# Simultaneously Fitting and Segmenting Multiple-Structure Data with Outliers

Hanzi Wang, *Senior Member*, *IEEE*, Tat-Jun Chin, *Member*, *IEEE*, and David Suter, *Senior Member*, *IEEE*

**Abstract**—We propose a robust fitting framework, called Adaptive Kernel-Scale Weighted Hypotheses (AKSWH), to segment multiple-structure data even in the presence of a large number of outliers. Our framework contains a novel scale estimator called Iterative Kth Ordered Scale Estimator (IKOSE). IKOSE can accurately estimate the scale of inliers for heavily corrupted multiple-structure data and is of interest by itself since it can be used in other robust estimators. In addition to IKOSE, our framework includes several original elements based on the weighting, clustering, and fusing of hypotheses. AKSWH can provide accurate estimates of the number of model instances and the parameters and the scale of each model instance simultaneously. We demonstrate good performance in practical applications such as line fitting, circle fitting, range image segmentation, homography estimation, and two-view-based motion segmentation, using both synthetic data and real images.

**Index Terms**—Robust statistics, model fitting, scale estimation, kernel density estimation, multiple structure segmentation.

## 1 Introduction

We consider the setting where one is given some data and within that data there are several structures of interest: where one knows the general form of a parametric model that these structures "fit." We are motivated by applications in computer vision such as line and circle finding in images, homography/fundamental matrix estimation, optical flow calculation, range image segmentation, motion estimation and segmentation. We adopt the paradigm that these models (line, circle, homography, fundamental matrix, etc.) can be found by robust fitting, "robust" because the data contain outliers or pseudo-outliers (the latter refers to the fact that data belonging to one structure are usually outliers to any other structure) [2], [3].

Traditional robust approaches such as M-estimators [4], LMedS [5], and LTS [6] cannot tolerate more than 50 percent of outliers. This is a severe limitation when these approaches are applied to multiple-structure data, where outliers (and pseudo-outliers to any structure) are a large fraction of the data.

A number of robust approaches claim to tolerate more than 50 percent of outliers: HT [7], RHT [8], RANSAC [9], MSAC [10], MUSE [11], MINPRAN [12], ALKS [13], RESC [14], pbM [15], HBM [16], ASSC [2], ASKC [17], etc. These cannot be considered as complete solutions to the above problem because they either require a sequential "*fit-and-remove*" procedure (i.e., sequentially estimate the parameters of one model instance, dichotomize the inliers belonging to that model instance from outliers, and remove the inliers from data) or some form of further analysis, in the case of HT/RHT, to segment multiple-modes in "Hough space" (i.e., model parameter space). Moreover, the methods (typified by RANSAC and related estimators) generally require one or more parameters related to the scale of inliers (or equivalently, the error tolerance) to be either specified by the user or estimated by a separate procedure.

Further, we note that the sequential "*fit-and-remove*" procedure has limitations: 1) If the parameters or the scale of a model instance are incorrect, the inliers of the remaining model instances may be wrongly removed, and the inliers to the current model instance may be incorrectly left in. This will lead to inaccurate estimation of the remaining model instances in the following steps. 2) Also, the sequential "*fit-and-remove*" procedure is not computationally efficient because it requires generating a large number of hypotheses in each step.

Alternatives have been proposed, but these too have limitations. MultiRANSAC [18] requires the user to specify both the number of model instances and the inlier scales. GPCA [19] is restricted to fitting linear subspaces, and requires that the user specify the number of populations. Moreover, GPCA becomes problematic for data involving outliers and a large number of structures. Mean Shift (MS) [20], [21] and its variants [22], [23] attempt to detect/estimate model instances by finding peaks in parameter space; however, this becomes very difficult when the inlier noise is high and/or the percentage of outliers is high. Moreover, it is not trivial to find a global optimal bin size for HT/RHT or an optimal bandwidth for the MS-based approaches to both achieve accurate results and correctly localize multiple significant peaks in parameter space.

- *H. Wang is with the School of Information Science and Technology, Xiamen University, Fujian, 361005, China, the University of Adelaide, Australia, and the Fujian Key Laboratory of the Brain-like Intelligent Systems (Xiamen University), Xiamen, China. E-mail: hanzi.wang@ieee.org.*
- *T.-J. Chin and D. Suter are with the School of Computer Science, The University of Adelaide, North Terrace, South Australia 5005, Australia. E-mail: {tat-jun.chin, david.suter}@adelaide.edu.au.*

*Manuscript received 6 July 2010; revised 6 Aug. 2011; accepted 21 Sept. 2011; online 1 Nov. 2011.*
*Recommended for acceptance by C. Stewart.*
*For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2010-07-0509.*
*Digital Object Identifier no. 10.1109/TPAMI.2011.216.*

RHA [24] is predicated on an interesting property of residuals (to hypothesized fits) when the model instances are parallel to each other; however, any guarantee of performance when models are not parallel relies on (limited) empirical support. Also, RHA requires the user to specify both the histogram bin size (limitation shared by HT/RHT and others) and a threshold for selecting the significant local peaks corresponding to the model instance number.

A Jaccard-distance linkage (J-linkage)-based agglomerative clustering technique [25] is more effective than RHA when model instances are not parallel. However, they dichotomize inliers/outliers by a user-specified inlier scale, which is not known in practice. Like RANSAC, the performance of J-linkage greatly depends on the input inlier scale.

In the kernel fitting (KF) approach [26]: Gross outliers are identified first and removed, and then the multiple structures are estimated in parallel. Unlike J-linkage and RANSAC, KF does not require a user to specify the inlier scale. However, KF uses a step size $h$ in computing the ordered residual kernel. KF also requires a user to specify the relative weighting ratio between fitting error and model complexity in implementing the model selection procedure.

We note that although the MS-based approaches [20], [22], [23], RHA [24], J-linkage [25], and KF [26] claim to be able to detect the number of model instances, they all require some user-specified thresholds whose values are crucial in determining the number of model instances (e.g., Subbarao and Meer [23] suggest for the MS-based approaches that the significant peaks are selected if the first N modes clearly dominate the (N+1)th mode but the authors do not discuss how to judge if the Nth mode dominates the (N+1)th mode, nor do they specify a threshold value for the judgment. RHA [24] uses a threshold to eliminate spurious peaks in the residual histogram, while J-linkage determines the number of model instances by selecting significant bins of the hypothesis histogram whose value is larger than a user-specified threshold. The value of the step size $h$ and the weighting ratio used in KF has a significant influence on determining the number of model instances [26]). *Detecting the number of model instances is still one of the most difficult things in model fitting* [27], [ 28].

Our approach is based on sampling parameter space, Section 3.1 (since each point in this space is a potential model, we also call this "hypothesis space")—this starting point is similar to HT/RHT. Fig. 1 illustrates the basic steps—starting from weighting the hypotheses samples in Fig. 1c. A key element in efficiency is that we prune this space (Section 3.3), using a principled *information theoretic criteria*, after weighting each hypothesis so as to emphasize likely candidates (Section 3.2). Since multiple surviving (hence highly likely to be approximations to a true structure) samples in hypothesis space may actually come from the same structure, we cluster the surviving hypotheses (Section 3.4) using an approach inspired by J-linkage (but with very important differences—see the discussion therein) that, while providing an advance toward good clusters, does have a tendency to oversegment the clusters (Fig. 1d). Thus, we refine these clusters (Section 3.5) using, again, a principled information theoretic criterion (Fig. 1e).

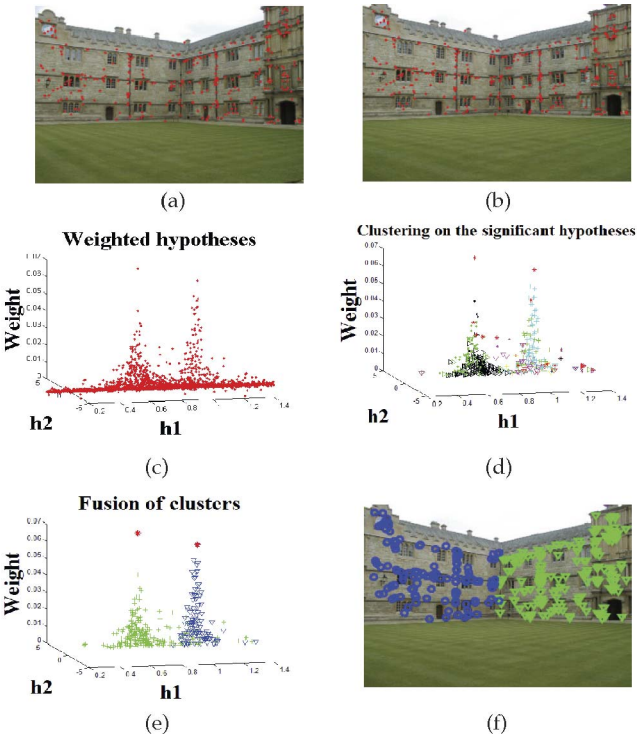



Fig. 1. Overview of our approach: (a) and (b) Image pair with SIFT features, (c) weighted hypotheses (plotted using the first two coefficients of homography matrix), (d) clusters (each color represents a different cluster) after removing weak hypotheses and clustering, (e) the surviving two hypotheses (red) and the color code of hypotheses judged to belong to each structure, (f) feature point data segmented according to structure.

Like most algorithms, our approach is crucially dependent on the quality of the output of the first stage—in our case, the weighting of hypotheses before pruning. Our weighting is based on the notion that a "good hypothesis" is one that "explains well" the data "around it." In essence, this notion, like in many methods, is based upon an analysis of the smaller residuals (if the model is a true one, then the data giving rise to these residuals would likely "belong" to that model) that essentially characterizes an effective inlier scale. Here, we do two things—we improve the scale estimator KOSE [13] (the new scale estimator is called IKOSE—Section 2.2) and we also improve on the scoring function (Section 3.2) used in [17], [29] (first with a better estimate of scale and second in how we use that scale).

Our main contribution is that we develop a framework (AKSWH) that simultaneously fits all models to multiple-structured data and segments the data. *In the framework, the relevant quantities (the parameters, the scales, and the number of model instances) are determined automatically*. In so doing, we have original contributions in various building blocks: 1) We propose a robust scale estimator called Iterative Kth Ordered Scale Estimator (IKOSE), which can accurately estimate the scale of inliers for heavily corrupted data with multiple structures. IKOSE is of interest by itself since it can be a component in any robust approach that uses a data driven scale estimate. 2) We propose an effective hypotheses weighting function that contributes to rapid elimination of poor hypotheses. Our combination of this with a (fully automatic) entropy thresholding approach leads to a high computational efficiency. 3) From that point, our algorithm

uses the relatively common "*divide into small clusters (i.e., oversegmented clusters) and then merge*" procedure, but with a couple of novel features, such as effectively using a Jaccard distance-based clustering algorithm refined by the use of mutual information theory (MIT) to fuse the oversegmented clusters (e.g., when the inlier noise of a structure is large, there may be more than one cluster corresponding to the structure). Overall, as a result of these innovations, *AKSWH does not require the user to specify the number of structures nor the inlier scale of each model instance. AKSWH can deal with multiple-structured data heavily corrupted with outliers*. Our experimental results show that the proposed IKOSE and AKSWH outperform competing approaches in both the scale estimation and model fitting.

We are not claiming to have solved the impossible—a completely automated system that will determine exactly how many structures there are in a scene and recover all these structures with high precision and reliability. Such is impossible because it depends upon what is considered as an important structure (a wall is a single plane at one resolution but a collection of closely aligned planes at finer resolution). Rather, what we are claiming is that: when one has a number of roughly comparable sized structures in the scene, our approach will determine (and, to an extent, better than other competitors) not only the number of those structures, but also reliably fit and segment them.

The remainder of the paper is constructed as follows: In Section 2, we review existing scale estimation techniques and then propose a novel scale estimator and evaluate its performance. In Section 3, we develop the main components of the proposed approach. We put all the developed components together to obtain a complete framework in Section 4. In Section 5, we provide experimental results (with both synthetic and real data), and we give a concluding discussion in Section 6.

## 2 Scale Estimation

Scale estimation plays an important role in robust model fitting and segmentation. The accuracy of the scale estimate (or error tolerance) greatly affects the performance of many robust estimators such as RANSAC, HT, RHT, multiRANSAC, M-estimators, MSAC, ALKS, ASSC, and J-linkage. The importance of the scale estimation derives from: 1) It can be used to dichotomize inliers and outliers (e.g., in [9], [13], [15], [18], [25]); 2) it can be used to select the best hypothesis (say, in [13], [30]); 3) it can be used to determine various data dependent parameters such as the bandwidth value (e.g., for [15], [30]) or the bin size (e.g., for [7], [8]). Thus, a better scale estimator is a crucial component in solving the multi-structure fitting and segmentation problems.

In this section, we review several popular robust scale estimators and then propose a new scale estimator (IKOSE), followed by an experimental evaluation.

### 2.1 Review of Scale Estimators

Given a set of data points in $d$ dimensions $\mathbf{X} := \{\boldsymbol{x}_i | \boldsymbol{x}_i := (x_{i1}, \ldots, x_{id})\}$, where $i = 1, \ldots, n$, and $\boldsymbol{x}_i \in \mathbb{R}^d$, and the parameter estimate of a model $\hat{\boldsymbol{\theta}} := (\hat{\theta}_1 \cdots \hat{\theta}_p)' (\hat{\boldsymbol{\theta}} \in \mathbb{R}^p)$, the residual $r_i$ corresponding to the $i$th data sample is written as

$$r_i := F(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}), \tag{1}$$

where $F(.)$ is a function computing the residual of a data point $\boldsymbol{x}_i$ to the parameter estimate of a model $\hat{\boldsymbol{\theta}}$. Denote by $|\tilde{r}_i|$ the sorted absolute residual such that $|\tilde{r}_1| \leq \cdots \leq |\tilde{r}_n|$.

