

# Cameras

---



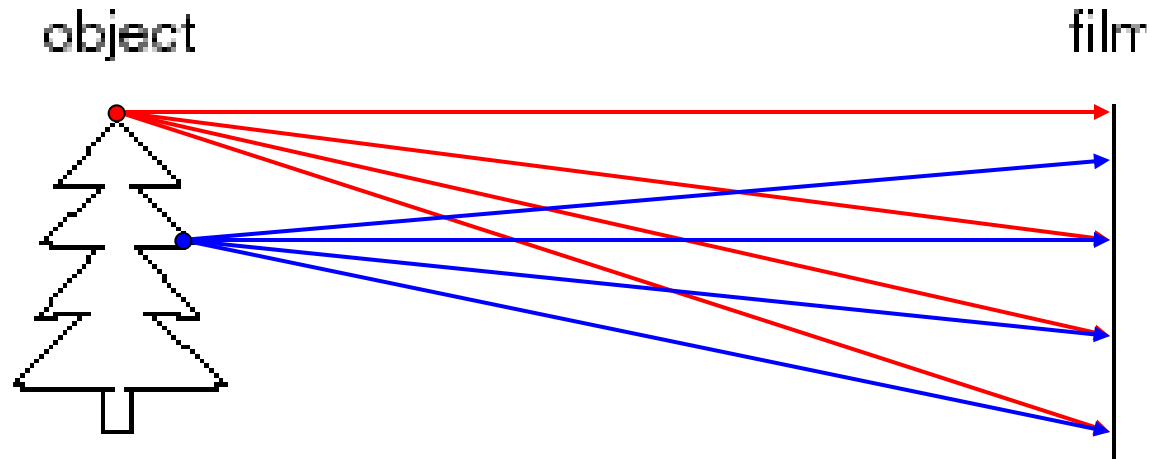
# План лекции

---

- **Камеры – формирование изображения**
  - Простейшая модель камеры
  - Особенности формирования изображения
  - Камеры с линзами
  - Реальные камеры
- **Цветное изображение**
  - Краткая история цветного фото
  - Цифровые цветные камеры
  - Задание! Ранняя цветная фотография (Прокудин-Горский)
- **Введение в MATLAB**
  - Описание среды MATLAB
  - Основные конструкции языка и матричные операции

# Let's design a camera

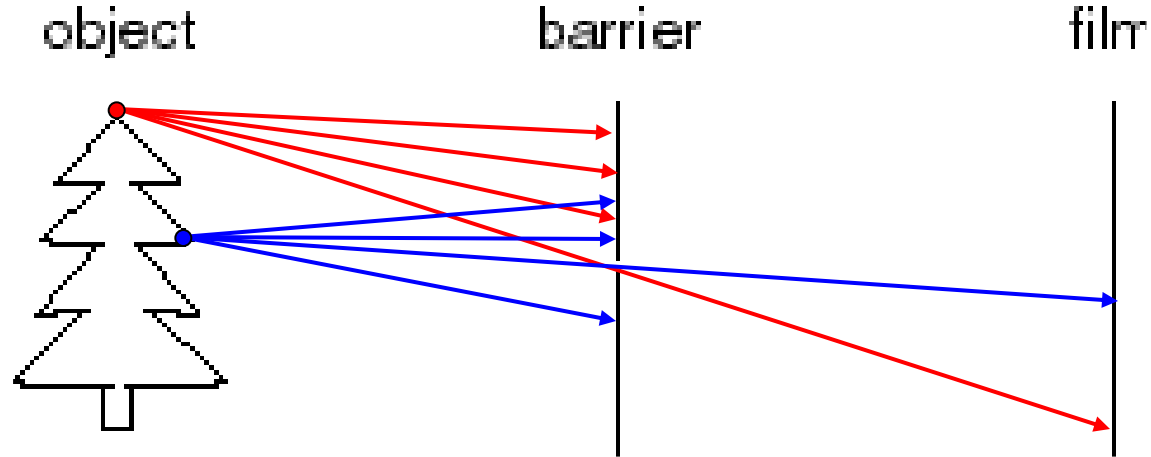
---



Idea 1: put a piece of film in front of an object  
Do we get a reasonable image?

# Pinhole camera

---

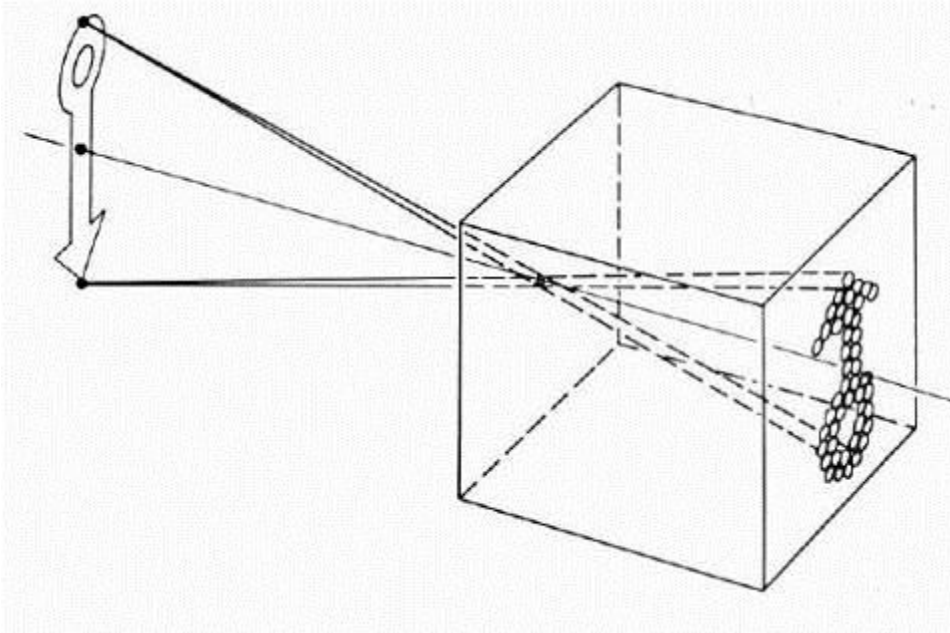


Add a barrier to block off most of the rays

- This reduces blurring
- The opening is known as the **aperture**

# Pinhole camera model (стенон)

---



## Pinhole model:

- Captures **pencil of rays** – all rays through a single point
- The point is called **Center of Projection (focal point)**
- The image is formed on the **Image Plane**

# Perspective distortion

---

- What does a sphere project to?

