and V together capture its chrominance. Thus, a bounding box expressed in YUV space can achieve greater stability with respect to changes in illumination than is possible in RGB space.

The CMVision color sensor achieves a resolution of 160 x 120 and returns, for each object detected, a bounding box and a centroid. The software for CMVision is available freely with a Gnu Public License at [161].

Key performance bottlenecks for both the CMVision software, the CMUcam hardware system, and the Cognachrome hardware system continue to be the quality of imaging chips and available computational speed. As significant advances are made on these frontiers one can expect packaged vision systems to witness tremendous performance improvements.

4.2 Representing Uncertainty

In section 4.1.2 we presented a terminology for describing the performance characteristics of a sensor. As mentioned there, sensors are imperfect devices with errors of both systematic and random nature. Random errors, in particular, cannot be corrected, and so they represent atomic levels of sensor uncertainty.

But when you build a mobile robot, you combine information from many sensors, even using the same sensors repeatedly, over time, to possibly build a model of the environment. How can we scale up, from characterizing the uncertainty of a single sensor to the uncertainty of the resulting robot system?

We begin by presenting a statistical representation for the random error associated with an individual sensor [12]. With a quantitative tool in hand, the standard Gaussian uncertainty model can be presented and evaluated. Finally, we present a framework for computing the uncertainty of conclusions drawn from a set of quantifiably uncertain measurements, known as the *error propagation law*.

4.2.1 Statistical representation

We have already defined *error* as the difference between a sensor measurement and the true value. From a statistical point of view, we wish to characterize the error of a sensor, not for one specific measurement but for any measurement. Let us formulate the problem of sensing as an estimation problem. The sensor has taken a set of n measurements with values ρ_i. The goal is to characterize the estimate of the true value $E[X]$ given these measurements:

$$E[X] = g(\rho_1, \rho_2, \ldots, \rho_n) \tag{4.50}$$

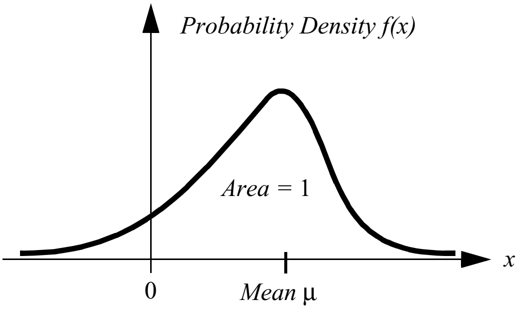


Figure 4.30
A sample probability density function, showing a single probability peak (i.e., unimodal) with asymptotic drops in both directions.

From this perspective, the true value is represented by a random (and therefore unknown) variable X. We use a *probability density function* to characterize the statistical properties of the value of X.

In figure 4.30, the density function identifies for each possible value x of X a probability density $f(x)$ along the y-axis. The area under the curve is 1, indicating the complete chance of X having *some* value:

$$\int_{-\infty}^{\infty} f(x)dx = 1 \tag{4.51}$$

The probability of the value of X falling between two limits a and b is computed as the bounded integral:

$$P[a < X \leq b] = \int_{a}^{b} f(x)dx \tag{4.52}$$

The probability density function is a useful way to characterize the possible values of X because it not only captures the range of X but also the comparative probability of different values for X. Using $f(x)$ we can quantitatively define the mean, variance, and standard deviation as follows.

The *mean value* μ is equivalent to the expected value $E[X]$ if we were to measure X an infinite number of times and average all of the resulting values. We can easily define $E[X]$:

$$\mu = E[X] = \int_{-\infty}^{\infty} xf(x)dx \tag{4.53}$$

Note in the above equation that calculation of $E[X]$ is identical to the weighted average of all possible values of x. In contrast, the *mean square value* is simply the weighted average of the squares of all values of x:

$$E[X^2] = \int_{-\infty}^{\infty} x^2 f(x)dx \tag{4.54}$$

Characterization of the "width" of the possible values of X is a key statistical measure, and this requires first defining the *variance* σ^2:

$$Var(X) = \sigma^2 = \int_{-\infty}^{\infty} (x-\mu)^2 f(x)dx \tag{4.55}$$

Finally, the *standard deviation* σ is simply the square root of varianceσ, and σ^2 will play important roles in our characterization of the error of a single sensor as well as the error of a model generated by combining multiple sensor readings.

4.2.1.1 Independence of random variables.

With the tools presented above, we often evaluate systems with multiple random variables. For instance, a mobile robot's laser rangefinder may be used to measure the position of a feature on the robot's right and, later, another feature on the robot's left. The position of each feature in the real world may be treated as random variables, X_1 and X_2.

