Noname manuscript No.
(will be inserted by the editor)

Sampling Minimal Subsets with Large Spans for Robust Estimation

Quoc Huy Tran · Tat-Jun Chin · Wojciech Chojnacki · David Suter

Received: date / Accepted: date

Abstract When sampling minimal subsets for robust parameter estimation, it is commonly known that obtaining an all-inlier minimal subset is not sufficient; the points therein should also have a large spatial extent. This paper investigates a theoretical basis behind this principle, based on a little known result which expresses the least squares regression as a weighted linear combination of all possible minimal subset estimates. It turns out that the weight of a minimal subset estimate is directly related to the span of the associated points. We then derive an analogous result for *total least squares* which, unlike ordinary least squares, corrects for errors in both dependent and independent variables. We establish the relevance of our result to computer vision by relating total least squares to geometric estimation techniques. As practical contributions, we elaborate why naive distance-based sampling fails as a strategy to maximise the span of all-inlier minimal subsets produced. In addition we propose a novel method which, unlike previous methods, can consciously target all-inlier minimal subsets with large spans.

Keywords Least squares · total least squares · minimal subsets · robust fitting · hypothesis sampling

1 Introduction

One of the earliest recorded usage of minimal subsets in statistical estimation occurred in 1755, when Boscovich attempted to determine the meridian arc near Rome from five measurements [34]. He solved for the two unknowns of the arc using all ten possible pairings of the data. Two of the pairs were ignored for yielding what Boscovich considered to be unusual outcomes, and the remaining estimates were simply averaged for his final result. Boscovich's work predated Gauss's paper on least squares (published 1809), but did not gain traction due to a lack of analytical basis.

Presently however, the usage of minimal subsets has become an integral part of robust parameter estimation, especially in computer vision for the estimation of multiple view geometry from noisy images [16]. This stems from the fact that many robust criteria (e.g., least median squares [32], maximum consensus [8]) do not have closed form solutions. Also, many geometric models of interest (e.g., fundamental matrix) have a large number of parameters, thus sampling and testing model hypotheses from minimal subsets is often the only way to obtain good solutions in reasonable time.

Intuitively, drawing an *all-inlier* minimal subset is not sufficient to guarantee a reasonably good model hypothesis; the inliers therein should also have a large span. To illustrate this notion, consider the problem of line fitting on the 2D data in Fig. 1, where the data has been generated without outliers for simplicity. Two particular choices of (all-inlier) minimal subsets are highlighted; clearly Set A yields a better estimate than Set B, as can be verified by a suitable goodness-of-fit function (e.g., [8, 32]). It is also apparent that the points in Set A are separated by a larger distance.

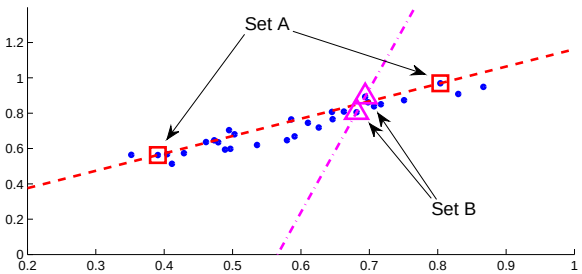


Fig. 1 Line fitting on 2D data. Two particular minimal subset estimates of the line are highlighted.

The Australian Centre for Visual Technologies, and
School of Computer Science, The University of Adelaide,
North Terrace, SA 5005, Australia.
Tel.: +61-8-8313-5586. Fax: +61-8-8313-4366
E-mail: {huy,tjchin,wojtek,dsuter}@cs.adelaide.edu.au

The observation above explains to a large degree previous findings [5,36,33] that the number of randomly drawn minimal subsets required before finding a satisfactory model is often far higher than predicted. The truth is such predictions assume that retrieving a single all-inlier sample is sufficient, which ignores the fact that the all-inlier samples differ in terms of their spatial coverage, and hence intrinsic quality. In effect the predicted number is merely a lower bound of the number of samples required [25].

There remains a lack of theoretical and algorithmic treatments for this long-standing issue. It has been remarked that taking minimal subsets amplifies the noise magnitude [25], however little study was devoted to this phenomenon and, more importantly, the manner in which it affects different minimal subsets. A sampling scheme which aims to sample inliers that fully "cover" the structure was proposed in [33]. However, the algorithm merely raises the expected number of samples required, and does not actively seek all-inlier minimal subsets with a large spatial extent.

This paper aims to show that a more principled reasoning exists for the intuition above. We show that an explanation lies in a little known result which expresses the least squares regression estimate as a weighted linear combination of all possible minimal subset estimates. From this result it is immediately clear that the quality of a minimal subset estimate is proportional to the span of the associated data. As theoretical contributions, we derive a minimal subset expansion for *total least squares* (TLS) [11], which unlike traditional regression, accounts for noise in both dependent and independent variables. This is of interest to computer vision since TLS is closely related to geometric estimation techniques such as direct linear transformation [28].

As practical contributions, we investigate the true performance of distance-based guided sampling (frequently used to speed up the retrieval of all-inlier minimal subsets). In particular, we highlight the danger posed by *proximity* sampling [29,21] which actually *limits* the span of minimal subsets, and we elaborate why sampling *based on distances alone* is not a generally reliable strategy for generating minimal subsets with large spans. Based on these insights, we propose a novel sampling strategy which, unlike previous methods, consciously targets all-inlier minimal subsets with large spans. The effectiveness of our algorithm is benchmarked against state-of-the-art methods [29,5,21,3,36,1,2].

A promising alternative to fitting on minimal subsets is to sample and fit on large data *clusters* instead. This approach was investigated in our prior work [31], which involves building a sparse adjacency graph on the data. Bond variables are attached to all the graph edges and then sampled to produce data clusters with large spans, thus alleviating the problem exemplified in Fig. 1. However the method imposes the assumption of spatial smoothness, i.e., inliers must form a single coherent structure. While this is satisfied in some applications (e.g., fitting multiple motion models from rigid objects [31]) it is clearly not the general case (e.g., a single line may consist of multiple distant line segments). Our aim here is to conduct a more fundamental study into parameter estimation using minimal subsets.

It is also vital to avoid degeneracies in two-view geometry estimation [3,9]. Some scenes may contain a dominant plane on which most of the keypoints lie. Degenerate parameters occur when they are estimated from a minimal subset containing wholly keypoint matches from the dominant plane. Previous works have tackled this issue by detecting when a degenerate estimate is produced, then recovering from the degenerate estimates using the plane-and-parrallax method [3,9]. Theoretically, degenerate minimal subsets correspond to data with very small spans (insufficient degrees of freedom). Since our new sampling algorithm directly targets minimal subsets with large spans, it *actively* prevents degeneracies; see results in Sec. 5.3. Our sampling method can also be complemented with the degeneracy detection and recovery techniques of [3,9].

The rest of the paper is as follows: Sec. 2 introduces the minimal subset expansion for least squares regression. We describe TLS in Sec. 3 and develop an equivalent minimal subset expansion. Sec. 4 explores the connection our result with geometric estimation in computer vision. In Sec. 5 we propose a novel sampling scheme for all-inlier minimal subsets with large spans, and compare it against state-of-the-art approaches. We then conclude in Sec. 6.

2 Minimal Subset Expansion for Least Squares

The problem of linear regression involves deducing an unknown parameter vector $\boldsymbol{\beta} \in \mathbb{R}^m$ such that the multiplication of $\boldsymbol{\beta}$ by design matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ yields the observation vector $\mathbf{y} \in \mathbb{R}^n$. Given an overdetermined system ($n > m$), the least squares approach solves for $\boldsymbol{\beta}$ as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \quad \text{s.t.} \quad \mathbf{X}\boldsymbol{\beta} = \hat{\mathbf{y}} \tag{1}$$

where $\hat{\mathbf{y}}$ is the corrected version of $\mathbf{y}$. The solution can be obtained in closed form by calculating

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{2}$$

The problem in (1) is often called *ordinary least squares* (OLS) to discern it from other types of estimation problems. A geometrical interpretation of (1) is that $\mathbf{X}\hat{\boldsymbol{\beta}}$ is the orthogonal projection of $\mathbf{y}$ onto the column span of $\mathbf{X}$.

It turns out that $\hat{\boldsymbol{\beta}}$ can be expanded as a linear combination of minimal subset estimates. To derive this, first we apply Cramer's rule on (2) to write the j-th value of $\hat{\boldsymbol{\beta}}$ as

$$\hat{\boldsymbol{\beta}}_j = \frac{|(\mathbf{X}^T\mathbf{X})_j|}{|\mathbf{X}^T\mathbf{X}|} = \frac{|\mathbf{X}^T\mathbf{X}_j|}{|\mathbf{X}^T\mathbf{X}|} \tag{3}$$

where $|\cdot|$ calculates the determinant of a matrix. We define $(\mathbf{X}^T\mathbf{X})_j$ as $\mathbf{X}^T\mathbf{X}$ with its j-th column replaced by $\mathbf{X}^T\mathbf{y}$, and $\mathbf{X}_j$ as $\mathbf{X}$ with its j-th column replaced by $\mathbf{y}$. Via the Binet-Cauchy formula, we can expand (3) as

$$\hat{\boldsymbol{\beta}}_j = \frac{\sum_\lambda |\mathbf{X}(\lambda)||\mathbf{X}_j(\lambda)|}{\sum_\lambda |\mathbf{X}(\lambda)||\mathbf{X}(\lambda)|} \tag{4}$$

where λ indicates a combination of m integers from the set $\{1,\ldots,n\}$, and $\mathbf{X}(\lambda)$ and $\mathbf{X}_j(\lambda)$ are square matrices formed by the m rows of $\mathbf{X}$ and $\mathbf{X}_j$ indexed by λ. The summations in (4) are taken over all $\binom{n}{m}$ possibilities of λ.

Picking the rows of $\mathbf{X}$ and $\mathbf{y}$ according to a λ amounts to choosing a minimal subset, since m cases are sufficient to uniquely determine $\boldsymbol{\beta}$. The minimal estimate from λ is

$$\hat{\boldsymbol{\beta}}^{(\lambda)} = \mathbf{X}(\lambda)^{-1}\mathbf{y}(\lambda) \tag{5}$$

where $\mathbf{y}(\lambda)$ is the vector formed by the m rows of $\mathbf{y}$ indexed by λ. Via Cramer's rule again, the j-th value of $\hat{\boldsymbol{\beta}}^{(\lambda)}$ is

$$\hat{\boldsymbol{\beta}}_j^{(\lambda)} = \frac{|\mathbf{X}_j(\lambda)|}{|\mathbf{X}(\lambda)|}. \tag{6}$$

By substituting $|\mathbf{X}_j(\lambda)| = |\mathbf{X}(\lambda)|\ \hat{\boldsymbol{\beta}}_j^{(\lambda)}$ in (4) we obtain

$$\hat{\boldsymbol{\beta}}_j = \frac{\sum_\lambda |\mathbf{X}(\lambda)||\mathbf{X}(\lambda)|\hat{\boldsymbol{\beta}}_j^{(\lambda)}}{\sum_\lambda |\mathbf{X}(\lambda)||\mathbf{X}(\lambda)|} \tag{7}$$

or in vectorial form for the full parameter vector

$$\boxed{\hat{\boldsymbol{\beta}} = \sum_\lambda w_\lambda \hat{\boldsymbol{\beta}}^{(\lambda)} \quad w_\lambda = \frac{|\mathbf{X}(\lambda)|^2}{\sum_\lambda |\mathbf{X}(\lambda)|^2}} \tag{8}$$

where $0 \le w_\lambda \le 1$ and $\sum_\lambda w_\lambda = 1$. The quantity w_λ is the *weight* or *importance* of minimal subset λ towards estimating $\hat{\boldsymbol{\beta}}$. This little known result is due to Jacobi [18], and later rediscovered by others, e.g., [35].

Jacobi's result provides an algebraic justification to the intuition of maximising the span of minimal subsets. To illustrate, consider 2D linear regression again where we have

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \tag{9}$$

and $\boldsymbol{\beta}$ contains the usual parameters of intercept and slope. The weight of the estimate corresponding to a minimal subset $\lambda = \{s_1, s_2\}$ is proportional to

$$|\mathbf{X}(\lambda)|^2 = \left|\begin{bmatrix} 1 & x_{s_1} \\ 1 & x_{s_2} \end{bmatrix}\right|^2 = (x_{s_1} - x_{s_2})^2 \tag{10}$$

i.e., widely separated points provide better line estimates. More generally, $|\mathbf{X}(\lambda)|$ is the hypervolume of the parallelotope whose vertices are the rows of $\mathbf{X}(\lambda)$, a quantity that is closely related to the relative span of the data indexed by λ.

2.1 Generalising to non-minimal subsets

Can the weighted expansion in (8) be generalised to using subsets of size greater than m? Let ν index a subset of size $m+i \le n$, and $\mathbf{X}(\nu)$ be the (non-square) submatrix of $\mathbf{X}$ containing the $m+i$ rows selected according to ν. The OLS estimate can actually also be expanded as

$$\boxed{\hat{\boldsymbol{\beta}} = \sum_\nu w_\nu \hat{\boldsymbol{\beta}}^{(\nu)} \quad w_\nu = \frac{|\mathbf{X}(\nu)^T\mathbf{X}(\nu)|}{\sum_\nu |\mathbf{X}(\nu)^T\mathbf{X}(\nu)|}} \tag{11}$$

where the summation is over all $\binom{n}{m+i}$ choices of ν; we refer the reader to [17] for the proof. Observe that (8) is a special case of (11). To gain a geometrical understanding of the weights, applying the Binet-Cauchy formula again yields

$$w_\nu \propto |\mathbf{X}(\nu)^T\mathbf{X}(\nu)| = \sum_\lambda |\mathbf{X}(\lambda|\nu)|^2 \tag{12}$$

where λ indexes over all $\binom{m+i}{m}$ *minimal* subsets of $\mathbf{X}(\nu)$, and we define $\mathbf{X}(\lambda|\nu)$ as the submatrix of $\mathbf{X}(\nu)$ indexed by λ. In other words, the weight of $\mathbf{X}(\nu)$ is proportional to the sum of the weights of all the minimal subsets from $\mathbf{X}(\nu)$.

