Tina Memo No. 2005-011
Modified version to appear in IJCV, 68-2, 203-216, 2005.

Comparing the Performance of Least-Squares Estimators: when is GTLS Better than LS?

A. Nayak, E. Trucco and N.A. Thacker

Last updated
3 / 5 / 2005

This document forms part of the **Features and Measurement** Series available from www.tina-vision.net.

WWW.TINA-VISION.NET

Imaging Science and Biomedical Engineering Division,
Medical School, University of Manchester,
Stopford Building, Oxford Road,
Manchester, M13 9PT.

Comparing the Performance of Least-Squares Estimators: when is GTLS Better than LS?

Arvind Nayak[1], Emanuele Trucco[1] and Neil A. Thacker[2]
1: Signal & Image Processing Research Area, EECE-EPS,
Heriot-Watt University, Edinburgh - EH14 4AS, Scotland, UK.
2: Division of Imaging Science and Biomedical Engineering Division,
Medical School, University of Manchester, Manchester, M13 9PT, UK
corresponding author: amn1@hw.ac.uk

Abstract

Several computer vision problems lead to linear systems affected by noise. These are commonly solved by least-squares estimators, the most popular being ordinary least squares (LS), total least squares (TLS) and generalized total least squares (GTLS). However, the statistical or structural assumptions of these theoretical estimators are very often violated in practice. Given that their computational cost is very different, what should one choose, and in which conditions? We give empirical guidelines, in the absence of a general theoretical answer, observing the behaviour of errors of LS, TLS and GTLS with a representative computer vision problem (homography estimation) in varying noise conditions (noise distribution, intensity, correlation level, added to image co-ordinates or to system matrix). We find that the much more expensive GTLS brings significant advantages with high intensity, highly correlated noise in the system matrix; in other conditions, the three estimators give comparable errors.

`Least squares, Total least squares, Generalized total least squares,`
`2-D Homography, Correlated noise.`

1 Introduction

Estimation problems are ubiquitous in computer vision (CV), and considerable effort has been directed towards devising well-behaved estimation algorithms for a variety of problems of practical interest. Ordinary least-squares (LS) [GvL96, Str88], is possibly the most used class of estimators. However, in several practical situations, the standard system $A\mathbf{x}=\mathbf{b}$ violates LS statistical assumptions (e.g., both A and $\mathbf{b}$ very often contain noise), or structural assumptions (e.g., the noise may corrupt only a subset of entries of A, the rest being constant). Empirical (simulated) error analyses exist on the effects of noise and outliers [CC91, MM98], but not, to our knowledge, on the effects of noise with varying correlation or violation of the structural hypotheses.

Awareness of the limits of ordinary LS has prompted interest for more general techniques. The ones featuring most frequently in the CV literature are *total least squares* (TLS) [dG96, GvL96] and *generalized total least squares* (GTLS) [vHV89], although other variations exist [CGGS98, DC96, GL97, GvL96, MM98, PS02, KMvH02, KMvH04]. These algorithms cover more general situations than ordinary LS by relaxing structural and statistical assumptions. GTLS, in particular, cover a rather general case: constant and noisy columns in $[A;\mathbf{b}]$ and correlated noise. Van Huffel and Vandewalle [vHV89] introduced the GTLS algorithm used in our analysis, and a study of its theoretical properties.

As CV practitioners, we are interested to predict the behaviour of errors, i.e., the quality of estimates; unfortunately, the assumptions made by theoretical results do not always apply in practice. For instance, the structure of the system matrix, A, does not meet the GTLS assumptions in several, common CV problems (Section 3). We address therefore two questions of practical interest.

1. If the noise is correlated, what levels of correlation and intensity justify dropping LS in favour of GTLS, a much more expensive algorithm in terms of complexity?
2. Since the theory of the three estimators consider additive noise on entries of the system matrix, how do errors change adding noise with given statistical properties to the system matrix as opposed to the image co-ordinates?

We address these questions by comparing experimental LS, TLS and GTLS errors in the presence of varying correlated, additive, normally and uniformly distributed noise affecting either the system matrix or the image co-ordinates, as well as violations of the assumptions on the system structure. We do not consider outliers in this exercise.

The testbed CV problem chosen is homography estimation from image correspondences, leading to a linear system with properties similar to those of systems arising in several other CV problems, e.g., calibration [HZ02], motion estimation [CC91, MM98, MM01, KMvH02], and view interpolation [KBG99].

The structure of the paper is as follows. Section 2 discusses briefly some relevant, related work. Section 3 is a succinct reminder of the three least squares techniques under scrutiny, their assumptions and algorithms. Section 4 introduces the reference problem of our analysis (estimation of the homography between two views from point correspondences). Section 5 describes our experimental protocols and presents our findings. Section 6 is a concise discussion of our results.

2 Related computer vision work

CV researchers have been aware of TLS and, to a lesser extent, GTLS, for quite some time. Weber and Malik [WM94] report an application of TLS in the context of multi-scale optic flow; Chaudhuri and Chatterjee [CC91] analyse the performance of TLS for motion problems (see below). More recently, Kennedy et al. [KBG99] adopt TLS for view synthesis by linear combination of views; they observe that GTLS should actually be used as TLS structural assumptions are not fully met in all cases, but do not report GTLS results.

Relatively few CV authors have compared the performance of different least squares estimators in typical vision problems. Chaudhuri and Chatterjee [CC91] study LS and TLS errors with linear, 3-D motion estimation from point correspondences. They plot the error in the Frobenius norm of the rotation matrix, as well as on individual entries. The analysis include sensor noise and outliers (possible mismatches in the correspondence set). The authors derive the Cramer-Rao lower bound of the error covariance matrix under the assumption of uncorrelated, additive Gaussian noise, concluding that the bound for TLS is always higher than that for LS. We take this as suggesting that the TLS estimate is more realistic, rather than being an indication of superior performance for LS. The possible effects of correlated noise and structural hypothesis violation, or the application of GTLS are not considered.

A recent, comprehensive treatment of least-squares methods in CV is given by Mühlich and Mester [MM98, MM99, MM01]. In [MM98], they analyse the statistical properties of four variations of the 8-point algorithm for estimating the essential matrix: plain TLS with no data normalisation, two data transformations suggested by Hartley [Har97], and a statistically optimal transformation they derive in [MM99]. The latter includes a GTLS-like procedure to take care of noise-free columns. Simulations are reported with 10 levels of additive, uniform noise on co-ordinates of corresponding points for the case of translation only. Correlation is not considered. More recently, the statistical optimisation is put in the general framework of subspace methods and equilibration [MM01, MM99], deemed to lead to better estimates with non-recursive methods and theoretically sound data transformations. Importantly, the authors observe that subspace methods can withstand substantial amounts of identically distributed, independent noise, but violating this assumption (e.g., correlated noise) may lead to serious errors [MM99]. No systematic analysis of this point is reported.