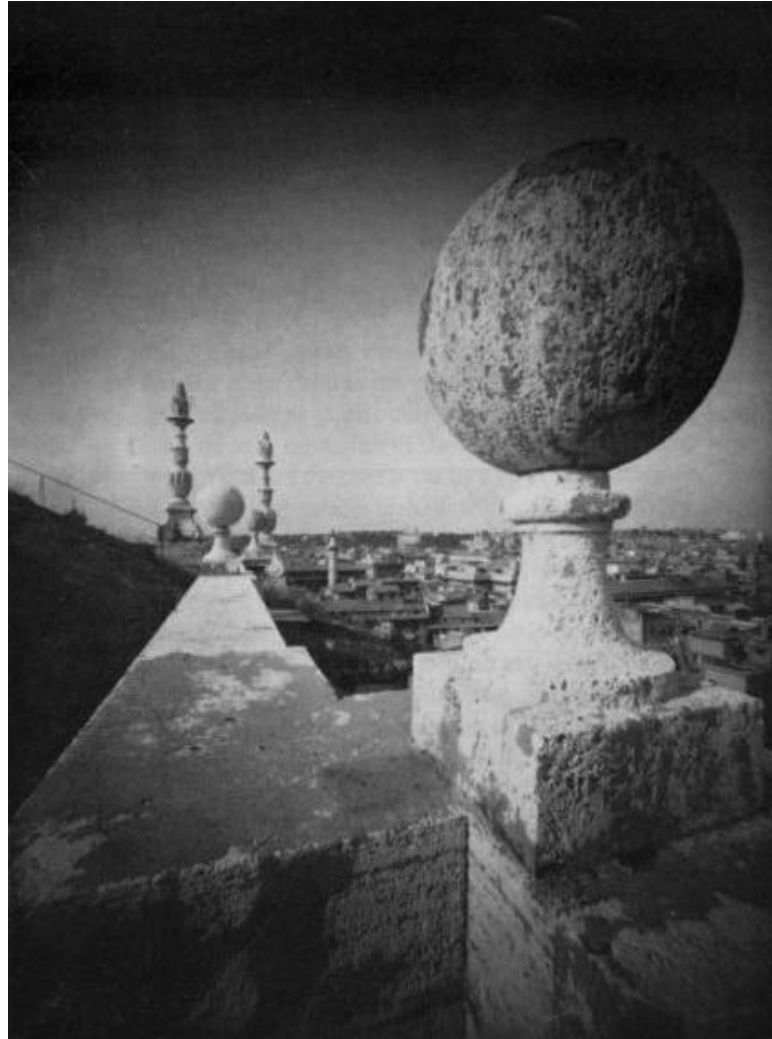
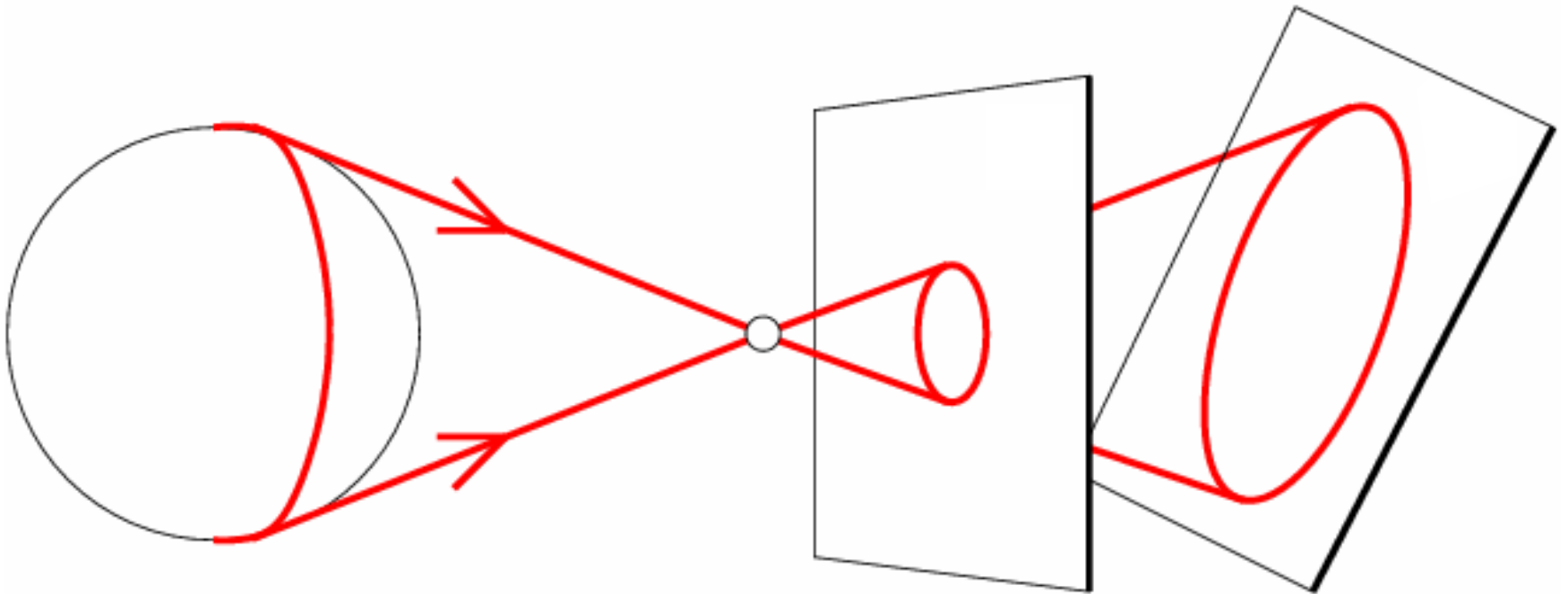


Image source: F. Durand

# Perspective distortion

---

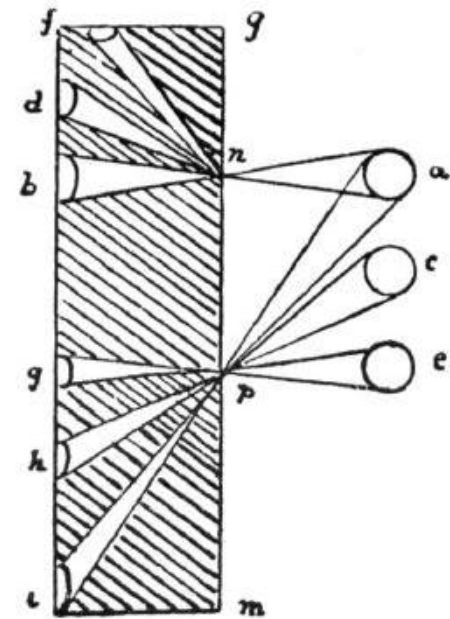
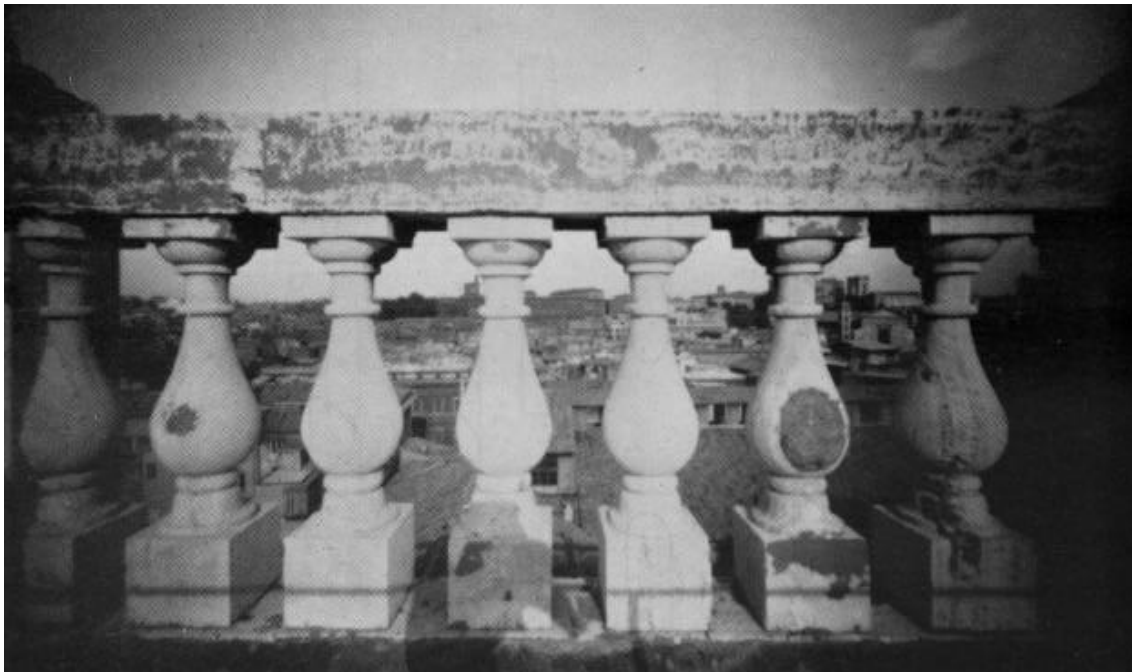
- What does a sphere project to?



