Using Quantitative Statistics for the Construction of Machine Vision Systems.

Neil.A.Thacker
Imaging Science and Biomedical Engineering Division, University of Manchester, U.K.
Further author information: n.thacker@man.ac.uk, Telephone: +161 2755147

August 29, 2003

Abstract

This paper describes a design methodology for constructing machine vision systems. Central to this is the use of empirical design techniques and in particular quantitative statistics. The approach views both the construction and evaluation of systems as one and is based upon what could be regarded as a set of self-evident propositions;

- Vision algorithms must deliver information allowing practical decisions regarding interpretation of an image.
- Probability is the only self-consistent computational framework for data analysis, and so must form the basis of all algorithmic analysis processes.
- The most effective and robust algorithms will be those that match most closely the statistical properties of the data.
- A statistically based algorithm which takes correct account of all available data will yield an optimal result. [1].

Machine vision research has not emphasised the need for (or necessary methods of) algorithm characterisation, which is unfortunate, as the subject cannot advance without a sound empirical base. In general this problem can be attributed to one of two factors; a poor understanding of the role of assumptions and statistics, and a lack of appreciation of what is to be done with the generated data.

The methodology described here focuses on identifying the statistical characteristics of the data and matching these to the assuptions of the underlying techniques. The methodology has been developed from more than a decade of vision design and testing, which has culminated in the construction of the TINA open source image analysis/ machine vision system [htt://www.tina-vision.net].

1 Background

Attempting to solve vision problems of any real complexity necessitates, as in other engineering disciplines, a modular approach (a viewpoint popularised as a model for the human vision system by David Marr [18]). Therefore most algorithms published in the machine vision literature attend to only one small part of the "vision" problem, with the implicit intention that the algorithm could form part of a larger system What follows from this is that bringing these

[1]Where the definition of optimal can be unambiguously defined by the statistical specification of the problem.

together as components in a system requires that the statistical characteristics of the data generated by one module match the assumptions underpinning the next.

In many practical situations problems cannot be easily formulated to correspond exactly to a particular computation. Compromises have to be made, generally in assumptions regarding the statistical form of the data to be processed, and it is the adequacy of these compromises which will ultimately determine the success or failure of a particular algorithm. Thus, understanding the assumptions and compromises of a particular algorithm is an essential part of the development process. The best algorithms not only model the underlying statistics of the measurement process but also propagate these effects through to the output. Only if this process is performed correctly will algorithms form robust components in vision systems.

2 Technology and Scenario Evaluation

The evaluation of vision systems cannot be separated from the design process. Indeed it is important that the system is *designed for test* by adopting a methodology within which performance criteria can adequately be defined. When a modular strategy is adopted, system testing can be usefully considered as a two stage process [21] ;

- the evaluation of the statistical distributions of the data and comparison with algorithmic assumptions in individual modules; **technology evaluation**,
- the evaluation of the suitability of the entire system for the solution of a particular type of task; **scenario evaluation**.

The process of scenario evaluation is often time consuming and not reusable. The process of technology evaluation is complex and involves multiple objectives, however the results are reusable for a range of applications. It therefore merits effort and should be attempted. Ideally, we would like to be able to specify a limited set of summary variables which define the requirements of the input data and the main characteristics of the output data, in a manner similar to an electronic component databook. However, it must be remembered that it is the suitability of the output data for use in later modules which defines performance, and in some circumstances it may not be easy (or even possible) to define performance independent of practical use of the data. For instance, problems can arise when the output data of one algorithm is to be fed into several subsequent algorithms, each having different or even conflicting requirements. The most extreme example of this is perhaps scene segmentation where, in the absence of a definite goal, a concise method for the evaluation of such algorithms is likely to continue to be a challenge [26].

Machine vision research has not emphasised the need for (or necessary methods of) algorithm characterisation. This is rather unfortunate, as the subject cannot advance without a sound empirical base [12]. In our opinion this problem can generally be attributed to one of two main factors; a poor understanding of the role of assumptions and statistics; and a lack of appreciation of what is to be done with the generated data. The assumptions behind many algorithms are rarely clearly stated and it is often left to the reader to infer them[2]. The failure to present clearly the assumptions of an algorithm often leaves the reader confused as to the novel or valid aspects of the published research and can give the impression that it is

[2]A process we have previously called "inverse statistical identification" an allusion to the analogous problem of system identification in control theory.

possible to create good algorithms by accident rather than design. In addition, the inability to match algorithms to tasks may lead those who require practical solutions to real problems to conclude that little (if anything) published in this area really works. When in fact, virtually all published algorithms can be expected to work, provided that the input data satisfy the assumptions implicit in the technique. It is the unrealistic nature of these assumptions (e.g. noise free data) which is more likely to render algorithms useless.

The following is a description of a methodology for the design of vision module components. This methodology focuses on identifying the statistical characteristics of the data and matching these to the assumptions of particular techniques. The methods given in the appendices have been drawn from over a decade of vision system design and testing, which has culminated in the construction of the TINA machine vision system [29]. These include a combination of standard techniques, (such as covariance estimation and error propegation) and less standard ones (such as modal arithmetic, equal variance transforms and optimal fusion of hypothesis tests) which we have developed to address specific problems in algorithm design. Other techniques, more specific to evaluation of matching and classification, such as reciever operator curves and algorithmic modelling [34], are beyond the scope of this paper.

3 A Methodology Based on Quantitative Statistics

There are several common models for statistical data analysis, all of which can be related at some stage to either the principle of maximum likelihood or hypothesis tests. The likelihood framework provides methods for the estimation and propagation of errors, which are essential for characterising data at all stages in a system. Likelihood based approaches begin by assuming that the data under analysis conforms to a particular distribution. This distribution is used to define the probability of the data given an assumed model (appendix A). Hypothesis tests are based upon the probability that data drawn from the model would be less like that which has been observed (appendix F). The following sections discuss the checks that must be made on data in order to use these approaches properly.

