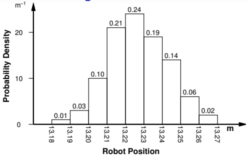
Quantifying Uncertainty Using Information Theory

Andrew Davison
Department of Computing
Imperial College London

May 30, 2010

Probability and Entropy



Using Mackay's notation [3], uncertain knowledge of the value of a parameter x whose possible value lies within the discrete 'alphabet' $A_X = \{a_1, a_2, \ldots\}$ of numeric values is represented probabilistically by a set of mutually-exclusive statements '$x = a_i$', assigned probabilities $P(x = a_i)$ which sum to one. The *information entropy* $H(X)$ of this probability distribution is the expectation of the information content of whichever statement turns out to be true:

$$\begin{aligned} H(X) &= E\left[\log_2 \frac{1}{P(x)}\right] \\ &= \sum_{x \in A_X} P(x) \log_2 \frac{1}{P(x)} \ , \end{aligned}$$

where we use $P(x)$ for $P(x = a_i)$. $H(X)$, in *bits*, is a measure of the average surprise value of the distribution, and its uncertainty.

Joint Entropy

Uncertain knowledge of two parameters x and y, where the extra parameter y is known to have one of a second alphabet of values $B_Y = \{b_1, b_2, \ldots\}$, is represented by a set of statements '$x = a_i, y = b_i$' covering all possible combinations to which the observer assigns probabilities $P(x = a_i, y = b_i)$ which sum to one. This is a joint probability distribution over X and Y, which has a joint entropy representing total uncertainty defined as expected:

$$\begin{aligned} H(XY) &= E\left[\log_2 \frac{1}{P(xy)}\right] \\ &= \sum_{x \in A_X, y \in A_Y} P(xy) \log_2 \frac{1}{P(xy)} \ , \end{aligned}$$

where we have abbreviated $P(x = a_i, y = b_i)$ to $P(xy)$.

Conditional Entropy

Now if the observer were to learn the exact value of one of the uncertain parameters, for instance that $y = b_i$, he would be left with a residual entropy in the distribution over x called the conditional entropy of X given $y = b_i$:

$$H(X|y = b_i) = \sum_{x \in A_X} P(x|y = b_i) \log_2 \frac{1}{P(x|y = b_i)} .$$

If the observer is not told the value of y but considers the expected effect on the entropy of X of each possibility, he can calculate the expected conditional entropy of X given Y; the expected new entropy of X on learning the value of y, without knowing in advance what that value will be:

$$\begin{aligned} H(X|Y) &= E\left[\log_2 \frac{1}{P(x|y)}\right] \\ &= \sum_{x \in A_X, y \in A_Y} P(xy) \log_2 \frac{1}{P(x|y)} . \end{aligned}$$

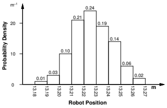
Mutual Information

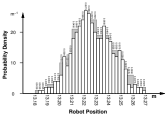
We are led directly to the *mutual information* $I(X;Y)$, defined as the average expected reduction in entropy of one parameter on learning exact value of the other. The reduction in entropy equates to how much *information* learning the value one parameter is expected to give the observer about the other, and $I(X;Y)$ is defined as follows:

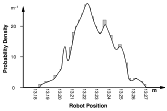
$$I(X;Y) = H(X) - H(X|Y) \ .$$

Note that it is easy to show that $I(X;Y) = I(Y;X)$.

Entropy of Continuous Distributions







The entropy of a probability density function $p(x)$ over an uncertain parameter x which may take a continuum of different values over a range X is not well-defined. This can be seen by splitting the range X into discrete intervals of width δx to form a histogram where the probability that x has a value within each particular bin is approximately $p(x)\delta x$. The entropy of this distribution is:

$$H(X) = \sum_{x \in X} p(x)\delta x \log_2 \frac{1}{p(x)\delta x} \ .$$

On attempting to find the entropy of the continuous distribution by taking the limit $\delta x \to 0$, we find that $H(X)$ diverges since $\log_2 \frac{1}{p(x)\delta x}$ increases by one bit with every halving of the width of δx.

