  The camera array paper.
  Make a lightfield camera from a pinhole array (and many other interesting stuff about lightfield cameras).