3.1 Input data

The first step in evaluating an algorithmic module is identification of the appropriate model and empirical confirmation of the distribution with sample data. Appropriate methods for this task include; correlation analysis, histogram fitting and the Kolmogorov-Smirnov test [28]. The interpretation of the results from such processes require knowledge of the consequences of deviation from the expected distribution. In general, the greatest problems are caused by outliers (see below) although, the closer the data distributions conform to the assumed model, the better the expected results. Assumptions which prove valid for one algorithm, can often prove useful in the design of new algorithms. Some distributions commonly used in the machine vision literature are listed in table 1.

Although there are no general restrictions on the shape of these distributions the most common are Gaussian, Binomial, Multinomial and Poisson. These correspond to commonly occurring data generation processes. The central limit process ensures that the assumption of Gaussian distributed data forms the basis of many algorithms. This leads to tractable algorithms as the log-likelihood formulation of a Gaussian assumed model takes the particularly simple form of a least-squares statistic, which can often be formulated as a closed form

Example Task	Data	Error Assumption
Basic Data	Images	Uniform random Gaussian
Statistical Analysis	Histograms	Poisson sampling statistics
Shape Analysis	Edge location	Gaussian perpendicular to edge
	Line fits	Uniform Gaussian on end-points
Motion	Corner features	Circular (Elliptical) Gaussian
3D Objection Location	Stereo data	Uniform in disparity space

Table 1: Standard error model assumptions.

solution (appendix A). It is therefore useful to know that certain non-linear functions will transform the other common distributions to a form which approximates a Gaussian with sufficient accuracy to enable least-squares solutions to be employed (appendix D).

Unfortunately, most practical situations generate data with long tailed distributions (outliers). The problems associated with outliers in data analysis are well known. However, what appears less well understood is the reason for the complete lack of closed form solutions based upon a long tailed distribution. By definition only a simple quadratic form (or monotonic mapping thereof) for the log-likelihood, can be guaranteed to have a unique minimum. Long tailed (non-Gaussian) likelihood distributions inevitably result in multiple local minima which can only be located by explicit search (e.g. the Hough transform) or optimisation (e.g. gradient descent).

Other assumptions in the likelihood formulation generally include those of data independence. Independence can be confirmed by plotting joint distributions. Uncorrelated data will produce joint distributions which are entirely predicted by the outer product of the marginal distributions. Correlations (the lack of independence) in data can have several consequences. Strong correlations may produce suboptimal estimates from the algorithm and covariances may not concisely describe the error distribution. Data correlation can be elimiated using techniques such as Principle Component Analysis (PCA), Independent Component Analysis (ICA) or counter-propagation neural networks.

3.2 Output data

The next step in module analysis is to estimate the errors on the output data. If the output is the result of a log-likelihood measure then errors can be computed using covariance estimation (appendix B). Covariance estimation is possible even in the presence of outliers, provided that a robust kernel is used [19]. If the output quantities from a module are computed from noisy data the errors on the results can be calculated using error propagation (appendix C). Both of these theoretical techniques assume Gaussian distributed errors and locally linear behaviour of the algorithmic function.

These assumptions require validation (i.e. checks to ensure that the theory is an accurate representation of reality), which can be achieved using Monte-Carlo approaches. Once again, techniques such as histogramming, fitting and Kolmogorov-Smirnov tests are useful. High degrees of non-linear behaviour can be addressed using a technique we call modal arithmetic [35] (appendix E). Non-linear transformation of estimated variables may be necessary in order to make better approximations to Gaussian distributions. It may also be necessary to combine variables in order to eliminate data correlation. Selecting data representations which provide appropriate descriptions of statistical distributions is of fundamental importance [3]. The definition of the parameters passed between algorithms can be substantially different to naive expectation e.g. 3D data from a stereo algorithm is best represented in disparity space (appendix D). Notice, the evaluation process has a direct influence on the process of system design, underscoring the earlier statements that system design and performance evaluation cannot (and should not) be treated separately.

In many cases the division of tasks into modules will be driven by the statistical characteristics of the processed data and cannot be specified *a priori* without a very clear understanding of the expected characteristics of all system modules. Given the source of data typical of machine vision applications it is also very likely that algorithms will produce outlier data which cannot be eliminated by transformation or algorithmic improvement and will therefore require appropriate (robust) statistical treatment in later modules.

A rigid application of the above design and test process (see figure 1) will produce verifiable optimal outputs from each module. Ultimately however, we will need to know if this data is of sufficient quality to achieve a particular task, a process we will call **scenario evaluation**. Under many circumstances it should be sufficient to determine the required accuracy of the output data in order to achieve this task. Alternatively, the covariance estimates from the technology evaluation could be used to quantify the expected performance of the system on a per-case basis.

Statistical measures of performance can be obtained by testing on a representative set of data. We would anticipate the need to compute the probability of a particular hypothesis, either as a direct interpretation of scene contents or as the likely outcome of an action (appendix F). Such probabilities are directly testable and can be described as *honest probabilities* [10] only if they agreee with the true frequency of occurence of events, (e.g. classification probabilities $P(C|data)$ should be wrong $1 - P(C|data)$ of the time). Tested hypotheses, such as a particular set of data being generated by a particular model, should have a uniform probability distribution. The importance of this feature in relation to the work presented here is that knowledge of the expected distribution for the output provides a mechanism for self-test. Some approaches to pattern recognition, such as k-nearest neighbours, are almost guaranteed to be honest by construction. In addition the concept of honesty provides a very powerful way of assessing the validity of probabilistic approaches. In [24] it was shown that iterative probabilitstic update schemes which drive probability estimates to converge to 0 or 1 cannot be honest and are therefore also not optimal.

An algorithm which makes use of all available data in the correct manner must deliver an optimal result. This is not as uncommon occurrence in computer vision as may be assumed and many problems (camera calibration [30], and shape recognition [31]) do have optimal solutions. If this can be established for an algorithm then extensive evaluation (e.g. on a large number of images) can be expected to prove only one thing, that the algorithm can only be bettered by one which takes account of more data or assumes a more restricted model. Use of a more restricted model will of course limit use of the algorithm, and any assumption which prevents the generic use of an algortihm needs to be considered very carefully. It is all to easy to design algorithms which work (at least qualitatively) on a very limited subset of images and this is a criticism which is often made of work in this area. Using more information rather than assumptions to solve the problem might therefore be the preffered option. In a modular system, where input data has been separated in order to make data processing

[3] yet is often overridden by preconceived ideas of algorithm design.

more manageable, use of more data corresponds to fusion of output data. For this reason quantitative methods of optimal data combination are of fundamental importance. Within the probabilistic framework described above there are three ways of achieving this; combination of probability (using a learning technique such as a neural network), combination of likelihoods (using covariances), and combination of hypothesis tests. All three of these are described in greater detail in appendix G.

