Sparse One-Grab Sampling with Probabilistic Guarantees

Maryam Jaberi, Marianna Pensky, and Hassan Foroosh, *Senior Member, IEEE*

Abstract—Sampling is an important and effective strategy in analyzing "big data," whereby a smaller subset of a dataset is used to estimate the characteristics of its entire population. The main goal in sampling is often to achieve a significant gain in the computational time. However, a major obstacle towards this goal is the assessment of the smallest sample size needed to ensure, with a high probability, a faithful representation of the entire dataset, especially when the data set is compiled of a large number of diverse structures (e.g., clusters). To address this problem, we propose a method referred to as the Sparse Withdrawal of Inliers in a First Trial (SWIFT) that determines the smallest sample size of a subset of a dataset sampled in one grab, with the guarantee that the subset provides a sufficient number of samples from each of the underlying structures necessary for the discovery and inference. The latter is established with high probability, and the lower bound of the smallest sample size depends on probabilistic guarantees. In addition, we derive an upper bound on the smallest sample size that allows for detection of the structures and show that the two bounds are very close to each other in a variety of scenarios. We show that the problem can be modeled using either a hypergeometric or a multinomial probability mass function (pmf), and derive accurate mathematical bounds to determine a tight approximation to the sample size, leading thus to a sparse sampling strategy. The key features of the proposed method are: (i) sparseness of the sampled subset for analyzing data, where the level of sparseness is independent of the population size; (ii) no prior knowledge of the distribution of data, or the number of underlying structures in the data; and (iii) robustness in the presence of overwhelming number of outliers. We evaluate the method thoroughly in terms of accuracy, its behavior against different parameters, and its effectiveness in reducing the computational cost in various applications of computer vision, such as subspace clustering and structure from motion.

Index Terms—Sampling big data, sample size, probabilistic guarantees, parameter/structure estimation, subspace clustering

1 Introduction

Estimating and clustering the underlying structures in data are among the most fundamental problems in computer vision, machine learning, and data analytics. However, the unprecedented growth in data with often high ratio of outliers has created an ever more pressing need for faster and more accurate methods [1]. A popular approach explored in the literature to tackle these problems of size and dimensionality is to estimate the characteristics of the entire data population using only sampled subsets of the data. Methods like RANdom SAmpling Consensus (RANSAC) [2], use sampling to find a single structure in the data, typically for outlier rejection or removal [3], [4], [5]. In most applications, however, multiple instances of a model (structure) exist simultaneously. Examples include multiple independently moving objects in a video, multiple planes in a scene, or multiple instances of the same face in a database under varying lighting conditions. One of the main challenges in such multi-structured data is a high percentage of outliers in the population due to: (i) gross outliers, which are comprised of points that do not belong to any model instance, and (ii) pseudo-outliers, which are points that are inliers to one model instance (structure), but effectively act as outliers to all other model instances in the population [6].

Existing sampling techniques can be broadly divided into two groups: (i) Random iterative (multi-grab) sampling methods that sequentially grab multiple subsets of data in order to fit model instances, and (ii) One-time or one-grab sampling methods that generate all possible hypotheses or model instances from a single subset of data. More details on these two groups of methods are discussed in the next section. However, it is worth noting that a major issue that is overlooked in one-grab sampling is the question of the minimal sample size, which guarantees with high probability that all underlying structures or model instances are discovered. In simple terms, if not enough samples are taken, then one may miss some model instances, and if too many samples are taken, then the benefits of sampling for reducing the computational cost is diminished or may be lost. The study reported herein is focused on this latter problem in one-grab sampling. We provide tight upper and lower bounds on the sample size required for discovering all underlying structures. We provide strong theoretical guarantees and confirm them via simulations and experiments. In particular, we verify that our method leads to a much smaller sample size than the state-of-the-art methods in the literature [7], [8]. In addition, our experiments on real data demonstrate that the proposed solution reduces computational cost without compromising accuracy.

- *M. Jaberi and H. Foroosh are with the Department of Computer Science, UCF, Orlando, FL 32816. E-mail: {mjaberi, foroosh}@cs.ucf.edu.*
- *M. Pensky is with the Department of Mathematics, UCF, Orlando, FL 32816. E-mail: marianna.pensky@ucf.edu.*

Manuscript received 26 Sept. 2016; revised 11 Aug. 2018; accepted 29 Aug. 2018. Date of publication 24 Sept. 2018; date of current version 31 Oct. 2019. (Corresponding author: Maryam Jaberi.)
Recommended for acceptance by H. Kjellstrom.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2018.2871850

More specifically, we answer the following question:

"*Given a large population of N points with C embedded structures and gross outliers, what is the minimum number of points r to be selected randomly in one grab in order to make sure with probability P that at least ε points are selected on each structure, where ε is the number of degrees of freedom of each structure*."

The answer to this question is significant because of the following reasons: (i) We will show that even under a huge number of pseudo-outliers and gross outliers, r is extremely small (i.e., using our estimate for the sample size yields in practice a sparse subset of the population); (ii) Although P is an intricate function of r (difficult to invert), we prove that it is a non-decreasing function. Hence, r can be mathematically approximated and found by a simple one-dimensional search, regardless of the dimensionality of the data; (iii) The sample size r is very slowly growing with the number of underlying structures C, keeping the sampled subset sparse even under overwhelming numbers of pseudo-outliers; (iv) The method does not assume any prior knowledge about the distribution of data; (v) The sparsity of the sampled subset implies a significant reduction in computation.

The rest of this paper is organized as follows. Section 2 provides a survey of the related work. In Section 3, we introduce our sparse sampling scheme and demonstrate with some simple examples. Section 4 shows the accuracy of the presented method using experiments. Section 5 presents example applications in computer vision. Finally, Section 6 provides a discussion of the important findings and the concluding remarks.

2 Related Work

Below, we review the related work on statistical sampling and discuss their pros and cons.

Iterative Multi-Grab Sampling Methods. These methods focus on iteratively sampling subsets of points for finding consensus sets that yield the underlying model instances [9]. Greedy methods such as RANSAC or RANSAC-like methods [2], [10], [11], [12] focus on sequentially detecting structures and estimating their parameters. In order to detect each of the structures, many subsets of *ε-tuples* are sampled randomly until a set consisting of only inliers is determined. Here, ε represents the number of degrees of freedom of the model instances and *ε-tuples* is a subset of ε points. Multi-RANSAC [8], on the other hand, attempts to find all model instances in parallel, but it assumes that the underlying structures do not intersect, which is an impractical limitation for most applications. Iterative multi-grab methods are generally suboptimal for multi-structure data (e.g., when multiple model instances co-exist in data), since the stopping criterion is usually non-trivial, and inaccurate initial fitting can significantly affect the detection of the model instances [13]. These sequential methods also assume the outliers are distributed uniformly, which is violated when multiple model instances are present, forming thus pseudo-outliers that are not distributed uniformly. This issue is discussed in [14], where it is shown that clustered pseudo-outliers are more difficult to handle than uniformly distributed gross outliers. More recent studies [15], [16], [17] attempt to use regularization for better model fitting, while handling multiple intersecting structures. These methods are also iterative. However, instead of using a greedy sequential approach, they resort to a constraint optimization process to find the underlying model instances that can optimally represent the entire data. A major drawback of these methods is that at many iterations one may end up testing unnecessary model hypotheses. Moreover, the method is in part supervised, since the number of model instances is assumed to be known *a priori*, which in practice may not be feasible.

One-Grab Sampling Methods. These methods strive to find the best segmentation of the entire data by estimating a set of putative model instances from a sampled subset of data. Starting with a set of putative models, they attempt to determine those that best fit the entire population. In [18] a set of randomly sampled points are used to grow the models that best fit the data. Similarly, the methods in [14], [19] use a subset of points to accelerate the model fitting process, but provide no guaranty as to how big the sample set should be. The method proposed in [20] achieves optimal model selection by measuring a residual error based on histogram distribution of every point in the data for all possible predicted models from a sampled subset of the population. The methods in [21] and [22] also start with a set of initial models derived from randomly selected points and merge them to obtain the best segmentation of the entire data. In a similar manner, the Multi-Bernoulli SAmple Consensus (MBSAC) method proposed in [7] grabs a subset of *ε-tuples* and uses a multi-Bernoulli filtering approach [23] in order to detect all the model instances simultaneously. This method provides some guaranties on the number of required *ε-tuples* to be sampled, and determines the optimal model instances by removing the models with low probabilities. Random Cluster Models (RCM), proposed in [24], includes a conditional random sampling of possible model hypotheses from initially clustered points in a weighted graph. This method along with many others like [1], [23], [25] use some prior knowledge in order to sample points and select the initial hypotheses.

One-grab sampling methods handle multi-model fitting problems well and are not affected by the perils of iterative multi-grab sampling. However, they mostly suffer from poor computational efficiency. In order to detect all the models in the data, these methods either include the entire population [26], [27], or sample a subset of population with no guaranty on the optimal sample size, i.e., use heuristics. This often leads to either failing to detect some valid model instances, or taking too many samples leading to too many hypotheses in their search for the optimal segmentation of data. This becomes, in particular, problematic when dealing with "big data" in a high-dimensional space. To tackle the computational cost of these methods, a natural solution would be to avoid excessive oversampling by finding an accurate method of determining the required sample size, which is the focus of this paper. In particular, we generalize the one-grab sampling, since a tight choice of sample size also allows for sampling without any prior assumptions, while guaranteeing that all model instances can be discovered. The key is to determine the size of the one-grab sampled set, so that all the structures are still represented adequately and can be discovered with high probability, despite a substantial ratio of outliers. This turns out to be the solution to a complex nonlinear equation with no analytic closed form answer. We solve the problem by formulating it in terms of either a multinomial or a hypergeometric pmf and bounding the solution to reduce it to a binary search problem. We impose no constraints on the distribution of points. Thus the samples are taken uniformly, i.e., requiring no prior knowledge of the distribution of data. Moreover, the proposed method can handle structures with different dimensions.

