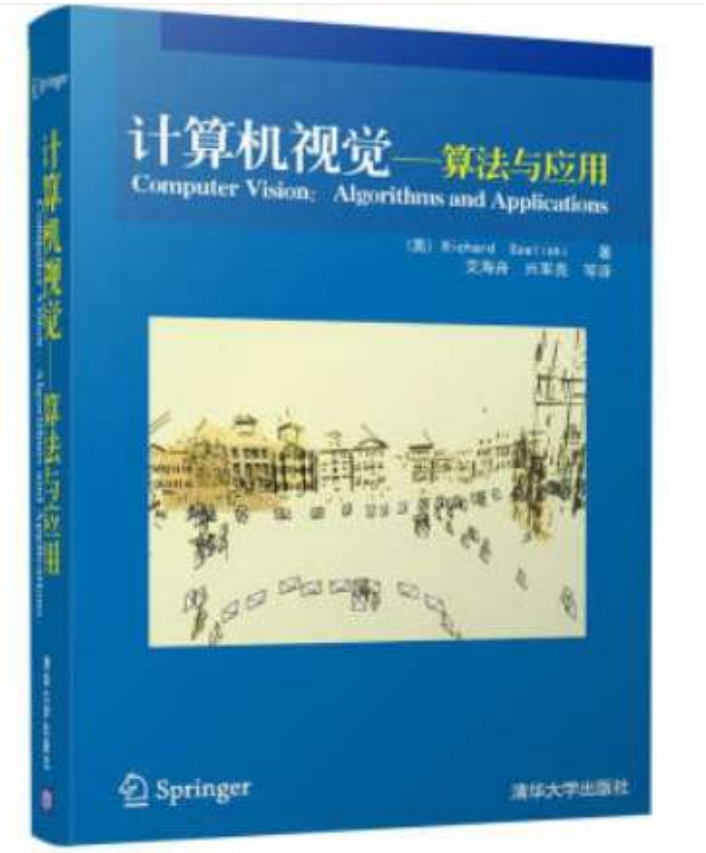
计算机视觉
Computer Vision

-- Introduction

钟 凡
zhongfan@sdu.edu.cn

参考书

- 计算机视觉—算法与应用
  - 【作者】**Richard Szeliski**
  - 【出 版 社】 清华大学出版社



课程任务

- 理解计算机视觉基本问题
- 掌握计算机视觉基本原理与方法；
- 学会使用**C++/OpenCV**编程解决问题；

考试？

- 期末考试：**50%**
- 实验+课堂：**50%**

联系方式

- **QQ群： 372125283**
- 作业提交邮箱：**cvxfan@163.com**
- 电话：**18754125732**

特别感谢！

Prof. Dr. Bastian Leibe
Room 124
Phone: +49 241 80 20 762
Fax: +49 241 80 22731
Email: leibe@vision.rwth-aachen.de

人工智能？

人工智能

- 用计算机模拟人的智能

人工智能

- 终极目标？

人工智能

- 人

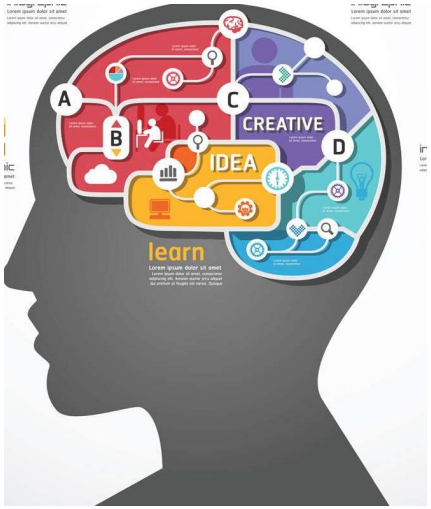
运动

触觉

听觉　　味觉

视觉　　嗅觉

感知



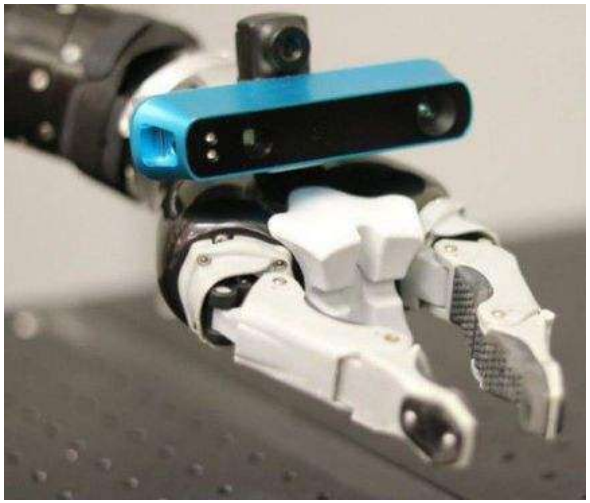
思维、情感、欲望

视觉信息占人接收信息的80%以上

人工智能

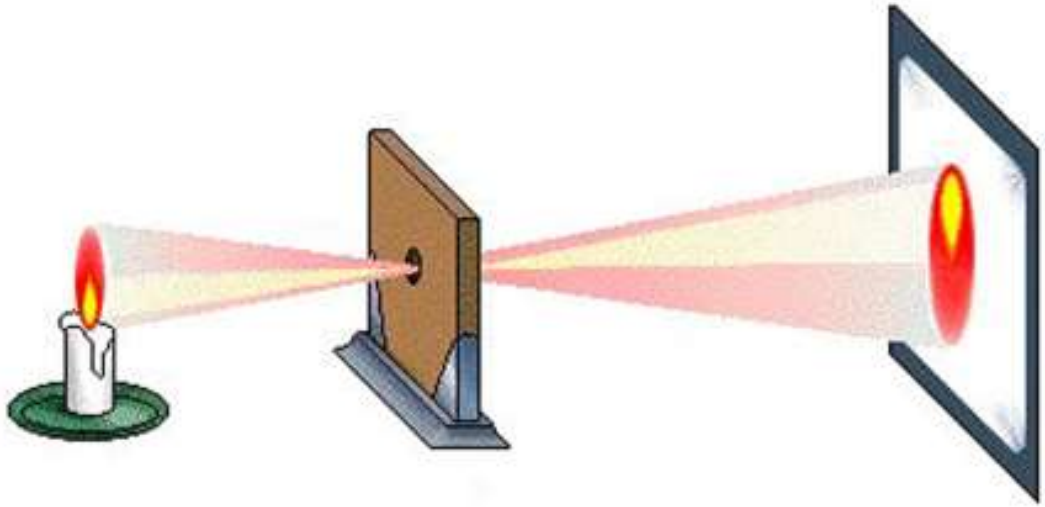
- 机器人学
- 计算机视觉
- 自然语言处理
- 认知与推理
- 博弈与伦理
- ……

视觉—机器之眼



视觉=图像获取？

图像：对客观世界视觉属性的描述

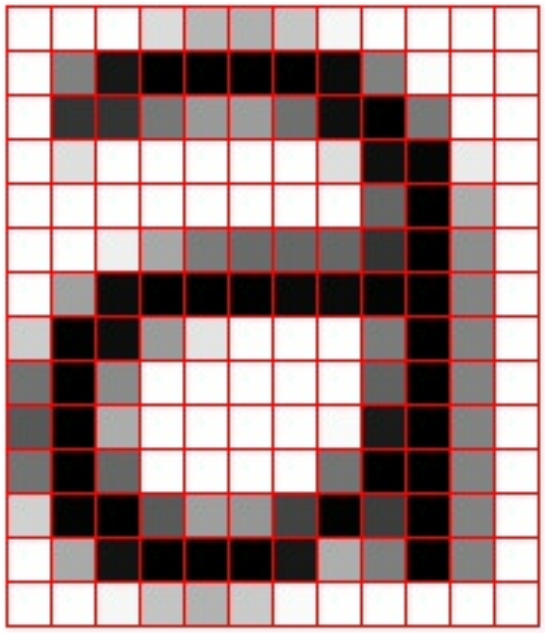


形　　像

数字图像

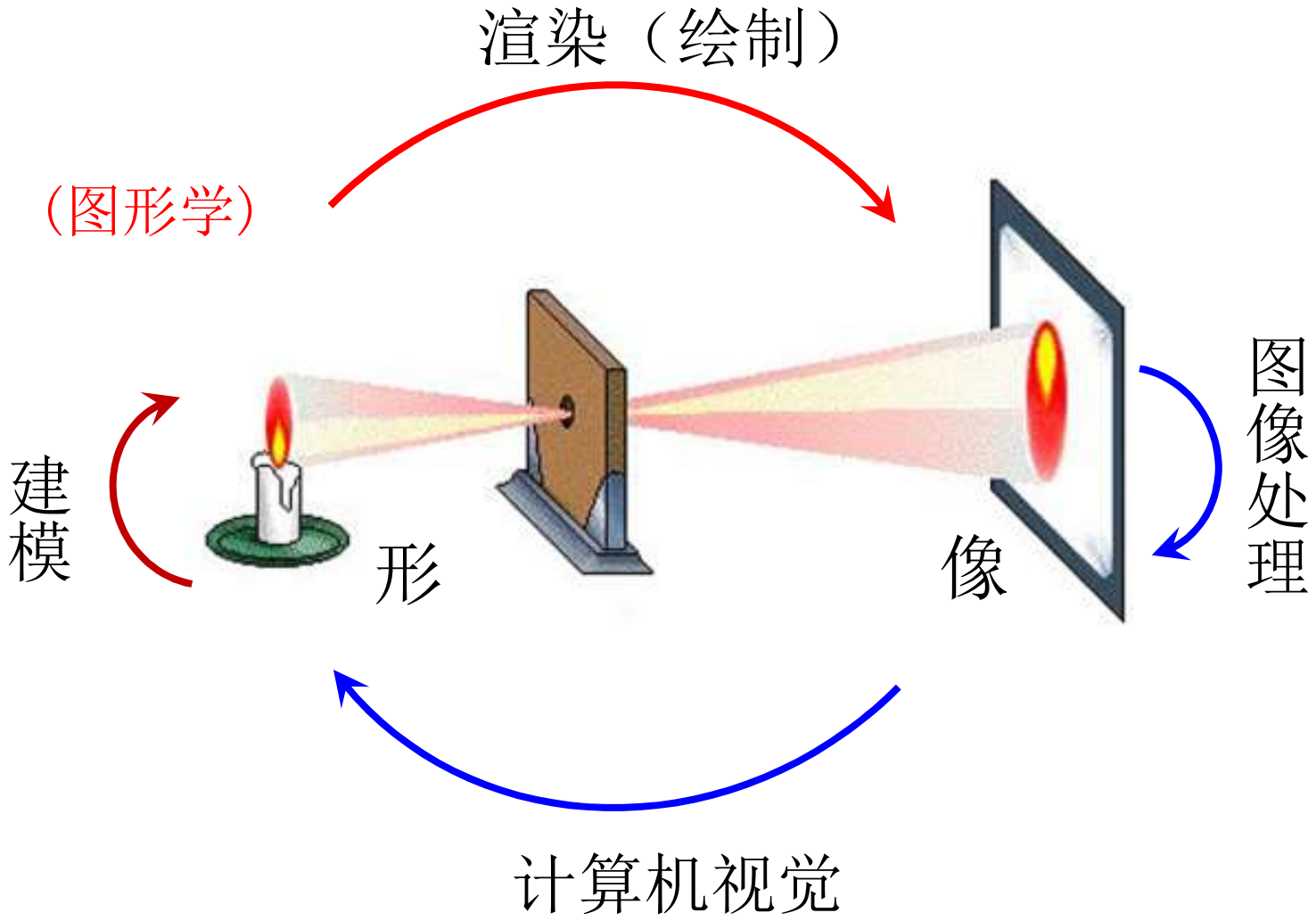
- 二维的像素阵列

1.0	1.0	1.0	0.9	0.6	0.6	0.6	1.0	1.0	1.0	1.0	1.0
1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0	1.0	1.0
1.0	0.2	0.2	0.5	0.6	0.6	0.5	0.0	0.0	0.5	1.0	1.0
1.0	0.9	1.0	1.0	1.0	1.0	1.0	0.9	0.0	0.0	0.9	1.0
1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
1.0	1.0	1.0	0.5	0.5	0.5	0.5	0.5	0.4	0.0	0.5	1.0
1.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0
0.9	0.0	0.0	0.6	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
0.5	0.0	0.6	1.0	1.0	1.0	1.0	1.0	0.5	0.0	0.5	1.0
0.5	0.0	0.7	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.5	1.0
0.6	0.0	0.6	1.0	1.0	1.0	1.0	0.5	0.0	0.0	0.5	1.0
0.9	0.1	0.0	0.6	0.7	0.7	0.5	0.0	0.5	0.0	0.5	1.0
1.0	0.7	0.1	0.0	0.0	0.0	0.1	0.9	0.8	0.0	0.5	1.0
1.0	1.0	1.0	0.8	0.8	0.9	1.0	1.0	1.0	1.0	1.0	1.0



视觉 >> 图像获取

视觉—基于图像的感知


渲染（绘制）
(图形学)
建模
形
像
图像处理
计算机视觉

What is Computer Vision?

- **Goal of Computer Vision**
  - ***Enable a machine to "understand" images and videos***

- **Automatic understanding**
  - **Computing properties of the 3D world from visual data *(measurement)***
  - **Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities. *(perception and interpretation)***

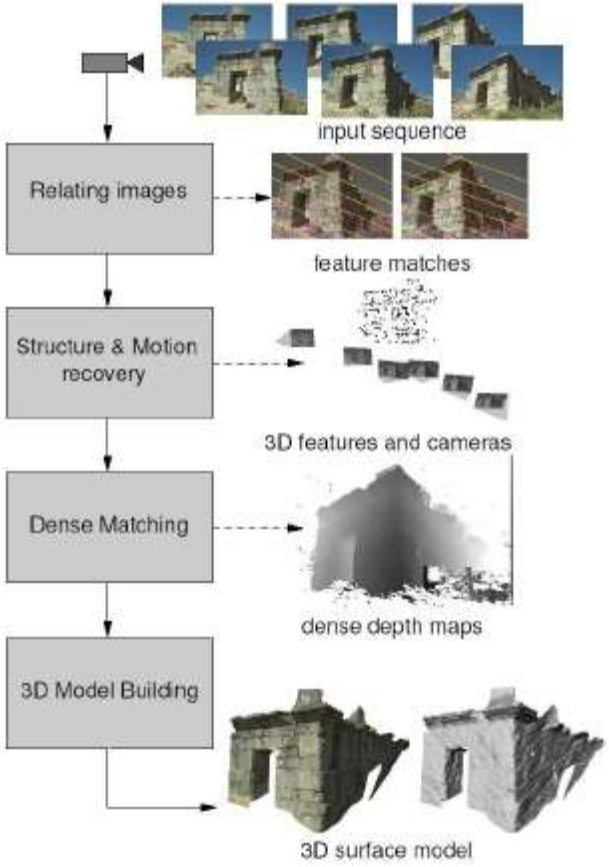
Slide credit: Kristen Grauman

Vision for Measurement

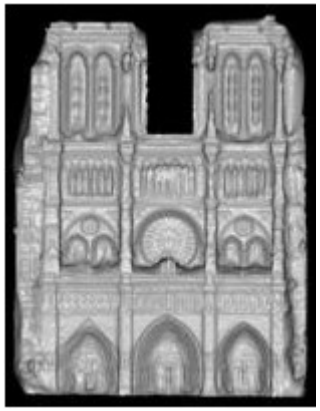
Real-time stereo

Pollefeys et al.

Structure from motion



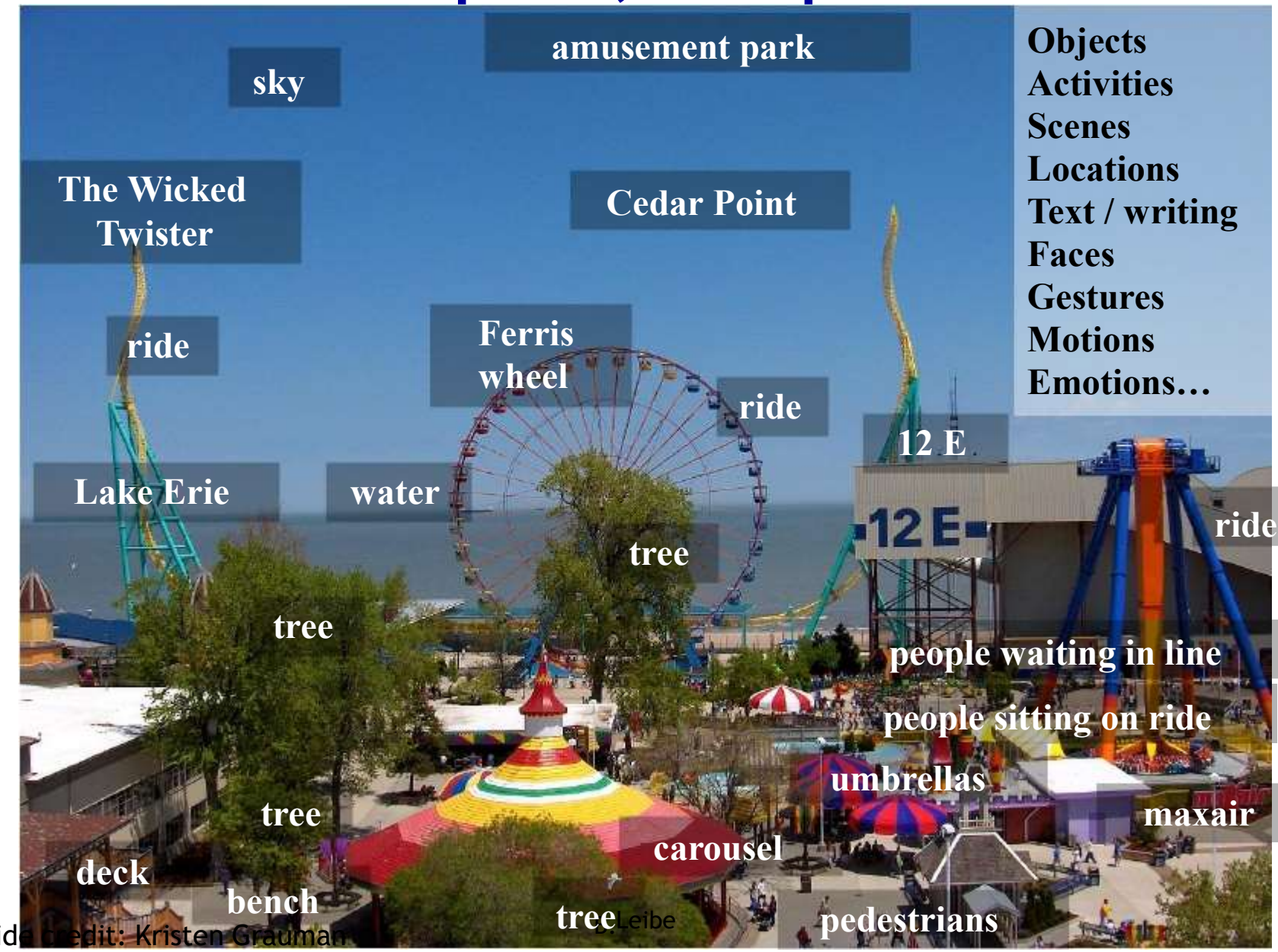
Multi-view stereo for community photo collections



Goesele et al.