4 The TINA 3D Model Matching System

We can illustrate the quantitative statistical methodology presented here with the example of the wireframe object location system in TINA. The original version of the 3D model matcher was presented in [22] and [23]. Briefly the system used a sparse edge based depth map extracted from pairs of binocular stereo images together with the corresponding camera calibration information. A geometric interpretation of the scene was constructed by fitting lines and arcs to the depth map data. Statistical matching of 3D scene descriptions to a stored wireframe model enabled the location of the model within the scene to be identified. Each of these stages involve maintaining a model of data accuracy so that the assumptions made at each stage are consistent with the input data provided. Unfortunately, accurate determination of object location in the later stages is not possible with this simple scheme as many of the assumptions necessary to construct a working system are often violated. In particular, illumination artifacts and shadowing often move features from their expected position or introduce new ones. These effects, combined with a least-squares estimation of object location, guarantees problems with resulting output so that results at this stage are not fully quantitative. The closed loop validation stage (CLV) [15] closes the loop on the process, testing the generated hypothesis against the original image data and refining this estimate without the constraints imposed by the previous algorithmic stages (Table 2). The CLV used an iterative robust approach which deals appropriately with outliers, thereby regaining a sub-pixel accurate estimate of object pose. Formulation within the appropriate likelihood framework also raises the possibility of the computation of covariances on transformation parameters, which we would expect to be necessary in any working system. Figure 2 outlines how the CLV stages (dashed lines) compliments the model matcher (solid lines).

5 Summary and Conclusions

This document suggests a quantitative statistical approach to the design and testing of machine vision systems which could be considered as an extension of methodologies suggested by other authors [3, 13]. We have focused on the use of likelihood and hypothesis testing paradigms and it would be natural for a reader familiar with the machine vision literature to feel that we have missed out other approaches which have (or have had) a higher profile in the literature (e.g. computational geometry and image analysis as inverse optics). However, we would argue that for the modular approach to system building to succeed we must have appropriate control over the statistical distributions generated during analysis. Inevitably, to acquire quantitative data for use in a system, error analysis will be required. This is possible with likelihood based techniques because they enable the construction of measures to

Key	Purpose	Algorithms	Assumptions
A	Edge & Corner Detection 2D Geometry Fitting	Canny [7]	Edges present in expected locations Curves and lines can be correctly linked and fitted
B	Stereo Matching 3D Geometry Fitting	PMF GDB	Accurate epi-polar geometry and match metrics Accurate camera calibration
C	Sequential Model Building	GEOMstat	Accurate feature locations
D	Wireframe Model Matcher	SMM	Gaussian errors on all extracted features Closed form solution is appropriate
E	Camera Calibration	Tsai [36]	Known calibration object present

Table 2: Algorithmic descriptions and assumptions for the 3DMM with key for shaded components in figure 2.

determine the best interpretation of the data (such as least squares) and also allow quantitative predictions to be made of the stability of estimated parameters (such as covariances). The machine vision problem, therefore, does not stop once a closed form solution is found (see [14] for a discussion of the use of statistics in closed form solutions). This difficult step is often missing in the work found in the literature, yet attempting to do it can completely alter our understanding of the apparent value or even validity of the approach. The work of Maybank [17] demonstrated exactly this point with regard to the use of affine invariants for object recognition.

The reader may at this point feel that there is a broader context for probability theory than likelihoods and hypothesis testing. In particular likelihood based techniques have well known limitations, such as bias in finite samples [9]. The problem of model selection is endemic in the machine vision area and likelihoods cannot be directly compared between two different model hypotheses. Approaches which aim to directly address these issues are thus acceptable extensions to the above methodology [32]. However, some popular areas of probability theory do not (at least yet) have comparable quantitative capabilities (e.g. Bayesian approaches) and may therefore be unsuitable for system building. We have made an attempt to summarise these issues in [5]. It remains to be seen whether advocates of these approaches and others (such as Dempster-Schafer theory) are able to address these issues.

Other approaches to algorithm design use methods which are based upon apparently different principles, such as entropy and mutual information [37]. However, we regard these as only alternative ways to formulate problems and believe that most experienced researchers would accept that all approaches should be reconcilable with probability theory. Thus if there already exists a likelihood based formulation of the technique, this should be taken as the preferred approach. Obviously, if the research community as a whole accepted this viewpoint many papers would already have been written and presented differently. As the construction of systems from likelihood based formulations is generally likely to require optimisation of robust statistics, generic algorithms for the location of multiple local optima should be regarded as a fundamental research issue. So too should the problem of covariance estimation from common optimisation tasks and popular algorithmic constructs, (such as Hough transforms), which have already been shown to be consistent with likelihood approaches [27, 1].

Many attempts at algorithmic evaluation in the literature focus on the specification of particular performance metrics. Although these metrics may give some indication as to the basic workings of an algorithm, quantitative evaluation should set as the ultimate goal an understanding of the performance of the system. Performance metrics for modules should therefore be specified with this in mind.

Non-quantitative evaluation is probably of more use in the early stages of algorithm construction than during the final integration into a system. However, in the methodology described a key aspect is the identification of assumptions. Knowledge of these assumptions (and suitable methods for determining their validity) allows comparisons of algorithms to be carried out at the theoretical level. Also, we should not be surprised when algorithms which are built upon the same set of founding assumptions within a sensible probabilistic framework, give near identical performance. This has been well illustrated in several pieces of work including that by Fisher et. al [11], where alternative techniques for location of 3D models in 3D range data were found to give equivalent results to within floating point accuracy. If careful statistical analysis of data did not give this result then it would be an indication that probability theory itself was not self-consistent. Also, when performing comparative testing of modules we should be aware that algorithmic scope, as determined by the restrictions imposed by the assumptions, should be taken into account in the final interpretation of results. Algorithms which give apparently weaker performance on the basis of performance metrics may still be more applicable for some tasks. A simple example of this is that least squares fitting will generally give a better bounded estimate of a set of parameters than robust techniques, yet robust techniques are essential in the presence of outliers.