Leedan [Lee97] studies the estimation of parameters of quadratic problems in computer vision. Various linear and non-linear estimators including GTLS are compared under varying amounts of zero-mean normal noise on image co-ordinates only. In one experiment, pixel correlation is introduced by magnifying an image (pixel replication).

Another motivation for our analysis is that, in practice, additive noise is normally assumed *on image co-ordinates*, but least-squares theory considers additive noise *on the entries of the system matrix.* Such entries are functions of the co-ordinates, so that the noise affecting matrix entries is statistically different from the noise affecting image co-ordinates. For example, assuming independent, identically distributed (iid) noise on co-ordinates does not guarantee iid noise on the system matrix, as requested by the estimators considered. This is rightly pointed out by Kukush et al. [KMvH02] who report algorithms for consistent estimation that take this situation into account. A simple example is provided by matrix entries formed by multiplying co-ordinates affected by additive Gaussian noise. The pdf of the product of two independent, normal variables is *not* normal, but an analytically complex distribution [Spr79] illustrated in Figure 1. We look at the consequences of this problem for the case of LS and TLS by obtaining estimation errors in two situations; one, when noise of varying conditions is added to elements of the noise-free (reference) system matrix; second, when noise of similar conditions is added to the normalized (mentioned later) image co-ordinates.

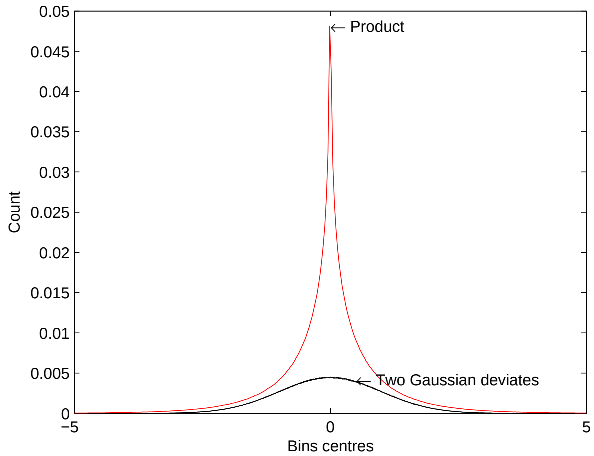


Figure 1: Experimental pdf (histogram), normalised to unit area, of the product of two independently derived, standardised Gaussian deviates (10 million samples).

3 Brief review of LS, TLS, GTLS

This section summarises the three algorithms considered and their assumptions, referring the reader to detailed treatments wherever appropriate. Our account follows mainly [MM98, vHV89].

3.1 Ordinary least squares

Consider the problem of estimating $\mathbf{x} \in \mathbb{R}^n$ in the standard overdetermined linear system

$$A\mathbf{x} = \mathbf{b}, \tag{1}$$

where $A \in \mathbb{R}^{m \times n}$ is the *data matrix* and $\mathbf{b} \in \mathbb{R}^m$ the *observation vector*, and $m \geq n$.

Ordinary LS assume additive noise affecting only $\mathbf{b}$. A is assumed noise-free (constant). Such assumptions are often violated in practice. If, in addition, the noise is also Gaussian, the solution is the maximum likelihood estimate, i.e., optimal in the ML sense [MM99, MMRK91]. It is well-known that the LS solution of Equation (1) is

$$\hat{\mathbf{x}}_\ell = (A^T A)^{-1} A^T \mathbf{b}. \tag{2}$$

More generally (e.g., if the data matrix does not have maximum rank), the solution can be computed by singular value decomposition (SVD) [GvL96, Str88]. If $A = F\Sigma G^T$ is the SVD of A, where Σ is the matrix of the singular values (in decreasing order), and F and G are unitary matrices, then

$$\hat{\mathbf{x}}_\ell = A^+ \mathbf{b}, \tag{3}$$

where $A^+ = G\Sigma^+ F^T$ is the *pseudoinverse* of A, and Σ^+ is the diagonal matrix of the reciprocals of the singular values.

3.2 Total least squares

In practice, A is nearly invariably a function of image measurements; the assumption of an error-free data matrix is hardly ever satisfied. *Total least squares (TLS)* [GvL96, CC91, MM98, dG96, KBG99, WM94] assume noise in both A and $\mathbf{b}$. Additive noise must affect all entries of $[A; \mathbf{b}]$, and be independent, zero-mean and equilibrated, i.e., all additive-noise variables have equal variance (but if not, well-known data transformations can be applied to meet this requirement). In these assumptions, the TLS estimate below is, under mild conditions [vHV89], a strongly consistent estimate of the true value:

$$\hat{\mathbf{x}}_t = (A^T A - \phi_{ts} I)^{-1} A^T \mathbf{b}, \tag{4}$$

where $I \in \mathbb{R}^{n \times n}$ is the identity matrix, and ϕ_{ts} is the smallest singular value of the measurement matrix $[A; -\mathbf{b}]$, such that

$$[A; -\mathbf{b}] \begin{bmatrix} \mathbf{x} \\ \mathbf{1} \end{bmatrix} = \mathbf{0}. \tag{5}$$

The LS solution, Equation (2), is found as a special case of equation (4) when $\phi_{ts} = 0$.

3.3 Generalized total least squares

The *Generalized total least squares (GTLS)* [vHV89] relax further the LS assumptions by allowing some error-free (constant) columns in A, and correlated noise in the remaining columns of A as well as in $\mathbf{b}$. The problem and its solution can be summarised as follows.

$$\text{Partition } A = [A_1; A_2], \qquad A_1 \in \mathbb{R}^{m\times n_1}, A_2 \in \mathbb{R}^{m\times n_2}, n = n_1 + n_2 \tag{6}$$

$$\mathbf{x} = [\mathbf{x}_1^T; \mathbf{x}_2^T]^T, \qquad \mathbf{x}_1 \in \mathbb{R}^{n_1\times 1}, \mathbf{x}_2 \in \mathbb{R}^{n_2\times 1} \tag{7}$$

Assume that A_1 is known exactly (i.e., it contains all the constant columns of A), and that $C = E(\Delta^T\Delta) \in \mathbb{R}^{(n_2+1)\times(n_2+1)}$ is (up to a proportionality factor) the covariance matrix of the errors $\Delta \in \mathbb{R}^{m\times(n_2+1)}$ in the perturbed matrix $[A_2; \mathbf{b}]$. Let $R_C^T R_C$ (nonsingular) be the Cholesky decomposition of C. Then a GTLS solution of problem (1) is any solution of

$$\hat{A}\mathbf{x} = A_1\mathbf{x}_1 + \hat{A}_2\mathbf{x}_2 = \hat{\mathbf{b}} \tag{8}$$

where $\hat{A} = [A_1; \hat{A}_2]$ and $\hat{\mathbf{b}}$ are determined so that

$$\| [\Delta\hat{A}_2; \Delta\hat{\mathbf{b}}]R_C^{-1} \|_F = \| [A_2 - \hat{A}_2; \mathbf{b} - \hat{\mathbf{b}}]R_C^{-1} \|_F \tag{9}$$

is minimal, and Range($\hat{\mathbf{b}}$) $\subseteq$ Range($\hat{A}$).