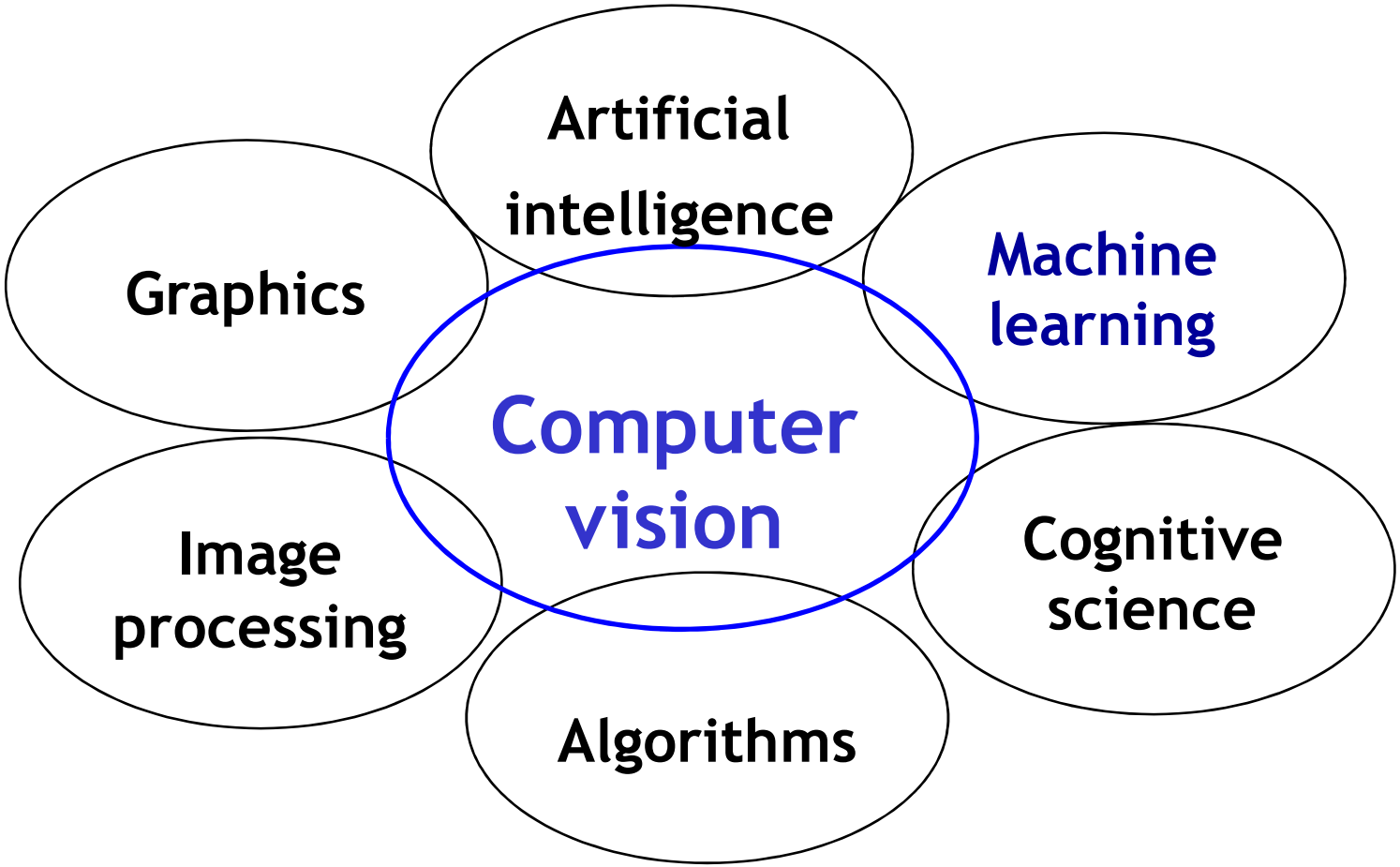
Slide credit: Svetlana Lazebnik

Vision for Perception, Interpretation



Slide credit: Kristen Grauman

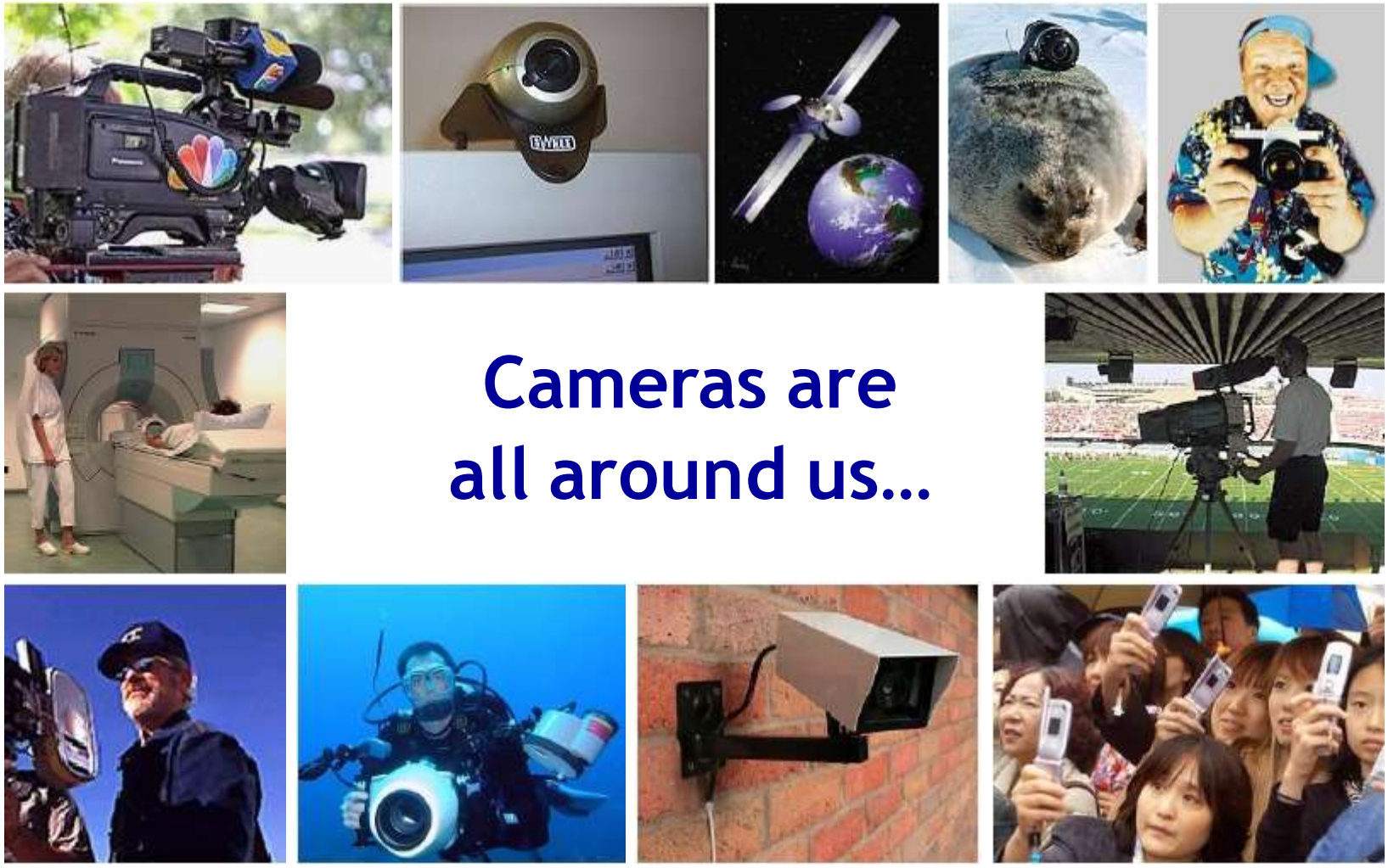
Related Disciplines



Directions to Computer Vision

- Science
  - Foundations of perception. How do WE see?
- Engineering
  - How do we build systems that perceive the world?
- Many applications
  - Medical imaging, surveillance, entertainment, graphics, …

应用



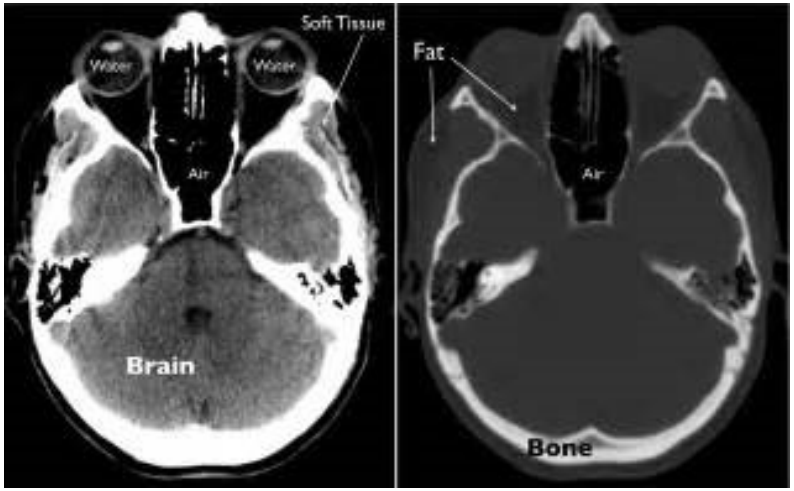
Slide credit: Kristen Grauman

智能监控

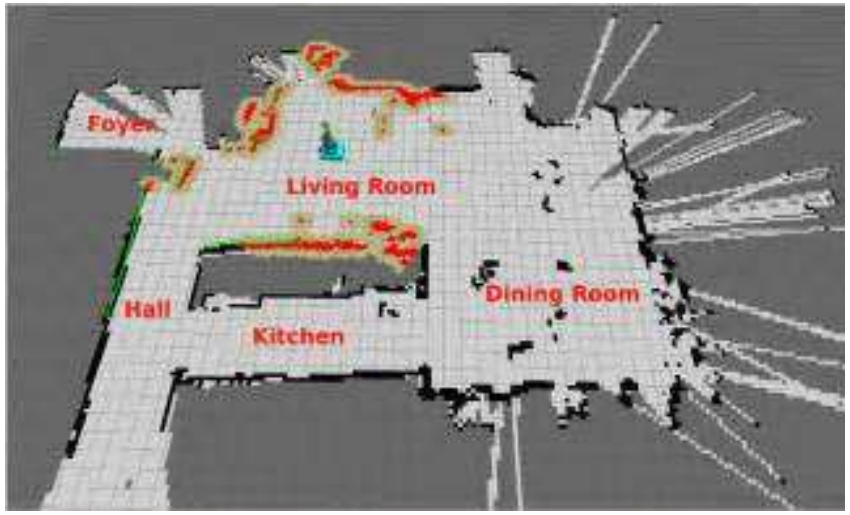
无人车

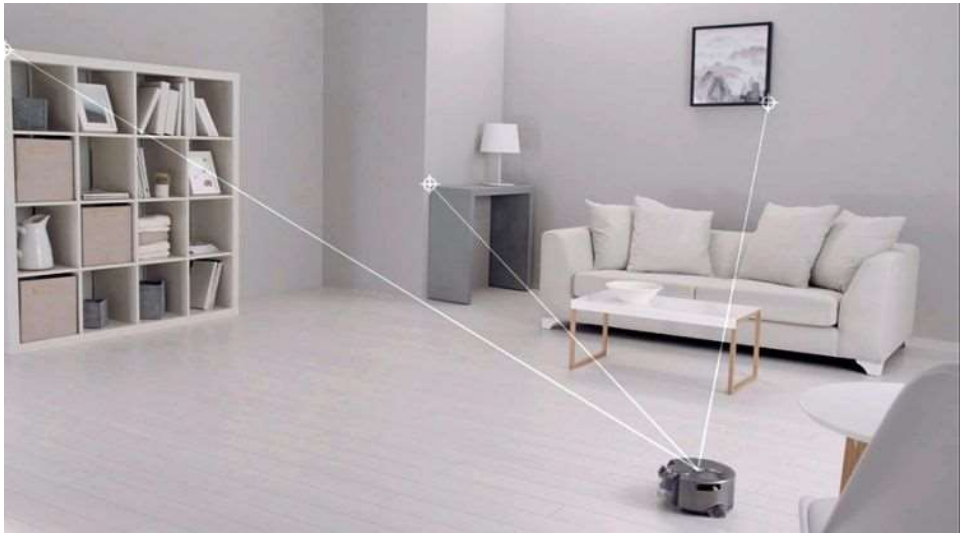

WAYMO 01
RIDERS
IN-CAR DISPLAY
AERIAL VIEW
Velodyne laser scanner
GPS
cameras
odometer
radar
lux laser scanner
AUTONOMOS

医学影像分析



机器人（定位、导航）

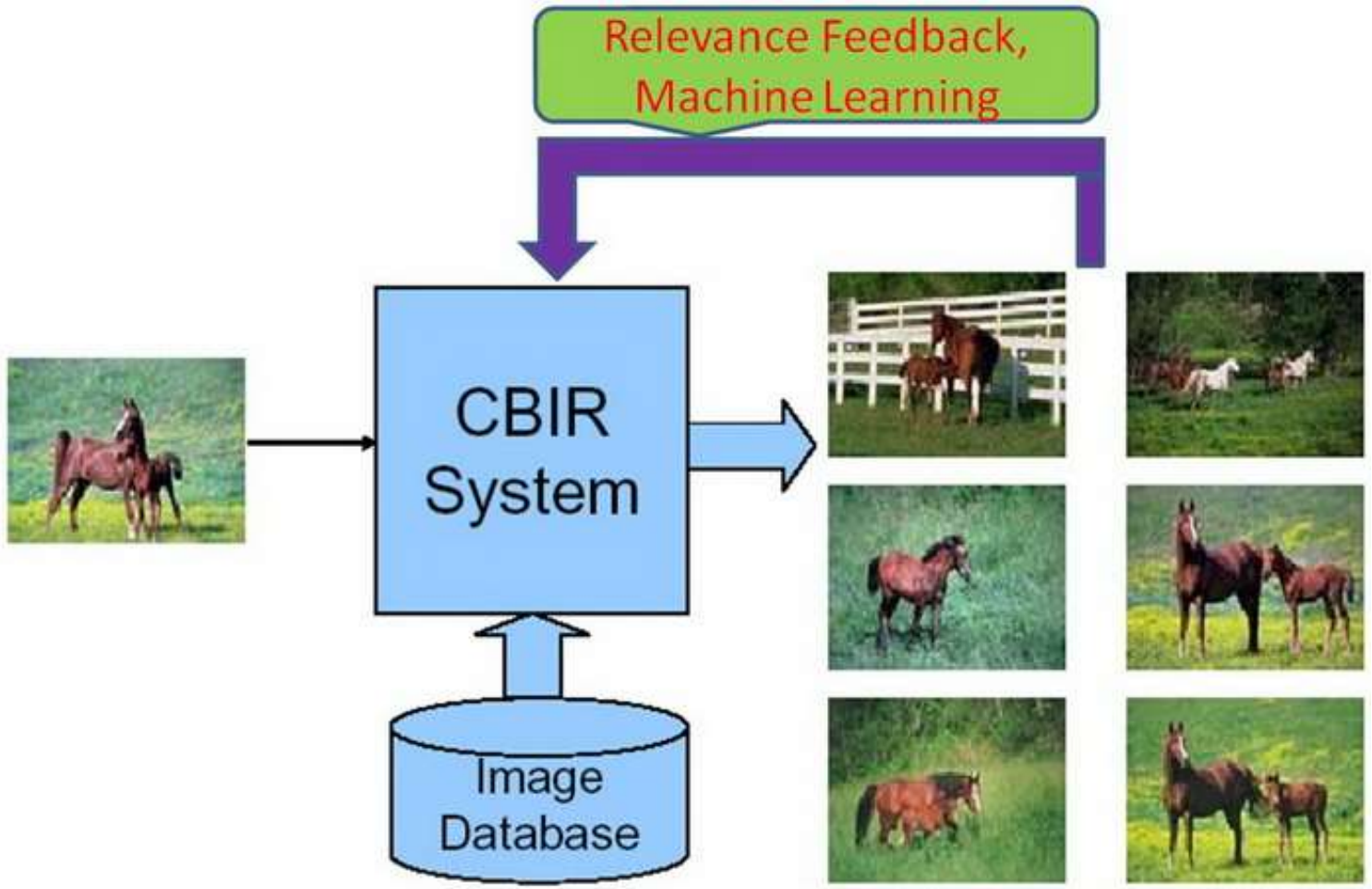




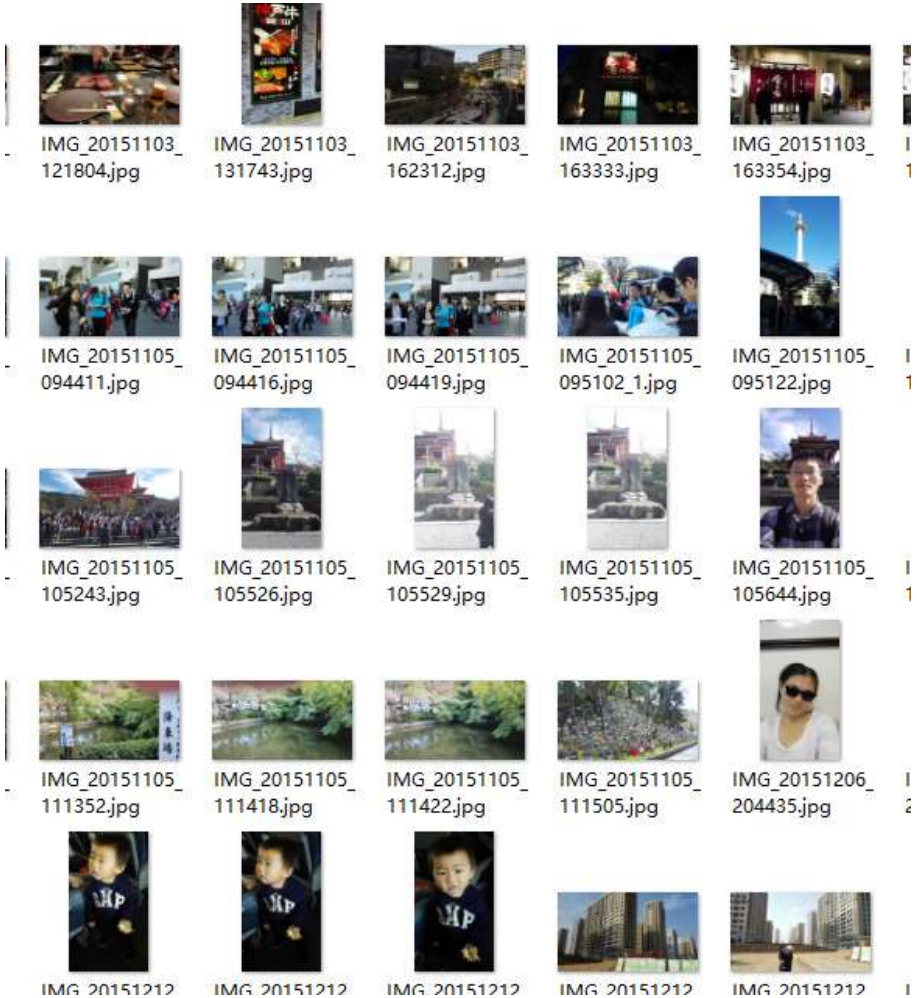
增强/混合现实


Kings Place, N1 9GU
Jazz here from 7pm today
Network Rail
The Observer
King's Cross
300m
expand page?
See menu
See reviews
Taxi - AVAILABLE

图像检索


Relevance Feedback,
Machine Learning
CBIR
System
Image
Database

智能图像管理/编辑


IMG_20151103_121804.jpg
IMG_20151103_131743.jpg
IMG_20151103_162312.jpg
IMG_20151103_163333.jpg
IMG_20151103_163354.jpg
IMG_20151105_094411.jpg
IMG_20151105_094416.jpg
IMG_20151105_094419.jpg
IMG_20151105_095102_1.jpg
IMG_20151105_095122.jpg
IMG_20151105_105243.jpg
IMG_20151105_105526.jpg
IMG_20151105_105529.jpg
IMG_20151105_105535.jpg
IMG_20151105_105644.jpg
IMG_20151105_111352.jpg
IMG_20151105_111418.jpg
IMG_20151105_111422.jpg
IMG_20151105_111505.jpg
IMG_20151206_204435.jpg


原图
Peach

刷脸

Applications: Faces and Digital Cameras

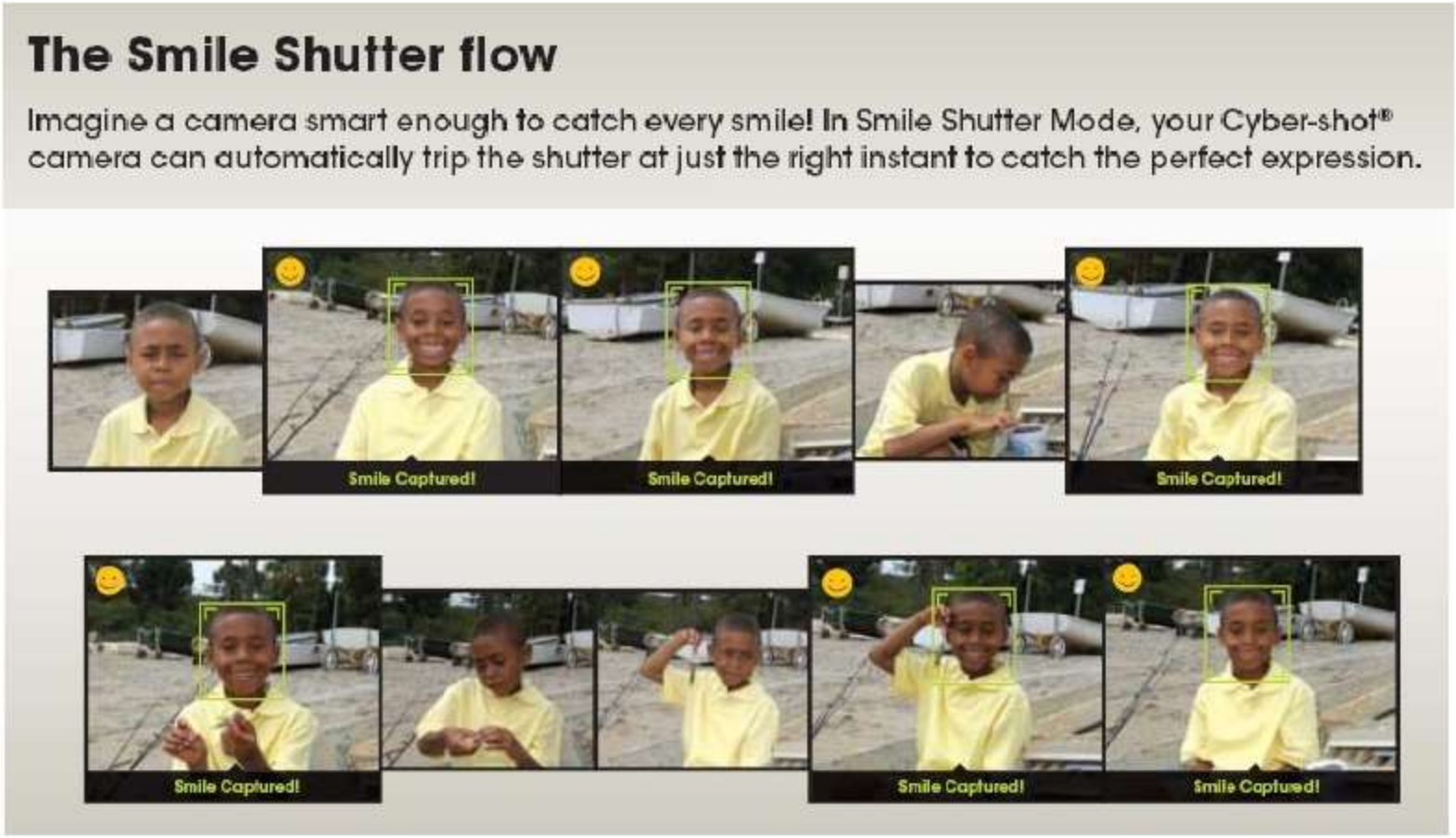
Setting camera focus via face detection

Camera waits for everyone to smile to take a photo [Canon]

Automatic lighting correction based on face detection

Slide credit: Kristen Grauman, Rob Fergus

Smile detection



Sony Cyber-shot® T70 Digital Still Camera

Slide credit: Svetlana Lazebnik, Steve Seitz

Matching

- **Stitch your photos together to create panoramas**

AUTOSTITCH

Available on the iPhone App Store

Applications: Vision-based Interfaces

Games
(Microsoft Kinect)

Assistive technology systems
Camera Mouse
Boston College

Applications: Visual Special Effects

The Matrix

MoCap for *Pirates of the Carribean*, Industrial Light and Magic

Slide adapted from Svetlana Lazebnik, Kristen Grauman

Ok, Let's Do It - Any Obstacles?