A Common Likelihood Formulations

Maximum Likelihood statistics involves the identification of the event Y which maximises a probability of the form

$$P(X_0X_1X_2...X_n|Y)P(Y) = P(X_0|X_1X_2...X_nY)P(X_1|X_2...X_nY)......P(X_n|Y)P(Y)$$

where X_i are the observed data. For large numbers of variables this is an impractical method for probability estimation. Even if the events were simple binary variables there are clearly an exponential number of possible values for even the first term in $P(XY)$ requiring a prohibitive amount of data storage. In the case where each observed event is independent of all others we can write.

$$P(X|Y) = P(X_0|Y)P(X_1|Y)P(X_2|Y)...P(X_n|Y)$$

The rather redundant use of the conditional terms $|Y$ is often dropped for convenience. A more detailed treatment of the theory and techniques of Maximum Likelihood statistics can be found in [9].

Dealing with Binary Evidence

The simplest likelihood model is for binary observations of a set of variables with known probabilities. If we make the assumption that the event X_i is binary with probability $P(X_i)$ then we can construct the probability of observing a particular binary vector X as:

$$P(X) = \Pi_i (P(X_i)^{X_i}(1 - P(X_i))^{(1-X_i)}$$

The log likelihood function is therefore

$$log(P) = \sum_i X_i log(P(X_i)) + (1 - X_i) log(1 - P(X_i))$$

This quantity can be minimised or directly evaluated in order to form a statistical decision regarding the likely generator of X. This is therefore a useful equation for methods of statistical pattern recognition.

Poisson and Gaussian Data Distributions

A very common problem in machine vision is that of determining a set of parameters in a model. Take for example a set of data described by the function $f(a, Y_i)$ where a defines the set of free parameters defining f and Y_i is the generating data set. If we now define the variation of the observed measurements X_i about the generating function with some random error we can see that the probability $P(X_0|X_1X_2...X_NaY_0)$ will be equivalent to $P(X_0|aY_0)$ as the model and generation point completely define all but the random error.

Choosing Gaussian random errors with a standard deviation of σ_i gives;

$$P(X_i) = A_i exp(\frac{-(X_i - f(a, Y_i))^2}{2\sigma_i^2})$$

where A_i is a normalisation constant. We can now construct the maximum likelihood function;

$$P(X) = \Pi_i A_i exp(\frac{-(X_i - f(a, Y_i))^2}{2\sigma_i^2})$$

which leads to the χ^2 definition of log likelihood;

$$log(P) = \frac{-1}{2} \sum_i \frac{(X_i - f(a, Y_i))^2}{\sigma_i^2} + const$$

This expression can be maximised as a function of the parameters a and this process is generally called a least squares fit. Whenever least squares is encountered there is implicit assumption of independence and of a Gaussian distribution. In practical situations the validity of these assumptions should be checked by plotting the distribution of $X_i - f(a, Y_i)$ to make sure that it is Gaussian.

Often when working with measured data we need to interpret frequency distributions of continuous variables, for example in the form of frequency histograms. In order to do this we must know the statistical behaviour of these measured quantities. The generation process for a histogram bin quantity (making an entry at random according to a fixed probability) is strictly a multi- distribution, however for large numbers of data bins this rapidly becomes well described by the Poisson distribution. The probability of observing a particular number of h_i for an expected probability of p_i is given by;

$$P(h_i) = exp(-p_i)\frac{p_i^k}{h_i!}$$

For large expected numbers of entries this distribution approximates a Gaussian with $\sigma = \sqrt{h_i}$. These facts allow us to see that the standard χ^2 statistic is appropriate for comparing two frequency distributions h_i and j_i for equal sized samples;

$$\chi^2 = \sum_i (h_i - j_i)^2/(h_i + j_i)$$

However, this is not necessarily the best way to analyse such data [33].

B Covariance Estimation

The concept of error covariance is very important in statistics as it allows us to model linear correlations between parameters. For locally linear fit functions f we can approximate the variation in a χ^2 metric about the minimum value as a quadratic. Starting from the χ^2 definition as a least squares formulation using the same notation as previously;

$$\chi^2 = \frac{1}{2}\sum_i^N \frac{(X_i - f(Y_i, a))^2}{\sigma_i^2}$$

We can compute the first and second order derivatives as follows;

$$\frac{\partial \chi^2}{\partial a_n} = \sum_i^N \frac{(X_i - f(Y_i, a))}{\sigma_i^2}\frac{\partial f}{\partial a_n}$$

$$\frac{\partial^2 \chi^2}{\partial a_n \partial a_m} = \sum_i^N \frac{1}{\sigma_i^2}\left(\frac{\partial f}{\partial a_n}\frac{\partial f}{\partial a_m} - (X_i - f(y_i, a))\frac{\partial^2 f}{\partial a_n \partial a_m}\right)$$

The second term in this equation is expected to be negligible compared to the first and with an expected value of zero if the model is a good fit. Thus the cross derivatives can be approximated to a good accuracy by;

$$= \sum_i^N \frac{1}{\sigma_i^2}\left(\frac{\partial f}{\partial a_n}\frac{\partial f}{\partial a_m}\right)$$

The following quantities are often defined;

$$\beta_n = \frac{1}{2}\frac{\partial \chi^2}{\partial a_n} \qquad \alpha_{nm} = \frac{1}{2}\frac{\partial^2 \chi^2}{\partial a_n \partial a_m}$$

As these derivatives must correspond to the first coefficients in a polynomial (Taylor) expansion of the χ^2;

$$C = \alpha^{-1} \quad \text{where } \alpha = \begin{vmatrix} \alpha_{11} & \alpha_{12} & \dots \\ \alpha_{21} & \alpha_{22} & \dots \\ \dots & \dots & \alpha_{nm} \end{vmatrix}$$

And the expected change in χ^2 for a small change in model parameters can be written as $\Delta\chi^2 = \Delta a^T \alpha \Delta a$.

