24: Perception

- 24.1 Introduction
- 24.2 Image Formation
- 24.3 Image-Processing Operations For Early Vision

24.1 Introduction

- Perception is initiated by **sensors** . A sensor is anything that can change the computational state of the agent in response to a change in the state of the world.

- The basic approach taken is to first understand how **sensory stimuli** are created by the world, and then to ask the following question:

  – *if sensory stimuli are produced in such and such a way by the world, then what must the world have been like to produce this particular stimulus?*

- Let the sensory stimulus be S, and let W be the world (where W will include the agent itself). If the function f describes the way in which the world generates sensory stimuli, then we have

$$S = f(W)$$

- Now, our question is: given f and S, what can be said about W?

$$W = f^{-1}(S)$$

- Unfortunately, f does not have a proper **inverse** .
- A second, and perhaps more important, drawback of the straightforward approach is that it is trying to solve too difficult a problem.

Visual Perception

Let us look at some of the possible *uses for vision:*

- Manipulation
- Navigation
- Object recognition

24.2 Image Formation

24.2.1 Pinhole camera

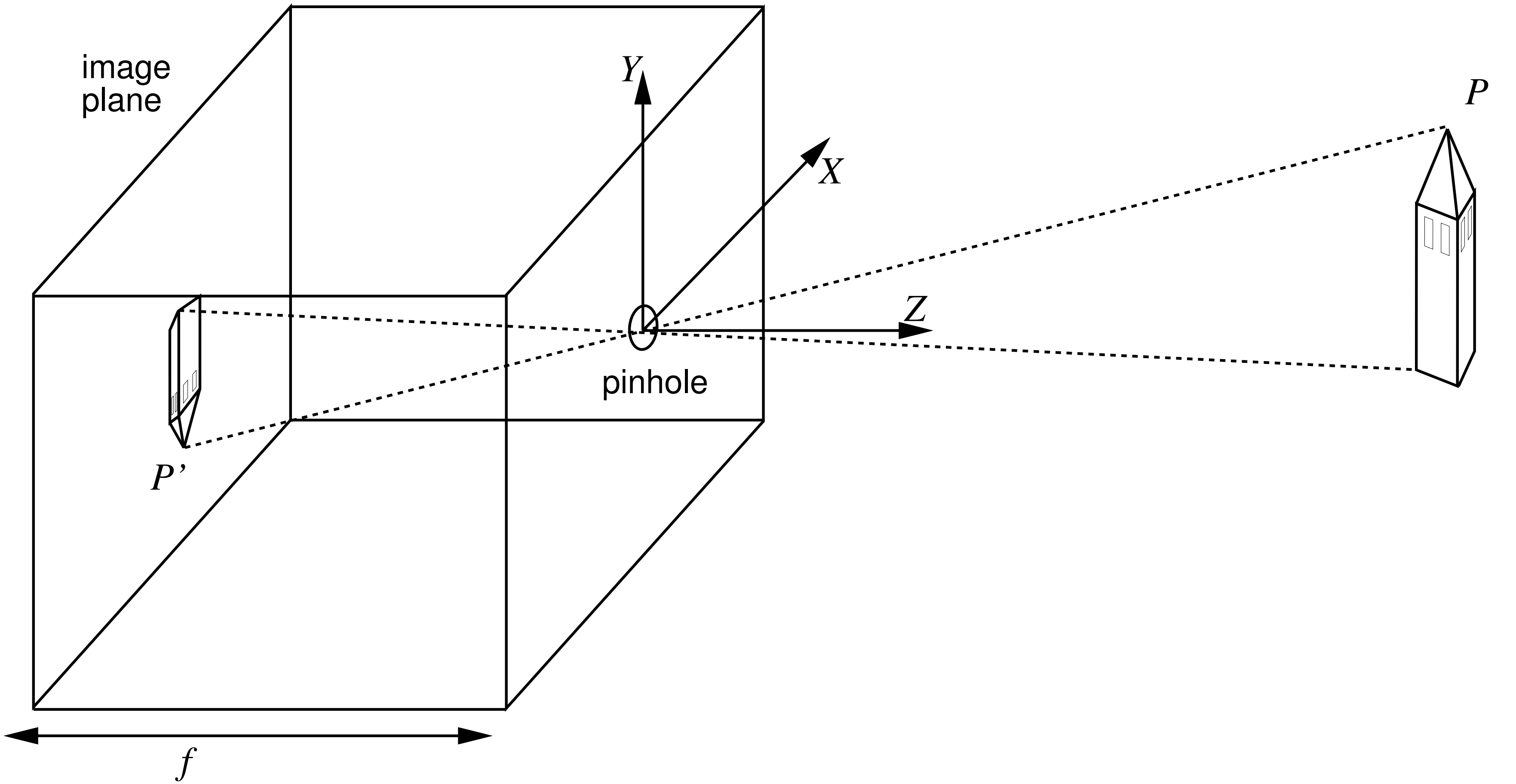

Fig.24.1 Geometry of image formation in the pinhole camera.

- The simplest way to form an image is to use a **pinhole camera** . Let P be a point in the scene, with coordinates (X, Y, Z), and P' be its image on the *image plane,* with coordinates (x, y, z). If f is the distance from the pinhole O to the image plane, then by similar triangles, we can derive the following equations:

$$\frac{-x}{f} = \frac{X}{Z}, \ \frac{-y}{f} = \frac{Y}{Z} \ \Rightarrow \ x = \frac{-fX}{Z}, \ y = \frac{-fY}{Z}$$

- These equations define an image formation process known as **perspective projection** .

- Equivalently, we can model the perspective projection process with the projection plane being at a distance f in $front$ of the pinhole.

- Under *perspective projection,* parallel lines appear to converge to a point on the horizon. An arbitrary point P_λ on the line passing through (X_0, Y_0, Z_0) in the direction (U, V, W) is given by $(X_0 + \lambda U, Y_0 + \lambda V, Z_0 + \lambda W)$, with λ varying between $+\infty$ and $-\infty$.