3 Proposed Method

Sparse Withdrawal of Inliers in a First Trial [28] is a one-grab sampling method that we propose in order to select a random subset of data with no prior assumption about the distribution of points, and with the guaranty that the estimated optimal sample size is both tightly bounded and sufficient to determine, with high probability, all the underlying model instances in the data. Since taking a one-grab sample of r points from a population of size N may be viewed as sampling r points one point at a time without replacement, as is well known, the probability of whether a randomly drawn point has some specified feature is determined by the hypergeometric pmf [29], [30]. Also, as is well-known, if the population size N is much larger than r, then this probability is closely approximated by a multinomial pmf [29], [30]. Intuitively, we can see why this is true, because multinomial pmf models sampling with replacement, and when $N \gg r$, the chance of drawing the same sample point after replacement would be extremely negligible, i.e., multinomial would asymptotically approach the hypergeometric. We will show later that, indeed for most practical applications, the condition $N \gg r$ is readily satisfied, i.e., our tightly bounding of the estimate of sample size yields also the desired sparsity. This is in contrast to other methods described earlier that rely on heuristics. Sampling with replacement modeled by multinomial pmf is easier to handle mathematically, due to the independence of events leading to simpler approximations. Therefore, below we study both the hypergeometric model, which is the true mathematical model for our problem, and the multinomial model that closely approximates our problem. We thus analyze and compare the accuracy of both models in Section 4.

3.1 Definitions and Notations

We consider the situation where a population of$N$ points is comprised of C classes with sizes $\theta_1, \ldots, \theta_C$ where $\sum_i^C \theta_i = N$. In a real situation, neither the number of classes nor their sizes are known, so we propose a sampling algorithm that does not assume this knowledge. In particular, our algorithm requires four input parameters: the population size N, the minimum size of a sample set per structure ε, the minimum model size θ, and the probability $1-\delta$ of grabbing at least ε points from each structure. While parameters N, ε, and θ are model-driven, parameter δ is custom defined and depends on the application at hand. For example, it is likely that one would choose a lower value for δ in health-related or safety-related applications than in analyzing marketing data. The value ε here is greater than or equal to the number of degrees of freedom of each model instance. The minimum model size, θ, is the minimum number of inlier points necessary to accept a candidate model. If the class size is smaller than θ, then we are not interested in extracting it.

The sampling is then carried out by grabbing r points at random from the population. If there are d_i points from each of C classes (model instances), then $\sum_i^C d_i = r$ and the probability of sampled points $d_i \geq \varepsilon$ for each of C classes is Δ, where $\sum_i^C \Delta = \delta$. A key novelty of our sampling method, as mentioned earlier, is that it provides the tight lower bounds $r = r(N, \varepsilon, \theta, \delta)$ such that $d_i \geq \varepsilon$ for each of the classes $i = 1, \ldots, C$, with probability of at least $1-\delta$. To derive the SWIFT sampling method, we start by assuming the worst-case scenario, where all model instances in the population are presumed to be of size θ. This implies that (i) all outliers are pseudo-outliers and (ii) the maximum number of possible model instances C that can be potentially present in the population is given by

$$C = \left\lceil \frac{N}{\theta} \right\rceil, \tag{1}$$

where $\lceil \cdot \rceil$ rounds the fraction to the nearest upper integer, and N is the population size. As explained earlier, with this one-time grab sampling, the vector $(d_1, \ldots, d_C)$ follows the multivariate hypergeometric distribution (since the sampling is equivalent to point-by-point sampling without replacement), so the derivation of SWIFT in this case does not require any additional assumptions.

3.2 Modeling SWIFT by Hypergeometric Distribution

We have a population of N points comprising of underlying C model instances. Suppose now we select a subset of r points sampled at random in a one-time-grab, with d_i points coming from the ith instance. Then, the pmf of the vector $(d_1, \cdots, d_C)$ follows the multivariate hypergeometric distribution (sampling without replacement) [29], [30]:

$$P(d_1 = x_1, \ldots, d_C = x_C) = \frac{\prod_{i=1}^C \binom{\theta_i}{x_i}}{\binom{N}{r}}, \tag{2}$$

where N is the population size, C is the number of classes and $\theta_1, \ldots, \theta_C$ are their respective sizes, with $\sum_{i=1}^C \theta_i = N$, $\sum_{i=1}^C x_i = r$ and $0 \leq x_i \leq \theta_i,\ i = 1, \ldots, C$. Equation (2) expresses the probability of a given sample set in terms of r. However, our goal in SWIFT sampling is to solve the inverse problem of finding r for a given probability. For this purpose, we recall that we are dealing with the symmetric case of $\theta_1 = \theta_2 = \cdots = \theta_C = \theta$ and the worst case scenario, where for a given N, the maximum possible classes is $C = \frac{N}{\theta}$. Therefore, $N = C\theta$ and Equation (2) becomes:

$$P(d_1 = x_1, \ldots, d_C = x_C) = \frac{\prod_{i=1}^C \binom{\theta}{x_i}}{\binom{C\theta}{r}}. \tag{3}$$

The objective of the method can then be expressed as follows: Find r such that, for a given value $\delta > 0$, the probability of selecting at least ε points in each of C model instances is at least $1-\delta$, that is

$$P\big(\cap_{i=1}^C (d_i \geq \varepsilon)\big) \geq 1 - \delta \quad \text{provided} \quad \sum_{i=1}^C d_i = r. \tag{4}$$

The solution to the above problem can be determined by finding the lower and upper bounds for the tail probabilities of the multivariate hypergeometric pmf (see, e.g., [31], [32], [33]) and use them for derivation of the value of r.

3.2.1 Lower Bound

We find a lower bound for the probability in the left-hand side of the inequality Equation (4) and use it for finding the lower bound for r. Note that $P(d_i \geq \varepsilon) = 1 - P(d_i \leq \varepsilon - 1)$ and, therefore,

$$\begin{aligned} P\big[\cap_{i=1}^C (d_i \geq \varepsilon)\big] &= 1 - P\big[\cup_{i=1}^C (d_i \leq \varepsilon - 1)\big] \\ &\geq 1 - \sum_{i=1}^C P(d_i \leq \varepsilon - 1). \end{aligned} \tag{5}$$

Using the above inequality, we can prove the following theorem:

Theorem 1. *Let $(C-1)\theta \geq r$. Then, one has*

$$P(\cap_{i=1}^{C}(d_i \geq \varepsilon)) \geq 1 - \sum_{i=1}^{C} P(d_i \leq \varepsilon - 1) = 1 - C\Delta, \quad (6)$$

where

$$\Delta = P(d_1 \leq \varepsilon - 1) \leq P(d_1 = 0) \sum_{k=0}^{\varepsilon-1} \binom{r}{k} \left(\frac{\theta}{N - r - \theta + k}\right)^k, \quad (7)$$

and

$$P(d_1 = 0) = \prod_{j=0}^{\theta-1} \frac{N - r - j}{N - j} \leq \left(1 - \frac{r}{N}\right)^{\theta} \leq e^{\frac{-r}{C}}. \quad (8)$$

The proof of this and later theorems are placed in the Appendix, which is available in the IEEE Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2018.2871850).

Algorithm 1. Estimating Sampling Size r by Hypergeometric pmf

Input: N and $(\varepsilon, \theta, \delta)$
Set $C := \lceil \frac{N}{\theta} \rceil$
  $r_{\texttt{Min}} := C \times \varepsilon$
  $r_{\texttt{Max}} := N$
  $\Delta := 0$
  Do Binary search {since $\Delta(r)$ is non-increasing on r}
while $r_{\texttt{Min}} < r_{\texttt{Max}}$ **do**
  $r := \frac{1}{2}(r_{\texttt{Min}} + r_{\texttt{Max}})$
  Find Δ using inequities (6)-(8)
  if $C\Delta > \delta$ **then**
    $r_{\texttt{Min}} := r$
  else
  $r_{\texttt{Max}} := r$
  end if
end while
return r

Setting $C\Delta = \delta$ and solving Equation (7) for r with $\Delta = \frac{\delta}{C}$, we obtain the necessary lower bound on the SWIFT sampling size. The derived inequality for $\Delta(r)$ in Equation (7) is a non-increasing function on r. This sets the statement in Equation (6) as a non-decreasing function of r when C is reasonably small. Given N and $(\varepsilon, \theta, \delta)$, we can simply find r by using a binary search through all possible values of r between $C \times \varepsilon$ and N. For $P(d_1 = 0)$, one can either use the exact formula or its upper bound in Equation (8). In our numerical studies, we used the exact expression for better accuracy. Algorithm 1 shows the detailed steps to find the sample size r. Note that, since we are using the lower bounds for the multivariate hypergeometric pmf, Theorem 1 provides the bound on the sample size r. In order to show that this lower bound is tight, in the next section, we study also the upper bound for r.

3.2.2 Upper Bound

In this section, we derive an upper bound for the tail probabilities of the hypergeometric pmf, which will set an upper bound on the sample size r for SWIFT. In particular, the following statement is true:

Theorem 2. *If $(C-2)\theta \geq r$, then*

$$P(\cap_{i=1}^{C}(d_i \geq \varepsilon)) \leq 1 - C\Delta + \frac{C(C-1)}{2}\Delta', \quad (9)$$

where Δ is defined in Equation (7),

$$\Delta' = P[(d_1 \leq \varepsilon - 1) \cap (d_2 \leq \varepsilon - 1)] \leq P_0 \times \sum_{k=0}^{\varepsilon-1} \sum_{l=0}^{\varepsilon-1} \binom{r}{k,\ l} \left[\frac{\theta}{(C-2)\theta - r}\right]^{k+l}, \quad (10)$$

and

$$P_0 = P(d_1 = 0, d_2 = 0) \leq \left(1 - \frac{r}{C\theta}\right)^{2\theta} \leq e^{\frac{-2r}{C}}. \quad (11)$$

By combining (6) and (9), the probability of choosing at least ε points in each model instances can be bounded above and below as:

$$1 - C\Delta \leq P(\cap_{k=1}^{C}(d_i \geq \varepsilon)) \leq 1 - C\Delta + \frac{C(C-1)}{2}\Delta'. \quad (12)$$

In Section 4, we demonstrate that the bounds derived above are indeed very tight, providing accurate estimates for r. The described estimation for the upper-bound is used to show the accuracy and tightness of estimated sample size r with respect to the ground-truth.

