Good Feature Selection for Least Squares Pose Optimization in VO/VSLAM

Yipu Zhao[1] and Patricio A. Vela[1]

Abstract— **This paper aims to select features that contribute most to the pose estimation in VO/VSLAM. Unlike existing feature selection works that are focused on efficiency only, our method significantly improves the accuracy of pose tracking, while introducing little overhead. By studying the impact of feature selection towards least squares pose optimization, we demonstrate the applicability of improving accuracy via good feature selection. To that end, we introduce the *Max-logDet* metric to guide the feature selection, which is connected to the conditioning of least squares pose optimization problem. We then describe an efficient algorithm for approximately solving the NP-hard *Max-logDet* problem. Integrating *Max-logDet* feature selection into a state-of-the-art visual SLAM system leads to accuracy improvements with low overhead, as demonstrated via evaluation on a public benchmark.**

I. Introduction

Least squares optimization techniques, such as Gauss-Newton and Levenberg-Marquardt methods, are widely used for optimizing camera pose in state-of-the-art VO/VLAM systems for robotics (e.g. ORB-SLAM[1], SVO[2], DSO[3]). Unfortunately, least squares are sensitive to perturbations in the source data. Incorporating robust influence functions mitigates this problem, but does not completely suppress the induced error. In VO/VSLAM, the perturbations from both measurements (e.g. noisy features/patches) and references (e.g. inaccurate mapping) negatively affects pose optimization with least squares techniques. Regarding accurate pose optimization, not all features/patches being matched contribute the same. If only those valuable towards accurate pose estimation are utilized, the total amount of noise introduced into the least squares can be reduced, while preserving the conditioning of the optimization problem.

The idea of enhancing the performance of VO/VSLAM with feature selection is not novel. Conventionally, fully data-driven and randomized methods such as RANSAC are used to reject outlier features [4]. Extensions to RANSAC improve its computational efficiency [5], [6]. These RANSAC-like approaches are utilized in many VO/VSLAM systems [1], [4], [7]. However, the scope of this paper is on "inlier selection", which differs from outlier rejection: outlier rejection aims to remove clearly wrong matches, while "inlier selection" aims to identify valuable inlier matches from useless ones. The two aspects are complementary. A high-level overview of inlier-selection in SLAM can be found in [8].

[1]Yipu Zhao `yipu.zhao@gatech.edu` and Patricio A. Vela `pvela@gatech.edu` are with Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA.

This work was supported, in part, by the National Science Foundation under Grant No. 1400256 and 1544857.

Image appearance has been commonly used to guide inlier selection: feature points with distinct color/texture patterns are more likely to get matched correctly [9]–[11]. However, these works solely rely on quantifying distinct appearance, while the structural information of the 3D world and the camera motion are ignored. While appearance cues are important in feature selection, the focus of this paper is on the latter properties: selecting features based on structural and motion information. These two complementary approaches should be combined into a general feature selection methodology.

To exploit structural and motion information, covariance-based feature selection methods are studied. The pose covariance matrix 1) contains both structural and motion information implicitly, and 2) approximately represents the uncertainty ellipsoid of pose estimation. Based on pose covariance, different metrics were introduced to guide the feature selection, such as information gain [12], [13], entropy [14], trace [15] and covariance ratio [16]. A potential issue is the pursuit of low uncertainty in estimation, rather than accuracy. These two objectives are not equivalent; an estimate can converge to a wrong pose with high confidence. In addition, the works above target efficiency of pose tracking; none of them explicitly target accuracy improvements via feature selection.

The works most related to this paper are [17], [18] and [19]. In [17], [18], the connection between pose tracking accuracy and observability conditioning of SLAM as a dynamic system was studied. The insight of their work being: the better conditioned the SLAM system is, the more tolerant the pose estimator will be to feature measurement error. To that end, the minimum singular value of observability matrix is used in to assess the observability condition of the SLAM system. Here, we employ a different metric, *Max-logDet*, and demonstrate its superiority to minimal singular value. Furthermore, we argue that bundle adjustment pipelines may benefit from an alternative set of matrices to consider for solution conditioning, ones more related to the underlying bundle adjustment problem.

In [19], feature selection is performed by maximizing the information gain of pose estimation within a prediction horizon. Two feature selection metrics were evaluated, minimal eigenvalue and log determinant (*Max-logDet*). Though the current investigation uses the log determinant metric, the algorithm for approximately selecting the *logDet* maximizing feature subsetdiffers, as does the matrix whose conditioning is optimized. We propose a lazier-greedy algorithm taking an order of magnitude less time than the greedy algorithm of [19], yet preserving the optimality bound. Further, we are