- The projection of P_λ on the image plane is given by

$$\left(f\frac{X_0 + \lambda U}{Z_0 + \lambda W}, f\frac{Y_0 + \lambda V}{Z_0 + \lambda W} \right)$$

- As $\lambda \to \infty$ or $\to -\infty$, this becomes $p_\infty = (fU/W, fV/W)$ if $W \neq 0$.

- We call p_∞ the **vanishing point** associated with the family of straight lines with direction (U, V, W).

24.2.2 Photometry of image formation

- The *brightness* of a pixel p in the image is proportional to the amount of light directed toward the camera by the surface patch S_p that projects to pixel p.

- This in turn depends on the *reflectance properties* of S_p, the position and distribution of the light sources.

- There is also a dependence on the reflectance properties of the rest of the scene because other scene surfaces can serve as indirect light sources by reflecting light received by them onto S_p.

- *Lambert's cosine law* is used to describe the reflection of light from a perfectly *diffusing* or **Lambertian** surface. The intensity E of light reflected from a perfect diffuser is given by

$$E = \rho E_0 \ cos\theta$$

where E_0 is the intensity of the light source; ρ is the albedo, which varies from 0 (for perfectly black surfaces) to 1 (for pure white surfaces); and θ is the angle between the light direction and the surface normal.

24.2.3 Spectrophotometry of image formation

- *Visible light* comes in a range of wavelengths—ranging from 400 nm on the violet end of the spectrum to 700 nm on the red end.
- The explanation is that *color* is quite literally in the eye of the beholder.
- There are three different cone types in the eye with three different spectral sensitivity curves $R_k(\lambda)$. The output of the kth cone at location (x, y) at time t then is

$$I_k(x, y, t) = \int I(x, y, t, \lambda) R_k(\lambda) d\lambda.$$

- The infinite dimensional wavelength space has been projected to a three-dimensional color space. This means that we ought to think of I as a three-dimensional vector at (x, y, t). Because the eye maps many different frequency spectra into the same color sensation, we should expect that there exist **metamers** —different light spectra that appear the same to a human.