To compute a solution, we consider the algorithm given in [vHV89], which is designed to maintain consistency under the assumptions stated at the beginning of this section. The algorithm hinges on the *generalized singular value decomposition* [GvL96, vHV89] and covers the case of singular covariance matrix. An advantage is that a consistent estimate is computed without explicit data transformations (see [vHV89] for details). Notice however that, in our experiments, we do apply Hartley's data transformation [Har97] to achieve better LS estimates.

Our implementation of van Huffel's GTLS algorithm adopts Gallo's solving formula [Gal82], given in Equation (12), which, as shown in [vHV89], is equivalent to van Huffel's but under slightly different conditions. The reason is best explained after the following, brief description of algorithm.

- Consider Equations (6) and (7). A_1 has full column rank n_1 and is known exactly.
- Let the covariance matrix $E(\Delta^{*T}\Delta^*)$ of the errors Δ^* in $[A, \mathbf{b}] = [A_1, A_2, \mathbf{b}]$ be given by C^*:

$$C^* = \begin{bmatrix} C_a^* & \mathbf{c}_{ab}^* \\ \mathbf{c}_{ab}^{*T} & c_b^* \end{bmatrix}, \tag{10}$$

$$C^* \in \mathbb{R}^{(n+1)\times(n+1)}, C_a^* \in \mathbb{R}^{n\times n}, \mathbf{c}_{ab}^* \in \mathbb{R}^{n\times 1}, \mathbf{c}_{ab}^{*T} \in \mathbb{R}^{1\times n} \text{ and } c_b^* \in \mathbb{R}^{1\times 1}$$

- Let R_C^* be any square root of C^*, defined by $C^* = R_C^{*T} R_C^*$ and partitioned as follows:

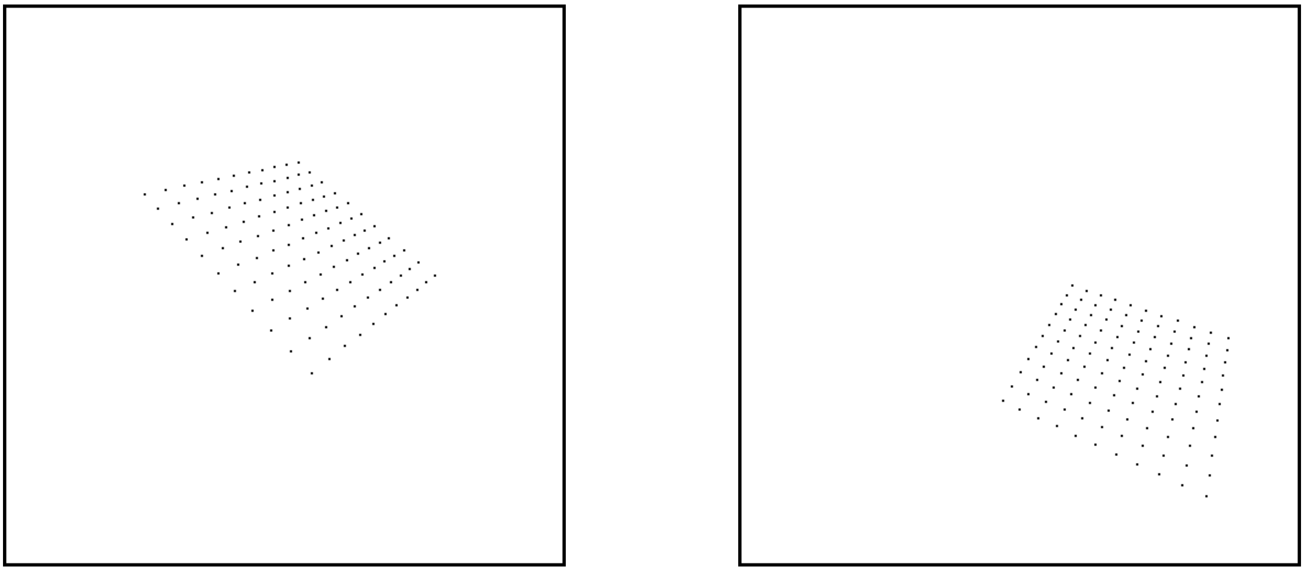
$$R_C^* = \begin{bmatrix} R_{C_a}^* & \mathbf{r}_{C_{ab}}^* \\ \mathbf{r}_{C_{ba}}^* & r_{C_b}^* \end{bmatrix}, \tag{11}$$

$$R_C^* \in \mathbb{R}^{(n+1)\times(n+1)}, R_{C_a}^* \in \mathbb{R}^{n\times n}, \mathbf{r}_{C_{ab}}^* \in \mathbb{R}^{n\times 1}, \mathbf{r}_{C_{ba}}^* \in \mathbb{R}^{1\times n} \text{ and } r_{C_b}^* \in \mathbb{R}^{1\times 1}$$

- The solution $\hat{\mathbf{x}}_g$ is given by

$$\hat{\mathbf{x}}_g = (A^T A - \phi_g^2 C_a^*)^{-1}(A^T\mathbf{b} - \phi_g^2 \mathbf{c}_{ab}^*). \tag{12}$$

The reason for choosing Gallo's formula is that our reference problem generates a numerical system meeting the uniqueness condition of Gallo's solution, i.e., $\phi_g' = \min(gsvd[A, R_{C_a}^*]) > \phi_g = \min(gsvd[A, \mathbf{b}], R_C^*)$, but not necessarily van Huffel's.



(a) (b)

Figure 2: (a) and (b), synthetically generated images of 121 point correspondences from a known homography matrix ($\mathbf{x}_{true}$).

4 Reference problem: Estimation of 2-D homography

An homography is a non-singular linear transformation of the projective plane onto itself. Given two sets of corresponding image points in projective co-ordinates, $\mathbf{p}, \mathbf{p}' \in \mathbb{P}^2$, we want to estimate the homography mapping each $\mathbf{p}$ to the corresponding $\mathbf{p}$ [HZ02]. The homography sought is a non-singular 3×3 matrix H such that

$$\gamma \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{13}$$

As known, H is defined up to a scale factor, hence the problem has 8 degrees of freedom. Each point provides two equations, so a minimum of 4 point correspondences are necessary to fully constrain H. In our experiments, we take 121 point correspondences to over-constrain the system, and fix the unknown scale factor by assuming $h_{33} = 1$. The point correspondences, $\mathbf{p}$ and $\mathbf{p}'$, are obtained from two synthetically generated images, Figures 2(a) and (b) respectively, with a known homography matrix.

