Balanced Exploration and Exploitation Model Search for Efficient Epipolar Geometry Estimation

Liran Goshen[1] and Ilan Shimshoni[2]

[1] Faculty of Industrial Engineering & Management, Technion, Haifa 32000, Israel
[2] Department of Management Information Systems, Haifa University, Haifa 31905, Israel

Abstract. The estimation of the epipolar geometry is especially difficult where the putative correspondences include a low percentage of inlier correspondences and/or a large subset of the inliers is consistent with a degenerate configuration of the epipolar geometry that is totally incorrect. This work presents the Balanced Exploration and Exploitation Model Search (BEEM) algorithm that works very well especially for these difficult scenes.

The BEEM algorithm handles the above two difficult cases in a unified manner. The algorithm includes the following main features: (1) Balanced use of three search techniques: global random exploration, local exploration near the current best solution and local exploitation to improve the quality of the model. (2) Exploits available prior information to accelerate the search process. (3) Uses the best found model to guide the search process, escape from degenerate models and to define an efficient stopping criterion. (4) Presents a simple and efficient method to estimate the epipolar geometry from two SIFT correspondences. (5) Uses the locality-sensitive hashing (LSH) approximate nearest neighbor algorithm for fast putative correspondences generation.

The resulting algorithm when tested on real images with or without degenerate configurations gives quality estimations and achieves significant speedups compared to the state of the art algorithms!

1 Introduction

The estimation of the epipolar geometry is an important task in computer vision. The RANdom SAmple Consensus algorithm (RANSAC) [1] has been widely used in computer vision in particular for recovering the epipolar geometry.

The estimation of the epipolar geometry is especially difficult in two cases. The first difficult situation is when the putative correspondences include a low percentage of inliers. In such a situation, the number of required iterations is usually high. A popular stopping criterion in a RANSAC like algorithm is

$$I = \log(1-p)/\log(1-\alpha^s) \approx \log(1-p)/\alpha^s, \tag{1}$$

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part II, LNCS 3952, pp. 151–164, 2006.

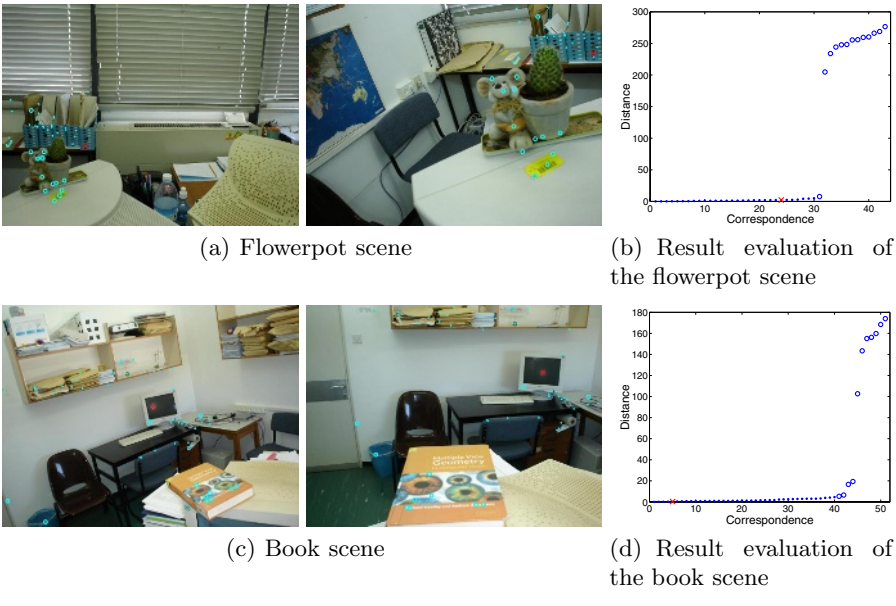
where s is the size of the random sample, I is the number of iterations, α is the inlier rate, and p is the required probability [1, 2]. For example, for $\alpha = 0.15$ the number of needed iterations for $s = 7$ and $s = 2$ are $I = 2,695,296$ and $I = 202$ respectively, for $p = 0.99$.

