Matching with GUISAC-Guided Sample Consensus

Hengyong Xiang\textsuperscript{1,††}, Li Zhou\textsuperscript{†}, Xiaohui Ba\textsuperscript{1,††a}, Nonmembers, and Jie Chen\textsuperscript{†}, Member

SUMMARY The traditional RANSAC samples uniformly in the dataset which is not efficient in the task with rich prior information. This letter proposes GUISAC (Guided Sample Consensus), which samples with the guidance of various prior information. In image matching, GUISAC extracts seed points sets evenly on images based on various prior factors at first, then it incorporates seed points sets into the sampling subset with a growth function, and a new termination criterion is used to decide whether the current best hypothesis is good enough. Finally, experimental results show that the new method GUISAC has a great advantage in time-consuming than other similar RANSAC methods, and without loss of accuracy.

\textbf{key words:} RANSAC, seed points, growth function, termination criterion

1. Introduction

Initial correspondences across images are achieved by matching hand-crafted features. In many cases, those correspondences only based on the similarity of local feature is not reliable, since the hand-crafted features can not perform well in all scenes. Therefore, how to remove bad correspondences from thousands of correspondences between images has become a subject worth studying.

Random sample consensus (RANSAC) \cite{1} is proposed by Fischler and Bolles for the task of model fitting with obvious outliers. It is a classic algorithm that could remove bad correspondences across images and can be divided into three steps. At first, it will sample a smallest subset randomly from all correspondences and estimate a homography or fundamental matrix hypotheses; Then it will verify the above result with remaining correspondences, this step will get a confidence of the hypotheses; Thirdly, it will repeat two steps above until the criterion of termination is met; Finally, correspondences that don't conform to the best hypothesis will be removed. Up to now, many improvement have been applied to increase the performance of RANSAC. The key point of this letter is the improvement of the sampling method of RANSAC. The efficiency of RANSAC could be increased greatly if a correct hypothesis is found early on. In many practical applications, the hypothesis generated from the sampling with prior information is more likely to be correct, and sampling completely random from a dataset which the traditional RANSAC does is not efficient because the correct hypothesis often comes late. Thus how to sample with prior information to meet a correct hypothesis early is a main research direction for sampling at present. N-Adjacent Points Sample Consensus (NAPSAC) \cite{2} exploiting the fact that inliers in the dataset tend to be clustered together and outliers tend to be distributed uniformly. In NAPSAC, a data point $c_0$ is selected randomly at first, then the region $R_{c_0}$ centered on $c_0$ with radius $r$ is divided and data points in $R_{c_0}$ are sampled uniformly if the size of the smallest subset is met. Progressive Sample Consensus (PROSAC) \cite{3} uses the observation that correspondences with better quality (distance ratio for a correspondence) is more reliable. In PROSAC, the sampling will be drawn on the subset consists of correspondences whose quality is higher, then the subset will be expanded until it meets the criterion of termination.

This letter proposes a new sampling algorithm to improve the performance of RANSAC in image matching, and we call it GUISAC (Guided Sample Consensus). In GUISAC, various prior information is integrated to increase the probability that a hypothesis generated from sampling is correct for the efficiency of RANSAC.

2. Guided Sample Consensus

In this letter, initial correspondences are established by hand-crafted features such as SIFT \cite{4} and ORB \cite{5}. Suppose that $N$ initial correspondences $C = \{(x_i, y_i)\}_{i=1}^N$ are achieved based on the similarity of SIFT feature descriptors between two images, where $x_i$ and $y_i$ are the position of the $i$th correspondence’s two feature points in two images. For every correspondence in $C$, a distance ratio (DR) is computed firstly, that is

\begin{equation}
\text{DR} = \frac{\text{dist}_\text{nearest}}{\text{dist}_\text{sec-nearest}}
\end{equation}

where $\text{dist}_\text{nearest}$ and $\text{dist}_\text{sec-nearest}$ are the distance between a feature point descriptor and its nearest or second nearest neighbor in the second image. Then orientation difference (OD) and scale ratio (SR) are computed for each correspondence, they are