24.3 Image-Processing Operations For Early Vision

- **Edge detection** : Fig.24.5 shows an image of a stapler resting on a table and all the edges detected (b).

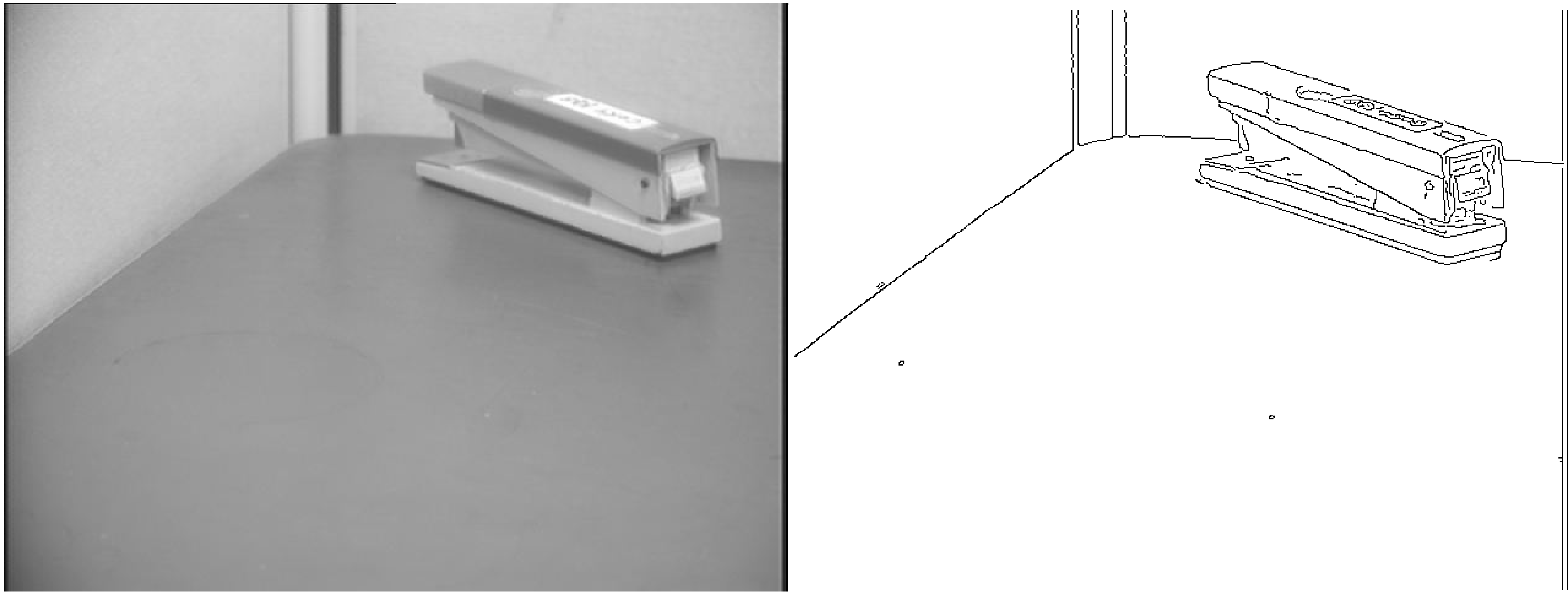

Fig.24.5 (a) Photograph of a stapler. (b) Edges computed from (a)

- In the example, we have different *kinds of edges:*

1. depth discontinuities, labelled 1;
2. surface-orientation discontinuities, labelled 2;
3. a reflectance discontinuity, labelled 3;
4. and an illumination discontinuity (shadow), labelled 4.