3.3 Modeling SWIFT by Multinomial Distribution

Although SWIFT is a "one-grab" sampling method, and hence is exactly modeled by the multivariate hypergeometric pmf, the estimation of r requires solution of nonlinear inequalities, which can be time consuming when both N and C are large. However, as mentioned earlier, when $N \gg r$ the hypergeometric pmf can be accurately approximated by the multinomial pmf, which basically implies that sampling with replacement approaches the sampling without replacement when $N \gg r$. Again, this is due to the fact that for $N \gg r$, the probability of grabbing any sample point more than once becomes extremely negligible. Since choosing r as small as possible is one of the goals of SWIFT, the assumption of $N \gg r$ is justified. This motivates us in this section to study the approximation of the tail probabilities based also on a multinomial pmf. An important outcome of the study in this section is that it eliminates the need for searching for r when $N \gg r$ (see Algorithm 2).

Algorithm 2. Estimating Sampling Size r by Multinomial pmf

Input: N and $(\varepsilon, \theta, \delta)$
Set r according to formula (16) or (18)
if $N \gg r$ **then**
  return r
else
  Find r using algorithm1 using formula (14) instead of (6)-(8)
  return r
end if

Suppose a set of r points is selected at random with replacement from a population of $N \gg r$ points comprised of C classes with sizes $\theta_1, \ldots, \theta_C$, so that $\sum_i^C \theta_i = N$. Then, the pmf of d_i is given by:

$$P(d_i = x_i) = \binom{r}{x_i} p_i^{x_i} (1 - p_i)^{r - x_i}, \quad (13)$$

where $\sum_i^C x_i = r$, $0 \leq x_i \leq \theta_i$ and $p_i = \frac{\theta_i}{N}$. Using Equation (5) and recalling that $C\theta = N$, $\theta_i = \theta$ and $p_i = \frac{1}{C}$ for $i = 1, \ldots, C$,

we obtain that the inequality Equation (6) still holds in this case, but with a different value of Δ:

$$\begin{aligned}\Delta &= P(d_1 \leq \varepsilon - 1) \\ &= \sum_{k=0}^{\varepsilon-1} \binom{r}{k} \frac{(C-1)^{r-k}}{C^r} = Bin\left(r, \frac{1}{C}\right),\end{aligned} \tag{14}$$

where $\sum_{i=1}^{C} d_i = r \geq \varepsilon \times C$ and $Bin(r, \frac{1}{C})$ is the binomial pmf. Note that r is the only unknown variable in Equation (14) and that the right-hand side of this equation is a non-decreasing function of r when C is relatively small. Thus, similar to Section 3.2.1, one can find the value of r by using a binary search through the possible values. Moreover, by applying Bernstein inequality for the tail probability of the binomial distribution: for any $t > 0$

$$\begin{aligned}&P\left(Bin\left(r, \frac{1}{C}\right) < \frac{r}{C} - t\right) \\ &\leq \exp\left(-\frac{t^2 C^2}{2r(C-1) + 4C^2 t/3}\right).\end{aligned} \tag{15}$$

We can find an explicit lower bound on r. In particular, the following statements hold.

Theorem 3. *Let N be large, so that the multinomial approximation of the hypergeometric distribution holds. If*

$$r \geq \frac{14}{3} C \ln\left(\frac{C}{\delta}\right) + 2C(\varepsilon - 1), \tag{16}$$

then

$$P(\cap_{i=1}^{C}(d_i \geq \varepsilon)) \geq 1 - \delta. \tag{17}$$

Proposition 1. *Based on the Central Limit Theorem [34], if the total sample size r is relatively large, (say, r is an order of magnitude bigger than C), then the binomial distribution for the sample size d_i in the ith model instance can be approximated by the normal distribution: $Bin(r, \frac{1}{C}) \approx \mathcal{N}(\frac{r}{C}, \frac{r(C-1)}{C^2})$. Thus, the sampling size r can be obtained by:*

$$r \geq \left(A^2(C-1) + 2C(\varepsilon - 1)\right), \tag{18}$$

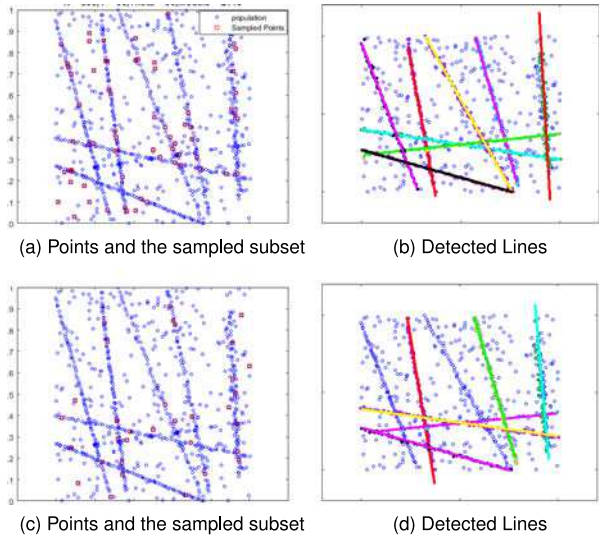
where A is:

$$A \equiv \max\left(1, \sqrt{2\ln\left(\frac{C}{\sqrt{2\pi}\delta}\right)}\right). \tag{19}$$

Based on what is described in this section, the multinomial distribution estimation can be used to calculate a lower-bound for r when the condition in inequity Equation (18) is satisfied. Otherwise, one needs to use Equation (14) or use the hypergeometric estimation of r. In Equation (4), we will evaluate the tightness of the inequality Equation (18). Algorithm 2 shows the steps for finding r using the multinomial model.

3.4 Clustering and Parameter Estimation

Once a SWIFT subset of a population of points is selected, the sampled points are used to estimate the model parameters. We can then detect the valid model instances in the population, where valid means a model size larger than θ. Different clustering method may be used as the back-end to our sampling step, e.g., [15], [19], [21], [35], [36]. In our experiments below, we used mean-shift [37], [38] (also used later in Section 5), which is a non-parametric unsupervised clustering method that does not require a prior knowledge of the number of clusters nor any constraints on the distribution of the clusters. The application studied in Section 5.4 uses a different algorithm [39] in the clustering step.



(a) Points and the sampled subset (b) Detected Lines

(c) Points and the sampled subset (d) Detected Lines

Fig. 1. The population includes eight lines and 50 percent gross outliers. (a) shows the data points and sampled subset when the size of the sample set is calculated using SWIFT. (b) shows the detected lines using SWIFT samples and mean-shift. (c) and (d) show the underestimated sample size and the detected lines, failing to find all the instances in the data.

To demonstrate immediately how SWIFT is useful in guaranteeing an accurate estimate of the required sample size in order to find all model instances from a sparse subset of data, we run a test on a simple synthetic data. In this experiment, we show that an accurate sample size r can yield sufficient number of points to detect all the model instances. On the other hand, one is likely to fail in finding all the model instances, when we do not have a guaranty on sample size. This example includes 2D lines in a population size of 850 points and is illustrated in Fig. 1. The points form 8 noisy lines in 2D, crossing randomly in the presence of 50 percent gross outliers. In Figs. 1a and 1b we used SWIFT to calculate the sample size r. The selected sample size is $r = 66$ which is computed with input parameters $r(\varepsilon = 2, \theta = 80, 1 - \delta = 0.95)$. Using this sample size, r points are uniformly sampled in a one-time grab, as shown with small red boxes in Fig. 1a. This subset generates 2145 hypotheses without any preprocessing for finding neighboring points (as in [19]) nor having any prior knowledge regarding the distribution of the data. Fig. 1b shows the clusters formed from the sampled subset and the subsequently detected inliers. As shown, a sufficient number of points are selected and all the models (lines) are detected accurately. On the other hand, Fig. 1c and 1d show how the same process for detecting model instances can fail, when the sample size is insufficient. In this example, we used the same data as in the previous experiment. The sample subset is 33, which is selected uniformly at random and generates 528 hypotheses. As illustrated, two out of eight model instances are not detected.

4 Experimental Evaluation

In this section, we evaluate the proposed sparse one-grab sampling method, and investigate the effect of different input parameters.

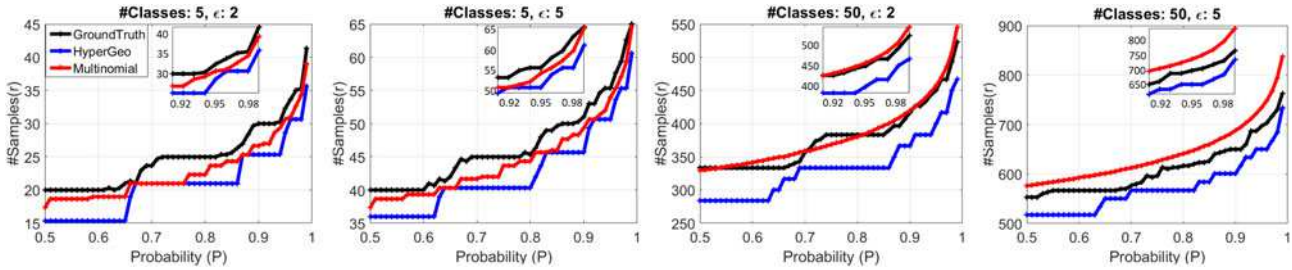


Fig. 2. Comparison of estimated r averaged over 200 independent trials versus ground-truth of r when $N \in \{10^3, 10^4, 10^5\}$, $\varepsilon \in \{2, 5\}$, and $C \in \{5, 50\}$.

4.1 Accuracy of the Proposed Sampling Method

As mentioned earlier, due to the non-decreasing property of the derived equations in Section 3, the SWIFT sample size r can be estimated by a simple search, with the time complexity of $O(\log(N))$. The success of the proposed SWIFT sampling highly depends on the accuracy of the estimated values for input parameters. In essence, by following the worst-case scenario, we are treating the problem as if there were no gross outliers in the population. Moreover, the parameter θ is chosen to be equal to the smallest possible size for a valid model instance. These two assumptions, plus the fact that the value of the probability P is in practice chosen as high, ensure that the computed sample size r can guarantee with high probability at least ε points on every structure. Below, we experimentally evaluate the accuracy of the estimated sample size r and the derived bounds in Equations (6), (12) and (14). For this purpose, we used a statistical simulation of sampling without replacement to generate a "ground-truth" data set for different sample sizes with the computed probabilities. The ground-truth simulation was averaged over 1000 independent trials.