Several approaches have been suggested to speed-up the RANSAC algorithm. In [3] the random sampling was replaced by guided sampling. The guidance of the sampling is based on the correlation score of the correspondences. PROSAC [4] exploits the linear ordering defined on the set of correspondences by the similarity function used in establishing putative correspondences. PROSAC samples are drawn from progressively larger sets of top-ranked correspondences. LO-RANSAC [5] exploits the fact that the model hypothesis from an uncontaminated minimal sample is often sufficiently near the optimal solution and a local optimization step is applied to selected models. In our previous work [6] the algorithm generates a set of weak motion models (WMMs) and generates an outlier correspondence sample. Using these the probability that each correspondence is an inlier is estimated and enable to guide the sampling. In [7, 8] it was suggested to use three affine region to region matches to estimate the epipolar geometry in each RANSAC sample. Under this framework s in Eq. (1) is changed from seven to three, reducing considerably the number of iterations.

The second difficult situation is when a large subset of inliers is consistent with a degenerate epipolar geometry. This situation often occurs when the scene includes a degeneracy or close to degenerate configurations. In this case standard epipolar geometry estimation algorithms often return an epipolar geometry with a high number of inliers that is however totally incorrect. The estimation of the fundamental matrix in such situations has been addressed before. In [9] a RANSAC-based algorithm for robust estimation of epipolar geometry in the possible presence of dominant scene plane was presented. The algorithm detects samples in which at least five correspondences are consistent with an homography. This homography is then used to estimate the epipolar geometry by the plane and parallax algorithm.

Consider the following two examples. Figure 1(a) shows the flowerpot image scene in which the inlier rate is low and it includes a dominant degenerate configuration. In this scene 17% out of 252 putative correspondences are inliers and 70% of the inliers lie in a small part of the scene which yields a degenerate configuration. A computation of the fundamental matrix based on only inliers from this small space results in a very unstable fundamental matrix. On this scene RANSAC often fails to find the correct fundamental matrix. Figure 1(a) shows a typical result of RANSAC. A dot represents inliers from the degenerate configuration, a circle represents inliers not belonging to the degenerate configuration and the $\times$ represents an outlier that RANSAC detected as inlier. In this example RANSAC succeeded to find all the inliers that belong to the degenerate configuration but failed to find any inliers outside it. This is demonstrated in Figure 1(b), which shows the square root of the symmetric epipolar distance of the inlier from the fundamental matrix. The distances of the inliers outside the degenerate configuration are large. Although, a large number of inliers were

found, the precision of the resulting fundamental matrix is very low. The number of iterations for this scene according to Eq. (1) for $p = 0.99$ is over one million. Figure 1(c) shows another example in which the inlier rate is 16.5% out of 310 putative correspondences and it includes a dominant plane degenerate configuration. In this scene 78% of the inliers lie near the plane. Figure 1(c) shows a typical result of the RANSAC which succeed to find part of the inliers that lie near the plane and failed to find any inliers not close to the plane. As a result, the fundamental matrix is totally incorrect as can be seen in Figure 1(d). The number of iterations for this scene according to Eq. (1) is again over one million.



(a) Flowerpot scene

(b) Result evaluation of the flowerpot scene

(c) Book scene

(d) Result evaluation of the book scene

Fig. 1. Image scenes and quality evaluation

In this paper we propose a novel algorithm for robust estimation of epipolar geometry. The algorithm handles the above two difficult cases in a unified manner. The algorithm can handle not only planar degeneracy, but scenes that include a variety of degeneracies or close to degenerate configurations.

The balanced exploration and exploitation model search (BEEM) algorithm includes a balanced use of three search techniques: global random exploration, local exploration near the current best solution and local exploitation to improve the quality of the model. Moreover, it exploits available prior information, the distance ratio of the closest to second-closest neighbors of a keypoint, to accelerate the search process. Also, it uses the best found model to guide the search process, escape from degenerate models and define an efficient stopping criterion. This is done by indirectly updating the probability that a correspondence is an inlier and by a smart sampling strategy. In addition, a simple and efficient

method to estimate the epipolar geometry from two SIFT correspondences is presented. The matching is sped up using the LSH [10] approximate nearest neighbor algorithm. The generation of the SIFT features can be sped up using the approximation described in [11].

The resulting algorithm when tested on real images with or without degenerate configurations gives quality estimations and achieves significant speedups, especially in scenes that include the aforementioned difficult situations.

2 Exploration and Exploitation

Any efficient search algorithm must use two general techniques to find the global maximum: exploration to investigate new and unknown areas in the search space and exploitation to make use of knowledge found at points previously visited to help find better points. These two requirements are contradictory, and a good search algorithm must strike a balance between them. A purely random search is good at exploration, but does no exploitation, while a purely hill climbing method is good at exploitation, but does little exploration. Combinations of these two strategies can be quite effective, but it is difficult to know where the best balance lies.