- 1966: Seymour Papert directs an undergraduate student to solve "the problem of computer vision" as a summer project.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group — July 7, 1966
Vision Memo. No. 100.

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

- Obviously, computer vision was too difficult for that…

Challenges: Many Nuisance Parameters

Illumination

Object pose

Clutter

Occlusions

Intra-class appearance

Viewpoint

Slide credit: Kristen Grauman

Challenges: Intra-Category Variation

Slide credit: Fergus, FeiFei, Torralba

B. Leibe

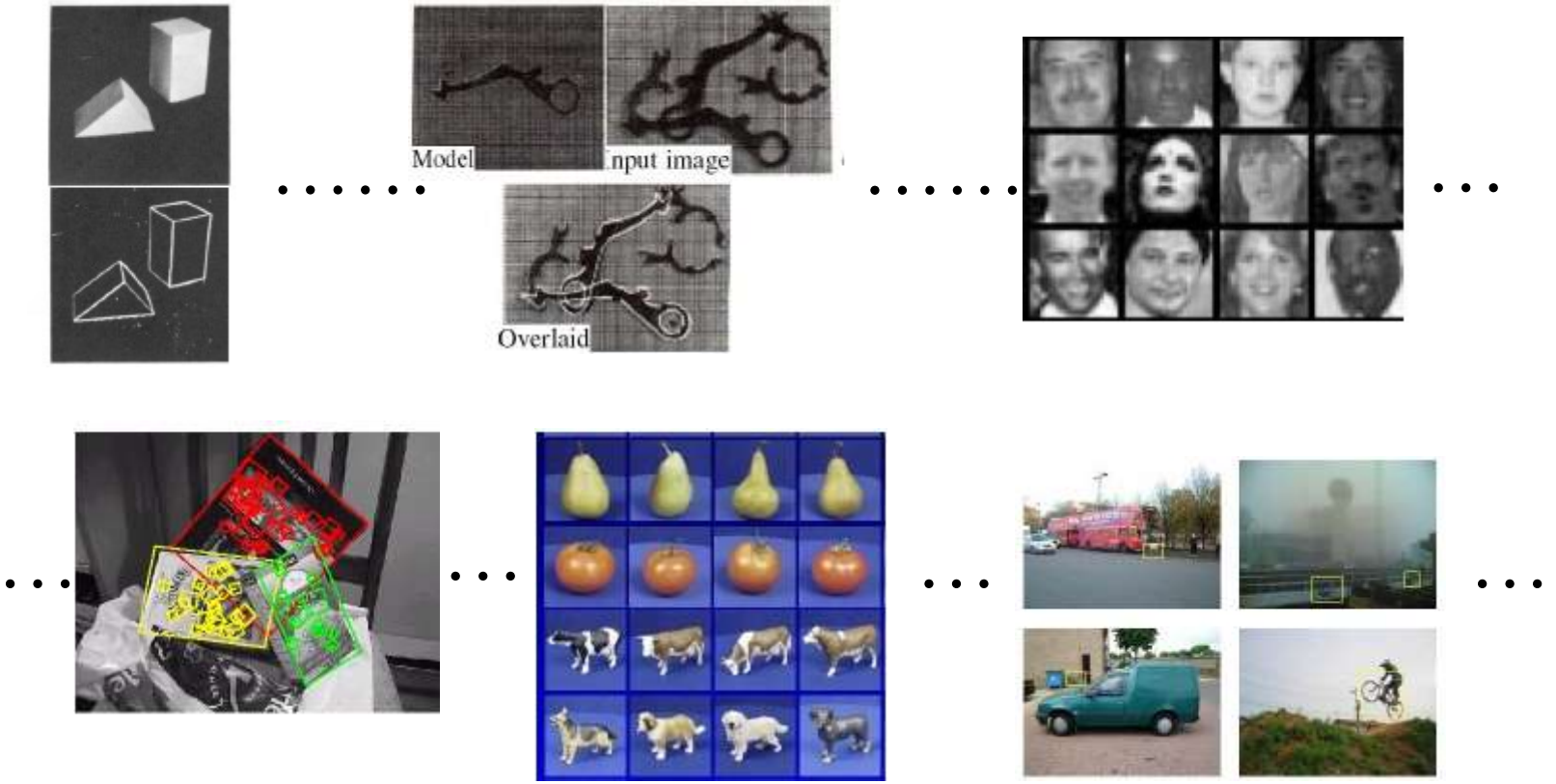
Challenges: Complexity

- Thousands to millions of pixels in an image
- 3,000-30,000 human recognizable object categories
- 30+ degrees of freedom in the pose of articulated objects (humans)
- Billions of images indexed by Google Image Search
- 18 billion+ prints produced from digital camera images in 2004
- 295.5 million camera phones sold in 2005
- About half of the cerebral cortex in primates is devoted to processing visual information [Felleman and van Essen 1991].

Slide credit: Kristen Grauman

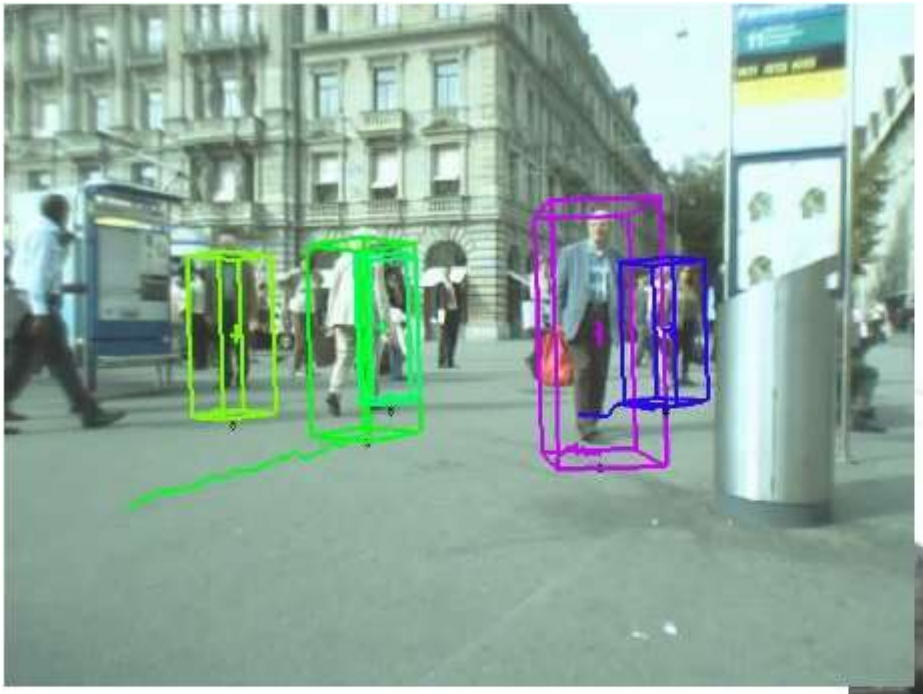
So, Should We Give Up?

- **NO! Very active research area with exciting progress!**



Slide credit: Kristen Grauman

Things Are Starting to Work…



- **Computer Vision in realistic scenarios is becoming feasible!**

[Ess, Leibe, Schindler, Van Gool, 2008]

人眼的视觉原理与特征

摄像机：*人眼*

- **Wandell. Foundations of vision. 1995**
  - 视野覆盖宽160°、高135°的区域
  - 也存在几何和色彩像差

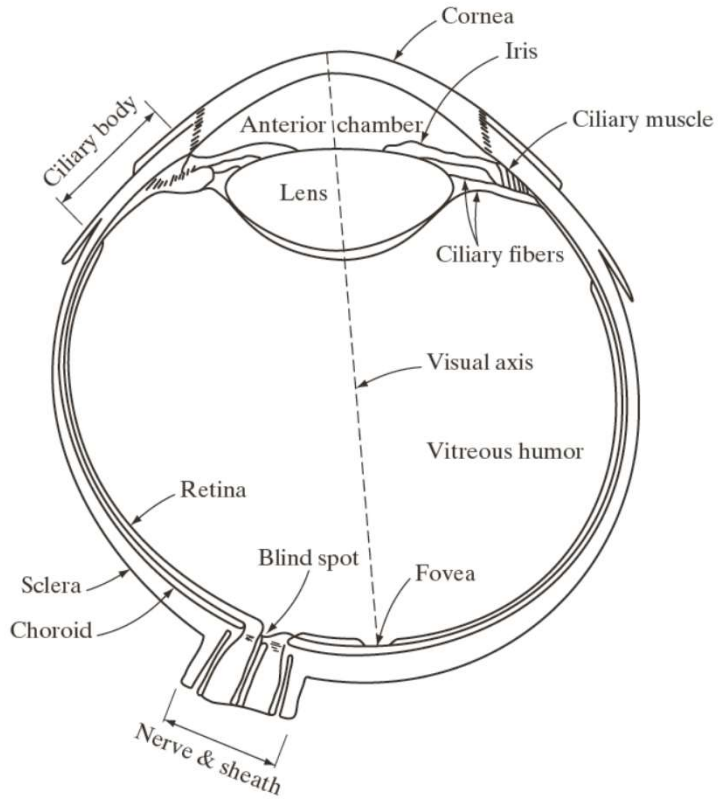
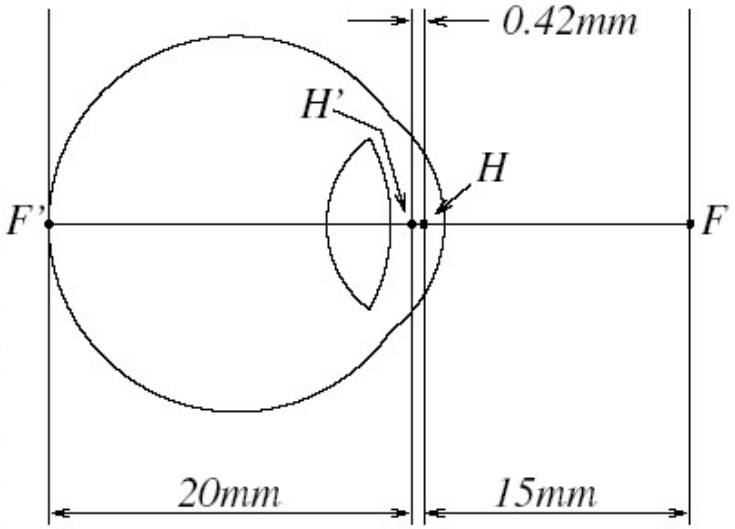


FIGURE 2.1
Simplified diagram of a cross section of the human eye.



人眼的成像

- 物体在视网膜上的成像为倒影
- 视网膜位于球面上

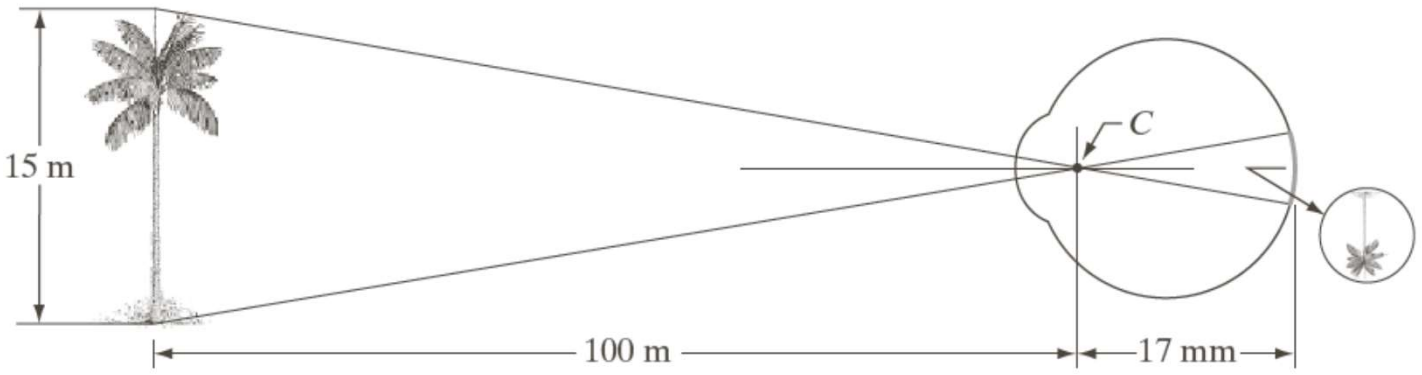


FIGURE 2.3 Graphical representation of the eye looking at a palm tree. Point *C* is the optical center of the lens.

人眼对光线的适应性

- 人眼对光线有很强的环境适应能力

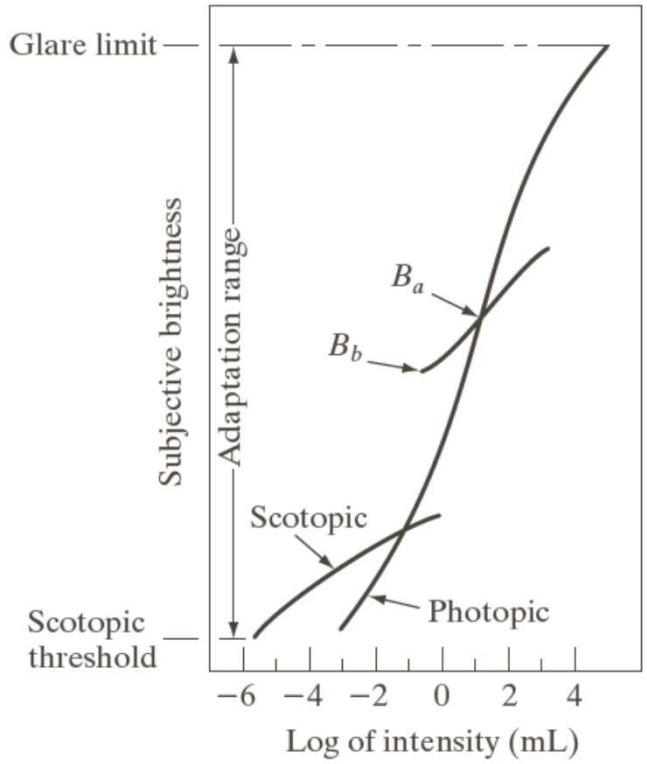


FIGURE 2.4
Range of subjective brightness sensations showing a particular adaptation level.

人眼对光亮度的感觉

- 人眼对亮度的感受与光线强度成对数关系

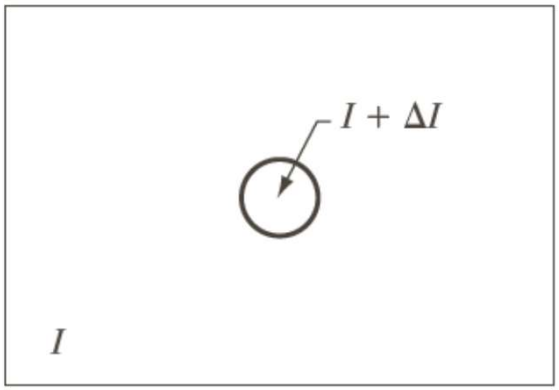


FIGURE 2.5 Basic experimental setup used to characterize brightness discrimination.

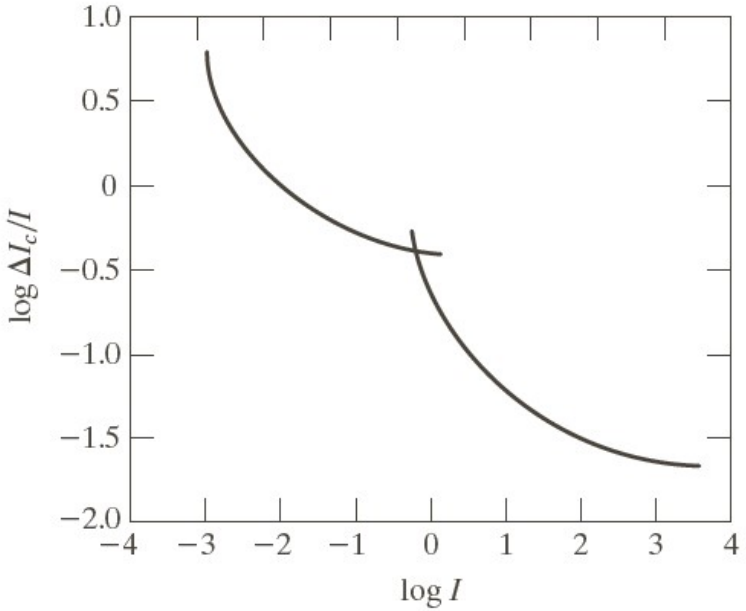
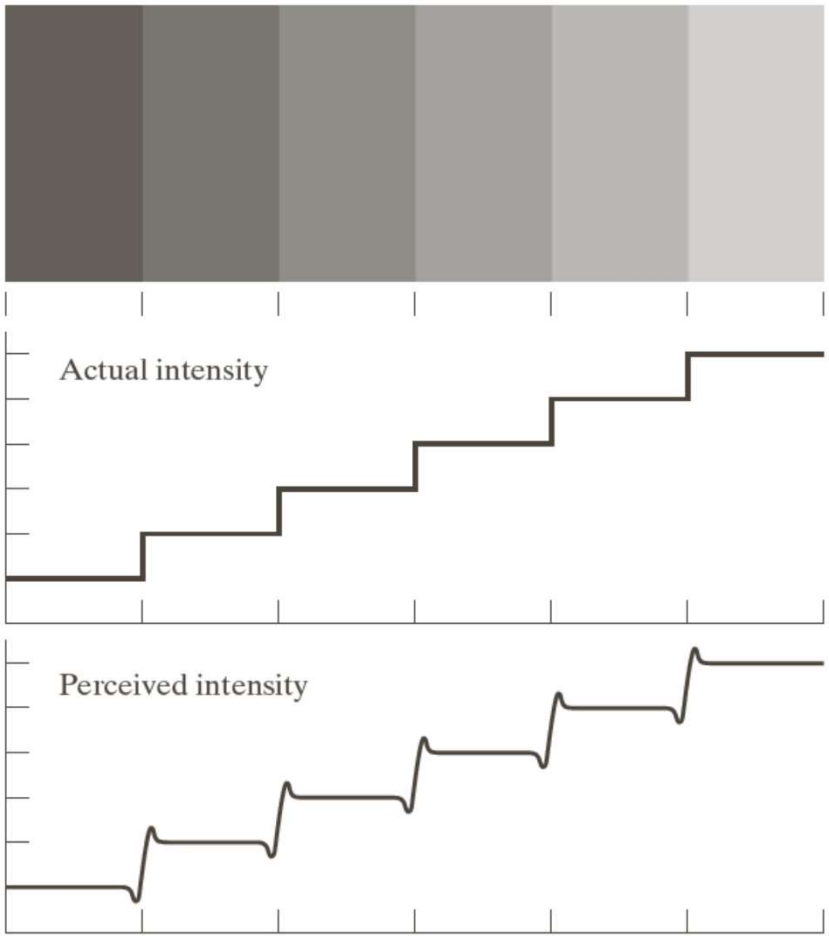


FIGURE 2.6 Typical Weber ratio as a function of intensity.

人眼对光亮度的感觉

- 马赫带

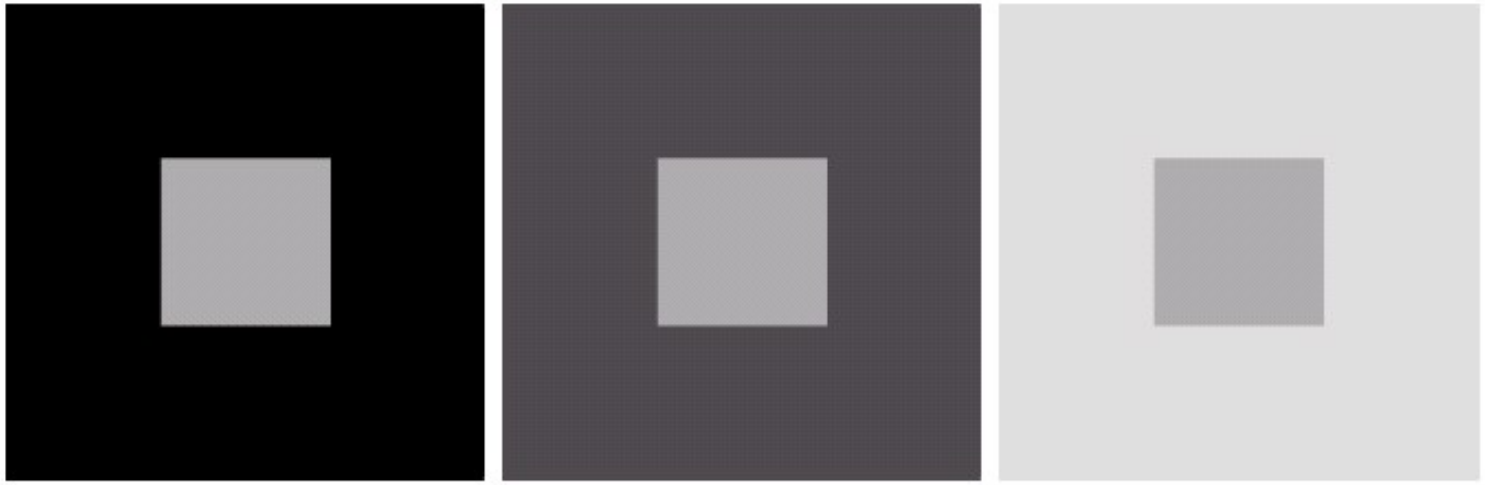


a
b
c

FIGURE 2.7
Illustration of the Mach band effect. Perceived intensity is not a simple function of actual intensity.