Given the scale of inlier noise $s$, the inliers can be dichotomized from outliers using the following equation:

$$|r_i/s| < \mathrm{E}, \tag{2}$$

where E is a threshold (98 percent of inliers of a Gaussian distribution is included when E is set to 2.5).

The MEDian (MED) and the Median Absolute Deviation (MAD) are two popular scale estimators. These scale estimators can be written as

$$\hat{s}_{\mathrm{MED}} := 1.4826\left(1 + \frac{5}{n-p}\right) med_i |r_i|, \tag{3}$$

$$\hat{s}_{\mathrm{MAD}} := 1.4826 med_i \{|r_i - med_i r_i|\}. \tag{4}$$

A generalization of the MED and MAD is the Kth Ordered Scale Estimator (KOSE) [13]:

$$\hat{s}_K := |\tilde{r}_{\mathrm{K}}| \Big/ \Theta^{-1}\left(\frac{1}{2}(1+\kappa)\right), \tag{5}$$

where $\Theta^{-1}(\cdot)$ is the argument of the normal cumulative density function. $\kappa (\kappa := K/n)$ is the fraction of the whole data points having absolute residuals less or equal to the Kth ordered absolute residual.

MED, MAD, and KOSE assume that the data do not include pseudo-outliers and the residuals of the inliers are Gaussian distributed. When the data contain multiple structures, the scale estimates obtained by these estimators are biased and may even break down.

Lee et al. [13] proposed an Adaptive Least Kth order Squares (ALKS) algorithm to optimize the K value of KOSE by minimizing the variance of the normalized errors:

$$\begin{aligned} \zeta_{\mathrm{K}}^2 &:= \arg\ \min_K \frac{\hat{\sigma}_{\mathrm{K}}^2}{\hat{s}_{\mathrm{K}}^2} \\ &= \arg\ \min_K \Theta^{-1}\left(\frac{1}{2}(1+\kappa)\right) \sum_{i=1}^{\mathrm{K}} r_i^2/(\mathrm{K}-p)|\tilde{r}_{\mathrm{K}}|, \end{aligned} \tag{6}$$

where $\hat{\sigma}_{\mathrm{K}}$ is the variance of the first K smallest absolute residuals. $\hat{s}_{\mathrm{K}}$ is the estimated scale by (5).

Alternatively, the Modified Selective Statistical Estimator (MSSE) [31] finds the K value which satisfies

$$\frac{\hat{\sigma}_{\mathrm{K}+1}^2}{\hat{\sigma}_{\mathrm{K}}^2} > 1 - \frac{\mathrm{E}^2 - 1}{\mathrm{K} - p + 1}. \tag{7}$$

Another approach [2] uses a Two-Step Scale Estimator (TSSE), which uses a mean shift procedure and a mean shift valley procedure to dichotomize inliers/outliers, and then employs the MED scale estimator to estimate the scale.

All of ALKS, MSSE, and TSSE claim that they can tolerate more than 50 percent outliers. However, ALKS may not work accurately when the outlier percentage in data is small. ALKS and MSSE cannot handle data involving extreme outliers. TSSE is computationally slow.

To improve the accuracy of scale estimation, the authors of [32] employ an expectation maximization (EM) procedure to estimate the ratio of inliers and outliers from which the scale of inlier noises can be derived. However, there are

several limits in that approach: 1) The authors assume that outliers are randomly distributed within a certain range. This assumption is less appropriate for multiple structured data. 2) They initialize the inlier ratio value to be 50 percent of the data points. However, when the initial ratio is far from the true case, the EM algorithm may converge to a local peak, which leads to a breakdown in the approach.

### 2.2 The Iterative Kth Ordered Scale Estimator

As mentioned in Section 2.1, when data contain multiple structures, MED, MAD, and KOSE can either break down or be badly biased. The reasons include: 1) The *breakdown is caused when the median (for MED/MAD), or the Kth ordered absolute residual (for KOSE), belongs to outliers.* That is, when the median or the Kth ordered point is an outlier, it is impossible for MED/MAD or KOSE to derive the correct inlier scale from that point. 2) *MED, MAD, and KOSE are biased for multiple-structure data due to the bias in estimating* $\kappa$ *in* (5) when $n$ (the number of whole data) is used instead of the number of data belonging to the structure of interest.

We propose to iteratively optimize the $\kappa$ estimate (abbreviated to IKOSE). Let $|\tilde{r}_i^J|$ be the $i$th sorted absolute residual given the parameters of the $J$th structure ($\boldsymbol{\theta}^J$) and $n^J$ be the inlier number of the $J$th structure[1]; IKOSE for the $J$th structure can be written as follows:

$$\hat{s}_K^J := \left|\tilde{r}_{\mathrm{K}}^J\right| \Big/ \Theta^{-1}\left(\frac{1}{2}\left(1+\kappa^J\right)\right), \tag{8}$$

$$\kappa^J := \mathrm{K}/n^J. \tag{9}$$

The difference between (8) and (5) is in the way of estimating $\kappa$. In (8), only the data points belonging to the $J$th structure are used, while in (5), the whole data points are used. When data contain only one population without any outliers, (8) becomes (5). When data involve multiple structures and/or random outliers, (8) should be used instead of (5).

One crucial question is how to estimate $n^J$. This is a *chicken-and-egg* problem because when we know $n^J$, we can dichotomize inliers and outliers and estimate the inlier scale belonging to the $J$th structure; on the other hand, if we know the true inlier scale, we can estimate the value of $n^J$. To solve this problem, we propose using an iterative procedure to estimate $n^J$, described in Fig. 2.

**Theorem 1.** *The scale estimates* $\{\hat{s}_{\mathrm{K},t}^J\}_{t=1,2\ldots}$ *in IKOSE converge and* $\{\hat{s}_{\mathrm{K},t}^J\}_{t=1,2\ldots}$ *is monotonically decreasing with* $t$.

**Proof.** According to (8) and (9):

$$\hat{s}_{\mathrm{K},1}^J = \frac{\left|\tilde{r}_{\mathrm{K}}^J\right|}{\Theta^{-1}\left(\frac{1}{2}\left(1+\mathrm{K}/\hat{n}_1^J\right)\right)} \quad \text{and} \quad \hat{s}_{\mathrm{K},2}^J = \frac{\left|\tilde{r}_{\mathrm{K}}^J\right|}{\Theta^{-1}\left(\frac{1}{2}\left(1+\mathrm{K}/\hat{n}_2^J\right)\right)},$$

where $\hat{n}_1^J = n$ and $\hat{n}_2^J$ are the number of points satisfying $|\tilde{r}_i^J|/\hat{s}_{K,1}^J < \mathrm{E}$. As the K value and the Kth ordered absolute residual $|\tilde{r}_{\mathrm{K}}^J|$ are fixed in the iteration procedure and $\Theta^{-1}((1+\kappa)/2)$ is monotonically increasing with $\kappa$, we obtain $\hat{s}_{\mathrm{K},2}^J \le \hat{s}_{\mathrm{K},1}^J$ because $\hat{n}_2^J \le \hat{n}_1^J = n$. Similarly, we have $\hat{s}_{\mathrm{K},t}^J \le \hat{s}_{\mathrm{K},t-1}^J$. □

1. We assume the multiple structures (or model instances) are from the same class but with different parameters.

**Algorithm 1** $\hat{s}_{\mathrm{K}}^J = \mathbf{IKOSE}(\Re^J, \mathrm{K})$

1. **Input** a set of residuals $\Re^J := \{r_i^J\}_{i=1,\ldots,n}$ and the K value
2: **Compute** the Kth ordered absolute residual $\left|\tilde{r}_i^J\right|$, $\hat{\kappa}_{t(=1)}^J$ (=K/$n$), and $\hat{s}_{\mathrm{K},t(=1)}^J$ by (8).
3: **While** $\hat{s}_{\mathrm{K},t}^J \neq \hat{s}_{\mathrm{K},t\text{-}1}^J$ and $\kappa_t^J < 1$
4: **Estimate** the scale $\hat{s}_{\mathrm{K},t}^J$ at step $t$ by (8).
5: **Calculate** the number of inliers $\hat{n}_t^J$ satisfying (2).
6: **Compute** $\hat{\kappa}_t^J$ by (9).
7: **End while**
8: **Output** the scale estimate $\hat{s}_{\mathrm{K}}^J (= \hat{s}_{\mathrm{K},t}^J)$.

Fig. 2. The IKOSE algorithm.

When the estimated scale $\hat{s}_{\mathrm{K},t}^J$ reaches the true inlier scale, the number of the inliers classified by (2) will not change and IKOSE outputs $\hat{s}_{\mathrm{K},t}^J$. IKOSE is simple but efficient and highly robust to outliers, as demonstrated in Section 2.4.

### 2.3 The Choice of the K Value

One issue in IKOSE is how to choose K. When the data contain a high percentage of outliers, one should set K to be a small value to guarantee that the Kth ordered point belongs to inliers (avoid breakdown); yet, when the data include a high percentage of inliers, one should set the K value as large as possible so that more inliers are included to achieve better statistical efficiency. It is more important to avoid breakdown. For all the experiments in this paper, the inlier/outlier percentage of data is not given to IKOSE. To be safe (using a small K value), we fix the K value to be 10 percent of the whole data points, and we do not change the K value for any individual experiment.

### 2.4 Performance of IKOSE

We compare the performance of IKOSE with that of seven other robust scale estimators (MED, MAD, KOSE, ALKS, MSSE, EM, and TSSE). In this section, we assume the true parameters of a model are given so that we can concentrate on the scale estimation alone (we evaluate the performance of IKOSE without using any prior knowledge about true model parameters in Section 5).

In the following experiment, we generate a "two crossing line" data set and a "two parallel plane" data set, each of which includes a total number of 1,000 data points, distributed in the range $[0, 100]$. The standard variance of the inlier noise is 1.0. For the "two crossing line" data set, the number of the data points belonging to the first line (in red color) is gradually decreased from 900 to 100, while the number of the data points belonging to the second line (in blue color) is fixed at 100. Thus, the percentage of outliers to the first line is increased from 10 to 90 percent. For the "two parallel plane" data set, we keep the number of data points belonging to the two planes the same. We gradually decrease the number of data points belonging to the planes, while we increase the number of random outlier from 0 to 800. Thus, the percentage of outliers corresponding to the first line is increased from 50 to 90 percent.

We measure the scale estimation error using the following equation:

TABLE 1
Quantitative Evaluation of the Different Scale Estimators on the Two-Line Data Set and the Two-Plane Data Set

| | Two-line dataset | | | Two-plane dataset | | |
|---|---|---|---|---|---|---|
| | Mean | Std.Var. | Max. Err. | Mean | Std.Var. | Max. Err. |
| Median | 11.1 | 12.9 | 39.8 | 35.7 | 2.32 | 40.8 |
| MAD | 11.0 | 12.8 | 39.9 | 27.3 | 1.33 | 36.7 |
| KOSE | 1.70 | 1.88 | 11.5 | 3.02 | 2.04 | 11.0 |
| ALKS | 1.27 | 0.71 | 24.1 | 1.40 | 1.02 | 23.2 |
| MSSE | 0.31 | 1.30 | 38.1 | 0.38 | 1.31 | 32.3 |
| EM | 0.33 | 0.52 | 26.6 | 1.57 | 2.70 | 87.4 |
| TSSE | 0.13 | 0.25 | 12.9 | 0.15 | 0.17 | 5.75 |
| IKOSE | 0.11 | 0.05 | 0.88 | 0.12 | 0.06 | 0.88 |

$$\Lambda_s(\hat{s}, s_{\mathrm{T}}) = \max\left(\frac{\hat{s}}{s_{\mathrm{T}}}, \frac{s_{\mathrm{T}}}{\hat{s}}\right), \tag{10}$$

where $\hat{s}$ is the estimated scale and $s_T$ is the true scale.

We repeat the experiments 50 times and show both the average results (shown in Figs. 3c and 3d) and the maximum error among the results (in Figs. 3e and 3f). Table 1 shows the mean, the standard variance, and the maximum scale estimation errors of the results. The K value is fixed to 100.

As shown in Fig. 3 and Table 1, IKOSE achieves the best performance among the eight competing scale estimators. When the outlier percentage is more than 50 percent, MED and MAD begin to break down. ALKS obtains less accurate results than the other approaches when the outlier percentage is less than 50 percent, but better results than MED/MAD for data involving more than 50 percent outliers and better than KOSE when the outlier percentage is more than 60 percent. MSSE and TSSE achieve better results than EM but less accurate results than IKOSE. IKOSE works well even when the data involve 90 percent outliers. Of course, when data involve more 90 percent outliers, IKOSE begins to break down. The reason is because, in that case, the Kth ordered point is an outlier from which IKOSE cannot correctly recover the inlier scale.