Robust estimation of the fundamental matrix can be thought of a search process. The search is for the parameters of the fundamental matrix and the set of inliers. Therefore, algorithms that estimate the epipolar geometry can be analyzed according to the way they combine the above techniques. The RANSAC algorithm [1] samples in each iteration a minimal subset of points and computes from it a model. This random process is actually an indirect global exploration of the parameter space. In the PbM algorithm [12, 13] each exploration iteration is followed by a standard exploitation step. A hill climbing procedure over the parameter space is performed using a local search algorithm. The LO-RANSAC algorithm [5] makes an exploitation step only when a new good model is found in an exploration iteration. The exploitation step is performed by choosing the random sample only from the set of suspected inliers, the model's support set. In cases that there exists a degenerate configuration the exploitation step tends to enlarge the support set but it includes only inliers belonging to the degeneracy.

One disadvantage of the above methods is that they do not have a step similar to the local exploration step that exists in methods like simulated annealing, i.e. even if they find a relatively good model that includes a large number of inliers, they do not use this information after the exploitation step. Once the exploitation step is over, they return to random sampling hoping to find by chance a better model. We suggest to add an intermediate technique that uses the previous best solution and explores its neighborhood looking for a better solution whose support set is larger and includes most of the support set of the previous best solution. To achieve this we need to generate a sample of inliers which includes beside members of the support set other correspondences. Once we have a "good" previous solution it can be assumed that the vast majority of its support set are inliers. Therefore, when choosing a subset for the RANSAC

step, we choose most of the subset from the support set and the rest from points that are outside the support set. When such a subset consists only of inliers the support set of the resulting model tends to break out from the confines of the set of inliers belonging to degeneracy yielding a more correct solution.

When incorporating a local exploration step into the algorithm several questions have to be addressed. First, local exploration is only effective when the best previous support set includes nearly only inliers. So, it is essential to be able to recognize such sets. Second, depending on the quality of the set a balance between the application of global exploration, local exploration and exploitation has to be struck. Finally, how to incorporate available prior information about the quality of each putative correspondence into the general scheme.

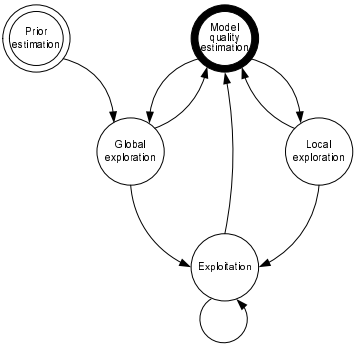


Fig. 2. State diagram of the balanced exploration and exploitation model search (BEEM) algorithm

The BEEM algorithm includes all the above components. Its state diagram is presented in Figure 2. The algorithm includes the following states:

- **Prior estimation.** Use prior available information to estimate the probability that a correspondence is an inlier. This probability is used to guide the sampling in the other states.
- **Global exploration.** Sample a minimal subset of correspondences and instantiate the model from the subset. If the size of the support set of the formed model is larger than all the models that were formed in this state goto the *exploitation* state, otherwise goto to the *model quality estimation* state.
- **Model quality estimation.** Estimate the quality of the best model found until now based on the size of its support set and the number of iterations that the algorithm has performed until now. Use this quality estimate to choose probabilistically the next state, *global exploration* or *local exploration*.
- **Local exploration.** Sample a subset of correspondences from the support set of the best model and sample a subset of correspondences from the rest of the correspondences. Instantiate the model from the union of the two subsets. If the size of its support set is larger than all the models that were

previously formed in this state goto the *exploitation* state, otherwise goto to the *model quality estimation* state.
- **Exploitation.** Iteratively try to improve the last formed model.

The various components of the algorithm are described in the following sections.

3 Using Prior Information of the Match

The best candidate match for each SIFT keypoint from the first image is found by identifying keypoints in the second image whose descriptor is closest to it in a Euclidian distance sense. Some features from the first image will not have any correct match in the second image. Therefore, it is useful to have the ability to discard them. A global threshold on the distance to the closest feature does not perform well, as some descriptors are much more discriminative than others. A more effective measure as suggested by [14] is obtained by comparing the distance of the closest neighbor to that of the second-closest neighbor. This measure performs well because for correct matches the closest neighbor is significantly closer than the closest incorrect match. For false matches, there will likely be a number of other false matches within similar distances due to the high dimensionality of the feature space. We can think of the second-closest match as providing an estimate of density of the false matches within this region of the feature space and at the same time identifying specific instances of feature ambiguity.