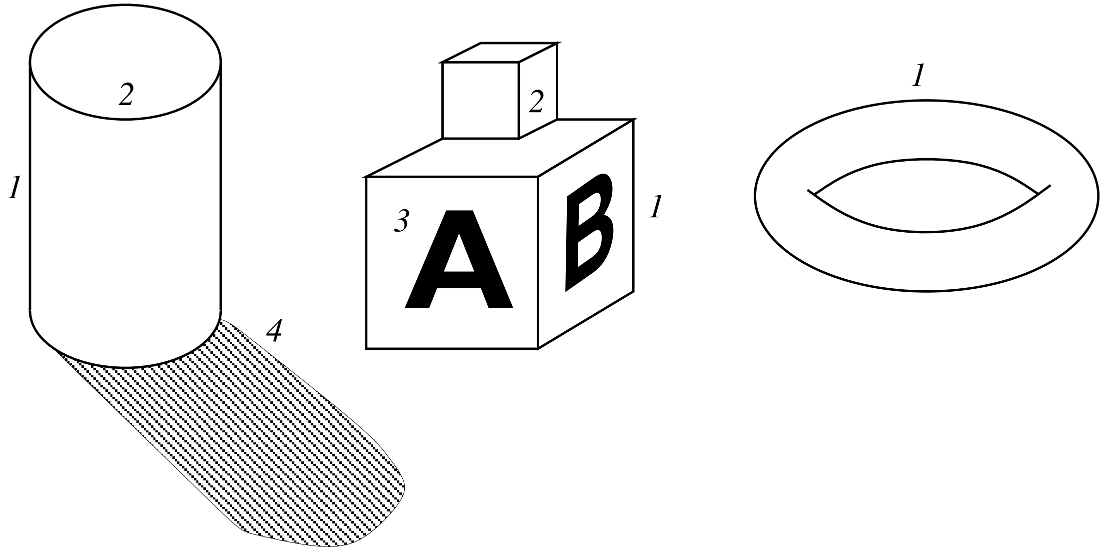

Fig.24.6

- Because edges correspond to locations in images where the brightness undergoes a *sharp change,* a naive idea would be to *differentiate* the image and look for places where the magnitude of the derivative $I'(x)$ is large.
- We get much better results by combining the differentiation operation with **smoothing** .
- To understand these ideas better, we need the mathematical concept of **convolution** . Many useful image-processing operations such as smoothing and differentiation can be performed by convolving the image with suitable functions.