Process	Calculation	Theoretical Error
Addition	$O = I_1 + I_2$	$\Delta O^2 = \sigma_1^2 + \sigma_2^2$
Division	$O = \frac{I_1}{I_2}$	$\Delta O^2 = \frac{\sigma_1^2}{I_2^2} + \frac{I_1^2 \sigma_2^2}{I_2^4}$
Multiplication	$O = I_1 . I_2$	$\Delta O^2 = I_2^2 \sigma_1^2 + I_1^2 \sigma_2^2$
Square-root	$O = \sqrt(I_1)$	$\Delta O^2 = \frac{\sigma_1^2}{I_1}$
Logarithm	$O = log(I_1)$	$\Delta O^2 = \frac{\sigma_1^2}{I_1^2}$
Polynomial Term	$O = I_1^n$	$\Delta O^2 = (nI_1^{n-1})^2 \sigma_1^2$

Table 3: Error Propagation in Image Processing Operations

C Error Propagation

In order to use a piece of information $f(X)$ derived from a set of measures X we must have information regarding its likely variation. If X has been obtained using a measurement system then we must be able to quantify the precision of this system. Therefore, we require a method for propagating likely errors on measurements through to $f(X)$. Assuming knowledge of error covariance this can be done as follows;

$$\Delta f(X) = \nabla f^T C_X \nabla f$$

The method simply uses the derivative of the function f as a linear approximation to that function. This is sufficient provided that the expected variation in parameters ΔX is small compared to the range of linearity of the function. Application of this technique to even simple image processing functions gives useful information regarding the expected stability of each method (Table 3). When constructing algorithms from such image processing modules any data dependency will produce problems with noise stability unless the errors are propegated fully for later use [8]. When the problem does not permit algebraic manipulation in this form (due to significant non-linear behaviour in the range of $\Delta f(X)$ or functional discontinuities) then numerical (Monte-Carlo) approaches may be helpful in obtaining the required estimates of precision.

D Transforms to Equal Variance

The choice of a least squares error metric gives many advantages in terms of computational simplicity and is also used extensively for definitions of error covariance and optimal combination of data (Appendices B and G). However, the distribution of random variation on the observed data X is something that generally we have no initial control over and could well be arbitrary and so we have the problem of adjusting the measurements in order to account for this. In addition, we have the problem that different choices for the way we represent the data will produce different likelihood measures. Take for example a set of measurements made from a circle, we can choose to measure the size of a circle as a radius or as an area. However, it can be easily shown that constructing a likelihood technique based upon sampled distributions will produce different (inconsistent) formulations for these two representations of the same underlying data. Transferrring the likelihood from a distribution of radial errors will not produce the impirically observed distribution for area due the non-linear transformation between these variables. Which should we choose as correct (or are both wrong)? Initially these may be seen as separate problems, but in fact they are related and may have one common solution. To understand this we need to consider non-linear data transformations and the reasons for applying them.

In many circumstances it is possible to make distributions more suitable for use of standard ML formulations (eg: least squares) by transformation $g(X_i)$ and $g(f(a, Y_i))$, where g is chosen so that the initial distribution of X_i maps to an equal variance distribution (near Gaussian) in g. Examples of this for statistical distributions are the use of the square-root transform for Poisson distributed variables [33] and the $asin$ mapping for binomial distributed data. However, this problem can occur more generally due to the need to have to work with quantities which are not measured directly.

One good example of this is in the location of a known object in 3D data derived from a stereo vision system. In the coordinate system where the viewing direction corresponds to the z axis, x and y measures have errors determined by image plane measurement. However, the depth z_i for a given point is given by;

$$z_i = fI/(X_{li} - X_{ri})$$

where I is the interocular separation, f is the focal length and X_{li} and X_{ri} are image plane measurements. Attempts to perform a least squares fit directly in (x, y, z) space results in instability due to the non-Gaussian nature of the z_i distribution. However, transformation to $(x, y, 1/\sqrt{2}z)$ yields Gaussian distributions and good results. In general, observation of a dependency of the error distribution of a derived variable with that variable (in the above case the dependency of σ_z on z), is very often a sign that the likelihood distribution is skewed. For a known functional dependency h the transformation g which maps the variable X_i to one with equal variance follows directly from the method of error propagation and is given by;

$$g \;=\; \int \frac{1}{h(X)} dX$$

All of the transformations mentioned above can be generated from this process, including those which map standard statistical distributions to more Gaussian ones, though the extent to which this is a general property of this method is unclear. We are now also in a position to answer our questions regarding data representation in ML. The selection of measured variables from the equal variance domain provides a unique solution to the problem of identification of the source data space.

E Modal Arithmetic

Sometimes the effects of non-linear calculations on data with a noise distribution affects not only the variance of the computed quantity but also the mean value. From a likelihood point of view we can define the ideal result from a computation as the most frequent (or modal) value that would have resulted from data drawn from the expected noise distribution. We can find such values directly, via the process of Monte-Carlo, but we can also predict these values analytically. We have termed the algorithm design technique which addressed this issue *modal arithmetic* [35]. The general method of modal arithmetic for a measured value with distribution $D(x)$ and a non-linear function $f(x)$ would be to find the solution x_{max} of

$$\partial[\frac{D(x)}{\partial f(x)/\partial x}]/\partial x \; = \; 0$$

with the modal solution of $f(x_{max})$. Modal arithmetic is unconditionally stable, as peaks in probability distributions cannot occur at infinity. It also has much similarity with some approaches in statistics which advocate the use of the **mode** rather than the **mean** as the most robust indicator of a distributed variable. In [35] we applied this technique to deconvolution, as a complex division in the presence of noise, and were able to show that the resulting solution regenerated the Wiener filter [38], without the need to assume a linear form for the optimal filter.

F Hypothesis Testing

Having made quantitative measurements from our system we will ultimately need to make decisions based upon those measurements in comparison to some predefined model. For example, do not attempt to move the mobile vehicle through a doorway unless the vision system estimates that it will pass. Many statistical tests are based on the idea of generating the probability that data drawn from the expected test distribution would be more frequent than the example under test. This approach leads to the common statistical techniques of z-scores, T tests, and Chi-squared tests to name a few. This follows directly from the original definition of a confidence interval, due to Neyman [20] and yet is rarely used in machine vision. This is unfortunate, as the methods do not suffer the same restrictions regarding distributions which apply to covariances.