The equation contributed by each corresponding pair can be re-arranged to obtain

$$\begin{aligned} u'(h_{31}u + h_{32}v + h_{33}) &= h_{11}u + h_{12}v + h_{13} \\ v'(h_{31}u + h_{32}v + h_{33}) &= h_{21}u + h_{22}v + h_{23}. \end{aligned} \tag{14}$$

A set of n such equation pairs, contributed by n point correspondences, form an overdetermined linear system $A\mathbf{x} = \mathbf{b}$. To make the system as consistent as possible with the structure assumed by the GTLS algorithm, we move the constant columns of A (formed by 0 and 1) to the left:

$$\begin{bmatrix} 1 & 0 & u_1 & v_1 & 0 & 0 & -u_1 u_1' & -u_1' v_1 \\ 0 & 1 & 0 & 0 & u_1 & v_1 & -u_1 v_1' & -v_1 v_1' \\ \vdots & & & & \vdots & & & \vdots \\ 1 & 0 & u_n & v_n & 0 & 0 & -u_n u_n' & -u_n' v_n \\ 0 & 1 & 0 & 0 & u_n & v_n & -u_n v_n' & -v_n v_n' \end{bmatrix} \begin{bmatrix} h_{13} \\ h_{23} \\ h_{11} \\ h_{12} \\ h_{21} \\ h_{22} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} u_1' \\ v_1' \\ \vdots \\ u_n' \\ v_n' \end{bmatrix} \tag{15}$$

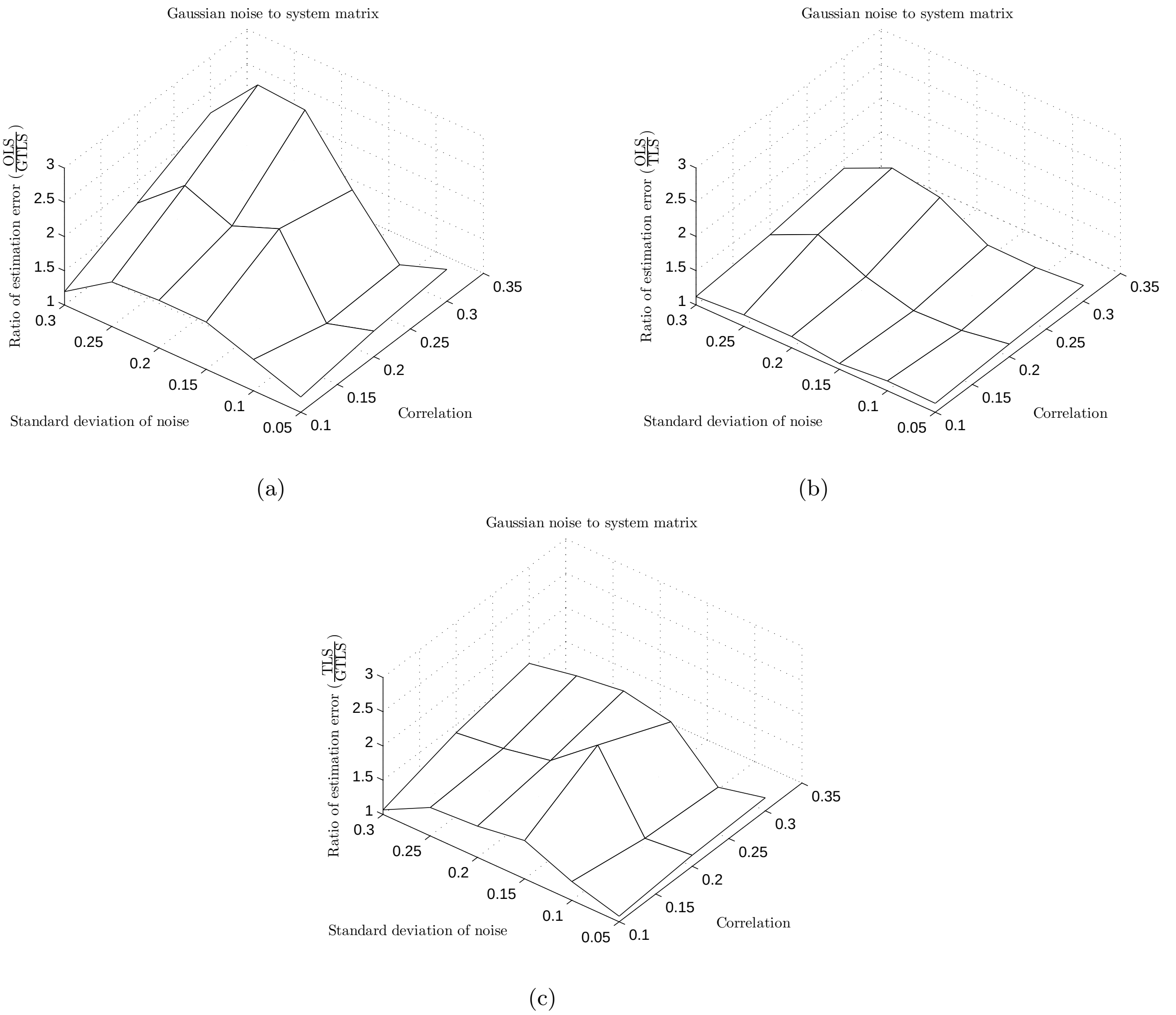


Figure 3: Ratios of estimation errors ($\mathcal{E}_{LG}$, $\mathcal{E}_{LT}$, $\mathcal{E}_{TG}$) with Gaussian distributed noise added to matrix entries. X axis (right) is correlation ρ, Y axis (left) is standard deviation of noise (Normal distributed) σ.

The first two columns of A are now constant (and the elements of $\mathbf{x}$ have been re-arranged accordingly). Notice however that, contrary to GTLS structural assumptions, constant entries are not just confined to the leftmost columns. To improve the conditioning of the problem and the properties of the LS and TLS solutions, we apply Hartley's data normalisation [Har95]. The final estimate is obtained by inverse-transforming the result of the normalised problem.

5 Experimental results

This section describes our experiments with a linear system arising from a realistic CV situation, homography estimation. We observe the errors of LS, TLS and GTLS estimators with varying levels of noise intensity and correlation. Structural and statistical assumptions of the theoretical algorithms are violated, sometimes deliberately (e.g., using uniform noise instead of Gaussian noise), sometimes by necessity (e.g., the system structure is a consequence of the geometry of the problem). We intend to compare the quantitative consequences of such violations, and ultimately to formulate empirical guidelines for choosing LS, TLS or GTLS in practical CV problems. We consider separately additive noise on image co-ordinates and on matrix entries, this being the only one considered in theoretical results. Throughout, the error metric adopted is *ratios of estimation errors* avoiding arbitrary scaling altogether:

$$\mathcal{E}_{LG} = \frac{\|\hat{\mathbf{x}}_\ell - \mathbf{x}_{true}\|_F}{\|\hat{\mathbf{x}}_g - \mathbf{x}_{true}\|_F}, \mathcal{E}_{LT} = \frac{\|\hat{\mathbf{x}}_\ell - \mathbf{x}_{true}\|_F}{\|\hat{\mathbf{x}}_t - \mathbf{x}_{true}\|_F}, \mathcal{E}_{TG} = \frac{\|\hat{\mathbf{x}}_t - \mathbf{x}_{true}\|_F}{\|\hat{\mathbf{x}}_g - \mathbf{x}_{true}\|_F}. \tag{16}$$

5.1 Additive noise on system matrix entries

This experiment observes the system error in the presence of normally distributed noise on entries of A and $\mathbf{b}$. For completeness and reproducibility, we give the pseudo-code for this particular experiment.