\begin{equation}
\text{OD} = \text{Ori}_\text{first} - \text{Ori}_\text{second}
\end{equation}

\begin{equation}
\text{SR} = \frac{\text{Scale}_\text{first}}{\text{Scale}_\text{second}}
\end{equation}

where $\text{Ori}_\text{first}$ and $\text{Ori}_\text{second}$ are the orientation of the
first feature point and second feature point in one correspondence, and the value of OD is limited between -180 to 180 degrees. Scale\textsubscript{first} and Scale\textsubscript{second} are the diameter of the neighborhood from which the first feature point and second feature point in one correspondence are extracted.

GUI SAC could be used to remove bad correspondences in \( C \) between two images. It improves the sampling of RANSAC, various prior information such as the distance ratio, the orientation, the scale, and the local spatial constraint of correspondence is integrated into the sampling. The process of GUI SAC is as follows: firstly, a limited number of seed data points which are correspondences with low distance ratios are well distributed in two images. Secondly, seed data points extracted from areas centered on seed data points are scored by the orientation and scale constraints of data points (correspondences). Finally, the subset for sampling is initiated by correspondences in the seed point set with the highest score, and new seed point set are added in the subset gradually sorted by score. In the following content, GUI SAC will be introduced from three parts: first, the seed points sets selection which introduces how to generate the sampling subset; second, the growth function which defines how fast the sampling subset should grow; third, The termination criterion of GUI SAC.

2.1 Seed Points Sets Selection

In GUI SAC, to increase the efficiency of sampling, the samples are not drawn from all data but a subset that is more reliable. In this letter, the subset consists of seed points sets, and this section will tell how to generate seed points sets.

We suppose that keypoint in a local region should be extracted from the same 3d structure, thus correspondences between two images in a neighborhood should have something in common. For a correspondence \( c_i = \{(x_i, y_i)\}, c_j \in C \), if a correspondence \( c_j = \{(x_j, y_j)\}, c_j \in C \) satisfy the following three constraints, we call that \( c_j \) is the supporting correspondence of \( c_i \).

\[
\|x_j - x_i\| \leq r_1 \land \|y_j - y_i\| \leq r_2 \tag{4}
\]

\[
|OD_j - OD_i| \leq t_{OD} \tag{5}
\]

\[
\left|\ln\left(\frac{SR_j}{SR_i}\right)\right| \leq t_{SR} \tag{6}
\]

Where \( r_1 \) and \( r_2 \) are the radii of \( c_i \)'s local region, and Eq. (4) is the local spatial constraint of a seed correspondence with its supporting correspondences. \( t_{OD} \) is the threshold of orientation difference of a seed correspondence with its supporting correspondences, \( t_{SR} \) is the threshold of scale difference, and formula(5) and formula(6) are orientation and scale constraints of a seed correspondence. The final confidence of a seed correspondence is distinguished by those constraints. For the value of thresholds, the smaller the threshold, the fewer supporting correspondences of a seed correspondence will be found. To provide a reference, \( r_1 \) and \( r_2 \) could be set by formula(7), \( t_{OD} \) could be set to 30 degrees and \( t_{SR} \) could be set to 1.5, but they are not strictly necessary.

In the details of implementation, seed points are detected at first. Seed points should be more reliable and well distributed, and this could be implemented with high effectiveness by a local non-minimum suppression over the distance ratio. Thus correspondences with the lowest distance ratio within a radius \( r \) are selected finally. The next step is to generate a seed point set with a confidence score for every seed point. Actually the seed point set consists of seed point’s supporting correspondences, and the score of this seed point set is the ratio of the number of correspondences in the seed point set to the area of the neighborhood of seed point. To find a neighborhood of a seed point quickly, a simple fast approximate nearest neighbors algorithm [6] is used. Besides, \( r \) is determined by

\[
r = \sqrt{\frac{HW}{\pi\pi}} \tag{7}
\]

where \( H \) and \( W \) are the height and width of an image, and \( n \) is the number of seed points which usually depends on the number of total correspondences. What end up with are seed points sets \( S = [s_i]_{i=1}^n \), where \( s_i \) is the set of the \( i \)th seed point, and the number of correspondences in \( s_i \) is \( n_i \). Seed points sets in \( S \) are sorted in descending order with respect to the score.