Let r_i be the distance ratio of the closest to the second-closest neighbors of the i^{th} keypoint of the first image. Figure 3(a) shows the value of this measure for real image data for inliers and outliers. In [14] it was suggested to reject all matches in which the distance ratio is greater than $r_{tresh} = 0.8$. The probabilistic meaning of this is that each correspondence whose score is below this threshold is sampled uniformly. PROSAC exploits this ratio even more and its samples are drawn from progressively larger sets from the set of correspondences ordered by

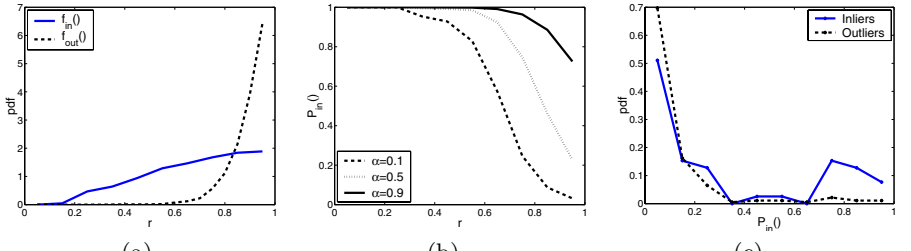


Fig. 3. (a) The empirical distributions of the distance ratio, r, for inlier and outliers were generated based on twenty image pairs. (b) The probability that a correspondence is an inlier as a function of r for several values of the inlier rate, α. (c) The distributions of the estimated probability $P_{in}()$ of the inliers and the outliers, for the book scene image pair.

this ratio. This improves the performance of the algorithm. In this work we make an additional step by giving an empirical probabilistic meaning to this ratio.

The distance ratio can be thought of as a random variable and is modeled as a mixture model:

$$f_r(r_i) = f_{in}(r_i)\alpha + f_{out}(r_i)(1-\alpha),$$

where $f_{in}(r_i) = f(r_i|p_i \leftrightarrow p'_i\ inlier)$, $f_{out}(r_i) = f(r_i|p_i \leftrightarrow p'_i\ outlier)$, and α is the mixing parameter which is the probability that any selected correspondence is an inlier. The probability, $P_{in}(i)$, that correspondence $p_i \leftrightarrow p'_i$ is an inlier can be calculated using Bayes' rule:

$$P_{in}(i) = \frac{f_{in}(r_i)\alpha}{f_{in}(r_i)\alpha + f_{out}(r_i)(1-\alpha)}. \tag{2}$$

We estimate this probability in a non-parametric manner. We generate two samples from real images:

- S_{in}, a sample of $\tilde{N}_{in}$ inlier ratio distances.
- S_{out}, a sample of $\tilde{N}_{out}$ outlier ratio distances.

We estimates $f_{in}()$ and $f_{out}()$ using a kernel density estimator over S_{in} and S_{out} respectively.

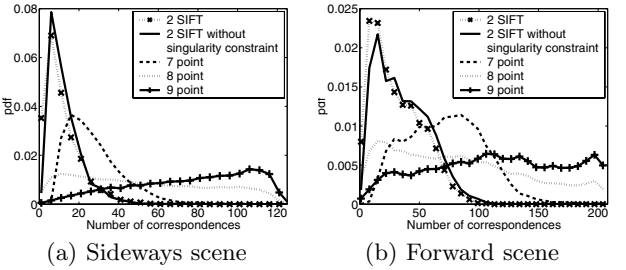
We estimate α for a given image pair using curve fitting of the empirical cumulative distribution function (cdf) of S_{in}, S_{out} and the set of ratios of the putative correspondences. Once α has been estimated $P_{in}()$ can be estimated for all putative correspondences using Eq. (2). Figure 3(b) shows the probability $P_{in}()$ for several values of α. Figure 3(c) shows the distributions of the estimated $P_{in}()$ of the inliers and the outliers, for the book scene image pair. As can be seen in the graph, a large portion of the correspondences that got high probabilities are indeed inliers. In this example the inlier rate is 16.5% and the estimated α is 15.7% which is quite accurate.

