Active Search for Real-Time Vision

Andrew J. Davison
Department of Computing, Imperial College London, 180 Queen's Gate, SW7 2AZ, London, UK
ajd@doc.ic.ac.uk
http://www.doc.ic.ac.uk/~ajd/

Abstract

In most cases when information is to be extracted from an image, there are priors available on the state of the world and therefore on the detailed measurements which will be obtained. While such priors are commonly combined with the actual measurements via Bayes' rule to calculate posterior probability distributions on model parameters, their additional value in guiding efficient image processing has almost always been overlooked. Priors tell us *where to look* for information in an image, how much computational effort we can expect to expend to extract it, and of how much utility to the task in hand it is likely to be. Such considerations are of importance in all practical real-time vision systems, where the processing resources available at each frame in a sequence are strictly limited — and it is exactly in high frame-rate real-time systems such as trackers where strong priors are most likely to be available.

In this paper, we use Shannon information theory to analyse the fundamental *value* of measurements using mutual information scores in absolute units of *bits*, specifically looking at the overwhelming case where uncertainty can be characterised by Gaussian probability distributions. We then compare these measurement values with the computational cost of the image processing required to obtain them. This theory puts on a firm footing for the first time principles of 'active search' for efficient guided image processing, in which candidate features of possibly different types can be compared and selected automatically for measurement.

1 Introduction

In most modern geometrical vision algorithms, image processing is treated as a separable step — a bottom-up operation applied uniformly to incoming images, detecting or re-finding various 'features'. The results are fed forward to estimation processing which calculates quantities of interest like the global position of an object or camera motion and structure. This methodology is recently being reinforced by the success of new types of detector such as SIFT [7] which can find features with a high degree of uniqueness and invariance, matchable despite weak priors. New images are by default scanned exhaustively by such a detector, and the raw image data need not again be examined.

Here we argue however that in many cases in *real-time vision* it makes more sense to keep image processing 'in the loop', top-down and adaptive. Let us be clear why it should 'make sense': only if the limited processing available in a real-time budget can be used more effectively by this guided approach than by bottom-up methods. The top-down approach to visual processing has often been called *active vision*, and been associated with moving robotic cameras (e.g. [10]). Controlling a camera always requires up-front decisions to be taken about where to look next depending on the task in hand. The active approach is however equally applicable to passive cameras in any real-time system, encompassing wide fields such as tracking, robot guidance or SLAM/SFM. Top-down decisions can be made about *where in each image to focus processing resources*,

It has long been understood that guided search is of benefit in real-time tracking: rather than searching for a moving target from scratch in each image of a sequence, knowledge of motion continuity is used to initialise a search from the image location previously found, or more sophisticated predictions made based on models of dynamics. The particle filter-based visual tracking methods introduced by Isard and Blake [4] for instance achieve this in a particularly satisfactory way by performing local image searches in the neighbourhoods of the distribution of particles representing the probability distribution of the location of a target after motion prediction — no attention is paid to parts of the image where the probability of the target's existence is low. Trackers using the Extended Kalman Filter to propagate Gaussian probability distributions often work in a similar way by restricting search to "gated" elliptical regions around the predicted target location, of size determined by the estimate covariance and a chosen number of standard deviations.

However, none of these methods has told anything like the whole story of active search. In [4], a local search is performed close to each particle position by examining the neighbourhoods of a set of fixed measurement sites spaced evenly around the contour of the tracked object model. No consideration is given to the fact that some sites are likely to

be much more discriminitive than others, or that many will give redundant information — and no calculation made of how many features are needed to give good performance.

Overall, there has been surprisingly little theoretical investigation of the mechanisms by which active search should be achieved. Davison [2] used a simple innnovation covariance criterion to great effect to choose between candidate point measurements in a real-time SLAM system, but this lacked genericity or justification. In this paper we promote mutual information as surely the absolute measure of the value of image measurements. As the use of Bayesian probability theory has become standard in computer vision, its natural extension in information theory has been criminally underused. We show with full theoretical support that mutual information is a tool of wide use, and present the foundations on which the first rigorous algorithms for active search can be constructed.

2 Mutual Information

In this section we will derive from first principles the mutual information (MI) of two uncertain vectors of parameters described by a single joint Gaussian distribution. We will go into detail because we feel that this is a result of wide importance which has not previously been made explicit.

2.1 Probability and Entropy

Using the notation of Mackay [8], an observer's 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 of the form '$x = a_i$', 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:

$$H(X) = E\left[\log_2 \frac{1}{P(x)}\right] \quad (1)$$

$$= \sum_{x \in A_X} P(x) \log_2 \frac{1}{P(x)} , \quad (2)$$

where we use shorthand $P(x)$ for the more precise $P(x = a_i)$. $H(X)$, in bit units, is a measure of the average surprise value of the distribution, and therefore the uncertainty in the observer's state of knowledge about the parameter.