3 The Case of Total Least Squares

Here we develop the main theoretical result of this paper — a minimal subset expansion for TLS. In contrast to OLS, TLS (also called *errors-in-variables* modelling [25] or *orthogonal regression* [38]) corrects for errors in both the independent and dependent measurements $(\mathbf{X}, \mathbf{y})$. The TLS estimate is

$$\breve{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}, \breve{\mathbf{X}}} \left\| [\mathbf{X}\ \mathbf{y}] - [\breve{\mathbf{X}}\ \breve{\mathbf{y}}] \right\|_F^2 \quad \text{s.t.} \quad \breve{\mathbf{X}}\boldsymbol{\beta} = \breve{\mathbf{y}} \tag{13}$$

where we use the breve accent ($\breve{}$) to distinguish TLS results from those of OLS. The solution for (13) can be reasoned as follows [38]: Assume $[\mathbf{X}\ \mathbf{y}] \in \mathbb{R}^{n\times(m+1)}$ to be full rank, i.e., rank $m+1$ since $n > m$. To make the following system

$$[\mathbf{X}\ \mathbf{y}] \begin{bmatrix} \boldsymbol{\beta} \\ -1 \end{bmatrix} \approx \mathbf{0} \tag{14}$$

compatible, we must reduce the rank of $[\mathbf{X}\ \mathbf{y}]$ by one. Let

$$[\mathbf{X}\ \mathbf{y}] = \mathbf{A}\boldsymbol{\Sigma}\mathbf{B}^T \tag{15}$$

be the SVD of $[\mathbf{X}\ \mathbf{y}]$. From the Eckart-Young Theorem the closest rank-m matrix to $[\mathbf{X}\ \mathbf{y}]$ in the Frobenius sense is

$$[\breve{\mathbf{X}}\ \breve{\mathbf{y}}] = \mathbf{A}\breve{\boldsymbol{\Sigma}}\mathbf{B}^T \tag{16}$$

where $\breve{\boldsymbol{\Sigma}}$ is obtained by setting the $(m+1)$-th singular value σ_{m+1} in $\boldsymbol{\Sigma}$ to 0. Let $\mathbf{b}_{m+1}$ be the $(m+1)$-th right singular vector of $[\mathbf{X}\ \mathbf{y}]$. Then the optimal approximation to (14) is

$$[\breve{\mathbf{X}}\ \breve{\mathbf{y}}]\mathbf{b}_{m+1} = \mathbf{0} \tag{17}$$

which holds for arbitrary scalings of $\mathbf{b}_{m+1}$. The TLS estimate $\breve{\boldsymbol{\beta}}$ is obtained by scaling $\mathbf{b}_{m+1}$ as

$$\begin{bmatrix} \breve{\boldsymbol{\beta}} \\ -1 \end{bmatrix} = -\frac{1}{\mathbf{b}_{m+1}(m+1)}\mathbf{b}_{m+1} \tag{18}$$

such that the $(m+1)$-th element of the vector is -1. Here, $\mathbf{b}_{m+1}(m+1)$ denotes the $(m+1)$-th element of $\mathbf{b}_{m+1}$.

Clearly, for $\breve{\boldsymbol{\beta}}$ to exist the element $\mathbf{b}_{m+1}(m+1)$ cannot be zero. A sufficient condition for this is for the m-th singular value s_m of $\mathbf{X}$ to be strictly greater than σ_{m+1}:

$$s_m > \sigma_{m+1} \Longrightarrow \mathbf{b}_{m+1}(m+1) \neq 0 \text{ and } \sigma_m > \sigma_{m+1}. \tag{19}$$

See [38, Chapter 2] for the proof. As argued in [37], this condition is not restrictive and is usually satisfied. Moreover, as we will explain in Sec. 4, convenient pre- and postprocessing are available to prevent the condition from becoming debilitating for geometric estimation in computer vision.

3.1 Minimal subset expansion for TLS

Following the above derivations, the eigenvector identity

$$[\mathbf{X}\ \mathbf{y}]^T[\mathbf{X}\ \mathbf{y}]\mathbf{b}_{m+1} = \sigma_{m+1}^2\mathbf{b}_{m+1} \tag{20}$$

holds for $\mathbf{b}_{m+1}$ and arbitrary scalings of $\mathbf{b}_{m+1}$. We can thus rewrite and expand the eigenvector identity as

$$\begin{bmatrix} \mathbf{X}^T\mathbf{X} & \mathbf{X}^T\mathbf{y} \\ \mathbf{y}^T\mathbf{X} & \mathbf{y}^T\mathbf{y} \end{bmatrix} \begin{bmatrix} \breve{\boldsymbol{\beta}} \\ -1 \end{bmatrix} = \sigma_{m+1}^2 \begin{bmatrix} \breve{\boldsymbol{\beta}} \\ -1 \end{bmatrix}. \tag{21}$$

Multiplying through the top part and rearranging yields

$$\breve{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X} - \sigma_{m+1}^2\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}. \tag{22}$$

See also [38, Chapter 2] for the derivation of (22). To develop our TLS minimal subset expansion we define

$$\mathbf{Z} := \mathbf{X} - \sigma_{m+1}^2(\mathbf{X}^T)^\dagger \tag{23}$$

where $(\mathbf{X}^T)^\dagger$ is the Moore-Penrose generalised inverse of $\mathbf{X}^T$, and rewrite (22) as

$$\breve{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{Z})^{-1}\mathbf{X}^T\mathbf{y}. \tag{24}$$

We first prove the following intermediate result which appears to be novel and is of interest in its own right.

Proposition 1 *The solution to the following OLS problem*

$$\arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \quad \text{s.t.} \quad \mathbf{Z}\boldsymbol{\beta} = \hat{\mathbf{y}} \tag{25}$$

coincides with the TLS estimate $\breve{\boldsymbol{\beta}}$.

Proof If $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ is the SVD of $\mathbf{X} \in \mathbb{R}^{n\times m}$, then

$$(\mathbf{X}^T)^\dagger = \mathbf{U}\mathbf{S}^{-1}\mathbf{V}^T \tag{26}$$

is the SVD of $(\mathbf{X}^T)^\dagger$, where we define $\mathbf{S}^{-1}$ as taking the reciprocal of the diagonal elements of $\mathbf{S}$, while leaving the other elements unchanged. From (23) we can rewrite $\mathbf{Z}$ as

$$\mathbf{Z} = \mathbf{U}(\mathbf{S} - \sigma_{m+1}^2\mathbf{S}^{-1})\mathbf{V}^T := \mathbf{U}\tilde{\mathbf{S}}\mathbf{V}^T \tag{27}$$

where $\mathbf{U}\tilde{\mathbf{S}}\mathbf{V}^T$ is a valid SVD of $\mathbf{Z}$ since the diagonal values in $\tilde{\mathbf{S}}$ are still in descending order; recall that we impose condition (19) such that σ_{m+1} be strictly smaller than the smallest singular value of $\mathbf{X}$. Therefore, since $\mathbf{X}$, $(\mathbf{X}^T)^\dagger$ and $\mathbf{Z}$ share the same left singular vectors $\mathbf{U}$,

$$\mathcal{R}(\mathbf{X}) = \mathcal{R}((\mathbf{X}^T)^\dagger) = \mathcal{R}(\mathbf{Z}) \tag{28}$$

i.e., the column spans of the three matrices are equal.

If $\tilde{\boldsymbol{\beta}}$ is the solution of (25), then by invoking the closed form expression for the OLS estimate

$$\tilde{\boldsymbol{\beta}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} \tag{29}$$

which can be further manipulated to yield

$$\mathbf{X}^T\mathbf{Z}\tilde{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} + \sigma_{m+1}^2(\mathbf{X}^T)^{\dagger T}(\mathbf{Z}\tilde{\boldsymbol{\beta}} - \mathbf{y}). \tag{30}$$

Since $\tilde{\boldsymbol{\beta}}$ is the solution to the OLS problem (25), $\mathbf{Z}\tilde{\boldsymbol{\beta}}$ is the projection of $\mathbf{y}$ onto $\mathcal{R}(\mathbf{Z})$, and consequently $\mathbf{Z}\tilde{\boldsymbol{\beta}} - \mathbf{y}$ is orthogonal to $\mathcal{R}(\mathbf{Z})$. Hence, the equality (30) reduces to

$$\mathbf{X}^T\mathbf{Z}\tilde{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} \tag{31}$$

since $\mathbf{Z}\tilde{\boldsymbol{\beta}} - \mathbf{y}$ will also be orthogonal to $\mathcal{R}((\mathbf{X}^T)^\dagger)$. Comparing (24) with (31) yields the result $\breve{\boldsymbol{\beta}} = \tilde{\boldsymbol{\beta}}$. □

Proposition 1 states that, given the measurements $[\mathbf{X}\ \mathbf{y}]$, the TLS estimate can be calculated as

$$\breve{\boldsymbol{\beta}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} \tag{32}$$

where $\mathbf{Z}$ is as defined in (23). In other words, we perturb $\mathbf{X}$ by the amount $-\sigma_{m+1}^2(\mathbf{X}^T)^\dagger$ to become $\mathbf{Z}$, while leaving $\mathbf{y}$ unchanged, such that TLS on $(\mathbf{X}, \mathbf{y})$ can be solved as OLS on $(\mathbf{Z}, \mathbf{y})$. Fig. 2 illustrates this idea on 2D line fitting.

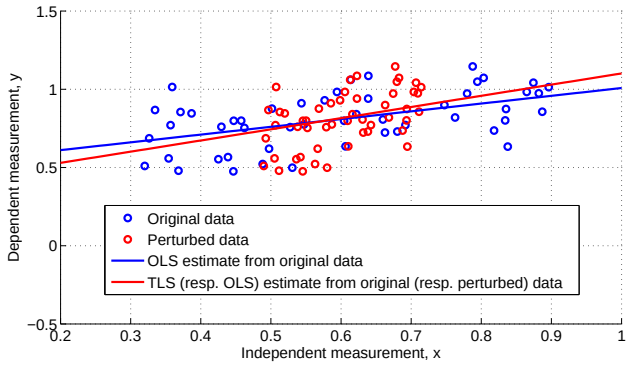
To exploit Proposition 1, we define

$$\breve{\boldsymbol{\beta}}^{(\lambda)} := \mathbf{Z}(\lambda)^{-1}\mathbf{y}(\lambda) \tag{33}$$

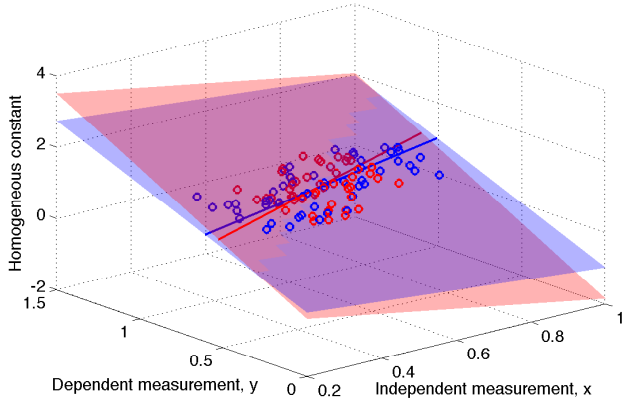
as the minimal subset estimate for $\breve{\boldsymbol{\beta}}$ based on the m rows of $\mathbf{Z}$ and $\mathbf{y}$ as indexed by λ. Applying the steps taken in Sec. 2 on (32), we obtain a minimal subset expansion for $\breve{\boldsymbol{\beta}}$

$$\boxed{\breve{\boldsymbol{\beta}} = \sum_\lambda w_\lambda \breve{\boldsymbol{\beta}}^{(\lambda)} \quad w_\lambda = \frac{|\mathbf{Z}(\lambda)|^2}{\sum_\lambda |\mathbf{Z}(\lambda)|^2}} \tag{34}$$

where again w_λ is the weight of minimal subset λ, with $0 \leq w_\lambda \leq 1$ and $\sum_\lambda w_\lambda = 1$. Extending (34) to using non-minimal subsets also follows easily from (11).



(a) In 2D line fitting, the measurements $\{x_i, y_i\}_{i=1}^n$ are collected into matrix $\mathbf{X} \in \mathbb{R}^{n\times 2}$ and vector $\mathbf{y} \in \mathbb{R}^n$ as defined in (9), where the first column of $\mathbf{X}$ contains the homogeneous constants 1. Following (23), $\mathbf{X}$ is perturbed to become $\mathbf{Z}$. From Proposition 1 the TLS estimate on $(\mathbf{X}, \mathbf{y})$ is the same as the OLS estimate on $(\mathbf{Z}, \mathbf{y})$. The data after perturbation are plotted in red — note that the homogeneous constants in $\mathbf{Z}$ would also have been perturbed away from 1, which mirrors the fact that TLS (13) will correct all columns in $\mathbf{X}$. To plot $(\mathbf{Z}, \mathbf{y})$ here, we "dehomogenise" each datum by dividing the measurements with the corresponding homogeneous constant. See also panel (b) below.



(b) Line fitting on 2D data $\{x_i, y_i\}_{i=1}^n$ is really accomplished as 2D subspace fitting on 3D data $(\mathbf{X}, \mathbf{y})$, where points in the subspace satisfies $[c\ x]\boldsymbol{\beta} - y = 0$, and c is an auxiliary variable corresponding to the homogeneous constant. Proposition 1 states that TLS on $(\mathbf{X}, \mathbf{y})$ is the same as OLS on $(\mathbf{Z}, \mathbf{y})$, where $\mathbf{Z}$ is perturbed from $\mathbf{X}$ according to (23). Here, we plot $(\mathbf{X}, \mathbf{y})$ and $(\mathbf{Z}, \mathbf{y})$ in $\mathbb{R}^3$, where the homogeneous constants in $\mathbf{X}$ and $\mathbf{Z}$ are plotted in the vertical axis. The 2D subspaces fitted on $(\mathbf{X}, \mathbf{y})$ and $(\mathbf{Z}, \mathbf{y})$ by OLS are also plotted. The lines fitted on $\{x_i, y_i\}_{i=1}^n$ are 1D *affine* subspaces within the 2D subspaces. Panel (a) is the projection of (b) onto the plane $c = 1$.