人眼对光亮度的感觉

- 亮度的相对性

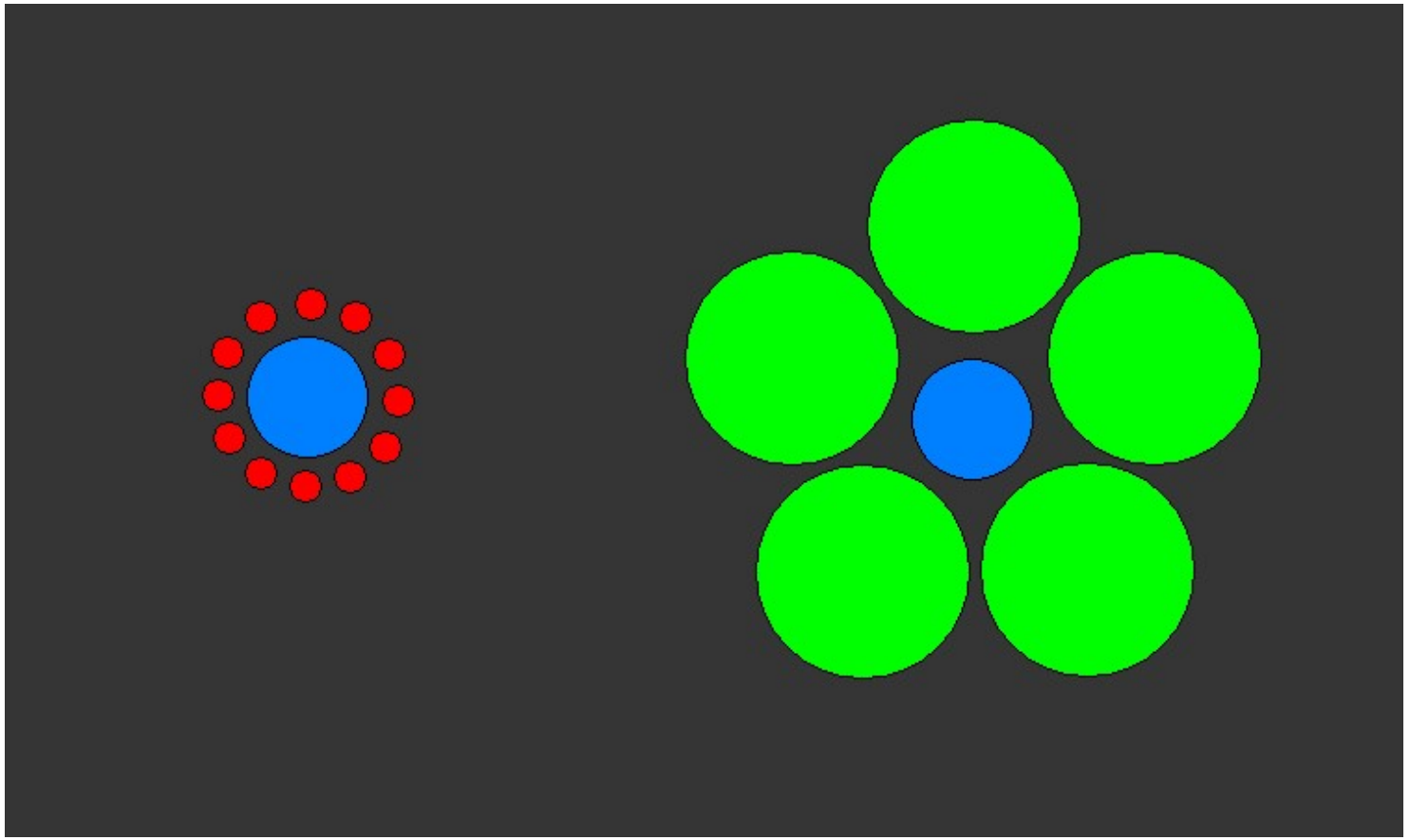


a b c

FIGURE 2.8 Examples of simultaneous contrast. All the inner squares have the same intensity, but they appear progressively darker as the background becomes lighter.

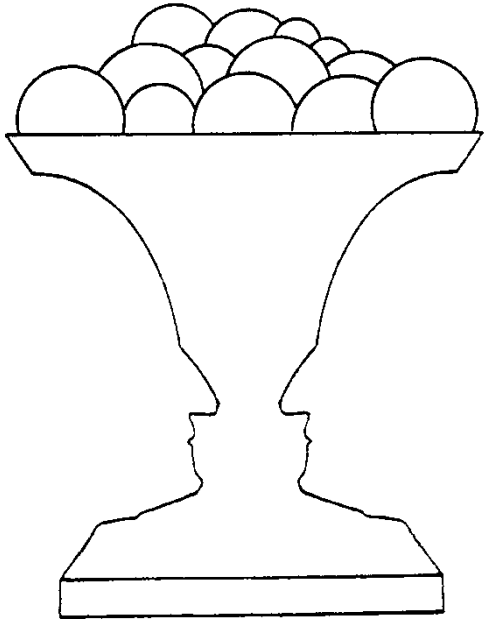
人眼的视觉特性

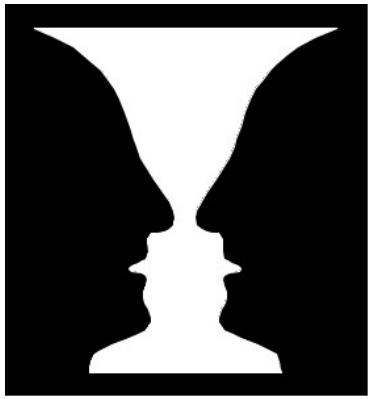
- 对比引起的相对性



人眼的视觉特性

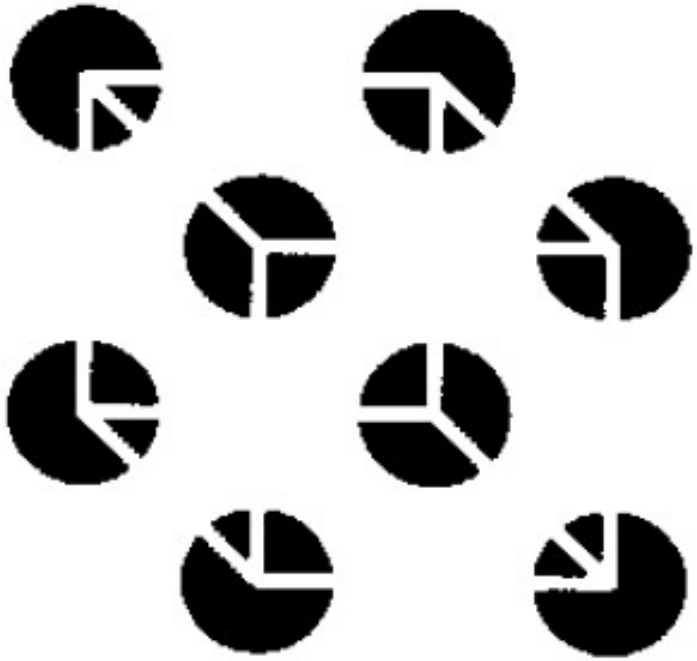
- 前景与背景的相对性





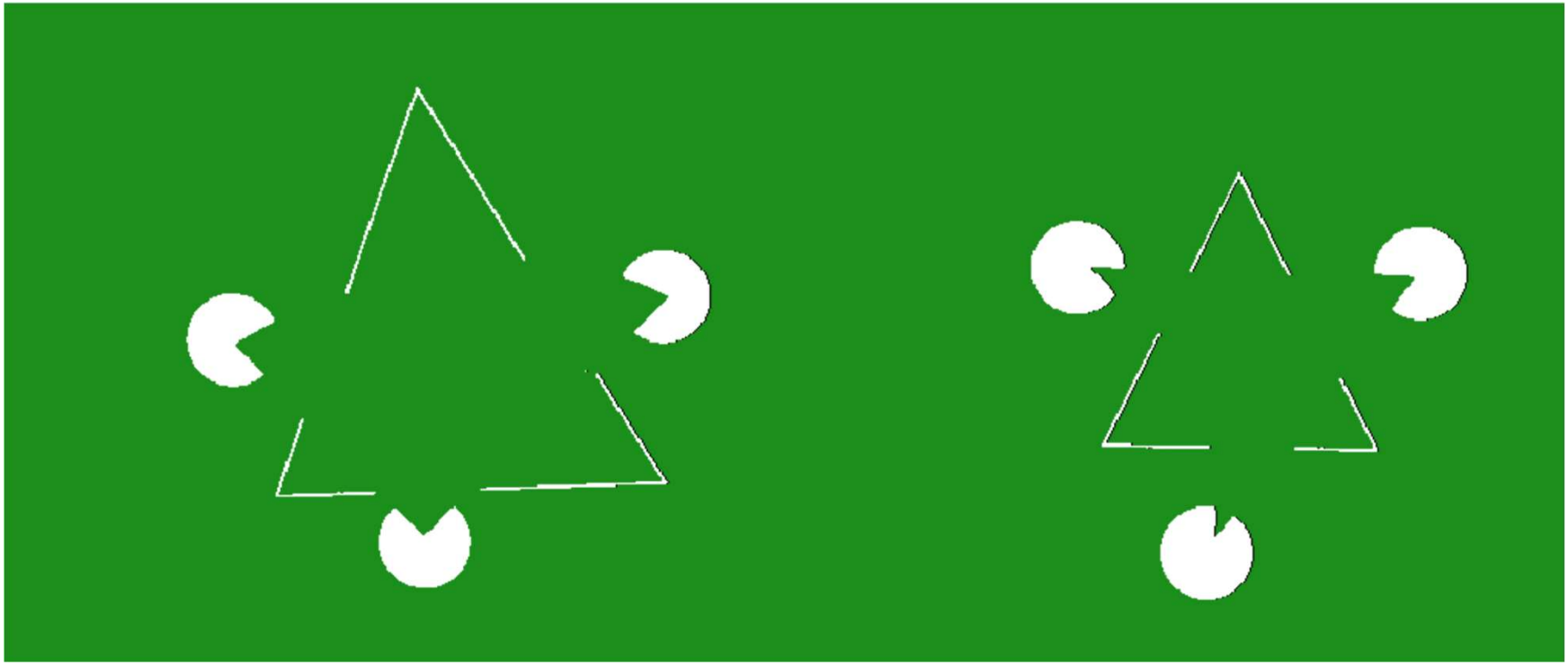
人眼的视觉特性

- 整体性：主观轮廓

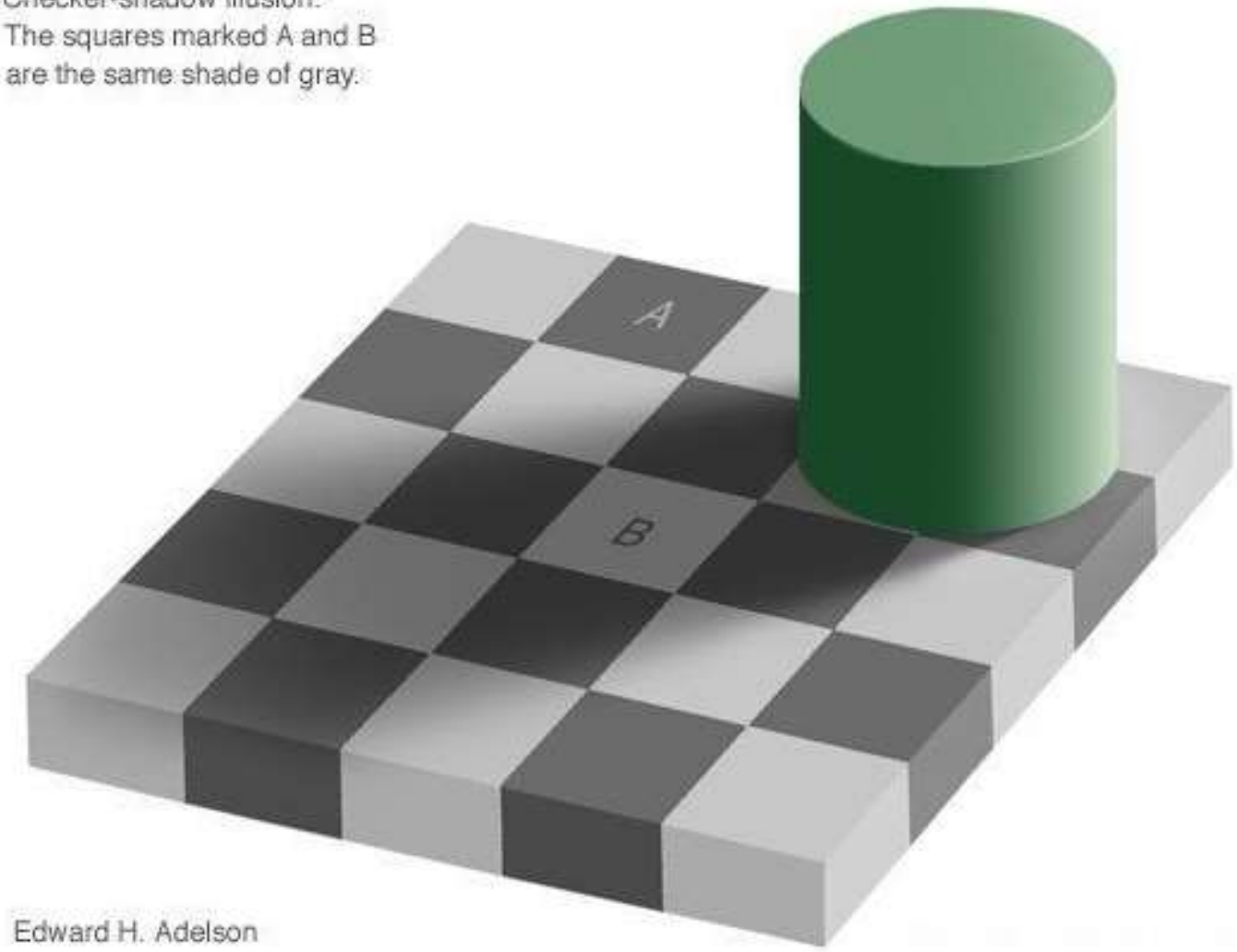


人眼的视觉特性

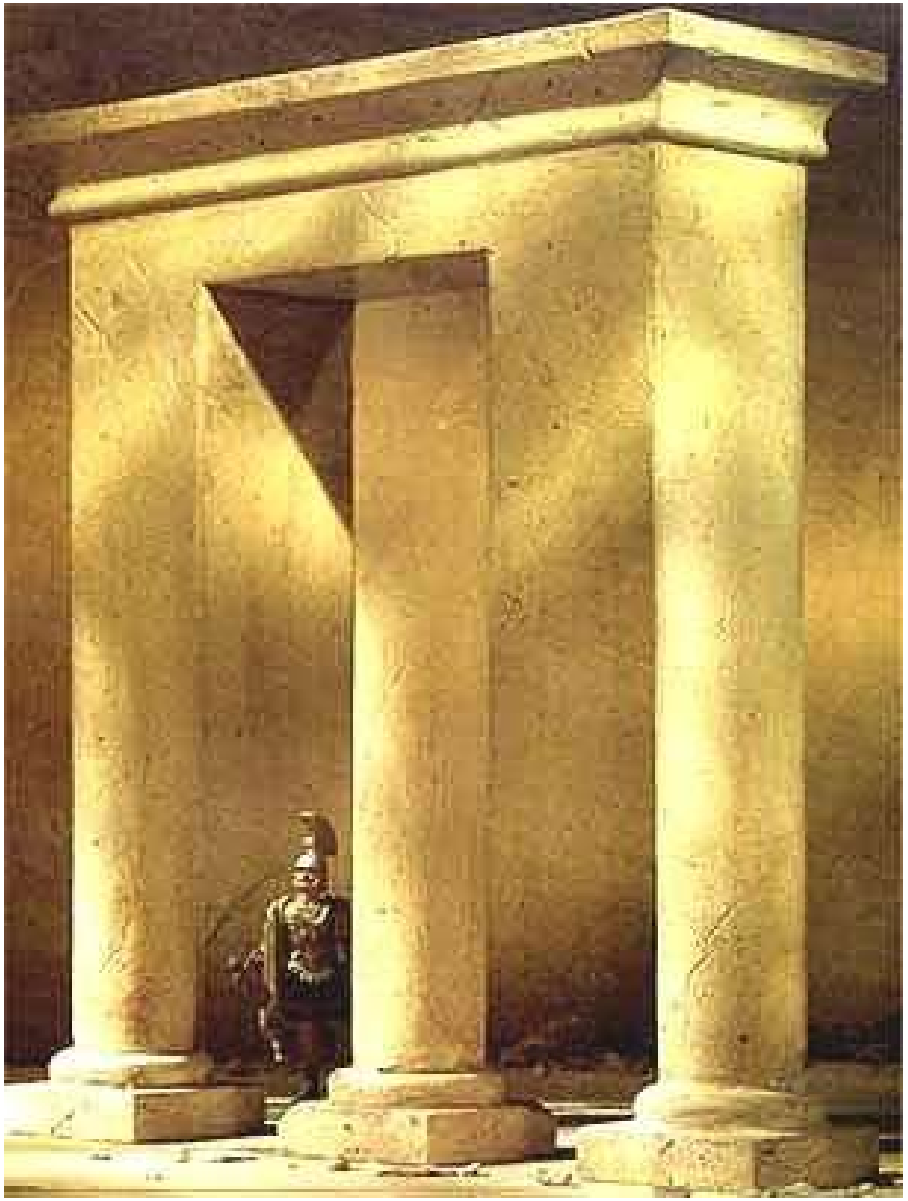
- 整体性：主观轮廓

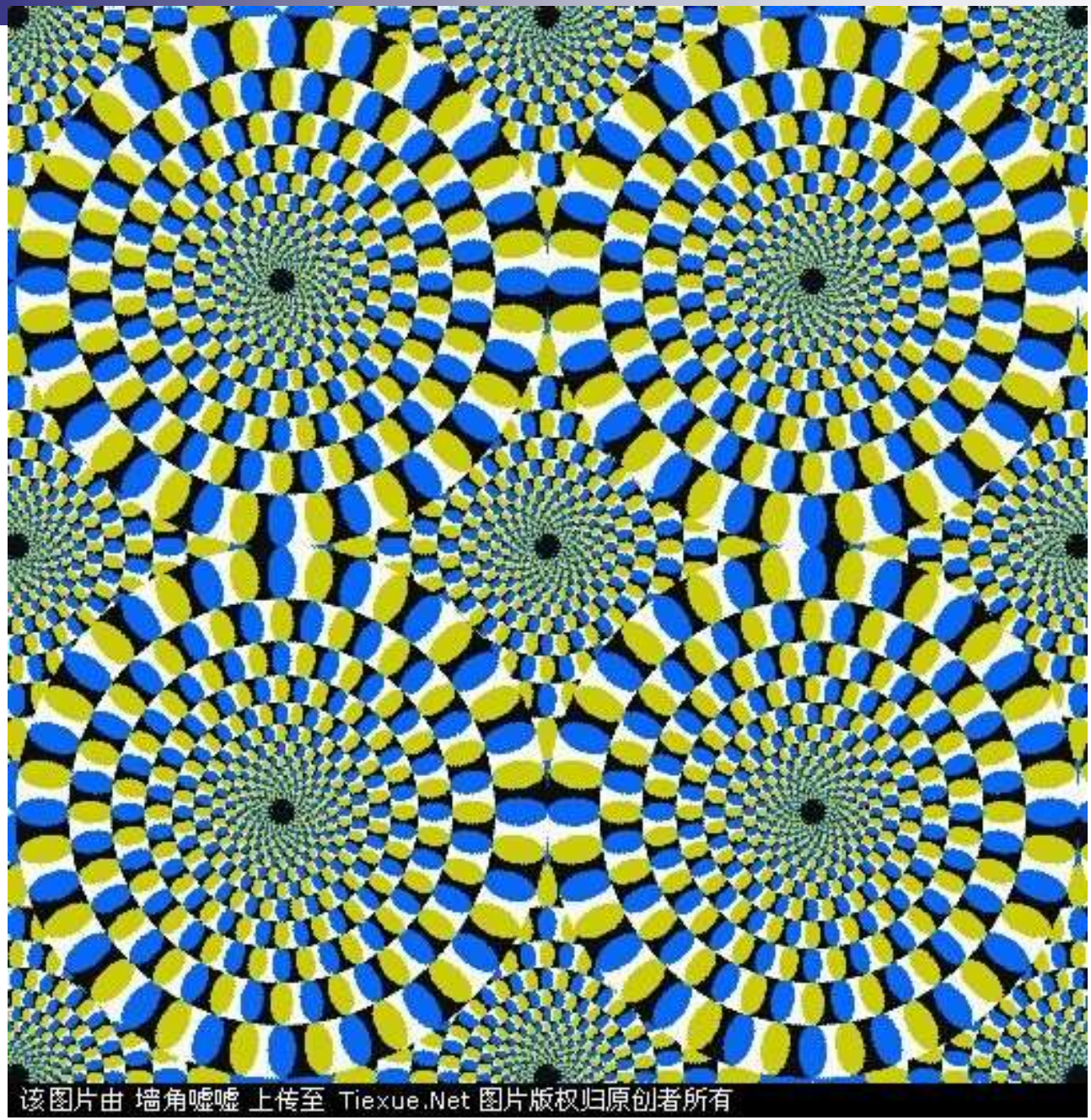


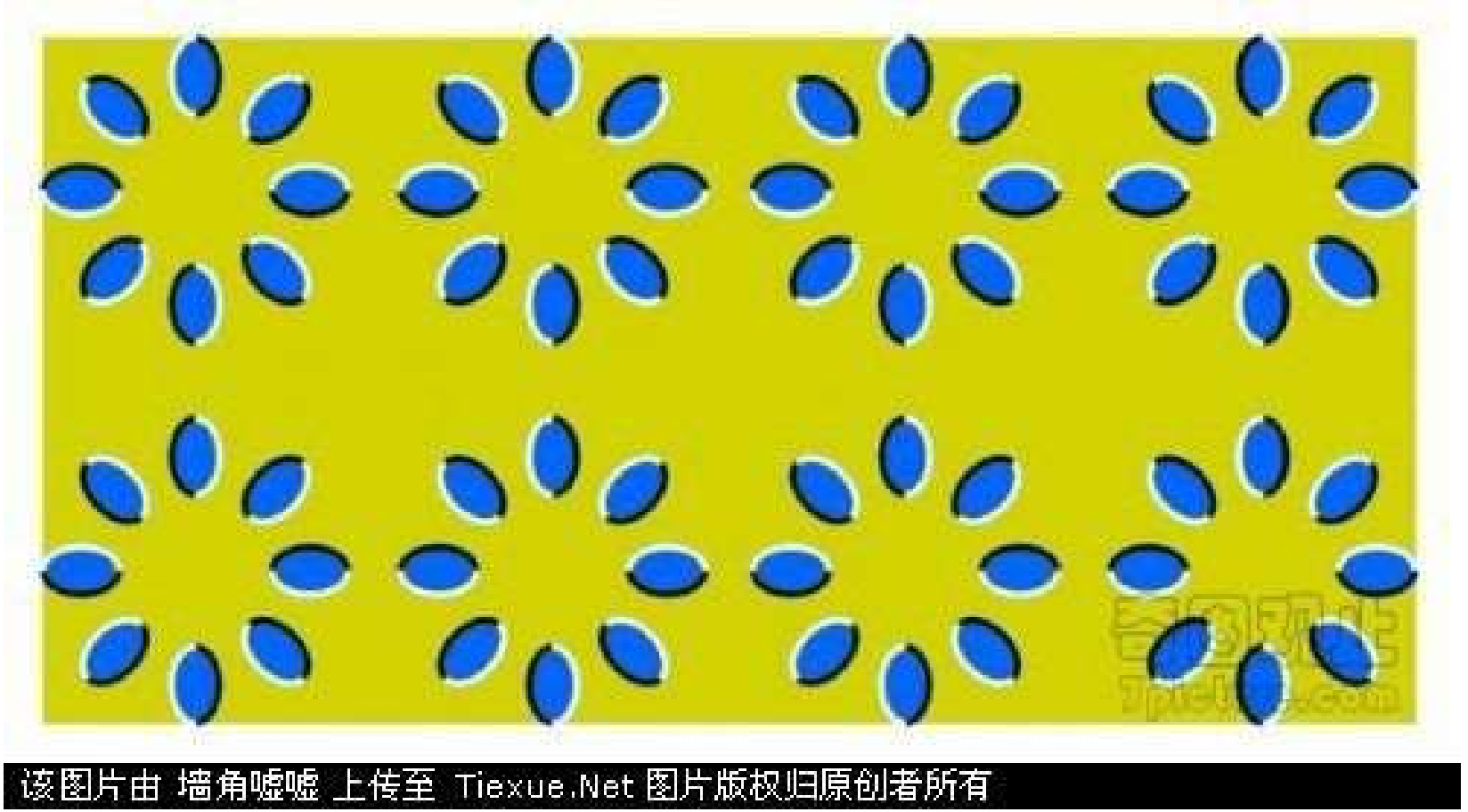
Checker-shadow illusion:
The squares marked A and B
are the same shade of gray.