2.2 Joint Probability Distributions, Conditional Entropy and Mutual Information

Uncertain knowledge of two parameters x and y, where the extra paramater 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:

$$H(XY) = E\left[\log_2 \frac{1}{P(xy)}\right] \quad (3)$$

$$= \sum_{x \in A_X, y \in A_Y} P(xy) \log_2 \frac{1}{P(xy)} , \quad (4)$$

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

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 $y = b_i$:

$$H(X|y = b_i) = \sum_{x \in A_X} P(x|y = b_i) \log_2 \frac{1}{P(x|y = b_i)} . \quad (5)$$

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:

$$H(X|Y) = E\left[\log_2 \frac{1}{P(x|y)}\right] \quad (6)$$

$$= \sum_{x \in A_X, y \in A_Y} P(xy) \log_2 \frac{1}{P(x|y)} . \quad (7)$$

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:

$$I(X;Y) = H(X) - H(X|Y) . \quad (8)$$

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

2.3 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:

$$H(X) = \sum_{x \in X} p(x)\delta x \log_2 \frac{1}{p(x)\delta x} . \quad (9)$$

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

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) && (10)\\ &= \sum_{x\in X} p(x)\delta x \log_2 \frac{1}{p(x)\delta x} && (11)\\ &\quad - \sum_{x\in X, y\in Y} p(x,y)\delta x\delta y \log_2 \frac{1}{p(x|y)\delta x} \\ &= \sum_{x\in X, y\in Y} p(x,y)\delta x\delta y \log_2 \frac{p(x|y)}{p(x)}\ , && (12)\end{aligned}$$

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:

$$I(X;Y) = \int_{x,y} p(x,y) \log_2 \frac{p(x|y)}{p(x)} dxdy \qquad (13)$$

2.4 MI of a Multi-Variate Gaussian

The special characteristics and wide applicability of the Gaussian as the central distribution of probability theory [5] are so well established that we have no qualms about becoming Gaussian-specific at this point. 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 $\mathtt{P}_{aa}$. Explicitly, the PDF is:

$$p(\mathbf{a}) = (2\pi)^{-\frac{N}{2}} |\mathtt{P}_{aa}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\mathbf{a}-\hat{\mathbf{a}})^\top \mathtt{P}_{aa}^{-1}(\mathbf{a}-\hat{\mathbf{a}})}\ . \qquad (14)$$

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:

$$\hat{\mathbf{a}} = \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix}\ ;\ \mathtt{P}_{aa} = \begin{bmatrix} \mathtt{P}_{\alpha\alpha} & \mathtt{P}_{\alpha\beta} \\ \mathtt{P}_{\beta\alpha} & \mathtt{P}_{\beta\beta} \end{bmatrix}\ . \qquad (15)$$

The mutual information of α and β is as follows:

$$I(\alpha;\beta) = E\left[\log_2 \frac{p(\alpha|\beta)}{p(\alpha)}\right]\ . \qquad (16)$$

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

$$p(\alpha) = (2\pi)^{-\frac{N_\alpha}{2}} |\mathtt{P}_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathtt{P}_{\alpha\alpha}^{-1}(\alpha-\hat{\alpha})}\ . \qquad (17)$$

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.*[3]. If we learn the exact values of all elements of β, the state vector and covariance of α can be updated to:

$$\begin{aligned} \hat{\alpha}' &= \hat{\alpha} + \mathtt{P}_{\alpha\beta}\mathtt{P}_{\beta\beta}^{-1}(\beta - \hat{\beta}) && (18)\\ \mathtt{P}'_{\alpha\alpha} &= \mathtt{P}_{\alpha\alpha} - \mathtt{P}_{\alpha\beta}\mathtt{P}_{\beta\beta}^{-1}\mathtt{P}_{\beta\alpha}\ . && (19)\end{aligned}$$