Hypothesis tests (i.e. does the data conform to the assumed model?) are perfomed on the basis of one model at a time, in contrast to Bayesian approaches which require all possible generators (models) of the data. In addition, such statistical tests are fully quantitative. Probabilities computed from such statistics have the characteristic that the distribution of values drawn from the assumed model will be flat. This is useful as a mechanism for self test. The most common form of this statistic is that for a Gaussian and is known as the **error function** which is provided as a mathematical function in most languages (e.g. the erf() library function). However, such statistics can be generated for any model for which the expected data distribution is known, using the ordering principle. This states that the ordering of integration along the measurement axis should be defined so that the probability density is monotonically decreasing. For the Gaussian case shown above this gives the rather trivial result that we integrate along the standard measurement axis x away from the peak, as the function is monotonically decreasing from $x = 0$. Although this is not the only way to order the data (there are potentialy infinite numbers of equivalent possible ordering schemes depending upon how we define our variables e.g. x^2) this is the one which gives confidence limits which are maximally compact in the chosen parameter domain. Generally, the preferred parameter domain would be selected as the space in which x was uniformly accurate, so that this compactness has meaning from the point of view of measureable localisation. This is sometimes referred to as a “natural” parameterisation and is related to the concept of the equal variance transform (appendix D).

In image processing the required distributions can often be bootstrapped directly from the image (e.g. as in [6]). Under these circumstances the possibility of multi-modal density functions makes the application of the ordering principle slightly less straightforward.

Finally, as the only requirement for the use of such probabilities is that they have a uniform distribution, empirical approaches can be used to re-flatten distributions which result from imprecise analysis. Such hypothesis tests are also easily combined using standard statistical approaches (See appendix G).

G Data Fusion

Optimal Combination using Covariances

Given two estimates of a set of parameters a_1 and a_2 and their covariances (α_1 and α_2) we can combine the two sets of data as follows;

$$a_T = \alpha_T^{-1}(\alpha_1 a_1 + \alpha_2 a_2) \quad with \quad \alpha_T^{-1} = \alpha_1^{-1} + \alpha_2^{-1}$$

This method combines the data in the least squares sense, that is the approximation to the χ^2 stored in the covariance matrices has been combined directly to give the minimum of the quadratic form. The method can be rewritten slightly giving

$$a_T = a_1 + \alpha_T^{-1}\alpha_2 \Delta a$$

where $\Delta a = a_2 - a_1$. This form is directly comparable to the information filter form of the Kalman filter.

Optimal Combination of Hypothesis Tests

Hypothesis test probabilities should have uniform distributions (if they are honest). Given n quantities each having a uniform probability distribution $p_{i=1,n}$, the product $p = \prod_{i=1}^{n} p_i$ can be renormalised to have a uniform probability distribution $F_n(p)$ using;

$$F_n(p) = p \sum_{i=0}^{n-1} \frac{(-\ln p)^i}{i!} \tag{1}$$

Proof of this relationship can be generated in the following manner. The quantities p_i can be plotted on the axes of an n dimensional sample space, bounded by the unit hypercube. Since they are uniform, and assuming no spatial correlation, the sample space will be uniformly populated. Therefore, the transformation to $F_n(p)$ such that this quantity has a uniform probability distribution can be achieved using the probability integral transform, replacing any point in the sample space p with the integral of the volume under the contour of constant p. Generalisation of this process to non-integer numbers (which is useful for cases where we have an effective number of degrees of freedom) and other useful results are presented in [4].

Optimal Combination from Example Data

When the area of neural networks re-emerged as a popular topic in the mid 80's much was claimed about the expected capabilities regarding flexibility, suitability for system identification and robustness. Most of these claims were subsequently shown to be optimistic. However, one problem that neural networks are relatively good at is non-linear data fusion. A neural network when trained on an appropriate form of data with the correct algorithm will approximate Bayes probabilities as outputs.

The mathematics describing this process is given in [16] but a more intuitive argument is as follows. Each input vector pattern X defines a unique point in input space. Associated with each data point is the ideal required output, for example a binary output classification. As the number of samples grows large the number of examples of data in the region of each point also grows large. If training with a least squares error function the target output for each point in pattern space will be the mean of local values. For a binary coding problem the mean value is the Bayes probability of the model given the data.

Given $P(A|B)$ and $P(A|C)$ can we compute $P(A|BC)$? We can clearly solve this problem provided these probabilities are independent by simple multiplication. If however the measures are correlated there is no standard statistical method for this process. This is unfortunate as we would expect a modular (AI) decision system to need to solve this task. Standard neural network architectures trained in the standard way will however approximate $P(A|P(A|B)P(A|C))$ for the reasons described above [2]. Provided that there is enough information in the set of probabilities being fused to regenerate the original data the fusion process will be able to achieve optimality.

Acknowledgements

I would like to acknowledge the support of the EU PCCV (Performance Characterisation of Computer Vision Techniques) IST-1999-14159 grant and BAe Systems (who sponsored the development of the methodology). I am also grateful to the numerous people who have reviewed and commented on both this paper and the ideas behind it, particularly Anthony Lacey, Paul Bromiley, Patrick Courtney and Gareth Rees. I would like to acknowledge discussions with Adrian Clarke, Paul Whelan, John Barron, Emanuel Truecco and Mike Brady. Finally the author would like to acknowledge Ross Beveridge, Kevin Bowyer, Henrik Christensen, Wolfgang Foerstner, Robert Haralick and Jonathon Phillips for their contributions to the area and whose work has inspired this paper.

References

1. A.P.Ashbrook, N.A.Thacker and P.I.Rockett, *Pairwise Geometric Histograms. A Scaling Solution for the Recognition of 2D Rigid Shape.*, Proc. for SCIA95, Uppsala, Sweden, pp271, 1995.