Fig. 2 Illustrating Proposition 1 on the problem of line fitting in 2D.

3.2 Comparing weights of minimal subsets under TLS

In many cases (e.g., in robust estimation) we are mainly interested in comparing the goodness (from the aspect of span) of two minimal subsets and not the actual weight values. This motivates the following result on weight comparison.

Proposition 2 *Given two minimal subsets λ_1 and λ_2,*

$$|\mathbf{X}(\lambda_1)|^2 > |\mathbf{X}(\lambda_2)|^2 \implies |\mathbf{Z}(\lambda_1)|^2 > |\mathbf{Z}(\lambda_2)|^2. \tag{35}$$

Proof Recall the SVD of $\mathbf{X}$ and $\mathbf{Z}$ in Proposition 1. Since both $\mathbf{X}$ and $\mathbf{Z}$ are of size $n \times m$ with $n > m$, they can be expanded using the first-m left singular vectors $\mathbf{U}_m$ of $\mathbf{X}$

$$\mathbf{X} = \mathbf{U}_m\mathbf{S}_m\mathbf{V}^T \quad \mathbf{Z} = \mathbf{U}_m\tilde{\mathbf{S}}_m\mathbf{V}^T \tag{36}$$

where $\mathbf{S}_m$ is the first $m \times m$ submatrix of $\mathbf{S}$ (similarly for $\tilde{\mathbf{S}}$). The determinants $|\mathbf{X}(\nu)|$ and $|\mathbf{Z}(\nu)|$ can be obtained as

$$|\mathbf{X}(\lambda)| = |\mathbf{U}_m(\lambda)||\mathbf{S}_m\mathbf{V}^T| \quad |\mathbf{Z}(\lambda)| = |\mathbf{U}_m(\lambda)||\tilde{\mathbf{S}}_m\mathbf{V}^T|$$

where we define $\mathbf{U}_m(\lambda)$ as the m rows of $\mathbf{U}_m$ selected according to λ. It is clear that

$$|\mathbf{X}(\lambda)| = \alpha|\mathbf{Z}(\lambda)| \quad \alpha := |\mathbf{S}_m\mathbf{V}^T|/|\tilde{\mathbf{S}}_m\mathbf{V}^T| \tag{37}$$

where α is a constant independent of λ. Therefore, given two minimal subsets λ_1 and λ_2, if $|\mathbf{X}(\lambda_1)|^2 > |\mathbf{X}(\lambda_2)|^2$ then

$$|\mathbf{X}(\lambda_1)|^2/|\mathbf{X}(\lambda_2)|^2 = |\mathbf{Z}(\lambda_1)|^2/|\mathbf{Z}(\lambda_2)|^2 > 1 \tag{38}$$

or $|\mathbf{Z}(\lambda_1)|^2 > |\mathbf{Z}(\lambda_2)|^2$. □

Proposition 2 states that the *relative* weights of two minimal subsets λ_1 and λ_2 are equivalent under TLS and OLS; it is thus sufficient to compute $|\mathbf{X}(\lambda)|^2$ and there is no need to obtain $|\mathbf{Z}(\lambda)|^2$ (which requires the singular value σ_{m+1}). In fact, Proposition 2 implies that the weight w_λ of a minimal subset λ is *equal* under (8) and (34), although the associated minimal estimates $\hat{\boldsymbol{\beta}}^{(\lambda)}$ and $\breve{\boldsymbol{\beta}}^{(\lambda)}$ may be different.

Extending Proposition 2 to compare the weights of non-minimal subsets is straightforward; refer to the Appendix.

3.3 TLS with frozen columns

In some applications it is useful to constrain the TLS correction to occur on some of the columns of $\mathbf{X}$ while leaving the other *known* columns unchanged or "frozen" [10] (in Sec. 4 we elaborate why this is relevant for geometric estimation). The problem is also known as *mixed OLS-TLS* [38]. Here we show that our result extends easily to this case.

Let $\mathbf{X} = [\mathbf{X}_1\ \mathbf{X}_2]$ be rearranged such that $\mathbf{X}_1 \in \mathbb{R}^{n\times m_1}$ are the frozen columns while $\mathbf{X}_2 \in \mathbb{R}^{n\times m_2}$ are to be corrected, and $m = m_1 + m_2$. The task is to estimate

$$\breve{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}, \breve{\mathbf{X}}_2} \left\| [\mathbf{X}_2\ \mathbf{y}] - [\breve{\mathbf{X}}_2\ \breve{\mathbf{y}}] \right\|_F^2 \quad \text{s.t.} \quad [\mathbf{X}_1\ \breve{\mathbf{X}}_2]\boldsymbol{\beta} = \breve{\mathbf{y}}.$$

We first perform the QR factorisation

$$[\mathbf{X}_1\ \mathbf{X}_2\ \mathbf{y}] = \mathbf{QR} \quad \text{with} \quad \mathbf{R} = \begin{array}{c} \\ m_1 \\ n-m_1 \end{array} \begin{array}{c} \begin{array}{ccc} m_1 & m_2 & 1 \end{array} \\ \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{r}_1 \\ \mathbf{0} & \mathbf{R}_{22} & \mathbf{r}_2 \end{bmatrix} \end{array}$$

where $\mathbf{R}_{11}$ is an upper triangular matrix. Basic TLS is invoked on $\mathbf{R}_{22}$ and $\mathbf{r}_2$ to solve for the last-m_2 elements of $\breve{\boldsymbol{\beta}}$.

These are then substituted back into the system to allow the first-m_1 variables to be obtained using OLS.

Let σ_{m_2+1} be the smallest singular value of $[\mathbf{R}_{22}\ \mathbf{r}_2]$. The mixed OLS-TLS estimate can be expressed as [38]

$$\breve{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X} - \sigma^2_{m_2+1}\mathbf{L}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{39}$$

where $\mathbf{L}$ is a "selector matrix" defined as

$$\mathbf{L} := \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}_{m_2}\end{bmatrix} \in \mathbb{R}^{m\times m} \tag{40}$$

with $\mathbf{I}_{m_2}$ the $m_2 \times m_2$ identity matrix. In effect, $\mathbf{L}$ chooses which columns of $\mathbf{X}$ get corrected. Defining $\mathbf{Z}$ now as

$$\mathbf{Z} := \mathbf{X} - \sigma^2_{m_2+1}(\mathbf{X}^T)^\dagger\mathbf{L} \tag{41}$$

we may also re-express (39) like (24) as

$$\breve{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{Z}\right)^{-1}\mathbf{X}^T\mathbf{y}. \tag{42}$$

Observe now that in (41) $\mathbf{L}$ chooses the columns of $\mathbf{X}$ that are perturbed. It turns out that Propositions 1 and 2 also hold for mixed OLS-TLS (see Appendix for proof), i.e., $\breve{\boldsymbol{\beta}}$ can be obtained as the solution of the OLS in (25), which motivates the expansion (34) for mixed OLS-TLS with minimal or non-minimal subsets.

3.4 Orthogonal distance fitting

We explore the consequence of our result on orthogonal distance fitting of lines onto 2D data. Given data $\{x_i, y_i\}_{i=1}^n$, we wish to estimate the $\boldsymbol{\beta} = [\beta_1\ \beta_2]^T \in \mathbb{R}^2$ that minimises

$$\sum_{i=1}^{n}\frac{(\beta_1 + \beta_2 x_i - y_i)^2}{\beta_2^2 + 1} \tag{43}$$

i.e., the sum of squared *orthogonal distances* to the line $\boldsymbol{\beta}$. Creating matrix $\mathbf{X}$ and vector $\mathbf{y}$ as in (9), the problem is equivalent to minimising the generalised Rayleigh quotient

$$\arg\min_{\theta}\frac{\theta^T[\mathbf{X}\ \mathbf{y}]^T[\mathbf{X}\ \mathbf{y}]\theta}{\theta^T\begin{bmatrix}\mathbf{L} & 0\\ 0 & 1\end{bmatrix}\theta} \tag{44}$$

where $\theta = [\boldsymbol{\beta}^T\ \ -1]^T$ and $\mathbf{L}$ is as defined in (40) with $m = 2$ and $m_2 = 1$. The solution $\breve{\theta} = [\breve{\boldsymbol{\beta}}^T\ \ -1]^T$ satisfies the generalised eigenvector equation

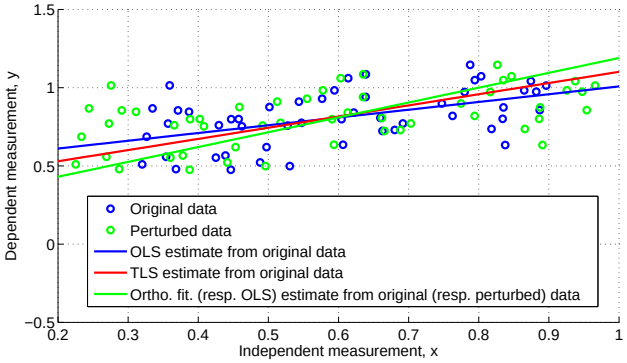
$$[\mathbf{X}\ \mathbf{y}]^T[\mathbf{X}\ \mathbf{y}]\begin{bmatrix}\breve{\boldsymbol{\beta}}\\ -1\end{bmatrix} = \eta\begin{bmatrix}\mathbf{L} & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\breve{\boldsymbol{\beta}}\\ -1\end{bmatrix} \tag{45}$$

with η the smallest generalised eigenvalue. Multiplying through the top part and rearranging yields

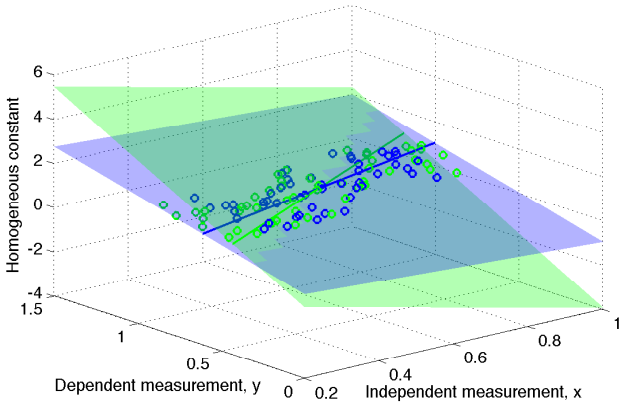
$$\breve{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X} - \eta\mathbf{L})^{-1}\mathbf{X}^T\mathbf{y} \tag{46}$$

which resembles (39). Identifying the first column in $\mathbf{X}$ as the frozen column (the homogeneous constants in $\mathbf{X}$ are prevented from being corrected), it turns out that orthogonal distance fitting is an instance of mixed OLS-TLS.

Fig. 3 illustrates the implications of Proposition 1 on orthogonal distance line fitting, using the same data as in Fig. 2. Here $\mathbf{Z}$ is perturbed from $\mathbf{X}$ following (41) which preserves the values of the homogeneous constants.



(a) Comparing estimates from OLS, TLS and orthogonal line fitting. Here, $\mathbf{Z}$ is perturbed from $\mathbf{X}$ following (41) which preserves the homogeneous constants at 1. Therefore, dehomogenisation of $(\mathbf{Z}, \mathbf{y})$ is not required to plot the perturbed data here, unlike in Fig. 2(a).



(b) Here $(\mathbf{X}, \mathbf{y})$ and $(\mathbf{Z}, \mathbf{y})$ are plotted in $\mathbb{R}^3$, where the homogeneous constants in $\mathbf{X}$ and $\mathbf{Z}$ are plotted in the vertical axis. The 2D subspaces fitted on $(\mathbf{X}, \mathbf{y})$ and $(\mathbf{Z}, \mathbf{y})$ by OLS are also plotted. Unlike Fig. 2(b), where under standard TLS the homogeneous constants in $\mathbf{X}$ are not preserved in $\mathbf{Z}$, here the homogeneous constants remain at 1.

Fig. 3 Illustrating Proposition 1 on orthogonal distance line fitting.

It is worthwhile to clarify why the literature also calls standard TLS defined in (13) "orthogonal regression" [38]. If there are no homogeneous constants in $\mathbf{X}$ (or equivalently no intercept parameter in $\boldsymbol{\beta}$), (13) estimates the *non-affine* subspace which minimises the sum of squared orthogonal distances to the points. If the linear relation involves an intercept, the solution to (13) no longer minimises the sum of squared orthogonal distances. See [13] for details.

4 TLS in Geometric Estimation Problems

In this section we establish the relevance of our result to geometric estimation problems, in particular to the technique of Direct Linear Transformation (DLT) [41]. The intimate connection between TLS and DLT has been established elsewhere, e.g., [28,23]. Here we restate the equivalence in the context of multiple view geometry [16], specifically in the estimation of fundamental matrices.

A pair of matching points (p, q) and (p', q') arising from a scene imaged in two views satisfies the constraint

$$[p'\ q'\ 1]\mathbf{F}[p\ q\ 1]^T = 0 \tag{47}$$

where $\mathbf{F}$ is a 3×3 matrix called the fundamental matrix. The constraint is linearised by multiplying through to yield

$$[p'p \quad p'q \quad p' \quad q'p \quad q'q \quad q' \quad p \quad q \quad 1]\,\mathbf{f} = 0 \tag{48}$$

where $\mathbf{f}$ is a column vector containing the nine elements of $\mathbf{F}$. Given a set of noisy point matches $\{(p_i, q_i), (p'_i, q'_i)\}_{i=1}^n$, the task is to estimate the $\mathbf{F}$ of $\mathbf{f}$ corresponding to the scene.