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:

$$p(\alpha|\beta) = (2\pi)^{-\frac{N_\alpha}{2}} |\mathtt{P}'_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathtt{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')}\ , \qquad (20)$$

and, using parts of an argument given by Manyika [9]:

$$\begin{aligned} I(\alpha;\beta) &= E\left[\log_2 \frac{|\mathtt{P}'_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathtt{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')}}{|\mathtt{P}_{\alpha\alpha}|^{-\frac{1}{2}} e^{-\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathtt{P}_{\alpha\alpha}^{-1}(\alpha-\hat{\alpha})}}\right] \\ &= \log_2 \frac{|\mathtt{P}_{\alpha\alpha}|^{\frac{1}{2}}}{|\mathtt{P}'_{\alpha\alpha}|^{\frac{1}{2}}} \\ &\quad + \frac{1}{ln2} E\left[-\frac{1}{2}(\alpha-\hat{\alpha}')^\top \mathtt{P}'^{-1}_{\alpha\alpha}(\alpha-\hat{\alpha}')\right] \\ &\quad + \frac{1}{ln2} E\left[\frac{1}{2}(\alpha-\hat{\alpha})^\top \mathtt{P}_{\alpha\alpha}^{-1}(\alpha-\hat{\alpha})\right] && (21)\\ &= \frac{1}{2}\log_2 \frac{|\mathtt{P}_{\alpha\alpha}|}{|\mathtt{P}'_{\alpha\alpha}|} + \frac{1}{ln2}\left(-\frac{1}{2}+\frac{1}{2}\right) && (22)\\ &= \frac{1}{2}\log_2 \frac{|\mathtt{P}_{\alpha\alpha}|}{|\mathtt{P}_{\alpha\alpha} - \mathtt{P}_{\alpha\beta}\mathtt{P}_{\beta\beta}^{-1}\mathtt{P}_{\beta\alpha}|}\ . && (23)\end{aligned}$$

3 Feature Search in Tracking

We now apply the theory developed above to the real-time tracking problem, presented first generically and then with a specific tracking example which will be analysed in detail.

In model-based tracking, a moving object is modelled by a finite vector of parameters $\mathbf{x}$ describing the configurations of its degrees of freedom, as well potentially as dynamics and other factors of interest. Note that this is the case whether the object moves rigidly, is articulated or is more generally deformable. An observer's uncertain knowledge of the state of the object at any instant is represented by a multi-dimensional probability density in the space of $\mathbf{x}$. When this probability density can be characterised as a multi-variate Gaussian, the observer's knowledge is captured by a state vector $\hat{\mathbf{x}}$ and covariance matrix $\mathtt{P}_x$.

Now we assume that the observer is in possession of a sensor or sensors which can make measurements of certain aspects of the object. Still speaking generally, we can call each of these measurable aspects a *feature*, of which a measurement attempt yields the vector of parameters $\mathbf{z}_i$, whose form can be broken down as $\mathbf{z}_i = \mathbf{h}_i(\mathbf{x}) + \mathbf{n}_m$. The function $\mathbf{h}_i(\mathbf{x})$ describes the functional relationship between the

expected measurement and the object state as far as understood via the models used of the object and sensor, and $\mathbf{n}_m$ is a vector representing unmodelled effects (sometimes called noise) which mean that an actual measurement will deviate from this. The form of $\mathbf{h}_i(\mathbf{x})$ depends on the type of feature, relative pose of sensor and object, sensor characteristics and so on. Assuming that the models have no systematic bias, the distribution of vector $\mathbf{n}_m$ will have mean zero, and in this analysis we assume that it is more specifically described by a multi-variate Gaussian with covariance $\mathtt{R}_i$ which is independent for each measurement.

At a given point in time, the observer will have a choice of which of the features available to measure, and we define the vector $\mathbf{x}_m$ which stacks the object state and candidate measurement vectors. Given his knowledge of the state of the object, the observer's knowledge of this vector is described by its mean vector and covariance:

$$\hat{\mathbf{x}}_m = \begin{pmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{z}}_1 \\ \hat{\mathbf{z}}_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_v \\ \mathbf{h}_1(\hat{\mathbf{x}}) \\ \mathbf{h}_2(\hat{\mathbf{x}}) \\ \vdots \end{pmatrix} \tag{24}$$

$$\mathtt{P}_{\mathbf{x}_m} = \begin{bmatrix} \mathtt{P}_x & \mathtt{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top & \mathtt{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top & \cdots \\ \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathtt{P}_x & \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top + \mathtt{R}_1 & \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top & \cdots \\ \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathtt{P}_x & \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_1}{\partial \mathbf{x}}^\top & \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_2}{\partial \mathbf{x}}^\top + \mathtt{R}_2 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} \tag{25}$$

The $\hat{\mathbf{z}}_i$ parts of $\hat{\mathbf{x}}_m$ contain the observer's best guess or *prediction* of the values of observations he would obtain if measurements were attempted now, and $\mathtt{P}_{\mathbf{x}_m}$ describes the uncertainty in this prediction. Importantly, generally $\mathtt{P}_{\mathbf{x}_m}$ will not be block-diagonal but a full matrix giving information on correlations between the predicted measurements of different features. The blocks along the top or left of $\mathtt{P}_{\mathbf{x}_m}$ give the correlation between each predicted measurement and the object state estimate itself, and the off-diagonal blocks the correlations between different measurements.