# Perspective distortion

---

- The exterior columns appear bigger
- The distortion is not due to lens flaws
- Problem pointed out by Da Vinci





# Perspective distortion: People

---



# Perspective distortion

---

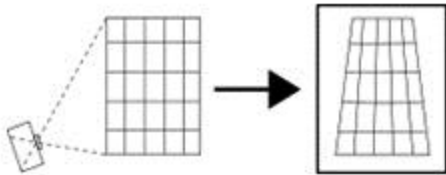
- Problem for architectural photography: converging verticals



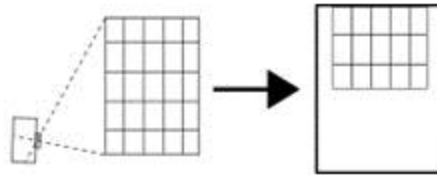
# Perspective distortion

---

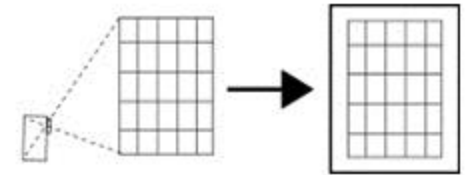
- Problem for architectural photography: converging verticals



Tilting the camera upwards results in converging verticals

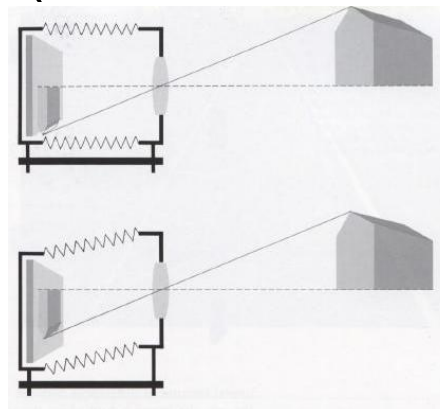
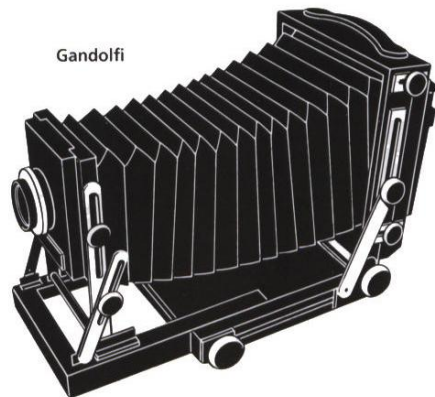


Keeping the camera level, with an ordinary lens, captures only the bottom portion of the building



Shifting the lens upwards results in a picture of the entire subject

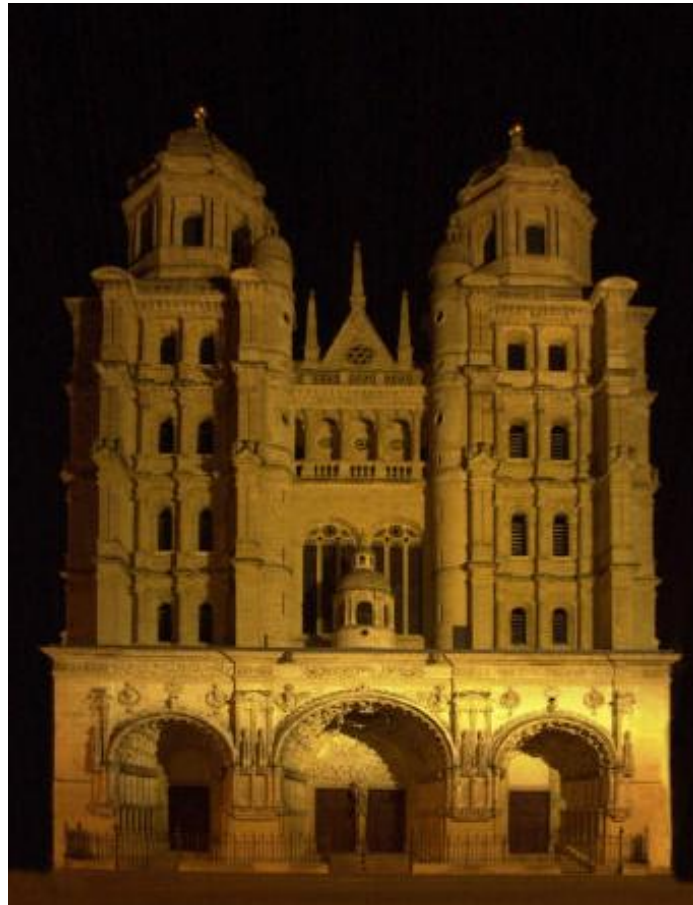
- Solution: view camera (lens shifted w.r.t. film)



# Perspective distortion

---

- Problem for architectural photography: converging verticals
- Result:



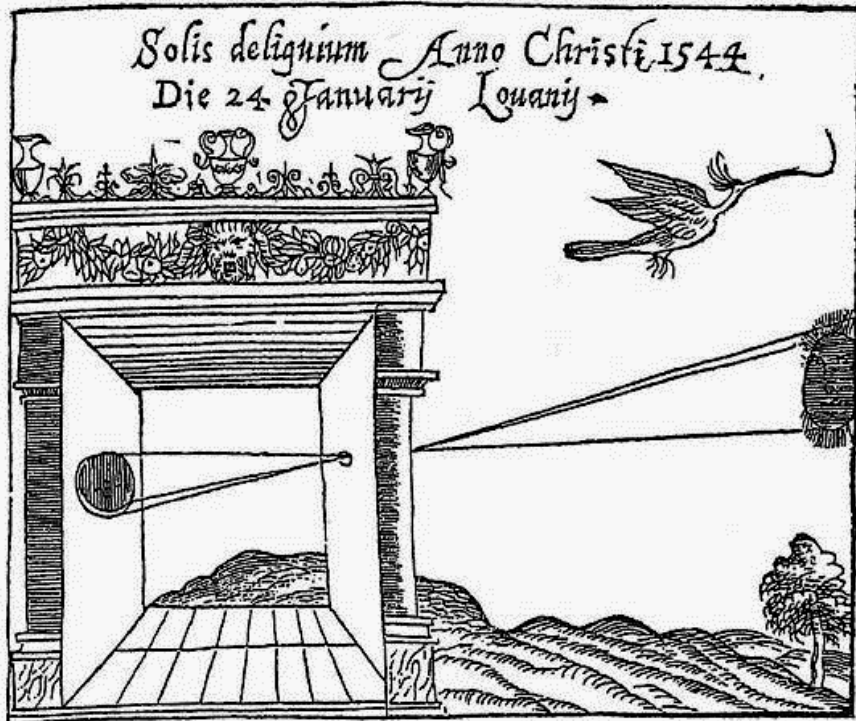
# Building a real camera

---



# Camera Obscura (темная комната)

---



Gemma Frisius, 1558

- Basic principle known to Mozi (470-390 BCE), Aristotle (384-322 BCE)
- Drawing aid for artists: described by Leonardo da Vinci (1452-1519)



# Home-made pinhole camera

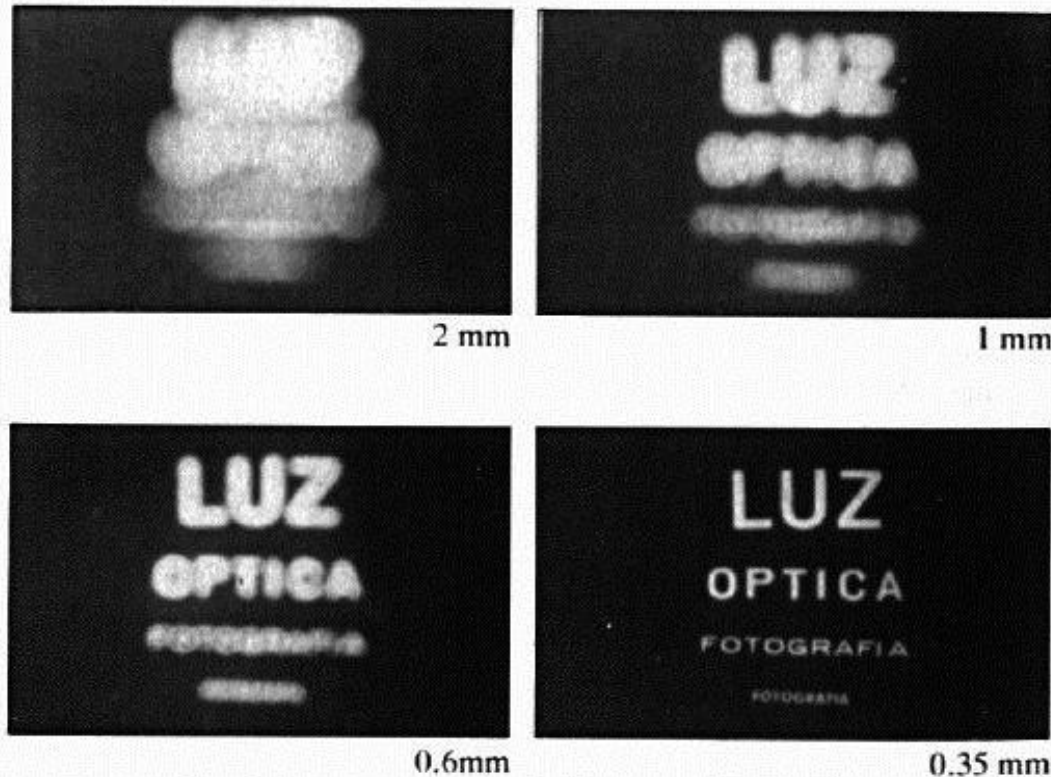
---



Why so  
blurry?