4 Epipolar Geometry from Two SIFT Correspondences

In [7, 8] it was suggested to use three affine region to region matches to estimate the epipolar geometry in each RANSAC sample. The novelty here is to use the SIFT descriptor in the computation in a similar manner. The SIFT descriptor is a very powerful descriptor for image matching. This descriptor is invariant to the similarity transformation. The ability to generate epipolar geometry from two SIFT correspondences instead of seven point correspondences is expected to reduce significantly the run-time according to Eq. (1). We suggest a simple method to estimate the epipolar geometry from two SIFT correspondences. Each SIFT keypoint is characterized by its location $p = (x, y)$, orientation θ of the dominant gradient and its scale s. We generate for each SIFT keypoint a set of four points $((x, y), (x + ls\cos(\theta), y + ls\sin(\theta), (x + ls\cos(\theta + \frac{2\pi}{3}), y + ls\sin(\theta + \frac{2\pi}{3}), (x + ls\cos(\theta + \frac{4\pi}{3}), y + ls\sin(\theta + \frac{4\pi}{3}))$. We set $l = \frac{7}{8}\frac{w}{2}$, where w is the width of the descriptor window. Thus, the three additional points lie within

the descriptor window. A set of two SIFT correspondences gives a set of eight point correspondences. These can be used to estimate the fundamental matrix using the linear normalized eight-point algorithm [15]. A SIFT correspondence is consistent with the hypothesized epipolar geometry only when all coincident four point correspondences, $(p_{s1}, p_{s2}, p_{s3}, p_{s4}) \leftrightarrow (p'_{s1}, p'_{s2}, p'_{s3}, p'_{s4})$, are consistent. The location of the first point in the set is quite accurate, whereas, the location of the last three points are less accurate because they are approximated from the SIFT characteristics. We use the error thresholds d for the first point in the set and $d\sqrt{s's}$ for the other three, where s and s' are the scale SIFT parameters of the keypoints of the first and the second SIFT descriptors respectively and d is a threshold parameter.

One may wonder how accurate is the estimation of the fundamental matrix using the 2-SIFT method. The 2-SIFT method generates four point correspondences from each SIFT keypoint. These four points are usually quite close to each other and the last three points are estimated less accurately. Therefore, a fundamental matrix which is based on such point correspondences is expected to be less accurate. To check the severity of this problem, the estimation quality of the 2-SIFT method, 7-point algorithm, normalized 8-point algorithm with 8 and 9 point correspondences were checked. Two types of real scenes without any dominant degenerate configurations were checked: a scene moving sideways and a scene moving forward. For each scene the inlier SIFT correspondences were found. For each algorithm in each scene 10,000 samples were taken from the inlier correspondences. For each sample a fundamental matrix was calculated and the number of correspondences consistent with the model was checked. Figure 4 shows the results. The results of the 2-SIFT method are less accurate than the 7-point algorithm and the 9-point algorithm as expected. However, it usually recovers enough supporting inliers to initialize the fundamental matrix estimation process. Clearly, the use of the exploitation step after the 2-SIFT method is very important. To improve the estimation quality, we checked one more method, the 2-SIFT without singularity constraint (2-SIFT-NSC) method. In this method the singularity constraint of the fundamental matrix is not enforced. The result is usually an illegal model, but in the sample step of the algorithm it is not necessary to work with legal models, because the main purpose of the sample



(a) Sideways scene (b) Forward scene

Fig. 4. Algorithm evaluation

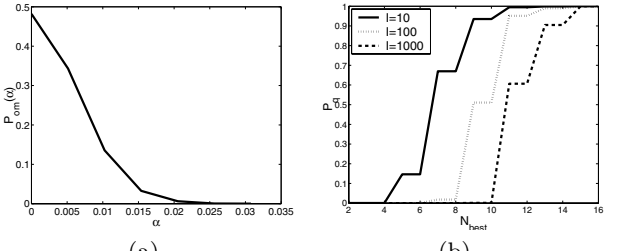
step is to detect large amounts of supporting inliers. The results of the 2-SIFT-NSC method which are also shown in Figure 4 outperform the 2-SIFT method. The reason for this is that the singularity constraint enforcement applied in the 8-point algorithm is not optimal since all the entries of the fundamental matrix do not have equal importance. Note also that the 2-SIFT-NSC method requires less computational cost, because it does not enforce the singularity constraint. For the above reasons we use the 2-SIFT-NSC method in our algorithm.