2. D.Booth, N.A.Thacker, M.K.Pidock and J.E.W.Mayhew. *Combining the Opinions of Several Early Vision Modules Using a Multi-Layered Perceptron.* Int.Journal of Neural Networks, 2, 2/3/4, June-December, 75-79, 1991.

3. K. W. Bowyer and P. J. Phillips *Empirical Evaluation Techniques in Computer Vision* edited by K.W. Bowyer and P.J. Phillips, IEEE press, ISBN 0-8186-8401-1, 2000

4. P. A. Bromiley, T.F. Cootes and N.A. Thacker, *Derivation of the Renormalisation Formula for the Product of Uniform Probability Distributions and Extension to Non-Integer Dimensionality.* Tina memo 2001-008.

5. P.A. Bromiley, M.L.J. Scott, M. Pokrić, A.J. Lacey and N.A. Thacker, *Bayesian and Non-Bayesian Probabilistic Models for Magnetic Resonance Image Analysis*, Submitted to Image and Vision Computing, Special Edition; The use of Probabilistic Models in Computer Vision.

6. P.A.Bromiley, N.A.Thacker and P.Courtney, *Non-Parametric Subtraction Using Grey Level Scattergrams*, BMVC 2000, Bristol, pp 795-804, Sept. 2000.

7. J F Canny. A computational approach to edge detection. *IEEE PAMI*, 8(6), 1986.

8. P. Courtney and N.A. Thacker, *Performance Characterisation in Computer Vision: The Role of Statistics in Testing and Design*, "Imaging and Vision Systems: Theory, Assessment and Applications", Jacques Blanc-Talon and Dan Popescu (Eds.), NOVA Science Books, 2001, ISBN 1-59033-033-1.

9. G. Cowan *Statistical Data Analysis*, Oxford University Press, ISBN 0-19-850156-0, 1998.

10. A.P. Dawid, *Probability Forecasting*, Encyclopedia of Statistical Science 7, pp 210-218. Wiley, 1986.

11. A. Lorusso, D.W. Eggert, and R.B. Fisher, *Estimating 3D Rigid Body Transformations: A Comparison of Four Algorithms*, Machine Vision Applications, 9 (5/6), 1997, pp.272-290.

12. W. Foerstner, *10 Pros and Cons Against Performance Characterisation of Vision Algorithms*, Proceedings of ECCV Workshop on Performance Characteristics of Vision Algorithms, Cambridge, UK, April 1996. Also in Machine Vision Applications, 9 (5/6), 1997, pp.215-218.

13. R.M. Haralick, *Performance Characterization in Computer Vision*, CVGIP-IE, 60, 1994, pp.245-249.

14. R.M. Haralick, C.N. Lee, K. Ottenberg and M. Noelle, *Review and Analysis of Solutions to the Three Point Perspective Pose Estimation Problem*, Intl. J. Computer Vision, 13(3), 1994, pp.331-356.

15. A.Lacey, N.A.Thacker, P.Courtney and S.Pollard, TINA 2001: The Closed Loop 3D Model Matcher. proc. BMVC, 2001, pp203-212.

16. M.D.Richard and R.P.Lippmann, *Neural Network Classifiers Estimate Bayesian a Posterioi Probabilities*, Neural Computation,3,461-483,1991.

17. S.J. Maybank, *Probabilistic Analysis of the Application of the Cross Ratio to Model Based Vision*, Intl. J. Computer Vision, 16, 1995, pp.5-33.

18. D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information* Publisher W. H. Freeman Company, NY 1982.

19. P. Meer, D. Mintz, A. Rosenfield and Dong Yoom Kim *Robust Regression Methods for Coputer Vision: A Review* Intl. J. Computer Vision, 6:1, 1991, pp. 59-70.

20. J. Neyman, *X-Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability*, Phil. Trans. Royal Soc. London, **A236**, pp. 333-380, 1937.

21. P. J. Phillips, A. Martin, C. L. Wilson and M. Przybocki, *An Introduction to Evaluating Biometric Systems* IEEE Computer Special Issue on Biometrics, pp. 56-63, Feb. 2000.

22. J Porill, S B Pollard, T Pridmore, J Bowen, J E W Mayhew, and J P Frisby. Tina: A 3d vision system for pick and place. In *Proceedings of the Alvey Vision Conference*, 1987.

23. J Porill, S B Pollard, T Pridmore, J Bowen, J E W Mayhew, and J P Frisby. *TINA: A 3D Vision System for Pick and Place*. MIT Press, 1991.

24. I. Poole, *Optimal Probabilistic Relaxation Labeling*, Proc. BMVC 1990, BMVA, 1990.

25. W. H. Press, B. P., Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipies in C* Cambridge University Press., 1991

26. G. Rees, P. Greenway and D. Morray, *Metrics for Image Segmentation*, Proceedings of ICVS Workshop on Performance Characterisation and Benchmarking of Vision Systems, Gran Canaria, January 1999.

27. R.S. Stephens, *A Probabilistic Approach to the Hough Transform*, British Machine Vision Conference BMVC90, 1990.

28. A. Stuart, K. Ord and S. Arnold *Kendall's Advanced Theory of Statistics* Vol. 2A, Classical Inference and the Linear Model, Sixth Edition, Arnold Publishers, 1999.

29. http://www.tina-vision.net/ *TINA: Open Source Image Analysis Environment* ISBE, University of Manchester, UK

30. N A Thacker and J E W Mayhew, *Optimal Combination of Stereo Camera Calibration from Arbitrary Stereo Images* Image and Vision Computing, Vol. 9 No. 1, pp. 27-32, 1991.

31. N.A.Thacker, P.A.Riocreux, and R.B.Yates, 'Assessing the Completeness Properties of Pairwise Geometric Histograms", Image and Vision Computing, 13, 5, 423-429, 1995.

32. N.A.Thacker, D.Prendergast and P.I.Rockett, B-Fitting: A Statistical Estimation Technique with Automatic Parameter Selection.',Proc, BMVC, pp 283-292, Edinburgh, 1996.

33. N.A.Thacker, F.Ahearne and P.I.Rockett, 'The Bhattacharryya Metric as an Absolute Similarity Measure for Frequency Coded Data.' Kybernetika, 34, 4, 363-368, 1997.