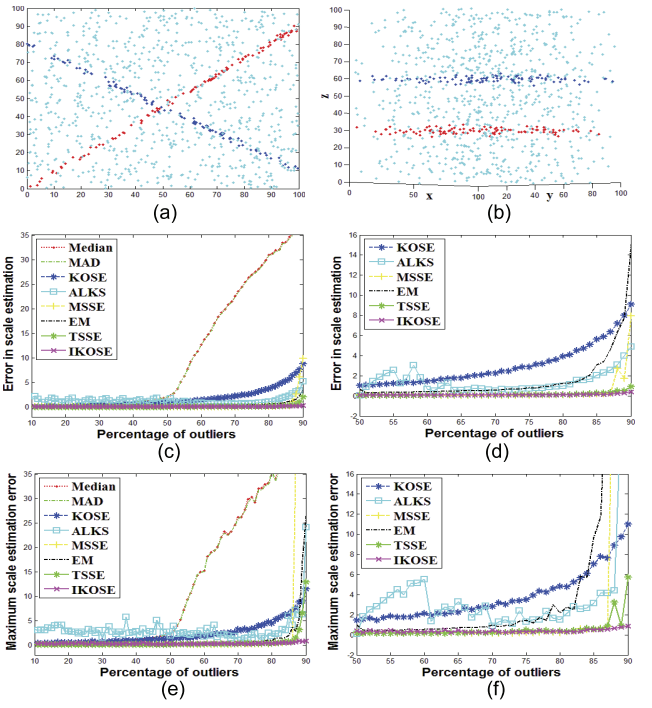



Fig. 3. (a) and (b) Two snapshots, respectively, showing the "two-line" data and the "two-plane" data with 90 percent outliers. (c) and (d) The error plots of the scale estimation versus the outlier percentages, respectively. (e) and (f) The maximum errors in scale estimation obtained by the eight competing estimators. In (d) and (f), we do not show the results of the Median and the MAD scale estimators, which cannot deal with more than 50 percent outliers and totally break down.

## 3 The AKSWH Approach

In this section, based on IKOSE (Section 2), we propose a novel robust approach which can simultaneously estimate not only the parameters of and the scales of model instances in data, but also the number of model instances in the data. In contrast to the experiment in Section 2.4, the parameters of model instances are not given as prior information in the following experiments, but are estimated from the data.

### 3.1 The Methodology

In our framework, we first sample a number of $p$-subsets *(a p-subset is the minimum number of data points which are required to calculate the model parameters, e.g., p is equal to 2 for line fitting and 3 for circle fitting).* The number of samples should be sufficiently large such that, with high probability, at least one all-inlier $p$-subset is selected for each structure in the data. Both random sampling techniques (such as [9]) and adaptive sampling techniques (e.g., [33], [34]) can be employed to sample $p$-subsets.

Having generated a set of $p$-subsets, we compute the model hypotheses using the $p$-subsets. We assign each hypothesis a weight. Let $\hat{\boldsymbol{\theta}}_i = (\hat{\theta}_{i,1}, \ldots \hat{\theta}_{i,p})$ be the $i$th hypothesis and $\mathcal{P}_i := (\hat{\boldsymbol{\theta}}_i, \hat{w}_i) = (\hat{\theta}_{i,1}, \ldots \hat{\theta}_{i,p}, \hat{w}_i)'$ be a weighted hypothesis corresponding to $\hat{\boldsymbol{\theta}}_i$; we represent the model parameter estimates with a set of weighted hypotheses as follows:

$$\mathcal{P} := \{\mathcal{P}_i\}_{i=1,2,\ldots} = \{(\hat{\boldsymbol{\theta}}_i, \hat{w}_i)\}_{i=1,2,\ldots}, \tag{11}$$

where $\hat{w}_i$ is the weight of the hypothesis $\mathcal{P}_i$ and $\hat{w}_i > 0$—Section 3.2. The assumption is that every structure of interest has at least one (locally) strong hypothesis associated with it—and that these (among other clean hypotheses) will survive the thresholding step—Section 3.3. The surviving hypotheses are clustered and single representatives of each cluster are selected—Sections 3.4 and 3.5. The net effect is intended so as to choose one hypothesis for each cluster: Let $\mathcal{P}^J := \{\mathcal{P}_i^J\} := \{(\hat{\boldsymbol{\theta}}_i^J, \hat{w}_i^J)\}_{i=1,2,\ldots}$ be the weighted hypotheses associated with the $J$th structure and $\hat{\mathcal{P}}^J$ be the estimated hypothesis for the $J$th structure with the parameter estimate $\hat{\boldsymbol{\theta}}_i^J$ and weight $\hat{w}_i^J$; AKSWH can be written as

$$\hat{\mathcal{P}} = \left\{ \hat{\mathcal{P}}^J \,\middle|\, \hat{\mathcal{P}}^J := \underset{\{\mathcal{P}_i^J\}}{\arg\max}\, \hat{w}_i \right\}. \tag{12}$$

### 3.2 Weighting Hypotheses

In the above, it is important to effectively weight each hypothesis as it is the key measure of the "goodness" of a hypothesis. Ideally, when a $p$-subset only contains inliers belonging to one structure, the weight of the corresponding hypothesis should be as high as possible. Otherwise,

the weight of that hypothesis should be as close to zero as possible.

We base our measure on nonparametric kernel density estimate techniques [35], [36]. Given a set of residuals $\{r_i(\hat{\boldsymbol{\theta}}_j)\}_{i=1,\ldots,n}, (r_i(\hat{\boldsymbol{\theta}}_j) \in \mathbb{R}^1)$ which are derived from the $j$th hypothesis $\hat{\boldsymbol{\theta}}_j$, the variable bandwidth kernel density estimate at $r$ is written as follows:

$$\hat{f}_{\mathbf{KN},\hat{\boldsymbol{\theta}}_j}(r) := \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h(\hat{\boldsymbol{\theta}}_j)} \mathbf{KN}\left(\frac{r - r_i(\hat{\boldsymbol{\theta}}_j)}{h(\hat{\boldsymbol{\theta}}_j)}\right), \tag{13}$$

where $\mathbf{KN}(\cdot)$and $h(\hat{\boldsymbol{\theta}}_j)$ are the kernel and the bandwidth.

The popular Epanechnikov kernel $\mathbf{KN}_E(r)$ [35] is used in our approach and it is defined as

$$\mathbf{KN}_E(r) := \begin{cases} \frac{3}{4}(1 - \|r\|^2) & \|r\| \leq 1 \\ 0 & \|r\| > 1, \end{cases} \quad \text{with}$$
$$\mathbf{kn}_E(r) := \begin{cases} 1 - r & 0 \leq r \leq 1 \\ 0 & r > 1, \end{cases} \tag{14}$$

where $\mathbf{kn}_E(r)$ is the profile of the kernel $\mathbf{KN}_E(r)$.

The bandwidth can be estimated using [35]

$$\hat{h}(\hat{\boldsymbol{\theta}}_j) = \left[\frac{243\int_{-1}^{1}\mathbf{KN}(r)^2 dr}{35n\int_{-1}^{1} r^2\mathbf{KN}(r)dr}\right]^{1/5} \hat{s}_K(\hat{\boldsymbol{\theta}}_j), \tag{15}$$

where $\hat{s}_K(\hat{\boldsymbol{\theta}}_j)$ is the scale estimate obtained by IKOSE in Section 2.2. The value of $\hat{s}_K(\hat{\boldsymbol{\theta}}_j)$ and the bandwidth $\hat{h}(\hat{\boldsymbol{\theta}}_j)$ is calculated for each $p$-subset.

Like [17], [29], we score a hypothesis using the density estimate at the origin ($\mathbf{O}$), but we refine to further suppress hypotheses that produce large scale estimates through

$$\hat{w}_j := \frac{\hat{f}_{\mathbf{KN},\hat{\boldsymbol{\theta}}_j}(\mathbf{O})}{\hat{s}_K(\hat{\boldsymbol{\theta}}_j)} = \frac{1}{n}\sum_{i=1}^{n} \frac{\mathbf{KN}(r_i(\hat{\boldsymbol{\theta}}_j)/h(\hat{\boldsymbol{\theta}}_j))}{\hat{s}_K(\hat{\boldsymbol{\theta}}_j)h(\hat{\boldsymbol{\theta}}_j)}. \tag{16}$$

Consider for the moment the case where the inlier scale is fixed (to some common value), then the bandwidth $h$ is also a constant value for all valid hypotheses. Thus, (16) can be rewritten as

$$\hat{w}_j \propto \frac{1}{n}\sum_{i=1}^{n} \mathbf{KN}(r_i(\hat{\boldsymbol{\theta}}_j)/h). \tag{17}$$

Note this is just a summation of values—one for each residual small enough to lie within the support of the kernel functions overlapping the origin. So it is like a weighted count of the number of residuals in a certain fixed width (by the data independent width of the kernel function). And it is like a generalization of the RANSAC criteria—sharing the fact that it would score each hypothesis only by a "globally defined" tolerance level and the number of residuals lying in that tolerance. Of course, this RANSAC score is implicitly related to the scale of noise that is consistent with that hypothesis—the larger the scale, the fewer samples one would expect in the tolerance band. In contrast, our criterion is deliberately more sharply dependent on a data driven scale. In effect, the band itself is being refined in a data dependent way. We will evaluate the two weighting functions ((16) and (17)) in Sections 4 and 5.

### 3.3 Selecting Significant Hypotheses

After assigning to each hypothesis a weight, the next question is how to select significant hypotheses (with high weight scores) while ignoring weak hypotheses (corresponding to bad $p$-subsets). One solution is to manually set a threshold. However, this is not desirable because it is not data driven.

Data driven thresholding approaches have been well surveyed in [37]. We use an information-theoretic approach similar to [38], which conducts thresholding as a process to remove redundancies in the data, while retaining as much information content as possible. As we show in the following, the method is effective and computationally efficient.

Given a set of hypotheses $\mathcal{P}(= \{\mathcal{P}_i\}_{i=1,2,\ldots})$, and the corresponding weights $\mathcal{W} := \{\hat{w}_i\}_{i=1,\ldots,n}$ calculated by (16) or (17), we define

$$\Psi_j := \max(\mathcal{W}) - \hat{w}_j^2, \tag{18}$$

where $\Psi_j \geq 0$. This means $\Psi_j$ is a quantity that is proportional to the gap between the weight of the $j$th hypothesis and the maximum weight. By regarding weights as estimates of the density at locations $\mathcal{P}$ $(= \{\mathcal{P}_i\}_{i=1,2,\ldots})$ in the parameter space, the prior probability of component $\Psi_j$ is written as

$$\mathrm{p}(\Psi_i) := \Psi_i \Big/ \sum_{j=1}^{n} \Psi_j. \tag{19}$$

The uncertainty of $\Psi_i$ is then defined as

$$H(\Psi_i) := -p(\Psi_i)\log p(\Psi_i). \tag{20}$$

The entropy of the set $\{\Psi_i\}_{i=1,\ldots,n}$ can then be calculated as the sum of the uncertainty of the hypothesis set:

$$H := \sum_{i=1}^{n} H(\Psi_i). \tag{21}$$

The quantity $\mathbf{H}$ essentially quantifies the "information content" available in the set of hypotheses $\mathcal{P}$ $(= \{\mathcal{P}_i\}_{i=1,2,\ldots})$.

Significant hypotheses $\mathcal{P}^*$ are then selected from the hypotheses $\mathcal{P}$ by selecting hypotheses satisfying the following condition:

$$P^* = \{P_i | H + \log p(\Psi_i) < 0\}. \tag{22}$$

In other words, we keep the hypotheses which contribute a significant proportion of the information content while rejecting uninformative hypotheses. With (22), weak hypotheses are rejected while effective hypotheses are identified (see Fig. 1d for an example).

### 3.4 Clustering Hypotheses

The surviving significant hypotheses are assumed to form clusters corresponding to the underlying modes of the model probability density distributions in parameter space. We cluster the significant hypotheses based on the Jaccard distance (referred to as J-distance), which measures to what extent the identified inliers are shared between two hypotheses. The hypotheses from the same structure share a high percentage of the identified inliers. The hypotheses with low J-distance scores are clustered together while

TABLE 2
The Fitting Errors in Parameter Estimation
(and the CPU Time in Seconds)

| | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 |
|---|---|---|---|---|---|---|---|---|
| 3 lines | 1.28 (0.80) | 1.21 (6.72) | 23.7 (7.56) | 25.1 (8.12) | 1.17 (25.2) | 0.99 (177) | 1.18 (15.4) | 1.13 (4.52) |
| 4 lines | 13.3 (0.84) | 1.16 (8.10) | 24.2 (5.87) | 48.5 (7.31) | 8.67 (19.5) | 1.06 (223) | 1.16 (17.7) | 1.12 (4.02) |
| 5 lines | 19.9 (0.79) | 1.40 (8.46) | 7.35 (7.26) | 116 (7.86) | 10.9 (23.7) | 9.27 (116) | 1.25 (14.2) | 1.29 (4.12) |
| 6 lines | 15.7 (1.21) | 1.21 (11.0) | 47.0 (5.53) | 178 (9.06) | 27.6 (23.2) | 1.20 (685) | 1.17 (20.2) | 1.15 (3.43) |

*M1-RHT; M2-ASKC; M3-MS; M4-RHA; M5-J-Linkage; M6-KF; M7-AKSWH1; M8-AKSWH2. We run the approaches on a laptop with an I7 2.66 GHz CPU on a Windows 7 platform.*

hypotheses with high J-distance scores stay independent of each other.