5 Best Found Model Quality Estimation

In the model quality estimation state the algorithm estimates the quality of the best found model as an inlier model, i.e. a model that nearly all the members of its support set are inliers. When an inlier model is detected it can help accelerate the search process using the local exploration state, whereas using an outlier model in that state is useless. In such situations we want to cause the BEEM algorithm to perform global exploration. To achieve this we have to estimate the probability that the model is supported by outliers that are by chance consistent with it. Let $P_{om}(i)$ be the probability that at most i outliers support an outlier model. Let $N_{best} = \max\{N_i\}_{i=1}^{I}$ be the maximal size of the support set after I iterations achieved by model M_{best}, where N_i is the size of the support set of the i^{th} iteration. Using the above definitions, the probability, P_q, that M_{best} is not an outlier model is estimated. This is equivalent to the probability that in all of the I iterations the support set of size N_{best} could not be achieved by an outlier model. Thus,

$$P_q = \forall_{i=1}^{I} Prob(N_i < N_{best}) = \prod_{i=1}^{I} Prob(N_i < N_{best}) = (P_{om}(N_{best} - 1))^I.$$

The BEEM algorithm uses the probability P_q as an estimate to the quality of the best found model. We estimate $P_{om}()$ using several unrelated image pairs in a non-parametric manner. We ran the algorithm for the above image pairs and



(a) (b)

Fig. 5. (a) The cdf $P_{om}()$ as function of the inlier rate, α. (b) The probability P_q as function of N_{best} for $I = 10$, $I = 100$ and $I = 1000$ where the number of putative correspondences is set to 400.

recorded the size of the support sets of the outlier models. Figure 5(a) shows the cdf $P_{om}()$ as a function of the inlier rate, α. Figure 5(b) shows the probability P_q as function of N_{best} for $I = 10$, $I = 100$ and $I = 1000$, where the number of putative correspondences is set to 400. Note that when the number of iterations increases the "belief" of the algorithm in small subsets decreases. As a result, the algorithm tends to do more global exploration.

6 The Algorithm

Up to this point, we have described the principles of the BEEM algorithm. Now, we will combine them all together, yielding the complete epipolar geometry estimation algorithm. The algorithm is summarized in Algorithm 1. The details of the algorithm are as follows:

Fundamental matrix generation. The generation of the fundamental matrix given a subset S of SIFT correspondences is done as follows: if $2 \leq |S| < 7$ then we use the normalized eight-point algorithm, where each SIFT correspondence provides four point correspondences, as described in Section 4. If $|S| = 7$ then we use the seven-point algorithm with seven points, one from each SIFT

Algorithm 1. The BEEM algorithm.

1: **Prior estimation.**
Estimates α and $P_{in}()$ of the set C of putative correspondences.
2: **Global exploration.**
a) Sample according to $P_{in}()$ a subset of two SIFT correspondences from C.
b) Instantiate the fundamental matrix F.
c) If the support set S of F is the best found in this state then goto *Exploitation* else goto *Model quality estimation*.
3: **Exploitation.**
a) Execute local optimization with inner RANSAC over S until I_l repetitions without improvement.
b) If found model with largest support until now keep its support set in S_{best}.
4: **Model quality estimation.**
a) Estimate P_q.
b) If the *stoping criterion* is satisfied terminate.
c) Choose with probability P_q to goto *Local exploration* else goto *Global exploration*.
5: **Local exploration.**
a) Sample according to $P_{in}()$ a subset of SIFT correspondences from S_{best}.
b) If $P_q < 1$ then sample according to $P_{in}()$ a single SIFT from $C \setminus S_{best}$. else choose the next SIFT correspondence from $C \setminus S_{best}$.
c) Instantiate the fundamental matrix F.
d) If the support set S of F is the largest found in this state then goto *Exploitation* else goto *Model quality estimation*.

correspondence. If $|S| > 7$ then we use the standard normalized eight-point algorithm with $|S|$ keypoints provided from the SIFT correspondences.

Exploitation. This state is very similar to the local optimization method described in [5] with a small improvement. In this state a new sampling procedure is executed. Samples are selected only from the support set S of the previous state. New models are verified against the whole set of putative correspondences. The size of the sample is set to $\min(S/2, N_F)$, where N_F is set to 14. For each fundamental matrix generated from a sample, all the correspondences in its support set are used to compute a new model using the linear algorithm. This process is repeated until no improvement is achieved. The modification we made to the original LO-RANSAC is that whenever a larger support set is found the exploitation process restarts again with it. The algorithm exits this state to the model quality estimation state after ten iterations without improvement.