Algorithm

```
Start
  function ArrangeModel{corresponding points in two images}
    Compute Transformed-Coordinates
    Form System-Matrices ([A], [x], [b])
    Re-arrange the columns of the System-Matrices
  return System-Matrices
  Backup the System-Matrices
  function AutoGenerate{[A], [b]}
    Compute the standard deviation extremes (stdstart, stdstep, stdend)
  return stdstart, stdstep, stdend
  Specify correlation coefficient extremes (corrstart, corrstep, corrend)
  Specify iterations (iterate)
  for(corrcoef=corrstart:corrstep:corrend)
    for(stddev=stdstart:stdstep:stdend)
      for iter=1:iterate
        function GenerateCorrelatedNoise{mean, stddev, corrcoef}
          while(stddev of [noise_to_A] and [noise_to_b] and their
                correlation coefficient not within specified tolerance)
            re-generate correlated noise ([noise_to_A], [noise_to_b])
        return [noise_to_A], [noise_to_b]
        [A_noise]=[A]+[noise_to_A]
        [b_noise]=[b]+[noise_to_b]
        function LeastSquares([A_noise],[b_noise])
          Compute estimate [x_ls]
        return [x_ls]
        error_ls=norm([x_ls]-[x_true])
        function TotalLeastSquares([A_noise],[b_noise])
          Compute estimate [x_tls]
        return [x_tls]
        error_tls=norm([x_tls]-[x_true])
        function GeneralizedTotalLeastSquares([A_noise],[b_noise])
          Compute estimate [x_gtls]
        return [x_gtls]
        error_gtls=norm([x_gtls]-[x_true])
      end (* iter for *)
      Avg_error_ls=average(error_ls)
      Avg_error_tls=average(error_tls)
      Avg_error_gtls=average(error_gtls)
      Ratio_LG = Avg_error_ls/Avg_error_gtls
      Ratio_LT = Avg_error_ls/Avg_error_tls
      Ratio_TG = Avg_error_tls/Avg_error_gtls
    end (* stddev for *)
  end (* corrcoef for *)
Stop
```

The co-ordinates of 121 point correspondences were normalised and the standard matrices A and $\mathbf{b}$ were then formed (`function ArrangeModel`) as described in Section 4. All estimates obtained in the tests were compared against the known homography (`x_true`). We added noise with a particular distribution (normal, uniform), correlation and standard deviation to the entries of A and $\mathbf{b}$. The level of the noise was decided from entries of the noise-free A and $\mathbf{b}$ (`function AutoGenerate`). As the numerical generation of correlated, random noise did not guarantee that every realisation of the noise met the specified requirements, each random-noise matrix generated was accepted only if its standard deviation and correlation, estimated numerically, were within strict tolerances of the expected values (`function GenerateCorrelatedNoise`). The tolerance limits were $\pm 10\%$ of the expected value for standard deviation and $\pm 10\%$ of the expected value for correlation. These tolerance limits were set empirically on the basis of computational time and of the effect on errors.

Figures 3 (a), (b) and (c) show the ratios of estimation error ($\mathcal{E}_{LG}$, $\mathcal{E}_{LT}$, $\mathcal{E}_{TG}$) as described in Equation (16) against correlation coefficient, ρ, and the standard deviation of the additive Gaussian noise, σ. Each value plotted was obtained by averaging 50 iterations, which seemed sufficient because the synthetic noise realisations were accepted only if satisfying the required theoretical constraints described earlier. In this set σ varied between 5% (0.05) and 30% (0.3), in steps of 5%, of the standard deviation of the entries of the noise-free (reference) version of A and $\mathbf{b}$ (in our case, ~ 1.0). The correlation coefficient varied between 0.1 and 0.3 in steps of 0.1.

Similarly, errors are also obtained in the presence of uniform distributed noise on the matrix entries. Results are shown in Figures 4, (a), (b) and (c).

For easy visual comparison Figure 5 (a) and (b) show the plot for normal and uniform distributed noise respectively. The standard deviation of noise in the two plots is fixed to 30% of the standard deviation of the entries of the noise-free (reference) version of A and $\mathbf{b}$. The ratios of errors are then plotted against increasing correlation (0.1 to 0.6).