Mutual Information for Continuous Distributions

Still well-defined, however, is the mutual information of two continuous distributions. With discrete bin sizes δx, δy the MI is:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= \sum_{x \in X} p(x)\delta x \log_2 \frac{1}{p(x)\delta x} \\ &\quad - \sum_{x \in X, y \in Y} p(x,y)\delta x \delta y \log_2 \frac{1}{p(x|y)\delta x} \\ &= \sum_{x \in X, y \in Y} p(x,y)\delta x \delta y \log_2 \frac{p(x|y)}{p(x)} , \end{aligned}$$

the δx terms in the logarithm cancelling. Taking the limit $\delta x \to 0, \delta y \to 0$ we obtain the MI of two continuous PDFs:

$$I(X;Y) = \int_{x,y} p(x,y) \log_2 \frac{p(x|y)}{p(x)} dxdy$$

MI in a Multi-Variate Gaussian

Consider vector $\mathbf{a}$ of N uncertain parameters for which we hold a continuous probability density described by a single multi-variate Gaussian. Such a probability distribution is parameterised by a 'state vector' of means $\hat{\mathbf{a}}$ of dimension N and an $N \times N$ covariance matrix P_{aa}. Explicitly, the PDF is:

$$p(\mathbf{a}) = (2\pi)^{-\frac{N}{2}} |\mathrm{P}_{aa}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{a}-\hat{\mathbf{a}})^\top \mathrm{P}_{aa}^{-1}(\mathbf{a}-\hat{\mathbf{a}})} \ .$$

Now let us suppose that $\mathbf{a}$ is divided into two interesting sets of parameters, α and β, of lengths N_α and N_β. We can partition the state vector and covariance matrix as follows:

$$\hat{\mathbf{a}} = \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} \ ; \ \mathrm{P}_{aa} = \begin{bmatrix} \mathrm{P}_{\alpha\alpha} & \mathrm{P}_{\alpha\beta} \\ \mathrm{P}_{\beta\alpha} & \mathrm{P}_{\beta\beta} \end{bmatrix} \ .$$

The mutual information of α and β is as follows:

$$I(\alpha;\beta) = E\left[\log_2 \frac{p(\alpha|\beta)}{p(\alpha)}\right] \ .$$

MI in a Multi-Variate Gaussian

Now distribution $p(\alpha)$ is described trivially by the relevant partitions of the joint state vector and covariance matrix:

$$p(\alpha) = (2\pi)^{-\frac{N_\alpha}{2}} |\mathtt{P}_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathtt{P}_{\alpha\alpha}^{-1}(\alpha-\hat{\alpha})} \ .$$

To obtain $p(\alpha|\beta)$, we use the general formula for conditioning one partition of a state vector and covariance with respect to another, as presented very clearly recently by Eustice *et al.*[2]. If we learn the exact values of all elements of β, the state vector and covariance of α can be updated to:

$$\begin{aligned}
\hat{\alpha}' &= \hat{\alpha} + \mathtt{P}_{\alpha\beta}\mathtt{P}_{\beta\beta}^{-1}(\beta - \hat{\beta}) \\
\mathtt{P}'_{\alpha\alpha} &= \mathtt{P}_{\alpha\alpha} - \mathtt{P}_{\alpha\beta}\mathtt{P}_{\beta\beta}^{-1}\mathtt{P}_{\beta\alpha} \ .
\end{aligned}$$

Note that this is essentially the update step of the Kalman Filter, where usually α would represent the state of the system in question and β a set of measurements. So:

$$p(\alpha|\beta) = (2\pi)^{-\frac{N_\alpha}{2}} |\mathtt{P}'_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathtt{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')} \ ,$$

MI in a Multi-Variate Gaussian

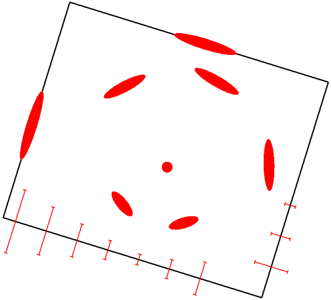
and, using parts of an argument given by Manyika [4]:

