

SWIFT: Sparse Withdrawal of Inliers in a First Trial*

Maryam Jaber¹, Marianna Pensky², and Hassan Foroosh¹

¹The Computational Imaging Lab., Computer Science, University of Central Florida, Orlando, FL, USA

²Department of Mathematics, University of Central Florida, Orlando, FL, USA

Abstract

We study the simultaneous detection of multiple structures in the presence of overwhelming number of outliers in a large population of points. Our approach reduces the problem to sampling an extremely sparse subset of the original population of data in one grab, followed by an unsupervised clustering of the population based on a set of instantiated models from this sparse subset. We show that the problem can be modeled using a multivariate hypergeometric distribution, and derive accurate mathematical bounds to determine a tight approximation to the sample size, leading thus to a sparse sampling strategy. We evaluate the method thoroughly in terms of accuracy, its behavior against varying input parameters, and comparison against existing methods, including the state of the art. The key features of the proposed approach are: (i) sparseness of the sampled set, where the level of sparseness is independent of the population size and the distribution of data, (ii) robustness in the presence of overwhelming number of outliers, and (iii) unsupervised detection of all model instances, i.e. without requiring any prior knowledge of the number of embedded structures. To demonstrate the generic nature of the proposed method, we show experimental results on different computer vision problems, such as detection of physical structures e.g. lines, planes, etc., as well as more abstract structures such as fundamental matrices, and homographies in multi-body structure from motion.

1. Introduction and Related Works

Extending robust parameter estimation methods, such as least median of squares (LMedS) [14] and random sampling consensus (RANSAC) [7], to detect multiple model instances simultaneously is an active area of research. One

of the main challenges in a multi-structured environment is the high percentage of outliers. This is due to the presence of two different types of outliers in the population: (i) Gross outliers, which are basically points that do not belong to any model instance, and (ii) Pseudo-outliers, which are points that are inliers to one model instance but effectively act as outliers to all other model instances in the population [2]. Early approaches in identifying multiple model instances estimated models sequentially using a single-model fitting method [7]. At each iteration inliers to a model instance were detected and removed and the process was repeated iteratively [18, 17, 22] for detecting additional model instances. However, the efficiency of these algorithms reduces significantly for multi-structure datasets, since the stopping criterion is usually non-trivial, and the removing of the inliers at each iteration can affect the detection of the remaining model instances. Moreover, an assumption in single-model estimation algorithms is that the outliers are uniformly distributed, which is violated when multiple model instances are present, because they form clusters of points as pseudo-outliers to any given model instance as described above. This issue is discussed in [13], where it is shown that clustered outliers are more difficult to handle than uniformly distributed gross outliers. One of the most popular single-model methods is the sequential RANSAC, which explores one model per iteration, removes the associated inlier points from the dataset, and then proceeds iteratively to detect other instances [21]. As mentioned above, a major drawback of these approaches, in addition to the difficulty of handling clustered outliers, is that the number of model instances needs to be known in advance in order to determine when to stop the iterations. This implies a supervised solution, where the number of model instances is assumed to be known *a priori*, which in many practical cases is not feasible.

More recent studies try to find all the model instances and their parameters simultaneously. Generally, the research in this area is focused on two main aspects: (i) the study of the sampling process, whereby subsets of the data

*This work was supported in part by the National Science Foundation under grants IIS-1212948, IIS-091686, DMS-1106564 and DMS-1407475.

points are selected with the purpose of simultaneously instantiating the underlying model instances, and (ii) the clustering process which is used for grouping the data points based on the information extracted from the sampled subsets, leading to partitioning of the entire data population including the gross outliers, and hence the estimation of the underlying model parameters [6]. A review of the literature in this category of methods reveals that the vast majority of the studies are focused primarily on the second aspect. Nevertheless, among the studies that also consider the sampling process, the primary goal has been statistical robustness, i.e. maximizing inlier selection for improving the breakdown point [19, 16]. This is mainly motivated by the desire to match the performance of RANSAC or RANSAC-like sequential methods that focus on building consensus sets.

The main common thread between most existing methods in terms of the sampling strategy is the idea of sampling based on maximizing some probability of selecting inlier points. This is either achieved by assuming some prior information (e.g. in the form of local neighborhood correlation) [19, 16], or by dividing the sampling set into subsets that are used successively to improve the inlier selections [5, 9]. The former approach is achieved by sampling a collection of minimum sampling sets rather than sampling a collection of points, where the minimum sampling set is typically equal to the number of degrees of freedom of each model instance in the population, e.g. for lines in 2D we would sample point pairs. This has the advantage of making the subsequent clustering step a simpler process. In this paper, our focus is mainly on the sampling step. In fact, our sampling method can be used as the front end to any unsupervised clustering method. Rather than focusing on the strategy of maximizing the probability of sampling inliers, our goal is to minimize the number of samples needed to instantiate all underlying model instances. More specifically, our goal is to answer the following question:

“Given a large population of points with multiple instances of a structure and gross outliers, what is the minimum number of points r to be sampled randomly from this population in one grab, in order to make sure with probability P that there are at least ε samples on each structure instance?”

