Uncertainty Based Camera Model Selection

Michal Polic
CTU in Prague*

Stanislav Steidl
CTU in Prague*

Cenek Albl
ETH Zurich

Zuzana Kukelova
CTU in Prague†

Tomas Pajdla
CTU in Prague*

Abstract

The quality and speed of Structure from Motion (SfM) methods depend significantly on the camera model chosen for the reconstruction. In most of the SfM pipelines, the camera model is manually chosen by the user. In this paper, we present a new automatic method for camera model selection in large scale SfM that is based on efficient uncertainty evaluation. We first perform an extensive comparison of classical model selection based on known Information Criteria and show that they do not provide sufficiently accurate results when applied to camera model selection. Then we propose a new Accuracy-based Criterion, which evaluates an efficient approximation of the uncertainty of the estimated parameters in tested models. Using the new criterion, we design a camera model selection method and fine-tune it by machine learning. Our simulated and real experiments demonstrate a significant increase in reconstruction quality as well as a considerable speedup of the SfM process.

1. Introduction

Structure from Motion (SfM) has many applications in 3D reconstruction [42, 43, 39], image matching [36], visual odometry [30, 7] and visual localization [46, 38, 44]. Large-scale 3D reconstruction pipelines, e.g. COLMAP [39], Meshroom [6], and RealityCapture [1] are widely used.

SfM pipelines use many parameters that are hard to set in practice. A crucial parameter to set is the camera model to be used. In fact, every absolute [25] and relative pose solver [24] is derived for one particular camera model and the user has to choose it. Using a too simple camera model may lead to under-fitting and inaccurate reconstruction. Using a too complex model may lead to over-fitting the data and result in degeneracies [5], as shown in Fig. 1.

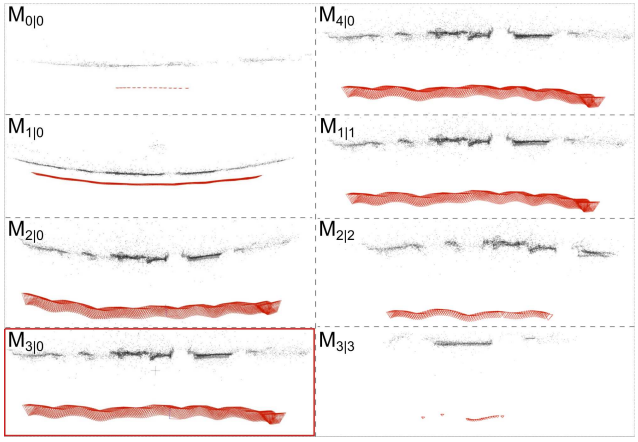

Figure 1: Cameras (red) and 3D points (black) reconstructed by COLMAP [39] with eight different radial distortion models. The best camera model selected using our method (framed in red) gives the most planar result for 3D points of a flat wall (using a dataset from [40]).

Model selection based on statistics is a well-studied problem. However, camera model selection in SfM by standard Information Criteria (IC) may not work for several reasons. First, the reconstruction has a singular statistical model due to the gauge freedom [21], i.e. the likelihood function of having a "good" model cannot be derived using the normal distribution [48]. Secondly, the prior distribution of the reconstruction parameters (e.g. camera poses and 3D points) is not known and thus Bayesian methods cannot be used either. Third, for different camera models and different reprojection thresholds, the final 3D reconstruction contains different numbers of registered 3D points and cameras, i.e. the size of data is not constant. Finally, standard ICs assume that residuals depend only on the selected model. However, camera model selection also depends on physical properties (e.g. lighting and view angle) [20].

The ultimate goal of a camera model selection method is to select a "good" model where (i) all images are registered, (ii) the reprojection error is minimal, and (iii) the number

*CIIRC - Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, † Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague. This work was supported by the European Regional Development Fund under IMPACT (reg. no. CZ.02.1.01/0.0/0.0/15 003/0000468), EU H2020 No. 856994 ARtwin, EU H2020 No. 871245 SPRING Projects, OP RDE project International Mobility of Researchers MSCA-IF at CTU reg. no. CZ.02.2.69/0.0/0.0/17_050/0008025 and OP VVV project Research Center for Informatics reg. no. CZ.02.1.01/0.0/0.0/16_019/0000765, and by grants SGS19/173/OHK3/3T/13 and SGS19/172/OHK3/3T/13 of the GA of the CTU in Prague.

of parameters is small. This goal is very hard to reach in practice (*c.f*. Table 4).