DLT computes (48) for each match $\{(p_i, q_i), (p'_i, q'_i)\}$, and stacks the rows to yield a matrix $\mathbf{D} \in \mathbb{R}^{n\times 9}$. The solution is obtained as

$$\mathbf{f}^* = \arg\min_{\mathbf{f}} \|\mathbf{D}\mathbf{f}\|^2 \quad \text{s.t.} \quad \|\mathbf{f}\| = 1 \tag{49}$$

where the quadratic constraint avoids the trivial solution $\mathbf{f} = \mathbf{0}$. It can be shown that $\mathbf{f}^*$ is the least significant right singular vector of $\mathbf{D}$ [41]. Equating $\mathbf{D}$ with $[\mathbf{X}\ \mathbf{y}]$ under TLS, it is clear that $\mathbf{f}^* = \mathbf{b}_{m+1}$ in (17), and apart from the rescaling step (18) the TLS and DLT estimates are equal.

Note that since $\mathbf{f}$ is homogeneous it has only eight degrees of freedom, mirroring the fact that the TLS parameter vector $\boldsymbol{\beta}$ has eight elements in this problem. Therefore a minimal subset of eight rows of $\mathbf{D}$ or $[\mathbf{X}\ \mathbf{y}]$ corresponding to eight point matches are sufficient to instantiate $\mathbf{f}$ or $\boldsymbol{\beta}$[1].

Our derivations in Sec. 3.1, however, require that the constraint (47) be dehomogenised. This amounts to fixing an element in $\mathbf{f}$ to -1 and moving the corresponding column in $\mathbf{D}$ to the RHS of (47) to become the observation vector $\mathbf{y}$. Such a modification is prevalent in the literature, e.g., [20,19,30,4], and is usually applied to bring to bear the framework of regression onto homogeneous estimation. If the element fixed to -1 in $\mathbf{f}$ has the true value 0, the result may be numerically unstable. A frequent solution (e.g., see [20]) is to detect when this occurs and then change another element in $\mathbf{f}$ to fix at -1 (see also the handling of such *non-generic collinearities* under TLS [38]). In some cases preprocessing of the data is available to ensure that an element in the homogeneous vector is nonzero [19,4].

[1] Actually 7 matches are sufficient since $\mathbf{F}$ is rank deficient by one, but the estimation process from 7 is more complicated. In any case the rank constraint can be imposed post-estimation from 8 matches [15].

From (48) it is also clear that one column in $\mathbf{D}$ consists of the homogeneous constant 1. This column should be frozen under TLS and not be moved to the RHS to yield the observation vector $\mathbf{y}$; see [28,14]. This is relevant to our result in Sec. 3.3. In the following experiments, we dehomogenise by fixing the first element in $\mathbf{f}$ to 1.

Note that DLT (49) minimises the sum of squared *algebraic* errors. It is also common to minimise *geometric* errors in geometry estimation, such as the Sampson distance [16]. Such errors are typically nonlinear functions of the parameters, thus extending our TLS expansion (34) to accommodate such errors is nontrivial. In any case, geometric errors are usually minimised during the refinement stage (i.e., after robust estimation) using iterative methods (e.g., Levenberg-Marquardt). Since the error contributions of all data are considered simultaneously, the issue of span is not as critical.

4.1 The influence of span to fundamental matrix estimation

Many keypoint detectors used in multi-view geometry are really *2D region* detectors. Some of the more popular detectors are designed to detect affine invariant regions [27], including the SIFT method [24]. This fact has been exploited to reduce the minimum number of keypoint matches used to estimate a fundamental matrix, since two unique planar correspondences contain the sufficient degrees of freedom.

In practice, the methods generate extra matches from the keypoint matches to make up the required data for estimation. Given three keypoint matches, Chum et al. [6] generate two extra matches per keypoint match, yielding in total nine matching point coordinates. Given two keypoint matches, Goshen and Shimshoni [12] generate three extra matches per keypoint match, producing eight matching point coordinates. Henceforth we call the two methods *2-* and *3-point* to highlight the number of unique keypoint matches used. The premise is that under a random sampling regime for robust estimation, only two or three inliers (correct keypoint matches) need to be sampled to get a good estimate.

To generate the extra point matches, the methods exploit the scale and orientation information output by keypoint detectors. The extra points are generated within the region (of size proportional to the scale) of the detected keypoint; see [6,12] for details. However, since the keypoint regions are usually small, the 2- and 3-point methods effectively limit the span of the data used for estimation. Here, we compare these estimation methods against those that actually use 8 or 9 unique keypoint matches (which we respectively call 8- and 9-point methods in the following).

On the *Barr-Smith* image pair in Fig. 12, we detect and match SIFT keypoints [24] using the VLFeat [39] toolbox and manually identify the correct matches (there are 75). For numerical stability, the matches are normalised such that

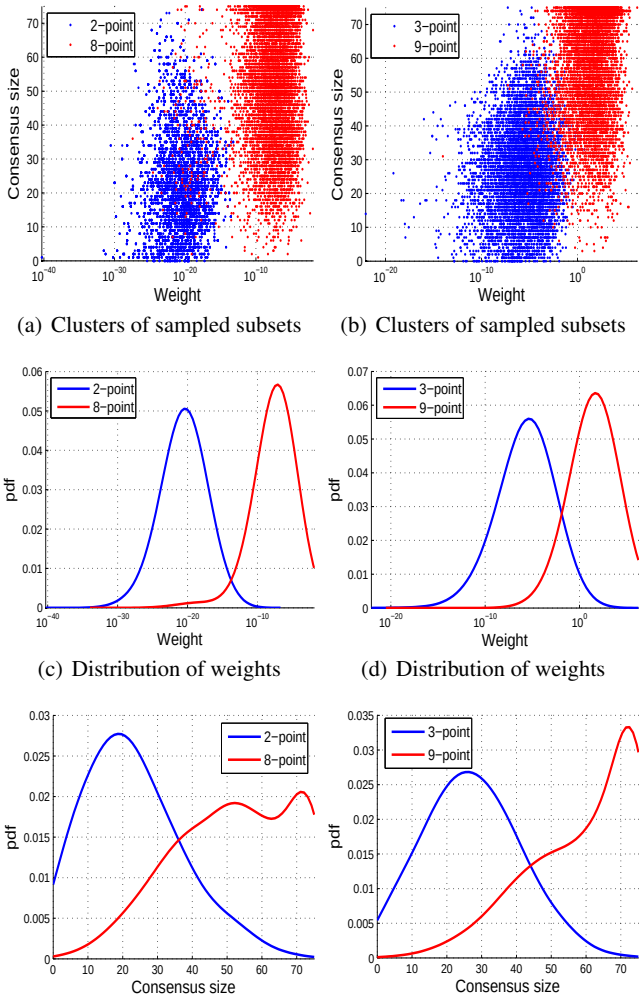
they are centred at the origin and the mean distance to the origin is $\sqrt{2}$ [15]. We randomly draw 10,000 samples of sizes 2, 3, 8 and 9 over the set of 75 inliers. For each sample, a fundamental matrix is estimated using DLT, where extra point matches are produced for the 2- and 3-point methods following [6,12]. We calculate the consensus size (for a tuned inlier threshold) and weight for each sample, based on the following rules:

- For the 8- and 9-point methods, we simply compute the weight as $|\mathbf{X}(\nu)^T\mathbf{X}(\nu)|$, where ν indexes a subset of 8 or 9 rows from the data matrix $\mathbf{X} \in \mathbb{R}^{n\times 8}$.
- For the 2- and 3-point methods, we calculate the weight as $|\bar{\mathbf{X}}^T\bar{\mathbf{X}}|$, where we define $\bar{\mathbf{X}}$ as the matrix containing the 2 or 3 rows sampled from $\mathbf{X}$ and the rows corresponding to the extra generated matches.

We plot consensus size versus weight of the samples. Fig. 4(a) contains the results of the 2- and 8-point methods, while Fig. 4(b) contains the results of the 3- and 9-point methods. We separated the plots since the subset weights calculated from using 8 and 9 rows of data have different "units". For clearer comparisons, Figs. 4(c)–(f) show the same results using kernel density estimates of weight and consensus size.

The results show that the 2- and 3-point methods do have a tendency to produce data with smaller spans. Further, there is a clear correlation between subset span and quality of estimate, indicating the danger posed by using data with small spans for estimation. Goshen and Shimshoni [12] have experimentally shown that the estimates from the 2- and 3-point methods are usually inferior to those from the 8- and 9-point methods. Our results conclusively show that this phenomenon is due in a large part to data span.

Note that another popular estimation approach is the 7-point method [16]. In theory, $\mathbf{F}$ has only 7 degrees of freedom since it is homogeneous and only rank-2, thus 7 unique keypoint matches are actually sufficient. However, the estimation involves solving nonlinear equations whereby the rank constraint is imposed *during* computation. This is unlike the 2-, 3-, 8- and 9-methods whose estimation (via DLT) is linear and where the rank constraint is imposed afterwards. Therefore, we did not test the 7-point method from the aspect of span. In any case, Goshen and Shimshoni [12] have shown that the 7-point method is usually inferior to the 8- or 9-point methods (see [12, Fig. 5]).



(a) Clusters of sampled subsets (b) Clusters of sampled subsets

(c) Distribution of weights (d) Distribution of weights

(e) Distribution of consensus sizes (f) Distribution of consensus sizes

Fig. 4 Comparison of estimation algorithms. First column: 2- and 8-point methods. Second column: 3- and 9-point methods. Note that in (a)–(d), the horizontal (weight) axis is in logarithmic scale.

5 Guided Sampling with Large Spans

We concentrate on the task of fundamental matrix estimation (see Sec. 4). In practice, automatic keypoint detection and matching algorithms invariably produce mismatches or outliers [27,26], thus robust estimators like LMedS [32] or RANSAC [8] are required. Most robust estimators sample and test model hypotheses fitted on minimal subsets. Various guided sampling methods have been proposed [29,5,21, 3,36,1,2,33] to speed up the retrieval of all-inlier minimal subsets, however none *consciously* targets all-inlier minimal subsets with large spans. We propose one such algorithm and demonstrate the gains in efficiency achievable. Note that clustering techniques [22] can also be applied to reduce the number of dimensions of the search space to speed up robust estimation, though we do not explore this idea here.

In the following we use minimal subsets of size eight to avoid limiting the span of the data; see Sec. 4.1.

5.1 Distance-based sampling

First, we study the performance of distance-based guided sampling. By this we mean algorithms that sample minimal subsets based on the distances between data points. Given

a set of point matches $\{(p_i, q_i), (p'_i, q'_i)\}_{i=1}^n$ between two views, the matches are normalised using the method of [15]. The linearisation (48) is then performed to yield data matrix $\mathbf{D}$, which we partition into $[\mathbf{X}\ \mathbf{y}] \in \mathbb{R}^{n\times 9}$; see Sec. 4. Note that each row of $[\mathbf{X}\ \mathbf{y}]$ is a "datum", which consists of dependent measurements $\mathbf{x}_i \in \mathbb{R}^8$ and an independent measurement y_i arising from the match $(p_i, q_i), (p'_i, q'_i)$.

Distance-based sampling algorithms require a notion of distance between a pair of data. For example, [29] uses the Euclidean distance of the coordinates prior to linearisation

$$\left\|[p_i\ q_i\ p'_i\ q'_i]^T - [p_j\ q_j\ p'_j\ q'_j]^T\right\| \tag{50}$$

while [21] simply use the distance in one of the views

$$\left\|[p_i\ q_i]^T - [p_j\ q_j]^T\right\|. \tag{51}$$

Here, to be consistent with our results in Sec. 3.1, we use the Euclidean distance of the dependent measurement vectors

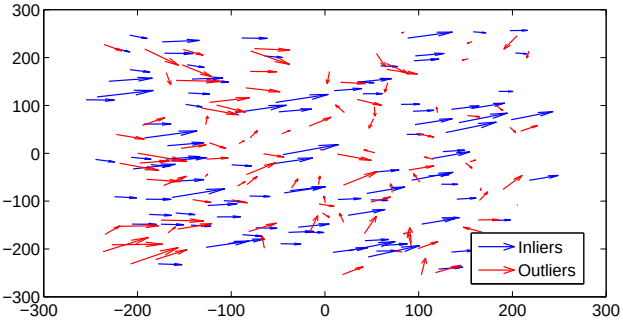
$$\|\mathbf{x}_i - \mathbf{x}_j\| \tag{52}$$

since it is the span of the rows of the design matrix $\mathbf{X}$ that matter. In any case, we have verified that using (52) gives similar results as (50) and better results than (51).

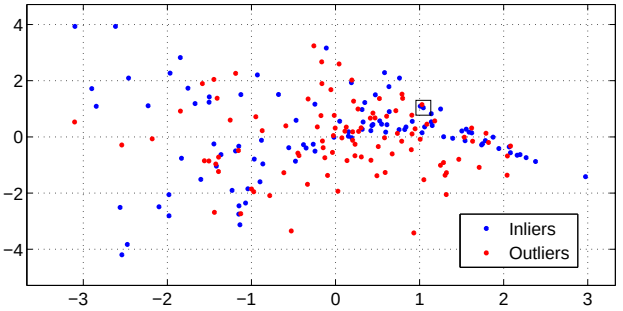
Fig. 5(a) shows 100 synthetic point matches (inliers) arising from an underlying fundamental matrix generated using Torr's SfM Toolkit[2], on top of which we add 100 uniformly sampled outliers ($n = 200$ matches in total). To visualise $\mathbf{X}$, we project it down to 2D using PCA. The results are shown in Fig. 5(b), where it can be seen that inliers tend to be situated closely. A particular inlier highlighted in Fig. 5(b) is chosen, and the set of data within distance r away from the chosen inlier are identified in Fig. 5(c), where r is 2 times the average nearest neighbour distance among the data. For the chosen inlier, within r the inlier percentage is 83.3%, which is much higher than the global inlier percentage of 50%. Therefore, by focussing the sampling on neighbouring data the chances of hitting all-inlier minimal subsets are increased. This is the premise behind *proximity sampling* [29, 21], which we outline in Algorithm 1. Fig. 6 shows the sampling weights (54) centred on the chosen inlier, where radius r is as above. It is clear that inliers have a higher probability of being sampled.