# Shrinking the aperture

---



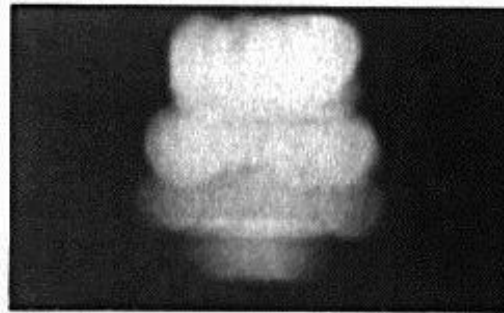
Why not make the aperture as small as possible?

- Less light gets through
- Diffraction effects...

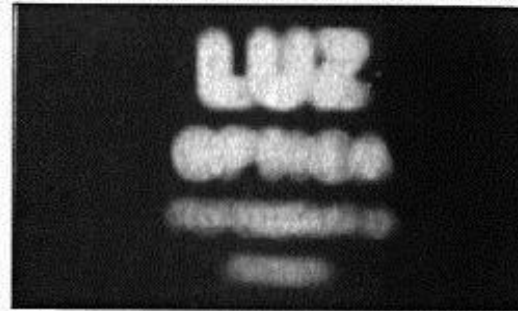


# Shrinking the aperture

---



2 mm



1 mm



0.6mm



0.35 mm



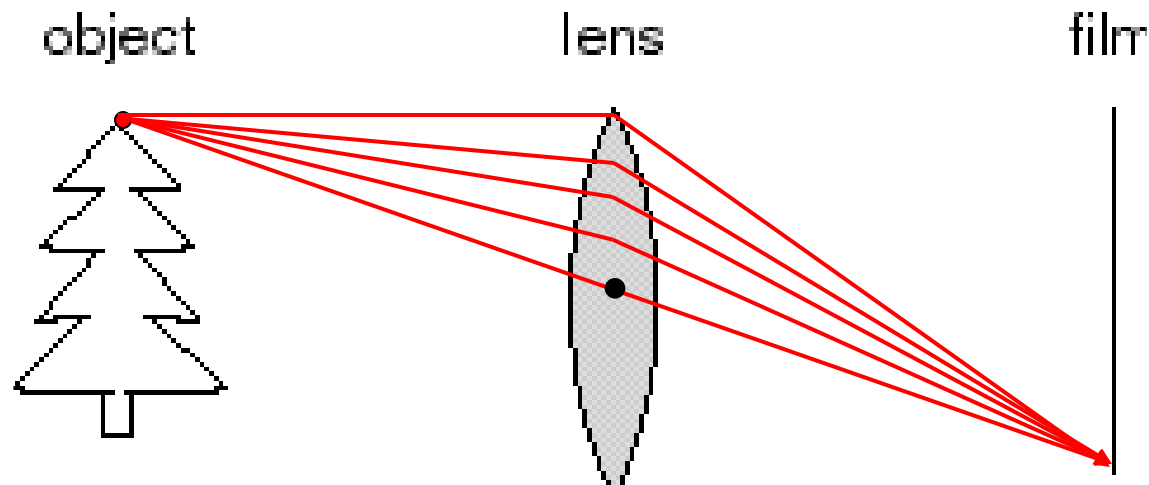
0.15 mm



0.07 mm

# Adding a lens

---

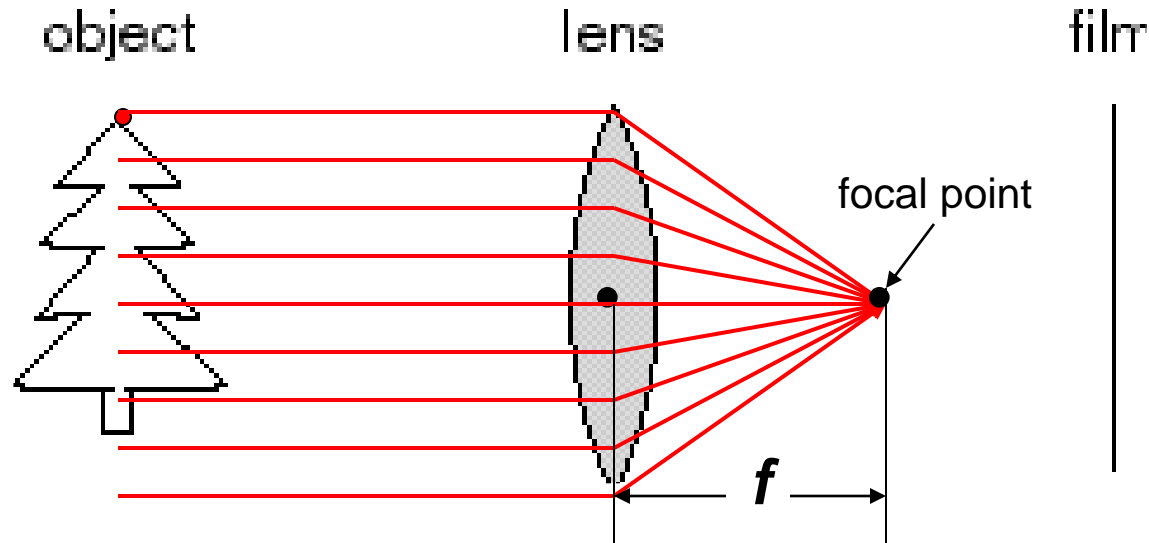


## A lens focuses light onto the film

- Thin lens model:
  - Rays passing through the center are not deviated (pinhole projection model still holds)

# Adding a lens

---

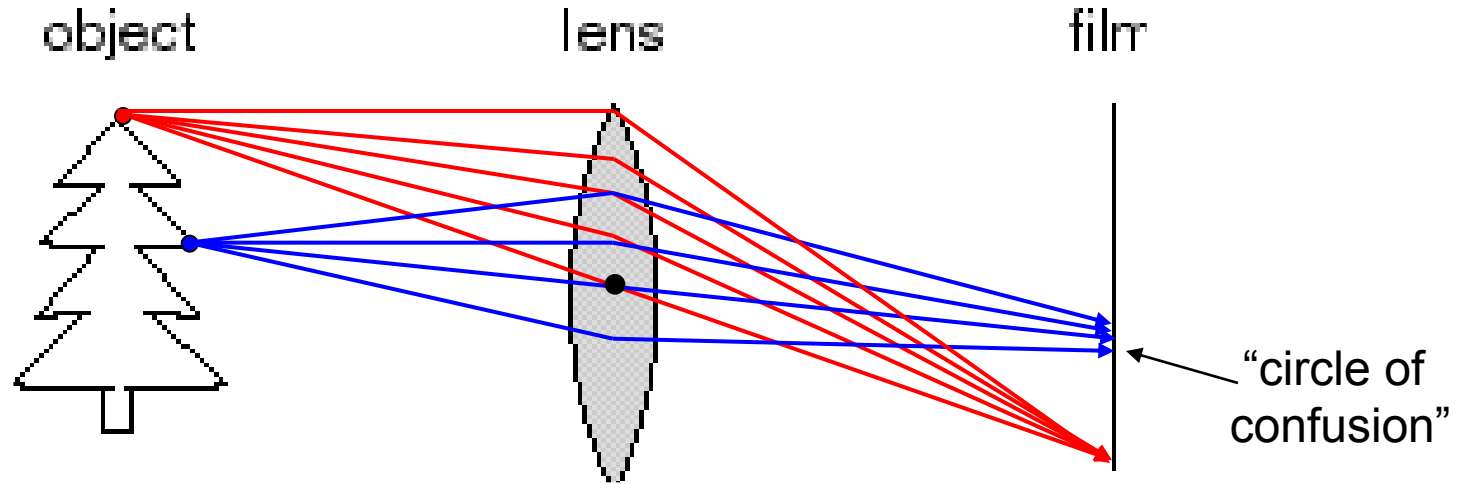


## A lens focuses light onto the film

- Thin lens model:
  - Rays passing through the center are not deviated (pinhole projection model still holds)
  - All parallel rays converge to one point on a plane located at the *focal length*  $f$