Here, ε is greater than or equal to the number of samples needed to determine the number of degrees of freedom of the 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. the one-grab sample size is in practice a sparse subset of the population), hence the name, Sparse Withdrawal of Inliers in a First Trial (SWIFT), (ii) although P is an intricate function of r (difficult to invert), we prove that it is a non-decreasing function, and hence r can be

mathematically approximated, and found by a simple one-dimensional search regardless of the dimensionality of the problem, (iii) unlike existing sampling strategies, r is very slowly growing with the minimum model size θ , keeping the one-grab sample set sparse even under overwhelming number of pseudo-outliers (iv) the sparsity of the sampled set implies that a sparse number of putative models can be instantiated with limited computation, which can then be used for a guided clustering of the entire data population, or any subsets of it, (v) finally, robustness is pushed into the clustering step, while keeping the sampled set sparse, which in a sense is the opposite of current practice. This idea of interchanging the role of sampling and clustering in terms of achieving sparseness versus robustness has the advantage that sparseness is attained by a sampling procedure, rather than by sophisticated methods of imposing norm constraints in an optimization scheme.

2. Proposed Method

SWIFT sampling requires three input parameters: the minimum sample set ε per structure, the minimum model size θ , and the probability P of grabbing at least ε points on each structure, where ε is greater than or equal to the number of samples needed to determine the number of degrees of freedom of each model instance, e.g. 2 for a line, and 3 for a circle. The minimum model size, θ , is the minimum number of inlier points necessary to accept a candidate model, i.e. we assume that if a model size is smaller than θ , then we are not interested in extracting it. To derive the SWIFT sampling scheme, 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 all outliers are pseudo-outliers. We call this the worst-case scenario, because it allows us to determine the maximum number of possible model instances C that can be potentially present in the population, which is given by:

$$C = \lceil \frac{N}{\theta} \rceil \quad (1)$$

where $\lceil \cdot \rceil$ rounds the fraction to the nearest upper integer, and N is the total number of points in the data set (i.e. the population size). Next, we derive the SWIFT sampling procedure.

2.1. SWIFT Sampling

As stated earlier, the main aim of the SWIFT sampling is to estimate the sampling size r in a one-time grab, in order to make sure that with a specified probability at least ε points are selected on each model, with the restriction that no model can be smaller than θ points. To solve this problem, we consider a population of size N in which individuals are grouped in C classes of $\theta_1, \dots, \theta_C$ with

$\sum_{i=1}^C \theta_i = N$. Suppose a set of r points is selected randomly in a one-time-grab sampling with x_i points from the i^{th} instance. Then, the probability mass function (pmf) can be modeled by the following multivariate hypergeometric distribution [11]:

$$P[\cap_{i=1}^C (d_i = x_i)] = \frac{\prod_{i=1}^C \binom{\theta_i}{x_i}}{\binom{N}{r}} \quad (2)$$

where $\sum_{i=1}^C x_i = r$ and $0 \leq x_i \leq \theta_i$, ($i = 1, \dots, C$).

Equation (2) expresses the probability of a given sample set in terms of r . However, our goal in SWIFT sampling is actually to solve the inverse problem of finding r for a given probability. For this purpose, we start by recognizing that in our problem, we are dealing with the symmetric case, where $\theta_1 = \theta_2 = \dots = \theta_C = \theta$. As mentioned earlier, in the worst case, for a given N the maximum possible classes would be $C = N/\theta$. Therefore, by substituting $N = C\theta$ in equation (2), we get:

$$P[\cap_{i=1}^C (d_i = x_i)] = \frac{\prod_{i=1}^C \binom{\theta}{x_i}}{\binom{C\theta}{r}} \quad (3)$$

The objective of the method can be equivalently expressed as the problem of finding 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

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

Note that the solution of the above problem is related to the question of finding upper bounds for the tail probabilities of the multivariate hypergeometric distribution (see, e.g. [3, 1, 12]). The usual approach is to use some kind of asymptotic expansion of the multivariate hypergeometric probability, or to approximate it by a multinomial probability. The difference between solution of the inequality (4) and the settings that have been considered previously is that (i) we are interested in constructing a non-asymptotic approach that will work for various parameter settings and (ii) we are interested in the solution of the inverse problem of finding r rather than estimating the probability in the left-hand side of the inequality (4).