Given a set of residuals $\{r_i(\hat{\boldsymbol{\theta}}_j)\}_{i=1,\ldots,n}$ derived from $\hat{\boldsymbol{\theta}}_j$, we can dichotomize inliers and outliers with an inlier scale estimate $\hat{s}$ obtained by IKOSE or specified by the user. The consensus set of residuals $\{r_i(\hat{\boldsymbol{\theta}}_j)\}_{i=1,\ldots,n}$ is formulated as

$$\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_j) := \{\boldsymbol{\mathcal{L}}(r_i(\hat{\boldsymbol{\theta}}_j))\}_{i=1,\ldots,n}, \text{where } \boldsymbol{\mathcal{L}}(r_i) = \begin{cases} 1 & \text{If } |r_i| \leq \mathrm{E}\hat{s} \\ 0 & \text{Otherwise.} \end{cases} \tag{23}$$

The J-distance between two consensus sets $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_j)$ and $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_k)$ is given by

$$\boldsymbol{\mathcal{J}}(\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}_j), \hat{\boldsymbol{\mathcal{C}}}(\hat{\boldsymbol{\theta}}_k)) := 1 - \frac{\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}_j) \cap \hat{\boldsymbol{\mathcal{C}}}(\hat{\boldsymbol{\theta}}_k)}{\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}_j) \cup \hat{\boldsymbol{\mathcal{C}}}(\hat{\boldsymbol{\theta}}_k)}. \tag{24}$$

When $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_j)$ and $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_k)$ are identical, their J-distance is zero; when $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_j)$ and $\boldsymbol{\mathcal{C}}(\hat{\boldsymbol{\theta}}_k)$ are totally different, their J-distance is 1.0.

We note that [25] also employs the J-distance as a clustering criterion. However, our use of J-distance is significantly different: 1) A consensus set in [25] is a set of classifications of the parameter hypothesis with respect to one data point. The J-distance must be calculated for all possible pairs of the data points, i.e., $n(n-1)/2$ pairs. The consensus set in our approach is the inlier/outlier binary classification of all data points with respect to one parameter hypothesis. With an effective weighting function, we can select significant hypotheses by pruning weak hypotheses and calculate the J-distances only for the significant hypotheses, by which the computational efficiency can be greatly improved. In our case, we only compute the J-distance for $\mathrm{M}^\dagger(\mathrm{M}^\dagger-1)/2$ significant modes (where $\mathrm{M}^\dagger$ is the number of significant hypotheses, which can be much smaller than the number of data points $n$). Thus, our approach is much faster than the J-linkage approach (see Table 2, and Tables 4, 5, and 6 in Section 5). 2) Toldo and Fusiello [25] cluster the pairs of data points, and select as the inliers the data points belonging to one cluster when the number of the data points is larger than a user-specified threshold. The parameters of the mode instance are then calculated using all the classified inliers. In contrast, we use a more straightforward way that clusters the pairs of hypotheses in a parameter space and we do not use any threshold to determine the number of clusters.

After we calculate the J-distances between all possible pairs of the selected significant weighted hypotheses $\mathcal{P}^*$, we apply the entropy thresholding approach (described in Section 3.3) to the J-distance values to obtain an adaptive cut-off value. This cut-off value is data driven rather than user specified. The pairs of hypotheses whose J-distance values are less than the cut-off value are clustered together through the traditional linkage algorithm [39], while the pairs of hypotheses whose J-distance values are larger than the cut-off value are left independent.

However, one structure may correspond to more than one cluster, especially when the inlier noise of the structure is large. To solve this problem, we need to effectively fuse the clusters belonging to the same structure (equivalently, this means elimination of duplicate representations of modes).

### 3.5 Fusing Clusters with Mutual Information Theory

To solve the above problem, we propose using Mutual Information Theory (MIT) [40]. This is a frequently used technique in various contexts: Yang and Zwolinski [41] employ MIT to select an optimal set of components in mixture models for accurately estimating an underlying probability density function. Lee et al. [42] extend the approach of Yang and Zwolinski [41] to speaker identification. In this paper, we propose to apply MIT to multiple-structure model fitting.

Each cluster, obtained by the approach described in Section 3.4, is represented by the most significant hypothesis with the highest weight score among all the hypotheses belonging to that cluster. Let $\boldsymbol{\mathcal{P}}^\dagger = \{\mathcal{P}_i^\dagger\}_{i=1,2,\ldots}$ be the most significant hypotheses in the clusters. Each hypothesis in $\boldsymbol{\mathcal{P}}^\dagger$corresponds to one model candidate. When two hypotheses in $\boldsymbol{\mathcal{P}}^\dagger$correspond to one structure (i.e., the corresponding $p$-subsets are sampled from the same structure), the information shared by the two hypotheses is large. On the other hand, when two hypotheses are from different structures, the mutual information between them is small.

Let $\mathrm{p}(\boldsymbol{\theta}_i^\dagger)$ denote the probability of $\boldsymbol{\theta}_i^\dagger$. As in (21), the uncertainty of the hypothesis $P_i^\dagger$ is defined as

$$H(\mathcal{P}_i^\dagger) \equiv H(\boldsymbol{\theta}_i^\dagger) := -\mathrm{p}(\boldsymbol{\theta}_i^\dagger)\log \mathrm{p}(\boldsymbol{\theta}_i^\dagger). \tag{25}$$

An estimate of the mutual information between two hypotheses $\mathcal{P}_i^\dagger$ and $\mathcal{P}_j^\dagger$ in $\mathcal{P}^\dagger$ can be written as

$$\mathcal{M}(\mathcal{P}_i^\dagger, \mathcal{P}_j^\dagger) \equiv \mathcal{M}(\boldsymbol{\theta}_i^\dagger, \boldsymbol{\theta}_j^\dagger) := \mathrm{p}(\boldsymbol{\theta}_i^\dagger, \boldsymbol{\theta}_j^\dagger)\log\frac{\mathrm{p}(\boldsymbol{\theta}_i^\dagger, \boldsymbol{\theta}_j^\dagger)}{\mathrm{p}(\boldsymbol{\theta}_i^\dagger)\mathrm{p}(\boldsymbol{\theta}_j^\dagger)}, \tag{26}$$

where

$$\frac{\mathrm{p}(\boldsymbol{\theta}_i^\dagger, \boldsymbol{\theta}_j^\dagger)}{\mathrm{p}(\boldsymbol{\theta}_i^\dagger)\mathrm{p}(\boldsymbol{\theta}_j^\dagger)} = \frac{n\sum_{l=1}^n \mathrm{p}(\boldsymbol{x}_l|\boldsymbol{\theta}_i^\dagger)\mathrm{p}(x_l|\boldsymbol{\theta}_j^\dagger)}{\sum_{l=1}^n \mathrm{p}(\boldsymbol{x}_l|\boldsymbol{\theta}_i^\dagger)\sum_{l=1}^n \mathrm{p}(\boldsymbol{x}_l|\boldsymbol{\theta}_j^\dagger)}. \tag{27}$$

Here, $\mathrm{p}(x_l|\boldsymbol{\theta})$ is the conditional probability of data $\mathbf{x}_i$ belonging to model $\boldsymbol{\theta}$. Recalling (1), and assuming a Gaussian inlier noise model, $\mathrm{p}(x_l|\boldsymbol{\theta})$ can be written as

$$\mathrm{p}(x_l|\boldsymbol{\theta}) \propto \frac{1}{s}\exp\left(-\frac{F(\boldsymbol{x}_i, \boldsymbol{\theta})^2}{2s^2}\right), \tag{28}$$

**Algorithm 2** $\mathcal{P}^{\Delta} = \textbf{Fusion}(\mathbf{X}, \mathcal{P}^{\dagger})$

**1: Input** the data points X and a set of hypotheses $\mathcal{P}^{\dagger}(=\{\mathcal{P}_i^{\dagger}\}_{i=1,2,...})$
2: **Sort** hypotheses $\mathcal{P}^{\dagger}$ according to the weights of $\mathcal{P}^{\dagger}$ to obtain $\tilde{\mathcal{P}}^{\dagger} = \{\mathcal{P}_{\lambda(1)}^{\dagger}, \mathcal{P}_{\lambda(2)}^{\dagger}, ...\}$ satisfying $\hat{w}_{\lambda(1)} \leq \hat{w}_{\lambda(2)} \leq \cdots$.
3: **Compute** the probability of **X** belonging to the two hypotheses $\mathcal{P}_{\lambda(i)}^{\dagger}$ and $\mathcal{P}_{\lambda(j)}^{\dagger}$ chosen from $\tilde{\mathcal{P}}^{\dagger}$ by (28)
4: **Compute** the mutual information $\mathcal{M}(\mathcal{P}_{\lambda(i)}^{\dagger}, \mathcal{P}_{\lambda(j)}^{\dagger})$ by (26)
5: **Repeat** step 3 and step 4 until the mutual information of all possible pairs of hypotheses in $\tilde{\mathcal{P}}^{\dagger}$ is calculated
6: **For** $i$= 1 to Size($\tilde{\mathcal{P}}^{\dagger}$)-1
7: **If** any $\mathcal{M}(\mathcal{P}_{\lambda(i)}^{\dagger}, \mathcal{P}_{\lambda(q)}^{\dagger}) > 0$ for Size($\tilde{\mathcal{P}}^{\dagger}$) $\geq q > i$ **then**
8: **Change** the cluster label of $\mathcal{P}_{\lambda(i)}^{\dagger}$ to that of $\mathcal{P}_{\lambda(q)}^{\dagger}$
9: **Remove** $\mathcal{P}_{\lambda(i)}^{\dagger}$ from $\tilde{\mathcal{P}}^{\dagger}$
10: **End if**
11: **End for**
12: **Output** the remaining hypotheses in $\tilde{\mathcal{P}}^{\dagger}$ $(=\mathcal{P}^{\Delta})$

Fig. 4. The Fusion algorithm.

where $s$ is the inlier scale corresponding to the hypotheses $\boldsymbol{\theta}$. Therefore, (26) analyzes the mutual information between two hypotheses by measuring the degree of similarity between the inlier distributions centered, respectively, on the two hypotheses.

When the mutual information between $\mathcal{P}_i^{\dagger}$ and $\mathcal{P}_j^{\dagger}$ (i.e., $\mathcal{M}(\mathcal{P}_i^{\dagger}, \mathcal{P}_j^{\dagger})$) is larger than zero, $\mathcal{P}_i^{\dagger}$ and $\mathcal{P}_j^{\dagger}$ are statistically dependent; otherwise, $\mathcal{P}_i^{\dagger}$ and $\mathcal{P}_j^{\dagger}$ are statistically independent. Yang and Zwolinski [41] employ MIT to prune mixture components in Gaussian mixture models. In contrast, we apply MIT in the multiple-structure model fitting. We are pruning/fusing related descriptions of general data (e.g., identifying "similar" lines, circles, planes, homographies, or a fundamental matrices—"similar" in that they "explain" the same portion of data).

The fusion algorithm is summarized in Fig. 4. The function Size(·) returns the number of hypotheses. The weight scores of hypotheses indicate the goodness of model candidates. We first order the hypotheses according to their weights: from the weakest to the most significant weight. When we find a pair of hypotheses which are statistically dependent on each other, we fuse the hypothesis with a lower weight score to the hypothesis with a higher weight score.

## 4 The Complete Algorithm of AKSWH

With all the ingredients developed in the previous sections, we now propose the complete AKSWH algorithm, summarized in Fig. 5. After the estimation, one can use all the indentified inliers to refine each estimated hypothesis.

Because some robust approaches such as RANSAC, MSAC, HT, RHT, MS, and J-linkage require the user to specify the inlier scale or the scale-related bin size or bandwidth and it is also interesting to evaluate the effectiveness of the weighting functions in (16) and (17), we provide an alternative version of AKSWH with a user-input

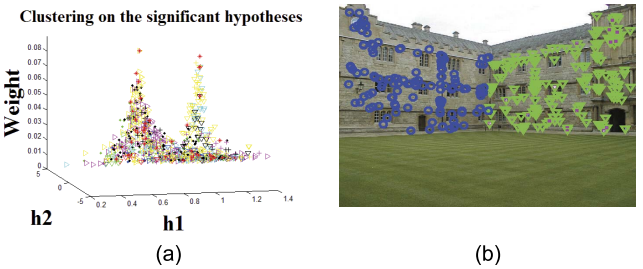
**Algorithm 3** $[\mathcal{P}^{\Delta}, \mathbb{Cs}] = \textbf{AKSWH}(\mathbf{X}, \mathrm{K})$

**Input** data points **X** and the K value
1: **Estimate** the parameter of a model hypothesis $\hat{\boldsymbol{\theta}}_j$ using a $p$-subset chosen from **X**
2: **Derive** the residual $\Re(\hat{\boldsymbol{\theta}}_j) = \{r_i(\hat{\boldsymbol{\theta}}_j)\}_{i=1,...,n}$ of the data points
3: **Estimate** the scale $\hat{s}_K^j = \textbf{IKOSE}(\Re(\hat{\boldsymbol{\theta}}_j), \mathrm{K})$ by Alg.1
4: **Compute** the weight of the hypothesis $\hat{w}_j$ by (16)
5: **Repeat** step 1 to step 4 many times to obtain a set of weighted hypotheses $\mathcal{P} = \{\mathcal{P}_i\}_{i=1,2,...} = \{(\hat{\boldsymbol{\theta}}_i, \hat{w}_i)\}_{i=1,2,...}$
6: **Select** significant hypotheses $\mathcal{P}^{*}(\subset \mathcal{P})$ from $\mathcal{P}$ by the approach introduced in Sec. 3.3
7: **Run** the clustering procedure introduced in Sec. 3.4 on $\mathcal{P}^{*}$ to obtain clusters and select the most significant hypothesis in each cluster according to the weights of hypotheses $\mathcal{P}^{\dagger} = \{\mathcal{P}_i^{\dagger}\}_{i=1,2,...} \subset \mathcal{P}^{*}$
8: **Fuse** the hypotheses $\mathcal{P}^{\dagger}$ to obtain $\mathcal{P}^{\Delta} = \{\mathcal{P}_i^{\Delta}\}_{i=1,2,...}$ by Alg. 2
9: **Derive** the inlier/outlier dichotomy using $\mathcal{P}^{\Delta}$
10: **Output** the hypotheses $\mathcal{P}^{\Delta}$ and the inlier/outlier dichotomies $\mathbb{Cs}$ $(=\{\mathbb{C}(\mathcal{P}_i^{\Delta})\}_{i=1,2,...})$ corresponding to each of the hypotheses in $\mathcal{P}^{\Delta}$

Fig. 5. The complete AKSWH algorithm.

inlier scale (here, we refer to our approach with a fixed inlier scale as AKSWH1 and refer to AKSWH with the adaptive scale estimation approach as AKSWH2). For AKSWH1, we input the inlier scale *instead of the K value*, and *we do not run step 3*—instead simply passing the user supplied tolerance as the scale estimate for all hypotheses and using this to determine the (single global) bandwidth. All the remaining steps are the same as AKSWH2.