Local exploration. The parameter space close to the best model found so far is searched in this state by choosing a sample of size $\min(|S_{best}|/2, N_F - 1)$ SIFT correspondences from S_{best} and a single SIFT correspondence from $C \setminus S_{best}$. The fundamental matrix is instantiated from the union of the above subset and the single SIFT correspondence, where the single SIFT correspondence always contributes four point correspondences. This way, the algorithm has a better chance to escape from degenerate configurations.

Once P_q is equal to one, the sampling strategy for correspondences from $C \setminus S_{best}$ changes. Each time a new maximum is found, i.e. S_{best} was updated, the correspondences in $C \setminus S_{best}$ are sorted in decreasing order according to $P_{in}()$. In each iteration a single SIFT correspondence is chosen from $C \setminus S_{best}$ according to the sorting order.

Stopping criterion. The BEEM algorithm terminates if in the last $|C| - |S_{best}|$ exploration samples the subset S_{best} was not updated and if P_q is equal to one in these samples. This criterion ensures with high confidence that nearly all the inliers will be detected. This suggested stopping criterion usually terminates much earlier than in the standard approach, because once the algorithm finds a model with an adequate number of inliers, P_q is estimated as one and the algorithm enters the final local exploration iterations. Because the correspondences in $C \setminus S_{best}$ are sorted in decreasing order according to $P_{in}()$, the rest of the inliers are rapidly found. Once S_{best} ceases to change $|C| - |S_{best}|$ iterations are performed. In the experiments that we have performed, the number of iterations until an adequate number of inliers are found is usually very small, thanks to the various components of the BEEM algorithm. As a result, the total number of iterations of the BEEM algorithm is in practice slightly higher than the number of outliers in the putative correspondence set. This number is much lower than the bound given by Eq. (1).

7 Experiments

The proposed algorithm was tested on many image pairs of indoor and outdoor scenes several of which are presented here. The cases that are presented here are difficult cases in which the inlier rate is low and include a dominant degeneracy.

For each image we applied the SIFT method to detect the keypoints. The descriptors of the first image were then stored in an LSH data structure and the descriptors of the second image were used for querying the data structure to find their approximate nearest neighbors to generate putative correspondences. We used the adapted version of the LSH [16] with data driven partitions. The LSH algorithm is simple for implementation and efficient. For example, the running time for the generation of the putative correspondences of the book scene was reduced from 25.6 seconds using a simple linear search to 0.45 seconds using the LSH algorithm on a Pentium 4 CPU 1.70GHz computer.

For illustration reasons, we divided the set of putative correspondences into three sets: outliers, inliers belonging to the degenerate configuration and the rest of the inliers of which most of them have to be part of the support set in order to generate an accurate fundamental matrix. The images of the scenes are shown in Figures 1 and 6. Their details are given in Table 1.

For each scene six algorithms were tested: the BEEM algorithm, LO-RANSAC using samples of two SIFT correspondences to generate fundamental matrixes (2SIFT LO-RANSAC), RANSAC using samples of two SIFT correspondences (2SIFT RANSAC), LO-RANSAC using samples of seven point correspondences where the samples were sampled according to the probability $P_{in}(i)$ (7pt

(a) Board scene (b) Car scene

Fig. 6. Image scenes

Table 1. The characteristics of the tested scenes. For each scene the table gives the type of degeneracy, number of correspondences, inlier rate, BEEM estimation of the inlier rate, the number of outliers, the number of inliers, the number of inliers belonging to the degeneracy, and the number of inliers not belonging to the degeneracy.

Scene	Degeneracy	N	α	$\hat{\alpha}$	Out.	In.	Deg. In.	Non-Deg. In.
Flowerpot	Small region	252	0.17	0.25	210	42	30	12
Book	Plane	310	0.17	0.16	260	50	44	6
Board	Plane	276	0.27	0.25	201	75	57	18
Cars	Several small regions	272	0.17	0.11	225	47	35	12