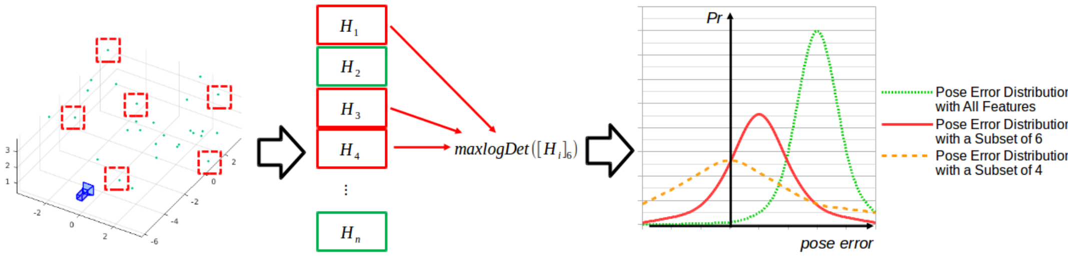

Fig. 1. A toy case to illustrate our approach. For least squares pose optimization, feature selection is equivalant to row block selection of Jacobian. With the *Max-logDet* metric, the subset of row blocks (and the corresponding feature subset) can be obtained. With a properly sized feature subset (e.g. the error distribution in red solid line), the bias of pose optmization is significantly reduced, while an acceptable amount of variance is introduced.

interested in improving the accuracy of pose tracking by selecting features with robustness properties, while preserving the time cost. As illustrated in Fig 1, this objective can be achieved by balancing between variance (e.g. minimizing the uncertainty of pose estimation) and bias (e.g. minimizing the expectation of pose error).

The proposed method hits all studied aspects to date: outlier rejection, appearance-based inlier selection, and structural-based inlier selection. We show that all three together outperform each individually. The contributions are:
1) Demonstrated **applicability** of improving accuracy via feature selection, mathematically and experimentally;
2) Exploration of **metrics** connected to the least squares conditioning of pose optimization, with quantification of *Max-logDet* as the optimal metric;
3) An **efficient algorithm** to approximately solve the NP-hard *Max-logDet* problem for real-time feature selection in the pose tracking step of VO/VSLAM; and
4) **Integration** of the algorithm **into a state-of-the-art visual SLAM** system and evaluation on public benchmark collected with a high-speed UAV. By selecting good features with the proposed method, tracking accuracy is significantly improved with minimal impact on the time cost.

II. Feature Selection in Least Squares Pose Optimization

The least squares objective of pose optimization in feature-based VO/VSLAM can be written as follows,

$$\arg\min \left\| h(x,p) - z \right\|^2 , \tag{1}$$

where x is the pose of the camera, p is the 3D feature points and z the is corresponding measurements on 2D image frame. The measurement function, $h(x,p)$, is a combination of world-to-camera transformation and pin-hole projection. We base the theory of feature selection upon this objective function. For direct VO/VSLAM, the objective function is slightly different. Nevertheless, the theory in the following can be easily extended to the direct version, once the direct residual term is properly approximated to first-order.

Solving the least squares objective of Eq (1) often involves the first-order approximation to the non-linear measurement function $h(x,p)$ linearization about initial guess $x^{(s)}$:

$$\left\| h(x,p) - z \right\|^2 = \left\| h(x^{(s)},p) + H_x(x - x^{(s)}) - z \right\|^2 . \tag{2}$$

To minimize of the first-order approximation Eq (2) via Gauss-Newton, the pose estimation is iteratively updated via

$$x^{(s+1)} = x^{(s)} - H_x^{+}(z - h(x^{(s)},p)). \tag{3}$$

Gauss-Newton accuracy is affected by two types of error: measurement error ϵ_z and map error ϵ_p. Again with the first-order approximation of $h(x,p)$ at the initial pose $x^{(s)}$ and assumed map point $p^{(s)}$, we can connect the pose optimization error to measurement and map errors:

$$\epsilon_x = H_x^{+}(\epsilon_z - H_p \epsilon_p). \tag{4}$$

Notice H_p is a block diagonal matrix of size $2n \times 3n$, where n is the number of matched features. We will discuss the influence of feature selection on pose optimization error ϵ_x.

a) Minimizing the Variance from Measurement / Map Error: Consider the case that only measurement error ϵ_z exists and is i.i.d. Gaussian with isotropic, diagonal covariance: $\epsilon_z(i) \sim N(0, \sigma_z^2)$. The pose covariance matrix will be

$$\Sigma_x = \sigma_z^2 (H_x^T H_x)^{-1} = \sigma_z^2 [\sum_{i=1}^{n} H_x(i)^T H_x(i)]^{-1}. \tag{5}$$

where $H_x(i)$ being the corresponding row block in H_x for feature i. The pose covariance matrix represents the uncertainty ellipsoid in pose configuration space. According to Eq (5), one should always use all the features/measurements available to minimize the uncertainty (i.e. variance) in pose estimation: with more measurements, the singular values of the measurement Jacobian H_x will increase in magnitude. The worst case variance would be proportional to $\sigma_{min}^{-2}(H_x)$, whereas in the best case it would be $\sigma_{max}^{-2}(H_x)$.

Similarly, consider minimizing the variance due to map error. With an i.i.d. Gaussian assumption on map error: $\epsilon_p(i) \sim N(0, \sigma_p^2)$, we can derive the pose covariance matrix,

$$\begin{aligned} \Sigma_x &= \sigma_p^2 H_x^{+} H_p H_p^T (H_x^{+})^T \\ &= \sigma_p^2 \{\sum_{i=1}^{n} H_x(i)^T [H_p(i) H_p(i)^T]^{-1} H_x(i)\}^{-1}. \end{aligned} \tag{6}$$

Still all map points being matched should be utilized. The worst case variance would be proportional to $\sigma_{min}^{-2}(H_p^{+} H_x)$, whereas in the best case it would be $\sigma_{max}^{-2}(H_p^{+} H_x)$.

b) Minimizing the Bias from Map Error: Yet another case to consider is the existence of biased map error (i.e. the mean of error distribution is non-zero). Biased map error may appear in real VO/VSLAM applications. For example, the map points could be batch-perturbed when triangulated with erroneous camera poses. Also, offset exists within a group of map points when they are jointly optimized with scale-drifted key frames. Here, we briefly discuss the case that map error ϵ_p follows non-zero-mean i.i.d. Gaussian, $\epsilon_p(i) \sim N(\mu_p, \sigma_p^2)$ and measurement error ϵ_z is unbiased.