We now define the *measurement information matrix* $\mathtt{I}(\mathbf{x}_m)$ of MI scores between object state estimate and feature measurement predictions:

$$\mathtt{I}(\mathbf{x}_m) = \begin{bmatrix} * & I(\mathbf{x}; \mathbf{z}_1) & I(\mathbf{x}; \mathbf{z}_2) & \cdots \\ I(\mathbf{z}_1; \mathbf{x}) & * & I(\mathbf{z}_1; \mathbf{z}_2) & \cdots \\ I(\mathbf{z}_2; \mathbf{x}) & I(\mathbf{z}_2; \mathbf{z}_1) & * & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} \tag{26}$$

Each scalar element $I(\alpha; \beta)$ is calculated as in Equation 23 from the elements of $\mathtt{P}_{\mathbf{x}_m}$ defined in Equation 25. Matrix $\mathtt{I}(\mathbf{x}_m)$ has dimension $(N+1) \times (N+1)$, where N is the number of measurements defined, and is symmetric thanks to the symmetric property of mutual information. The diagnonal elements marked '$*$' are left undefined since a vector's MI with itself is meaningless.

The elements of $\mathtt{I}(\mathbf{x}_m)$ neatly summarise the role information theory can play in active search. Each element $I(\mathbf{x}; \mathbf{z}_i)$ is the MI between a measurement and the object state, describing the expected information about the state the measurement is expected to furnish — the *value* of the measurement in pure information-theoretic terms. Elements $I(\mathbf{z}_i; \mathbf{z}_j)$ are the MI between individual measurements, describing to what degree they are independent or correlated, and therefore whether two are redundant. All elements of $\mathtt{I}(\mathbf{x}_m)$ are directly and appealingly in *bit* units: in our opinion it is extremely enlightening to see the yes/no concept of the bit we are all familiar with from basic computing theory available for our use here in a practical Bayesian treatment of a computer vision problem with continuous variables. As we shall see, this framework is therefore completely generic and allows direct comparison of the values of measurements of different types and different dimensionalities.

3.1 Tracking with Point and Edge Features

We now consider a specific visual tracking problem. An object is assumed to move and rotate in a plane which is fronto-parallel to a single observing camera. We parameterise the location of the centre of the object in image coordinates (u^I, v^I) relative to the top-left image corner, and its orientation with the angle ϕ in radians, to give state vector:

$$\mathbf{x} = \begin{pmatrix} u^I \\ v^I \\ \phi \end{pmatrix} . \tag{27}$$

The locations $\mathbf{f}_i^O$ of known measurable features are defined in the object coordinate frame O, and we deduce the component equation $\mathbf{h}_{Li}^I = \mathbf{r}^I + \mathtt{R}^{IO}\mathbf{f}_i^O$ for the location of the feature in the image, where $\mathtt{R}^{IO}$ is the rotation matrix:

$$\mathtt{R}^{IO} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} . \tag{28}$$

3.1.1 Point Features

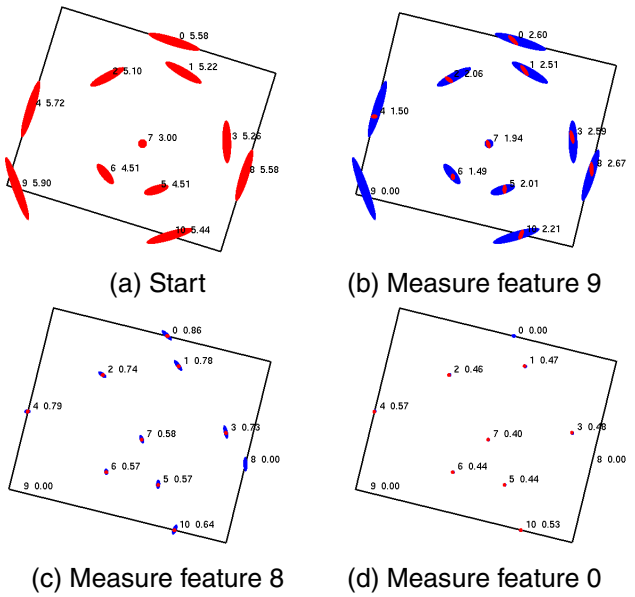
We consider two types of feature: firstly, for a point feature (a "corner" or other local salient marking on the object surface), we will directly measure its two-dimensional image location, so in this case the measurement function is:

$$\mathbf{h}_i = \mathbf{h}_{Li}^I = \begin{pmatrix} u_I + f_{iu}^O \cos\phi - f_{iv}^O \sin\phi \\ v_I + f_{iu}^O \sin\phi + f_{iv}^O \cos\phi \end{pmatrix} , \tag{29}$$

with diagonal two-dimensional measurement covariance

$$\mathtt{R}_i = \begin{bmatrix} \sigma_m^2 & 0 \\ 0 & \sigma_m^2 \end{bmatrix} \tag{30}$$