Denote $d = r/C$, so that r can be represented as $r = Cd$. Then, the probability of selecting less than ε points from the first model instance d_1 can be calculated as follows:

$$\begin{aligned} \Delta = P(d_1 \leq \varepsilon - 1) &= \sum_{k=0}^{\varepsilon-1} P(d_1 = k) = \sum_{k=0}^{\varepsilon-1} \frac{\binom{\theta}{k} \binom{(C-1)\theta}{Cd-k}}{\binom{C\theta}{Cd}} \\ &= P(d_1 = 0) \left[1 + \sum_{k=1}^{\varepsilon-1} \frac{P(d_1 = k)}{P(d_1 = 0)} \right] \quad (5) \end{aligned}$$

where $Cd - k = \sum_{i=2}^C d_i$. In order to find an upper-bound for the right-hand side of equation (5), we first find an upper-bound for $P(d_1 = 0)$:

$$P(d_1 = 0) = \frac{\binom{(C-1)\theta}{Cd}}{\binom{C\theta}{Cd}} = \prod_{j=0}^{\theta-1} \frac{C\theta - Cd - j}{C\theta - j} \quad (6)$$

If $(C-1)\theta \geq Cd$, then the right-hand side of equation (6) can be further simplified using the inequality $\frac{x-a}{y-a} \leq \frac{x}{y}$ if $a < x < y$:

$$P(d_1 = 0) \leq \prod_{j=0}^{\theta-1} \frac{C\theta - Cd}{C\theta} = \left(1 - \frac{r}{C\theta}\right)^\theta \leq e^{-r/C} \quad (6a)$$

If the value of θ is large while d is relatively small, then the right-hand sides of equation (6) provides an accurate approximation of the exact value of the probability equation (6a).

Now, let us find an upper bound for the ratio $P(d_1 = k)/P(d_1 = 0)$, the exact value of which can be evaluated as follows:

$$\begin{aligned} \frac{P(d_1 = k)}{P(d_1 = 0)} &= \frac{\binom{\theta}{k} \binom{(C-1)\theta}{Cd-k}}{\binom{(C-1)\theta}{Cd}} \\ &= \frac{(Cd)! [(C-1)\theta - Cd]!}{k! (Cd - k)! [(C-1)\theta - (Cd - k)]!} \prod_{j=1}^k (\theta - j + 1) \\ &= \binom{Cd}{k} \prod_{j=0}^{k-1} \frac{\theta - j}{(C-1)\theta - Cd + k - j} \\ &\leq \binom{Cd}{k} \left[\frac{\theta}{(C-1)\theta - Cd + k} \right]^k \quad (7) \end{aligned}$$

If the value of θ is relatively large and the value of ε (and hence k) is relatively small, so that d is much smaller than θ , then the right-hand side of the equation (7) can be further simplified as follows:

$$\begin{aligned} \frac{P(d_1 = k)}{P(d_1 = 0)} &\leq \binom{Cd}{k} \left[\frac{\theta}{(C-1)\theta - Cd + k} \right]^k \\ &\approx \binom{Cd}{k} \left(\frac{1}{C-1} \right)^k \quad (7a) \end{aligned}$$

Therefore, the probability of selecting less than ε points from the first instance of the model d_1 can be bounded above as:

$$\begin{aligned} \Delta = P(d_1 \leq \varepsilon - 1) &\leq P(d_1 = 0) \times \sum_{k=0}^{\varepsilon-1} \frac{P(d_1 = k)}{P(d_1 = 0)} \\ &\leq P(d_1 = 0) \sum_{k=0}^{\varepsilon-1} \binom{Cd}{k} \left(\frac{\theta}{N - r - \theta + k} \right)^k \quad (8) \end{aligned}$$

For $P(d_1 = 0)$, one can use either the exact formula in equation (6) or the approximation in equation (6a). We used equation (6) for better accuracy. Now, the total probability of selecting not less than ε points in each of the model instance can be bounded above using de Morgan’s laws and the symmetry

$$P(\cap_{i=1}^C (d_i \geq \varepsilon)) \geq 1 - \sum_{i=1}^C P(d_i \leq \varepsilon - 1) = 1 - C\Delta \quad (9)$$

Setting $C\Delta = \delta$ and solving equation (8) for r with $\Delta = \delta/C$, we obtain the value of r required for SWIFT sampling. Since equation (9) is a non-decreasing function when C is reasonably small, and because all variables in equations (8) and (9) are known except for r , we can simply find r for the given probability P by using a binary search through all possible values of r between 1 and N .

The accuracy of the estimated bounds are studied in section 3. Once r is estimated, the candidate points are selected randomly as a single one-time grab sampled set. An example of selecting random points over the population is presented in Figure 1. Figure 1a shows a population size of 890 points forming 8 randomly crossing lines and gross outliers. Additionally, Figure 1b shows the population and the candidate points selected with a probability of 0.90 when $\varepsilon = 2$, which is the number of degrees of freedom of a line in 2D.