The expectation of the pose optimization error will be biased by the non-zero-mean map error:

$$\mathbb{E}\left[\epsilon_x\right] = \mathbb{E}\left[H_x^+ H_p \epsilon_p\right] = H_x^+ H_p \mathbf{1}_n \mu_p \tag{7}$$

where $\mathbf{1}_n$ is a tall matrix of n smaller identity matrices. In the worst case scenario, pose error expectation $\mathbb{E}\left[\epsilon_x\right]$ is amplified by $\sigma_{max}(H_x^+ H_p)$, whereas in the best case it is only amplified by $\sigma_{min}(H_x^+ H_p)$. Subset selection affects the two components, H_x and H_p, in opposite ways: it will increase the amplification factor of H_x^+, while bounding the amount of noise induced by H_p. When the reduction of the latter is larger in magnitude than the increase of the prior, the pose optimization error should drop. Obviously, one possible objective of feature subset selection would be minimizing the factor of worst case scenario, $\sigma_{max}(H_x^+ H_p)$; another option would be minimizing both $\sigma_{max}(H_x^+ H_p)$ and $\sigma_{min}(H_x^+ H_p)$.

Furthermore, the two matrices, H_x^+ and H_p, can be combined into one. Move both to the left hand side of Eq (7),

$$H_p^+ H_x \mathbb{E}\left[\epsilon_x\right] = \mathbf{1}_n \mu_p. \tag{8}$$

Note that the projection Jacobian H_p is a $2n \times 3n$ block diagonal matrix, consisting of 2×3 denoted by $H_p(i)$. Meanwhile, each row block of H_x can be written as $H_x(i)$.

To remove the need for the pseudo inverse of H_p, add one more row $[0 \ 0 \ 1]$, to each block $H_p(i)$. In addition a zero row is added to each row block $H_x(i)$ to get new row block $H_x^n(i)$. This trick does not affect the structure of the least square problem, but it does allow inversion of the new diagonal block $H_p^n(i) = [H_p(i); 0 \ 0 \ 1]$. After performing block-wise multiplication, one can obtain the combined matrix H_c, consisting of concatenated row blocks $H_p^n(i)^{-1} H_x^n(i)$. Instead of working with two independent matrices H_x and H_p, we consider optimizing the spectral properties of their combination, H_c.

This section covered three perspectives of pose optimization under measurement & map error, and identified the scenario whereby feature selection might reduce estimation error. Under biased map errors (which is true in real VO/VSLAM applications), selecting a subset of features could improve least squares pose optimization accuracy.

III. Good Feature Selection Metrics

Analyzing the impact of map error on least squares pose optimization led to equations where the singular values of H_c and their extremal properties were connected to best/worst case outcomes. Actual outcomes would depend on the overall spectral properties of H_c. Therefore, we seek a sub-matrix of H_c preserving as best as possible the overall spectral properties, and at minimum the extremal spectral properties.

Under this spectral preservation objective, the feature selection problem is equivalent to selecting a subset of row blocks in the matrix H_c such that the norm of the selected sub-matrix is as large as possible. Once selected, the sub-matrix determines which measurements from the set of available measurements should be taken (these would be a subset of the good feature points). Submatrix selection with spectral preservation has been extensively studied in the fields of computational theory and machine learning [20], [21], for which several matrix-revealing metrics exist to score the subset selection process. They are listed in Table I.

Subset selection with any of the matrix-revealing metrics listed above is equivalent to a finite combinational optimization problem:

$$\max_{S \subseteq \{1,2,\ldots,n\}, |S|=k} f([H_c(S)]^T [H_c(S)]) \tag{9}$$

where S is the indices of selected row blocks from full matrix H_c, $[H_c(S)]$ is the corresponding concatenated submatrix, and f is the matrix-revealing metric.

TABLE I

Commonly used matrix-revealing metrics

Max-Trace	Trace $Tr(Q) = \sum_1^k Q_{ii}$ is max.
Min-Cond	Condition $\kappa(Q) = \lambda_1(Q)/\lambda_k(Q)$ is min.
Max-MinEigenValue	Min. eigenvalue $\lambda_k(Q)$ is max.
Max-logDet	Log. of determinant $\log\det(Q)$ is max.

A. Submodularity

The combinational optimization above can be solved with brute-force, but the exponentially-growing problem space quickly becomes impractical to search, especially for real-time VO/VSLAM applications. Heuristics for subset selection target one structural property, *submodularity* [19], [22], [23]. If a set function (e.g. matrix-revealing metric) is submodular and monotone increasing, then approximate, greedy combinational optimization of the set function (e.g. subset selection) has near optimality guarantees.