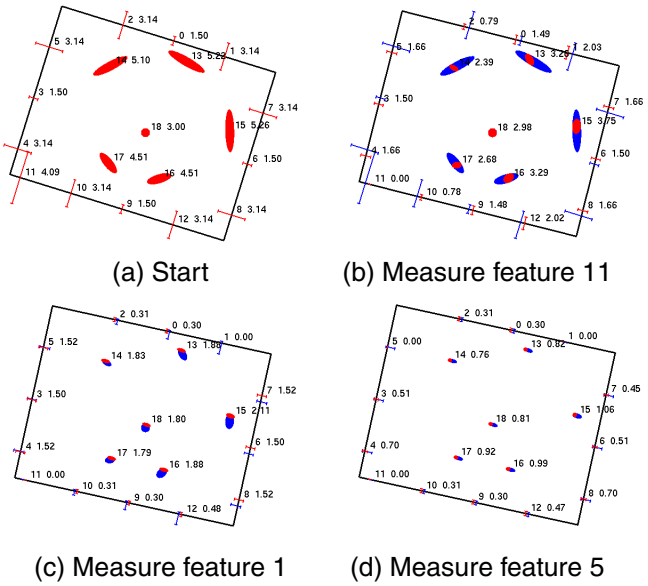
representing independent uncertainty in horizontal and vertical feature location measurements with $\sigma_m = 1$ pixel.



(a) Start (b) Measure feature 9

(c) Measure feature 8 (d) Measure feature 0

Figure 1: Choosing point measurements by mutual information with state. Three standard deviation uncertainty bounds are shown before and after measurement in blue and red respectively. Alongside the integer label of each measurement candidate is its MI with the object state.



(a) Start (b) Measure feature 11

(c) Measure feature 1 (d) Measure feature 5

Figure 2: Choosing edge measurements by MI with state. Some point features are shown, but not measured here.

3.1.2 Edge Features

An edge feature is a site on the object where a one-dimensional textural gradient can be measured, and is defined in the object frame both by the 2D measurement site $\mathbf{f}_i^O$ and angle ϵ_i the gradient direction makes with the object's u_O axis. An edge measurement reports the scalar distance to the edge along this direction from its predicted position, with scalar measurement covariance $\mathtt{R}_i = \left[\sigma_m^2\right]$.

4 Choosing Measurements by Information Value

In simulation, we set up the model of Section 3.1 to represent a snapshot in tracking with object state and covariance:

$$\hat{\mathbf{x}} = \begin{pmatrix} 320.0 \\ 260.0 \\ 0.3 \end{pmatrix}, \mathbf{P}_x = \begin{bmatrix} 7.0 & 0.0 & 0.0 \\ 0.0 & 7.0 & 0.0 \\ 0.0 & 0.0 & 0.007 \end{bmatrix}. \quad (31)$$

This represents a fairly uncertain object state estimate, probably directly after the application of a motion model (here the position and orientation estimates are currently uncorrelated and $\mathbf{P}_x$ is diagonal, though this will not generally be the case). First, in Figure 1 we consider the case where the only features available for measurement are points.

In Figure 1(a) we see the scenario at the start of the measurement process, where the state estimate and covariance of Equation 31 manifest themselves as differently shaped ellipses in image space representing three standard deviation confidence bounds. The shape of each ellipse is defined by the corresponding $\frac{\partial \mathbf{h}_i}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_i}{\partial \mathbf{x}}^\top + \mathtt{R}_i$ diagonal block of $\mathtt{P}_{\mathbf{x}_m}$ and the chosen number of standard deviations. A basic tenet of our approach to active search is that an attempt to measure a particular feature should be made by seeking an appearance match only within this bounded search region, as has been proven in real-time applications such as [2].

Each feature is at this stage a measurement candidate, labelled with its MI with the object state (from the top row of $\mathtt{I}(\mathbf{x}_m)$). Seeing that feature 9 has the highest MI score of 5.90 bits, and is therefore the single candidate with the highest value, we select it for first measurement, and simulate a search and successful match to this feature at a location within the ellipse. The state after the measurement is shown in (b), where we have updated the object state, remaining feature search regions and MI scores appropriately by updating $\hat{\mathbf{x}}_m$ and $\mathtt{P}_{\mathbf{x}_m}$ according to Equations 18 and 19. The first observation to make here is the dramatic reduction in search region size and MI scores for the other features which has been achieved by the measurement of 9. The highest remaining candidate by MI with the object state is feature 8 with a score of 2.67 bits, so this is selected for next measurement with the outcome displayed in (c). This has a further significant effect, with no feature now reporting an MI greater than 1 bit. The highest candidate left, feature 0

with score 0.86 bits is chosen for the third measurement. As the object position becomes yet better known, the remaining MI scores continue to shrink after this measurement in (d), but we see diminishing returns as only tiny changes can now be observed. The search ellipses all approach the uniform circular shape defined by measurement uncertainty $\mathtt{R}_i$.