Accuracy of Estimated Sample Size. In Sections 3.2 and 3.3, we introduced two different solutions for the lower bound of the sample size r, based on hypergeometric and multinomial pmf models. The defined $\delta(r)$ in both inequity Equation (7) and the equality Equation (14) are non-increasing with respect to r. Using binary search through all possible values of r, one can find the best sample size r. We validated the accuracy of these estimates against ground-truth. For this purpose, we chose different population sizes with different embedded model instances. The ground-truth and estimated values of r given by Equations (7) and (14) are plotted as a function of the probability $1 - \delta$ in Fig. 2. These plots present the average values of r over 200 independent trials for population sizes of $N \in \{10^3, 10^4, 10^5\}$. As expected, our approximations of the lower bound estimation of hypergeometric distribution and multinomial distribution follow the ground-truth closely. The multinomial distribution can overestimate the sample size when the number of classes grow.

In Section 3.2.1, we derived an estimate of r based on the hypergeometric model. To find the estimate, we used an assumption of $\theta \gg \varepsilon$ in Equation (7). Now, we examine how violating this assumption affects the accuracy of estimating the sample size r. For this purpose, we kept the population size N and ε fixed and increased the number of possible classes C. As a result of this increase, the minimum size for a valid class θ will decrease and get closer to ε. This violates the assumption of $\theta \gg \varepsilon$. As illustrated in Fig. 3, when $\theta \gg \varepsilon$ holds, the estimated sample size r is very accurate and close to the ground-truth. However, by decreasing the class size θ, the distance between θ and ε shrinks. This reduces the accuracy of the estimated value r, and in fact the method overestimates the sample size r.

In addition to the estimated upper/lower bounds, in Section 3.3, we introduced an approximation for sample size r under the condition of $r \gg C$. In this part, we examine the accuracy of the estimate in Equation (18). In Fig. 4, the result of the estimated values of r given by Eqaution (18) and the ground-truth are plotted against different desired probability values $1 - \delta$ and when $N \in \{10^3, 10^4, 10^5\}$. In Fig. 4a, the computed value of r satisfies the condition of $r \gg C$. Thus, the estimated value r (using Equation (18)) closely follows the ground-truth in most parts and overestimates the sample size when probability $1 - \delta$ is close to one. However, the experiment in Fig. 4b violated the condition of $r \gg C$. Therefore, the estimated sample size r using Equation (18) overestimated the sample size r and it is not close enough to the ground-truth.

Tightness of Upper/Lower Bounds. We evaluated the tightness of both upper and lower bounds given by the inequalities in Equations (6) and (9). For this purpose, we examined different population sizes with different numbers of model instances. The results are presented in Fig. 5. These graphs illustrate the average results over 200 independent trials computed for population sizes of $N = \{10^2, 10^3, 10^4\}$. Results demonstrate that both the lower bound and the upper bound approximations tightly follow the ground-truth of $1 - \delta$. Of course, a conservative estimate of r would be given by Equation (6). However, the proof of tightness of these bounds indicates that we would not drastically overestimate r using Equation (6).

4.2 Role of Different Parameters

In this section, we investigate the effect of changing each of the parameters in Equation (6) on estimating the value of r.

The Role of ε. One of the parameters in estimating r is the minimum sample set ε per structure to be withdrawn by SWIFT. If the population size N is fixed, then the growth in ε leads to an increase in the number of required samples r for a given probability $1 - \delta$. This behavior is illustrated in Fig. 6a in which the population size, N, and the number of model instances C are kept constant. However, we can see that the growth in r as ε increases is independent of $1 - \delta$.

Number of Model Instances C and Model Size θ. In a constant population size N, increasing the number of model instances forces the method to grab more points in order to guarantee, with probability $1 - \delta$, minimum ε points on each model instance. This behavior is shown in Fig. 6b. Note that the relation between θ and C is defined based on Equation (1). When N is constant, increasing C is equivalent to decreasing the value of θ. Therefore, a similar behavior is observed for θ in Fig. 6c. A very important observation is that, when the

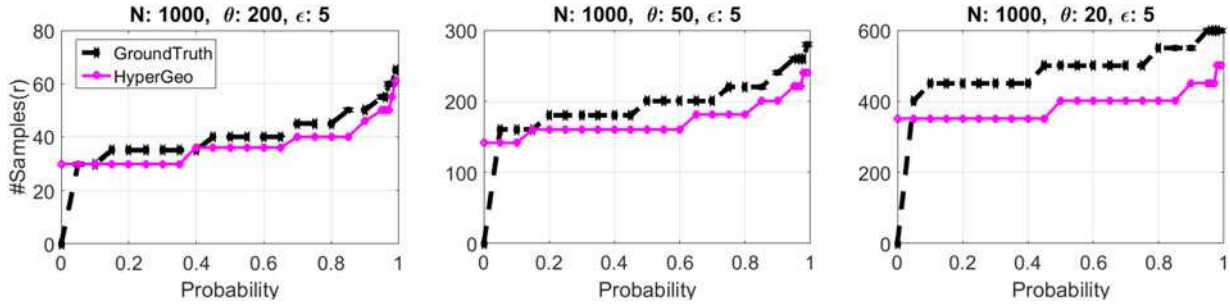


Fig. 3. Examine the difference between θ and ε on the accuracy of estimated r in hypergeometric distribution. Here, $N \in \{10^3, 10^4, 10^5\}$, $\varepsilon = 5$ and from left to right C is $C \in \{5, 20, 50\}$.

number of classes C is fixed, increasing the population size N does not affect the number of required sampled points r, i.e., *the level of sparseness is independent of the population size*. In other words, r remains small regardless of the population size N, which is a desirable sparseness property and justifies the multinomial approximation. On the other hand, in Equation (1), we see that adding more gross outliers increases the worst case estimate for C in Equation (6). In the next section, we study the possible applications that can use SWIFT as the front end. Later, we compare SWIFT with the state of the art method proposed in [7].

5 Application Examples

As a generic unsupervised sparse sampling method, SWIFT can be used in virtually any scenario where multiple structures need to be detected in a large population of points. Here, a structure could be in a physical space (e.g., planar surfaces or other 3D structures), or in some abstract feature space (e.g., the space of all fundamental matrices, all homographies in some configuration of scene/camera motion, or subspaces formed in some high dimensional spaces). Below, we give some examples.

5.1 Detecting 2D lines

In this experiment, we consider detecting 2D lines. This is a classical model detection and it is used in this section to study the effect of SWIFT sampling on accuracy and time complexity of detected models. We show that sample sizes smaller than the one given by SWIFT can fail to detect all model instances and decrease the accuracy in model detection. Also, oversampling does not increase the likelihood of detection and would increase the computational time.

We generated a dataset of up to eight noisy lines that intersect each other. An example of this dataset has been shown in Fig. 1. We detected the model instances by sampling a subset of points in a one-time grab sampling. Using this subset, we generated all the valid hypotheses and found their inliers. Finally, we applied mean-shift to discover the lines. To show that SWIFT sampling can guarantee a sufficient number of points and high accuracy, we ran three separate experiments: (i) using SWIFT to estimate the optimal sample size, (ii) an underestimated sample size, and (iii) an overestimated sample size. Table 1 shows the percentage of points sampled, the accuracy, and the time cost of the examined scenarios. As shown in the table, SWIFT sampling has similar accuracy compared with the result of the overestimated sample size. The underestimated sample size is faster but has lower accuracy. The time complexity using SWIFT sample size is an order of magnitude better than the overestimated case. The data in this dataset is contaminated with Gaussian noise and the results are shown in the presence of (i) no gross outliers and (ii) 50 percent gross outliers.

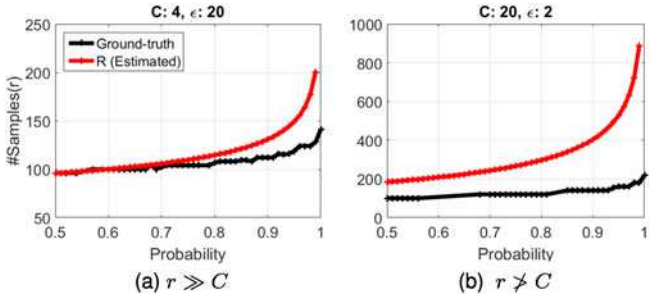


Fig. 4. Examine the average accuracy of estimated r using multinomial estimation in Equation (18). Here, $N = 10^5$ and from left to right $\varepsilon = \{20, 2\}$ and $C \in \{4, 20\}$. The condition of $r \gg C$ is satisfied in (a) and violated in (b).

5.2 Detecting 3D Planes

Plane detection is a prerequisite for various computer vision tasks. In this experiment, we investigated the accuracy of our sparse sampling method for detecting planes in 3D space. In the first step, we considered synthetic models as illustrated in Fig. 7. We examined two different scenarios. The first used a set of 3D points from Castelvecchio dataset [21] with three planes and no gross outliers (Fig. 7a). Using SWIFT sampling just 5.6 percent of the data is used to detect planes in this figure. In Fig. 7b, we used a synthetic dataset, with two planes and 50 percent gross outliers. Using SWIFT, 1.4 percent of data was selected for the detection step. As it is shown in Fig. 7, the planes are detected correctly and gross outliers are not included in the models. Note also that the estimated sample sizes for both cases were similar despite different population sizes.

In the second experiment, we examined real cases where images are collected with Kinect. Generally, the point clouds generated with Kinect include a huge number of points while the number of valid model instances in the scene is small. (i.e., N and θ are large relative to C). In these particular cases, the procedure of finding inliers for all the candidate model instances is a time consuming process since the total number of points $N = 167{,}028$ is extremely large. In order to overcome this problem, the SWIFT method was applied in two levels. In the first level, the value of r was computed to sample the minimum required number of points to instantiate each model candidate (which is $\varepsilon_1 = 3$ for detecting planes). Thus, the sample size r is used to instantiate all the model candidates. In the second level, we set the value of ε to a bigger number (e.g., $\varepsilon_2 = 100$) and sampled a subset of points from the

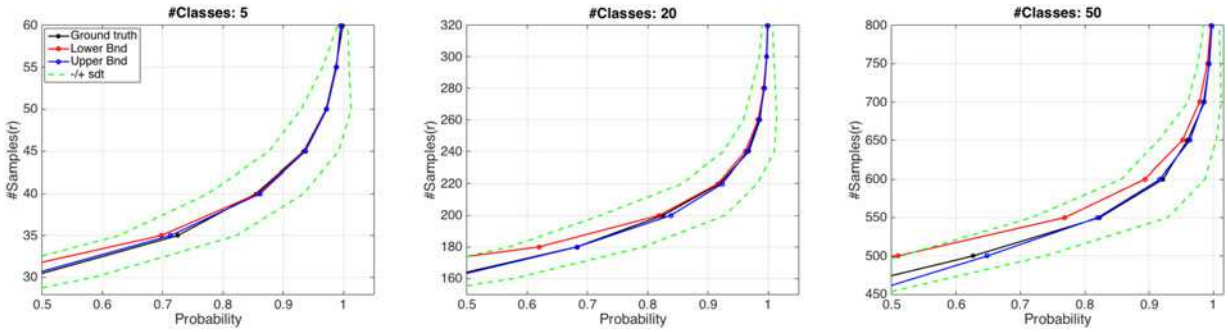


Fig. 5. Estimated values of lower bound and upper bound of r for $1-\delta$ averaged over 200 independent trials versus the ground-truth when $N = 10^4$ and $\varepsilon = 4$. From left to right: $C \in \{5, 20, 50\}$.