We propose to use an efficient evaluation of an approximation of the uncertainty of the estimated parameters in tested models to select the camera model, which leads to the "most accurate" estimated parameters. We demonstrate our method on the important task of selecting a suitable radial distortion model: Most modern cameras take images exhibiting radial distortion. Yet, using a wrong radial distortion model often leads to degeneracies (*c.f*. Fig. 1).

Contributions. We first present a comparison of standard, robust, and geometrical ICs on the task of radial distortion model selection. Motivated by the poor performance of these state-of-the-art (SOTA) ICs, we design a new Accuracy-based Criterion (AC) and an AC-based camera model Selection method (ACS). We further fine-tune this method by learning a camera model selection classifier (LACS) from ACS evaluations for different reprojection error thresholds. In extensive synthetic and real experiments, we show a significant increase in the reconstruction quality as well as a considerable speedup of the reconstruction process. Moreover, we show that the use of the accuracy of observations improves the inlier/outlier classification.

2. Previous work

The model selection problem has received considerable attention [4, 19, 9, 41, 16, 34, 35, 20, 37, 26, 8]. The Akaike criterion (AIC [4]) is based on the first-order estimate of the Kullback-Leibler (KL) distance between the densities given by the data and true (unknown) density function. AIC computes the likelihood of the fitted model parameters and its bias correction. Hurvich's AICc [19] is a second-order estimate of the KL distance, which can be seen as extension of AIC for small sample sizes. Takeuchi's TIC [11] is another extension of AIC, which shrinks the model parameters towards the maximum entropy distribution and therefore is more robust if the correct model is not in the set of candidates models. Bozdogan's CAIC [9] adjusts AIC by the assumption that the order of the models does not change if the sample size increases. Schwarz's BIC [41] is motivated by approximating the marginal probability density of the data under the model, which leads to a higher magnitude of bias correction w.r.t. AIC. Rissanen's MDL [34] is derived from the minimal code length necessary for describing the data. A valuable extension of the AIC, MDL up to geometric G-AIC, G-MDL was introduced by Kanatani in [20]. It highlights that the accuracy depends primarily on physical properties of observed 3D structure. All the approaches above do assume observations without outliers. The simplest robust IC is Ronchetti's RAIC [37]. It generalizes the ML-estimator to an M-estimator, which minimizes a robust loss function of the residuals. This idea can be applied to ICs mentioned above, as in, e.g., RBIC [26] and RTIC [8]. Watanabe's WAIC and WBIC [48] assume known priors on the model parameters. However, such priors are not always available in practice.

The most related work to our is the work of Kanatani [20] where the G-AIC and G-MDL information criteria were proposed. These criteria were applied in [23] to choose between affine and projective camera models. However, G-AIC, G-MDL methods in general do not work well for the camera model selection task because this task has a singular statistical model [48]. Another approach to radial distortion model selection was presented in [15, 31]. That approach assumes correspondences between planar calibration boards with a fixed number of detected observations without considering any outliers and simplifies used camera models to homographies [17] between pairs of images. These are very strong assumptions and, as far as we know, there are no methods for radial distortion model selection without the use of calibration pattern.