In Figure 2, we repeat the object measurement exercise with the same starting position and uncertainty but now by making observations of edge features. For reference, some of the point features from the previous experiment are also displayed. Figure 2(a) is the starting condition, where the initial uncertainty translates into differing search lengths and MI scores for the various edge features around the object's boundary. The first thing to note is that as expected edge features provide on the whole less information than point features, though some edges in advantageous positions actually have a higher MI than some of the point features. The first measurement in (b) is of edge 11 of MI 4.09 bits, and this is followed in (c) by edge 1 of 2.03 bits. The object position is still not well locked down, however, as seen by the fact that the point features still have MI scores around 2 bits. Measuring feature 5, one of several candidate edges at 1.52 bits removes most of the remaining uncertainty and the situation in (d) shows search regions which have collapsed to close to measurement uncertainty.

These simulations confirm what we would expect — that measuring two well-spaced point features or three well-chosen edges is sufficient to pin down well the location of an object moving and rotating in 2D. To see this fact demonstrated in this way using information theory, however, is in our opinion much more convincing than the algebraic arguments along the lines of '3 unknowns are determined by three equations' commonly used in geometrical computer vision papers — a line of reasoning Jaynes dismisses as a 'folk theorem' in his authoritative book on Bayesian probability theory [5]. When the points or edges are not so well-chosen as in these examples, we will see that the same number of measurements will not determine the state nearly so well, in a way often referred to vaguely as 'degeneracy', but which we can quantify precisely using information theory.

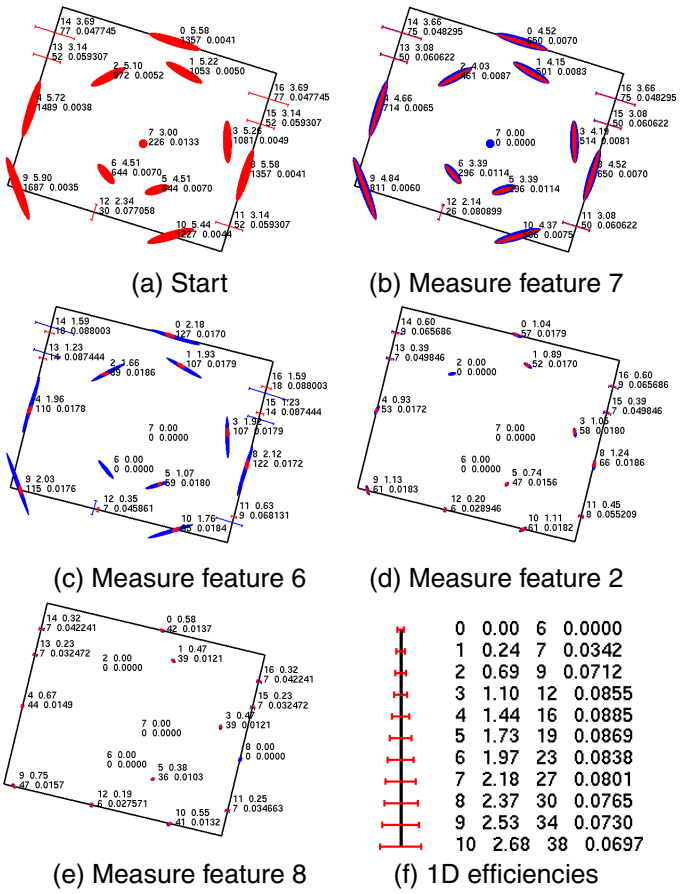


Figure 3: (a)–(e) Selecting point measurements on the basis of information efficiency. Each numbered point feature is labelled with its MI with the state, search ellipse area in square pixels and information efficiency score. Some edge features are displayed for reference, with search length and corresponding efficiency scores. (f) Edge searches for a linear object with only angular uncertainty.

5 Measurement Value versus Cost

The active search strategy of Section 4 selected features one by one to achieve the most rapid decrease in uncertainty per measurement, but let us remember that the goal of active search is to maximise computational efficiency — this simple method neglects the fact that some measurements require more computation to acquire than others. In this section we reconsider our active search strategy if we trade off information value against image processing cost.