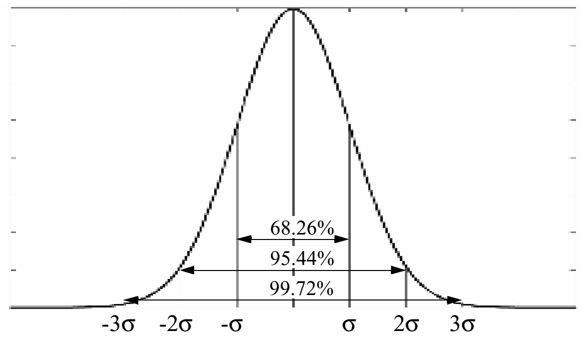
Two random variables X_1 and X_2 are *independent* if the particular value of one has no bearing on the particular value of the other. In this case we can draw several important conclusions about the statistical behavior of X_1 and X_2. First, the expected value (or mean value) of the product of random variables is equal to the product of their mean values:

$$E[X_1X_2] = E[X_1]E[X_2] \tag{4.56}$$

Second, the variance of their sums is equal to the sum of their variances:

$$Var(X_1+X_2) = Var(X_1)+Var(X_2) \tag{4.57}$$

In mobile robotics, we often assume the independence of random variables even when this assumption is not strictly true. The simplification that results makes a number of the existing mobile robot-mapping and navigation algorithms tenable, as described in



$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Figure 4.31
The Gaussian function with $\mu = 0$ and $\sigma = 1$. We shall refer to this as the reference Gaussian. The value 2σ is often refereed to as the signal quality; 95.44% of the values are falling within $\pm 2\sigma$.

chapter 5. A further simplification, described in section 4.2.1.2, revolves around one specific probability density function used more often than any other when modeling error: the Gaussian distribution.

4.2.1.2 Gaussian distribution

The Gaussian distribution, also called the *normal distribution,* is used across engineering disciplines when a well-behaved error model is required for a random variable for which no error model of greater felicity has been discovered. The Gaussian has many characteristics that make it mathematically advantageous to other ad hoc probability density functions. It is symmetric around the mean μ. There is no particular bias for being larger than or smaller than μ, and this makes sense when there is no information to the contrary. The Gaussian distribution is also unimodal, with a single peak that reaches a maximum at μ (necessary for any symmetric, unimodal distribution). This distribution also has tails (the value of $f(x)$ as x approaches $-\infty$ and ∞) that only approach zero asymptotically. This means that all amounts of error are possible, although very large errors may be highly improbable. In this sense, the Gaussian is conservative. Finally, as seen in the formula for the Gaussian probability density function, the distribution depends only on two parameters:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{4.58}$$

The Gaussian's basic shape is determined by the structure of this formula, and so the only two parameters required to fully specify a particular Gaussian are its mean, μ, and its

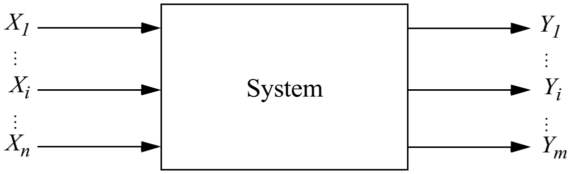


Figure 4.32
Error propagation in a multiple-input multi-output system with n inputs and m outputs.

standard deviation, σ. Figure 4.31 shows the Gaussian function with $\mu = 0$ and $\sigma = 1$.

Suppose that a random variable X is modeled as a Gaussian. How does one identify the chance that the value of X is within one standard deviation of μ? In practice, this requires integration of $f(x)$, the Gaussian function to compute the area under a portion of the curve:

$$Area = \int_{-\sigma}^{\sigma} f(x)dx \tag{4.59}$$

Unfortunately, there is no closed-form solution for the integral in equation (4.59), and so the common technique is to use a Gaussian *cumulative probability table*. Using such a table, one can compute the probability for various value ranges of X:

$$P[\mu - \sigma < X \leq \mu + \sigma] = 0.68;$$

$$P[\mu - 2\sigma < X \leq \mu + 2\sigma] = 0.95;$$

$$P[\mu - 3\sigma < X \leq \mu + 3\sigma] = 0.997.$$

For example, 95% of the values for X fall within two standard deviations of its mean. This applies to *any* Gaussian distribution. As is clear from the above progression, under the Gaussian assumption, once bounds are relaxed to 3σ, the overwhelming proportion of values (and, therefore, probability) is subsumed.

4.2.2 Error propagation: combining uncertain measurements

The probability mechanisms above may be used to describe the errors associated with a single sensor's attempts to measure a real-world value. But in mobile robotics, one often uses a series of measurements, all of them uncertain, to extract a single environmental measure. For example, a series of uncertain measurements of single points can be fused to extract the position of a line (e.g., a hallway wall) in the environment (figure 4.36).