3. Problem formulation

Our goal is to design a scoring function for camera models. Given a finite set of camera models $\mathcal{M} = \{\boldsymbol{M_1}, \dots, \boldsymbol{M_n}\}$, the camera model $\boldsymbol{M_b}$ that leads to the "most accurate" 3D reconstruction $\widetilde{\boldsymbol{\theta}}^{(b)}$, $b \in \{1, \dots, n\}$, should get the highest score. Let us assume that the reconstruction $\widetilde{\boldsymbol{\theta}}^{(i)}$ for the camera model $\boldsymbol{M_i}$ consists of $\widetilde{V}^{(i)}$ 3D points $\widetilde{\boldsymbol{X}}^{(i)} = \{\widetilde{\boldsymbol{X}}_1^{(i)}, \widetilde{\boldsymbol{X}}_2^{(i)}, \dots, \widetilde{\boldsymbol{X}}_{\widetilde{V}^{(i)}}^{(i)}\}$, $\widetilde{U}^{(i)}$ cameras $\widetilde{\boldsymbol{P}}^{(i)} = \{\widetilde{\boldsymbol{P}}_1^{(i)}, \widetilde{\boldsymbol{P}}_2^{(i)}, \dots, \widetilde{\boldsymbol{P}}_{\widetilde{U}^{(i)}}^{(i)}\}$ and radial distortion parameters $\widetilde{\boldsymbol{\theta}}_{rd}^{(i)}$, i.e. $\widetilde{\boldsymbol{\theta}}^{(i)} = \{\widetilde{\boldsymbol{P}}^{(i)}, \widetilde{\boldsymbol{X}}^{(i)}, \widetilde{\boldsymbol{\theta}}_{rd}^{(i)}\}$. Further, let $\widetilde{\boldsymbol{u}}^{(i)}$ be the observations of points $\widetilde{X}^{(i)}$ in images described by cameras $\widetilde{\boldsymbol{P}}^{(i)}$. In general, it is not possible to evaluate the accuracy of the reconstruction $\widetilde{\boldsymbol{X}}^{(i)}$ before all cameras are registered and all corresponding 3D points are triangulated. Running SfM for all considered camera models from $\mathcal{M}$ may be computationally extremely expensive. Therefore, we will evaluate the proposed camera model selection criterion for smaller sub-reconstructions with a fixed number of registered cameras $K \leq \min_i \widetilde{U}^{(i)}$. Note that we use $\widetilde{\boldsymbol{\theta}}^{(i)}$ for the complete reconstruction from all images and $\boldsymbol{\theta}^{(i)}$ for the reconstruction with K fixed registered cameras. It is obvious that considering more cameras in sub-reconstructions $\boldsymbol{\theta}^{(i)}$, i.e. a larger K, will lead to better approximations of the overall accuracy of the complete reconstructions $\widetilde{\boldsymbol{\theta}}^{(i)}$. Let us assume that the estimated parameters of these sub-reconstructions $\hat{\boldsymbol{\theta}}^{(i)}$ ($\hat{}$ denotes estimated values) were esti-

mated by an SfM pipeline, e.g. COLMAP [39], as

$$\hat{\boldsymbol{\theta}}^{(i)} = \underset{\boldsymbol{P}_l, \boldsymbol{X}_m, \boldsymbol{\theta}_{rd}}{\arg\min} \sum_{\forall (l,m) \in \boldsymbol{S}} \mathcal{L}(||\boldsymbol{p}^{(i)}(\boldsymbol{P}_l, \boldsymbol{X}_m, \boldsymbol{\theta}_{rd}) - \boldsymbol{u}_{l,m}||), \tag{1}$$

where l is the index of the camera and m is the index of the 3D point, $\boldsymbol{S}$ is an index set that determines which point is seen by which camera, $\boldsymbol{u}_{l,m} \in \mathbb{R}^2$ are observations of the 3D point $\boldsymbol{X}_m$ in the camera $\boldsymbol{P}_l$, $\boldsymbol{p}^{(i)}$ is the projection function for the camera model $\boldsymbol{M}_i$ and $\mathcal{L}$ is a loss function.

The estimated sub-reconstructions $\hat{\boldsymbol{\theta}}^{(i)}$, for model $\boldsymbol{M}_i$, consist of estimated parameters $\hat{\boldsymbol{P}}_l^{(i)}, \hat{\boldsymbol{X}}_m^{(i)}, \hat{\boldsymbol{\theta}}_{rd}^{(i)}$. The corresponding projections $\hat{\boldsymbol{u}}_{l,m}^{(i)}$ by the camera model $\boldsymbol{M}_i$ satisfy

$$\hat{\boldsymbol{u}}_{l,m}^{(i)} = \boldsymbol{p}^{(i)}(\hat{\boldsymbol{P}}_l^{(i)}, \hat{\boldsymbol{X}}_m^{(i)}, \hat{\boldsymbol{\theta}}_{rd}^{(i)}) \qquad \forall (l,m) \in \boldsymbol{S}. \tag{2}$$