However, it is clear that proximity sampling does not encourage minimal subsets with large spans. One may use a larger r to increase the data span, but this impacts the ability to hit inliers. Fig. 8 illustrates the effects of increasing r on the number of all-inlier minimal subsets and maximum consensus (for a suitably tuned inlier threshold) achievable within 300 iterations of Algorithm 1. These are median results over 100 synthetic data generated à la Fig. 5(a). Radius

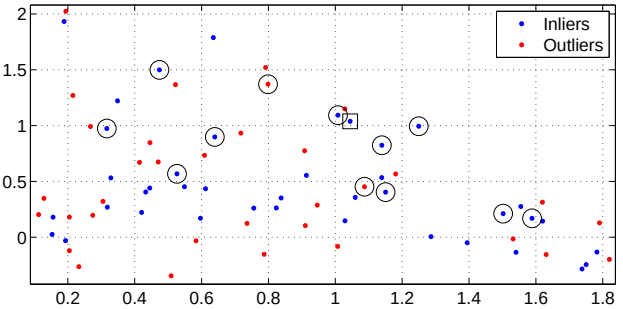
[2] http://cms.brookes.ac.uk/research/visiongroup/



(a) Synthetically generated point matches with 100 inliers arising from an underlying fundamental matrix, and 100 outliers sampled uniformly within the image domain.



(b) The data in (a) is linearised by (48) and then projected down to 2D via PCA (using the design matrix $\mathbf{X}$ only). An inlier (highlighted with a square) is chosen as the first datum for a minimal subset.



(c) Closeup of (b) showing data (circled) within r away from the chosen inlier, where r is 2 times the average nearest neighbour distance. Among the data within r, 10 out of 12 (83.3%) are inliers.

Fig. 5 Proximity sampling for fundamental matrix estimation.

r is varied by changing a multiplier k in

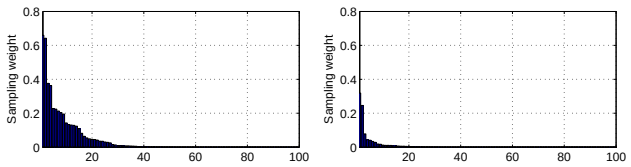
$$r = \frac{k}{n}\sum_{i=1}^{n}\min_{j\neq i}\|\mathbf{x}_i - \mathbf{x}_j\| \tag{53}$$

i.e., r is k times the average nearest neighbour distance. As k increases, the ability to find all-inlier minimal subsets decreases exponentially, without a commensurate increase in maximum consensus. Note that a good hypothesis should achieve a consensus of 100.

Instead of placing the mode of the sampling distribution on the first datum of the minimal subset, one might suggest to offset the distribution to favour data that are farther away.

Algorithm 1 Proximity sampling

Require: Design matrix $\mathbf{X} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_n]^T$, radius r.
1: $\mathbf{d}_1 \leftarrow$ a row of $\mathbf{X}$ sampled randomly.
2: **for** $j := 2, \ldots, m$ **do**
3: $\quad \mathbf{d}_j \leftarrow$ a row of $\mathbf{X}$ sampled based on the weights
$$P(\mathbf{x}_i) \sim \exp\left(-\|\mathbf{x}_i - \mathbf{d}_1\|^2 / 2r^2\right). \tag{54}$$
4: **end for**

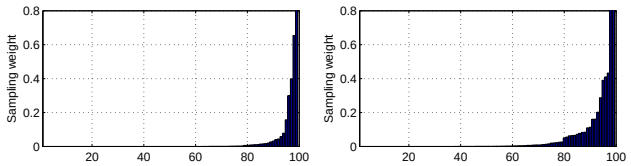


(a) Inlier weights (b) Outlier weights

Fig. 6 Weights for the data in Fig. 5(a) computed according to (54) when it is centred on the inlier chosen in Fig. 5(b). The data are first sorted in increasing distance to the chosen inlier. Radius r is 2 times the average nearest neighbour distance.

Algorithm 2 Proximity sampling with offset

Require: Design matrix $\mathbf{X} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_n]^T$, radius r, offset t.
1: $\mathbf{d}_1 \leftarrow$ a row of $\mathbf{X}$ sampled randomly.
2: **for** $j := 2, \ldots, m$ **do**
3: $\quad \mathbf{d}_j \leftarrow$ a row of $\mathbf{X}$ sampled based on the weights
$$P(\mathbf{x}_i) \sim \exp\left(-\left(\|\mathbf{x}_i - \mathbf{d}_1\| - t\right)^2 / 2r^2\right). \tag{55}$$
4: **end for**



(a) Inlier weights (b) Outlier weights

Fig. 7 Weights for the data in Fig. 5(a) computed according to (55) when it is centred on the inlier chosen in Fig. 5(b). The data are first sorted in increasing distance to the chosen inlier. Radius r is 2 times the average nearest neighbour distance, while offset t is 2 times the average pairwise distance *among all inliers*.

This method is summarised in Algorithm 2, which requires an extra offset parameter t. Ideally t should be proportional to the maximum achievable span among *all the inliers* in the data. We repeat the above experiment using Algorithm 2 for increasing k. The offset t is set as 2 times the average pairwise distance among the inlier portion of the data (this knowledge is not available in practice). Fig. 8 shows that the performance (median over 100 unique datasets) is not better than Algorithm 1. In fact the ability to sample all-inlier minimal subsets is significantly decreased.

One might suspect that the offset is not properly tuned. Fig. 7 plots the weights (55) of the data in Fig. 5(a) with the inlier chosen in Fig. 5(b) as the first datum. Again t is 2 times the average pairwise distance among all the inliers. Observe that the distribution favours the inliers which are as far away as possible, i.e., t is optimal in this case (in the general case the optimal offset is a priori unknown). However equally favoured are many outliers, rendering the distribution ineffective in sampling all-inlier minimal subsets.

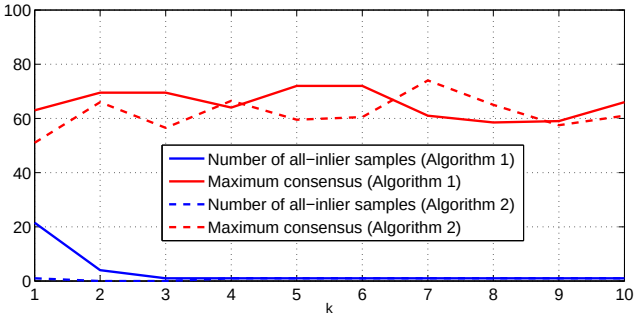


Fig. 8 Number of all-inlier minimal subsets and maximum consensus achieved within 300 iterations of Algorithm 1 as a function of k (53).

Contrary to popular intuition, the results indicate that simple distance-based sampling cannot be relied upon to retrieve minimal subsets with large spans. For Algorithm 2 to work, the inliers have to form a *single tight cluster*, a condition that cannot be guaranteed in general (cf. Algorithm 1 only requires that inliers are *locally dense*). In Sec. 5.3 we provide further results on synthetic and real data that support the findings in this section.

5.2 Combining Multi-GS with distance-based sampling

We use our recently proposed guided sampling algorithm called Multi-GS [1,2] as the basis of a new method. Instead of analysing distances, Multi-GS measures the correlation in preference among the data. Specifically, let $\{\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_M\}$ be a set of model hypotheses sampled thus far. For each datum $\mathbf{x}_i$, its residuals to the hypotheses are computed

$$\mathbf{r}^{(i)} = \begin{bmatrix} r_1^{(i)} & r_2^{(i)} & \ldots & r_M^{(i)} \end{bmatrix} \tag{56}$$

where $r_l^{(i)} = |y_i - \boldsymbol{\beta}_l \mathbf{x}_i|$. The residuals are then sorted in increasing order to yield the permutation

$$\mathbf{a}^{(i)} = \begin{bmatrix} a_1^{(i)} & a_2^{(i)} & \ldots & a_M^{(i)} \end{bmatrix} \tag{57}$$

where $r_{a_u^{(i)}}^{(i)} \leq r_{a_v^{(i)}}^{(i)}, \forall\ u < v$. The permutation $\mathbf{a}^{(i)}$ is called the preference of $\mathbf{x}_i$. The preference correlation between $\mathbf{x}_i$ and $\mathbf{x}_j$ is

$$f(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{h} |\mathbf{a}_{1:h}^{(i)} \cap \mathbf{a}_{1:h}^{(j)}| \tag{58}$$

where $\mathbf{a}_{1:h}^{(i)}$ are the first-h elements of $\mathbf{a}^{(i)}$, and $\cap$ is set intersection. The width h is typically set as $\lceil 0.1M \rceil$ [1,2]. Intuitively (58) measures the degree of overlap among the top-h most preferred hypotheses of $\mathbf{x}_i$ and $\mathbf{x}_j$.

Multi-GS (Algorithm 3) uses preference correlations as weights for guided sampling[3]. Fig. 9 shows the matrix of correlation for the data in Fig. 5(a), based on 150 previously generated fundamental matrix hypotheses. It can be seen that inliers have higher mutual preference. Fig. 10 shows the sampling weights when (58) is centred on the inlier chosen in Fig. 5(b) (these would be taken from the row corresponding to the inlier in Fig. 9). Observe that inliers in general have much higher weights than outliers. Note that this effect was achieved without having a hypothesis in $\{\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_M\}$ fitted on an all-inlier minimal subset; see [1,2] for details.

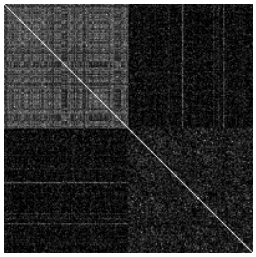


Fig. 9 Pairwise matrix of preference correlation values (58) for the data in Fig. 5(a), where the first-100 rows/columns correspond to inliers and the rest to outliers. The preferences are based on 150 fundamental matrix hypotheses.

Although it does not favour all-inlier minimal subsets with small spatial extents (observe that the inlier weights in Fig. 10 are evenly distributed), Multi-GS does not explicitly aim for large inlier spans either. This is a source of inefficiency. Moreover, the experiments in [1,2] only examined the ability of Multi-GS to sample all-inlier minimal subsets, and not the *actual quality* of the fitted hypotheses.

We propose an extension to Multi-GS to rectify this inadequacy. Our idea is to combine Multi-GS with distance-based sampling. Specifically, we multiply preference correlation with the distance-based distribution (55) to yield

$$P(\mathbf{x}_i) \sim f(\mathbf{x}_i, \mathbf{d}) \exp(-(\|\mathbf{x}_i - \mathbf{d}\| - t)^2 / 2r^2) \tag{59}$$

where $\mathbf{d}^T$ is the current centre of the distribution (e.g., the datum previously added to the minimal subset). Our method is summarised in Algorithm 4. Fig. 11 illustrates the idea by multiplying the weights in Figs. 10 and 7; the outlier weights are severely attenuated, while the weights corresponding to inliers with *maximal distance* remain high.

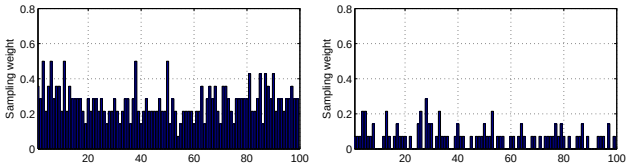
A practical difficulty is determining the appropriate value for t. In our experiments we obtain t as 2 times the average pairwise distance among the consensus set of the hypothesis

[3] Note that unlike Algorithms 1 and 2, Algorithms 3 and 4 update the sampling distribution (Step 5) according to the data available so far in the minimal subset. This can also be done for Algorithms 1 and 2, e.g., by recentring (54) and (55) on the datum last sampled. However our experiments suggest that this produces worse performance.

Algorithm 3 Multi-GS

Require: Design matrix $\mathbf{X} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_n]^T$, preferences $\{\mathbf{a}^{(i)}\}_{i=1}^n$ towards a set of hypotheses $\{\boldsymbol{\beta}_l\}_{l=1}^M$.

1: $\mathbf{d}_1 \leftarrow$ a row of $\mathbf{X}$ sampled randomly.
2: Initialise $\mathbf{w} = [f(\mathbf{x}_1, \mathbf{d}_1) \ \ldots \ f(\mathbf{x}_n, \mathbf{d}_1)]$.
3: **for** $j := 2, \ldots, m$ **do**
4: Sample a row $\mathbf{d}_j$ from $\mathbf{X}$ based on the weights
$$P(\mathbf{x}_i) \sim \mathbf{w}(i). \tag{61}$$
5: $\mathbf{w} = \mathbf{w} \odot [f(\mathbf{x}_1, \mathbf{d}_j) \ \ldots \ f(\mathbf{x}_n, \mathbf{d}_j)]$.
6: /* $\odot$ means element-wise product.*/
7: **end for**



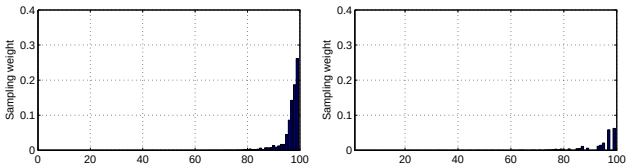
(a) Inlier weights (b) Outlier weights

Fig. 10 Weights for the data in Fig. 5(a) computed according to (58) when it is centred on the inlier chosen in Fig. 5(b). The data are first sorted in increasing distance to the chosen inlier. Preferences are induced from 150 previously generated hypotheses.

Algorithm 4 Multi-GS with offset

Require: Design matrix $\mathbf{X} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_n]^T$, preferences $\{\mathbf{a}^{(i)}\}_{i=1}^n$ towards a set of hypotheses $\{\boldsymbol{\beta}_l\}_{l=1}^M$, radius r, offset t.