$$\begin{aligned}
I(\alpha;\beta) &= E\left[\log_2 \frac{|\mathrm{P}'_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathrm{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')}}{|\mathrm{P}_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathrm{P}^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha})}}\right] \\
&= \log_2 \frac{|\mathrm{P}_{\alpha\alpha}|^{\frac{1}{2}}}{|\mathrm{P}'_{\alpha\alpha}|^{\frac{1}{2}}} \\
&\quad + \frac{1}{ln2} E\left[-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathrm{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')\right] \\
&\quad + \frac{1}{ln2} E\left[\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathrm{P}^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha})\right] \\
&= \frac{1}{2}\log_2 \frac{|\mathrm{P}_{\alpha\alpha}|}{|\mathrm{P}'_{\alpha\alpha}|} + \frac{1}{ln2}\left(-\frac{1}{2}+\frac{1}{2}\right) \\
&= \frac{1}{2}\log_2 \frac{|\mathrm{P}_{\alpha\alpha}|}{|\mathrm{P}_{\alpha\alpha} - \mathrm{P}_{\alpha\beta}\mathrm{P}^{-1}_{\beta\beta}\mathrm{P}_{\beta\alpha}|}\ .
\end{aligned}$$

Feature Search in Model-Based Tracking

As in [1]:

- Object state $\mathbf{x}$ and measurement candidates $\mathbf{z}_i = \mathbf{h}_i(\mathbf{x}) + \mathbf{n}_m$



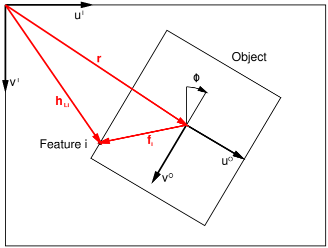
$$\hat{\mathbf{x}}_m = \begin{pmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{z}}_1 \\ \hat{\mathbf{z}}_2 \\ \vdots \end{pmatrix}, \; \mathrm{P}_{\mathbf{x}_m} = \begin{bmatrix} \mathrm{P}_x & \mathrm{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top & \mathrm{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top & \cdots \\ \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathrm{P}_x & \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathrm{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top + \mathrm{R}_1 & \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathrm{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top & \cdots \\ \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathrm{P}_x & \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathrm{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top & \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathrm{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top + \mathrm{R}_2 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

Measurement Information Matrix

$$\mathtt{I}(\mathbf{x}_m) = \begin{bmatrix} * & I(\mathbf{x};\mathbf{z}_1) & I(\mathbf{x};\mathbf{z}_2) & \dots \\ I(\mathbf{z}_1;\mathbf{x}) & * & I(\mathbf{z}_1;\mathbf{z}_2) & \dots \\ I(\mathbf{z}_2;\mathbf{x}) & I(\mathbf{z}_2;\mathbf{z}_1) & * & \dots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

- MI between each measurement and state
- MI between each pair of measurements

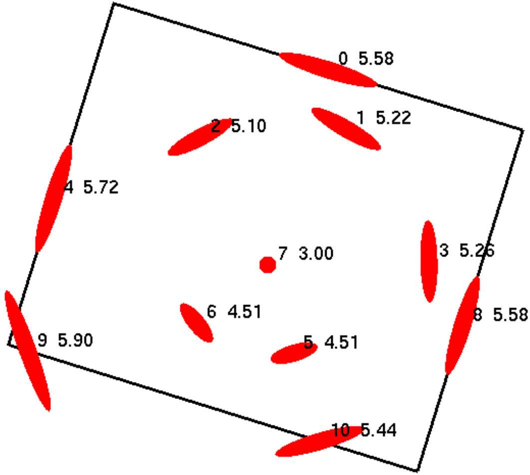
Tracking a Translating, Rotating Object in 2D



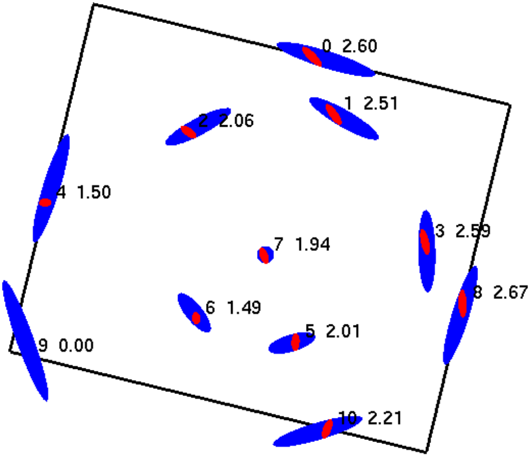
$$\hat{\mathbf{x}} = \begin{pmatrix} u \\ v \\ \phi \end{pmatrix} = \begin{pmatrix} 320.0 \\ 260.0 \\ 0.3 \end{pmatrix} \ , \ \mathbf{P}_x = \begin{bmatrix} 7.0 & 0.0 & 0.0 \\ 0.0 & 7.0 & 0.0 \\ 0.0 & 0.0 & 0.007 \end{bmatrix} .$$