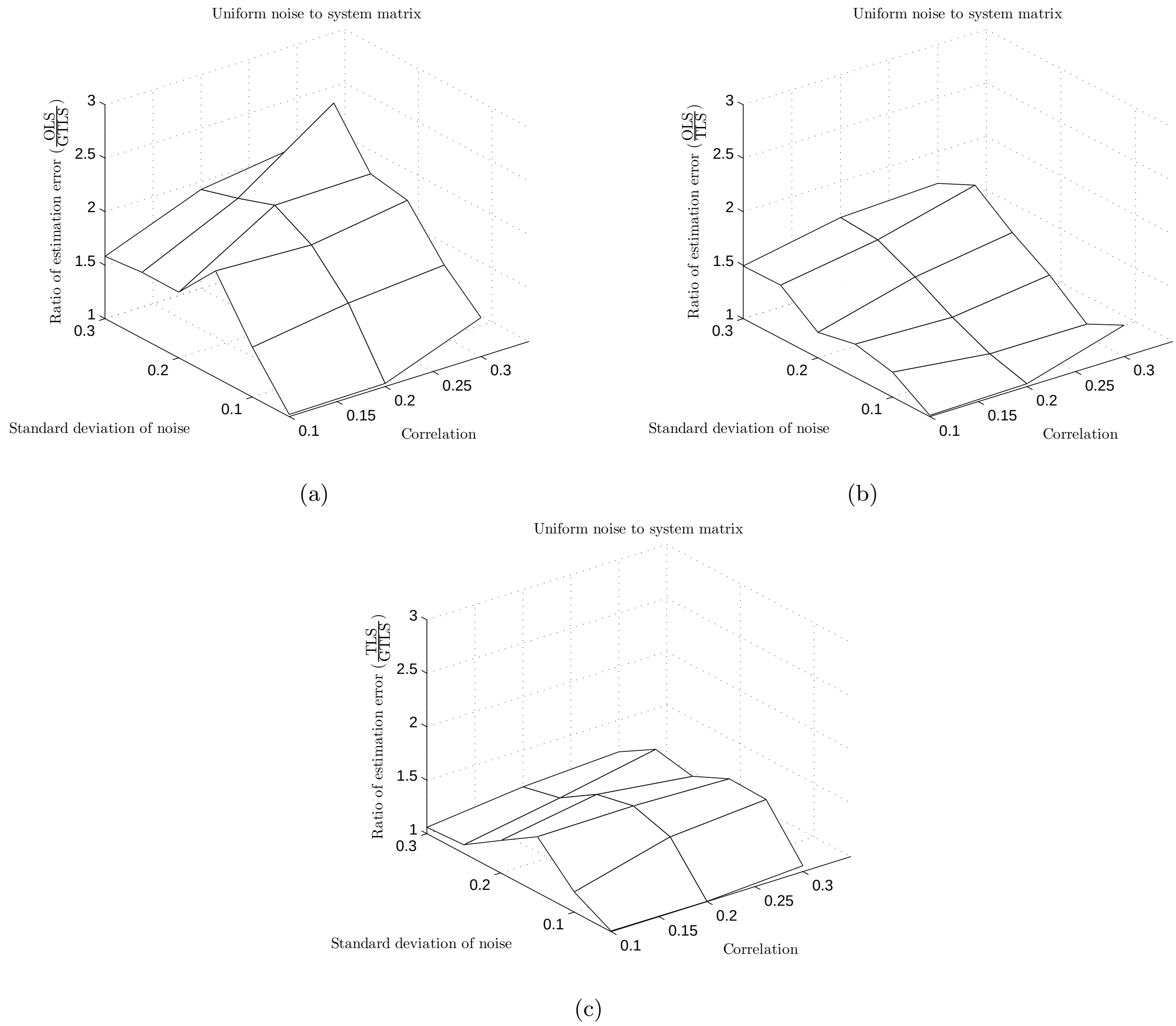


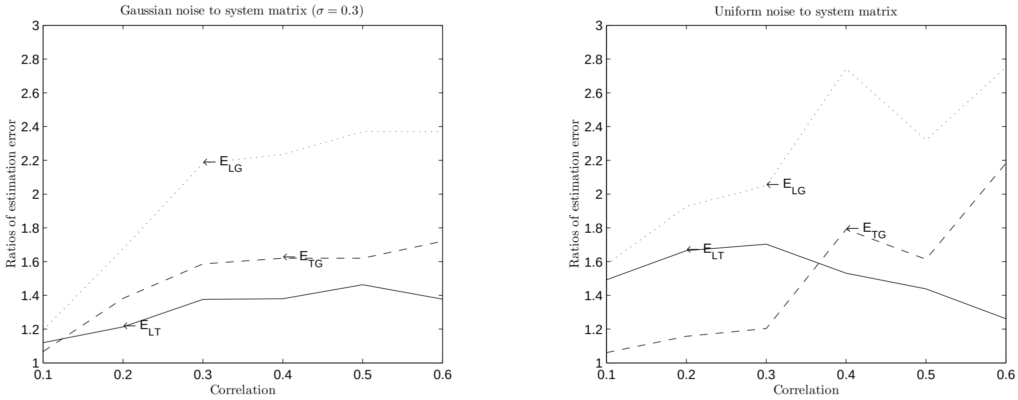
Figure 4: Ratios of estimation errors ($\mathcal{E}_{LG}$, $\mathcal{E}_{LT}$, $\mathcal{E}_{TG}$) with uniform distributed noise added to matrix entries. X axis (right) is correlation ρ, Y axis (left) is standard deviation of noise (Uniform distributed) σ.

Since our error metric is the ratio of estimation errors (see Equation 16) it is difficult to visualize the magnitude of errors we are dealing with. Hence, Figure 6 shows a typical plot of magnitude of errors in GTLS estimates for Gaussian and uniform distributed noise. Again, the standard deviation of noise was fixed to 30% of the standard deviation of the entries of the noise-free (reference) version of A and $\mathbf{b}$.

The results show that, for small amounts of noise, the three estimators give very similar results. Errors under uniform distributed noise of some fixed σ are in general greater than errors under normal distributed noise with same σ. On average the error in GTLS estimates is lower than LS and TLS estimates irrespective of noise levels, distribution, correlation. On a more quantitative note, from Figure 3, errors in LS and TLS estimates are on average 1.78 and 1.42 times higher than errors in GTLS estimates respectively for Gaussian noise. For uniform noise these are 1.77 and 1.27 respectively.

5.2 Additive noise on image co-ordinates

This experiment considers errors when noise is added to the normalised co-ordinates. The pseudo-code for this experiment is given next.



(a) (b)

Figure 5: (a) and (b) Ratios of estimation errors ($\mathcal{E}_{LG}$, $\mathcal{E}_{LT}$, $\mathcal{E}_{TG}$) with normal and uniform distributed noise respectively, added to matrix entries. X axis is correlation ρ, Y axis is ratios of estimation errors. The σ of noise is fixed at 0.3 (30%).

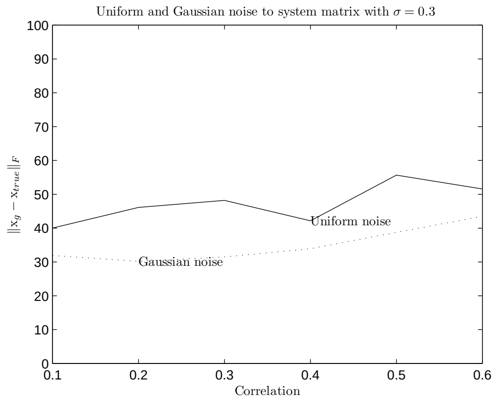
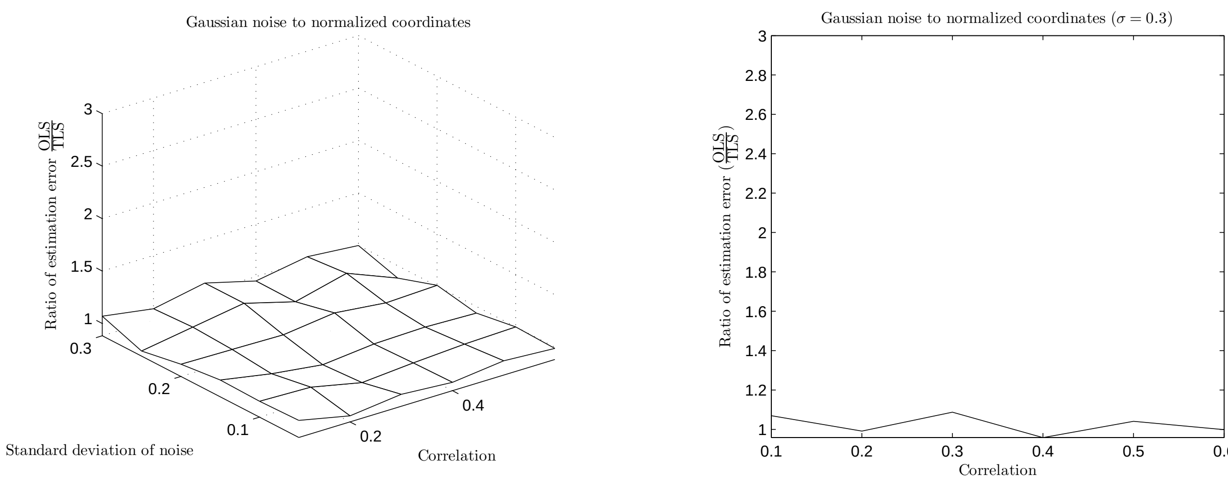


Figure 6: Estimation errors, $\|\hat{\mathbf{x}}_g - \mathbf{x}_{true}\|_F$, for GTLS estimator with Gaussian and uniform distributed noise added to matrix entries shown for visualization of magnitudes. X axis is correlation ρ, Y axis is estimation error. The σ of noise is fixed at 0.3 (30%).