2.2. Clustering and Parameter Estimation

Once a SWIFT subset of the population is selected, the sampled points are used to estimate the model parameters. Later, we detect the valid model instances in the population, where valid means a model size larger than θ . As pointed out earlier any clustering method may be used as the back-end to our sampling step, e.g. [6, 16, 4, 10]. We experimented with various clustering methods, including k-means, a supervised version of mean-shift, the standard unsupervised mean-shift, and the unsupervised method proposed in [4]. In each case, the clustering was constrained to use only the sparse SWIFT subset to instantiate the models. Due to space limitation, we do not expand further on this aspect. All examples shown hereafter, except the reported results in section 4.1, are using mean-shift as the clustering method, which is a non-parametric unsupervised clustering method that does not require a prior knowledge of the number of clusters, and does not constrain the shape of the clusters (Figure 1c and Figure 1d).

3. Experimental Evaluation

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

3.1. Accuracy of the Proposed Sampling Method

As mentioned in section 2.1, due to the non-decreasing property of equation (9), the required number of sampled points r can be estimated by a simple search, with the time complexity of $O(\log(N))$. However, the success of the proposed SWIFT sampling depends highly on the accuracy of the estimated r . If the estimated r is too small, then some of the model instances will not be detected. On the other hand, if r is highly overestimated, then we lose the sparsity. In essence, by following the worst-case scenario, we are treating the problem as if there were no gross outliers in the population. On the other hand, 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 probability is in practice chosen close to one, ensure that the estimated r is accurate.

To verify this prediction, we investigated the accuracy of our approximation derived in section 2.1 against the theoretical values. We chose different population sizes with different embedded model instances. The result of theoretical and estimated values of r given by equation (9) are plotted against different desired probability values in Figure 2. These plots illustrate the average values of r over 200 independent trials for population sizes of $N = \{100, 1000, 10000\}$.

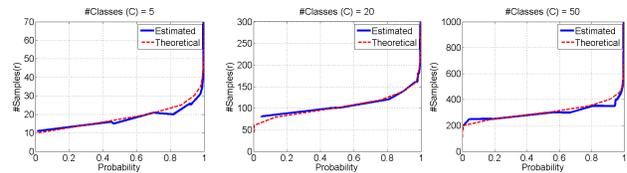


Figure 2: Comparison of estimated r averaged over 200 independent trials versus the theoretical value of r when $N = \{100, 1000, 10000\}$ and $\varepsilon = 2$. From left to right: $C = \{5, 20, 50\}$.

As expected, our approximations closely follow the theoretical values. Moreover, the two facts of using the estimated formula for the probability P and searching through that to find r explain the slight difference between the theoretical and estimated values.

3.2. Evaluation in Terms of Input Parameters

In this section, we study the effect of changing each of the parameters in equation (9) on the estimated value of r .

Value 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 P . This behavior is illustrated in Figure 3a in which the population size, N , and number of

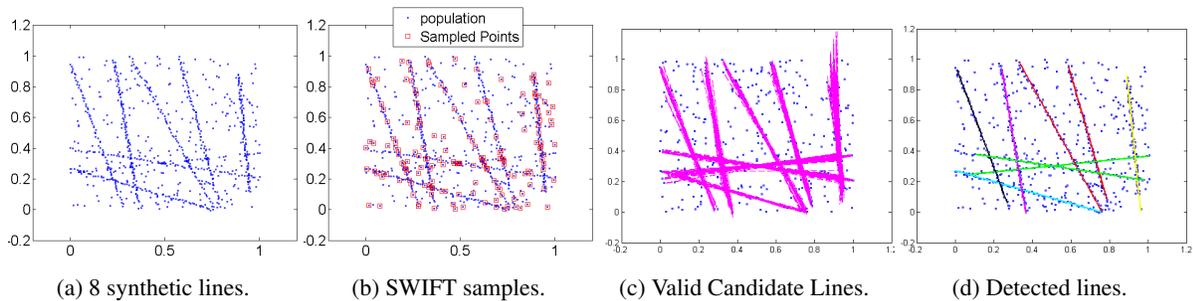


Figure 1: The population includes 8 lines and 50% gross outliers. (b): The calculated sample size of $r = 180$ when $N = 890$, $P = 0.9$ and $\theta = 30$.

model instances C are constant. However, we can see that the growth in r as ε increases is independent of P .

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 also grab more points in order to guarantee, with probability P , ε points on each model instance. This behavior is shown in Figure 3b. Note that, since the relation between θ and C is defined based on equation (1), in a constant N , increasing C is equivalent to decreasing the value of θ . Therefore, a similar behavior is observed for θ in Figure 3c.