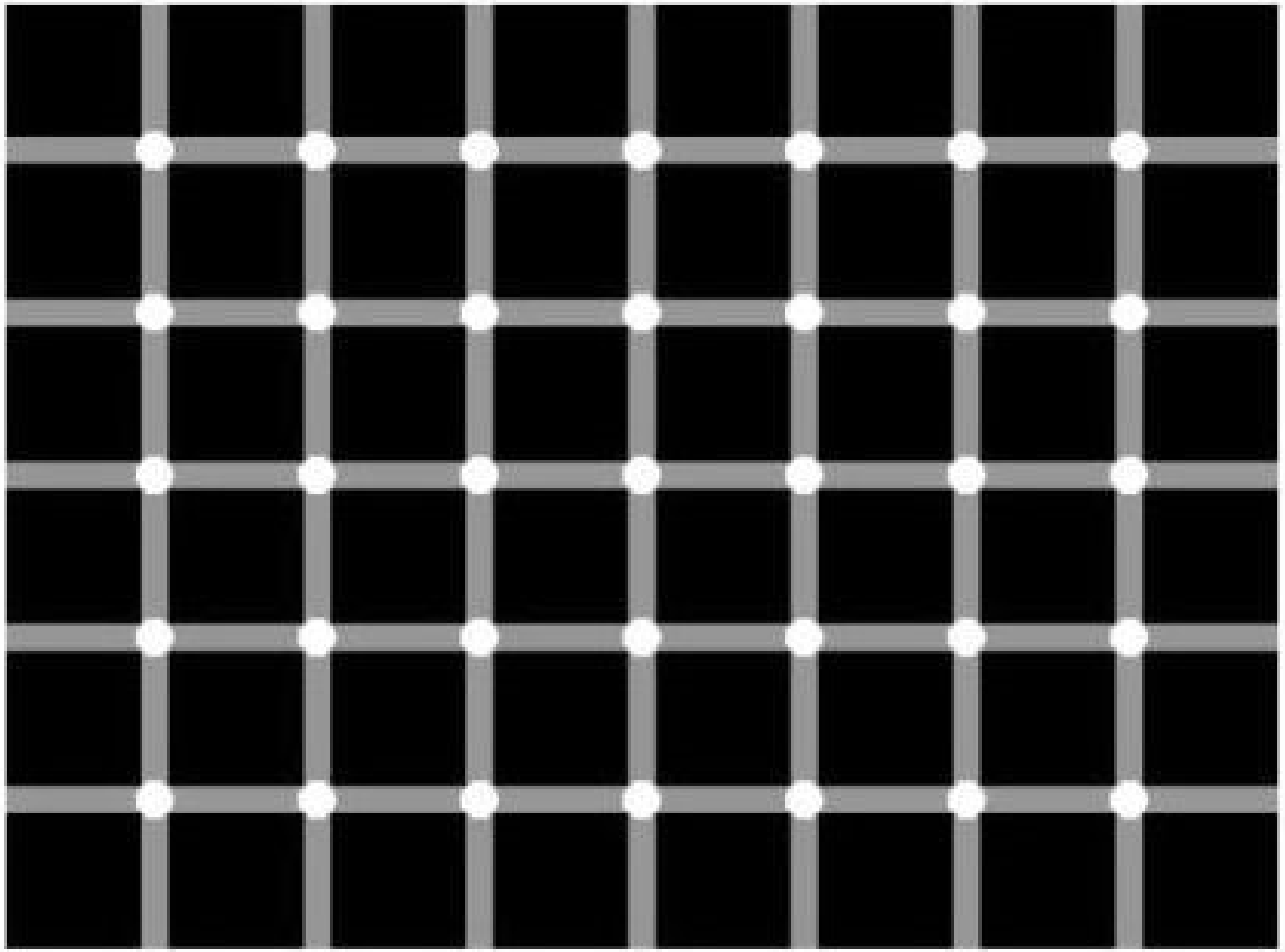


Edward H. Adelson

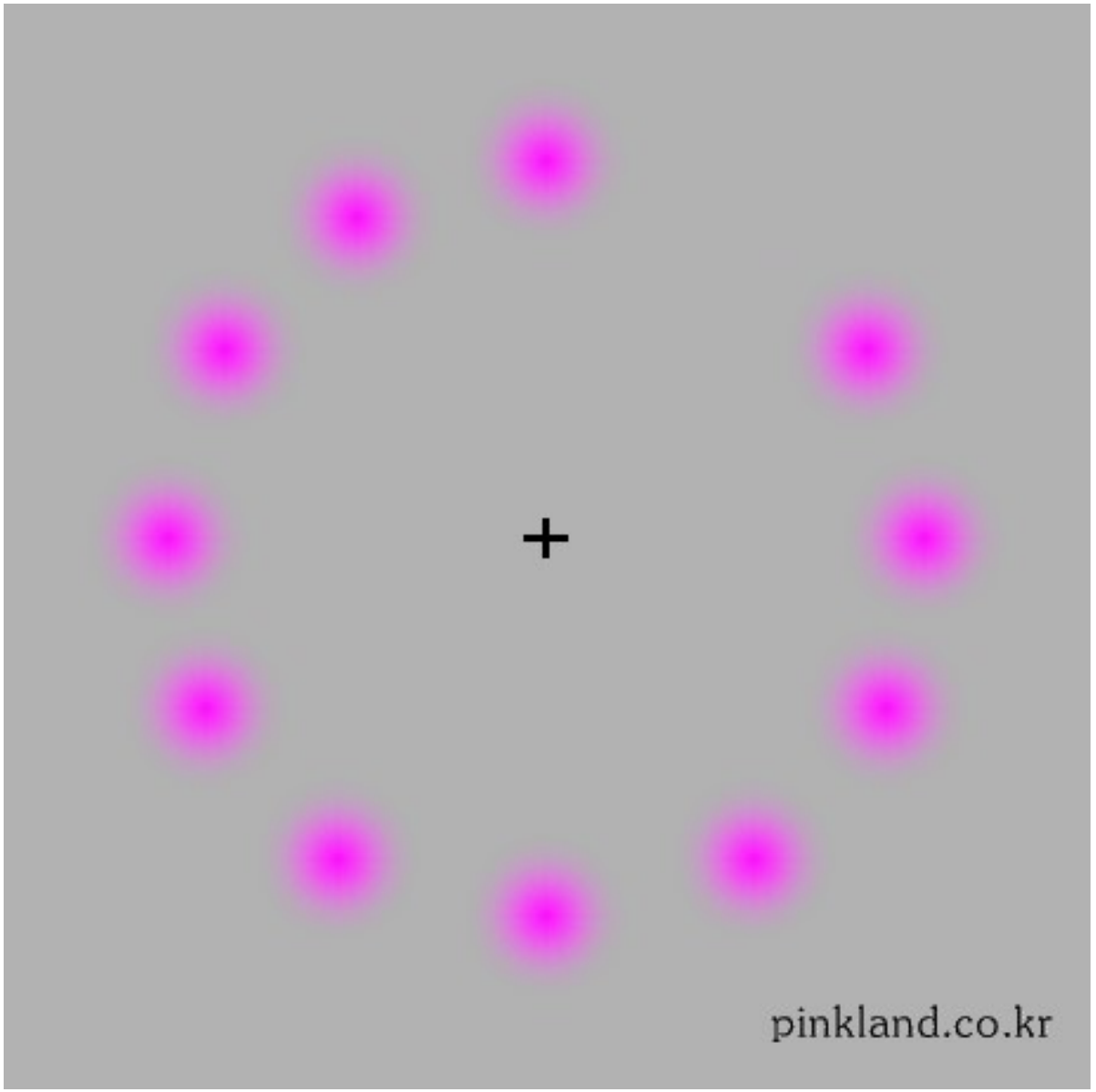


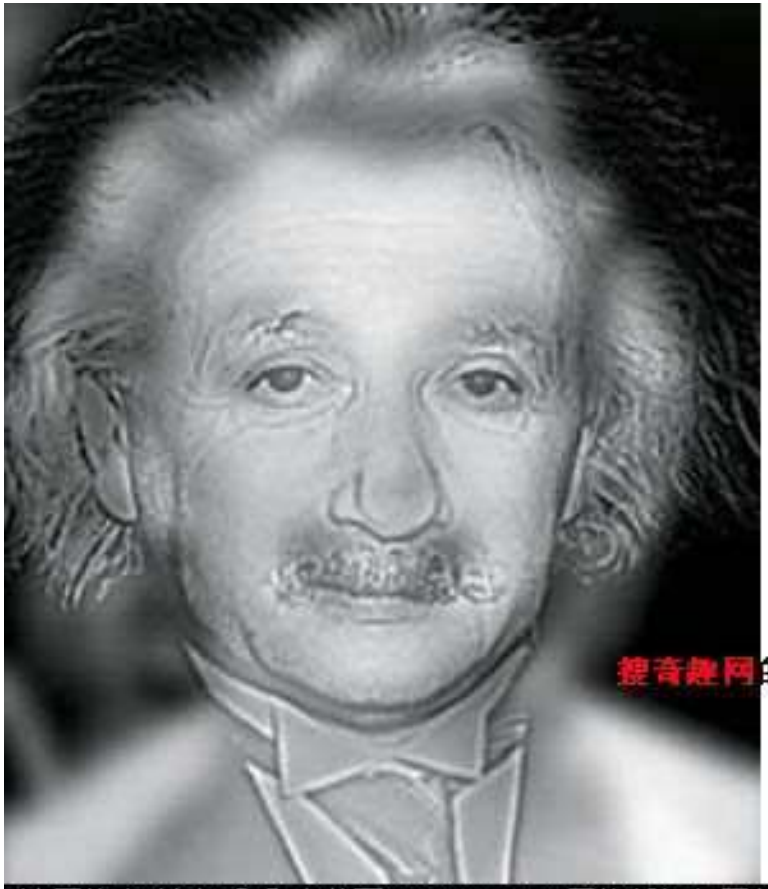

该图片由 墙角嘘嘘 上传至 Tiexue.Net 图片版权归原创者所有





Count the black dots! :o)


pinkland.co.kr



搜奇趣网搜奇图片
http://www.so77.net

不近视的人看到的是爱因斯坦
近视的人看到的是玛丽莲．梦露

视力正常的人眯着眼睛或者走到五米
外看也能看到玛丽莲．梦露

搜奇趣网的结论就是：
近视的人可以把男人看成美女！

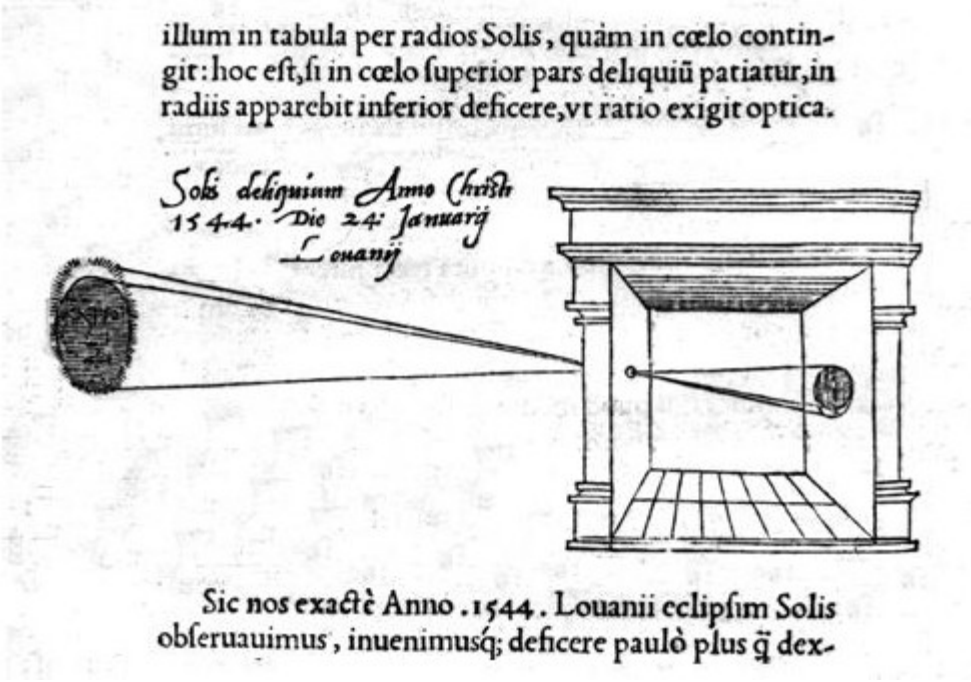

该图片由 墙角嘘嘘 上传至 Tiexue.Net 图片版权归原创者所有

图像获取

Camera Obscura

- Around 1519, Leonardo da Vinci (1452 - 1519)

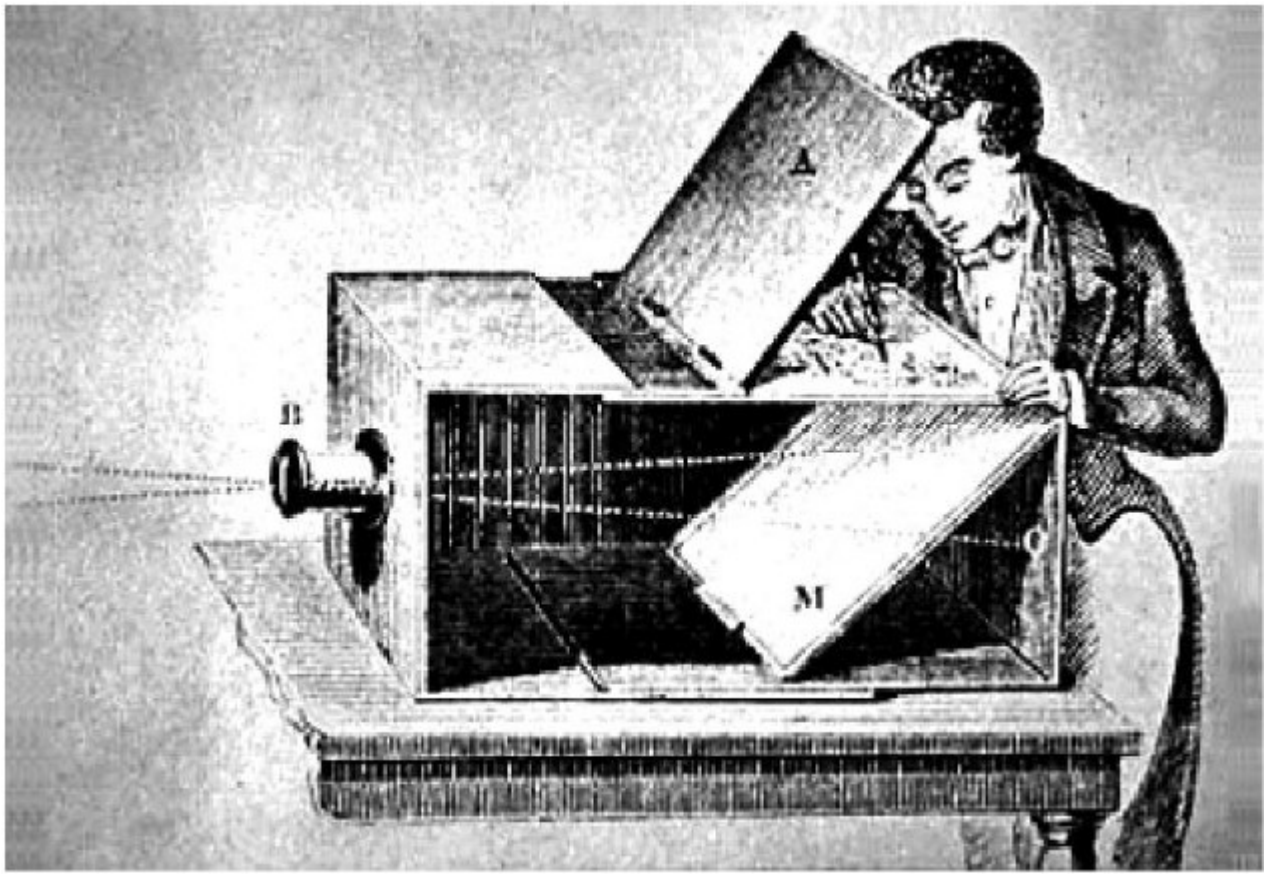
  - "When images of illuminated objects … penetrate through a small hole into a very dark room … you will see [on the opposite wall] these objects in their proper form and color, reduced in size … in a reversed position owing to the intersection of the rays"



Slide credit: Bernt Schiele

Source: http://www.acmi.net.au/AIC/CAMERA_OBSCURA.html

Camera Obscura



- **Used by artists (e.g. Vermeer 17th century) and scientists**

Slide credit: Bernt Schiele

Camera Obscura

Jetty at Margate England, 1898.