- 2D point feature measurements
- 1D edge feature measurement
- One pixel measurement uncertainty

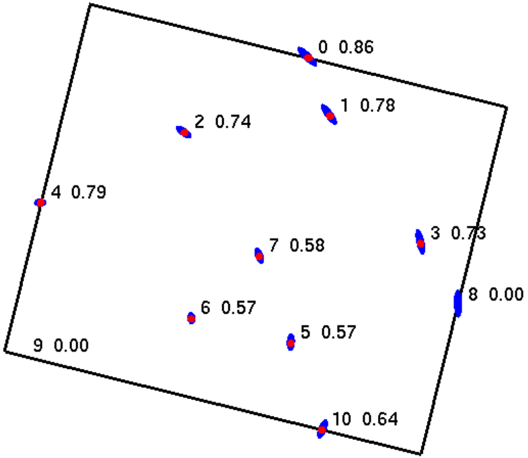
Information-Guided Search with Point Features



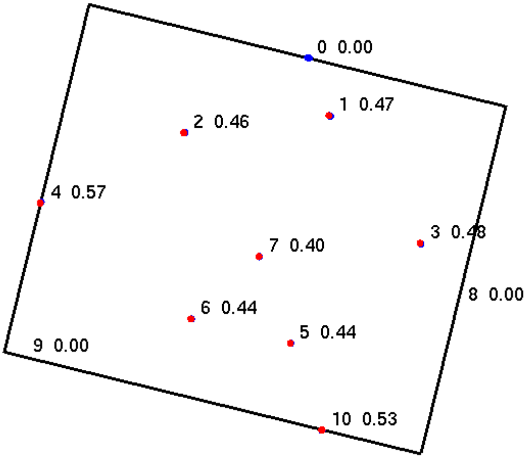
Information-Guided Search with Point Features



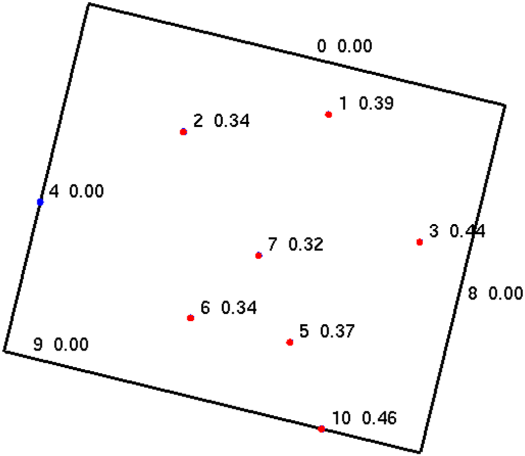
Information-Guided Search with Point Features



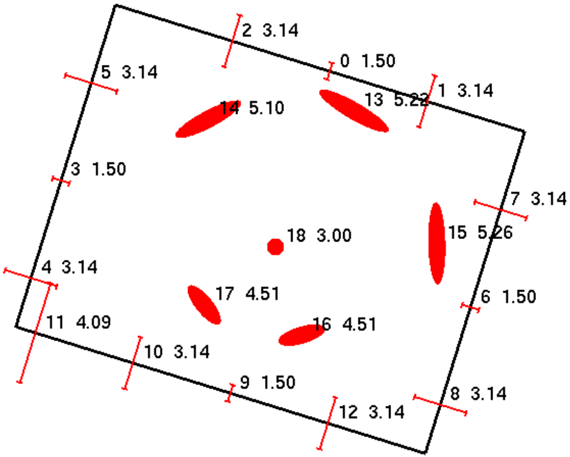
Information-Guided Search with Point Features



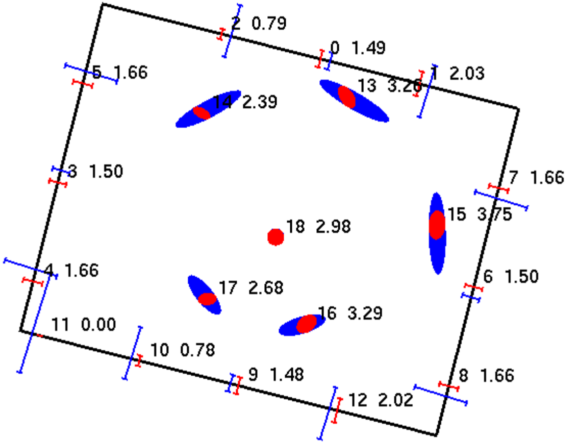
Information-Guided Search with Point Features