# Adding a lens

---

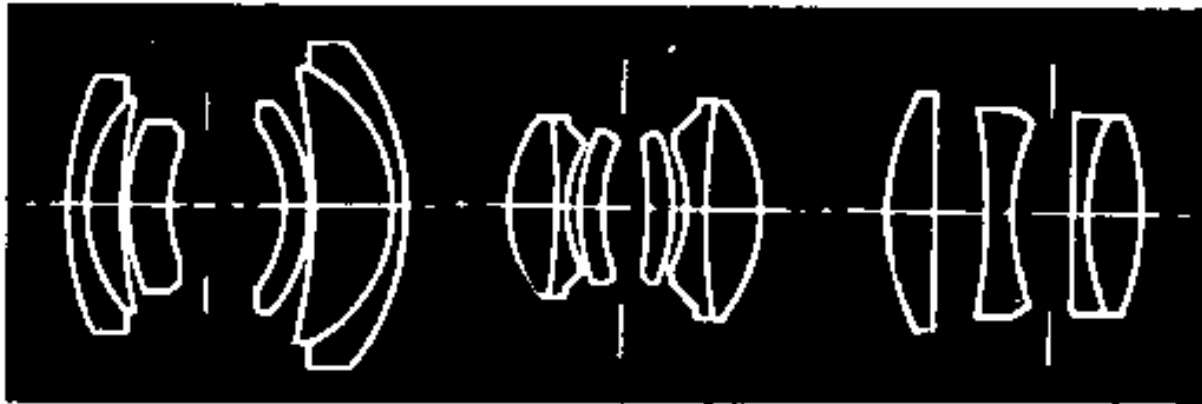


## A lens focuses light onto the film

- There is a specific distance at which objects are “in focus”
  - other points project to a “circle of confusion” in the image

# Real lenses

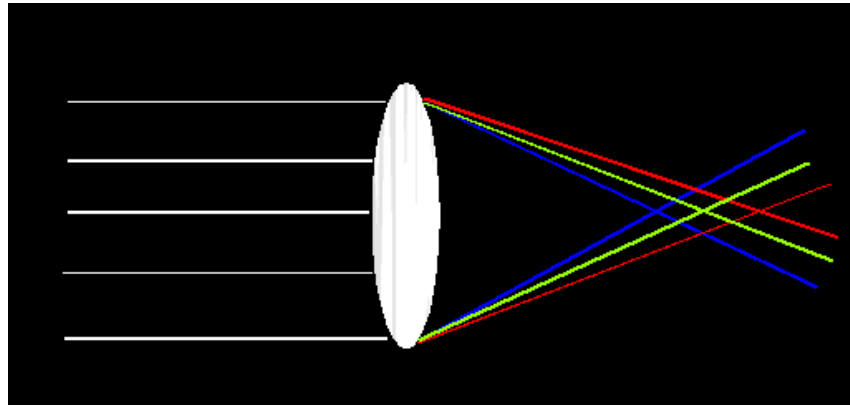
---



# Lens Flaws: Chromatic Aberration

---

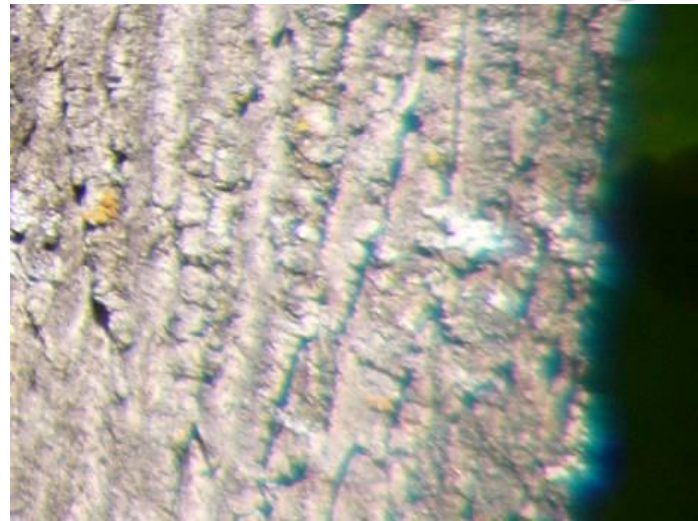
Lens has different refractive indices for different wavelengths: causes color fringing



Near Lens Center



Near Lens Outer Edge

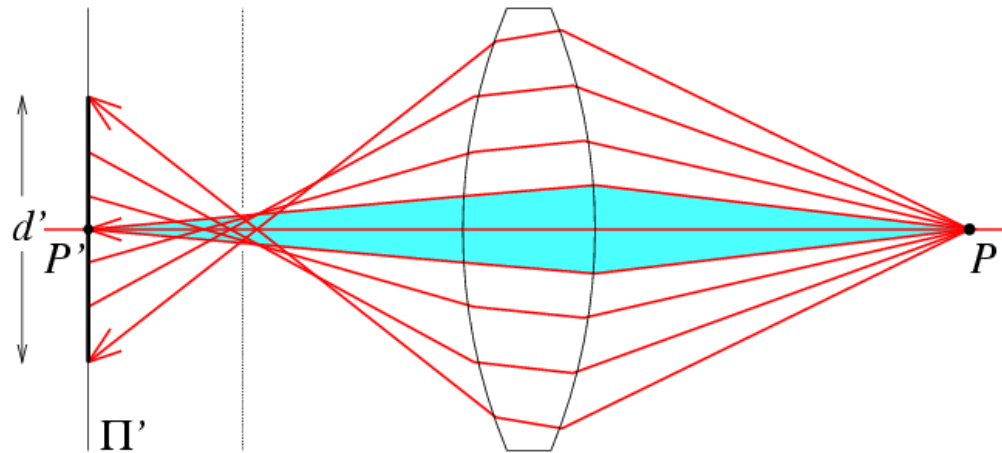


# Lens flaws: Spherical aberration

---

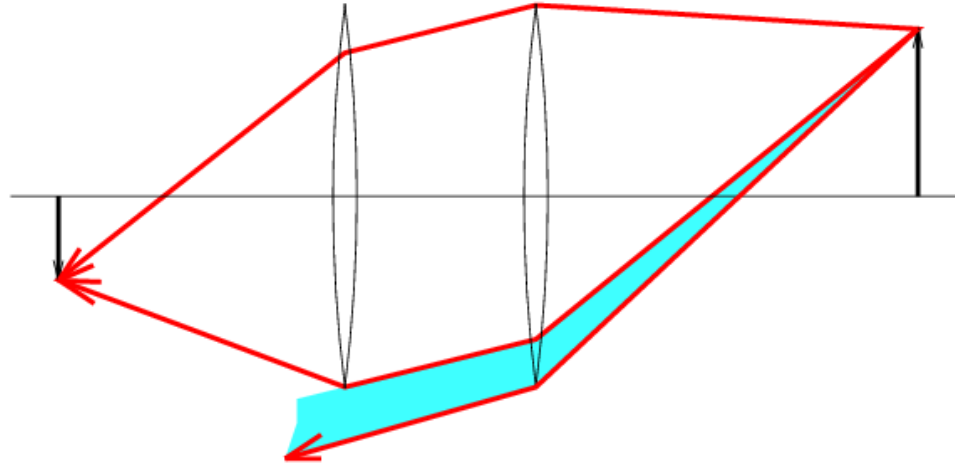
Spherical lenses don't focus light perfectly