Observations $\boldsymbol{u}_{l,m}$ that satisfy $\mathcal{L}(\hat{\epsilon}_{l,m}^{(i)}) = \mathcal{L}(||\boldsymbol{u}_{l,m} - \hat{\boldsymbol{u}}_{l,m}^{(i)}||) < \delta$, for some threshold δ, are the inliers of the model $\boldsymbol{M}_i$. Let us denote the index set of all inliers for the camera model $\boldsymbol{M}_i$ by $\boldsymbol{S}^{(i)}$. We assume that the sub-reconstructions $\hat{\boldsymbol{\theta}}^{(i)}$ contain inliers only.

Assuming one camera model $\boldsymbol{M_i}$, we skip the index $()^{(i)}$ whenever it is clear from context. The camera $\hat{\boldsymbol{P}}_l \in \mathbb{R}^9$ is composed of the focal length $\hat{f} \in \mathbb{R}$, the principal point $\hat{\text{pp}} \in \mathbb{R}^2$, Euler vector $\hat{\text{aa}} \in \mathbb{R}^3$ (i.e. a rotation axis multiplied by a rotation angle, which can be transformed into a rotation matrix by the function $\text{R}(\hat{\text{aa}}) \in \mathbb{R}^{3\times3}$), and the translation $\hat{\boldsymbol{t}} \in \mathbb{R}^3$. In the following, we also skip the indices $()_{l,m}$ for the brevity.

The radial distortion function $\boldsymbol{h}$, in general, depends on the distance r of the image point $\boldsymbol{u}$ from the distortion center, which we assume to be in the principal point $\hat{\text{pp}}$. The general projection function $\boldsymbol{p}^{(i)}$ for the camera model $\boldsymbol{M_i}$ under radial distortion can then be written as

$$\hat{\boldsymbol{u}}^{(i)} = \hat{f}\, \boldsymbol{h}^{(i)}(\hat{r}^2, \hat{\boldsymbol{\theta}}_{rd}) \underline{\hat{\boldsymbol{u}}} + \hat{\text{pp}} \ , \tag{3}$$

where $\hat{r}^2 = ||\underline{\hat{\boldsymbol{u}}}||^2$ and $\underline{\hat{\boldsymbol{u}}}$ is the projection in the image plane before applying radial distortion

$$\underline{\hat{\boldsymbol{u}}} = \begin{bmatrix} \underline{\hat{u}}_1 \\ \underline{\hat{u}}_2 \end{bmatrix} = \begin{bmatrix} \underline{\hat{x}}/\underline{\hat{z}} \\ \underline{\hat{y}}/\underline{\hat{z}} \end{bmatrix} . \tag{4}$$

$[\underline{\hat{x}}, \underline{\hat{y}}, \underline{\hat{z}}]^\top$ is the 3D point in camera coordinates obtained by rotating and translating the point $\hat{\boldsymbol{X}}$

$$\begin{bmatrix} \underline{\hat{x}} \\ \underline{\hat{y}} \\ \underline{\hat{z}} \end{bmatrix} = \text{R}(\hat{\text{aa}})\hat{\boldsymbol{X}} + \hat{\boldsymbol{t}} . \tag{5}$$

The radial distortion function is usually modelled as a rational function [25]

$$\boldsymbol{h}^{(i)}(\hat{r}^2, \hat{\boldsymbol{\theta}}_{rd}) = \frac{1 + \hat{k}_1 r^2 + \hat{k}_2 r^4 \ldots \hat{k}_R\, \hat{r}^{2B}}{1 + \hat{d}_1 \hat{r}^2 + \hat{d}_2 \hat{r}^4 \ldots \hat{d}_D\, \hat{r}^{2D}} \ , \tag{6}$$