Except for *Min-Cond*, all other three metrics list in Table I are proven to be either submodular, or approximatly submodular, and monotone increasing. [23] provides proof for submodularity of *Max-logDet*. The stronger property, modularity, holds for *Max-Trace* [22]. Though *Max-MinEigenValue* does not meet submodularity in general, it is recognized as approximately submodular [19]. Therefore, selecting row blocks (as well as the corresponding features) with these metrics can be approximated by greedy approach.

B. Simulation of Good Feature Selection

To identify the applicable cases of the good feature selection, and explore the matrix-revealing metrics that could guide good feature/row block subset selection, a simulation of least squares pose optimization was carried out. The simulation environment of [24], which assumes perfect data association, provides the testing framework. The evaluation scenario is depicted in Fig 2. The camera/robot is spawned at the origin of the world frame, and a fixed number (e.g. 200 in this synthetic test) of 3D feature points are randomly generated in front of the camera. After applying a small random pose transform to the robot/camera, the 2D projections of feature points are measured and perfectly matched with known 3D feature points. A Gauss-Newton optimizer estimates the random pose transform from the matches.

To simulate map error, the 3D feature points are perturbed with biased noise (Gaussian with mean of 0.05m, and standard deviation of 0.05m). The 2D measurements are also perturbed with two levels of measurement error: zero-mean Gaussian with standard deviation of 1 and 2 pixel. Subset size ranging from 80 to 200 are tested. To be statistically sound, 300 runs are repeated for each configuration.

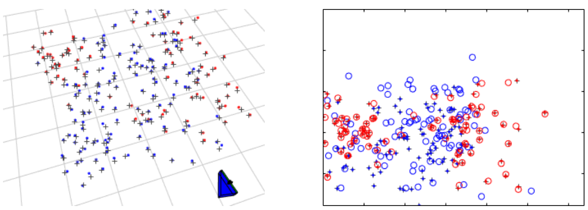

Fig. 2. Simulated pose optimization scenario. Left: map view; Right: camera view. Selected features are red and unselected ones are blue.

Feature selection occurs prior to Gauss-Newton pose optimization, so that only a subset of selected features is sent to the optimizer. Each of the matrix-revealing metrics listed in Table I were tested.

Feature selection is done in three steps: 1) compute the full measurement Jacobian H_x and projection Jacobian H_p, 2) combine the two into H_c, and 3) greedily select row blocks of $H_c(i)$, based on the matrix-revealing metric, until reaching the target subset size. The simulation results are presented in Fig 3. For reference, we also plot the simulation results with randomized subset selection (*Random*) and with all features available (*ALL*).

From Fig 3, the *Max-logDet* metric has the best overall performance. Under both low and high level of residual noise, it more quickly approaches the baseline error (*All*). Though marginal, the translational error of *Max-logDet* goes below the ALL baseline, while the rotational error equals the baseline, once the subset size exceeds 160. The results point to the value of *Max-logDet* good features selection.

IV. Efficient Max-logDet Subset Selection

Subset selection with *Max-logDet* metric has been studied in fields such as sensor selection [23] and feature selection [19]. There, a simple greedy algorithm is commonly used to approximate the original NP-hard combinational optimization problem. Since *Max-logDet* is submodular and monotone increasing, the approximation ratio of the greedy approach is $1 - 1/e$ [22], which is the best any polynomial time algorithm can achieve under the assumption $P \neq NP$.

However, the computational cost of the greedy algorithm is too high for feature selection in real-time VO/VSLAM applications. As reported in [19] and confirmed by us, the time cost of greedy selection exceeds the real-time requirement (e.g. 30ms per frame) with around 100 feature inputs. To select k feature out of n candidates, the greedy algorithm has to run k rounds. In each round it considers all remaining candidates to identify the current best feature. Hence the total complexity of greedy algorithm is $\mathcal{O}(nk)$.

To speed up the greedy feature selection, we explore the combination of deterministic selection (e.g. the greedy algorithm) and randomized acceleration (e.g. random sampling). One well-recognized method of combining these two, is stochastic greedy [25]. Each round of greedy selection evaluated a random subset of candidates to identify the current "best" feature, instead of going through all n candidates. The random subset size s is controlled by a decay factor ϵ: $s = \frac{n}{k}\log(\frac{1}{\epsilon})$. Complexity reduces to $\mathcal{O}(\log(\frac{1}{\epsilon})n)$.

More importantly, the expected approximation guarantee of stochastic greedy is proven to be $1 - 1/e - \epsilon$ [25]; compare to $1 - 1/e$, the best approximation ratio of any polynomial time algorithm [22]. Selecting a proper decay factor ϵ in stochastic greedy (e.g. $\epsilon = 0.1$ in the following experiments), slightly lowers the optimum bound, while significantly speeding up selection (16% vs 43x). Alg 1 summarizes the stochastic-greedy-based *Max-logDet* feature selection algorithm.

Algorithm 1: Proposed efficient approximation algorithm for *Max-logDet* feature selection.