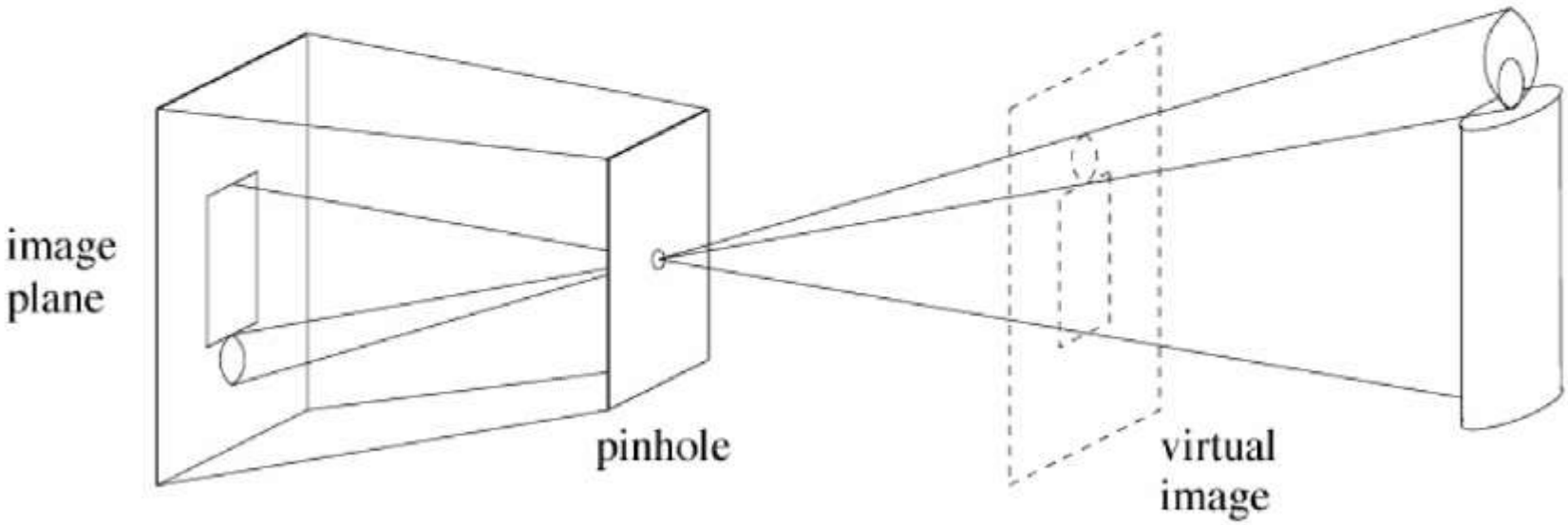
An attraction in the late 19th century

http://brightbytes.com/cosite/collection2.html

Adapted from R. Duraiswami

Pinhole Camera

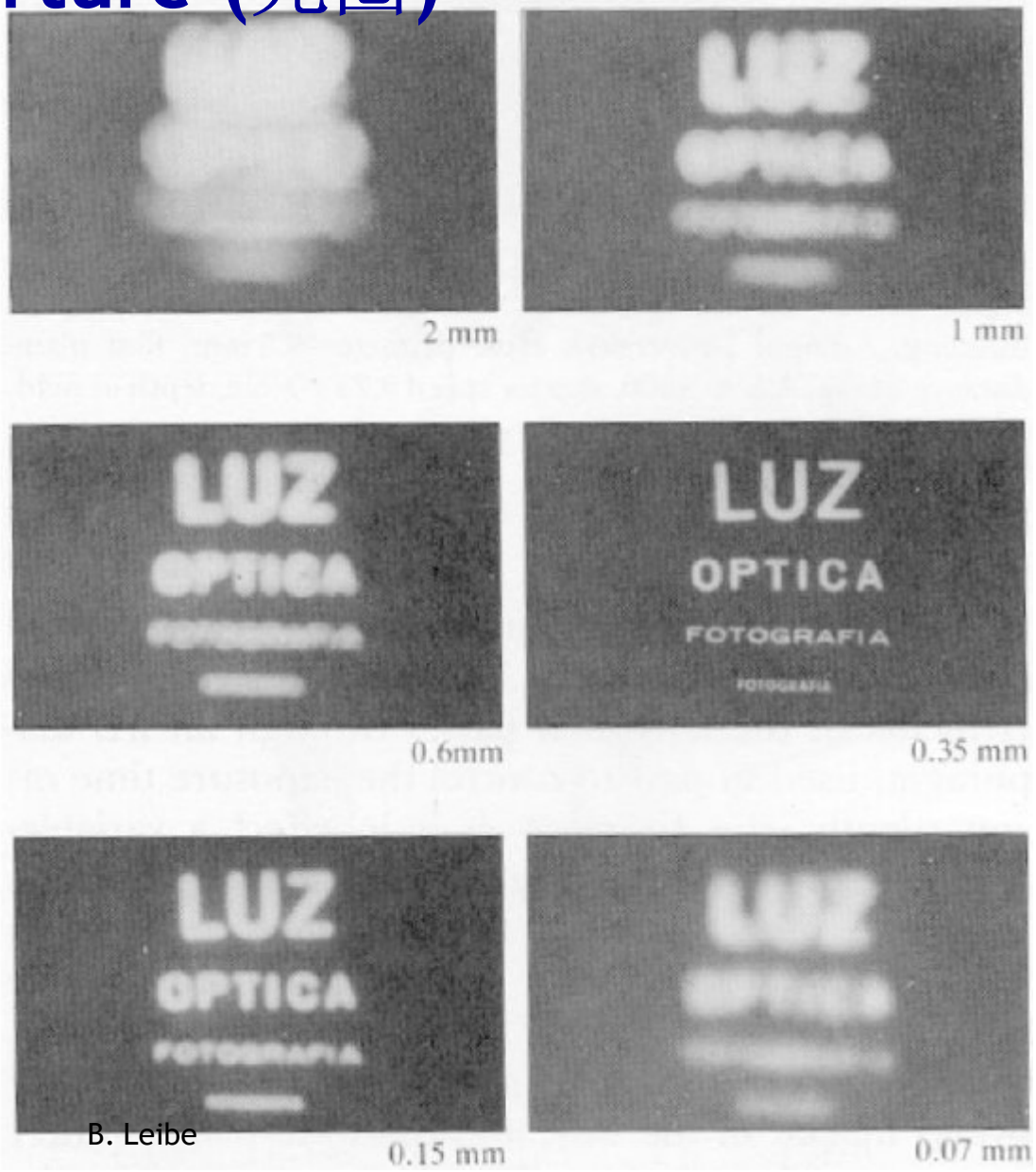
- (Simple) standard and abstract model today
  - Box with a small hole in it
  - Works in practice



Source: Forsyth & Ponce

Pinhole Size / Aperture (光圈)

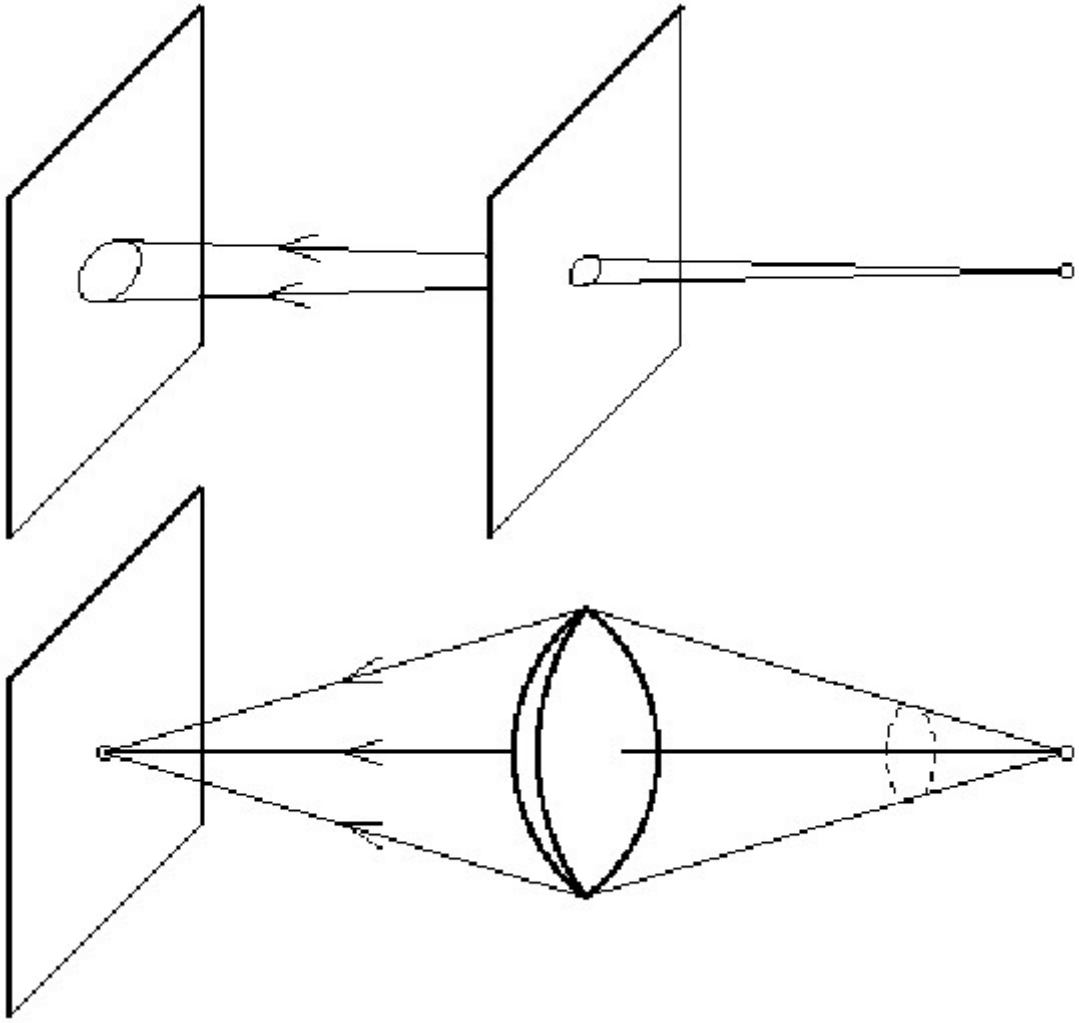
- Pinhole too big - many directions are averaged, blurring the image
- Pinhole too small - diffraction effects blur the image
- Generally, pinhole cameras are *dark*, because a very small set of rays from a particular point hits the screen.



Source: Forsyth & Ponce

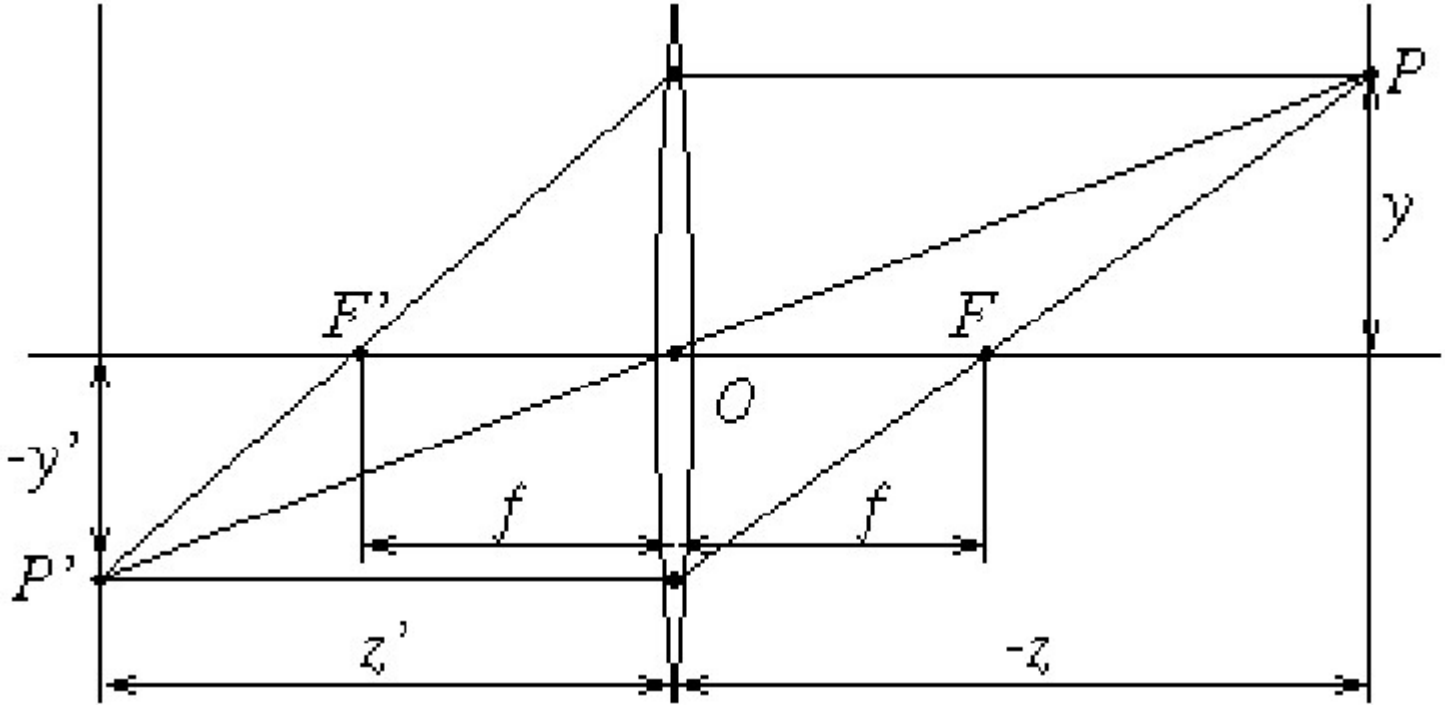
The Reason for Lenses

- **Keep the image in sharp focus while gathering light from a large area**



Source: Forsyth & Ponce

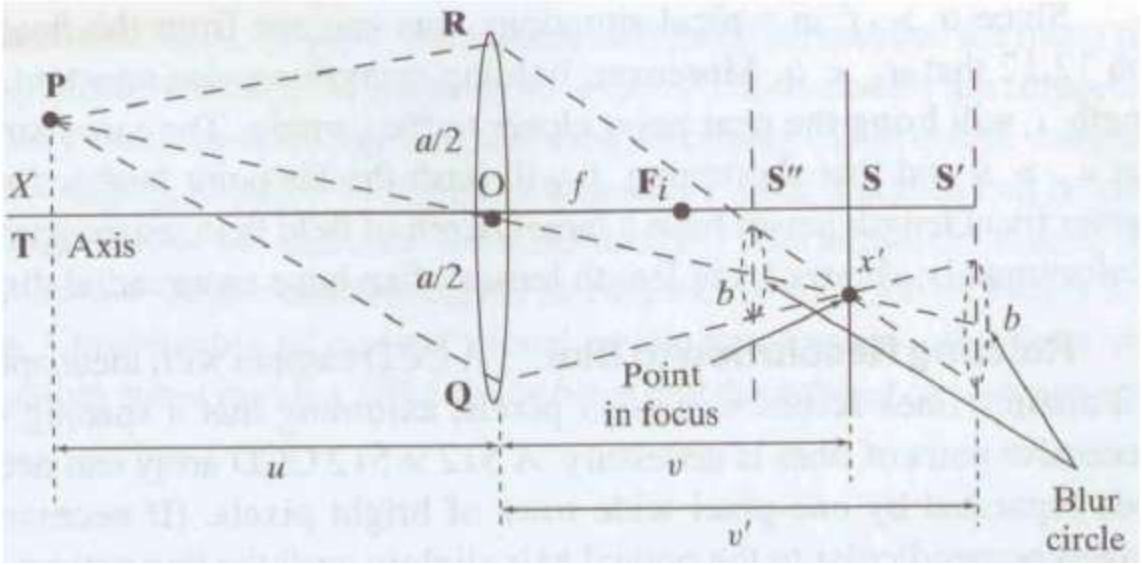
The Thin Lens



$$\frac{1}{z'} - \frac{1}{z} = \frac{1}{f}$$

Source: Forsyth & Ponce

Focus and Depth of Field (景深)



"circles of confusion"

Thin lens: scene points at distinct depths come in focus at different image planes.

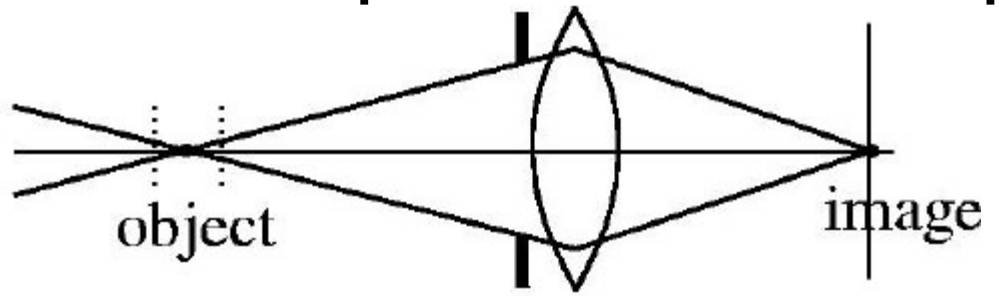
(Real camera lens systems have greater depth of field.)

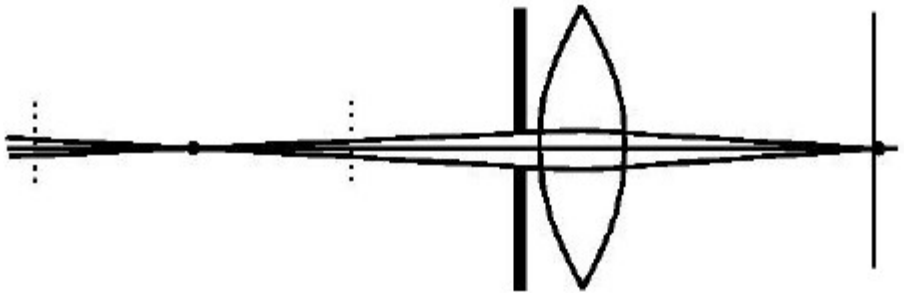
- Depth of field: distance between image planes where blur is tolerable

Source: Shapiro & Stockman

Focus and Depth of Field

- How does the aperture affect the depth of field?



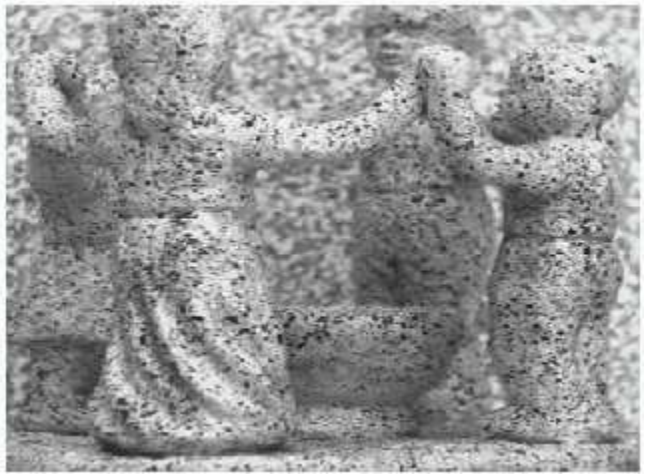


- A smaller aperture increases the range in which the object is approximately in focus

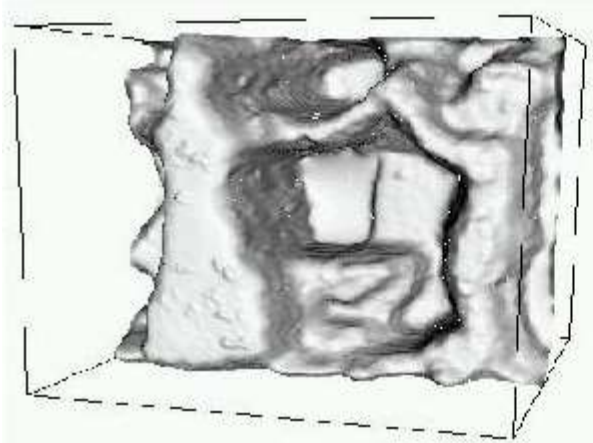
Flower images from Wikipedia http://en.wikipedia.org/wiki/Depth_of_field

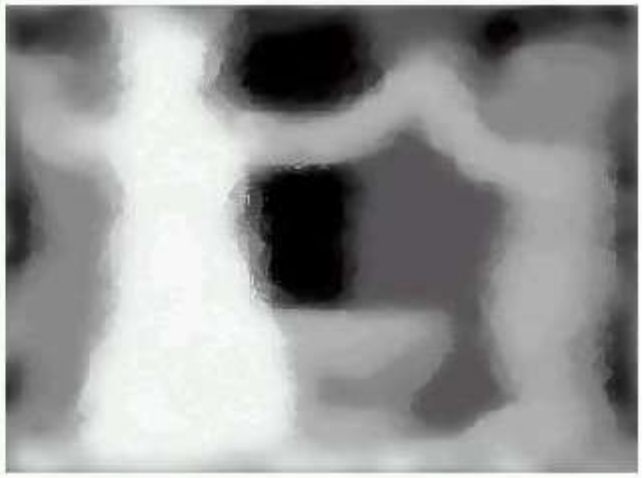
Slide from S. Seitz

Application: Depth from (De-)Focus



Images from same point of view, different camera parameters





3D Shape / depth estimates

Slide credit: Kristen Grauman

[figs from H. Jin and P. Favaro, 2002]

Field of View


28 mm lens, 65.5° × 46.4°


50 mm lens, 39.6° × 27.0°


70 mm lens, 28.9° × 19.5°


210 mm lens, 9.8° × 6.5°

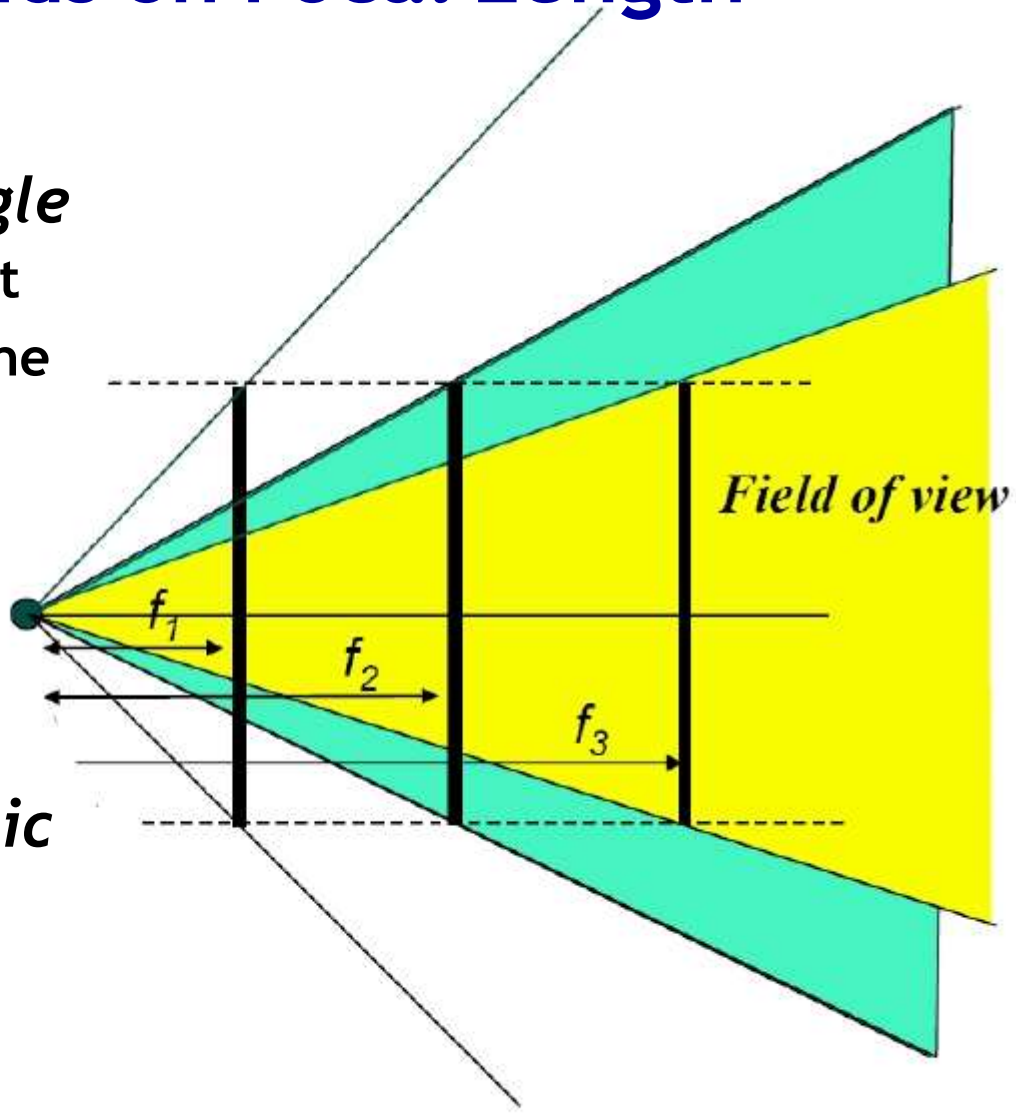
- **Angular measure of the portion of 3D space seen by the camera**