Figs. 1 and 6, respectively, illustrate some results of AKSWH2 and AKSWH1 (the results are similar; essentially only the computational time taken is different) in planar homography detection using "Merton College II" (referred to as "MC2"). We employ the *direct linear transform* (*DLT*) *algorithm* [43, Algorithm 4.2, p. 109] to compute the homography hypotheses from a set (5,000 in size) of randomly chosen $p$-subsets (here, a $p$-subset is four pairs of point correspondences in two images). For each estimated homography hypothesis, we use the *symmetric transfer error model* [43, p. 94,] to compute the residuals. For AKSWH2, there are 384 significant hypotheses selected from the



(a) (b)

Fig. 6. Some results obtained by AKSWH1. (a) Oversegmented clusters after the clustering step using a fixed inlier scale. (b) The final obtained segmentation result obtained by AKSWH1.

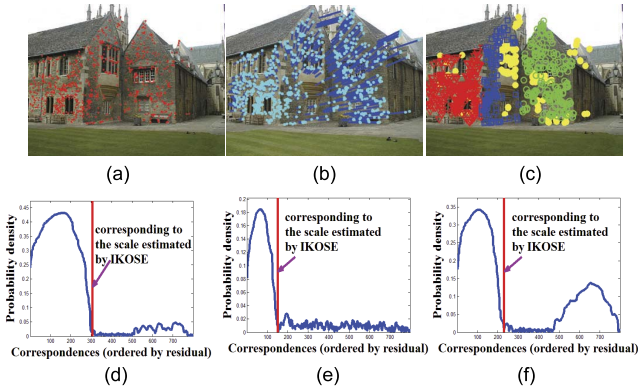



Fig. 7. Segmentation results for the "Merton College III" (referred to as "MC3") image pair obtained by AKSWH2. (a) and (b) The input image pair with SIFT feature [1] points and corresponding matches superimposed, respectively. (c) The segmentation results (the yellow dots are the detected outliers). (d) to (f) The inlier/outlier dichotomy obtained by IKOSE (corresponding to the left, middle, and right plane, respectively).

5,000 hypotheses, and 19 clusters are obtained at the clustering step. In comparison, the number of the selected significant hypotheses and the number of clusters before the fusion step obtained by AKSWH1 are, respectively, 806 and 45. Although the segmentation results obtained by AKSWH1 are the same as those obtained by AKSWH2, AKSWH2 is more effective in selecting significant hypotheses due to the more effective weighting function.

For the computational time used by AKSWH2, around 74 percent of the whole time is used to generate 5,000 hypothesis candidates from the randomly sampled $p$-subsets and to compute the weights of the hypotheses (step 1 to step 5 in Fig. 5), and about 25 percent of the whole time is used to cluster the significant hypotheses (step 7 in Fig. 5). The significant hypothesis selection (step 6) and the fusion of the clusters (step 8) take less than 1 percent of the whole time. In contrast, AKSWH1 uses about 26 percent of the computational time in generating hypothesis candidates and computing the weights of the hypotheses, and 73 percent of the whole time in clustering the significant hypotheses (because more hypotheses are selected for clustering when the less effective weighting function is used). Thus, we can see that effectively selecting significant hypotheses (step 6 in Fig. 5) will significantly affect the computational speed of AKSWH1/2. We will give more comparisons between AKSWH1/2 in Section 5.

Fig. 7 shows the homography-based segmentation results obtained by AKSWH2. AKSWH2 automatically selects 268 significant hypotheses from 10,000 hypotheses and correctly detects three planar homographies. Figs. 7d, 7e, and 7f illustrate the probability density distributions of the ordered (by the absolute residual value) correspondences for the selected model instance and the adaptive inlier/outlier dichotomy estimated by IKOSE (in AKSWH2). We can see that IKOSE has correctly detected the inlier-outlier dichotomy boundary.

## 5 Experiments

We compare AKSWH with RHT [8], ASKC [17], [29], MS [20], [21], RHA [24], J-linkage [25], and KF [26]. The choice of the competitors was informed by the suggestions of the reviewers and by the following reasons: 1) We choose J-linkage (which uses the Jaccard distance as a criterion to cluster data points) and ASKC (which is also developed based on kernel density estimate) because they are the approaches that are most related to the proposed approach. 2) We choose RHT and MS because both approaches work in the parameter space. 3) RHA and KF estimate the parameters of model instances simultaneously and claim to have the ability to estimate the number of structures. Thus, we also chose RHA and KF.

As indicated earlier—we test two versions of our approach: AKSWH1 with a fixed inlier scale and AKSWH2 with an adaptive inlier scale estimate. AKSWH1 is not intended to be a proposed method—we use it only to evaluate the relative effects of the more sophisticated weighting scheme introduced in Section 3.2. Nonetheless, we find that it performs remarkably well—albeit at slower speed and slightly less accuracy, generally.

In running RHT, MS, J-linkage, and AKSWH1, we manually specify the inlier scale for these approaches. We specify the number of model instances for ASKC and the minimum inlier number (MinN) (which has influence on determining the number of model instances) for RHT. MS, RHA, J-linkage, and KF need some user-specified thresholds, which have significant influence on determining the estimate of the model instance number (see Section 1 for more detail). Thus, we adjust the threshold values so that the approaches of RHT, MS, RHA, J-linkage, and KF output the correct model instance number. Neither the number of model instances nor the true scale is required for AKSWH2. The value of K in AKSWH1/2 is fixed to be 10 percent of the whole data points for all the following experiments and we do not adjust any other parameter or threshold for AKSWH2 (note: we specify the inlier scale for AKSWH1 for comparison purposes only).

Although there are other sampling techniques (e.g., [33], [34]), we employ random sampling for all the approaches, because: 1) It is easy to implement and it does not require extra thresholds (e.g., the threshold in the exponent function in [25], or the matching scores in [33], [34]); 2) it is the most widely used approach to generate hypothesis candidates.

To measure the fitting errors, we first find the corresponding relationship between the estimated model instances and the true model instances (for real data/image, we manually obtain the true parameters of model instances and the true inliers to the model instance). Then, we fit the true inliers to the corresponding model instances estimated by the approaches, and fit the classified inliers by the approaches to the true parameters of the corresponding model instances. For RHT, MS, RHA, J-linkage, and AKSWH1, the inliers belonging to a model instance are classified using the given ground truth inlier scale while for ASKC, KF, and AKSWH2, the inlier scales are adaptively estimated. There are 5,000 $p$-subsets generated for each data set used in Sections 5.1, 5.2.1, 5.2.2, and 5.2.3, 10,000 $p$-subsets for each data set used in Section 5.2.4, and 20,000 for each data set used in Section 5.2.5.

### 5.1 Simulations with Synthetic Data

We evaluate the eight approaches on line fitting using four sets of synthetic data (see Fig. 8). The residuals are calculated by the standard regression definition. We

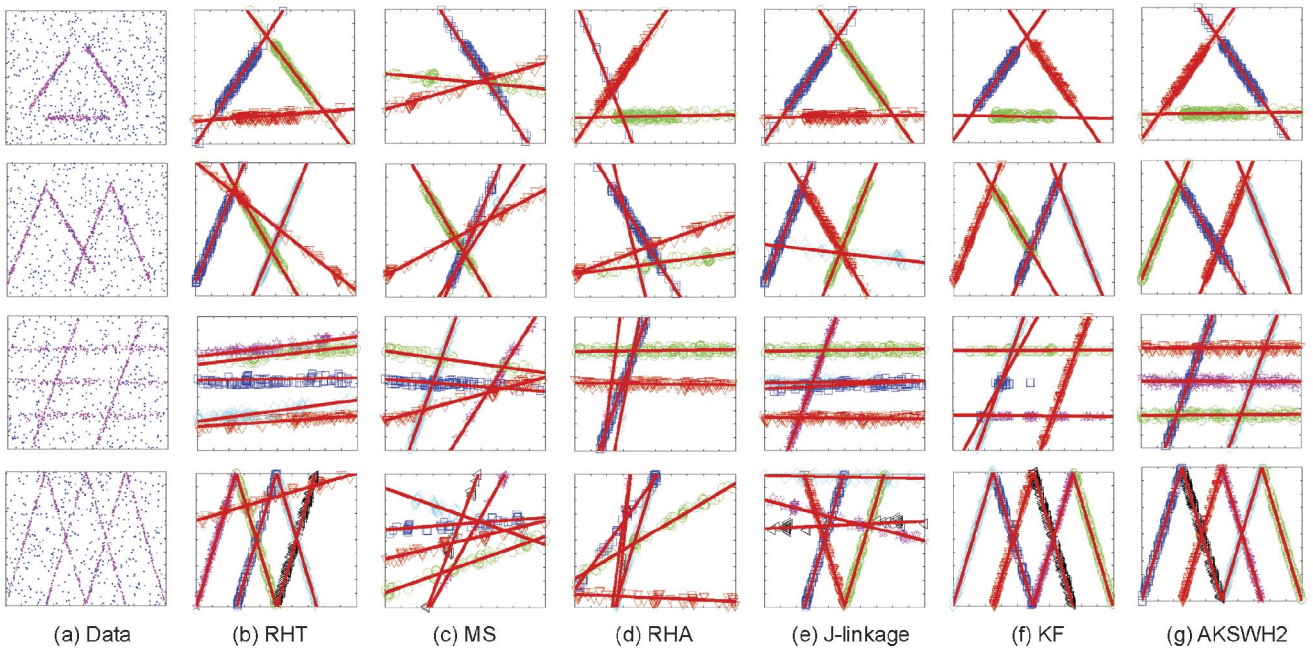



Fig. 8. Examples for line fitting and segmentation. The first to fourth rows, respectively, fit three, four, five, and six lines. The corresponding outlier percentages are, respectively, 85, 85, 87, and 90 percentage. The inlier scale is 1.5. (a) The original data with a total number of 1,000 in each data set, are distributed in the range of [0, 100]. (b) to (g) The results obtained by RHT, MS, RHA, J-linkage, KF, and AKSWH2, respectively. We do not show the results of AKSWH1 and ASKC, which are similar to those of AKSWH2, due to the space limit.

compare the fitting error results and the computational speed[2] (Table 2). From Fig. 8 and Table 2, we can see that **for the three-line data**, AKSWH1/2 succeed in fitting and segmenting the three lines automatically. *With some user-adjusted thresholds*, RHT, ASKC, J-linkage, and KF also correctly fit all three lines: RHT and ASKC achieves slightly higher fitting errors than AKSCH1/2, while KF achieves the lowest fitting error among the eight methods because the gross outliers are effectively removed at the first step. However, AKSWH2 is much faster than KF and J-linkage (about 40 times faster than KF and 5.5 times faster than J-linkage). AKSWH2 is faster than AKSWH1 (about 3.4 times faster) because AKSWH2 uses a more effective weighting function, resulting in fewer selected significant hypotheses (376 significant hypotheses) than AKSWH1 (1,682 selected significant hypotheses). RHT is faster (but has a larger fitting error) than AKSWH2 because RHT uses several additional user-specified thresholds, which are used to stop the random sampling procedure. MS and RHA, respectively, fit one/two lines but fail to fit the other lines.

**For the four line data**, AKSWH1/2, ASKC, and KF correctly fit the four lines. However, AKSWH2 is significant faster than KF and ASKC. In contrast, RHT and J-linkage correctly fit three lines but wrongly fit one. MS and RHA correctly fit two/one lines but fail to fit two/three, respectively.

**For the five line data**, only AKSWH1/2 and ASKC correctly fit all five lines, while ASKC achieves relatively higher fitting error than AKSWH1/2. J-linkage and KF correctly fit four of the five lines, while RHT and RHA fit three lines but wrongly fit two. MS only fits two lines correctly.

2. For MS, we use the C++ code from http://coewww.rutgers.edu/riul. For J-linkage, we use the Matlab code from http://www.toldo.info.