Table 2. Experiment results

Algorithm	Success	Iterations	In.	N.Deg.	Success	Iterations	In.	N.Deg.
	Flowerpot scene				Book scene			
BEEM	100%	(5.0) 213	40.6	11.2	95%	(6.3) 279	44.1	5.6
2SIFT LO-RANSAC	30%	356	29.8	3.6	5%	660	27.2	0.6
2SIFT RANSAC	0%	880	16.9	0	0%	2,449	11.2	0.2
7pt P-LO-RANSAC	65%	10,000	34.6	7.9	30%	10,000	35.1	1.8
7pt LO-RANSAC	15%	10,000	27.2	2.4	0%	10,000	19.9	0.2
7pt RANSAC	0%	10,000	19.5	1.2	0%	10,000	16.5	0.5
	Board scene				Car scene			
BEEM	90%	(1.7) 207	72.4	15.6	100%	(2.5) 230	44.8	10.9
2SIFT LO-RANSAC	5%	90	57.8	1.9	30%	533	31.3	5.7
2SIFT RANSAC	0%	1,964	31.9	1.0	0%	1,236	14.8	1.0
7pt P-LO-RANSAC	15%	10,000	61.3	4.9	70%	10,000	39.2	8.2
7pt LO-RANSAC	5%	10,000	57.9	2.1	25%	10,000	27.25	3.9
7pt RANSAC	0%	10,000	53.6	1.1	0%	10,000	18.05	2.3

P-LO-RANSAC), LO-RANSAC using samples of seven point correspondences (7pt LO-RANSAC), and RANSAC using samples of seven point correspondences (7pt RANSAC). The termination criterion for RANSAC and LO-RANSAC was based on Eq. (1), for $p = 0.99$. In cases where the number of iterations exceeded ten thousand the algorithm also terminated. Each algorithm has been applied to each image pair twenty times. For each algorithm the following statistics are presented: the success rate defined as the percentage of the experiments in which at least 75% of the inliers were found and at least 50% of the inliers outside the degenerate configuration were found, the number of iterations until the termination of the algorithm, the number of inliers found, and the number of inliers outside the degenerate configuration found. For the BEEM algorithm, in the iteration column the number of global exploration iterations is also given denoted in parentheses.

The results clearly show that the BEEM algorithm outperforms the other algorithms in the way that it deals with degeneracies, detecting almost always most of the inliers outside of the degenerate configuration. The quality of the results as represented by the overall number of detected inliers is also much higher. Finally, the number of iterations until termination of the algorithm is much lower than for the other algorithms. Finally, the number of global exploration iteration of the BEEM algorithm is very low as a result of the use of the prior information and the 2-SIFT method. As mentioned in the previous section, the number of iterations of the BEEM algorithm is in practice slightly higher than the number of outliers in the putative correspondence set. This number is much lower than the number of iterations of the other algorithms. The results of the other algorithms demonstrate the contribution of each component of the BEEM algorithm to the quality of the detection.

References

1. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comm. of the ACM **24** (1981) 381–395
2. Torr, P.: Motion segmentation and outlier detection. In: PhD thesis, Dept. of Engineering Science, University of Oxford. (1995)
3. Tordoff, B., Murray, D.: Guided sampling and consensus for motion estimation. In: European Conference on Computer Vision. (2002) I: 82–96
4. Chum, O., Matas, J.: Matching with PROSAC: Progressive sample consensus. In: CVPR. (2005) I: 220–226
5. Chum, O., Matas, J., Kittler, J.: Locally optimized RANSAC. In: German Pattern Recognition Symposium. (2003) 236–243
6. Goshen, L., Shimshoni, I.: Guided sampling via weak motion models and outlier sample generation for epipolar geometry estimation. In: CVPR. (2005) I: 1105–1112
7. Schaffalitzky, F., Zisserman, A.: Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In: ECCV. (2002) I: 414–431
8. Chum, O., Matas, J., Obdrzalek, S.: Enhancing RANSAC by generalized model optimization. In: ACCV. (2004) II: 812–817
9. Chum, O., Werner, T., Matas, J.: Two-view geometry estimation unaffected by a dominant plane. In: CVPR. (2005) I: 772–779
10. Gionis, A., Indyk, P., Motwai, R.: Similarity search in high dimensions via hashing. In: ICVL. (1999) 518–529
11. Grabner, M., Grabner, H., Bischof, H.: Fast approximated sift. In: Asian Conference on Computer Vision. (2006) 918–927
12. Chen, H., Meer, P.: Robust regression with projection based m-estimators. In: International Conference on Computer Vision. (2003) 878–885
13. Rozenfeld, S., Shimshoni, I.: The modified pbM-estimator method and a runtime analysis technique for the ransac family. In: CVPR. (2005) I: 1113–1120
14. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
15. Hartley, R.: In defense of the eight-point algorithm. IEEE Trans. Patt. Anal. Mach. Intell. **19** (1997) 580–593
16. Georgescu, B., Shimshoni, I., Meer, P.: Mean shift based clustering in high dimensions: A texture classification example. In: ICCV. (2003) 456–463