#	IC	formula
1	AIC [4]	$-2L + 2k$
2	$AICc$ [19]	$-2L + 2k + \frac{2k^2+2k}{N-k-1}$
3	$CAIC$ [9]	$-2L + k(log(N) + 1)$
4	BIC [41]	$-2L + k\, log(N)$
5	HQC [16]	$-2L + 2k\, log(log(N))$
6	MDL [34]	$-L + \frac{1}{2}k\, log(N)$
7	SSD [35]	$-2L + k\, log(\frac{N+2}{24}) + 2log(k+1)$
8	$GAIC$ [20]	$R - T + 2(Nd + k)\sigma^2$
9	$GMDL$ [20]	$R - T - (Nd + k)\sigma^2 log(\sigma^2)$
10	$RTIC_{tal}$ [8]	$-2T + 2R^{tal} + 2\frac{kR(I)}{\mathcal{D}(I)}$
11	$RTIC_{hub}$ [8]	$-2T + 2R^{hub} + 2\frac{k(R(I)+\mathcal{D}(S\setminus I)\delta^2)}{\mathcal{D}(I)}$
12	$FRIC_1$ [8]	$\frac{R(I)}{\sigma^2_{\bar{M}}(I)} - \mathcal{D}(I) + k_{\bar{M}} + 2k$
13	$FRIC_2$ [8]	$\frac{R(I)}{\sigma^2_{\bar{M}}(I)} - \mathcal{D}(I) + k_{\bar{M}} + 2k\, log(\mathcal{D}(I))$

Table 1: Summary of the information criteria. Please see the original papers or supplementary material for details.

where $\hat{k}_j$ and $\hat{d}_l$ are parameters of the radial distortion model. The most common models are polynomial (Brown) models with $\hat{d}_l = 0, \forall l$ or division models with $\hat{k}_j = 0, \forall j$. We denote the radial distortion model (6) with the first B non-zero parameters $\hat{k}_j$ and the first D non-zero parameters $\hat{d}_l$ as $\boldsymbol{M_{B|D}}$. $\boldsymbol{M_{0|0}}$ is the simple pinhole camera model with no radial distortion. Different SfM pipelines use different camera models, e.g. COLMAP [39] uses $\boldsymbol{M_{0|0}}$, $\boldsymbol{M_{1|0}}$, $\boldsymbol{M_{2|0}}$, $\boldsymbol{M_{3|3}}$[1], Meshroom [6] uses $\boldsymbol{M_{0|0}}$, $\boldsymbol{M_{3|0}}$ and Theia [45] uses $\boldsymbol{M_{0|0}}$, $\boldsymbol{M_{2|0}}$.

3.1. Accuracy of observations

Each observation $\boldsymbol{u}_{l,m}$ is located with its accuracy depending, e.g., on the view-angle, the keypoint contrast *etc*., but not depending on the estimated reprojection error $\hat{\epsilon}_{l,m}^{(i)}$. We assume that $\hat{\boldsymbol{e}}_{l,m}^{(i)} \in \mathcal{N}(0, \Sigma_{u_{l,m}})$, i.e. the residual $\hat{\boldsymbol{e}}_{l,m}^{(i)} = \boldsymbol{u}_{l,m} - \hat{\boldsymbol{u}}_{l,m}^{(i)}$ follows a zero-mean Normal distribution with covariance matrix $\Sigma_{u_{l,m}} \in \mathbb{R}^{2\times2}$. The covariance matrix is found according to [13] for each keypoint. Each keypoint is described by an affine region [28]. We use the DSP-SIFT detector [12] to find the regions, but any SOTA detector [10, 18, 47] may be used. The covariance matrix for each affine region is computed as the scaled inversion of the Structure Tensor [13]. The covariance matrix of all observations Σ_u is composed of blocks $\Sigma_{u_{l,m}}$ on the diagonal.

4. Comparison of ICs

The existing ICs can be seen as compositions of two terms. The first term expresses the goodness of the fit and the second term expresses the bias correction of the first term.

[1] In COLMAP, $\boldsymbol{M_{3,3}}$ includes also tangential distortion terms.

The standard definition of ICs [4, 19, 9, 27, 41, 16, 34, 35, 20, 37, 26, 8] assumes the goodness of the fit realized by the log-likelihood L of k estimated parameters. The log-likelihood L is usually derived from the Normal distribution, which approximates the true distribution of the parameters. If the number of observations is constant, which is the assumption of most of the existing ICs, the log-likelihood L can be decomposed into a constant term T and the sum of squared weighted residuals R, as $L = T - R$, and the constant term T may be ignored [4, 19, 9, 27, 41, 16, 34, 35, 20, 37, 26].