**In fitting the six lines**, AKSWH1/2, ASKC, and KF correctly fit the six lines. We note that KF costs much higher CPU time in fitting the six-line data than the three-line data (around four times slower). This is because KF obtains more clusters (30 clusters) for the six-line data than the three-line data (20 clusters) at the step of spectral clustering, and most of the CPU time is spent on merging clusters in the step of model selection. ASKC also uses more CPU time in fitting to the six-line data than the three-line data. In contrast, AKSWH2 uses comparable CPU time for both types of data and is about 200 times faster than KF and more than three times faster than ASKC.

We also evaluate the performance of the eight approaches in different inlier scales, outlier percentages, and the cardinality ratios of inliers belonging to each model instance. For the following experiment, we use the three-line data similar to that used in Fig. 8. **Changes in inlier scales:** We generate the three line data with 80 percent outlier percentage. The inlier scale of each line is slowly increased from 0.1 to 3.0. **Changes in outlier percentages**: We draw the breakdown plot to test the influence of outlier percentage on the performance of the eight approaches. We gradually decrease the inlier number of each line equally while we increase the number of the randomly distributed outliers. The inlier scale of each line is set to 1.0. **Changes in cardinality ratios between the inliers of each line**: The inlier numbers of the three lines are equal to each other at the beginning and the inlier scales of all lines are 1.0. We gradually reduce the inlier number of one line while we increase the number of data points belonging to the other lines. Thus, the relative cardinality ratio between the lines is increased gradually. We repeat the experiment 20 times and we report both the average results (in Fig. 9) and the averaged standard variances and the worst results (in Table 3) of the fitting errors.

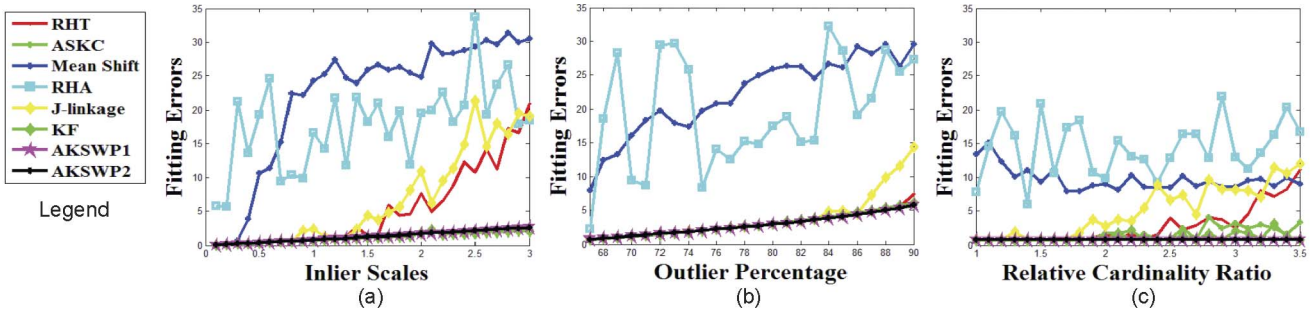



Fig. 9. The average results obtained by the eight approaches. (a)-(c) The influence of inlier scale, outlier percentage, and the relative cardinality radio of outliers to inliers, respectively.

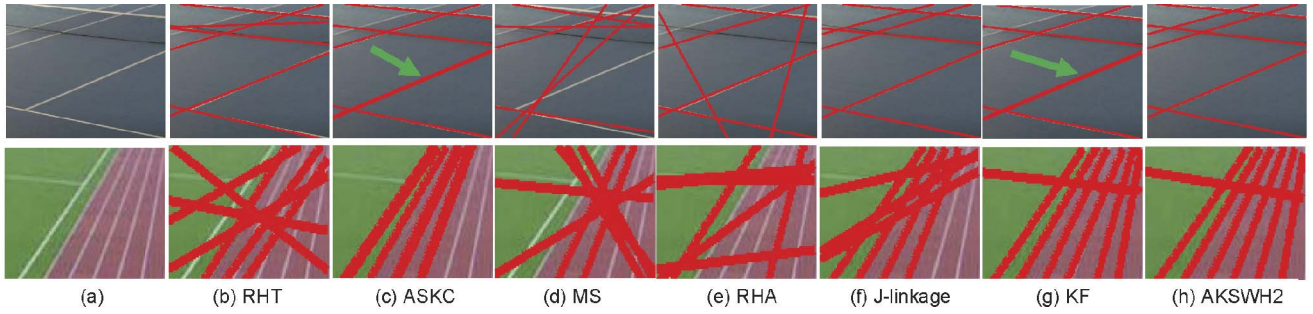



Fig. 10. Examples for line fitting with real images. First ("tennis court") and second ("tracks") rows, respectively, fit six and seven lines. (a) The original images. (b) to (g) The results obtained by RHT, ASKC, MS, RHA, J-linkage, KF, and AKSWH2, respectively. We do not show the results obtained by AKSWH1, which are very similar to those of AKSWH2, due to the space limit.

As shown in Fig. 9 and Table 3, AKSWH1/2 show robustness to the influence of the inlier scale, the outlier percentage, and the inlier cardinality ratio, and both achieve the most accurate results among the competing approaches. In contrast, RHA does not achieve good results for all three experiments. MS does not work well for the experiments in Figs. 9b and 9c, but it works reasonably well in Fig. 9a when the inlier scale is less than 0.3. Among the four approaches, RHT, ASKC, J-linkage, and KF, in Fig. 9a RHT and J-linkage, respectively, break down when the inlier scale is larger than 1.4 and 0.8; in contrast, ASKC and KF achieve relatively good performance when ASKC is given the number of model instances and the step size in KF is adjusted manually. Fig. 9b shows that J-linkage and RHT begin to break down when the outlier percentage is larger than 84 and 89 percent, respectively, and both achieve better results than MS and RHA but worse results than ASKC and KF. Fig. 9c shows that ASKC and KF work well when the relative cardinality ratio of inliers is small, but they begin to break down when the inlier cardinality ratio is larger than 2.0 and 2.2, respectively. J-linkage and RHT achieve worse results than ASKC and KF.

It is worth noting that when we fix the step-size $h$ to be 50 (as specified in [26, Fig. 2]), KF wrongly estimates the number of model instances in about 10 percent of the experiments in Fig. 9a, 2.3 percent in Fig. 9b, and 52.3 percent in Fig. 9c. When the inlier cardinality ratio of model instances in Fig. 9c is larger than 2.3, KF wrongly estimates almost all the number of model instances in the experiments. In contrast, AKSWH2 with a fixed K value correctly find the model instance number for all the experiments in Fig. 9.

TABLE 3
The Fitting Errors in Parameter Estimation

| | Inlier scale | | Outlier percentage | | Cardinality ratio | |
|---|---|---|---|---|---|---|
| | Std.Var. | Max.Err. | Std.Var. | Max.Err. | Std.Var. | Max.Err. |
| RHT | 4.75 | 36.1 | 0.36 | 31.2 | 2.90 | 26.7 |
| ASKC | 0.06 | 3.01 | 0.10 | 6.23 | 2.30 | 18.5 |
| MS | 5.03 | 46.3 | 7.70 | 49.6 | 3.85 | 32.8 |
| RHA | 19.3 | 333 | 27.0 | 334 | 14.3 | 143 |
| J-linkage | 5.60 | 41.0 | 2.06 | 45.6 | 5.95 | 22.6 |
| KF | 0.19 | 8.14 | 0.11 | 6.41 | 1.07 | 20.4 |
| AKSWH1 | 0.04 | 2.88 | 0.09 | 6.01 | 0.02 | 0.86 |
| AKSWH2 | 0.06 | 2.75 | 0.09 | 6.09 | 0.03 | 0.86 |

## 5.2 Experiments with Real Images[3]

### 5.2.1 Line Fitting

We test the performance of the eight approaches using real images for line fitting (shown in Fig. 10). For the "tennis court" image, which includes six lines, there are 4,160 edge points detected by the Canny operator [44]. As KF consumes a large memory resource in calculating the kernel matrix and in spectral clustering, we randomly resample 2,000 data points for "tennis court" and test KF on the resampled data. As shown in Fig. 10 and Table 4, AKSWH1/2 and J-linkage correctly fit all six lines. However, AKSWH1/2 are much faster than J-linkage and KF (AKSWH1/2 are, respectively, 12/25 and 208/426 times faster than J-linkage and KF). Among the other competing approaches, MS fits two lines but fails in fitting four lines; RHT, ASKC, and KF succeed in fitting five out of the six

3. To make it easy for readers to test and compare their approaches with the competing approaches in this paper, we put the data used in the paper (for line fitting, circle fitting, homography-based segmentation and fundamental matrix-based segmentation) and the corresponding ground truth (for homography-based segmentation and fundamental matrix-based segmentation) online: http://cs.adelaide.edu.au/~dsuter/code_and_data/.

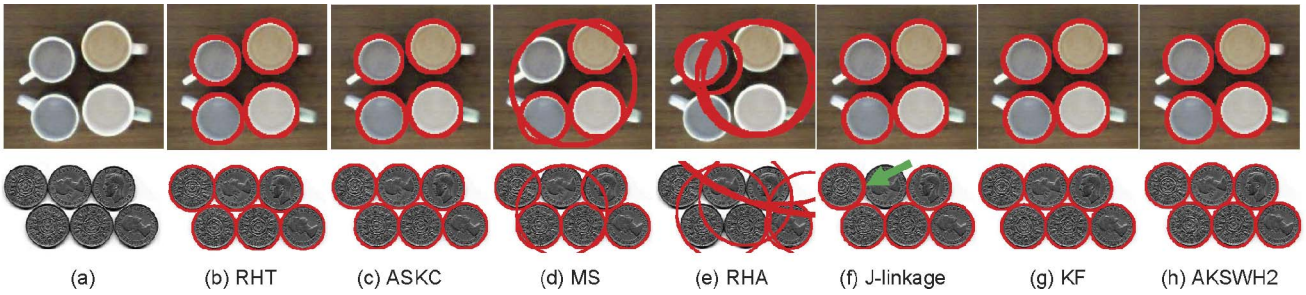



Fig. 11. Examples for circle fitting. First ("cups") to second ("coins") rows, respectively, fit four and six circles. (a) The original images. (b) to (h) The results obtained by RHT, ASKC, MS, RHA, J-linkage, KF, and AKSWH2, respectively.

lines while RHA correctly fit four lines only. We note that ASKC fits two estimated lines to the same real line and so does KF (as pointed by the green arrows in Fig. 10), while this does not occur in AKSWH1/2 because it uses MIT to effectively fuse overclustered modes belonging to the same model instance.

For the "tracks" image, which includes seven lines, 1,022 Canny edge points are detected. Only KF and AKSWH1/2 successfully fit all seven lines in this challenging case. ASKWH2 (4.23 seconds) is about 40 times faster than KF and 5.8 times faster than ASKWH1 (see Table 4). In contrast, both ASKC and J-linkage correctly fit four lines but fail in fitting three lines. The number of the successfully fitted lines by RHT, MS, and RHA is, respectively, one, two, and three.

### 5.2.2 Circle Fitting

In Fig. 11, we fit the rims of the four cups and of the six coins. For the "cups" image, there are 634 Canny edge points detected. The inliers corresponding to each of the four cups are about 17 to 20 percent of the data points. As shown in Fig. 11, AKSWH1/2, RHT, ASKC, J-linkage, and KF achieve accurate fitting results for all four cups. However, AKSWH2 is much faster than ASKC, J-linkage, and KF (respectively, about 5, 6, and 15 times faster—see Table 4) and requires less user-specified thresholds. MS correctly fits the circle rims of three cups but fails in one; RHA only fits one circle rim but fails in the other three.

For the "coins" image,[4] there are 1,168 edge points. As shown in Fig. 11 and Table 4, AKSWH1/2, RHT, ASKC, and KF correctly fit the six rims while AKSWH2 is 310 times faster than KF, and 6.5 times faster than ASKC. Although RHT is about 3 times faster than AKSWH2, it requires many user-specified thresholds. For J-linkage, there are two estimated model instances corresponding to one coin (pointed by the green arrow). The problem is due to the clustering criterion used in J-linkage. MS correctly fits five rims of the coins but wrongly fits one rim of the coins. RHA successfully fits only one rim of the coins.

TABLE 4
The CPU Time Used by the Approaches (in Seconds)

| | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 |
|---|---|---|---|---|---|---|---|---|
| court | 0.74 | 18.3 | 7.98 | 15.2 | 273 | 4600* | 22.1 | 10.8 |
| tracks | 0.85 | 9.92 | 6.57 | 22.5 | 31.2 | 167 | 24.5 | 4.23 |
| cups | 2.19 | 14.9 | 11.1 | 10.8 | 20.2 | 51.4 | 9.86 | 3.27 |
| coins | 1.25 | 26.7 | 3.91 | 12.5 | 28.8 | 1272 | 6.59 | 4.10 |
| 5planes | 3.40 | 164 | 10.1 | 29.6 | 295* | 5931* | 11.9 | 15.1 |
| blocks | 3.12 | 164 | 19.7 | 27.0 | 310* | 3445* | 11.3 | 19.8 |

*M1-RHT; M2-ASKC; M3-MS; M4-RHA; M5-J-Linkage; M6-KF; M7-AKSWH1; M8-AKSWH2. "*" means the approach uses the resampled data points.*

### 5.2.3 Plane Fitting

Range data usually contain multiple structures but they do not have many random outliers. However, inliers of one structure are pseudo-outliers to the other structures. Thus, the task of range data segmentation requires an approach to be robust. To make the experimentation feasible, we use 2,000 resampled data points for KF and 5,000 resampled data points for J-linkage (these are challenged by large numbers of data points). We model the plane segmentation as a standard planar regression problem for calculating the residuals.