1: $\mathbf{d}_1 \leftarrow$ a row of $\mathbf{X}$ sampled randomly.
2: Initialise $\mathbf{w} = [f(\mathbf{x}_1, \mathbf{d}_1) \ \ldots \ f(\mathbf{x}_n, \mathbf{d}_1)]$.
3: **for** $j := 2, \ldots, m$ **do**
4: Sample a row $\mathbf{d}_j$ from $\mathbf{X}$ based on the weights
$$P(\mathbf{x}_i) \sim \mathbf{w}(i) \exp(-(\|\mathbf{x}_i - \mathbf{d}_{j-1}\| - t)^2 / 2r^2). \tag{62}$$
5: $\mathbf{w} = \mathbf{w} \odot [f(\mathbf{x}_1, \mathbf{d}_j) \ \ldots \ f(\mathbf{x}_n, \mathbf{d}_j)]$.
6: /* $\odot$ means element-wise product.*/
7: **end for**



(a) Inlier weights (b) Outlier weights

Fig. 11 Weights for the data in Fig. 5(a) computed according to (59) when it is centred on the inlier chosen in Fig. 5(b). The data are first sorted in increasing distance to the chosen inlier. Radius r is 2 times the average nearest neighbour distance, while offset t is 2 times the average pairwise distance *among all inliers*.

$\boldsymbol{\beta}^*$ which has the largest consensus [8] thus far, i.e.,

$$t = \frac{4}{|S^*|(|S^*| - 1)} \sum_{\mathbf{x}_i, \mathbf{x}_j \in S^*} \|\mathbf{x}_i - \mathbf{x}_j\| \tag{60}$$

where $S^* = \{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbf{X}, |y_i - \boldsymbol{\beta}^* \mathbf{x}_i| \leq \gamma\}$, and γ is the inlier threshold. The idea is to iteratively gauge the maximum span based on the evidence available on-the-fly.

5.3 Experimental results

We focus on robust fundamental matrix estimation which is a common task in computer vision. We benchmark our method (Algorithm 4, henceforth, Multi-GS-Offset) against

1. Pure random sampling [8] (Random);
2. Proximity sampling [29, 21] (Proxim);
3. Proximity sampling with offset (Proxim-Offset);
4. Multi-GS [1, 2];
5. LO-RANSAC [5];
6. Guided-MLESAC [36]; and
7. PROSAC [3].

LO-RANSAC introduces an inner RANSAC loop which samples hypotheses from the data within the consensus of the best hypothesis thus far. This bears some similarity with Multi-GS-Offset which iteratively estimates the maximal span of the inliers. State-of-the-art methods like PROSAC [3] and Guided-MLESAC [36] use keypoint matching scores as a proxy for sampling weights. These scores are difficult to simulate realistically for synthetic data. Therefore we only evaluate PROSAC and Guided-MLESAC on real-data.

5.3.1 Stopping criterion

For all methods, we use the standard RANSAC stopping criterion; see [16, Eq. 4.18]. This requires an estimate of the inlier ratio, which is progressively increased based on the largest consensus found thus far. Although the stopping criterion assumes pure random sampling, following [5, 6, 29, 36] we also apply it in the guided sampling methods. Under guided sampling, the stopping criterion *overestimates* the number of samples to retrieve. This does not necessarily mean that guided sampling will take as long as pure random sampling, since guided sampling methods increase the consensus size (hence, inlier ratio estimate) faster.

In theory, given a sufficient amount of time, any sampling method (including pure random sampling) can obtain the maximum achievable consensus. An objective performance measure is therefore how soon does the method hit a sufficiently high consensus to satisfy the stopping criterion. A lower run time thus reflects a higher accuracy in sampling minimal subsets with good estimates. Consequently, we will use run time as the main benchmark in our experiments.

5.3.2 Synthetic data

We first test on synthetic data generated using SfM Toolkit; a sample is Fig. 5(a). For each data instance, a unique fundamental matrix is generated, from which 100 inlying point matches are sampled and then added with Gaussian noise of std. dev. 5 pixels. A number of outliers are created by randomly sampling points in the image domain (500×500 pixels). The points are normalised using the method of [15].

Parameter settings for all methods are as follows: Offset t for Proxim-Offset and Multi-GS-Offset are computed as in (60), while radius r (used also in Proxim) is set as $t/2$. The inlier threshold γ is set as 0.0005; this is used in calculating the consensus size, the consensus set in the inner loop of LO-RANSAC, and in evaluating (60) for t. Finally, for Multi-GS and Multi-GS-Offset, the preferences $\{\mathbf{a}_i\}_{i=1}^n$ are updated only after every 10 hypotheses; see [1, 2] for details.

We set the number of outliers as 50, 100, 150 and 200 (respectively, 33%, 50%, 60% and 67% outlier rates). For each outlier rate we generate 100 unique data instances and run each method. For each run we record

1. The number of minimal subsets sampled;
2. The number of *all-inlier* minimal subsets sampled;
3. The maximum and median span among *all-inlier* minimal subsets sampled, where span is measured as $|\mathbf{X}(\lambda)|^2$;
4. The maximum consensus among *all* hypotheses;
5. The number of true inliers within the consensus set of the maximum consensus hypothesis;
6. The classification error (number of mislabelled data, i.e., inliers labelled as outliers and vice versa) of the maximum consensus hypothesis;
7. The run time based on the RANSAC stopping criterion.

Note that LO-RANSAC contains an inner loop that samples larger than minimal subsets (following [5], the inner loop subset size is 14). To allow all sampled subsets from LO-RANSAC to be accounted for, we count an inner loop subset as a "minimal subset" so as to standardise notations. Further, we also approximate the span of an inner loop subset ν as

$$\max_{\lambda} |\mathbf{X}(\lambda|\nu)|^2 \tag{63}$$

where λ ranges over all minimal subsets, i.e., the largest span among all minimal subsets within ν. Although (12) is theoretically more justified, the value from (63) is more useful for comparisons with spans from other minimal subsets.

Table 1 illustrates the median results over 100 repetitions for each outlier rate. Note that although only 100 inliers were generated, it is possible that a few randomly produced outliers may align closely with the underlying model, thus contributing to consensus sizes above 100. A common trend is that all methods deteriorate with the increase in outlier rates. The top-3 methods are LO-RANSAC, Multi-GS and Multi-GS-Offset. At low outlier rates, the three methods give comparable sampling accuracy, although Multi-GS and Multi-GS-Offset appear to stop later due to their more involved computations. Nonetheless, as the outlier rate increases, Multi-GS and Multi-GS-Offset become much faster than the others, indicating their superior accuracy in sampling minimal subsets with good estimates (see Sec. 5.3.1).

The crux of this paper, however, is the maximisation of the span of *all-inlier* minimal subsets. Table 1 shows that Multi-GS-Offset is the most successful in this respect, as it

Table 1 Performance comparison of sampling algorithms on synthetic data. The best score on each measure is bolded.

Data		Random	Proxim	Proxim -Offset	LO-RAN SAC	Multi-GS	Multi-GS -Offset
100 inliers 50 outliers (33% outliers)	# min. subsets	268	326	248	**211**	252	232
	# all-inlier min. subsets	10	13	8	23	**101**	90
	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.58e0	4.93e-1	1.76e0	6.17e+1	8.01e0	**7.33e+1**
	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.08e-2	5.67e-4	1.16e-2	7.08e-2	3.25e-2	**1.97e-1**
	Max. consensus	93	90	94	100	99	**100**
	# of true inliers retrieved	91	88	92	98	98	**99**
	Classification error	11	14	10	4	3	**2**
	Run time	0.41	0.59	0.43	**0.28**	0.76	0.73
100 inliers 100 outliers (50% outliers)	# min. subsets	2395	2295	2319	1087	1031	**929**
	# all-inlier min. subsets	7	11	7	14	**255**	252
	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	5.68e-1	2.48e0	2.13e0	2.41e+1	3.84e+1	**6.94e+1**
	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.71e-3	4.53e-3	3.22e-3	2.79e-2	2.94e-2	**1.36e-1**
	Max. consensus	93	96	94	99	100	**101**
	# of true inliers retrieved	88	92	91	96	97	**99**
	Classification error	17	12	12	7	6	**3**
	Run time	4.43	5.11	5.06	**2.71**	7.38	7.24
100 inliers 150 outliers (60% outliers)	# min. subsets	11,339	13,425	10,256	5545	1056	**934**
	# all-inlier min. subsets	5	10	6	12	116	**120**
	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	5.97e-1	3.61e-1	7.27e-1	3.59e0	4.32e0	**3.89e+1**
	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.63e-2	5.17e-3	1.89e-2	2.64e-2	2.90e-2	**1.50e-1**
	Max. consensus	96	94	97	99	100	**102**
	# of true inliers retrieved	91	89	92	96	97	**99**
	Classification error	14	16	13	7	6	**4**
	Run time	26.95	31.37	24.46	13.09	8.14	**7.82**
100 inliers 200 outliers (67% outliers)	# min. subsets	46,622	51,694	62,870	36,527	1152	**947**
	# all-inlier min. subsets	4	9	5	16	74	**77**
	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	9.37e-2	5.61e-3	4.78e-4	5.92e0	7.40e0	**1.90e+1**
	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	4.16e-3	9.94e-4	7.24e-5	3.16e-2	5.54e-2	**2.08e-1**
	Max. consensus	95	94	92	97	98	**100**
	# of true inliers retrieved	90	89	87	93	94	**98**
	Classification error	15	16	18	11	10	**4**
	Run time	115.75	126.89	156.47	92.37	16.15	**14.53**

consistently obtains all-inlier minimal subsets with the highest span. This also translates into better sampling accuracy and run time compared to the closest competitor (Multi-GS), which does not deliberately maximise span.

5.3.3 Real data

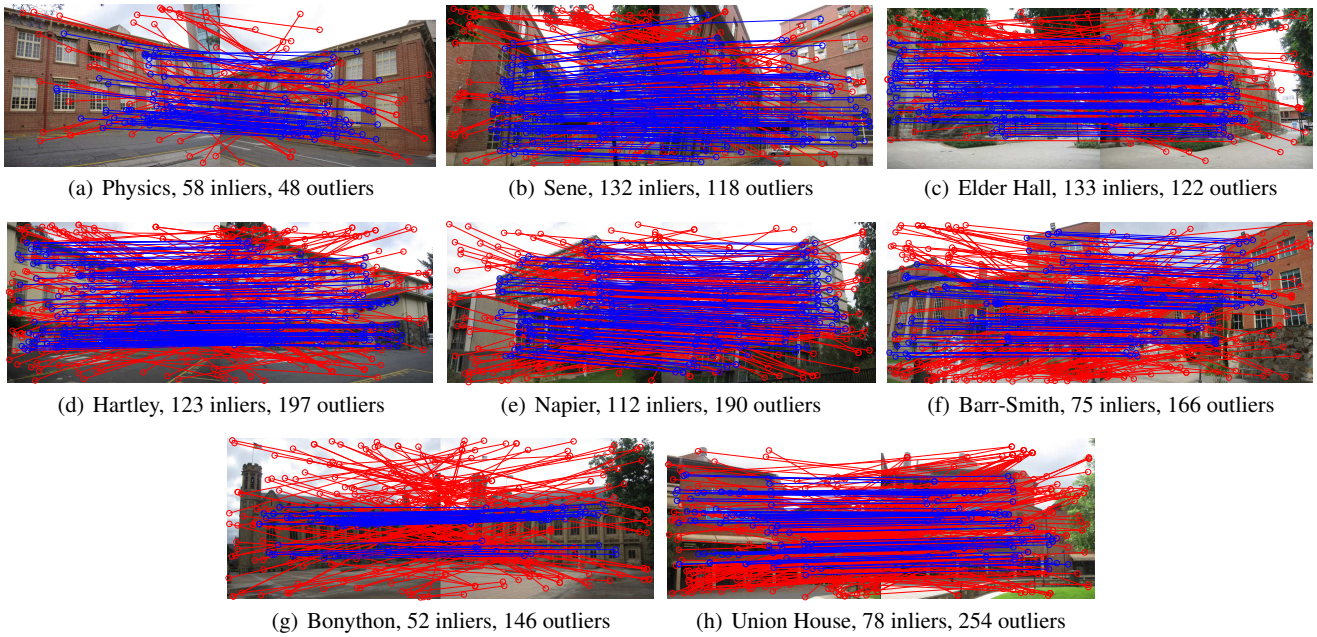
We now test on real data. Given a pair of overlapping views of a static scene, SIFT keypoints [24] are first detected and matched across the views (using VLFeat [39]). These point matches are manually identified as true matches (inliers) and false matches (outliers). The point coordinates are also normalised using the method of [15]. Parameter settings follow from the experiments on synthetic data, except for inlier threshold γ which is manually tuned for each dataset. The same threshold is given to all methods. SIFT matching scores are provided for PROSAC and Guided-MLESAC.

The datasets used are shown in Fig. 12. We use a subset of the publicly available AdelaideRMF dataset [40], where the more difficult datasets (e.g., higher outlier rates, larger inlier noise magnitudes) are chosen (the methods perform similarly on the other easier dataset). Each method is given 100 runs on each dataset. We record the same performance measures used in the synthetic data experiments. Table 2 shows the median results over 100 runs. Due to space constraints we omit the results from pure random sampling.

On datasets with low outlier rates, PROSAC and Guided-MLESAC are the fastest due to the benefits of SIFT matching scores. On datasets with higher outlier rates ($> 60\%$), the superior run time (and hence sampling accuracy) of Multi-GS and Multi-GS-Offset become apparent — on the three hardest datasets, Multi-GS and Multi-GS-Offset are almost 6 times faster than the others. In terms of maximising the span of all-inlier minimal subsets, it is clear that Multi-GS-Offset is the most successful in all datasets. This allows it to achieve a faster run time than Multi-GS.