Data: $H_c = \{H_c(1),\ H_c(2),\ \dots,\ H_c(n)\},\ k$
Result: $H_c^{sub} \subseteq H_c,\ |H_c^{sub}| = k$

```
H_c^sub ← ∅;
while |H_c^sub| < k do
    H_c^R ← a random subset obtained by sampling
        s = (n/k) log(1/ε) random elements from H_c;
    H_c(i) ← argmax_{H_c(i) ∈ H_c^R} log det(H_c(i)^T H_c(i)
        + [H_c^sub]^T [H_c^sub]);
    H_c^sub ← H_c^sub ∪ H_c(i);
    H_c ← H_c \ H_c(i);
return H_c^sub.
```

V. Experimental Results on Real-time VSLAM

This section evaluates the performance of the proposed *Max-logDet* feature selection on a state-of-the-art feature-based monocular visual SLAM system, ORB-SLAM [1]. By integrating the proposed feature selection to the real-time tracking thread of ORB-SLAM, we demonstrate significant improvement in pose tracking accuracy, while the time cost of pose tracking only increases slightly.

Feature selection is done in the pose refinement function, $TrackLocalMap$, of the real-time tracking thread of ORB-SLAM. All possible feature matches found between the current frame and the local map are fed into this function. However, feature selection is not conducted on the whole set of input matchings directly: the input set contains some outliers (i.e. *non-inliers*), which affect the performance of pose optimization when included. Outlier rejection needs to be applied to the tracked features prior to feature selection. Due to the lack of explicit outlier rejection in ORB-SLAM, we add an outlier rejection module by employing the ORB-SLAM pose optimization code. Pose optimization is conducted with the whole set of feature matchings, then tracked features with high re-projection error are rejected. Such an

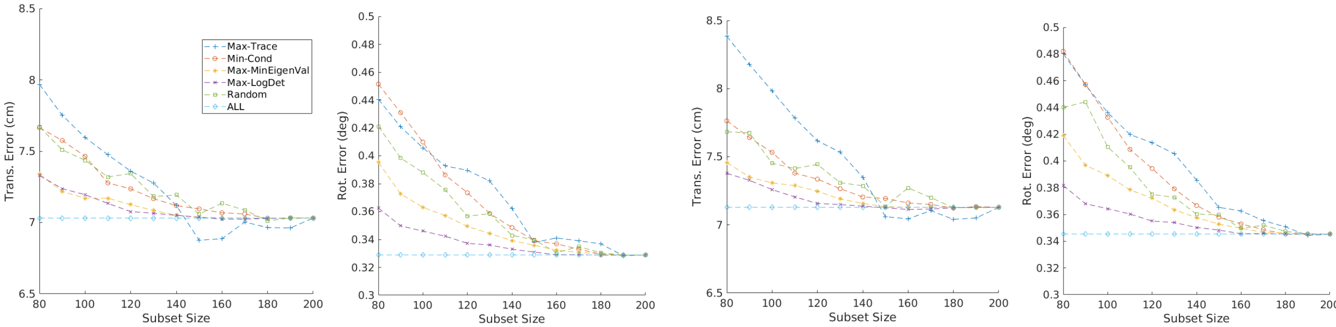

Fig. 3. Simulation results of least squares pose optimization. First 2 columns: RMS of translational / rotational error under low noise. Last 2 columns: RMS of translational / rotational error under high noise.

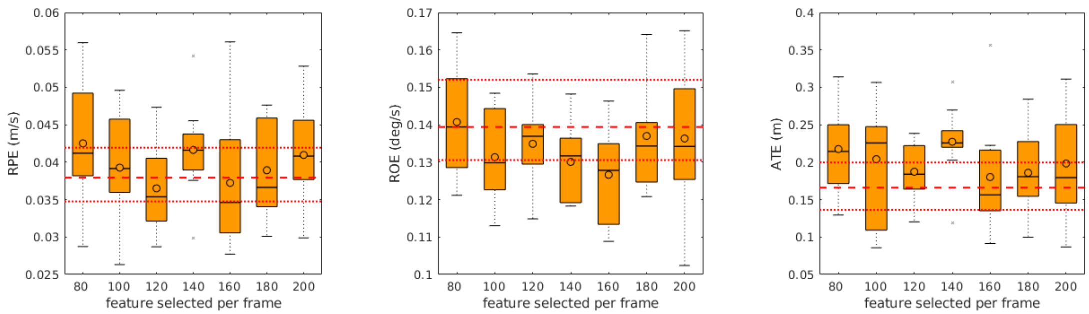

Fig. 4. Tracking accuracy vs. number of good features selected per frame. For *Max-logDet* ORB-SLAM, the error metric under each feature selection budget are presented as boxplot in orange; for INL-ORB the mean (red dashed) and the 25-75 percentile (red dotted) lines are plotted.

implementation of outlier rejection is far from efficient, but it will kick out most of the outliers.

Five feature selection approaches are implemented: 1) *Quality*, which selects based on the ORB-matching score; 2) *Bucket* [26], which divides the frame into grids and uniformly samples from them; 3) *Observability* (*Obs*) [18], which selects based on observability over a short time (here, last 3 segments); 4) *Max-logDet* (*MD*), which selects based on Alg 1; and 5) *Quality + MD*, which generates a subset of features based on the ORB-matching score first, then selects from them using the *Max-logDet* algorithm. Two baseline approaches are included: 1) *INL-ORB*, which has the explicit outlier rejection module on top of original ORB-SLAM; and 2) *ALL-ORB*, the original ORB-SLAM.