As shown in Fig. 12, RHA achieves the worst results. J-linkage and KF correctly fit all the planar surfaces for the "five plane" data, but wrongly fit one plane for the "block" range data. The other five approaches correctly fit all five planes in both cases, but the results of RHT and ASKC (due to the underestimated inlier scales) are less accurate than MS and AKSWH1/2. MS succeeds in both cases. One crucial reason that MS can succeed is because the inlier scales of the planes in the two range data are very small (about $6 \times 10^{-4}$ for the "five plane" and $8 \times 10^{-3}$ for the "block"). This is consistent with the previous analysis (in Fig. 9a) that MS works better when the inlier scale is smaller. AKSWH1/2 achieve better results than the other approaches and AKSWH1/2 can estimate the number of planes automatically. For computational efficiency, AKSWH1/2 are much faster than J-linkage (more than one order faster) and KF (more than two order faster) in dealing with a large number of data points and ASKC (8 to 14 times faster) in dealing with data involving multiple structures (see Table 4).

The computational times of the eight approaches for the line/circle/plane fitting experiments described in Sections 5.2.1, 5.2.2, and 5.2.3 are summarized in Table 4.

### 5.2.4 Homography-Based Segmentation

We use the approaches mentioned in Section 4 to obtain homography hypotheses and compute the residuals of the SIFT point correspondences, see Fig. 13.

To quantitatively compare the results of AKSWH1/2 with those of the competing approaches, we manually select a set of inliers for each model instance and use the chosen

4. The image is taken from http://www.murderousmaths.co.uk.

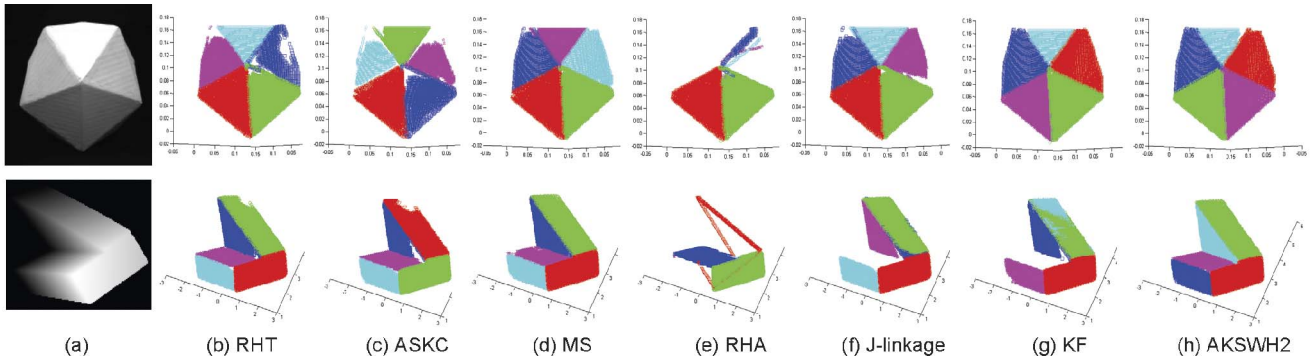



Fig. 12. Examples for range image segmentation. First ("five planes" including 10,842 data points) to second ("block" having 12,069 data points) rows fit five planes. (a) The original images. (b) to (g) The segmentation results obtained by RHT, MS, RHA, ASKC, MS, RHA, J-linkage, KF, and AKSWH2, respectively.

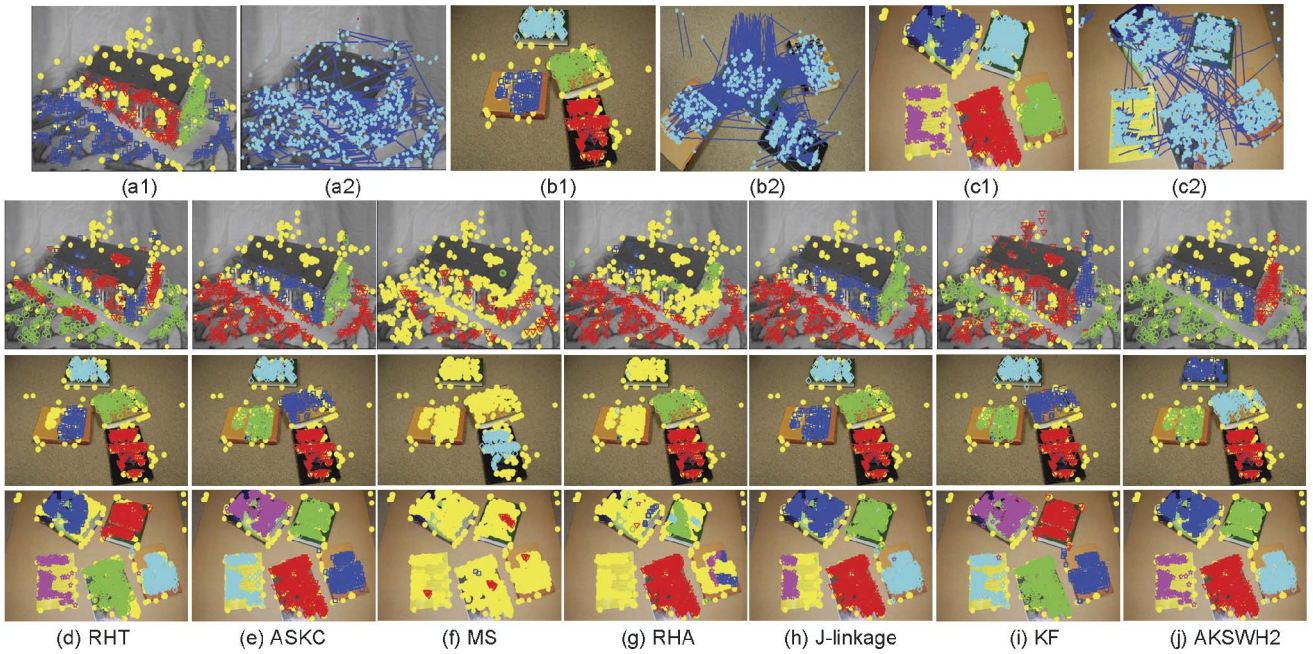



Fig. 13. More examples showing the ability of the eight approaches in estimating homographies and segmenting multiple-structure data with the image pairs of "Model House" (referred as to "MH"), four books (referred as to "4B"), and five books (referred as to "5B"). (a1), (b1), and (c1) The left images with the ground truth segmentation results superimposed. Each model instance is in one color. The yellow dots are the outliers. (a2), (b2), and (c2) The right images with the disparities of corresponding points superimposed. The second to fourth rows are the segmentation results obtained by RHT, ASKC, MS, RHA, J-linkage, KF and AKSWH2, respectively.

inliers to estimate the model parameters, which are used as the grand truth (GT) model parameters. All true inliers belonging to that model instance can be derived from the GT parameters (for the next section we also perform a similar manual ground truth extraction). The comparisons are shown in Table 5. For each data, we also report the total number (TN) of data points, the minimum inlier number, and the maximum inlier number (MaxN) corresponding to the model instances, from which the outlier percentage and the relative inlier cardinality ratio between the model instances can be calculated.

From Fig. 13 and Table 5 we can see that MS achieves the worst result and fails in all five data. RHA succeeds in fitting one data with less accuracy but fails in four. RHT and KF succeed in four data but fail in one. Only ASKC (given the number of model instances), J-linkage (given the inlier scale and a threshold for the number of model instances), and AKSWH1/2 correctly fit and segment all the data. For computational efficiency, we note that when the data contain fewer data points and a smaller number of model instances, e.g., for MC2, AKSWH2 takes similar computational time as J-linkage while AKSWH2 is about 80 percent faster than ASKC and more than one order faster than KF. In contrast, when data involve more data points and a greater number of model instances, such as the data "5B," AKSWH2 is much faster (6.9, 6.5, and 375 times faster, respectively) than ASKC, J-linkage, and KF.

As for the scale estimation error obtained by AKSWH2 ASKC, and KF, we do not consider MH because KF breaks down for this case. The mean scale error for the other four data obtained by AKSWH2, ASKC, and KF is, respectively, 0.28, 0.92, and 3.80; AKSWH2 achieves the most accurate accuracy in estimating the inlier scales.

TABLE 5
The Fitting Errors Obtained by the Eight Approaches and the CPU Time Used (in Seconds)

| | MinN | MaxN | TN | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MC2 | 131 | 200 | 347 | 1.89 (14.8) | 1.24 (30.8) | 44.6 (147) | 4.47 (9.02) | 1.29 (15.4) | 1.87 (302) | 1.29 (28.3) | 1.28 (16.8) |
| MC3 | 136 | 303 | 799 | 1.36 (16.0) | 0.49 (54.7) | 507 (121) | 361 (13.7) | 0.66 (26.5) | 0.66 (85.8) | 0. 52 (29.6) | 0.50 (12.5) |
| MH | 105 | 294 | 702 | 39.5 (39.5) | 1.57 (55.3) | 365 (136) | 151 (11.6) | 2.43 (29.0) | 15.6 (64.5) | 1.85 (29.3) | 1.56 (15.1) |
| 4B | 122 | 231 | 777 | 1.79 (33.0) | 0.86 (123) | 467 (189) | 350 (17.8) | 0.95 (33.6) | 2.11 (989) | 0.92 (20.7) | 0.87 (14.9) |
| 5B | 257 | 577 | 2394 | 1.27 (27.4) | 0.43 (168) | 132 (187) | 244 (29.4) | 0.85 (157) | 0.62 (8996) | 0.50 (36.0) | 0.44 (24.3) |

*M1-RHT; M2-ASKC; M3-MS; M4-RHA; M5-J-linkage; M6-KF; M7-AKSWH1; M8-AKSWH2. Images MC2, MC3, and MH are taken from http://www.robots.ox.ac.uk/~vgg/data.*

### 5.2.5 Fundamental Matrix Based Segmentation

In Fig. 14, we give examples showing the ability of AKSWH2 in two-view based multiple-body motion segmentation using the fundamental matrix model. We employ the *seven-point algorithm* [10] to obtain the fundamental matrix hypotheses and compute the residuals of the SIFT point correspondences using the *Sampson Distance* [10].

From Table 6, we can see that MS and RHA fail to fit both data. RHT and J-linkage achieve better results than MS and RHA for the "BC" data, but both approaches break down for the "BCD" data. Only ASKC, KF, and AKSWH1/2 correctly fit both data while AKSWH2 achieve more accurate results than ASKC and KF. The mean scale error obtained by ASKC, KF and AKSWH2 is, respectively, 1.25, 0.74, and 0.87, from which we can see KF achieves slightly more accurate results than AKSWH2 in scale estimation but AKSWH2 is about one order faster (9.7 and 8.4 times faster for "BC" and "BCD," respectively) than KF.

## 6 Conclusion

We have tackled three important problems in robust model fitting: estimating the number of model instances, estimating the parameters, and estimating the inlier scale of each model instance. At the heart of our solution are a novel scale estimator IKOSE and an effective robust fitting framework. The overall framework, AKSWH, uses accurate scale estimates of IKOSE to weight each sample hypothesis. This allows us to gain efficiency by early elimination of weak hypotheses corresponding to bad $p$-subsets (that may involve outliers). We demonstrate those efficiency gains by conducting experiments with two variants of our approach (AKSWH1/2) and comparing their runtimes with competing approaches. AKSWH1 represents a version of our approach that is similar to putting RANSAC (with a known correct error tolerance) into our framework instead of our enhanced method of scale estimation and hypothesis scoring. The efficiency gains are easily seen as ASKWH2 is almost always faster than ASKWH1 (and most of the competitors). Both AKSWH1/2 generally outperform competitors in terms of robustness and accuracy—this attests to the effectiveness of the rest of our framework.

Most remarkably, our approach is essentially free of any user choice of parameter—save one, related to the inlier percentage of the smallest structure likely to be resolved by our approach. When we do not know the inlier percentage, we fix the K value in AKSWH2 to a small number (10 percent in our case) to avoid breakdown. Actually, this is not such an unusual assumption as the inlier percentage estimate has been usually either assumed known, or fixed to some low value, and then used to determine the number of the required $p$-subsets *in many robust approaches which employ the popular random sampling scheme*. We acknowledge that in some cases (probably most) the user does not know the outlier percentage in any reasonably accurate or precise form. This issue is related to the issue of "how big should the recovered populations/structures be." This issue is compounded by other aspects related to structural scale (e.g., how close two planes are in fitting, for example, planar parts of a wall with small recesses, etc.). As stated in Section 1, we do not claim that we have proposed an automatic system which has solved all the problems of recovery of structure from image data: This is impossible because how many structures there are in a scene is usually a question without a unique answer. However, the number of structures can be determined *when we assume that all the structures of interest are relatively large and have roughly comparable size* (we ignore the structures with very few supporting data points). In our approach, such an assumption is implicit by using a conservative K (in this paper, we fix the K value to be 10 percent of the data points for all the experiments). This can be thought of as (roughly) saying we are assuming we are not interested in finer scale structure that makes up less than 10 percent of the data points. Our experiments show that the proposed method, while having no claims of optimality, is effective. Of course,

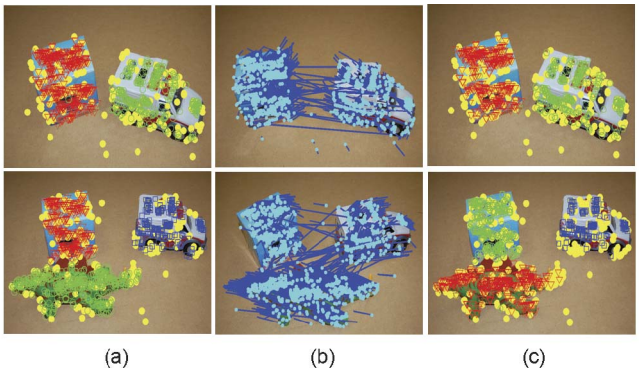



Fig. 14. The segmentation results obtained by AKSWH2 for the image pairs of "Box-Car" (referred as to "BC") and "Box-Car-Dinosaur" (referred as to "BCD"), respectively. (a) The left image with the ground truth segmentation superimposed. (b) The right image with the disparities of corresponding points superimposed. (c) The results obtained by AKSWH2.