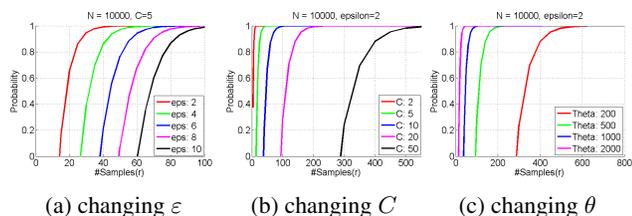


Figure 3: The effect of changing different parameters on the sample size r . (a): Increasing the value of ε forces the method to select more samples in order to remain with the same probability. (b) and (c): increasing C or decreasing θ forces the method to grab more samples to reach the same probability.

A very important observation is that, when the 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*. On the other hand, increasing gross outliers while θ is constant would increase C . In other words, from equation (1), we see that adding more gross outliers increases the worst case estimate for C in equation (9). These cases are reported further in the following section.

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 [9].

4. Applications and Comparisons

As a generic sparse sampling and unsupervised estimation method, SWIFT can be used in virtually any scenario where multiple structures need to be detected in a large population. Here, a population could be in a physical space (e.g. planar or 3D structures), or in some abstract feature space (e.g. the space of all fundamental matrices, or all homographies in some configuration of scene/camera motion).

4.1. SWIFT in 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 [20, 8]. Thus 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 [20, 15]. In this section, we use the method in [15] but replace the sampling step by SWIFT sampling method, and show that it guarantees the accuracy of the final outcome. The first assumption in [15] is that the number and parameters of the motions in the scene are unknown, 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, however, is that one must specify the number of samples in order to ensure detecting all the motions in the scene.

To investigate the effect of using 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 is created. The 3D-objects and the camera are moved randomly and independently and then the second 300×300

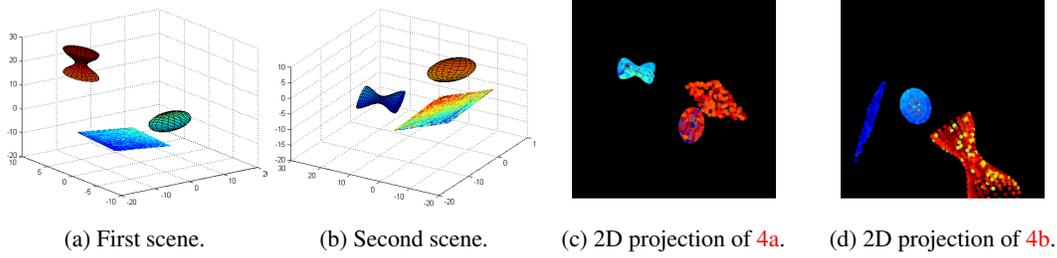


Figure 4: Synthetic data for multibody structure from motion. The outliers are added after moving objects and computing the 2D projections. These outliers are not shown in this figure. (a): The 3D objects that are not necessarily planar. (b): 3D objects are moved randomly and independently. (c): The image is taken from (a). (d): The image is taken from (b) after moving the camera.

image is taken. An example of 3D-objects and their 2D projections are shown in Figure 4. Using the images, point correspondences are 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 [15] divided images heuristically into 9 overlapped 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. We show later that the accuracy of image segmentation does not dramatically affect the final results.

In [15], since they sample a batch of ε -tuples, these sets are sampled separately. In SWIFT, however, since r points are sampled in a one-time grab, we can employ these samples for both homography and fundamental matrices, i.e. since ε for a fundamental matrix is greater than ε for a homography, we can sample r as the number of points required for finding all fundamental matrices. Then a subset of that is enough to find the homographies. In the first experiment, the effect of changing the accuracy of image segmentation is studied. In this experiment, based on the chosen value of θ , the sample size r is computed and grabbed out of the total population $N = 200$, where 50 points (25%) that are gross outlier are added to the correspondences. Sampling r points, the number of inliers sampled in each segment is computed. Figure 5 shows the number of inliers per segment as the accuracy of image segmentation is increasing and considering the homography and fundamental matrices with (a) $\varepsilon = 4$ and (b) $\varepsilon = 7$.

The key advantage of using SWIFT sampling is that the required number of samples to maintain a certain level of accuracy with a given probability can be calculated. Therefore, the accuracy of the results can be maintained stable as illustrated in Figure 6. In this experiment as the number of gross outliers is growing, the size of population N is also increasing based on the SWIFT sampling theory. Since, the

other parameters θ , P and ε are fixed, increasing N leads to selecting a more accurate number of points r . Figure 6a demonstrates this idea of stability of SWIFT sampling in terms of accuracy of results.