Since the focus is on real-time pose tracking, all evaluations are performed on the instantaneous output of pose tracking thread; key-frame poses after posterior bundle adjustment are not used. Relocalization and loop closing are disabled in all implementations. For ORB-SLAM with feature selection, the number of tracked features used is fixed (100 features per frame). For the *Quality+MD* combination, a candidate pool of 200 features is selected using Quality, from which the good feature subset is further extracted based on the proposed *Max-logDet* algorithm. Meanwhile, for the baseline approaches *INL-SLAM* and *ALL-SLAM*, as many as 2000 features can be used to optimize the pose per frame.

The benchmark used is the EuRoC MAV dataset [27]. It consists of stereo images and inertial data recorded from a micro aerial vehicle. Only the images from the left camera are used in this monocular visual SLAM experiment. In total 11 sequences are recorded under 3 different indoor environments, with a total length of 19 minutes. Challenging cases such as low-texture, illumination changes, fast motion and motion blur are covered. Each sequence has ground-truth from a motion capture system (Vicon or Leica MS50).

Due to the initialization procedure and multi-threaded structure of ORB-SLAM, all approaches are run 10 times per sequence. The platform was an Intel i7 quadcore 4.20GHz CPU (passmark score of 2583 per thread) with ROS Indigo. Accuracy of real-time pose tracking is evaluated with three metrics [28] between ground truth and SLAM estimates (aligned to ground truth with a *Sim3* transform):
1) **Absolute Trajectory Error (ATE)**, the root-mean-square difference between the ground truth and the entire estimated trajectory;
2) **Relative Position Error (RPE)**, the average drift of pose tracking over a short period of time;
3) **Relative Orientation Error (ROE)**, the average orientation drift similar to RPE.
RPE and ROE are averaging windows are 3 seconds.

A. Accuracy vs. Subset Size

The connection between the number of good features selected and the pose optimization accuracy is assessed on one EuRoC sequence, *MH 05 diff*. Fast camera motion and changing lighting conditions challenge accurate tracking and mapping. When running on this sequence, the the measurements and mapped features are expected to be noisy; good feature selection should mitigate the effects of the noise.

Fig 4 consists of box plots for 10-runs of *Max-logDet* ORB-SLAM on the example sequence under feature selection budgets ranging from 80 to 200. One plot for each evaluation metric. For reference, we plot (in red) the outcomes for

TABLE II

RELATIVE POSITION (M/S) / ORIENTATION (DEG/S) ERROR ON EUROC SEQUENCES

Budget	Approach 100 Feature per Frame					2000 Feature per Frame	
Seq.	*Quality*	*Bucket*	*Obs*	*MD*	*Quality+MD*	*INL-ORB*	*ALL-ORB*
MH 01 easy	0.012 0.09	0.012 0.08	0.013 0.11	**0.012 0.08**	0.013 0.09	0.011 0.07	**0.011 0.07**
MH 02 easy	0.032 0.36	0.024 0.28	0.030 0.33	**0.022 0.25**	0.029 0.33	0.109 0.62	**0.046 0.41**
MH 03 med	0.027 0.12	**0.026 0.12**	0.029 0.14	0.053 0.27	0.028 0.12	0.028 0.11	**0.027 0.10**
MH 04 diff	0.096 0.28	0.077 0.25	0.084 0.26	0.077 0.31	**0.064 0.19**	**0.071 0.20**	0.094 0.26
MH 05 diff	0.063 0.20	0.063 0.21	0.046 0.18	0.041 0.14	**0.041 0.14**	**0.038 0.14**	0.060 0.14
VR1 01 easy	0.038 0.46	0.038 0.45	0.038 0.47	**0.038 0.45**	0.038 0.45	**0.038 0.45**	0.038 0.45
VR2 01 easy	0.016 0.27	0.014 0.25	0.015 0.26	0.015 0.26	**0.011 0.20**	**0.011** 0.24	0.012 **0.22**
VR2 02 med	0.093 0.61	0.153 0.89	0.126 1.02	0.078 0.57	**0.032 0.52**	**0.165** 0.87	0.200 **0.86**
Average	0.047 0.30	0.051 0.32	0.048 0.35	0.042 0.29	**0.032 0.26**	**0.059** 0.35	0.061 **0.32**
# Seq. with Perf. Loss	6	6	6	5	**3**	5	-
# Seq. with Perf. Gain	2	2	2	3	**5**	3	-
Average Perf. Loss	0.002 0.03	0.001 0.03	0.002 0.05	0.010 0.06	0.001 0.01	0.032 0.05	-
Average Perf. Gain	-0.060 -0.15	-0.021 -0.07	-0.028 -0.04	-0.036 -0.15	-0.047 -0.10	-0.013 -0.02	-

inlier-only ORB-SLAM (tracking up to 2000 features/frame). The improvement of feature selection is mostly significant on the boxplot of ROE (between the budget of 100 and 180). The improvement on RPE is less obvious: feature selection leads to a slight reduction of RPE for budgets of 120 and 160. The absolute metric (ATE) is less sensitive to subset selection. In the subsequent evaluations on good feature selection, the smallest budget that leads to accuracy improvement will be used, 100 feature/frame.