However, for the problem of camera model selection the term T depends on the number of observations N, which varies for different camera models and different reprojection error thresholds. Therefore, we keep the term T in all ICs that we will later use for comparison with our new criterion. All the compared ICs are summarized in Table 1. We can divide them into standard ICs (1-9) and robust ICs (10-13). The standard (resp. robust) ICs assume that the parameters $\hat{\boldsymbol{\theta}}$ are estimated by a maximum likelihood (ML) estimator (resp. M estimator). The ML estimators minimize the squared reprojection errors. The M estimators minimize a robust loss function, e.g., the Tarwal or Huber loss function [8]. Note that for our task $\hat{k}^{(i)} = 9\hat{U}^{(i)} + 3\hat{V}^{(i)} + \mathcal{D}(\hat{\boldsymbol{\theta}}_{rd}^{(i)})$, where $\mathcal{D}(\hat{\boldsymbol{\theta}}_{rd}^{(i)})$ is the dimension of estimated radial distortion parameters. A detailed description of all ICs is provided in the original papers and the supplementary material.

5. Accuracy-based criterion (AC)

Here we describe our new Accuracy-based criterion and our new camera model selection method.

The accuracy of observations $\Lambda_u = \Sigma_u^{-1}$, i.e. the information matrix Λ_u, which is the inversion of the covariance matrix Σ_u, can be propagated to the reconstruction $\boldsymbol{\theta}^{(i)}$. However, in practice, each individual reconstruction $\boldsymbol{\theta}^{(i)}$ is in a different coordinate system with different gauge of the covariance matrix. To have comparable values, we need to (i) specify the gauge of the coordinate systems, (ii) specify the gauge of the covariance matrix [13, p.109].

The number of reconstruction parameters $\hat{k}^{(i)}$ varies for different initial pairs of the reconstruction, different reprojection error thresholds δ, and different camera models. To obtain a comparable representation of the accuracy, we need to define a subset of parameters $\boldsymbol{\theta}_A$, which are common to all the reconstructions, i.e. $\boldsymbol{\theta}_A \subset \boldsymbol{\theta}^{(i)}, \forall i$ and the remaining parameters $\boldsymbol{\theta}_B^{(i)} = \{\boldsymbol{\theta}^{(i)} \setminus \boldsymbol{\theta}_A\}$. We assume that all estimated reconstruction parameters can be decomposed as $\hat{\boldsymbol{\theta}}^{(i)} = \{\hat{\boldsymbol{\theta}}_A^{(i)}, \hat{\boldsymbol{\theta}}_B^{(i)}\}$.

To specify the gauge of the coordinate system, we chose one reference reconstruction, e.g. $\boldsymbol{M_r} \in \mathcal{M}$. Next, we transform the coordinate system of each reconstruction such that the transformation minimizes the distances of the centres of cameras and the angles between the optical axes of cameras in these reconstructions to the centres and the optical axes of the corresponding cameras in $\hat{\boldsymbol{\theta}}^{(r)}$.

Let us assume in the following that the coordinate systems of reconstructions $\hat{\boldsymbol{\theta}}^{(i)}, \forall i$ were already aligned to one reference coordinate system, e.g., to the coordinate system of $\hat{\boldsymbol{\theta}}^{(r)}$. The next step is to apply *S-transformation* $\mathtt{S}^{(i)}$ for each reconstruction to specify the gauge of the covariance matrix [13] of the reconstruction $\Sigma_{\hat{\theta}^{(i)}}$. To do so, we need to write the general equation for the propagation of the accuracy Λ_u from observation $\boldsymbol{u}^{(i)}$ to $\hat{\boldsymbol{\theta}}^{(i)}$.