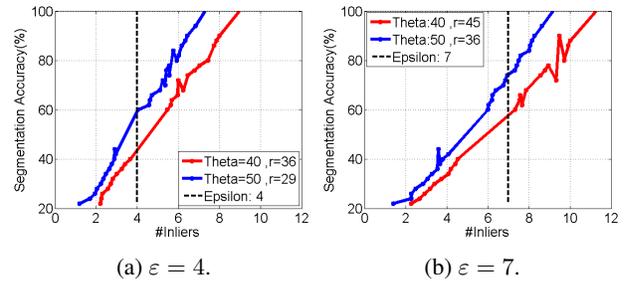


Figure 5: The effect of changing the accuracy of image segmentation on the average number of inliers sampled in each model instance when $N = 200$, $P = 0.9$, with 25% (50 points) outliers and 3 motions. (a): $\varepsilon = 4$ and the value of sampled point r when $\theta = \{40, 50\}$ is $r = \{36, 29\}$ respectively. (b): $\varepsilon = 7$ and the value of sampled point r when $\theta = \{40, 50\}$ is $r = \{45, 36\}$, respectively.

To examine the method on a real case, we used the image presented in [15]. We first applied a basic image segmentation to cluster objects of the scene (Figure 7a). Then the initial candidate homographies and fundamental matrices are calculated using the approach in [15], with segmentation as a proximity constraint. In this example the average number of initial candidates (homographies and fundamental matrices), generated over 20 trials, was 13,073. Basically, this number of samples guarantees with probability P that the group of candidates includes all the existing motions in the scene.

4.2. Two-Level SWIFT to Detect 3D Planes

In this experiment, we investigated the accuracy of SWIFT sampling method for detecting planes in 3D space.

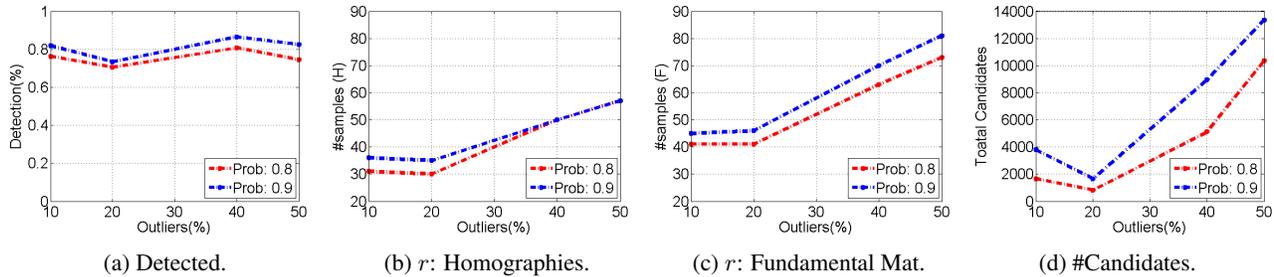


Figure 6: The effect of changing percentage of outliers on required sampled points and accuracy of multibody structure form motion in [15]. The results are in the presence of 3 independent motions. The image segmentation has an average accuracy of 65% and $\theta = 40$. (a): % of Detected models. (b), (c): Values of r when $\varepsilon = \{4, 7\}$ respectively. (d): initial candidates for both homographies and fundamental matrices

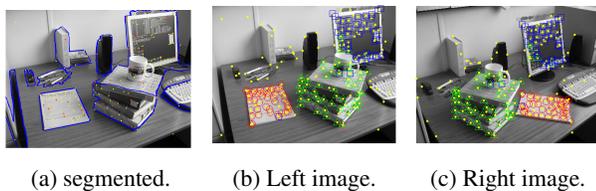


Figure 7: Using SWIFT to detect multibody structure from motion in [15]. The result is in the presence 80% accuracy for image segmentation. $N = 200$ with 25% gross outliers, $\theta = 45$, $P = 0.9$ and $r = \{36, 50\}$ for finding homographies and fundamental matrices respectively. (a): segmented image. Yellow dots are the correspondences and red stars are the sampled points. (b),(c): images with 3 objects moved independently and the detected motions.

In the first step, we considered synthetic models as illustrated in Figure 8. We examined two different scenarios. The first used a set of 3D points from Castelvechio dataset [16] with three planes and no gross outliers (Figure 8a). The second one used a synthetic dataset, with two planes and 50% gross outliers (Figure 8b). As it is shown in Figure 8 the planes are detected correctly and gross outliers are not considered as inliers to any of the instances.