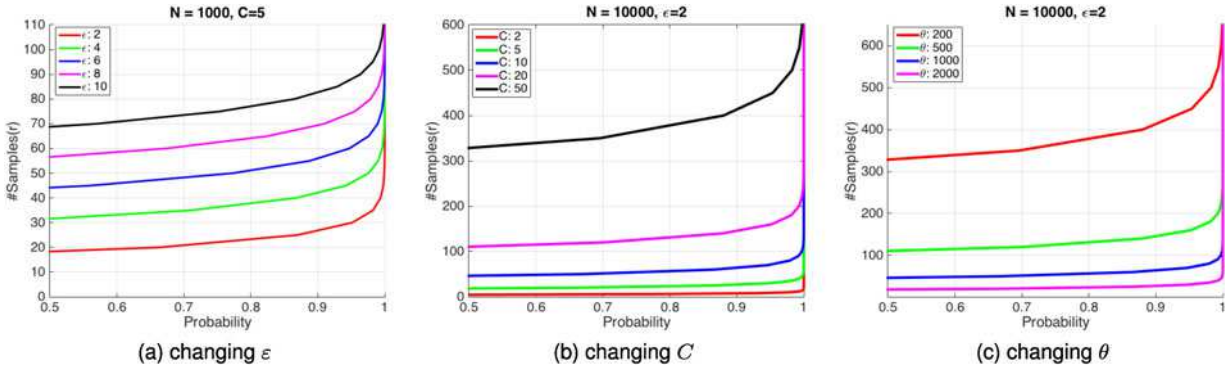


Fig. 6. The effect of changing different parameters on the sample size r. (a) Increasing the value of ε forces the method to select more samples in order to remain with the same probability. (b) and (c) Increasing C or decreasing θ forces the method to grab more samples to reach the same probability.

TABLE 1
Performance Comparison Among 3 Different Scenarios of
(i) SWIFT Sample Size (ii) Underestimated Sample Size (iii) Overestimated Sample Size

Gross Outlier: 0%

#Classes	2			4			6			8		
	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)
Underestimated	2.06	0.68	0.02	2.26	0.77	0.03	2.16	0.76	0.03	2.59	0.88	0.05
Overestimated	25.4	**1.00**	0.08	26.6	**1.00**	0.36	25.9	**1.00**	0.99	31.2	0.98	3.24
SWIFT (r)	**8.45**	**1.00**	**0.05**	**8.86**	**1.00**	**0.08**	**8.65**	**1.00**	**0.14**	**10.4**	**0.99**	**0.35**

Gross Outlier: 50%

#Classes	2			4			6			8		
	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)	%Sample	%Acc	Time(s)
Underestimated	2.19	0.96	0.03	2.60	0.91	0.05	2.56	0.92	0.08	2.58	0.87	0.10
Overestimated	26.2	**1.00**	0.87	31.2	**1.00**	2.68	30.4	0.99	6.49	30.6	0.98	11.9
SWIFT (r)	**8.74**	**1.00**	**0.14**	**10.4**	**1.00**	**0.33**	**10.1**	**1.00**	**0.69**	**10.2**	**0.98**	**1.15**

population with a guaranty of selecting at least $\varepsilon_2 = 100$ points in each plane. The second subset of points was then used, instead of the entire population, as the group of points from which we selected the inliers for each model candidate. This example shows that SWIFT can be inherently implemented also in a multi-level setting. Fig. 8 shows the point cloud data from Kinect used to accurately detect three planes in the scene using the SWIFT algorithm. By filtering the points with $depth = 0$ the total number of points in the cloud was $N = 167{,}028$ and $\theta = 30{,}000$. Setting $1-\delta = 0.9$, the size of sampled points in the first level when $\varepsilon_1 = 3$ is $r = 43$ and in the second level when $\varepsilon_2 = 100$ is $r = 722$.

5.3 Multibody Structure from Motion

Estimating motion models in a video sequence is a classical problem in computer vision. This problem gets more complicated in dynamic cases when multiple rigid objects move independently in a 3D scene [40], [41]. Multibody structure from motion refers to the problem when there are several views of a 3D scene and the motions, structures, and camera calibration are unknown. Recent studies in this area suggest various solutions to this problem [40], [42]. In this section, we used the method in [42] but used SWIFT in the sampling step. The first assumption in [42] is that the number of independent motions in the scene and their parameters are unknown,

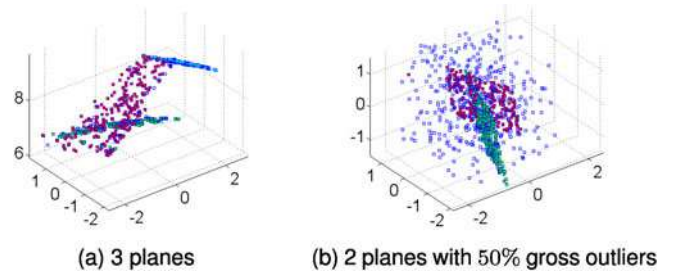


Fig. 7. Using SWIFT to detect 3D Planes when $1-\delta=0.9$. (a) Data is from the Castelvecchio dataset [21] where $r=42$ and $N=754$. (b) Blue points are outliers that are not grouped in any model when $r=43$ and $N=3{,}000$.

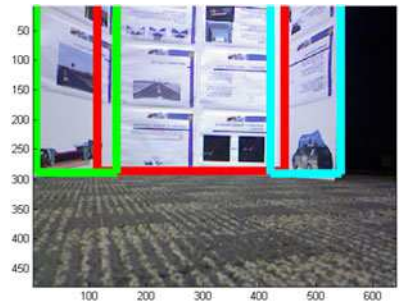


Fig. 8. Detecting planes in 3D point cloud data collected using a Kinect.

where each motion may either be estimated with a homography or a fundamental matrix. Thus, to start the process a set of 2-D point correspondences is required. Then a fixed number of point sets are randomly sampled to generate candidate homographies and fundamental matrices, using the constraints that the minimum required correspondences for a homography is 4 and for fundamental matrix is 7. A shortcoming of the method in [42], however, is that one must specify the number of samples in order to ensure detecting all the motions in the scene. To investigate the application of SWIFT in this problem, we generated 100 synthetic scenes each containing three 3D-objects (not necessarily planar) and a single moving camera. For each synthetic scene, an initial 300×300 image was created. The 3D-objects and the camera were moved randomly and independently and then a new 300×300 image was taken. An example of 3D-objects and their 2D projections are shown in Fig. 9. Using the images, point correspondences were generated by selecting 50 random points from each object. Assuming that points at close proximity are likely to belong to the same object, the sampling strategy explained in [42] divided images heuristically into 9 overlapping areas and sampled points locally. In our experiment, to exploit the proximity constraint, we applied a simple image segmentation algorithm to divide the image into separate clusters. Later, we show that the accuracy of image segmentation does not affect the final results.

The proposed method in [42] samples a batch of ε-tuples to detect fundamental matrices and the homographies. Since the degree of freedom of fundamental matrices and the homographies are different, separate sets have to be sampled for detecting each of the matrices. In SWIFT, however, r points are sampled in a one-time grab. We can employ the sampled set r to find both homographies and fundamental matrices, i.e., since ε for a fundamental matrix is greater that ε for a homography, we can sample r as the number of points required for finding all fundamental matrices. A subset of r is sufficient to find the homographies. In the first experiment, the effect of changing the accuracy of image segmentation was studied. In this experiment, based on the chosen value of θ, the sample size r was computed and grabbed from the total population of $N=200$ points, with 50 points (25 percent) of gross outliers added to the correspondences. Using the sample set of r points, the number of inliers selected in each segment is computed. Fig. 10 shows the number of inliers per segment as the accuracy of image segmentation is increasing with $\varepsilon_h=4$ and $\varepsilon_f=7$ for the homographies and fundamental matrices, respectively. As can be seen from this experiment, a key advantage of using SWIFT sampling is that the required number of samples to maintain a certain level of accuracy with a given probability $1-\delta$ can be calculated. Therefore, the accuracy of the results can be maintained stable as illustrated in Fig. 11. In fact, as the number of gross outliers is growing, the size of population N is also increasing. Since, the other parameters θ, δ and ε are fixed, increasing N leads to selecting a more accurate number of points r. Figs. 11a, 11b, 11c, 11d demonstrate the idea of automatic adaption of r and stability of SWIFT sampling in terms of accuracy of results. The parameters in this figure, in (a), the computed SWIFT sample size for ($\varepsilon=4$, $\theta=\{40,50\}$, $1-\delta=0.9$) is $r=\{42,32\}$ for multinomial model and $r=\{36,29\}$ for hypergeometric model. In (b) for ($\varepsilon=7$, $\theta=\{40,50\}$, $1-\delta=0.9$), the computed SWIFT sample size $r=\{63,49\}$ and $r=\{45,36\}$ for the multinomial and hypergeometric models respectively.