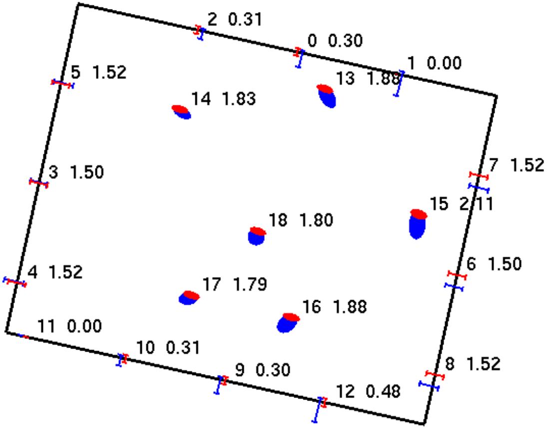
Information-Guided Search with Edge Features



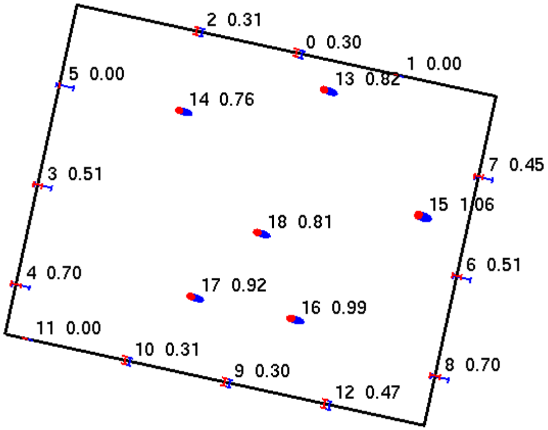
Information-Guided Search with Edge Features



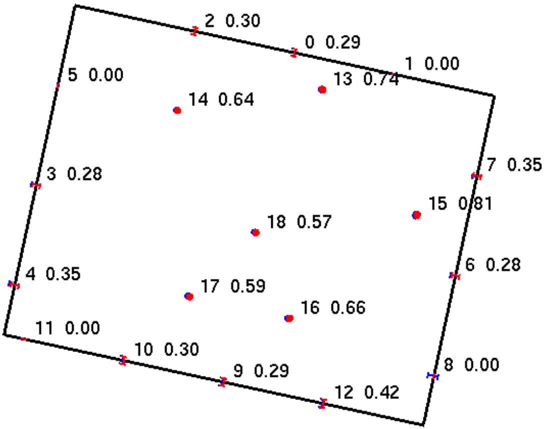
Information-Guided Search with Edge Features



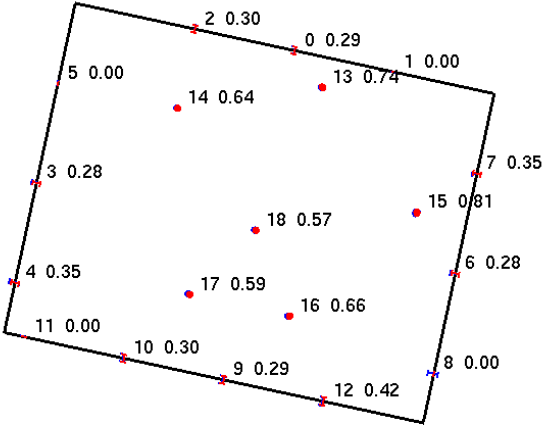
Information-Guided Search with Edge Features



Information-Guided Search with Edge Features



Information-Guided Search with Edge Features



References

A. J. Davison.
Active search for real-time vision.
In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2005.

R. M. Eustice, H. Singh, and J. J. Leonard.
Exactly sparse delayed state filters.
In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

D. Mackay.
Information Theory, Inference and Learning Algorithms.
Cambridge University Press, 2003.

J. Manyika.
An Information-Theoretic Approach to Data Fusion and Sensor Management.
PhD thesis, University of Oxford, 1993.