2.2 The Growth Function and Sampling

\( C \) is a set of \( N \) data points. \( m \) is the size of each sample. In traditional RANSAC, there are \( \binom{N}{m} \) different samples. In the sample-and-test cycles, ideally as long as one all-inliers sample, the iteration will terminate. In GUI SAC, the samples are drawn from seed points sets which are more reliable, in order to meet an all-inliers sample early. At the initial moment, the sampling subset is \( s_1 \) which is the seed point set with the highest score. There are \( \binom{n_1}{m} \) different samples now, and a new seed point set should be added to the sampling subset when the number of samples exceeds this value. The expansion of the sampling subset is similar to PROSAC, thus almost the same growth function is used after the sampling set is initialized. Let \( T_k \) be an average number of samples from the current sampling subset which is a set of \( k \) data points.

\[
T_k = T_N \left(\frac{k}{m}\right) = T_N \prod_{i=0}^{m-1} \frac{k - i}{N - i} \tag{8}
\]

where \( T_N \) is an average number of samples from the whole dataset, and according to the above formula, \( T_{k+1} \) and \( T_{k+1} \) could be recursively defined as

\[
T_{k+1} = \frac{k + 1}{k + 1 - m} T_k \tag{9}
\]
\[ T_{k+1} = \prod_{i=1}^{l} \frac{k+i}{k+i-m} T_k \]  (10)

At first, when \( k < m \), \( T_k = 0 \), and with \( T_m = 1 \), the value of \( T_k(k > m) \) can be derived. Then if the value is not integer, we define \( T_{m+1} \), and

\[ T'_k = T'_m + \lceil T_k - T_m \rceil \]  (11)

The final growth function is defined as

\[ g(t) = \min \{ k : T'_k \geq t \} \]  (12)

where \( t \) is the \( r \)th sample. The sampling subset can be expanded at the right time according to the above formula.

2.3 The Termination Criterion

Traditional RANSAC is terminated when the probability of that at least an all-inliers sample has already happened is bigger than \( \eta \) (typically set to 95%). Suppose that \( \beta \) is the inlier ratio, and the probability of an all-inliers sample is approximately equal to \( \beta^m \). Thus

\[ (1 - \beta^m)^i < 1 - \eta \]  (13)

where \( i \) is the number of iteration. the formula could be further transformed to

\[ t > \frac{\log(1 - \eta)}{\log(1 - \beta^m)} \]  (14)

when \( t \) satisfies formula(14), we can assume that an all-inliers sample has already occurred (probability is more than \( \eta \)).

In the implementation of RANSAC, \( \beta \) is set to the inlier ratio of the current optimal hypothesis, but GUISAC is not sampling uniformly from all data but a subset which is more reliable, thus the probability of an all-inliers sample is much higher than \( \beta^m \). In this letter, the termination criterion above is conservative and not applicable to GUISAC. To reduce the number of unnecessary iterations, we try to relax the restriction of the termination criterion. The new termination criterion is set to

\[ t > \frac{\log(1 - \eta)}{\log(1 - \beta + \gamma \beta^m)} \]  (15)

where \( \gamma \in [0, 1 - \beta] \) is used to reduce the number of iteration. It can be seen that the range of \( \gamma \) is not fixed, thus \( \gamma = \alpha(1 - \beta) \) is used in the following experiment, where \( \alpha \in [0, 1) \) has a fixed range and is easy for parameter adjustment.

The final termination criterion is

\[ t > \frac{\log(1 - \eta)}{\log(1 - \beta + \alpha(1 - \beta)^m)} \]  (16)

In the end, the outline of the GUISAC is as follows which is based on the assumption \( k > m \).