Searching an image for a match to a known feature template requires essentially an amount of computation proportional to the search region size: the template must simply be tried exhaustively in all possible positions. Faster gradient descent search, such as in the Lucas/Kanade feature tracker [1], can only be used in the restrictive case where the search region is smaller than the 'basin of attraction' of the match function minimum. We could take advantage of gradient descent search when a search region is large using multiple starting points uniformly spread across the region. However, such a technique can only improve on the computational requirements of exhaustive search by a constant factor and proportionality to the search region area remains.

In the case of a point feature, therefore, the image processing cost of a measurement will be proportional to the search ellipse area $A = \pi N_\sigma^2 \sqrt{|\mathtt{P}_i|}$, while for an edge it will be proportional to the search length $L = 2N_\sigma \sqrt{|\mathtt{P}_i|}$, where

in both cases $\mathtt{P}_i$ is the relevant diagonal block of $\mathtt{P}_{\mathbf{x}_m}$. If the constants of proportionality are known, which will depend on the precise feature operators, we can compare edges directly with points. We can make comparisons between features of the same type without knowing the constants.

In Figure 3 we show a sequence of point feature measurements starting from the same object position and uncertainty but now selected based on the *information efficiency* ratio calculated as the MI of the feature with the state divided by the area of its search ellipse (note that some edge features are also drawn, but only for reference). It can be seen that this leads to a measurement order very different from Section 4. The first feature chosen is 7 in the centre, which has a low MI score of 3.00 but a search area of only 226 square pixels, leading to the highest efficiency score of 0.0133. Making this measurement does not have a big impact on the object state estimate, but its value is in cheaply reducing the size of the search regions for the other point features significantly as can be seen in (b). Subsequently features 6, 2 and 8 are chosen, again each not the most informative measurement candidate but 'good value' in reducing uncertainty for small search cost. After these three measurements, in (e) a situation has been reached where the highest point MI is feature 9 at 0.75 bits, which is the ballpark of an acceptable object position estimate but at the cost of only $226 + 296 + 69 + 66 = 657$ pixel search operations — for comparison, the cost of choosing most informative feature 9 for the first measurement is 1687 operations alone.

It may appear initially that this criterion is acting in exactly the opposite way to the previous one, choosing always the least informative features to measure first, but this is not the case — in Figure 3(c) and (d) for instance it can be seen that feature 2 is selected despite having a larger search region than other features. In fact, it turns out that there is an optimal size of search region for which the efficiency score peaks. This can be seen most clearly in edge search example (f), where an object is assigned dominantly angular uncertainty and edge searches occur perpendicular to a line radially extending from the rotation centre. Each feature is shown with an integer label, MI score, search length and efficiency score, and a peak in efficiency is seen at feature 4. Analytically, this peak occurs at the maximum with respect to $\sigma_p^2 = \frac{\partial \mathbf{h}_i}{\partial \mathbf{x}} \mathtt{P}_x \frac{\partial \mathbf{h}_i}{\partial \mathbf{x}}^\top$, the prior uncertainty in the position of a feature before measurement, of the function for efficiency E_i:

$$E_i = \frac{\frac{1}{2}\log_2((\sigma_p^2 + \sigma_m^2)/\sigma_m^2)}{2N_\sigma\sqrt{\sigma_p^2 + \sigma_m^2}} \,. \tag{32}$$

6 The Cost of Making Decisions

So information theory can lead us to efficient measurement strategies which maximise information transfer per image processing operation. Before each measurement, a decision can be made about observing which feature will be most beneficial, and all image processing can be pre-planned and guided in this manner. However, this analysis neglects the fact that there is a computational cost associated with actually *making the decisions* in active search.

In Sections 4 and 5, deciding which feature to measure next requires the calculation of the MI of each measurement with the state — we must compute the N elements of the top row of measurement information matrix $\mathtt{I}(\mathbf{x}_m)$. Once the measurement has been chosen and carried out, its effect must be transmitted to the rest of the state/measurement vector $\mathbf{x}_m$ and its covariance $\mathtt{P}_{\mathbf{x}_m}$ so that the next measurement can be chosen. This state and covariance update using Equations 18 and 19 is an operation of order $O(N^2)$.

Depending on the size of N and the efficiency of matrix calculations available, it is possible that this computational cost before each measurement can be made may become comparable with the cost of image processing operations — and we must then start to question whether the efficiency in pure measurement terms that we are gaining through this active approach merits the extra probabilistic calculations involved. This is really the crux of the question of top-down vs. bottom-up processing, and we hope in future work to be able to describe their meeting point theoretically.