(a) (b)

Figure 7: Ratio of estimation error ($\mathcal{E}_{LT}$) when Gaussian distributed noise is added to normalised co-ordinates. X axis (right) is correlation ρ, Y axis (left) is standard deviation of noise σ.

Algorithm

```
Start
  Compute Transformed-Coordinates
  function AutoGenerate{[u], [u']}
    Compute the standard deviation extremes (stdstart, stdstep, stdend)
  return stdstart, stdstep, stdend
  Specify correlation coefficient extremes (corrstart, corrstep, corrend)
  Specify iterations (iterate)
  for(corrcoef=corrstart:corrstep:corrend)
    for(stddev=stdstart:stdstep:stdend)
      for iter=1:iterate
        function GenerateCorrelatedCoodNoise{mean, stddev, corrcoef}
        while(stddev of [noise_to_u] and [noise_to_u'] and their
              correlation coefficient not within specified tolerance)
           re-generate correlated noise ([noise_to_u], [noise_to_u'])
        return [noise_to_u], [noise_to_u']
        [u_noise]=[u]+[noise_to_u]
        [u'_noise]=[u']+[noise_to_u']
        function ArrangeModel{corresponding points in two images}
          Form System-Matrices ([A], [x], [b])
          Re-arrange the columns of the System-Matrices
        return System-Matrices
        function LeastSquares([A],[b])
          Compute estimate [x_ls]
        return [x_ls]
        error_ls=norm([x_ls]-[x_true])
        function TotalLeastSquares([A],[b])
          Compute estimate [x_tls]
        return [x_tls]
        error_tls=norm([x_tls]-[x_true])
        function GeneralizedTotalLeastSquares([A],[b])
          Compute estimate [x_gtls]
        return [x_gtls]
        error_gtls=norm([x_gtls]-[x_true])
      end (* iter for *)
      Avg_error_ls=average(error_ls)
      Avg_error_tls=average(error_tls)
      Avg_error_gtls=average(error_gtls)
      Ratio_LG = Avg_error_ls/Avg_error_gtls
      Ratio_LT = Avg_error_ls/Avg_error_tls
      Ratio_TG = Avg_error_tls/Avg_error_gtls
    end (* stddev for *)
  end (* corrcoef for *)
Stop
```

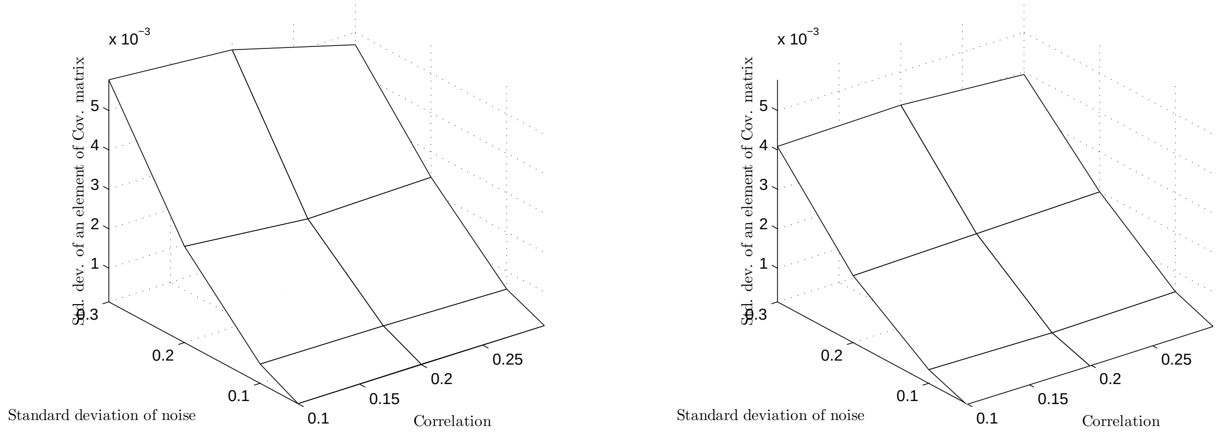
The rest of the estimation procedure was similar to the one described in Subsection 5.1, but the noise was added to the normalized co-ordinates (`function GenerateCorrelatedCoodNoise`), not to A and $\mathbf{b}$. The statistics of noise was similar to that defined for experiments in Subsection 5.1, as the mean and standard deviation of the entries of A and $\mathbf{b}$ are close to those of the normalised co-ordinates.

In this case we cannot control the amount of correlation on the system matrix nor vary it systematically because of the reason described in Section 2. In addition, the covariance matrix of the noise in the system matrix is not known exactly, as was the former case when noise was added to the system matrix. The GTLS solution, hence, should be computed by an iterative procedure generating a new covariance matrix every time until the solution converges. We do not consider this case here as the covariance matrix generated during each iteration cannot be related to the noise on actual co-ordinates. Hence, we present only the results for LS and TLS in Figure 7 (a) and (b) respectively. The plots show that the relative errors are smaller than those obtained by adding the same level of noise to the noise-free (reference) system matrix. This is in line with the plot shown in Figure 1, where it can be clearly seen that the samples of products of two standardized Gaussian deviates, $\mathcal{N}(0,1)$, which is approximately our case, concentrate near the null mean. This effectively means the actual level of noise on the system matrix will be smaller.

5.3 Consistency test of noise covariance matrix for GTLS

This observation is specifically concerned with the GTLS estimator. As described in Subsection 3.3, the GTLS estimator assumes that the covariance matrix, C^*, of the noise added to the system matrix satisfies certain conditions. We wanted to observe how much the elements of the C^* deviate during each iteration at particular level of noise and correlation. This observations help us understand whether the estimation becomes inconsistent during several passes at the same level of noise and correlation.

Figures 8 (a) and (b) show plots of standard deviation of two typical elements of C^*. These plots are representatives for those of the other entries as well. Each value was obtained over `50` iterations (`50` samples to estimate the standard deviation). For clarity, we show in Table 1 some of the numerical values from Figure 8 (a). The results



(a) (b)

Figure 8: Standard deviation of two elements of the noise covariance matrix (C^*) taken over 50 iterations. X axis (right) is correlation ρ, Y axis (left) is standard deviation of noise (Gaussian distributed) σ.