Rays farther from the optical axis focus closer



# Lens flaws: Vignetting

---

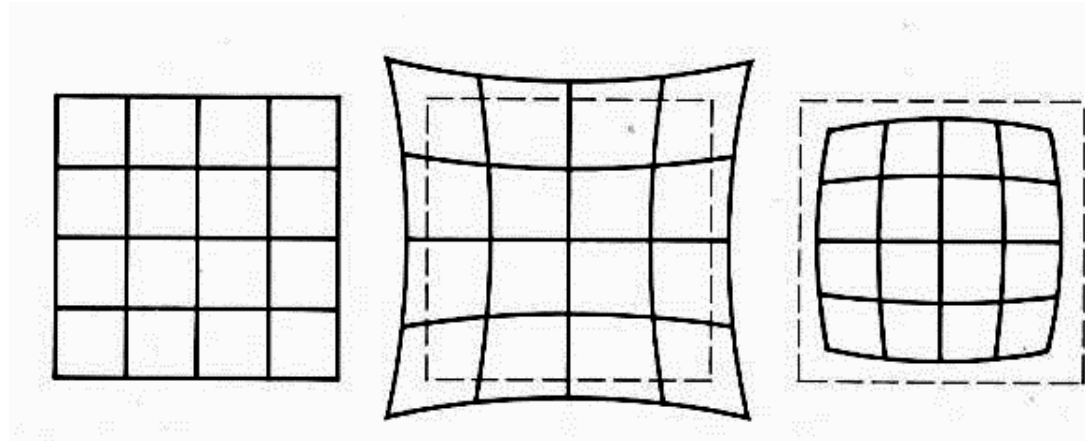




# Radial Distortion

---

- Caused by imperfect lenses
- Deviations are most noticeable near the edge of the lens



No distortion

Pin cushion

Barrel



# Color photography

---



Nikolaevskii Cathedral from southwest in Mzhaisk (1911)

# Short history of color photography

---

- 1611, Marco Antonio de Dominis, synthesis of all colors by mixing red, green, violet
- 1666, Newton, splitting light into spectrum
- 1801, Thomas Young, three types of color-sensitive nerves (Young-Helmholtz theory)
- 1861, James Clark Maxwell, first color photo demonstration through three color-filtered projectors
- 1907, Lumiere brothers, first commercial color film, the Autochrome plates

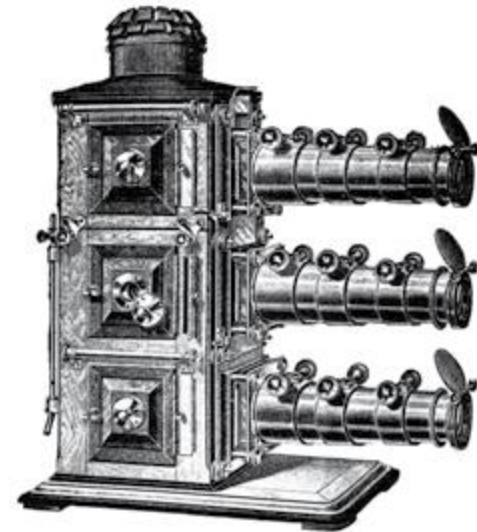
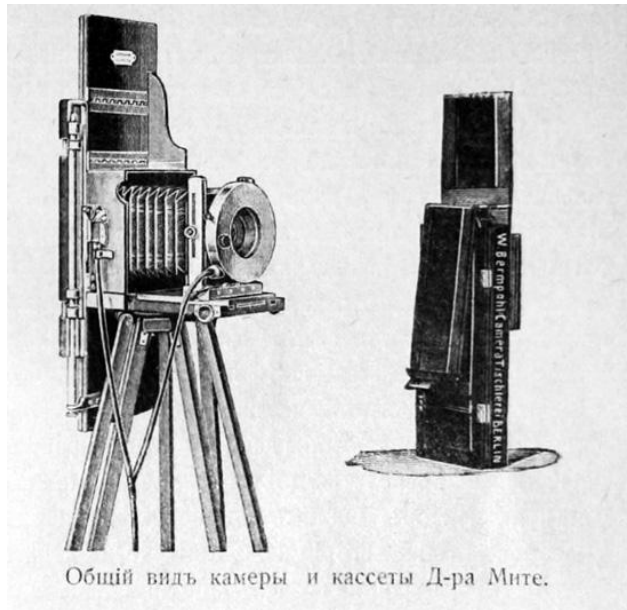




# Prokudin-Gorsky

---

1901 – Adolf Miethe constructed camera for color photography



Lantern projector

1902 – Prokudin-Gorsky working in Adolf Miethe laboratory improving sensitizers

1903 – special color projector built

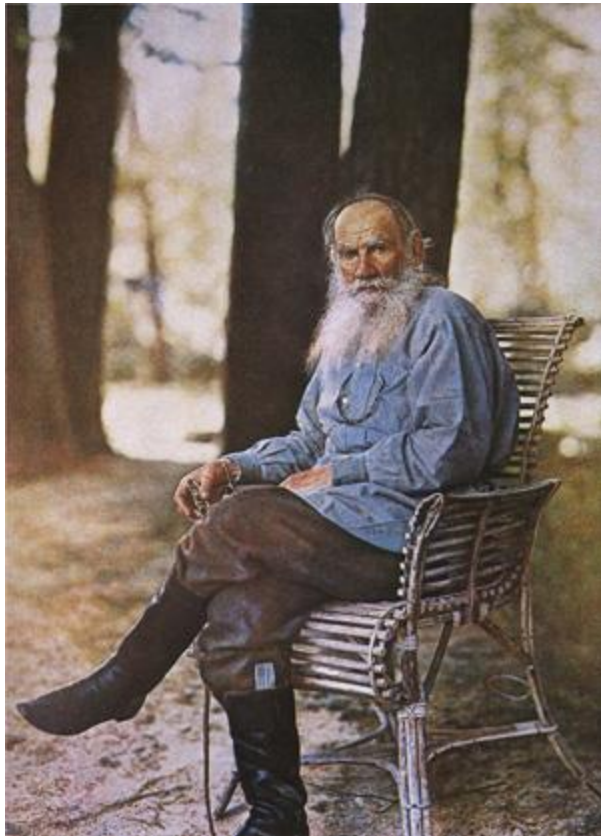
# Prokudin-Gorsky

---

1905 – started photographic expeditions

1909 – expeditions supported by Nikolai II

1909-1916 – about 2 thousands of photos taken



L.N. Tolstoy, may 1908



2 expeditions along  
Volga river, 1910

# Prokudin-Gorsky technique

---

Сергей Михайлович Прокудин-Горский (1863-1944)



Задание!

[http://en.wikipedia.org/wiki/Sergei\\_Mikhailovich\\_Prokudin-Gorskii](http://en.wikipedia.org/wiki/Sergei_Mikhailovich_Prokudin-Gorskii)

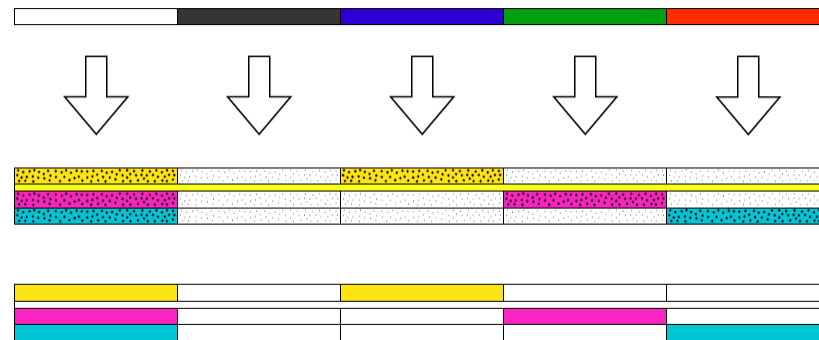
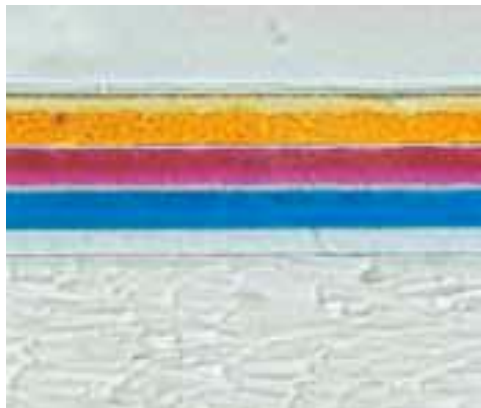
<http://www.loc.gov/exhibits/empire/>