In the meantime, an alternative to selecting feature measurements one by one in between updates suggests itself thanks to our earlier definition of the full matrix $\mathtt{I}(\mathbf{x}_m)$, not just its top row. This matrix can be used to select several candidate features at once. The following table is a selection of the elements of $\mathtt{I}(\mathbf{x}_m)$ at the start of the simulation in Section 4, showing entries relating to the object state $\mathbf{x}$ and some of the most informative features 0, 4, 8, 9 and 10.

	x	**0**	**4**	**8**	**9**	**10**
x	$*$	5.58	5.71	5.58	5.89	5.43
0	5.58	$*$	3.21	**3.14**	**2.98**	2.68
4	5.71	3.21	$*$	2.76	4.21	2.85
8	5.58	**3.14**	2.76	$*$	**2.91**	3.54
9	5.89	**2.98**	4.21	**2.91**	$*$	3.22
10	5.43	2.68	2.85	3.54	3.22	$*$

(33)

The features actually chosen for measurement in Section 4 were, in order, 9, 8 and 0. Looking at the table, these features all have high MI with the state but are not the three highest — feature 4 has the second-highest score of 5.71 bits. The table shows, however, that feature 4's weakness is a high MI of 4.21 bits with measurement 9, the individually best candidate, thanks to their nearby locations. Feature 4 becomes largely redundant when measured alongside feature 9 — a fact that would have dropped out of our previous step-by-step analysis, but which is also apparent before *any* measurements from $\mathtt{I}(\mathbf{x}_m)$. Features 9, 8 and 0 have relatively low inter-feature MI scores, highlighted in bold face,

to go with their high MI with the object state, and this reflects their well-separated locations. Similarly useful sets of three features, such as 0, 4 and 10, also suggest themselves.

Note that after each of these candidate features is measured, the appropriate reduction in the search area of the other selected features can be carried out by only updating the relevant sub-blocks of $\mathbf{x}_m$ and $\mathtt{P}_{\mathbf{x}_m}$, avoiding the full $O(N^2)$ cost between measurements, using a method similar to Postponement in SLAM [6]. Only one full update needs to be performed when all the measurements are completed.

7 Conclusions

We have shown that mutual information is the absolute measure of the value of measurements with the theoretical support to be the basis for rigorous theories of active search, and hope that this will inspire growing work in this area.

This paper presents analysis of the properties of information-theoretic guided search in the case where model and measurement uncertainties can be described with Gaussian distributions. In practical vision systems, this is of course not the whole story because of data association uncertainty (possible mismatches). We aim to continue this work by extending our analysis to more realistic cases with matching uncertainty, principally by applying the theory developed here to a mixture of Gaussians model for multiple hypotheses. We feel confident that information theory will be able to guide active search in the presence of outliers, and lead to active algorithms for search and matching which include image processing in the loop and are much more satisfactory than RANSAC with its reliance on random numbers and arbitrary thresholds. In inference, the only reason to resort to random sampling must be when the cost of actively deducing a more informative measurement strategy is prohibitively high, and we should aim to describe this crossover theoretically as pointed to in Section 6.

A particularly exciting area for active vision research is in high frame-rate tracking. With active search, processing requirements do not scale in proportion to frame-rate, as would be the case with bottom-up methods. As inter-image time gaps decrease, dynamical predictions become more accurate, reducing search regions and therefore processing time. Using information theory we will be able to analyse how adapting frame-rate affects performance — as in our study of measurement efficiency, there must be a trade-off relating frame-rate and target motion bandwidth.

As a final point, we note that our pure analysis of computational costs assumes monolithic processing resources, but the situation may be very different in practical systems with specialised embedded processors — image processing effectively comes for free with the correct hardware.

Acknowledgements

This work was performed when the author was at the University of Oxford. I am very grateful to Ian Reid, Ben Tordoff, David Murray and other members of Oxford's Active Vision Laboratory for discussions and software collaboration. This research is funded by EPSRC grant GR/T24685 and an EPSRC Advanced Research Fellowship.

References

[1] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 1. *International Journal of Computer Vision*, 56(3):221–255, 2004.

[2] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proceedings of the 9th International Conference on Computer Vision, Nice*, 2003.

[3] 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*, 2005.

[4] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proceedings of the 4th European Conference on Computer Vision, Cambridge*, pages 343–356, 1996.

[5] E. T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[6] J. G. H. Knight, A. J. Davison, and I. D. Reid. Constant time SLAM using postponement. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems*, 2001.

[7] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the 7th International Conference on Computer Vision, Kerkyra*, pages 1150–1157, 1999.

[8] D. Mackay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

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

[10] D. W. Murray, K. J. Bradshaw, P. F. McLauchlan, I. D. Reid, and P. M. Sharkey. Driving saccade to pursuit using image motion. *International Journal of Computer Vision*, 16(3):205–228, 1995.