3. Performance Evaluation

The purpose of this chapter is to test the method in this letter. For the following experiment, it is necessary to explain in advance. First, experiments are carried out on the VGG dataset[7], which contains nine sequences with five different changes in imaging conditions: viewpoint changes, scale changes, image blur, JPEG compression, and illumination, and image pairs in each sequence are evenly distributed from simple to difficult. Second, we construct the initial 2000 correspondences between images with the SIFT algorithm. Third, with the homography matrix ground-truth between a pair of images, the threshold between inlier and outlier is set to 1 pixel. Fourth, the experimental environment is Ubuntu 18.04, OpenCV 3.4.1, the processor is Intel (R) Core (TM) i7-9750h, no multi-threading acceleration.

3.1 The Test of Parameter \( \alpha \)

At first, we try to verify the influence of the termination parameter \( \alpha \) in formula(16) to GUISAC. In all image pairs of VGG dataset, the average precision, recall, number of iterations and time of GUISAC are plotted as the function of parameter \( \alpha \) in Fig. 1. It can be seen that the number of iterations and time of GUISAC get smaller as \( \alpha \) gets bigger, recall, and precision of GUISAC almost do not change until \( \alpha \) is bigger than 0.8. This shows that GUISAC could achieve a great result only in a few iterations.

3.2 Compare with Other RANSAC Algorithms

In this section, we will compare GUISAC with other RANSAC sampling algorithms which are the traditional RANSAC, NAPSAC, and PROSAC. At the same time, GUISAC is composed of three techniques: seed points set selection, growth function, and termination criterion, and the contribution of termination criterion has been show in the 3.1. To show contributions of seed point set selection and growth function, we test GUISAC without growth function (GUISAC no GF) and GUISAC without growth function and seed point set selection (GUISAC no GF and SP). All methods are implemented through USAC (a universal framework for RANSAC)[8]. \( \alpha \) of GUISAC is set to 0.7,
Sample Consensus) is proposed to reject false matches between images. The sample and termination criterion of traditional RANSAC is changed. Firstly GUISAC extracts seed points and generates seed point set, then expands the sampling subset with a growth function; Finally, a new termination criterion is used to decide whether the current best hypothesis is good enough. In the last, experimental results show that the new method could generate a good enough hypothesis in a few iterations, it has a great advantage in time-consuming than other similar algorithms.

**Acknowledgements**

This work was supported by the National Key R&D Program of China (2019YFB2204200), NSFC (U1832217), and the Young Talent Supporting Program of the China Association for Science and Technology (Grant No. 2019QNRC001).

**References**

[1] 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, vol.24, no.6, pp.381–395, 1981.

[2] D.R. Myatt, P.H.S. Torr, S.J. Nasuto, J.M. Bishop, and R. Craddock, “NAPSAC: high noise, high dimensional robust estimation,” British Machine Vision Conference (BMVC), pp.458–467, 2002.

[3] O. Chum, and J. Matas, “Matching with PROSAC - progressive sample consensus,” 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.1, pp.220–226, San Diego, USA, 2005.

[4] D.G. Lowe. “Distinctive image features from scale-invariant keypoints.” International Journal of Computer Vision, vol.60, no.2, pp.91–110, 2004.

[5] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” IEEE International Conference on Computer Vision (ICCV), pp.2564–2571, Barcelona, Spain, Nov. 2011.

[6] D. Aiger, E. Kokiopoulou, and E. Rivlin. “Random grids: Fast approximate nearest neighbors and range searching for image search,” Proceedings of the IEEE International Conference on Computer Vision, pp.3471–3478, 2013.

[7] K. Mikolajczyk, and C. Schmid, “A performance evaluation of local descriptors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.27, no.10, pp.1615–1630, Madison, USA, 2005.

[8] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.M. Frahm, “USAC: a universal framework for random sample consensus,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.35, no.8, pp.2022–2038, 2013.