# Color film photography

---

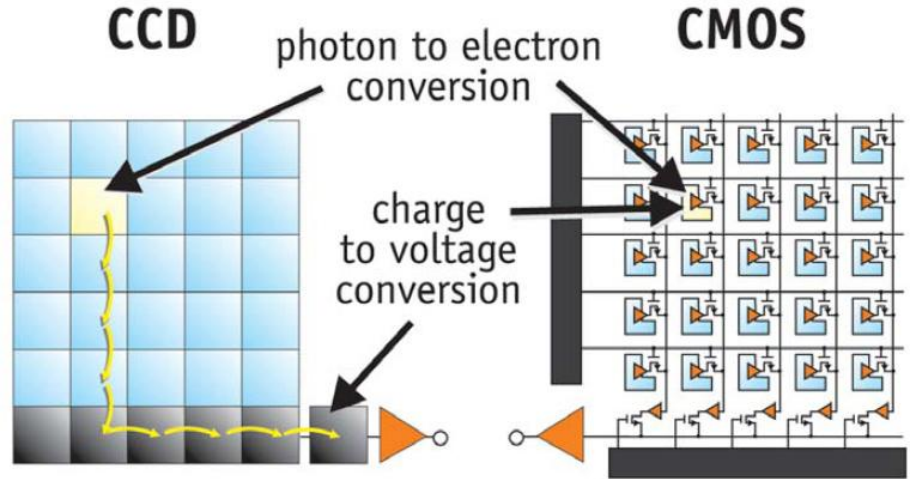
- Development of layered photographic film
- Kodak three-layer film



film development

- Detailed history of color processes:  
[http://www.iml.unibas.ch/SKRIPTEN/ScriptColor/color\\_photography\\_history.pdf](http://www.iml.unibas.ch/SKRIPTEN/ScriptColor/color_photography_history.pdf)

# Digital camera



CCDs move photogenerated charge from pixel to pixel and convert it to voltage at an output node. CMOS imagers convert charge to voltage inside each pixel.

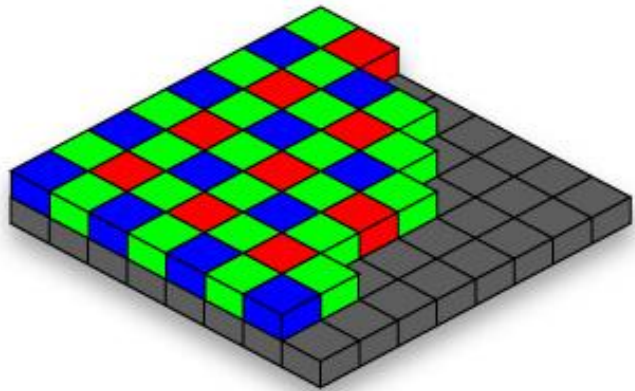
## A digital camera replaces film with a sensor array

- Each cell in the array is light-sensitive diode that converts photons to electrons
- Two common types
  - **Charge Coupled Device (CCD)**
  - **Complementary metal oxide semiconductor (CMOS)**
- <http://electronics.howstuffworks.com/digital-camera.htm>

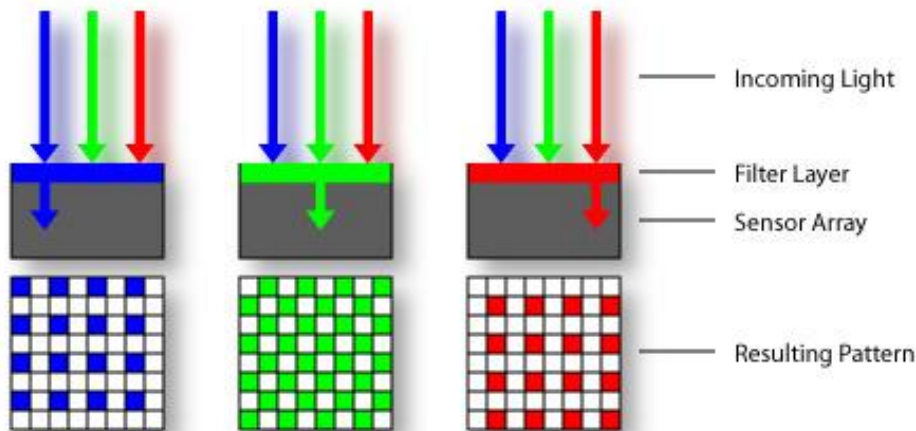


# Color sensing in camera: Color filter array

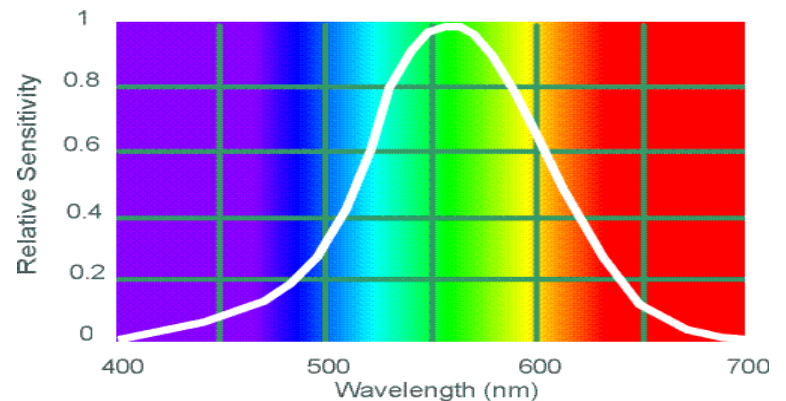
## Bayer grid



Estimate missing components from neighboring values (demosaicing)



## Why more green?



Human Luminance Sensitivity Function

# Problem with demosaicing: color moire

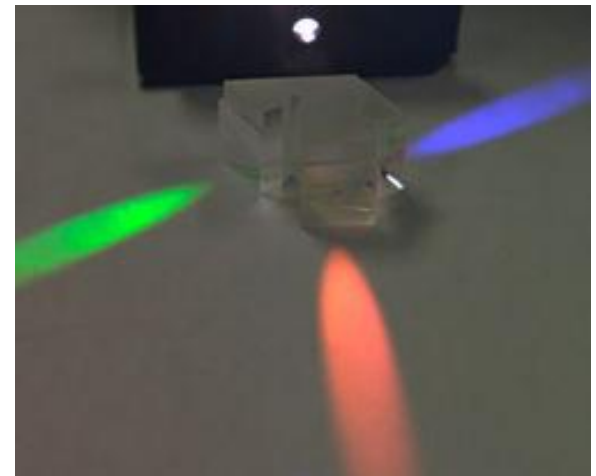
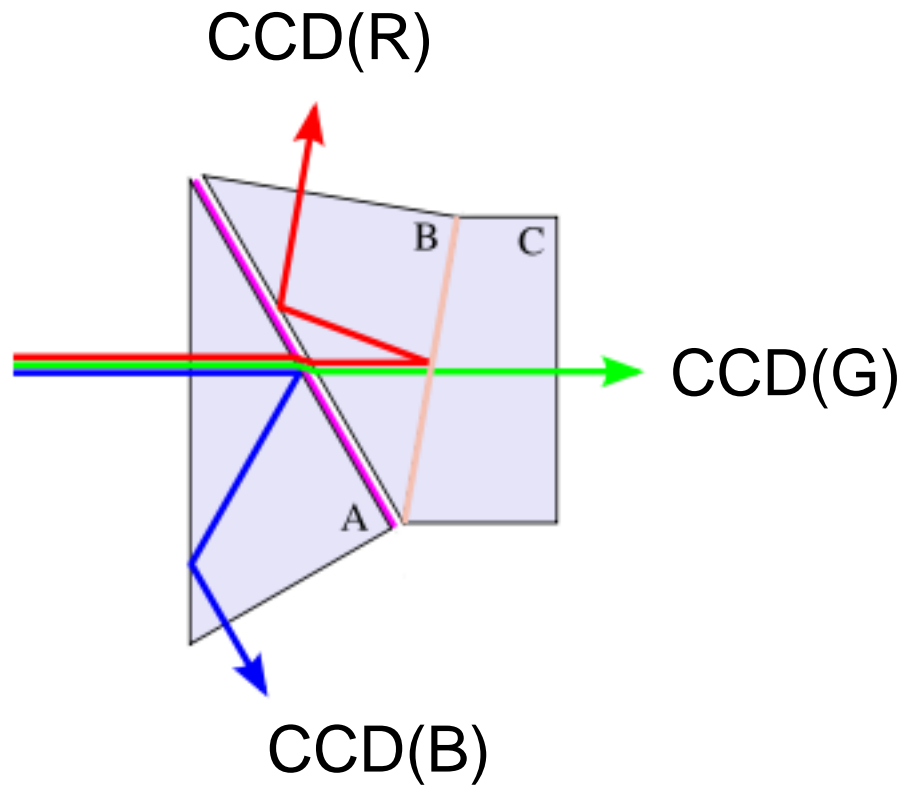
---