TABLE 6
The Fitting Errors Obtained by the Eight Approaches and the CPU Time Used (in Seconds)

| | MinN | MaxN | TN | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BC | 378 | 500 | 1116 | 0.39 (12.6) | 0.17 (145) | 5.92 (139) | 14.5 (149) | 0.27 (116) | 0.15 (626) | 0.14 (198) | 0.15 (64.5) |
| BCD | 232 | 460 | 1227 | 29.7 (13.4) | 0.53 (197) | 26.5 (229) | 36.4 (168) | 27.6 (123) | 0.51 (519) | 0. 51 (274) | 0.26 (61.5) |

*M1-RHT; M2-ASKC; M3-MS; M4-RHA; M5-J-linkage; M6-KF; M7-AKSWH1; M8-AKSWH2.*

our approach is not without some drawbacks and possible improvements. For example, guided sampling approaches may improve the efficiency. Likewise, though we have demonstrated superiority to the comparator methods, as described in their published forms, that does not preclude improvements to those approaches. Finally, it must also be acknowledged that one is not always strictly comparing apples with apples and oranges with oranges. For example, many techniques in this area were developed under the assumption that there is a dominant single structure—and thus were never explicitly designed to recover multiple structures, though perhaps others have tried to extend the methods to do so by fit-and-remove iterations, etc. When one has only a single dominant structure, or perhaps knows the scale of noise, other methods still suffice. However, our method still generally performs well—albeit slower.

Compared to the more recent methods of ASKC [17], [29], J-Linkage [25], and KF [26], the advantage of our method can be understood as such: ASKC is not developed for the multistructure fitting, thus it may suffer from the problems related to sequential fit-and-remove. J-Linkage requires a user-given global inlier threshold (which may be difficult to determine), thus it assumes similar noise scales across all structures. Finally, as our experiments show, KF does not tolerate different inlier population sizes very well. In contrast, for reasons already described above (i.e., accurate scale estimator, parallel fitting via hypothesis weighting, and fusion), AKSWH is significantly more tolerant to the issues faced by the other methods.

## ACKNOWLEDGMENTS

The authors would like to thank the reviewers for their valuable comments, which greatly helped to improve the paper. Parts of our code are developed based on the "*Structure and Motion Toolkit*" provided by Professor P. Torr. They also thank Dr. R. Toldo and Professor A. Fusiello for sharing the code of J-linkage, and Dr. R. Subbarao and Professor P. Meer for sharing the code of MS. They thank Professor A. Bab-Hadiashar for providing some range image data. This work was supported by the Australian Research Council (ARC) under the project DP0878801, by the National Natural Science Foundation of China under project 61170179, the Special Research Fund for the Doctoral Program of Higher Education of China under Project 20110121110033, and by the Xiamen Science & Technology Planning Project Fund (3502Z20116005) of China. Most of this work was completed when Hanzi Wang was at the University of Adelaide.

## REFERENCES

[1] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[2] H. Wang and D. Suter, "Robust Adaptive-Scale Parametric Model Estimation for Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 11, pp. 1459-1474, Nov. 2004.

[3] C.V. Stewart, "Bias in Robust Estimation Caused by Discontinuities and Multiple Structures," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 8, pp. 818-833, Aug. 1997.

[4] P.J. Huber, *Robust Statistics.* Wiley, 1981.

[5] P.J. Rousseeuw, "Least Median of Squares Regression," *J. Am. Statistical Assoc.,* vol. 79, pp. 871-880, 1984.

[6] P.J. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection.* John Wiley & Sons, 1987.

[7] P.V.C. Hough, "Methods and Means for Recognising Complex Patterns," US Patent 3 069 654, 1962.

[8] L. Xu, E. Oja, and P. Kultanen, "A New Curve Detection Method: Randomized Hough Transform (RHT)," *Pattern Recognition Letters,* vol. 11, no. 5, pp. 331-338, 1990.

[9] M.A. Fischler and R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. ACM,* vol. 24, no. 6, pp. 381-395, 1981.

[10] P. Torr and D. Murray, "The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix," *Int'l J. Computer Vision,* vol. 24, no. 3, pp. 271-300, 1997.

[11] J.V. Miller and C.V. Stewart, "MUSE: Robust Surface Fitting Using Unbiased Scale Estimates," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* pp. 300-306, 1996.

[12] C.V. Stewart, "MINPRAN: A New Robust Estimator for Computer Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 10, pp. 925-938, Oct. 1995.

[13] K.-M. Lee, P. Meer, and R.-H. Park, "Robust Adaptive Segmentation of Range Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 2, pp. 200-205, Feb. 1998.

[14] X. Yu, T.D. Bui, and A. Krzyzak, "Robust Estimation for Range Image Segmentation and Reconstruction," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 16, no. 5, pp. 530-538, May 1994.

[15] H. Chen and P. Meer, "Robust Regression with Projection Based M-Estimators," *Proc. IEEE Ninth Int'l Conf. Computer Vision,* pp. 878-885, 2003.

[16] R. Hoseinnezhad and A. Bab-Hadiashar, "A Novel High Breakdown M-Estimator for Visual Data Segmentation," *Proc. IEEE 11th Int'l Conf. Computer Vision,* pp. 1-6, 1997.

[17] H. Wang et al., "Robust Motion Estimation and Structure Recovery from Endoscopic Image Sequences with an Adaptive Scale Kernel Consensus Estimator," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[18] M. Zuliani, C.S. Kenney, and B.S. Manjunath, "The multiRANSAC Algorithm and Its Application to Detect Planar Homographies," *Proc. IEEE Int'l Conf. Image Processing,* pp. 153-156, 2005.

[19] R. Vidal, Y. Ma, and S. Sastry, "Generalized Principal Component Analysis (GPCA)," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 12, pp. 1945-1959, Dec. 2005.

[20] K. Fukunaga and L.D. Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition," *IEEE Trans. Information Theory,* vol. 21, no. 1, pp. 32-40, Jan. 1975.

[21] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach towards Feature Space a Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 5, pp. 603-619, May 2002.

[22] O. Tuzel, R. Subbarao, and P. Meer, "Simultaneous Multiple 3D Motion Estimation via Mode Finding on Lie Groups," *Proc. 10th IEEE Int'l Conf. Computer Vision,* pp. 18-25, 2005.

[23] R. Subbarao and P. Meer, "Nonlinear Mean Shift over Riemannian Manifolds," *Int'l J. Computer Vision,* vol. 84, no. 1, pp. 1-20, 2009.

[24] W. Zhang and J. Kosecka, "Nonparametric Estimation of Multiple Structures with Outliers," *Proc. Int'l Workshop Dynamical Vision,* pp. 60-74, 2006.

[25] R. Toldo and A. Fusiello, "Robust Multiple Structures Estimation with J-Linkage," *Proc. European Conf. Computer Vision,* pp. 537-547, 2008.

[26] T.-J. Chin, H. Wang, and D. Suter, "Robust Fitting of Multiple Structures: The Statistical Learning Approach," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 413-420, 2009.

[27] Y. Weiss and E.H. Adelson, "A Unified Mixture Framework for Motion Segmentation: Incorporating Spatial Coherence and Estimating the Number of Models," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 321-326, 1996.

[28] K. Kanatani and C. Matsunaga, "Estimating the Number of Independent Motions for Multibody Motion Segmentation," *Proc. Asian Conf. Computer Vision,* pp. 7-12, 2002.

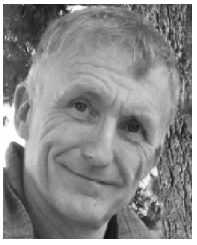
[29] H. Wang, D. Mirota, and G. Hager, "A Generalized Kernel Consensus Based Robust Estimator," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 32, no. 1, pp. 178-184, Jan. 2010.

[30] H. Wang and D. Suter, "Robust Fitting by Adaptive-Scale Residual Consensus," *Proc. Eighth European Conf. Computer Vision,* pp. 107-118, 2004.

[31] A. Bab-Hadiashar and D. Suter, "Robust Segmentation of Visual Data Using Ranked Unbiased Scale Estimate," *ROBOTICA, Int'l J. Information, Education and Research in Robotics and Artificial Intelligence,* vol. 17, pp. 649-660, 1999.
[32] P. Torr and A. Zisserman, "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry," *Computer Vision and Image Understanding,* vol. 78, no. 1, pp. 138-156, 2000.
[33] B. Tordoff and D.W. Murray, "Guided Sampling and Consensus for Motion Estimation," *Proc. European Conf. Computer Vision,* pp. 82-96, 2002.
[34] O. Chum and J. Matas, "Matching with PROSAC: Progressive Sample Consensus," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 220-226, 2005.
[35] M.P. Wand and M. Jones, *Kernel Smoothing.* Chapman & Hall, 1995.
[36] B.W. Silverman, *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, 1986.
[37] M. Sezgin and B. Sankur, "Survey over Image Thresholding Techniques and Quantitative Performance Evaluation," *J. Electronic Imaging,* vol. 13, no. 1, pp. 146-165, 2004.
[38] L. Ferraz et al., "A Density-Based Data Reduction Algorithm for Robust Estimators," *Proc. Third Liberian Conf. Pattern Recognition and Image Analysis,* pp. 355-362, 2007.
[39] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis.* John Wiley and Sons, 1973.
[40] C.E. Shannon, "The Mathematical Theory of Communication," *Bell Systems Technical J.,* vol. 27, pp. 379-423, 1948.
[41] Z.R. Yang and M. Zwolinski, "Mutual Information Theory for Adaptive Mixture Models," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 4, pp. 396-403, Apr. 2001.
[42] Y. Lee, K.Y. Lee, and J. Lee, "The Estimating Optimal Number of Gaussian Mixtures Based on Incremental k-Means for Speaker Identification," *Int'l J. Information Technology,* vol. 12, no. 7, pp. 13-21, 2006.
[43] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision,* second ed. Cambridge Univ. Press, 2004.
[44] J.F. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 6, pp. 679-698, Nov. 1986.

**Hanzi Wang** received the PhD degree in computer vision from Monash University, Australia. He is currently a distinguished professor and "Minjiang Scholar" at Xiamen University, China, and an adjunct professor at the University of Adelaide, Australia. He was a senior research fellow (2008-2010) at the University of Adelaide, Australia, an assistant research scientist (2007-2008) and a postdoctoral fellow (2006-2007) at the Johns Hopkins University, and a research fellow at Monash University, Australia (2004-2006). He was awarded the Douglas Lampard Electrical Engineering Research Prize and Medal for the best PhD thesis in the department. His research interests are concentrated on computer vision and pattern recognition, including visual tracking, robust statistics, model fitting, object detection, video segmentation, and related fields. He has published approximately 50 papers in major international journals and conferences, including the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *International Journal of Computer Vision*, ICCV, CVPR, ECCV, NIPS, MICCAI, etc. He is an associate editor for the *IEEE Transactions on Circuits and Systems for Video Technology* (*T-CSVT*) and he was a guest editor of *Pattern Recognition Letters* (September 2009). He has served as a reviewer for more than 30 journals and conferences. He is a senior member of the IEEE.

**Tat-Jun Chin** received the BEng degree in mechatronics engineering from the Universiti Teknologi Malaysia (UTM) in 2003, and subsequently, in 2007, the PhD degree in computer systems engineering from Monash University, Victoria, Australia. He was a research fellow at the Institute for Infocomm Research in Singapore from 2007-2008. Since 2008 he has been a lecturer at the University of Adelaide, Australia. His research interests include robust estimation and statistical learning methods in computer vision. He is a member of the IEEE.



**David Suter** received the BSc degree in applied mathematics and physics (The Flinders University of South Australia 1977), the Grad. Dip. Comp. (Royal Melbourne Institute of Technology 1984), and the PhD degree in computer science (La Trobe University, 1991). He was a lecturer at La Trobe from 1988 to 1991, and a senior lecturer (1992), an associate professor (2001), and professor (2006-2008) at Monash University, Melbourne, Australia. Since 2008, he has been a professor in the School of Computer Science at the University of Adelaide. He is the head of the School of Computer Science. He served on the Australian Research Council (ARC) College of Experts from 2008-2010. He is on the editorial board of the *International Journal of Computer Vision*. He has previously served on the editorial boards of *Machine Vision and Applications* and the *International Journal of Image and Graphics*. He was a general cochair of the Asian Conference on Computer Vision (Melbourne 2002) and is currently cochair of the IEEE International Conference on Image Processing (ICIP 2013). He is a senior member of the IEEE.