5.3.4 Performance under degeneracies

We now examine the benefits of sampling with large spans for avoiding degeneracies. For the 8-point estimation method, a degenerate estimate is obtained when more than six matches among eight in the minimal subset lie on the same plane. This occurs frequently when there exists a dominant plane in the scene [7,9]. Theoretically, a degenerate minimal sub-



(a) Physics, 58 inliers, 48 outliers (b) Sene, 132 inliers, 118 outliers (c) Elder Hall, 133 inliers, 122 outliers

(d) Hartley, 123 inliers, 197 outliers (e) Napier, 112 inliers, 190 outliers (f) Barr-Smith, 75 inliers, 166 outliers

(g) Bonython, 52 inliers, 146 outliers (h) Union House, 78 inliers, 254 outliers

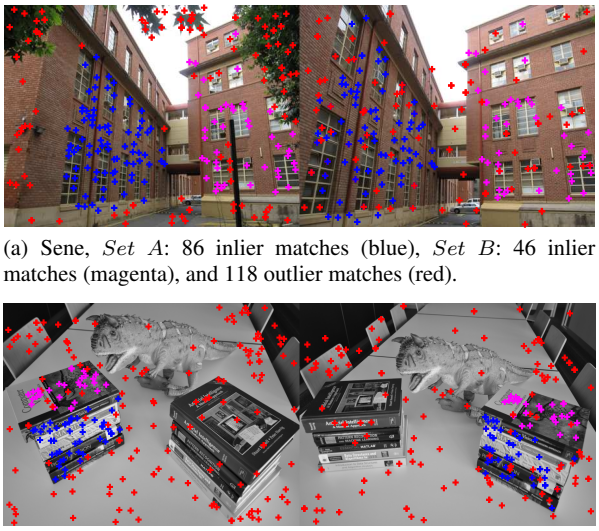
Fig. 12 Real datasets used in the experiments. Features are detected and matched across the two views using SIFT [24].

set has very small span ($|\mathbf{X}(\lambda)|$ is closed to 0). It is crucial for sampling methods to avoid degenerate minimal subsets.

We follow the experimental methodology of [2]. Scenes with two large planes are chosen, specifically *Sene* in Fig. 12 and *Dinobooks* in [2]. We detect and match SIFT keypoints which we then manually categorise into inliers and outliers. On each dataset, we keep the outliers and separate the inliers into two sets ($Set\ A$ and $Set\ B$) based on the plane on which they lie; see Fig. 13. We then create different levels of degeneracies by maintaining $Set\ A$ (the dominant plane inliers) while controlling the number of inliers in $Set\ B$ (the "off-plane" inliers). This yields the ratio between the dominant plane inliers and the total inliers as follows

$$\gamma = |Set\ A|/(|Set\ A| + |Set\ B|). \tag{64}$$

First, we focus on comparing Multi-GS and Multi-GS-Offset. The ratio γ is fixed at 0.7, and for each method, 10,000 minimal subsets are drawn from the set $Set\ A \cup Set\ B$. The distribution (obtained via kernel density estimation) of the weights $|\mathbf{X}(\lambda)|^2$ and consensus sizes of the minimal subsets are illustrated in Fig. 14. Observe that in both datasets, the consensus size distribution of Multi-GS tends to peak at $|Set\ A|$, indicating that Multi-GS tends to fit the dominant plane — this trend has been observed in [2]. In contrast, the peak of the consensus size distributions of Multi-GS-Offset is close to $|Set\ A \cup Set\ B|$, indicating that Multi-GS-Offset is more capable of capturing all of the inliers in the scene ($Set\ A$ and $Set\ B$). This represents clear evidence of the ability of Multi-GS-Offset to avoid degeneracies by sampling minimal subsets with large spans.

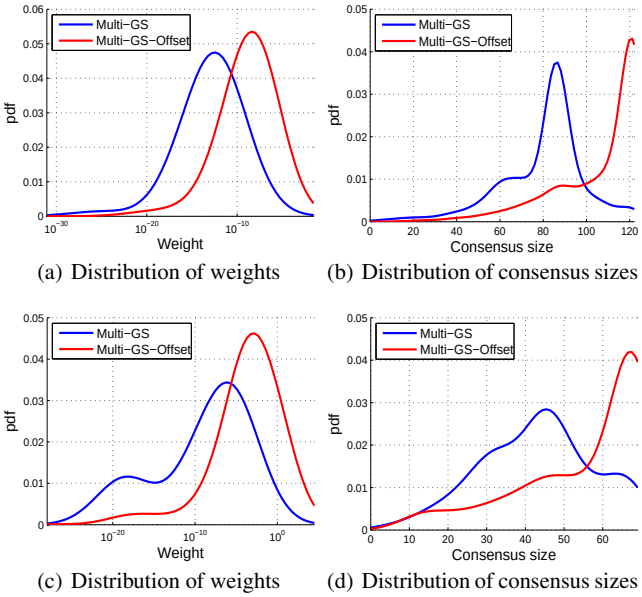


(a) Sene, $Set\ A$: 86 inlier matches (blue), $Set\ B$: 46 inlier matches (magenta), and 118 outlier matches (red).

(b) Dinobooks [2], $Set\ A$: 49 inlier matches (blue), $Set\ B$: 29 inlier matches (magenta), and 155 outlier matches (red).

Fig. 13 Image pairs of a static scene (thus supporting one fundamental matrix structure) where the inlier matches lie on two distinct planes.

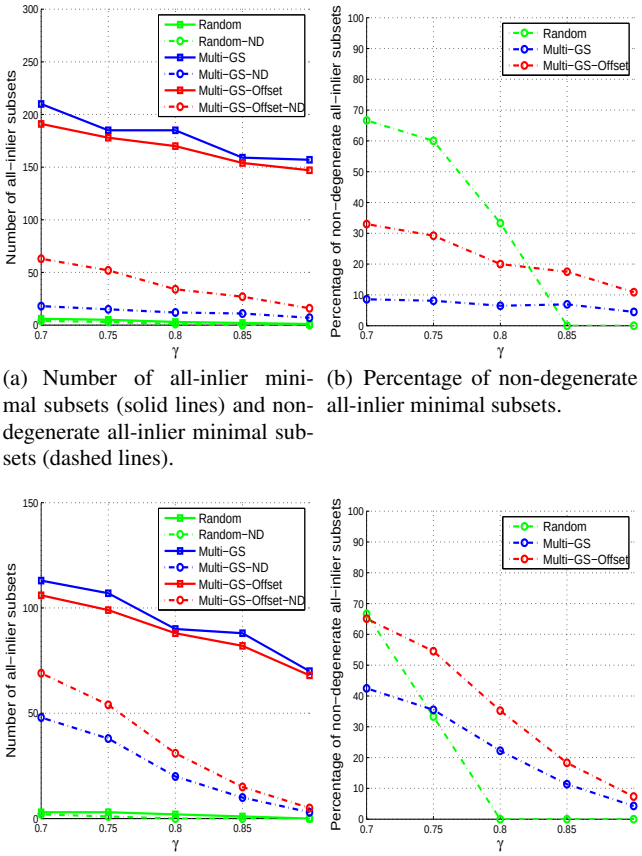
Next, we analyse with different levels of degeneracies. The ratio γ is varied within $[0.7, 0.9]$, For each γ, 100 instances of the data are generated; here, besides $Set\ A$ and $Set\ B$ we also include all the outliers. Each sampling method in Sec. 5.3 is invoked using the RANSAC stopping criterion. We record the number of all-inlier minimal subsets, and the number of *non-degenerate* all-inlier minimal subsets achieved — the latter is calculated by checking the plane on which each inlier lies. Fig. 15 shows the median results



(a) Distribution of weights (b) Distribution of consensus sizes

(c) Distribution of weights (d) Distribution of consensus sizes

Fig. 14 Comparison of Multi-GS and Multi-GS-Offset on degenerate configurations with $\gamma = 0.7$. Row 1: *Sene*, Row 2: *Dinobooks*. Note that in (a) and (c), the horizontal (weight) axis is in logarithmic scale.



(a) Number of all-inlier minimal subsets (solid lines) and non-degenerate all-inlier minimal subsets (dashed lines).

(b) Percentage of non-degenerate all-inlier minimal subsets.

(c) Number of all-inlier minimal subsets (solid lines) and non-degenerate all-inlier minimal subsets (dashed lines).

(d) Percentage of non-degenerate all-inlier minimal subsets.

Fig. 15 Performance comparison on degenerate configurations with $\gamma \in [0.7, 0.9]$. Row 1: *Sene*. Row 2: *Dinobooks*.

of Multi-GS and Multi-GS-Offset, with Random also included as the baseline (the performances of the other methods are significantly worse than Multi-GS and Multi-GS-Offset, so we did not plot them in the figure). Expectedly all methods deteriorate with the increase in γ, since this entails the increase in outlier rates. Although obtaining fewer all-inlier samples than Multi-GS, Multi-GS-Offset achieves more non-degenerate all-inlier samples; see the dashed lines in Figs. 15(c) and 15(a). This naturally leads to a higher percentage of non-degenerate samples over all-inlier samples; see Fig. 15(d) and 15(b). Note that since Random hits very few all-inlier minimal subsets, its percentage of non-degenerate minimal subsets is sometimes high by chance.

6 Conclusions

In this paper we have investigated the role of data span in parameter estimation. The starting point was a result by Jacobi which expresses the least squares estimate as weighted sum of minimal estimates, where the weight of a minimal estimate is a function of the span of the associated data. Our main theoretical contribution is to show that such an expression can be developed for TLS. We also highlighted the equivalence between TLS and DLT which is a common parameter estimation technique in computer vision.

Another main contribution is an algorithm that can sample all-inlier minimal subsets with large spans for robust parameter estimation. We showed that simple distance-based sampling is not effective for searching for all-inlier minimal subsets with large spans. In constrast our algorithm can consciously target minimal subsets with large spans without a decreased accuracy in finding all-inlier minimal subsets. This also permits an ability to avoid sampling degenerate minimal subsets. The superior performance of the proposed algorithm is demonstrated on synthetic and real data.

Appendix

Proof of Proposition 2 for TLS with non-minimal subsets

From Sec. 2.1, the weight of a non-minimal subset ν is proportional to $|\mathbf{X}(\nu)^T\mathbf{X}(\nu)|$ which, from (36), is equal to

$$|\mathbf{X}(\nu)^T\mathbf{X}(\nu)| = |\mathbf{V}\mathbf{S}_m^T\mathbf{U}_m(\nu)^T\mathbf{U}_m(\nu)\mathbf{S}_m\mathbf{V}^T| \quad (65)$$

$$= |\mathbf{U}_m(\nu)^T\mathbf{U}_m(\nu)||\mathbf{S}_m\mathbf{V}^T|^2. \quad (66)$$

Similarly,

$$|\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)| = |\mathbf{U}_m(\nu)^T\mathbf{U}_m(\nu)||\tilde{\mathbf{S}}_m\mathbf{V}^T|^2. \quad (67)$$

Therefore, $|\mathbf{X}(\nu)^T\mathbf{X}(\nu)| = \alpha|\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)|$ where α is a constant which does not depend on ν. This proves that

$$|\mathbf{X}(\nu_1)^T\mathbf{X}(\nu_1)| > |\mathbf{X}(\nu_2)^T\mathbf{X}(\nu_2)| \tag{68}$$

$$\implies |\mathbf{Z}(\nu_1)^T\mathbf{Z}(\nu_1)| > |\mathbf{Z}(\nu_2)^T\mathbf{Z}(\nu_2)|. \tag{69}$$

Proof of Propositions 1 and 2 for mixed OLS-TLS

We aim to prove that Proposition 1 holds for the mixed OLS-TLS problem (Sec.3.3), i.e., the solution to the OLS problem

$$\arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \quad \text{s.t.} \quad \mathbf{Z}\boldsymbol{\beta} = \hat{\mathbf{y}} \tag{70}$$

coincides with the mixed OLS-TLS estimate

$$\breve{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X} - \sigma^2_{m_2+1}\mathbf{L}\right)^{-1}\mathbf{X}^T\mathbf{y} = (\mathbf{X}^T\mathbf{Z})^{-1}\mathbf{X}^T\mathbf{y} \tag{71}$$

where

$$\mathbf{Z} := \mathbf{X} - \sigma^2_{m_2+1}(\mathbf{X}^T)^\dagger\mathbf{L}. \tag{72}$$

See Sec. 3.3 for the definition of the other symbols involved.

Let $\tilde{\boldsymbol{\beta}}$ be the solution to (70). Then

$$\tilde{\boldsymbol{\beta}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} \tag{73}$$

which, following the proof of Proposition 1, can be rearranged to become

$$\mathbf{X}^T\mathbf{Z}\tilde{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y} + \sigma^2_{m+1}\mathbf{L}^T(\mathbf{X}^T)^{\dagger T}(\mathbf{Z}\tilde{\boldsymbol{\beta}} - \mathbf{y}). \tag{74}$$

As shown in the proof of Proposition 1, the column spans of $\mathbf{Z}$ and $(\mathbf{X}^T)^\dagger$ are equal. Since vector $(\mathbf{Z}\tilde{\boldsymbol{\beta}} - \mathbf{y})$ is orthogonal to $\mathcal{R}(\mathbf{Z})$, it is also orthogonal to $\mathcal{R}((\mathbf{X}^T)^\dagger)$, thus the second component on the RHS of (74) equates to 0, yielding

$$\mathbf{X}^T\mathbf{Z}\tilde{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{y}. \tag{75}$$

Comparing (75) to (71) proves $\tilde{\boldsymbol{\beta}} = \breve{\boldsymbol{\beta}}$, i.e., the mixed OLS-TLS estimate $\breve{\boldsymbol{\beta}}$ coincides with the solution of the OLS (70).