B. Accuracy vs. Feature Selection Approaches

Table II summarizes the relative metrics (RPE and ROE). Each cell first reports the average RPE (units: m/s), then the average ROE (units: deg/s). For each selection approach type (100 feat. and 2000 feat.), the lowest relative errors per sequence are in bold. Three sequences are not included due to frequent failures (since relocalization is disabled).

On almost all sequences, either the *MD* or the *Quality+MD* combination has the lowest relative error of the feature selection approaches. On challenging sequences such as *MH 04 diff*, *MH 05 diff* and *VR2 02 med*, the combined approach reduces the relative error significantly. The exception is *MH 03 med* where the combined approach results in a slightly higher RPE than the lowest one (generated by *Bucket*). Overall, the *MD* approach reduces pose tracking error on several sequences by exploiting the structural and motion information. Integrating *MD* with appearance information (i.e. *Quality*) further improves performance.

Now, compare *Quality+MD* with the two baselines. On sequences such as *MH 02 easy*, *MH 04 diff* and *VR2 02 med*, *Quality+MD* clearly leads to lower relative error. Meanwhile on other sequences, the relative error of *Quality+MD* is either the same as baselines or slightly worse. The performance gains on the harder sequences far outweigh the performance loss on the easy sequences, as presented in the last 4 rows of Table II. When under-performing, the *Quality+MD* approach has the lowest performance loss. When over-performing, it does so more often and by a significant amount. The average RPE and ROE scores for *Quality+MD* improve by 47% and 19%, respectively, versus *ALL-ORB*.

C. Good Feature ORB-SLAM vs. Other VO/VSLAM

The accuracy improvement of good feature selection using Quality+MD is further demonstrated by comparing against other state-of-the-art VO/VSLAM methods. Two direct approaches, SVO [2] and DSO [3], are chosen as baselines. For fair comparison, both SVO and DSO are evaluated under the same configuration as above: 1) monocular vision input only with real-time enforcement, 2) up to 2000 (patch) matchings per frame, 3) real-time pose tracking results of the entire sequence being evaluated (both [2] and [3] remove the beginning part with strong motion in evaluation), and 4) only those succeeding for all 10 runs are reported (no tracking failure allowed). Performance is measured with absolute translation error (ATE), as per [2]. Table III reports the ATEs.

With *Quality+MD* feature selection, the ATE on sequences *MH 02 easy*, *MH 04 diff*, and *VR2 02 med* are significantly reduced, while the accuracy advantage are preserved on the rest. The error metrics statistics given in the last three rows indicate that *Quality+MD* ORB-SLAM has the lowest average ATE, as well as the lowest maximum ATE compared to the approaches evaluated. The two direct baselines do not perform as well: SVO has the worst ATE on all 10 trackable sequences; DSO only tracks on 5 sequences completely, and has the 2nd worst average ATE.

D. Efficiency vs. Feature Selection Approaches

Table IV present a breakdown of the computation time for each feature selection approach (averaged over all EuRoC sequences). The *Base* column measures the pre-processing steps (ORB extraction, initial tracking, and outlier rejection) before feature selection. Due to the outlier rejection step, all methods except ALL-SLAM, incur increased timing. Of the structural-based selection approaches, *Quality+MD* is the 2nd fastest. *Bucket* is extremely efficient, but does not improve as much the accuracy. When comparing *Quality+MD* to baseline ORB-SLAM, outlier rejection time cost is almost offset by the time savings in pose optimization. We imagine better implemented outlier rejection or integration of outlier rejection and feature (inlier) selection, could consume less time than *ALL-ORB*, while still enhancing performance.

TABLE III
ABSOLUTE TRANSLATION ERROR (M) ON EUROC SEQUENCES

Seq.	Approach ORB-SLAM ALL	INL	Quality+MD	SVO	DSO
MH 01 easy	**0.03**	**0.03**	0.04	0.30	-
MH 02 easy	0.33	0.70	**0.15**	-	-
MH 03 med	**0.04**	0.05	0.05	0.39	0.75
MH 04 diff	0.96	0.48	**0.41**	5.82	-
MH 05 diff	0.27	**0.17**	0.22	4.15	-
VR1 01 easy	0.04	0.04	0.04	0.77	0.64
VR1 02 med	-	-	-	0.77	0.53
VR1 03 diff	-	-	-	0.65	-
VR2 01 easy	0.04	0.04	0.04	0.18	0.29
VR2 02 med	0.75	0.61	**0.12**	1.57	1.04
VR2 03 diff	-	-	-	1.66	-
Average	0.31	0.27	**0.13**	1.63	0.65
Min.	**0.03**	**0.03**	0.04	0.18	0.29
Max.	0.96	0.70	**0.41**	5.82	1.04

TABLE IV
TIME COST (MS) PER FRAME BREAKING DOWN FOR POSE TRACKING

	Base	**Feat. Sel.**	**Pose Opt.**	**Total**
Quality	28.6	0.00	0.35	29.0 (-0.7%)
Bucket	28.5	0.11	0.34	29.0 (-0.7%)
Obs	28.5	2.6	0.32	31.4 (+7.5%)
MD	28.6	3.0	0.40	32.0 (+9.6%)
Quality+MD	28.5	1.5	0.34	30.3 (+3.8%)
INL-ORB	28.8	-	2.4	31.3 (+7.2%)
ALL-ORB	26.5	-	2.6	29.2