		ρ		
		0.1	0.2	0.3
	0.05	0.00014	0.00015	0.00014
σ	0.10	0.00063	0.00061	0.00055
	0.20	0.00259	0.00229	0.00236
	0.30	0.00578	0.00555	0.00469

Table 1: Standard deviation values of a typical element of the covariance matrix (C^*). Each value is obtained by observing 50 samples at a particular level of noise (σ) and correlation (ρ).

show that the standard deviation of the elements of C^* observed over range of iterations (50 here) is far below 2% of the theoretical standard deviation of the noise that was used to obtain the covariance matrix. This shows that C^* does not change drastically in practice; hence, the estimation obtained is practically consistent.

6 Conclusions

This paper presented an experimental investigation into the behaviour of three least-squares estimators, LS, TLS, and GTLS. Our work was prompted by the observation that common CV problems, in different ways, do not meet in full the assumptions of *any* theoretical algorithm, based upon closed form solutions of linear equations. Though clearly, in many cases, exact likelihood formulations could be constructed for an iterative solution. We could legitimately ask if a non-conformity to theory is a problem we should be worried about. Our view is that algorithms which make invalid statistical assumptions are likely to generate biased estimates, where the bias varies according to the distribution of data. The problem therefore is one of algorithm characterisation, an algorithm which produces data dependant bias is not charaterisable in any simple way. This issue may be recognised as one of the key criticisms leveled against algorithms in this literature over the last few decades. That is; algorithms have performance which cannot be predicted on datasets which are different to those in the original publications, thus limiting the scientific value of work. It is therefore important to design algorithms so that they are expected to be valid for the statistical characteristics of the data under analysis. In this case it therefore becomes necessary to compare errors in a variety of conditions, with the purpose of establishing practical guidelines for choosing an estimator, especially as the complexity of GTLS is much greater than that of TLS or LS. The results allow us to answer the two questions posed in Section 1.

What levels of noise correlation and intensity justify dropping LS in favour of the more costly GTLS? For small levels of noise, the three estimators showed similar performance; on average, GTLS yielded the smallest errors. The higher computational cost of GTLS seems justified only with intense and strongly correlated noise affecting the system matrix, in which case the GTLS leads to significantly smaller errors than LS and TLS. This suggests that, in practice, GTLS gives real benefits if the intensity and correlation levels are significant. The numerical values in our experiments can be used for guidance.

What is the difference between assuming that noise with given statistical properties is added to image co-ordinates and assuming that it is added to the system matrix? Errors are smaller when adding noise to image co-ordinates than when adding the noise with similar conditions to the system matrix. This is because the distribution of the product of two standardized Gaussian deviates, which is approximately our case, is concentrated near the null mean. Additive noise on the system matrix, at a parity of intensity, (equivalent variance) is the more taxing situation in practice, leading to worse estimates.

Finally, we stress that our results were obtained in specific conditions, but these have been chosen to reflect system structures and values representative for common CV problems. Hence, our conclusions provide reasonably general guidelines, although not exact rules, to choose a LS estimator.

Acknowledgement

Arvind Nayak is supported by a BAE SYSTEMS studentship within the Strategic Alliance programme. We thank Sabine Van Huffel, Ivan Markovsky and Alexander Kukush for useful discussions and input.

References

[CC91] S. Chaudhuri and S. Chatterjee. Performance Analysis of Total Least Squares Methods in Three-Dimensional Motion Estimation. *IEEE Transactions on Robotics and Automation*, 7(5):707–714, October 1991.

[CGGS98] S. Chandrasekaran, G. H. Golub, M. Gu, and A. H. Sayed. Parameter estimation in the presence of bounded data uncertainties. *SIMAX*, 19(1):235–252, 1998.

[DC96] S. G. Deshpande and S. Chaudhuri. Recursive Estimation of Illuminant Motion from Flow Field. In *Proceedings of International Conference on Image Processing (ICIP)*, Lausanne, Switzerland, 1996.

[dG96] P. de Groen. An Introduction to Total Least Squares. *Nieuw Archief voor Wiskunde, 4th Series*, 14:237–253, 1996.

[Gal82] P. P. Gallo. Consistency of Regression Estimates when Some Variables are Subject to Error. *Communications in Statistics – Theory and Methods*, 11:973–983, 1982.

[GL97] L. El Ghaoui and H. Lebret. Robust solution to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 1997.

[GvL96] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[Har95] R. I. Hartley. In Defense of the Eight-Point Algorithm. In *Proceedings of Fifth International Conference on Computer Vision*, pages 1064–1070, June 1995.

[Har97] R. I. Hartley. In Defense of the Eight-Point Algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, June 1997.

[HZ02] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2002.

[KBG99] D. M. Kennedy, B. F. Buxton, and J. H. Gilby. Application of the Total Least Squares Procedure to Linear View Interpolation. In *Proceedings of British Machine Vision Conference (BMVC 99)*, 1999.

[KMvH02] A. Kukush, I. Markovsky, and S. van Huffel. Consistent Fundamental Matrix Estimation in a Quadratic Measurement Error Model Arising in Motion Analysis. *Computational Statistics and Data Analysis, Special Issue of Matrix Computations and Statistics*, 41(1):3–18, November 2002.

[KMvH04] A. Kukush, I. Markovsky, and S. van Huffel. Consistent Estimation in an Implicit Quadratic Measurement Error Model. *Computational Statistics and Data Analysis, Special Issue of Matrix Computations and Statistics*, 47(1):123–147, August 2004.

[Lee97] Y. Leedan. *Statistical Analysis of Quadratic Problems in Computer Vision*. PhD thesis, Rutgers, The State University of New Jersey, 1997.

[MM98] M. Mühlich and R. Mester. The Role of Total Least Squares in Motion Analysis. In *Proceedings of European Conference on Computer Vision*, 1998.

[MM99] M. Mühlich and R. Mester. Subspace methods and equilibration in computer vision. Technical Report XP-TR-C-21, Johann Wolfgang Goethe-Universität, Applied Physics, 1999.

[MM01] M. Mühlich and R. Mester. Improving motion and orientation estimation using an equilibrated total least squares approach. In *Proc. Int. Conf. on Image Processing ICIP'01*, pages 640–643, 2001.

[MMRK91] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim. Robust Regression Methods for Computer Vision: A Review. *International Journal of Computer Vision*, 6(1):59–70, 1991.

[PS02] C. C. Paige and Z. Strakos. Scaled total least squares fundamentals. *Numerische Mathematik*, 91:117–146, 2002.

[Spr79] M. D. Springer. *The Algebra of Random Variables*. John Wiley & Sons, 1979.

[Str88] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace Jovanovich, 1988.

[vHV89] S. van Huffel and J. Vandewalle. Analysis and Properties of the Generalized Total Least Squares Problem $a\mathbf{x} = \mathbf{b}$ when some or all Columns in a are Subject to Error. *SIAM Journal on Matrix Analysis and Application*, 10(3):294–315, July 1989.

[WM94] J. Weber and J. Malik. Robust Computation of Optical Flow in a Multi-Scale Differential Framework. *International Journal of Computer Vision*, 2:5–19, 1994.