To examine the method on a real case, we used the image data in [42] that include three motions and %25 gross outliers. Using the SWIFT algorithm, we calculated the sample points for both homographies and fundamental matrices. First, we used SWIFT to sample a sufficient number of points. Assuming that points at close proximity belong to

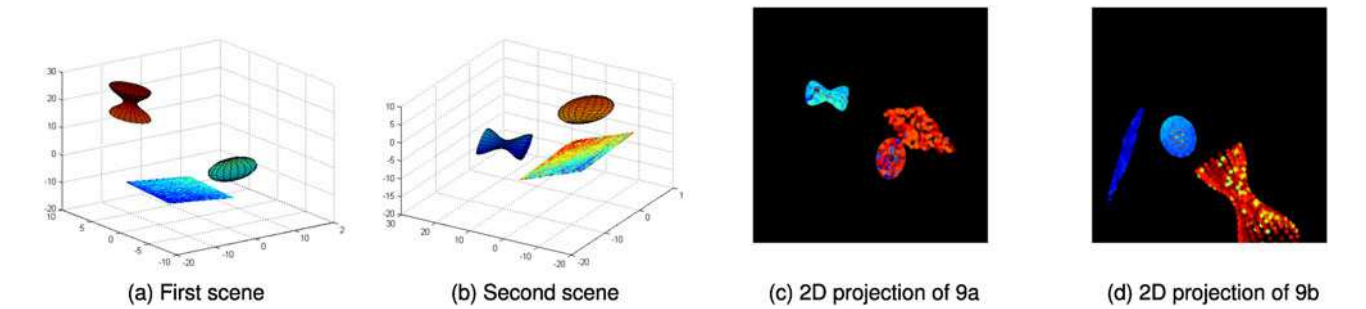


Fig. 9. Synthetic data for multibody structure from motion. The outliers are added after moving objects and computing the 2D projections. These outliers are not shown in this figure. (a) The 3D objects that are not necessarily planar. (b) 3D objects are moved randomly and independently. (c) The image is taken from (a). (d) The image is taken from (b) after moving the camera.

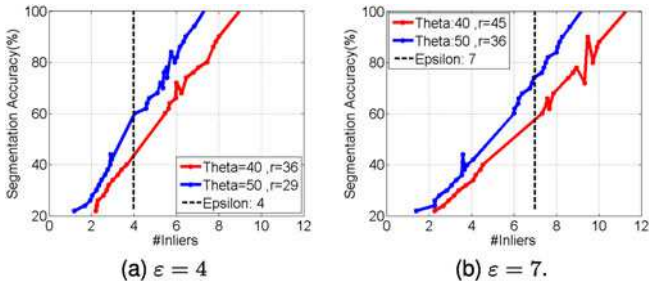


Fig. 10. The effect of image segmentation accuracy on the average number of inliers sampled in each model instance when we have three motions.

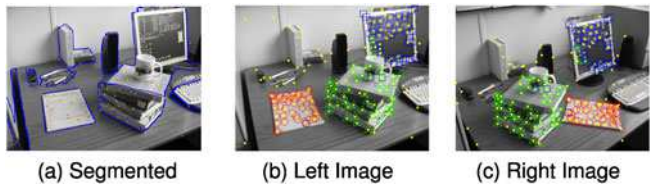


Fig. 12. Detecting multibody structure from motion when the segmentation accuracy is 80 percent and $N = 200$ includes 25 percent gross outliers. For fundamental matrices, the sample sizes are $r = 63$ and $r = 50$ for multinomial and hypergeometric models, respectively. (a) Segmented image. Yellow dots are the correspondences, and red stars are the sampled points using SWIFT. (b) and (c) images with three objects moved independently and the detected motions using the method in [42].

one motion, we applied a basic image segmentation in order to cluster objects of the scene (Fig. 12a). Then the initial candidate homographies and fundamental matrices were calculated using the method in [42], with segmentation used as a proximity constraint. In this example, the average number of initial candidates (homographies and fundamental matrices) generated over 20 trials, were 13,073. Basically, the estimated SWIFT sample size r guarantees with probability $1 - \delta = 0.9$ that the group of candidates includes all the existing motions in the scene.

5.4 Spectral Subspace Clustering

The problem of clustering high dimensional data when it forms multiple subspaces is studied in various areas of machine learning, computer vision, and pattern recognition. In a large variety of applications, data naturally forms clusters of low-dimensional subspaces. Therefore, the main aim of these subspace clustering algorithms is to discover such clusters by finding a sparse representation for each subspace. For instance, in a video of multiple moving rigid objects, the trajectories can be represented by high-dimensional vectors. Yet, they can span low-dimensional linear manifolds [43]. Thus, the goal is to cluster the trajectories of different motions in separate subspaces. Several methods are proposed in this area based on iterative [44], [45], algebraic [46], [47], statistical [48], [49], [50] and spectral clustering [51], [52], [53], [54]. Spectral clustering techniques [55] attempt to construct an affinity matrix to define the similarity between data points, and hence cluster the high-dimensional data. These methods use either local [47], [51] or global information [52], [53], [54] for forming the similarity matrix. Below we show that the Sparse Subspace Clustering (SSC) algorithm presented in [36] and [56] can be more efficiently implemented using SWIFT as a preprocessing step.

The main idea in [36] and [56] is that, by having sufficient number of points in each subspace, any point in a subspace can be represented as a linear combination of other points in that subspace. In other words, if $\{x_i \in \mathbb{R}^n\}_{i=1}^N$ defines a collection of N data points in an ambient space of dimension n drawn from a union of C independent linear subspaces $\{S_j\}_{j=1}^C$ with dimensions $\{d_j \ll n\}_{j=1}^C$, any data point x_i can be represented as $x_i = X_{s_j} Z$, where $x_i \in S_j$, and X_{s_j} are all the data points in S_j except x_i. In a general case, when the population is a union of points from multiple subspaces, x_i is represented as $x_i = X_{\hat{i}} Z_i$ and can be recovered as a sparse solution of the following ℓ_1 optimization problem[36].

$$\min ||Z_i||_\ell \quad \text{subject to} \quad x_i = X_{\hat{i}} Z_i, \tag{20}$$

where $X_{\hat{i}}$ includes all the points in the population of size N except x_i. It is expected that the optimal solution would include non-zero coefficients corresponding to the columns of $X_{\hat{i}}$ that are in the same subspace as x_i. Thus, the coefficient matrices Z_i's can be used to build the affinity matrix and determine the subspace clusters. The SSC method can estimate the cluster with a high level of accuracy [55]. However, it is computationally expensive when we have a very large dataset. An efficient solution to reduce the time complexity of the SSC algorithm would be to start with a subset of points sampled from the dataset[57], [58]. The process of finding sparse coefficients and clustering points can be done in 2 steps. First, find sparse coefficients for a sample set of r points. Second, locate inliers for each cluster. Although a similar idea was explored in [59], their choice of size for the sampled subset was rather ad-hoc. In particular, SSC requires that each subspace of dimension d_j have at least $d_j + 1$ points for the method to work. This is obviously a natural setting for SWIFT, since it guarantees such a minimum subset with high probability. Furthermore, it provides a tight estimate to ensure low time-complexity. Once the subspaces are estimated in the subset sampled by SWIFT,

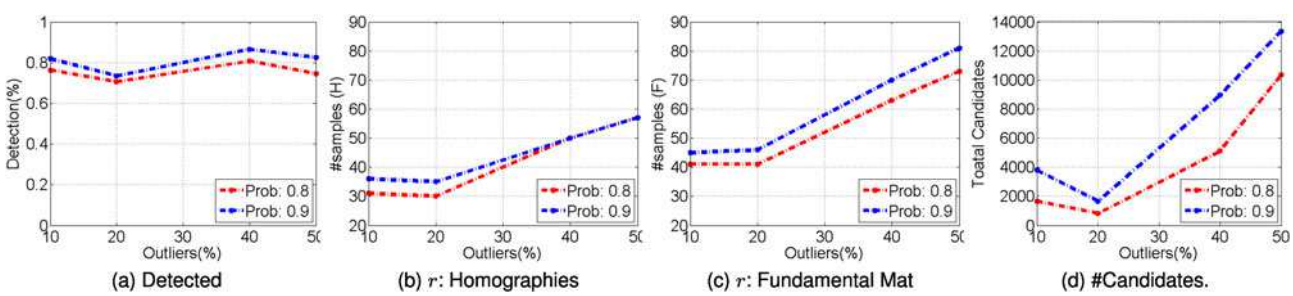


Fig. 11. The effect of changing percentage of outliers on required sampled points and accuracy of multibody structure form motion in [42]. The results are in the presence of three independent motions and when the image segmentation has an average accuracy of 65 percent and $\theta = 40$. (a) % of Detected models. (b) and (c) Values of r when $\varepsilon = \{4, 7\}$, respectively. (d) Initial candidates for both homographies and fundamental matrices.

they can serve as the initial solution to cluster the whole population of data. The proposed SWIFT-SSC is summarized in Algorithm 3.

Density of Subspaces. The study in [60] shows that improving the accuracy of SSC depends on increasing the number of inliers in each subspace as well as decreasing the affinity between subspaces. They show that, under some conditions, increasing the number of inliers of subspaces does not noticeably change the accuracy of the SSC algorithm. This confirms that oversampling in those cases can in fact slow down the clustering process without improving the accuracy of the final result. They define a density parameter $\rho_j = \frac{N_j}{d_j}$ as the ratio of the inliers to the dimensions of the subspaces, and show that one gets a stable level of accuracy when the ratio $\rho \approx 3$. In this experiment, we show that SWIFT sampling can be used to sample sufficient points to detect all the subspaces with a high level of accuracy. Using the study [60], we can estimate the parameter ε. We set $\varepsilon = \rho d + 1$ and show that when $\rho \in \{3, 4\}$, we sample sufficient inliers from each subspace to cluster them with a high accuracy. Oversampling increases the time complexity without noticeable changes in accuracy.

Algorithm 3. Subspace Clustering Using SWIFT

Input: a set of data points $X \in \mathbb{R}^{n \times N}$ and the number of desired clusters C.

1: **SWIFT sampling:** Using N as the population size, C as the number of classes, and $\varepsilon = max_j[d_j]$, sample r points using SWIFT for a choice of $1 - \delta$.
2: **Cluster sampled points:** Cluster the randomly sampled r data points into C classes using the SSC algorithm.
3: **Find inliers:** Using the proposed method in [59], determine the inliers for each cluster in the entire population.

Below, we show that SWIFT sampling indeed attains the optimal number of inliers for each subspace. The data in this experiment included 3 subspaces of dimension $d = 20$ in $\mathbb{R}^{40}$ with 500 points in each subspace. The angles between the subspaces were $\frac{\pi}{3}$, which caused the subspaces to overlap. The results in [60] show that the accuracy of SSC did not change noticeably after $\rho_j = \frac{N_j}{d_j} = 3.25$. In this experiment, to compare the effect of SWIFT sampling, we varied the ratio ρ in the range $\rho \in [1, 7]$. Ground-truth is defined as sampling $\rho \times d$ points from each subspace. SWIFT sample size is computed using $\varepsilon = \rho d + 1$, $N = 4{,}500$, $C = 3$, and $1 - \delta = 0.9$. As demonstrated in Fig. 13, the accuracy is roughly stable when $\rho \in [3, 4]$ and the estimated SWIFT sample size is very close to the ground truth. We show the SWIFT sampling for both the hypergeometric pmf Equations (6), (7), (8) and multinomial pmf Equation (14) models are very close to the ground truth.