VI. CONCLUSION

This paper presented the idea of good feature selection for least squares pose optimization. Under a biased noise assumption, selecting a subset of features should improve optimization accuracy. The connection between matrix subset selection methods and the solution conditioning of least squares optimization was discussed. Through a controlled experiment, the *Max-logDet* matrix revealing metric was shown to perform best. For rapid subset selection, a near optimal heuristic approach to *Max-logDet* is used. Integrating the proposed good feature selection approach with a feature point quality scoring selector and outlier rejection leads to a more accurate visual odometry within a SLAM system with nearly the same computational cost.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “ORB-SLAM: a versatile and accurate monocular SLAM system,” *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015. [Online]. Available: https://github.com/raulmur/ORB_SLAM

[2] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, “SVO: Semidirect visual odometry for monocular and multicamera systems,” *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2017. [Online]. Available: http://rpg.ifi.uzh.ch/svo2.html

[3] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. [Online]. Available: https://github.com/JakobEngel/dso

[4] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, “MonoSLAM: Real-time single camera SLAM,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[5] A. Vedaldi, H. Jin, P. Favaro, and S. Soatto, “KALMANSAC: Robust filtering by consensus,” in *IEEE International Conference on Computer Vision*, 2005, pp. 633–640.

[6] J. Civera, O. G. Grasa, A. J. Davison, and J. Montiel, “1-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry,” *Journal of Field Robotics*, vol. 27, no. 5, pp. 609–631, 2010.

[7] G. Klein and D. Murray, “Parallel tracking and mapping for small AR workspaces,” in *IEEE / ACM International Symposium on Mixed and Augmented Reality*, 2007, pp. 225–234.

[8] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[9] J. Shi and C. Tomasi, “Good features to track,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.

[10] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson, “Landmark selection for vision-based navigation,” *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 334–349, 2006.

[11] Z. Shi, Z. Liu, X. Wu, and W. Xu, “Feature selection for reliable data association in visual SLAM,” *Machine Vision and Applications*, pp. 1–16, 2013.

[12] A. Davison, “Active search for real-time vision,” in *IEEE International Conference on Computer Vision*, vol. 1, 2005, pp. 66–73.

[13] M. Kaess and F. Dellaert, “Covariance recovery from a square root information matrix for data association,” *Robotics and Autonomous Systems*, vol. 57, no. 12, pp. 1198–1210, 2009.

[14] S. Zhang, L. Xie, and M. D. Adams, “Entropy based feature selection scheme for real time simultaneous localization and map building,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 1175–1180.

[15] R. Lerner, E. Rivlin, and I. Shimshoni, “Landmark selection for task-oriented navigation,” *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 494–505, 2007.

[16] F. A. Cheein, G. Scaglia, F. di Sciasio, and R. Carelli, “Feature selection criteria for real time EKF-SLAM algorithm,” *International Journal of Advanced Robotic Systems*, vol. 6, no. 3, p. 21, 2009.

[17] G. Zhang and P. A. Vela, “Optimally observable and minimal cardinality monocular SLAM,” in *IEEE International Conference on Robotics and Automation*, 2015, pp. 5211–5218.

[18] ——, “Good features to track for visual SLAM,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1373–1382.

[19] L. Carlone and S. Karaman, “Attention and anticipation in fast visual-inertial navigation,” in *IEEE International Conference on Robotics and Automation*, 2017, pp. 3886–3893.

[20] M. Gu and S. C. Eisenstat, “Efficient algorithms for computing a strong rank-revealing QR factorization,” *SIAM Journal on Scientific Computing*, vol. 17, no. 4, pp. 848–869, 1996.

[21] C. Boutsidis, M. W. Mahoney, and P. Drineas, “An improved approximation algorithm for the column subset selection problem,” in *ACM-SIAM Symposium on Discrete Algorithms*, 2009, pp. 968–977.

[22] T. H. Summers, F. L. Cortesi, and J. Lygeros, “On submodularity and controllability in complex dynamical networks,” *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 91–101, 2016.

[23] M. Shamaiah, S. Banerjee, and H. Vikalo, “Greedy sensor selection: Leveraging submodularity,” in *IEEE Conference on Decision and Control*, 2010, pp. 2572–2577.

[24] J. Sola, T. Vidal-Calleja, J. Civera, and J. M. M. Montiel, “Impact of landmark parametrization on monocular EKF-SLAM with points and lines,” *International Journal of Computer Vision*, vol. 97, no. 3, pp. 339–368, 2012.

[25] B. Mirzasoleiman, A. Badanidiyuru, A. Karbasi, J. Vondrak, and A. Krause, “Lazier than lazy greedy,” in *AAAI Conference on Artificial Intelligence*, 2015.

[26] A. Geiger, J. Ziegler, and C. Stiller, “Stereoscan: Dense 3d reconstruction in real-time,” in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 963–968.

[27] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, “The EuRoC micro aerial vehicle datasets,” *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[28] J. Sturm, W. Burgard, and D. Cremers, “Evaluating egomotion and structure-from-motion approaches using the TUM RGB-D benchmark,” in *Workshop at the IEEE/RJS International Conference on Intelligent Robot Systems*, 2012.