34. P. Courtney, N.A.Thacker and A.Clark, *Algorithmic Modelling for Performance Evaluation*, Machine Vision and Applications, 9, 219-288, 1997.

35. N.A.Thacker and A.J.Reader, *Modal Division and its Application to Medical Image Analysis*, Proc. MIUA, pp 7-10, London. 10th-11th July, 2000.

36. R Y Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proc. IEEE CVPR 86*, pages 364–374, 1986.

37. P. Viola, *Alignment by Maximisation of Mutual Information* M.I.T. PhD Thesis, 1995.

38. N. Wiener, *Extrapolation, Interpolation and Smoothing of Stationary Time Series with an Appendix by N. Levinson* Technology Press of the MIT and J. Wiley, New York, 1949.

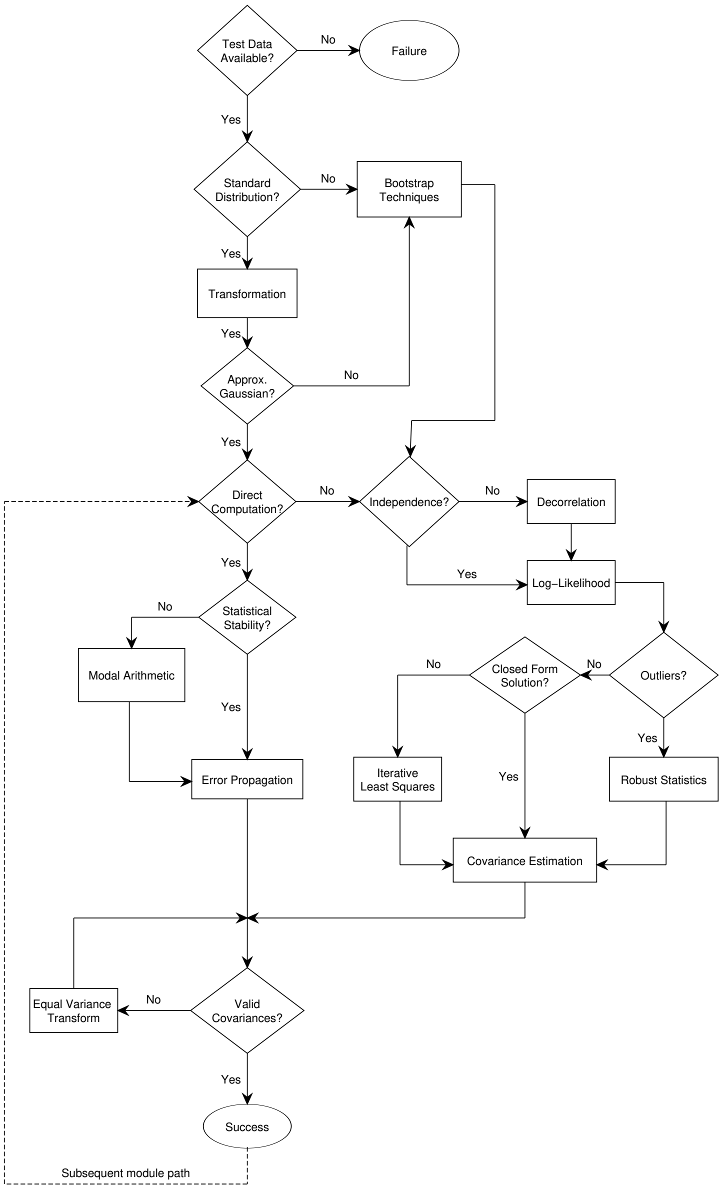


Figure 1: Technology evaluation flow chart. This diagram identifies the major design decisions which must be addressed in order to deliver quantified outputs from an algorithm. Transforms are suggested at various stages in order to solve problems associated with non-Gaussian behaviour. The label Bootstrap is intended to refer to custom made statistical measures constructed from sample data.

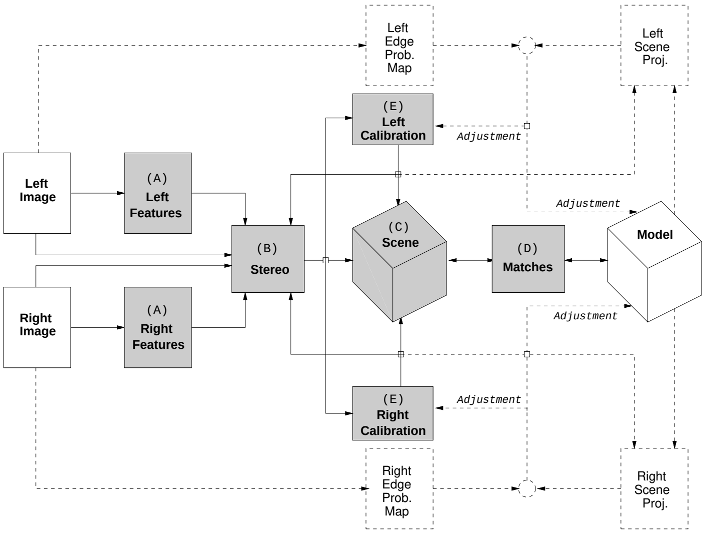
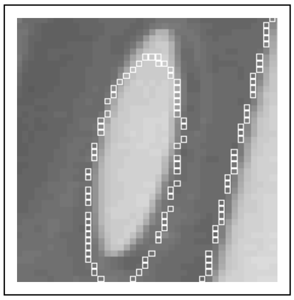


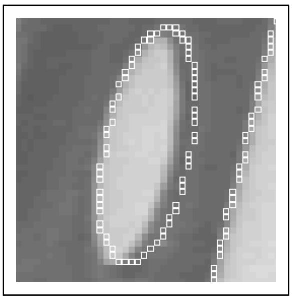
Figure 2: Block diagram of the updated 3D model matcher. The dashed sections represent the additional processing of the closed-loop validation. The letters reference table 2

(a) Object and located model overlayed - front view

(b) Object and located model overlayed - rear view



(c) Detail showing accuracy before CLV



(d) Detail showing accuracy following CLV

Figure 3: Typical performance of the original 3DMM on an industrial component.