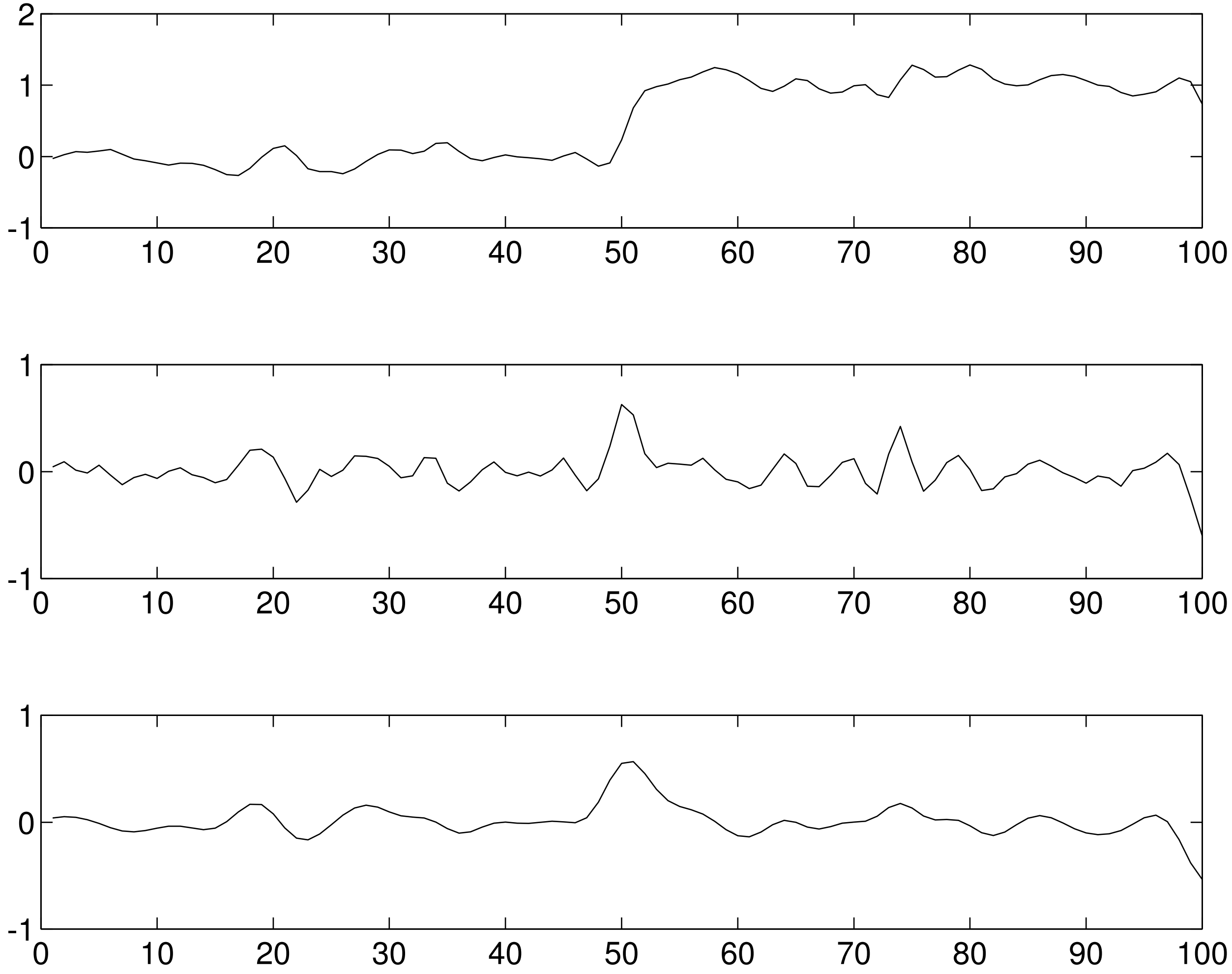

Fig.24.7 (a) Intensity profile $I(x)$ along a 1-D section across a step edge. (b) Its derivative $I'(x)$. (c) The result of the *convolution* $R(x) = i * G'_\sigma$. Looking for large values in this function is a good way to find edges in (a).

24.3.1 Convolution with linear filters

- The result of *convolving* two functions f and g is the new function h, denoted as $h = f * g$, which is defined by

$$h(x) = \int_{-\infty}^{+\infty} f(u)\ g(x-u)du \text{ and } h(x) = \sum_{u=-\infty}^{+\infty} f(u)\ g(x-u)$$

for *continuous* and *discrete* domains respectively.

$$h(x,y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(u,v)\ g(x-u, y-v)du\ dv$$

$$h(x,y) = \sum_{-\infty}^{+\infty} \sum_{-\infty}^{+\infty} f(u,v)\ g(x-u, y-v)$$

24.3.2 Edge detection

- One standard form of *smoothing* is to *convolve* the image with a *Gaussian* function

$$G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$$

- Now it can be shown that for any functions f and g, $f * g' = (f * g)'$.

$$G'_\sigma(x) = \frac{-x}{\sqrt{2\pi}\sigma^3} e^{-x^2/2\sigma^2}$$

- So, we have a simple algorithm for *1-D edge detection:*

1. Convolve the image I with G'_σ to obtain R.
2. Find the absolute value of R.
3. Mark those peaks in $||R||$ that are above some prespecified threshold T_n. The threshold is chosen to eliminate spurious peaks due to noise.

- In *2-D,* the algorithm for detecting *vertical edges* then is as follows:

1. Convolve the image $I(x, y)$ with $f_v(x, y) = G'_\sigma(x)G_\sigma(y)$ to obtain $R_v(x, y)$.
2. Find the absolute value of $R_v(x, y)$.
3. Mark those peaks in $||R_v||(x, y)$ that are above some prespecified threshold T_n.