Subspaces with Different Dimensions. In the case of subspaces with different dimensions d_j, we are still able to compute the required sample size using the SWIFT algorithm. We need to make sure that a sufficient number of points, $\rho d_j + 1$, are sampled from each subspace. This means that we need to make sure a subspace with the largest dimension, $d_m = \max_j\{d_j\}$, has enough points to be detected. Thus, in the SWIFT algorithm, we can set $\varepsilon = \rho d_m + 1$, which forces (with the probability $1 - \delta$) the sampling process to select ε points from each of the subspaces, even those with lower dimensions. As it turns out, this still gives us a very sparse subset of the points that can be used to detect the subspaces with a very high probability and accuracy. To show the sparsity of the computed sample size in SWIFT, we generated a dataset of 4 subspaces. The subspaces have dimensions $d_j = \{2, 5, 10, 15\}$, with each containing 2,500 points in an ambient space of dimension 50. In order to compute the sample size r, we need to set $\varepsilon = \rho \times d_{\max} + 1$, where $d_{\max} = 15$ is the maximum dimension of the subspaces. Table 2 shows the values of the computed sample sizes using SWIFT and compares them with manual sampling of points, setting $\varepsilon = \rho \times \{2, 5, 10, 15\}$. However, a manual sampling would be only possible if one can accurately distinguish the subspaces to sample the required number of points from each of them. Since, in practice, we do not have such prior knowledge about the subspaces, using SWIFT is the most optimal option. SWIFT selects point uniformly with sufficient number of points in each subspace, even though it adds a small overhead to some subspaces. As presented in Table 2, the computed value of the SWIFT sample size is still very sparse compared to the population size. The ground truth of the optimal sample size using statistical simulation is also added in this table for comparison.

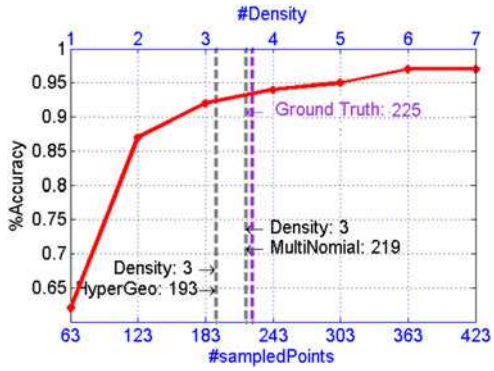


Fig. 13. The effect of changing the ratio of inliers (ρ) on accuracy of clustering. The estimated SWIFT sample size when $1 - \delta = 0.9$ is shown in gray dotted lines. When $\rho = 3$, the actual size of the sampled population needs to be 225 (ground truth), the calculated value by SWIFT sampling using hypergeometric and multinomial pmf models are 193 and 219, respectively. These results were averaged over 100 trials.

Face Clustering as a Subspace Clustering Example. Face clustering is one of the many applications in subspace clustering. Face image clustering techniques try to cluster images of the same subject under varying lighting conditions in one group. Studies in [61], [62], [63] show that a set of images of an object under varying illumination lies in a low-dimensional linear subspace of the image space of up to nine dimensions. This can be used to determine the minimum sample set ε in the SWIFT method. The Extended Yale B Database [64] is a facial dataset widely used in subspace clustering literature [56], [59], [65] and contains 2,414 frontal face images of 38 human subjects taken under approximately 64 different illumination conditions (Fig. 14). In this experiment, we used the SWIFT algorithm to study the sample size required to cluster face images using [36]. We used subsets of 5 different subjects from the dataset. Each subject includes 64 images ($\theta = 64$). By changing the ratio of the inliers (ρ) in the subspaces, we examined the accuracy of the subspace clustering algorithm. The graph in Fig. 15 shows the accuracy of the clustering algorithm as a function of ρ. In addition, the calculated sampled size using the SWIFT algorithm is shown for $\rho = 3$. As we can see in this figure, both multinomial and

TABLE 2
Comparing the Size of SWIFT Sampling, the Manually Sampled Points, and the Ground-Truth

Population Size	ε	Density	#Manually Sampled	#HyperGeo	#Multinomial	Ground Truth
10000	15	1	36	93	93	96
	30	2	68	157	164	167
	45	3	100	221	232	234
	60	4	132	277	299	300
	75	5	164	332	365	366
	90	6	196	381	432	431

In the SWIFT algorithm, $d_{\max} = 15$ *and* $P = 0.9$.

hypergeometric SWIFT estimations give answers that are very close to the ground truth. However, the multinomial pmf model overestimates the sample size, while the hypergeometric model underestimates it, but both are close to the optimal sample size.

6 Discussion and Conclusion

This paper introduces a sparse one-time grab random sampling method, which together with an unsupervised clustering method, such as mean-shift, can be used to simultaneously detect multiple structures or subspaces in a large population of high-dimensional data with overwhelming percentage of outliers. When the data is too big to handle, a popular approach is to sample a subset of the data that remains representative of the whole population. SWIFT provides a generic solution to a well-known question: How big should the sampled "subset" be in order to remain "representative" of the whole population? Application examples are numerous, and some were discussed in the paper. For instance, one case is when a large set of points are tracked across many frames for a moving camera observing several independently moving objects in the scene—a problem known as multi-body structure from motion in computer vision.

We proved that one-time grab random sampling can be accurately modeled using either a hypergeometric or a multinomial pmf (i.e., as either sampling without replacement or with replacement under some constraints). The hypergeometric pmf is the exact mathematical model for one-grab sampling, since it is equivalent to sampling without replacement. We found a non-decreasing relation for estimating the sample size for the hypergeometric model, where one can find the sample size by using a binary search in a time complexity of $O(\log N)$. We carefully derived tight upper bound and lower bound for this model and illustrated experimentally the accuracy. However, when N is too large, finding the sample size can be time consuming. Thus, we derived also an approximation of the random sample size using a multinomial model of pmf (i.e., sampling with replacement assuming $N \gg r$). This approximation can be obtained in $O(1)$ and does not require a searching process. As it is shown, this approximation is not as tight as the estimated sample size through the search process. We carefully studied the behavior and accuracy of these models, providing a practical method of selecting the minimum sample size that guarantees with some probability the detection of all model instances (e.g., subspaces) in the data. In addition to accuracy, one desirable behavior of

Fig. 14. Example of images in the Extended Yale Database B.

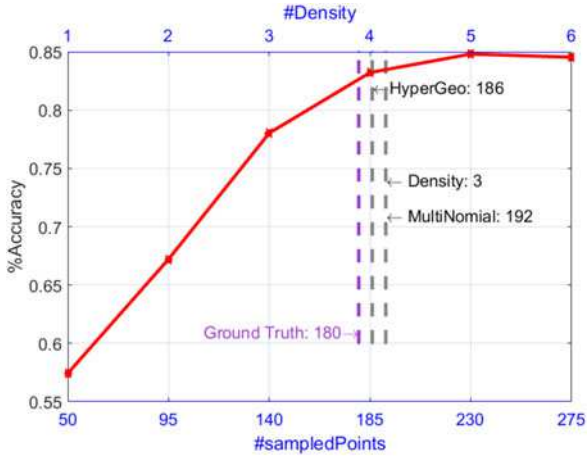


Fig. 15. Accuracy of clustering face images. Computed sample sizes using both multinomial and hypergeometric SWIFT methods are shown in gray lines. In this experiment, $N = 320$, $P = 0.9$, and $C = 5$. Results is averaged over 100 images.

SWIFT sampling is "sparseness", which we have shown is independent of the population size, making our solution important for processing and analyses of "big data".

An important lesson learned in this study is that our solution makes one-grab sampling methods now competitive with multi-grab methods. Popular multi-grab methods such as RANSAC or its variations have long established the answer to their sampling questions such as sample size (per grab) and the number of sampling iterations. Our study now reveals a main disadvantage of multi-grab methods that they have to guarantee sufficient number of samples at each iteration, which over a large number of iterations leads to a huge number of samples. This is in particular is exacerbated when multiple structures need to be discovered. By settling the unanswered sampling question of sufficient sample size for one-grab methods and providing high probability guarantees, we show that this class of methods has a huge advantage of ensuring that a sparse subset of data can solve usual data mining problems in "Big Data" without resorting to heuristics. To demonstrate these points, we compared the sample sizes computed by SWIFT against the number of points sampled in other existing methods. As mentioned earlier, sampling methods, such as [7], sample a batch of ε-tuples, with the total number of sampled points as ε times the number of ε-tuples. This is in certain ways similar to multi-RANSAC. Fig. 16 compares the number of sampled points in SWIFT with the method in [7], and the sequential-RANSAC [11]. The values for sequential-RANSAC are computed using the number of required sampled points to detect each model instance, times the maximum number of instances C [7]. As illustrated, the sample size obtained by SWIFT sampling is significantly smaller than the values in the other two methods.

To conclude, we solve an important problem in sampling a high-dimensional "big data" for mining knowledge, i.e., finding structures, patterns, or subspaces. The key question

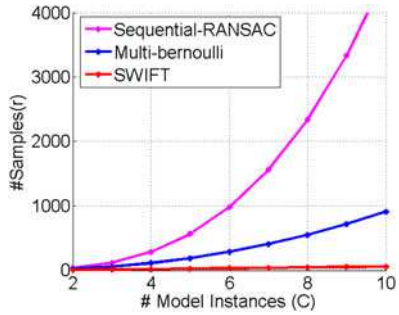


Fig. 16. Comparing the averaged number of samples r over 200 trials in sequential RANSAC, the proposed method (MBSAC) in [7], and SWIFT when $N = \{10^2, 10^3, 10^4\}$, $\varepsilon = 2$, and $P = 0.9$.

of how many samples one should take to ensure with some probability that all subspaces or structures are adequately represented in the sampled subset has been answered in this paper. The problem involves the solution to a hard inverse problem, which we solved by finding tight bounds to the solution. One problem that will need further investigation is the extension of SWIFT to a muti-level or a hierarchical SWIFT method. Note that this idea of multi-level sampling is not the same as sequential sampling, since at each level all the subspaces are sought to be determined. We experimented with this idea in one of the example applications, and the results indicated that this divide-and-conquer approach can further reduce the time complexity of solving these problems by sampling.