In the second step, we examined real cases where images are collected with a 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 is extremely large. In order to overcome to this problem, the SWIFT method can be applied on two separate levels. On the first level, the value of r is computed to sample minimum required number of points to generate each model candidate (which is 3 for detecting

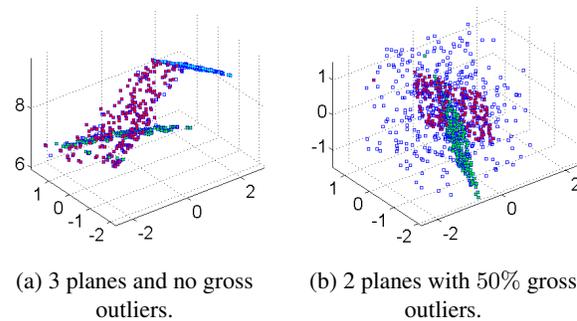


Figure 8: Using SWIFT to detect 3D Planes when $P = 0.9$ and in the presence of 66% outliers, including both pseudo-outliers and gross outliers. (a): Data is from Castelvechio dataset [16] where $r = 42$ and $N = 754$. (b): Blue point are outliers that are not grouped in any model when $r = 43$ and $N = 3000$

planes). Thus, the sample size r is used to generate all the model candidates. On the second level, we can set the value of ε to a bigger number (like 100) and sample a group of points from the population with a guarantee of selecting at least $\varepsilon = 100$ points in each plane. The second group of points can then be used, instead of the entire population, as the group of points from which we select inliers for each model candidate. This two-level process can dramatically decrease the computation time. In Figure 9, the point cloud data from a Kinect is used and three planes are detected using the SWIFT algorithm.

4.3. Comparison in Terms of Sparseness

This section compares SWIFT with other existing methods in terms of sparseness and precision/recall. Since one of the primary goals of SWIFT is sparse sampling, we compared the sample size computed by SWIFT against the number of sampled points in other existing methods. As

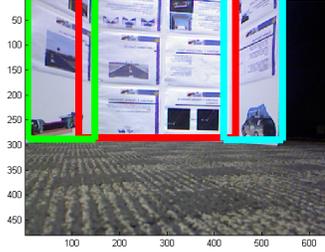


Figure 9: Detecting planes in 3D point cloud data collected using a Kinect. By filtering the points with $depth = 0$ the total number of points in the cloud is $N = 167,028$ and $\theta = 30,000$. By setting $P = 0.9$, the size of sampled point on the first level when $\varepsilon = 3$ is $r = 43$ and on the second level when $\varepsilon = 100$ is $r = 714$.

mentioned earlier, sampling methods such as [9] sample a batch of M ε -tuples (referred to as minimum sampling sets). Thus, the total number of sampled points is $\varepsilon \times M$. Figure 10 shows the result of comparing the computed number of sample points in SWIFT, the method in [9], and the sequential-RANSAC. The illustrated values for sequential-RANSAC is computed using the number of required sample points to detect each model instances times to maximum number of instances C [9]. As demonstrated, the sample size obtained by SWIFT sampling is significantly smaller than the computed value in the two other methods.

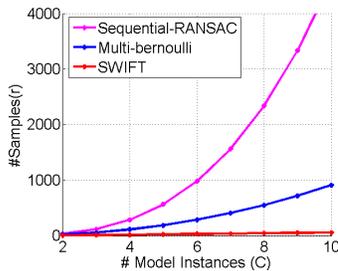


Figure 10: Comparing the averaged number of samples r over 200 trials in sequential RANSAC, the proposed method in [9], and SWIFT when $N = \{100, 1000, 10000\}$, $\varepsilon = 2$ and $P = 0.9$.

4.4. Comparison with Sequential Sampling

Iterative methods, in detecting multiple model instances, remove inliers belonging to a detected model instance before exploring next instances in the population. Thus, the accuracy of their algorithm can be diminished when the points belong to more than one model instance. In this section we compare the impact of SWIFT sampling versus the sequential sampling used in sequential-RANSAC.

We examined the experiment using a set 200 independent images with eight randomly crossing lines and 50%

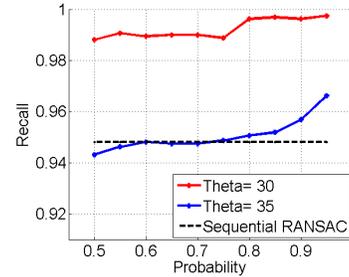


Figure 11: Comparing SWIFT and sequential RANSAC in presence of 50% gross outliers and when $P = \{0.5, \dots, 0.95\}$, $\theta = \{30, 35\}$.

gross outliers. Furthermore, to make a fair comparison, we used a supervised version of mean-shift for clustering valid models sampled by SWIFT. To show the accuracy of SWIFT in compare with sequential RANSAC, we used precision-recall graphs. As shown in Figure 11, the average recall for SWIFT when $\theta = \{30, 35\}$ and the probability is ranging between 0.50 and 0.95 is over 99%. Obviously, the accuracy of sequential RANSAC is steady since it is not a function of the probability (P), nor it is dependent on θ . As illustrated, the accuracy of the proposed method is distinctly superior especially when the $P > 0.70$. Note that, since a supervised clustering is used in this experiment, the value of precision, which is $\frac{TP}{TP+FP}$, is always equal to one as FP is equal to zero.