Consider the system in figure 4.32, where X_i are n input signals with a known probability distribution and Y_i are m outputs. The question of interest is: what can we say about

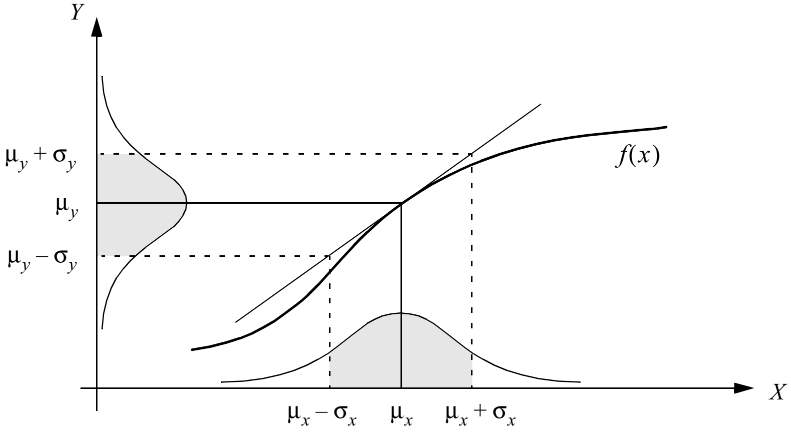


Figure 4.33
One-dimensional case of a nonlinear error propagation problem.

the probability distribution of the output signals Y_i if they depend with known functions f_i upon the input signals? Figure 4.33 depicts the 1D version of this error propagation problem as an example.

The general solution can be generated using the first order Taylor expansion of f_i . The output covariance matrix C_Y is given by the error propagation law:

$$C_Y = F_X C_X F_X^T \tag{4.60}$$

where

C_X = covariance matrix representing the input uncertainties;

C_Y = covariance matrix representing the propagated uncertainties for the outputs;

F_x is the *Jacobian* matrix defined as

$$F_X = \nabla f = \left[\nabla_X \cdot f(X)^T\right]^T = \begin{bmatrix} f_1 \\ : \\ f_m \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial X_1} \cdots \frac{\partial}{\partial X_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial X_1} & \cdots & \frac{\partial f_1}{\partial X_n} \\ : & \cdots & : \\ \frac{\partial f_m}{\partial X_1} & \cdots & \frac{\partial f_m}{\partial X_n} \end{bmatrix} . \tag{4.61}$$

This is also the transpose of the gradient of $f(X)$.

We will not present a detailed derivation here but will use equation (4.60) to solve an example problem in section 4.3.1.1.

4.3 Feature Extraction

An autonomous mobile robot must be able to determine its relationship to the environment by making measurements with its sensors and then using those measured signals. A wide variety of sensing technologies are available, as shown in the previous section. But every sensor we have presented is imperfect: measurements always have error and, therefore, uncertainty associated with them. Therefore, sensor inputs must be used in a way that enables the robot to interact with its environment successfully in spite of measurement uncertainty.

There are two strategies for using uncertain sensor input to guide the robot's behavior. One strategy is to use each sensor measurement as a raw and individual value. Such raw sensor values could, for example, be tied directly to robot behavior, whereby the robot's actions are a function of its sensor inputs. Alternatively, the raw sensor values could be used to update an intermediate model, with the robot's actions being triggered as a function of this model rather than the individual sensor measurements.

The second strategy is to extract information from one or more sensor readings first, generating a higher-level *percept* that can then be used to inform the robot's model and perhaps the robot's actions directly. We call this process *feature extraction*, and it is this next, optional step in the perceptual interpretation pipeline (figure 4.34) that we will now discuss.

In practical terms, mobile robots do not necessarily use feature extraction and scene interpretation for every activity. Instead, robots will interpret sensors to varying degrees depending on each specific functionality. For example, in order to guarantee emergency stops in the face of immediate obstacles, the robot may make direct use of raw forward-facing range readings to stop its drive motors. For local obstacle avoidance, raw ranging sensor strikes may be combined in an occupancy grid model, enabling smooth avoidance of obstacles meters away. For map-building and precise navigation, the range sensor values and even vision sensor measurements may pass through the complete perceptual pipeline, being subjected to feature extraction followed by scene interpretation to minimize the impact of individual sensor uncertainty on the robustness of the robot's mapmaking and navigation skills. The pattern that thus emerges is that, as one moves into more sophisticated, long-term perceptual tasks, the feature extraction and scene interpretation aspects of the perceptual pipeline become essential.

Feature definition. Features are recognizable structures of elements in the environment. They usually can be extracted from measurements and mathematically described. Good features are always perceivable and easily detectable from the environment. We distinguish