Acknowledgments

This work was supported in part by the National Science Foundation under grants IIS-1212948 and DMS-1712977.

References

[1] D. Yan, L. Huang, and M. I. Jordan, "Fast approximate spectral clustering," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 907–916.

[2] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[3] A. Adam, E. Rivlin, and I. Shimshoni, "Ror: Rejection of outliers by rotations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 1, pp. 78–84, Jan. 2001.

[4] A. P. Bustos and T.-J. Chin, "Guaranteed outlier removal for point cloud registration with correspondences," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2017.2773482.

[5] T.-J. Chin, Y. H. Kee, A. Eriksson, and F. Neumann, "Guaranteed outlier removal with mixed integer linear programs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5858–5866.

[6] H. Chen, P. Meer, and D. E. Tyler, "Robust regression for data with multiple structures," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2001, pp. I–1069.

[7] R. Hoseinnezhad and A. Bab-Hadiashar, "Multi-bernoulli sample consensus for simultaneous robust fitting of multiple structures in machine vision," *Signal, Image and Video Processing*, vol. 9, pp. 1–10, 2014.

[8] M. Zuliani, C. S. Kenney, and B. Manjunath, "The multiransac algorithm and its application to detect planar homographies," in *Proc. IEEE Int. Conf. Image Process.*, 2005, pp. III–153.

[9] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm, "USAC: A universal framework for random sample consensus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 2022–2038, Aug. 2013.

[10] P. H. Torr, "Geometric motion segmentation and model selection," *Philosophical Trans. Roy. Soc. London A: Math. Phys. Eng. Sci.*, vol. 356, no. 1740, pp. 1321–1340, 1998.

[11] T. Vincent and R. Laganiére, "Detecting planar homographies in an image pair," in *Proc. 2nd Int. Symp. Image Signal Process. Anal.*, 2001, pp. 182–187.

[12] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: Randomized hough transform (RHT)," *Pattern Recognit. Lett.*, vol. 11, no. 5, pp. 331–338, 1990.

[13] H. S. Wong, T.-J. Chin, J. Yu, and D. Suter, "A simultaneous sample-and-filter strategy for robust multi-structure model fitting," *Comput. Vis. Image Understanding*, vol. 117, no. 12, pp. 1755–1769, 2013.

[14] D. M. Rocke and D. L. Woodruff, "Identification of outliers in multivariate data," *J. Amer. Statistical Assoc.*, vol. 91, no. 435, pp. 1047–1061, 1996.

[15] H. Isack and Y. Boykov, "Energy-based geometric multi-model fitting," *Int. J. Comput. Vis.*, vol. 97, no. 2, pp. 123–147, 2012.

[16] N. Lazic, I. Givoni, B. Frey, and P. Aarabi, "Floss: Facility location for subspace segmentation," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 825–832.

[17] L. Magri and A. Fusiello, "Multiple model fitting as a set coverage problem," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3318–3326.

[18] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov, "Fast approximate energy minimization with label costs," *Int. J. Comput. Vis.*, vol. 96, no. 1, pp. 1–27, 2012.

[19] T.-J. Chin, H. Wang, and D. Suter, "Robust fitting of multiple structures: The statistical learning approach," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 413–420.

[20] W. Zhang and J. Ksecká, "Nonparametric estimation of multiple structures with outliers," in *Dynamical Vision*. New York, NY, USA: Springer, 2007, pp. 60–74.

[21] R. Toldo and A. Fusiello, "Robust multiple structures estimation with j-linkage," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 537–547.

[22] L. Magri and A. Fusiello, "T-linkage: A continuous relaxation of j-linkage for multi-model fitting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3954–3961.

[23] B.-N. Vo, B.-T. Vo, N.-T. Pham, and D. Suter, "Joint detection and estimation of multiple objects from image observations," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5129–5141, Oct. 2010.

[24] T. T. Pham, T.-J. Chin, J. Yu, and D. Suter, "The random cluster model for robust geometric fitting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1658–1671, Aug. 2014.

[25] R. Unnikrishnan and M. Hebert, "Robust extraction of multiple structures from non-uniformly sampled data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, pp. 1322–1329.

[26] S. Birchfield and C. Tomasi, "Multiway cut for stereo and motion with slanted surfaces," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, pp. 489–495.

[27] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[28] M. Jaberi, M. Pensky, and H. Foroosh, "Swift: Sparse withdrawal of inliers in a first trial," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4849–4857.

[29] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Discrete Multivariate Distributions*, vol. 165. New York, NY, USA: Wiley, 1997.

[30] J. A. Rice, *Mathematical Statistics and Data Analysis*, 3rd ed., Thomson Brooks, CA, USA, 2007.

[31] A. Childs and N. Balakrishnan, "Some approximations to the multivariate hypergeometric distribution with applications to hypothesis testing," *Comput. Statist. Data Anal.*, vol. 35, no. 2, pp. 137–154, 2000.

[32] R. W. Butler and R. K. Sutton, "Saddlepoint approximation for multivariate cumulative distribution functions and probability computations in sampling theory and outlier testing," *J. Amer. Statistical Assoc.*, vol. 93, no. 442, pp. 596–604, 1998.

[33] J. E. Kolassa, "Multivariate saddlepoint tail probability approximations," *Ann. Statist.*, vol. 31, pp. 274–286, 2003.

[34] R. C. Bradley, "Central limit theorems under weak dependence," *J. Multivariate Anal.*, vol. 11, no. 1, pp. 1–16, 1981.

[35] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, pp. 1197–1203.

[36] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2790–2797.

[37] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[38] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[39] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: Spectral clustering and normalized cuts," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 551–556.
[40] R. Vidal, Y. Ma, S. Soatto, and S. Sastry, "Two-view multibody structure from motion," *Int. J. Comput. Vis.*, vol. 68, no. 1, pp. 7–25, 2006.
[41] A. W. Fitzgibbon and A. Zisserman, "Multibody structure and motion: 3-D reconstruction of independently moving objects," in *Eur. Conf. Comput. Vis.*, 2000, pp. 891–906.
[42] K. Schindler and D. Suter, "Two-view multibody structure-and-motion with outliers through model selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 983–995, Jun. 2006.
[43] J. Yan and M. Pollefeys, "A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 94–106.
[44] T. Zhang, A. Szlam, and G. Lerman, "Median k-flats for hybrid linear modeling with many outliers," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops*, 2009, pp. 234–241.
[45] P. Tseng, "Nearest q-flat to m points," *J. Optimization Theory Appl.*, vol. 105, no. 1, pp. 249–252, 2000.
[46] T. E. Boult and L. G. Brown, "Factorization-based segmentation of motions," in *Proc. IEEE Workshop Vis. Motion*, 1991, pp. 179–186.
[47] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1945–1959, Dec. 2005.
[48] Y. Ma, H. Derksen, W. Hong, and J. Wright, "Segmentation of multivariate mixed data via lossy data coding and compression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1546–1562, Sep. 2007.
[49] A. Y. Yang, S. R. Rao, and Y. Ma, "Robust statistical estimation and segmentation of multiple subspaces," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, 2006, pp. 99–99.
[50] J. Yan and M. Pollefeys, "Articulated motion segmentation using ransac with priors," in *Dynamical Vision* New York, NY, USA: Springer, 2007, pp. 75–85.
[51] K. Kanatani, "Motion segmentation by subspace separation and model selection," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, pp. 586–591.
[52] G. Chen and G. Lerman, "Spectral curvature clustering (SCC)," *Int. J. Comput. Vis.*, vol. 81, no. 3, pp. 317–330, 2009.
[53] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.
[54] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1615–1622.
[55] R. Vidal, "A tutorial on subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, Mar. 2011.
[56] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.
[57] F. Nie, D. Xu, I. W. Tsang, and C. Zhang, "Spectral embedded clustering," in *Proc. Int. Joint Conf. Artif. Intell.*, 2009, pp. 1181–1186.
[58] L. Wang, C. Leckie, R. Kotagiri, and J. Bezdek, "Approximate pairwise clustering for large data sets via sampling plus extension," *Pattern Recognit.*, vol. 44, no. 2, pp. 222–235, 2011.
[59] X. Peng, L. Zhang, and Z. Yi, "Scalable sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 430–437.
[60] M. Soltanolkotabi, E. J. Candes, et al., "A geometric analysis of subspace clustering with outliers," *Ann. Statist.*, vol. 40, no. 4, pp. 2195–2238, 2012.
[61] K.-C. Lee, J. Ho, and D. Kriegman, "Nine points of light: Acquiring subspaces for face recognition under variable lighting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2001, pp. I–519.
[62] P. N. Belhumeur and D. J. Kriegman, "What is the set of images of an object under all possible illumination conditions?" *Int. J. Comput. Vis.*, vol. 28, no. 3, pp. 245–260, 1998.
[63] J. Ho, M.-H. Yang, J. Lim, K.-C. Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2003, pp. I–11.
[64] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.
[65] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

Maryam Jaberi received the MS degree in computer science and engineering from the University of Nevada, Reno, in 2012 and the PhD degree in computer science from the University of Central Florida, in 2018, where she was a member of the Computational Imaging Lab. Her research interests include machine learning, computer vision, and image/video processing. She was the recipient of the UCF David T. & Jane M. Donaldson Memorial Scholarship and the UNR Redfield School Foundation student award.

Marianna Pensky is a professor with the Department of Mathematics, University of Central Florida (UCF). She has authored and co-authored more than 100 peer-reviewed journal and conference papers. She is an associate editor of the *Journal of the Statistical Planning and Inference* and the *Journal of Nonparametric Statistics* and an elected member of the International Statistical Institute.

Hassan Foroosh (M'02-SM'03) is a CAE Link professor of computer science with the University of Central Florida (UCF). He has authored and co-authored more than 150 peer-reviewed journal and conference papers. He has been serving on the editorial boards and the organizing committees of various IEEE transactions, conferences, and working groups. His research has been supported by the NSF, NASA, ONR, DIA, and various industries. He is a senior member of the IEEE.