Slide credit: Kristen Grauman

Images from http://en.wikipedia.org/wiki/Angle_of_view

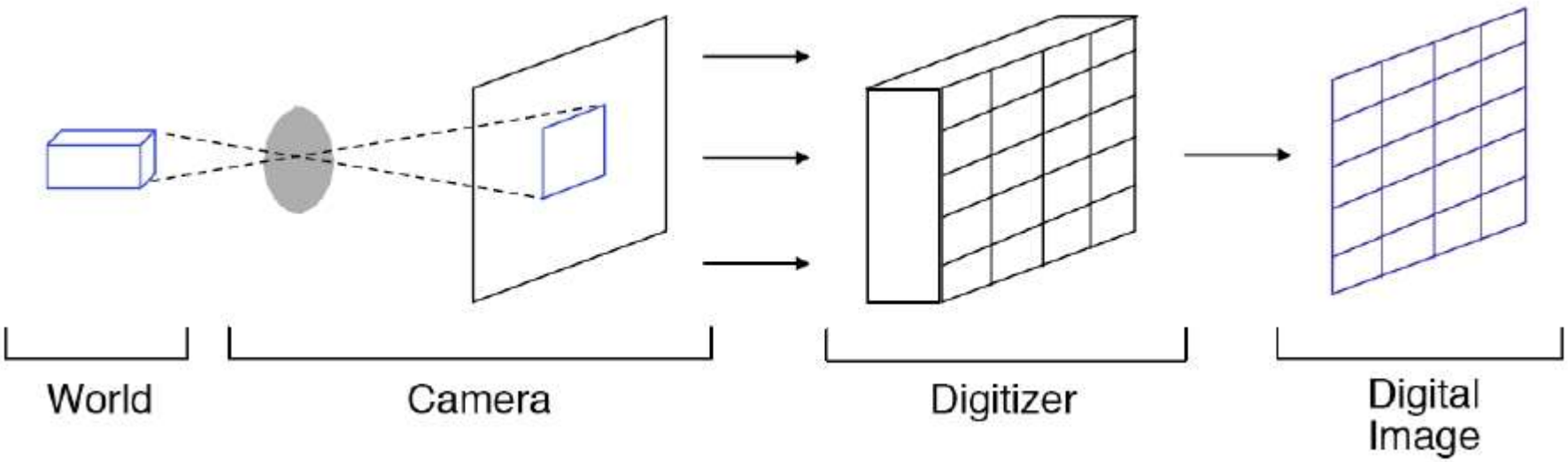
Field of View Depends on Focal Length

- As *f* gets smaller, image becomes more *wide angle*
  - More world points project onto the finite image plane
- As *f* gets larger, image becomes more *telescopic*
  - Smaller part of the world projects onto the finite image plane



from R. Duraiswami

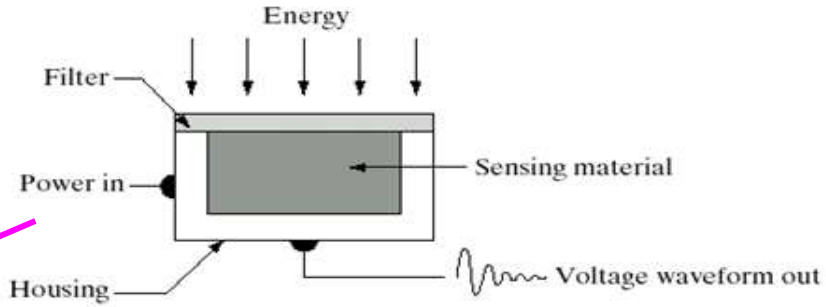
Digital Images



- Film is replaced by a sensor array
- Current technology: arrays of *charge coupled devices* (CCD)
- *Discretize* the image into pixels
- *Quantize* light intensities into pixel values.

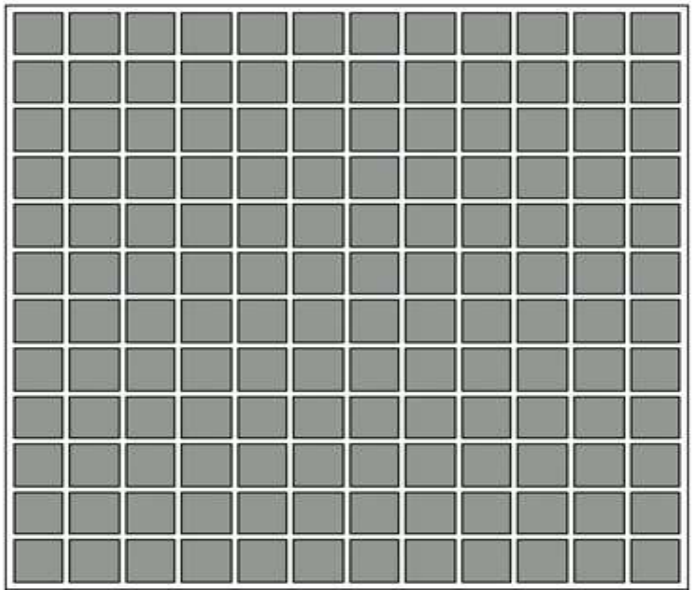
Image source: Michael Black

Image Acquisition



Single imaging sensor

Line sensor

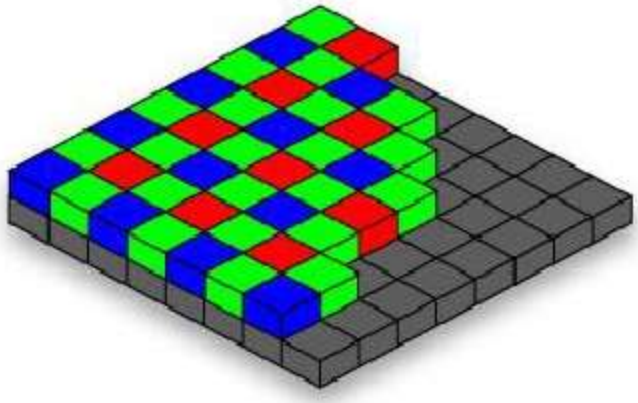


Array sensor

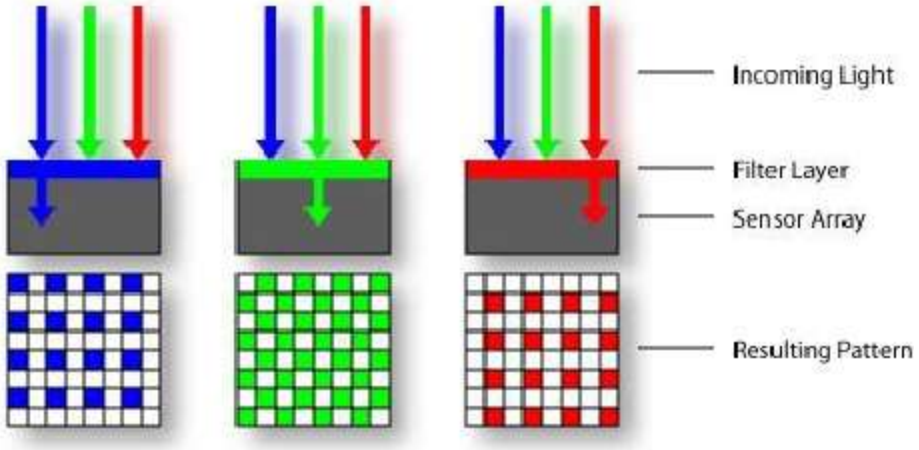
CCD: Charge-Coupled device

Color Sensing in Digital Cameras

Bayer grid

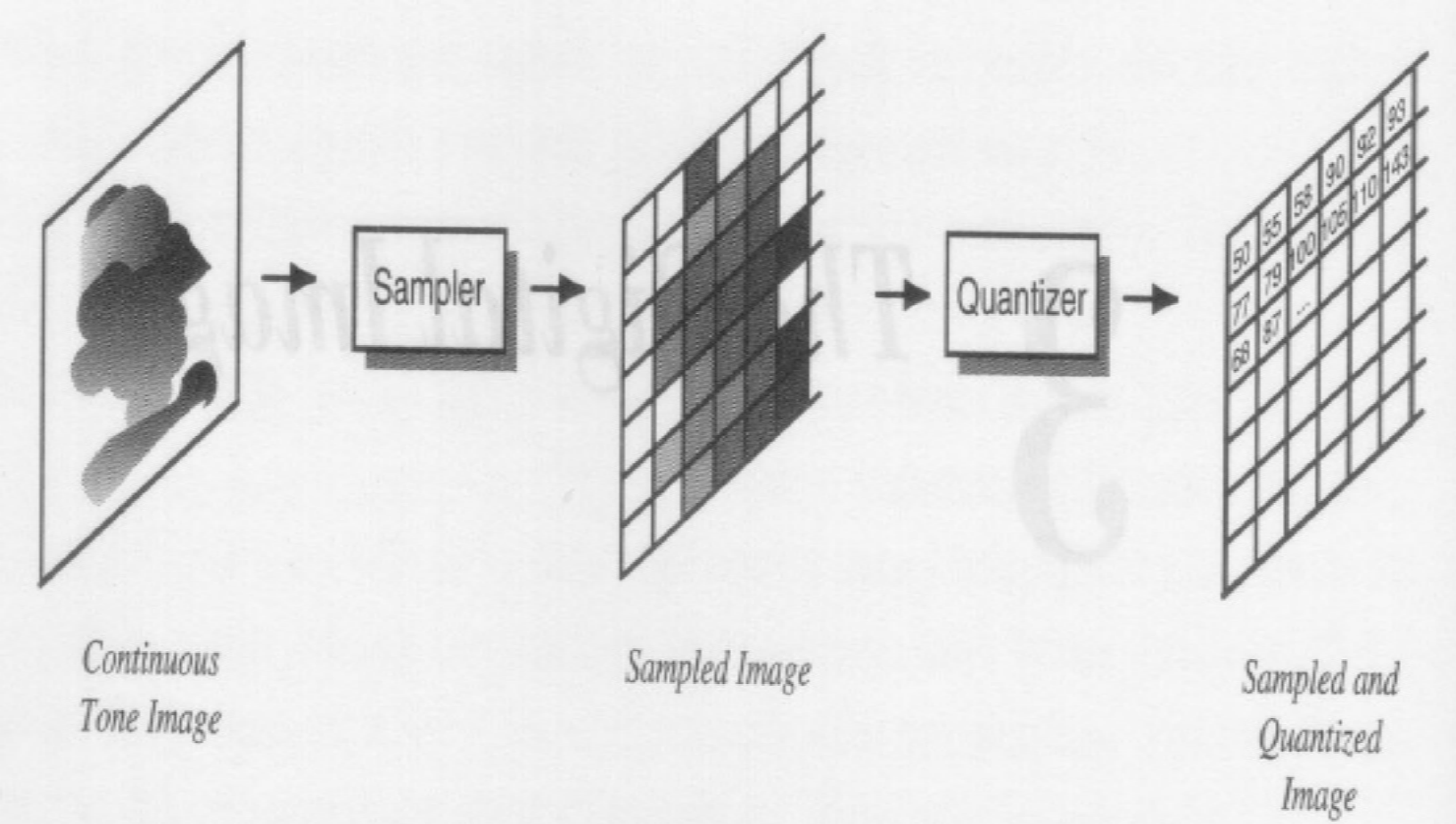


Estimate missing components from neighboring values (demosaicing)



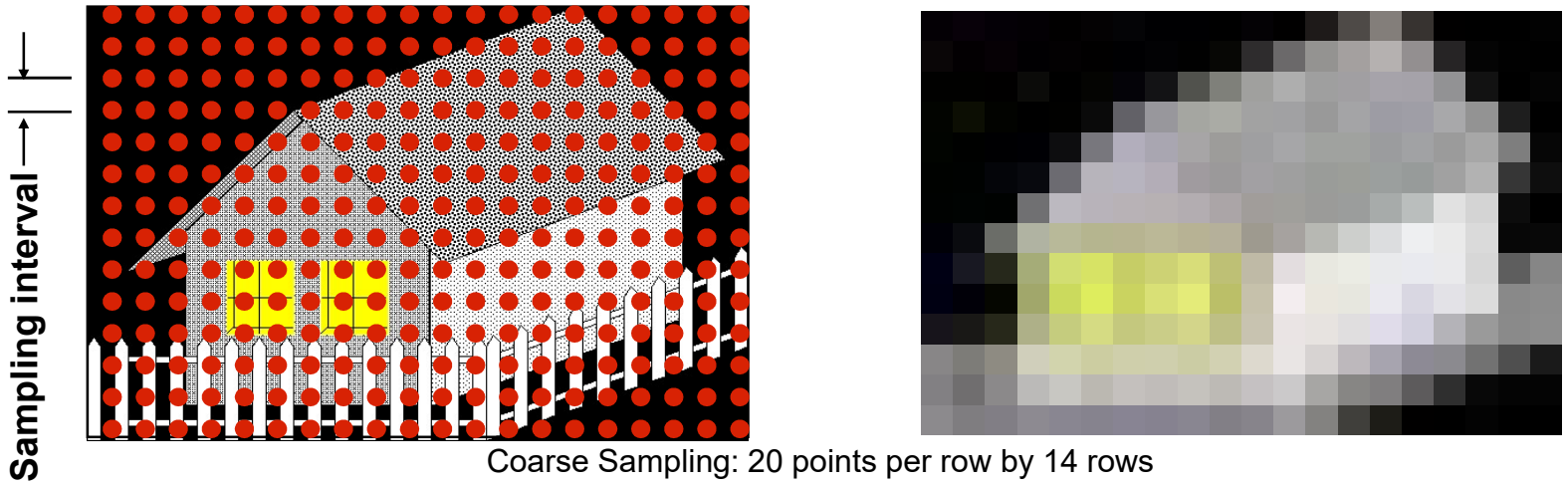
Source: Steve Seitz

图像采样与量化

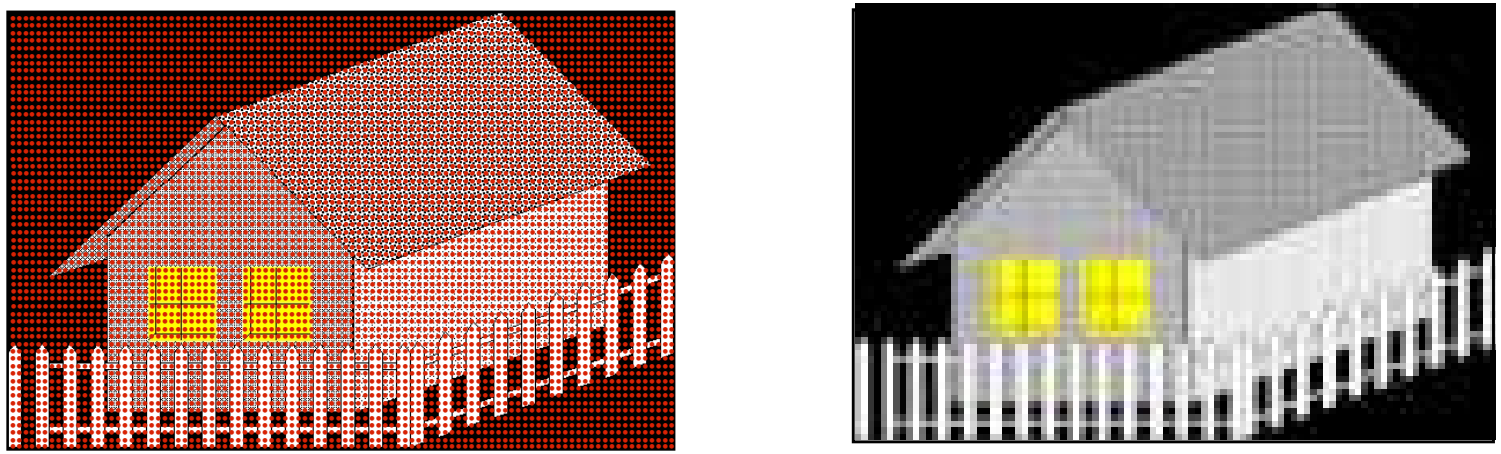

Sampler
Quantizer
50 55 58 90 92 93
77 79 100 105 110 143
68 87 ...
Continuous Tone Image
Sampled Image
Sampled and Quantized Image

采样

- **Spatial Resolution**



Coarse Sampling: 20 points per row by 14 rows



Finer Sampling: 100 points per row by 68 rows

采样

1024*1024

512*512

256*256

128*128

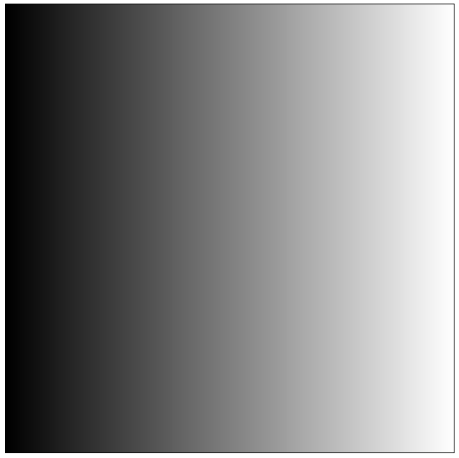
64*64

32*32

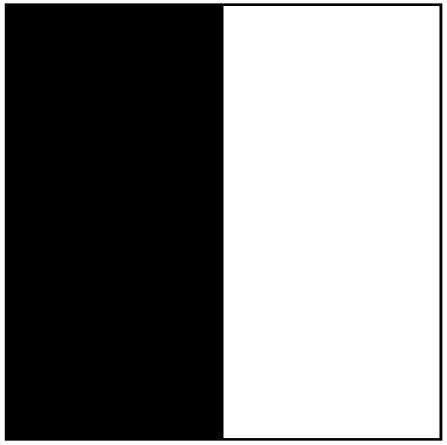
量化

- **Signal Quantization**
  - **quantize to K values 0,1,....K-1**
  - **K usually chosen to be a power of 2**