5. Conclusion

This paper introduces a sparse one-time grab random sampling method, which combined with an unsupervised clustering method, such as mean-shift, can be used to simultaneously detect multiple structures (distributions) in a large population of data with overwhelming percentage of outliers. This important problem occurs frequently in various applications of pattern recognition, machine learning, computer vision, and in general multi-distribution model estimation problems. We prove that the problem can be accurately modeled using a multivariate hypergeometric distribution, and by studying its bounds, we determine a method of selecting the minimum sample size that guarantees with some probability the detection of all model instances in the data.

Thorough analyses of the proposed SWIFT sampling method show that the method has a lot desirable behaviors in terms of accuracy and sparseness that is independent of the population size. We show that the approach outperforms sequential and RANSAC-like methods, and produces the same quality of results compared with the state-of-the-art multi-distribution methods, with the advantage of sparseness, i.e. reducing the sample size to its bare minimum.

References

- [1] R. W. Butler and R. K. Sutton. Saddlepoint approximation for multivariate cumulative distribution functions and probability computations in sampling theory and outlier testing. *Journal of the American Statistical Association*, 93(442):596–604, 1998. [3](#)
- [2] H. Chen, P. Meer, and D. E. Tyler. Robust regression for data with multiple structures. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–1069. IEEE, 2001. [1](#)
- [3] A. Childs and N. Balakrishnan. Some approximations to the multivariate hypergeometric distribution with applications to hypothesis testing. *Computational statistics & data analysis*, 35(2):137–154, 2000. [3](#)
- [4] T.-J. Chin, H. Wang, and D. Suter. Robust fitting of multiple structures: The statistical learning approach. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 413–420. IEEE, 2009. [4](#)
- [5] T.-J. Chin, J. Yu, and D. Suter. Accelerated hypothesis generation for multi-structure robust fitting. In *Computer Vision–ECCV 2010*, pages 533–546. Springer, 2010. [2](#)
- [6] D. Comaniciu and P. Meer. Mean shift analysis and applications. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1197–1203. IEEE, 1999. [2](#), [4](#)
- [7] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. [1](#)
- [8] A. W. Fitzgibbon and A. Zisserman. Multibody structure and motion: 3-d reconstruction of independently moving objects. In *Computer Vision–ECCV 2000*, pages 891–906. Springer, 2000. [5](#)
- [9] 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*, pages 1–10, 2014. [2](#), [5](#), [8](#)
- [10] H. Isack and Y. Boykov. Energy-based geometric multi-model fitting. *International journal of computer vision*, 97(2):123–147, 2012. [4](#)
- [11] N. L. Johnson, S. Kotz, and N. Balakrishnan. *Discrete multivariate distributions*, volume 165. Wiley New York, 1997. [3](#)
- [12] J. E. Kolassa. Multivariate saddlepoint tail probability approximations. *Annals of statistics*, pages 274–286, 2003. [3](#)
- [13] D. M. Rocke and D. L. Woodruff. Identification of outliers in multivariate data. *Journal of the American Statistical Association*, 91(435):1047–1061, 1996. [1](#)
- [14] P. J. Rousseeuw and A. M. Leroy. *Robust regression and outlier detection*, volume 589. John Wiley & Sons, 2005. [1](#)
- [15] K. Schindler and D. Suter. Two-view multibody structure-and-motion with outliers through model selection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(6):983–995, 2006. [5](#), [6](#), [7](#)
- [16] R. Toldo and A. Fusiello. Robust multiple structures estimation with j-linkage. In *Computer Vision–ECCV 2008*, pages 537–547. Springer, 2008. [2](#), [4](#), [7](#)
- [17] B. Tordoff and D. W. Murray. Guided sampling and consensus for motion estimation. In *Computer Vision ECCV 2002*, pages 82–96. Springer, 2002. [1](#)
- [18] P. H. Torr and A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000. [1](#)
- [19] R. Unnikrishnan and M. Hebert. Robust extraction of multiple structures from non-uniformly sampled data. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 2, pages 1322–1329. IEEE, 2003. [2](#)
- [20] R. Vidal, Y. Ma, S. Soatto, and S. Sastry. Two-view multibody structure from motion. *International Journal of Computer Vision*, 68(1):7–25, 2006. [5](#)
- [21] T. Vincent and R. Laganière. Detecting planar homographies in an image pair. In *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, pages 182–187. IEEE, 2001. [1](#)
- [22] M. Zuliani, C. S. Kenney, and B. Manjunath. The multi-ransac algorithm and its application to detect planar homographies. In *Image Processing, 2005. ICIIP 2005. IEEE International Conference on*, volume 3, pages III–153. IEEE, 2005. [1](#)