To prove that Proposition 2 also holds for mixed OLS-TLS, it is sufficient to show that, given a non-minimal data subset ν of size $m + i \leq n$,

$$|\mathbf{X}(\nu)^T\mathbf{X}(\nu)| \propto |\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)|. \tag{76}$$

To begin, let $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ be the SVD of $\mathbf{X}$. Then $(\mathbf{X}^T)^\dagger = \mathbf{U}\mathbf{S}^{-1}\mathbf{V}^T$ is the SVD of $(\mathbf{X}^T)^\dagger$. Also, since $n > m$

$$\mathbf{X} = \mathbf{U}_m\mathbf{S}_m\mathbf{V}^T \quad (\mathbf{X}^T)^\dagger = \mathbf{U}_m\mathbf{S}_m^{-1}\mathbf{V}^T. \tag{77}$$

Then, from (72)

$$\mathbf{Z} = \mathbf{U}_m(\mathbf{S}_m\mathbf{V}^T - \sigma^2_{m_2+1}\mathbf{S}_m^{-1}\mathbf{V}^T\mathbf{L}) = \mathbf{U}_m\boldsymbol{\Gamma} \tag{78}$$

where we define the square matrix

$$\boldsymbol{\Gamma} := (\mathbf{S}_m\mathbf{V}^T - \sigma^2_{m_2+1}\mathbf{S}_m^{-1}\mathbf{V}^T\mathbf{L}). \tag{79}$$

Also, observe that

$$\mathbf{Z}(\nu) = \mathbf{U}_m(\nu)\boldsymbol{\Gamma}. \tag{80}$$

In (66), the following determinant has been established

$$|\mathbf{X}(\nu)^T\mathbf{X}(\nu)| = |\mathbf{U}_m(\nu)^T\mathbf{U}_m(\nu)||\mathbf{S}_m\mathbf{V}^T|^2. \tag{81}$$

The determinant $|\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)|$ is then

$$|\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)| = |\mathbf{U}_m(\nu)^T\mathbf{U}_m(\nu)||\boldsymbol{\Gamma}|^2 \tag{82}$$

which implies $|\mathbf{X}(\nu)^T\mathbf{X}(\nu)| \propto |\mathbf{Z}(\nu)^T\mathbf{Z}(\nu)|$. Note that this result also holds for minimal subsets by setting $i = 0$.

References

1. Chin, T.J., Yu, J., Suter, D.: Accelerated hypothesis generation for multi-structure robust fitting. In: European Conference on Computer Vision (ECCV) (2010)
2. Chin, T.J., Yu, J., Suter, D.: Accelerated hypothesis generation for multi-structure data via preference analysis. IEEE Trans. Pattern Anal. Mach. Intell. **34**(4), 625–638 (2012)
3. Chum, O., Matas, J.: Matching with PROSAC- progressive sample consensus. In: Computer Vision and Pattern Recognition (CVPR) (2005)
4. Chum, O., Matas, J.: Planar affine rectification from change of scale. In: Asian Conference on Computer Vision (ACCV) (2010)
5. Chum, O., Matas, J., Kittler, J.: Locally optimized RANSAC. In: Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM) (2003)
6. Chum, O., Matas, J., Obdrzakek, S.: Enhancing RANSAC by generalized model optimization. In: Asian Conference on Computer Vision (ACCV) (2004)
7. Chum, O., Werner, T., Matas, J.: Two-view geometry estimation unaffected by a dominant plane. In: Computer Vision and Pattern Recognition (CVPR) (2005)
8. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981)
9. Frahm, J.M., Pollefeys, M.: RANSAC for (quasi-)degenerate data (QDEGSAC). In: Computer Vision and Pattern Recognition (CVPR) (2006)
10. Golub, G.H., Hoffman, A., Stewart, G.W.: A generalization of the Eckart-Young-Mirksy matrix approximation theorem. Linear Algebra Appl **88–89**, 317–327 (1987)
11. Golub, G.H., van Loan, C.F.: An analysis of the total least squares problem. Numer. Anal. **17**, 883–893 (1980)
12. Goshen, L., Shimshoni, I.: Balanced exploration and exploitation model search for efficient epipolar geometry estimation. IEEE Trans. Pattern Anal. Mach. Intell. (2008)
13. de Groen, P.: An introduction to total least squares. Nieuw Archief voor Wiskunde **4**(14), 237–253 (1996)
14. Harker, M., O'Leary, P.: Direct estimation of homogeneous vectors: an ill-solved problem in computer vision. In: Indian Conference on Computer Vision, Graphics and Image Processing (2006)
15. Hartley, R.: In defense of the eight-point algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **19**(6), 580–593 (1997)
16. Hartley, R., Zisserman, A.: Multiple View Geometry, 2nd edn. Cambridge University Press (2004)
17. Hoerl, A.E., Kennard, R.W.: A note on least squares estimates. Communications in Statistics: Simulation and Computation **9**(3), 315–317 (1980)
18. Jacobi, C.G.J.: De formatione et proprietatibus determinantium. J. Reine Angew. Math. **9**, 315–317 (1841)

19. Kahl, F., Hartley, R.: Multiple-view geometry under the l_∞-norm. IEEE Trans. Pattern Anal. Mach. Intell. **30**(9), 1603–1617 (2008)
20. Kahl, F., Henrion, D.: Globally optimal estimates for geometric reconstruction problems. In: International Conference on Computer Vision (ICCV) (2005)
21. Kanazawa, Y., Kawakami, H.: Detection of planar regions with uncalibrated stereo using distributions of feature points. In: British Machine Vision Conference (BMVC) (2004)
22. Kemp, C., Drummond, T.: Dynamic measurement clustering to aid real time tracking. In: International Conference on Computer Vision (ICCV) (2005)
23. Kukush, A., Markovsky, I., Huffel, S.V.: Consistent fundamental matrix estimation in a quadratic measurement error model arising in motion analysis. Computational Statistics and Data Analysis **3**(18), 3–18 (2002)
24. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**(2), 91–110 (2004)
25. Meer, P.: Robust techniques for computer vision. In: G. Medioni, S.B. Kang (eds.) Emerging Topics in Computer Vision. Prentice Hall (2004)
26. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2004)
27. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. Int. J. Comput. Vision **65**(1), 43–72 (2005)
28. Mühlich, M., Mester, R.: The role of total least squares in motion analysis. In: European Conference on Computer Vision (ECCV) (1998)
29. Myatt, D.R., Torr, P.H.S., Nasuto, S.J., Bishop, J.M., Craddock, R.: NAPSAC: high noise, high dimensional robust estimation - it's in the bag. In: British Machine Vision Conference (BMVC) (2002)
30. Olsson, C., Eriksson, A., Hartley, R.: Outlier removal using duality. In: Computer Vision and Pattern Recognition (CVPR) (2010)
31. Pham, T.T., Chin, T.J., Yu, J., Suter, D.: The random cluster model for robust geometric fitting. In: Computer Vision and Pattern Recognition (CVPR) (2012)
32. Rousseeuw, P.J., Leroy, A.M.: Robust regression and outlier detection. Wiley (1987)
33. Scherer-Negenborn, N., Schaefer, R.: Model fitting with sufficient random sample coverage. Int. J. Comput. Vision **89**, 120–128 (2010)
34. Stigler, S.M.: The history of statistics: the measurement of uncertainty before 1900, 8th edn., chap. 1. The Belknap Press of Harvard University Press (2000)
35. Subrahmanyam, M.: A property of simple least squares estimates. Sankhya B **34**, 3 (1972)
36. Tordoff, B.J., Murray, D.W.: Guided-MLESAC: Faster image transform estimation by using matching priors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1523–1535 (2005)
37. van Huffel, S., Wandewalle, J.: Algebraic connections between the least squares and total least squares problem. Numer. Math. **55**, 431–449 (1989)
38. van Huffel, S., Wandewalle, J.: The total least squares problem: computational aspects and analysis. SIAM Publications (1991)
39. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/ (2008)
40. Wong, H.S., Chin, T.J., Yu, J., Suter, D.: Dynamic and hierarchical nulti-structure geometric model fitting. In: International Conference on Computer Vision (ICCV) (2011)
41. Zhang, Z.: Parameter estimation techniques: a tutorial with application to conic fitting. Image and Vision Computing **15**(1), 59–76 (1997)

Table 2 Performance comparison of sampling algorithms on real data. The best score on each measure is bolded.

Data		Proxim	Proxim-Offset	LO-RANSAC	Guided-MLESAC	PROSAC	Multi-GS	Multi-GS-Offset
Physics	# min. subsets	571	572	504	500	**498**	522	504
58 inliers	# all-inlier min. subsets	18	11	25	14	26	**215**	188
48 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	2.27e-7	2.40e-8	5.53-6	3.12e-6	1.34e-6	1.49e-5	**2.46e-5**
(45% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	3.37e-10	4.78e-11	5.88e-8	4.25e-8	2.90e-8	1.57e-6	**2.90e-6**
	Max. consensus	58	58	59	59	59	**60**	**60**
	# of true inliers retrieved	56	56	57	57	57	**58**	**58**
	Classification error	4	4	3	3	3	**2**	**2**
	Run time	0.89	0.91	0.70	0.70	**0.68**	1.85	1.81
Sene	# min. subsets	614	666	564	562	**559**	652	615
132 inliers	# all-inlier min. subsets	8	5	31	48	143	**206**	185
118 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.51e-8	5.76e-7	1.28e-6	1.85e-6	3.75e-6	1.35e-6	**8.75e-5**
(47% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	2.07e-11	1.13e-10	5.32e-9	8.13e-8	9.29e-8	1.45e-7	**3.39e-6**
	Max. consensus	133	133	134	134	134	134	**135**
	# of true inliers retrieved	129	130	130	131	131	131	**132**
	Classification error	7	5	6	4	4	4	**3**
	Run time	1.67	1.77	1.43	1.42	**1.39**	5.30	5.12
Elder Hall	# min. subsets	662	675	595	592	589	630	**585**
133 inliers	# all-inlier min. subsets	9	4	26	44	143	**188**	172
122 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	7.12e-9	6.13e-10	2.82e-7	4.22e-7	1.30e-6	4.62e-7	**4.91e-5**
(48% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.52e-11	2.40e-12	2.26e-8	3.24e-8	1.14e-7	1.19e-8	**1.37e-6**
	Max. consensus	132	131	132	132	132	132	**134**
	# of true inliers retrieved	127	126	128	131	131	131	**133**
	Classification error	11	12	9	3	3	3	**1**
	Run time	1.90	1.93	1.68	1.60	**1.56**	6.20	5.71
Hartley	# min. subsets	4078	4325	3632	3850	3848	474	**446**
123 inliers	# all-inlier min. subsets	5	2	15	19	66	**84**	79
197 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	9.39e-9	4.53e-10	4.15e-7	3.81e-7	5.56e-7	7.36e-7	**2.47e-6**
(62% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	1.11e-10	2.31e-12	1.02e-8	1.13e-8	2.48e-8	4.04e-7	**1.07e-6**
	Max. consensus	127	127	128	128	128	128	**129**
	# of true inliers retrieved	118	118	120	121	121	121	**123**
	Classification error	14	14	11	9	9	9	**6**
	Run time	12.92	14.02	11.67	12.47	12.22	3.73	**3.67**
Napier	# min. subsets	15,277	13,145	6491	7409	8774	724	**688**
112 inliers	# all-inlier min. subsets	7	4	18	12	16	**119**	107
190 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	4.62e-7	7.48e-7	5.98e-6	1.46e-6	9.71e-6	9.92e-6	**5.15e-5**
(63% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	8.77e-11	4.55e-10	1.75e-8	2.59e-9	6.35e-8	7.51e-8	**8.31e-7**
	Max. consensus	112	114	119	118	119	120	**121**
	# of true inliers retrieved	104	106	110	109	110	111	**112**
	Classification error	16	14	11	12	11	10	**9**
	Run time	44.39	38.54	18.35	21.19	24.26	6.45	**5.90**
Barr-Smith	# min. subsets	47,453	54,462	42,178	39,423	38,578	639	**583**
75 inliers	# all-inlier min. subsets	8	5	24	19	22	85	**88**
166 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	2.42e-10	4.19e-11	6.09e-8	7.25e-8	8.86e-8	1.21e-7	**2.84e-7**
(69% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	6.22e-11	8.82e-12	5.15e-9	6.07e-9	6.89e-9	3.29e-8	**7.23e-8**
	Max. consensus	76	75	78	78	78	79	**80**
	# of true inliers retrieved	71	70	72	73	73	74	**75**
	Classification error	9	10	9	7	7	6	**5**
	Run time	80.45	91.87	70.75	66.18	64.53	4.28	**4.09**
Bonython	# min. subsets	462,871	470,628	330,352	394,366	388,741	830	**713**
52 inliers	# all-inlier min. subsets	7	5	33	21	35	113	**118**
146 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	5.92e-15	1.12e-15	5.35e-13	7.82e-14	8.77e-14	9.35e-13	**1.62e-11**
(74% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	2.89e-17	1.25e-17	3.09e-15	3.37e-16	4.59e-16	4.01e-15	**4.94e-13**
	Max. consensus	47	47	49	48	48	49	**51**
	# of true inliers retrieved	44	43	46	46	46	47	**50**
	Classification error	11	13	9	8	8	7	**3**
	Run time	911.32	927.68	650.64	778.16	765.29	5.48	**5.19**
Union House	# min. subsets	374,852	415,178	307,632	286,186	277,379	696	**584**
78 inliers	# all-inlier min. subsets	5	4	17	26	48	73	**76**
254 outliers	Max. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	6.31e-13	2.05e-13	4.41e-12	9.22e-12	7.83e-11	1.47e-9	**4.32e-9**
(77% outliers)	Med. all-inlier span $\|\mathbf{X}(\lambda)\|^2$	7.28e-15	3.13e-15	2.83e-14	6.19e-14	5.35e-13	1.39e-11	**9.61e-10**
	Max. consensus	81	80	83	84	84	85	**86**
	# of true inliers retrieved	73	72	74	75	75	76	**78**
	Classification error	13	14	13	12	12	11	**8**
	Run time	1096.52	1213.76	900.35	837.74	811.72	6.98	**5.51**