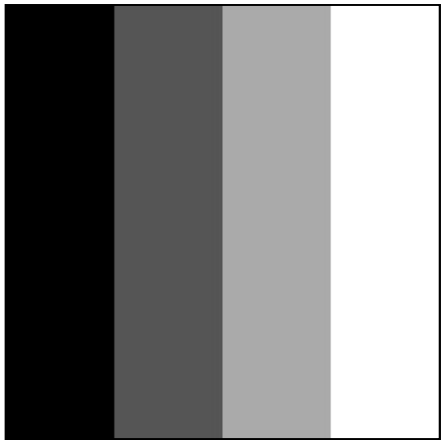
Original



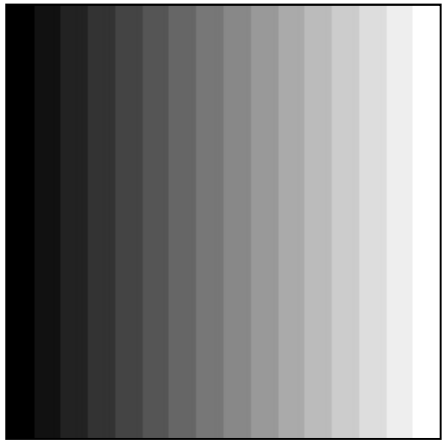
Linear Ramp



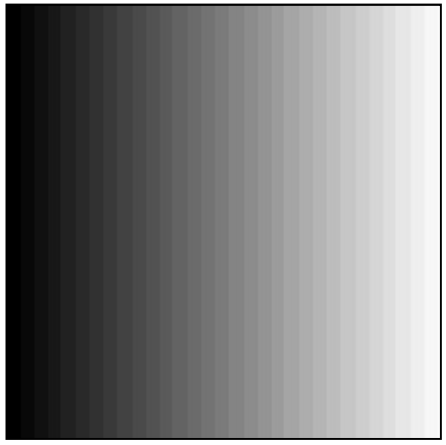
K=2



K=4



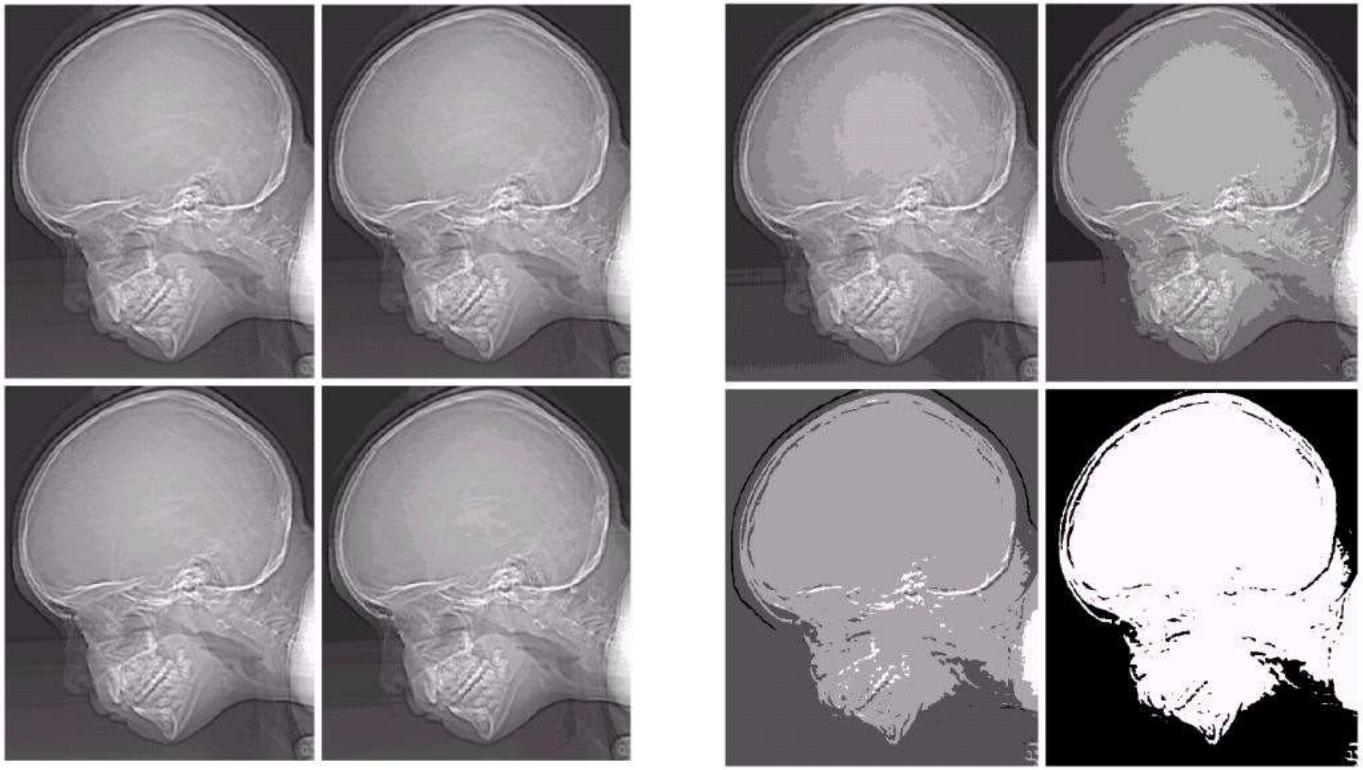
K=16



K=32

量化

- **Signal Quantization - Gray Scale Image**



量化

256-level

16-level

8-level

4-level

图像表示

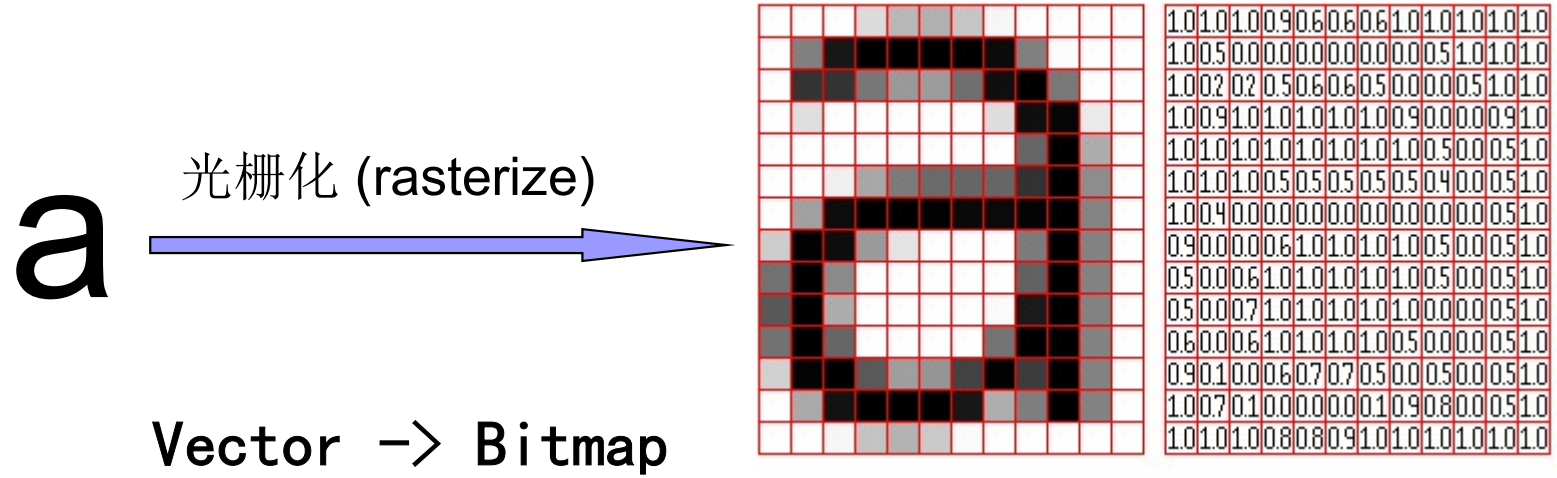
Image File Formats

- **Vector images（.ai, .eps, .ps, …）**
  - ✓ No aliasing and blur when scaling;
  - ✗ Difficult to be obtained, limited applications in practice;

```
draw circle
        center 0.5, 0.5
        radius 0.4
        fill-color yellow
        stroke-color black
        stroke-width 0.05
draw circle
        center 0.35, 0.4
        radius 0.05
        fill-color black
……
```

Image File Formats

- **Bitmap（ .bmp, .jpg, .png, .gif,…）**
  - ✓ Easy to get, wide applications;
  - ✗ Becomes blur and aliased when scaling;



GIF — Graphics Interchange Format

- **8-bit indexed, can be saved with a maximum of 256 colors;**
- **having the option to *dither* - (will mix pixels of two different available colors to create a suggestion of another color)**
- **can be *animated* . *transparent*.**

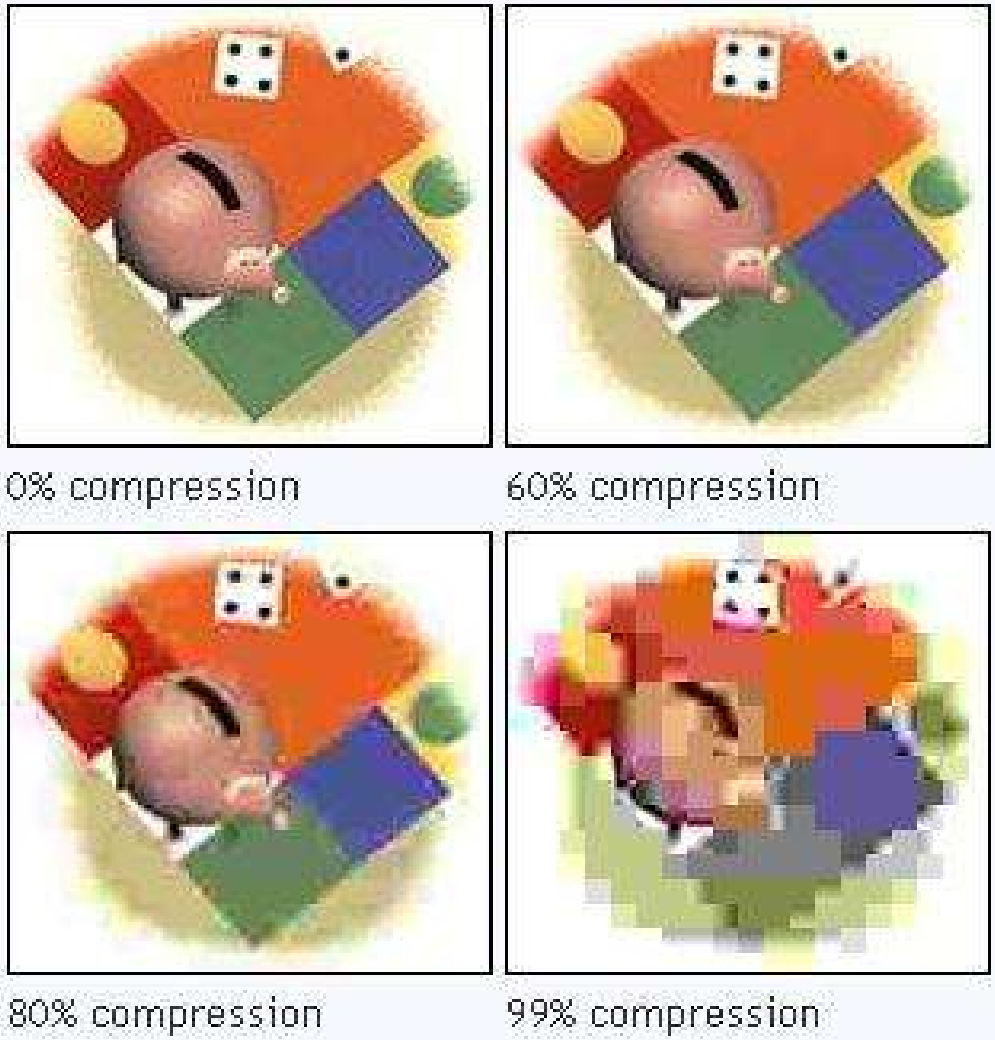
The ***superiority:***

its ***small size*** & ***high quality***

JPEG — Joint Photographic Experts Group

16-bit-- **capable of displaying millions of colors** at once without dithering.

a compression setting of about ***60%*** will result in the optimum balance of quality and file size .



PNG — Portable Network Graphics

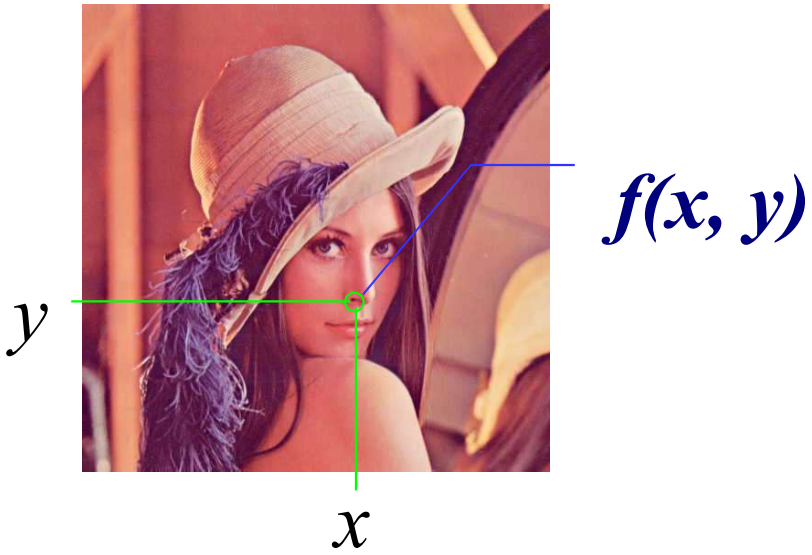
- ZIP based lossless compression
- Can be transparent (4-channel images)

BMP — Windows Bitmap

- Simple uncompressed
- Can either be indexed or not
- DIB (Device Independent Bitmap) / DDB (Device Dependent Bitmap)

Image (bitmap) representation

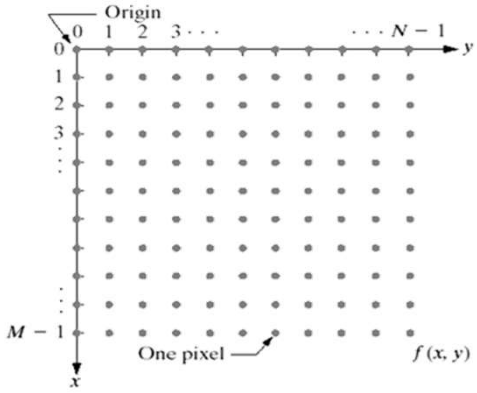
- **Image**
  - Function defined over 2D domain, $f(x, y)$



lena

Image Representation

- **Digital Image**
  - ***x, y, f(x, y) take only* discrete *values***
    - **Formed with finite elements**
  - ***Each element is called a* pixel (像素)**
    - **picture elements**
    - **image elements**
    - **pels**
    - **pixels - most widely used**



pixel

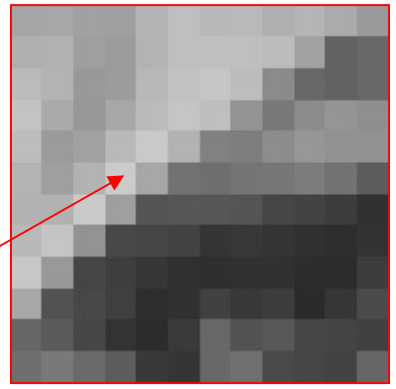


Image in Memory

- **2D or 3D array**

[B,G,R] [B,G,R] … [B,G,R]

[B,G,R] [B,G,R] … [B,G,R]

[B,G,R] [B,G,R] … [B,G,R]

……

[B,G,R] [B,G,R] … [B,G,R]

[B,G,R] [B,G,R] … [B,G,R]

[B,G,R] [B,G,R] … [B,G,R]

交叉存贮 (Interlaced)

[B] [B] [B] [B] … [B]

[B] [B] [B] [B] … [B]

……

[G] [G] [G] [G] … [G]

[G] [G] [G] [G] … [G]

……

[R] [R] [R] [R] … [R]

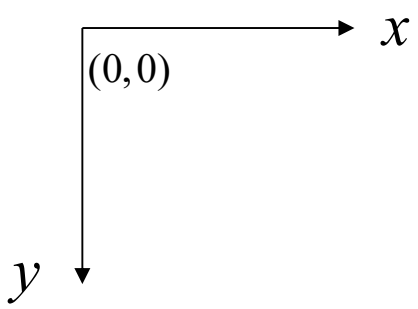
[R] [R] [R] [R] … [R]

……

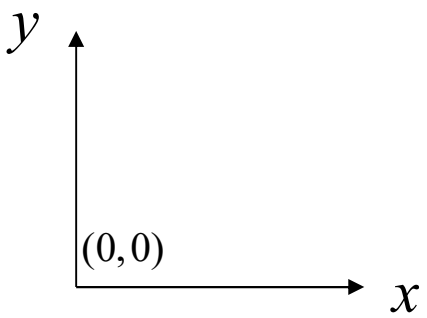
顺序存贮 (Sequential)

Image in Memory

- **Size (resolution, dpi, width*height, number of pixels)**
- **Color Space (RGB, CMYK, YUV, Lab, …)**
- **Channels（1，2，3，4，gray&color）**
- **Depth（number of bits for each channel, 8bits, 12bits,……,LDR ＆HDR）**
- **Coordinate system：**



Left Handed



Right Handed

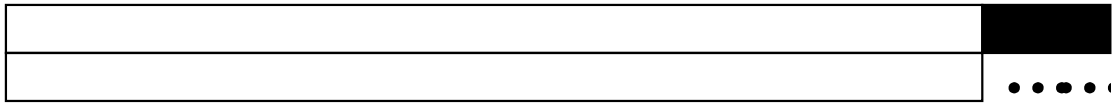
Image in Program

图像处理程序库……

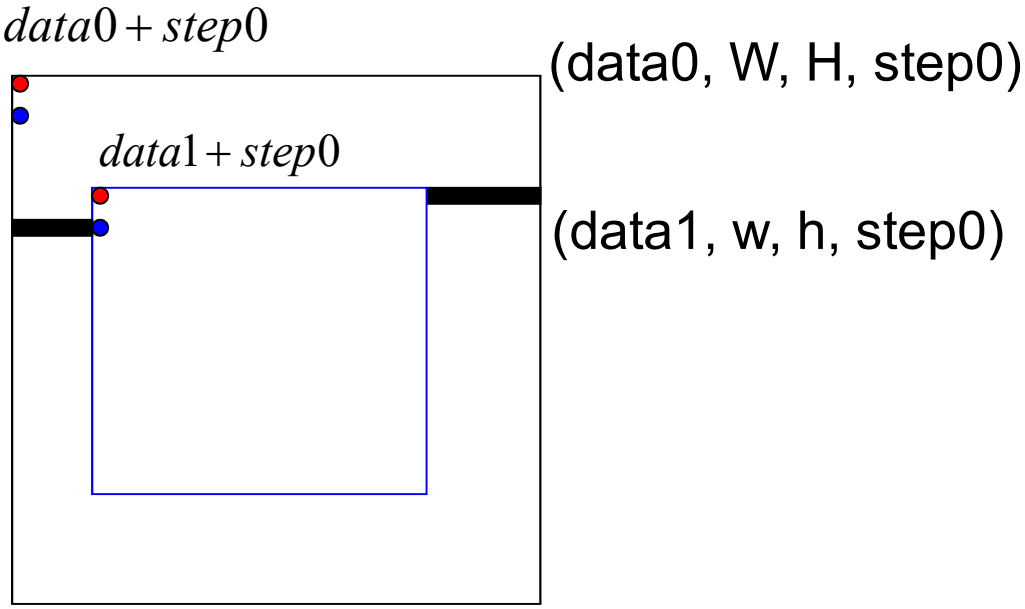
```
struct MyImage
{
    int     width, height;   // 大小
    int     type;            // 类型，含通道数、位深度信息
                             /* CV_8UC3 :  unsigned char [3]
                               CV_32SC1          :  int [1]
                               CV_32UC1          :  uint [1]
                               CV_32FC4          :  float [4]
                               */

    void*  data;             // 图像数据
    int     step;            // 步长（每行所占用的字节数）
};
```

step ?（stride, 步长）

- **For data-alignment:** make each row start from address that are multiple of 4, 8, or 16.



- **For representing sub-region (ROI):**



Access Pixels

```
struct MyImage
{
    int       width, height;
    int       type;

    void*     data;
    int       step;
};
```

- img.type=CV_8UC3：8位无符号，3通道数据

```
uchar*  get_pixel(const MyImage &img, int x, int y)
{
    ???????????
}
```

- img.type=CV_32SC3: 32位带符号，3通道数据

```
int*  get_pixel(const MyImage &img, int x, int y)
{
    ??????????????
}
```

Access Pixels

- img.type=CV_8UC3：8位无符号，3通道数据

```
uchar*  get_pixel(const MyImage &img, int x, int y)
{
      //  return (uchar*)img.data+y*img.width*3+x*3;

       return (uchar*)img.data+y*img.step+x*3;
}
```

step != width*nc

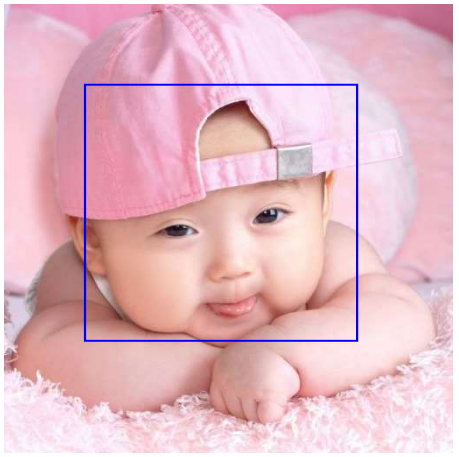
- img.type=CV_32SC3：32位带符号，3通道数据

```
int*  get_pixel(const MyImage &img, int x, int y)
{
      // return (int*)( (char*)img.data+y*img.step*4+x*3*4 );

       return (int*)( (char*)img.data+y*img.step+x*3*4 );
}
```

step始终是字节数

Traverse Pixels

```
void scan_pixels(uchar *data, int width, int height, int step, int nc)
{
    uchar *row=data;
    for(int yi=0; yi<height; ++yi, row+=step)
    {
        uchar *px=row;
        for(int xi=0; xi<width; ++xi, px+=nc)
        {
            // px now address the pixel  (xi, yi)
        }
    }
}
```

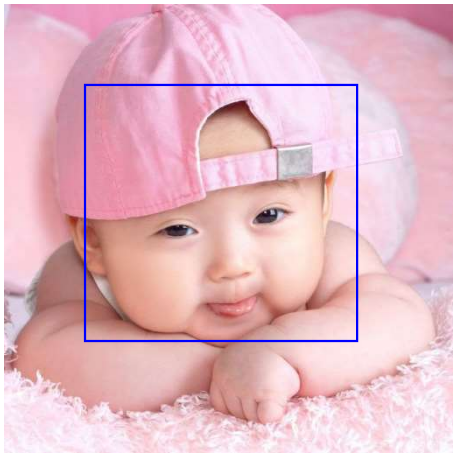


Traverse Pixels in ROI

```
void scan_roi_pixels(MyImage &img, int x, int y, int roi_width, int roi_height)
{// 通道数nc=img.nc();
    ??????????????????????????
}
```

Traverse Pixels

```
void scan_pixels(uchar *data, int width, int height, int step, int nc)
{
//.......
}
```



Traverse Pixels in ROI

```
void scan_roi_pixels(MyImage &img, int x, int y, int roi_width, int roi_height)
{// 通道数nc=img.nc();

    scan_pixels( get_pixel(img, x, y),  roi_width, roi_height, img.step, img.nc() );

}
```

OpenCV

- CV=Computer Vision
- Created by Intel and maintained by Willow Garage.
- Available for C, C++, and Python
- Cross-platform: Windows, Linux/Mac, Android, iOS
- Open Source and free
- Plenty of features : more than 500 functions for image processing and computer vision
- Being actively developed and updated
- …
- Google for more

OpenCV API Reference

- Introduction
- core. The Core Functionality
- imgproc. Image Processing
- highgui. High-level GUI and Media I/O
- video. Video Analysis
- calib3d. Camera Calibration and 3D Reconstruction
- features2d. 2D feature detection and matching
- objdetect. Object Detection
- ml. Machine Learning
- flann. Clustering and Search in Multi-Dimensional Spaces
- gpu. GPU-accelerated Computer Vision
- stitching. Images stitching