# Color sensing in camera: Prism

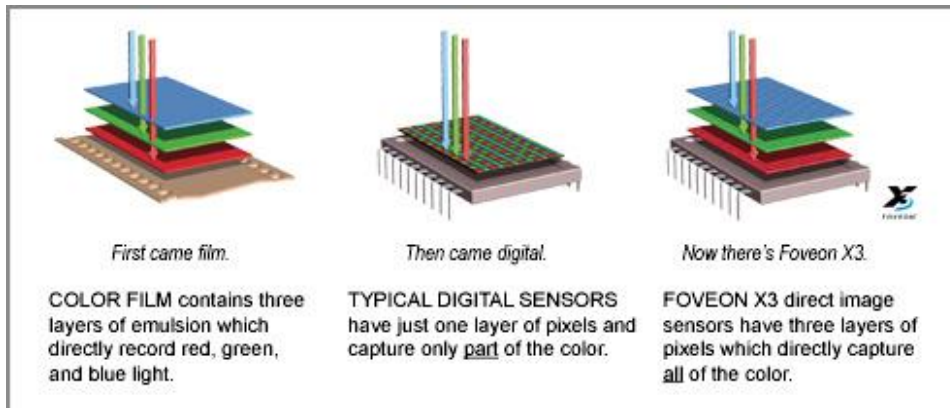
---

- Requires three chips and precise alignment
- More expensive

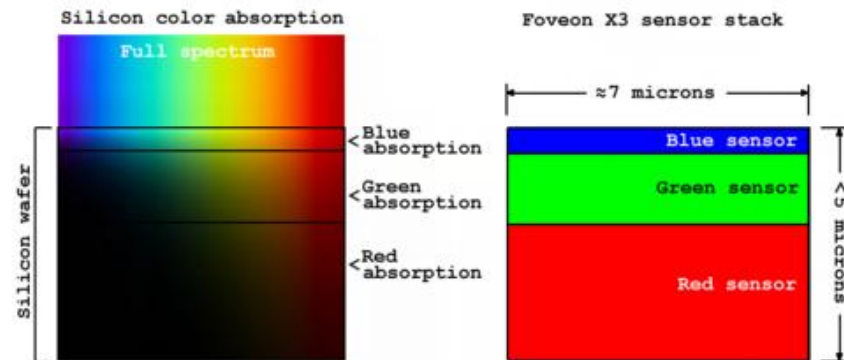


# Color sensing in camera: Foveon X3

- CMOS sensor
- Takes advantage of the fact that red, blue and green light penetrate silicon to different depths

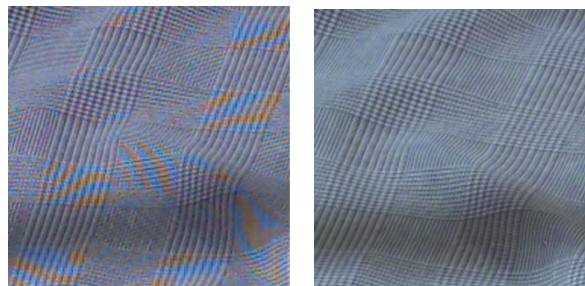


<http://www.foveon.com/article.php?a=67>



[http://en.wikipedia.org/wiki/Foveon\\_X3\\_sensor](http://en.wikipedia.org/wiki/Foveon_X3_sensor)

**better image quality**



# Digital camera artifacts

---

## Noise

- low light is where you most notice [noise](#)
- light sensitivity (ISO) / noise tradeoff
- stuck pixels



## In-camera processing

- oversharpening can produce [halos](#)



## Compression

- JPEG artifacts, blocking

## Blooming

- charge [overflowing](#) into neighboring pixels



## Color artifacts

- [purple fringing](#) from microlenses,
- white balance

# Задание

---

- **Обязательная часть**
  - Написать программу на MATLAB для автоматического совмещения трех каналов изображения. Программа должна быть хорошо прокомментирована.
  - Подготовить краткий отчет с описанием алгоритма и результатами (pdf)
  - Выслать архив с программой и отчетом (название файла *фамилия.zip*) на [mizotin@cs.msu.ru](mailto:mizotin@cs.msu.ru) с тегом [ProkudinTask].
- **Опциональные улучшения**
  - Поиск смещения с помощью пирамиды или FFT
  - Совмещение с субпиксельной точностью
  - Автоматическое ретуширование дефектов (inpainting)
  - Постобработка и другие улучшения по вашему усмотрению (напр. автокоррекция цвета)

# Правила выполнения работы

---

- Разрешается обсуждение идей реализации алгоритмов
- Запрещается детальное объяснение реализации, копирование кода
- Разрешается использование любых источников информации, источник должен быть упомянут в отчете
- Разрешается использование программ для MATLAB из интернета (также с указанием источника) если они не решают поставленную задачу целиком

# Идея выполнения

---

- Разбить изображение на 3 (можно поровну поделить по высоте)
- Выбрать диапазон поиска ( $\pm 15$  пикселей по горизонтали и вертикали)
- Парно совместить каналы (R с G, B с G): для каждой пары сдвигов вычислить ошибку наложения, выбрать для совмещения сдвиг, где ошибка наименьшая
- Возможные функции измерения ошибки: SSD (сумма квадратов отклонений попиксельно), скалярное произведение



## Идея выполнения (2)

---

- Скользящее скалярное произведение называется кросс-корреляцией

$$(f \star g)[n] \stackrel{\text{def}}{=} \sum_{m=-\infty}^{\infty} f^*[m] g[n + m]. \quad (\text{дискретная})$$

- Кросс-корреляция часто используется для поиска в сигнале (1d, 2d) «шаблона».
- Для изображений лучше применять normalized cross-correlation (учёт разного контраста)
- Связь корреляции с преобразованием Фурье (теорема умножения)

$$\mathcal{F}\{f \star g\} = (\mathcal{F}\{f\})^* \cdot \mathcal{F}\{g\},$$

# Субпиксельное уточнение (1)

---

Для поиска наилучшего наложения каналов изображений с субпиксельной точностью рекомендуется использовать один из двух подходов:

1. Расчет кросс-корреляции через ДПФ (дискретное преобразование Фурье) и использование техники дополнения преобразования ДПФ нулями для интерполяции результата. О дополнении нулями кратко написано здесь:

[http://www.vibration.ru/preobraz\\_fur.shtml](http://www.vibration.ru/preobraz_fur.shtml)

ДПФ, естественно быстрое - БПФ

## Субпиксельное уточнение (2)

---

### 2. Квадратичная интерполяция

(приближение поверхностью вида  $sf(x,y) = p_{00} + p_{10}x + p_{01}y + p_{20}x^2 + p_{11}xy + p_{02}y^2$ ) в окрестности максимума кросс-корреляции. За точное значение сдвига принимаются координаты максимума найденной поверхности.

Поверхность можно найти методом наименьших квадратов.

После нахождения оптимального сдвига интерполировать одно из совмещаемых изображений (функция `interp2`).