$\mathtt{J}_A^{(i)}$ denotes the Jacobian of $\boldsymbol{p}^{(i)}$ w.r.t. $\boldsymbol{\theta}_A$ evaluated in $\hat{\boldsymbol{\theta}}_A^{(i)}$ and $\mathtt{J}_B^{(i)}$ is the Jacobian of $\boldsymbol{p}^{(i)}$ w.r.t. $\boldsymbol{\theta}_B^{(i)}$ evaluated in $\hat{\boldsymbol{\theta}}_B^{(i)}$. Then, we can write the propagation as

$$\Lambda_\theta^{(i)} = \begin{bmatrix} \Lambda_{AA}^{(i)} & \left(\Lambda_{AB}^{(i)}\right)^\mathsf{T} \\ \Lambda_{AB}^{(i)} & \Lambda_{BB}^{(i)} \end{bmatrix} = \begin{bmatrix} \left(\mathtt{J}_A^{(i)}\right)^\mathsf{T} \\ \left(\mathtt{J}_B^{(i)}\right)^\mathsf{T} \end{bmatrix} \Lambda_u \begin{bmatrix} \mathtt{J}_A^{(i)} & \mathtt{J}_B^{(i)} \end{bmatrix}, \tag{7}$$

where $\Lambda_\theta^{(i)}$ is a symmetric positive semi-definite matrix with 7 degrees of freedom, and $\Lambda_{AA}^{(i)}, \Lambda_{BB}^{(i)}, \Lambda_{AB}^{(i)}$ are blocks of $\Lambda_\theta^{(i)}$ corresponding to $\boldsymbol{\theta}_A$ and $\boldsymbol{\theta}_B^{(i)}$.

Note that we will work directly with the information matrix $\Lambda_\theta^{(i)}$ and not the covariance matrix, since the covariance matrix requires a pseudo-inversion of dense matrix $\Lambda_\theta^{(i)} \in \mathbb{R}^{\hat{k}^{(i)} \times \hat{k}^{(i)}}$ ($\hat{k}^{(i)}$ is the number of parameters $\hat{\theta}^{(i)}$) leading to numerical instabilities. Further, we define a transformation matrix $\mathtt{S}^{(i)}$ such that the covariance related to the common parameters $\boldsymbol{\theta}_A$ is independent from $\boldsymbol{\theta}_B^{(i)}$, i.e.

$$\begin{bmatrix} \Lambda_A^{(i)} & 0 \\ 0 & \Lambda_B^{(i)} \end{bmatrix} = \mathtt{S}^{(i)} \begin{bmatrix} \Lambda_{AA}^{(i)} & \left(\Lambda_{AB}^{(i)}\right)^\mathsf{T} \\ \Lambda_{AB}^{(i)} & \Lambda_{BB}^{(i)} \end{bmatrix} \left(\mathtt{S}^{(i)}\right)^\mathsf{T}, \tag{8}$$

where the matrix $\mathtt{S}^{(i)}$ is

$$\mathtt{S}^{(i)} = \begin{bmatrix} \mathtt{1} & -\left(\Lambda_{AB}^{(i)}\right)^\mathsf{T} \left(\Lambda_{BB}^{(i)}\right)^{-1} \\ 0 & \mathtt{1} \end{bmatrix}. \tag{9}$$

The matrices $\mathtt{1}$ in (9) are identity matrices of suitable sizes. The submatrix $\Lambda_A^{(i)}$ has the same dimension $\hat{k}_A$ for all n tested camera models $\boldsymbol{M_i}$, i.e. $\hat{k}_A = \mathcal{D}(\Lambda_A^{(i)}) = \mathcal{D}(\Lambda_A^{(j)}), \forall i, j \in \{1, \ldots, n\}$. This way we specified the gauge of the coordinate system as well as the gauge of the covariance matrix. We can express $\Lambda_A^{(i)}$ from the previous equations as the Schur complement of a block matrix, i.e.

$$\Lambda_A^{(i)} = \Lambda_{AA}^{(i)} - \left(\Lambda_{AB}^{(i)}\right)^\mathsf{T} \left(\Lambda_{BB}^{(i)}\right)^{-1} \Lambda_{AB}^{(i)}. \tag{10}$$

Let us denote the Moore-Penrose (MP) inversion by $^+$. The largest eigenvalue $\lambda_{max}(\Sigma_A^{(i)})$ of the covariance matrix $\Sigma_A^{(i)} = (\Lambda_A^{(i)})^+$ is the squared magnitude of the main diagonal of the equiprobability ellipsoid defined by $\Sigma_A^{(i)}$, i.e. the variance of the most uncertain parameter in $\hat{\boldsymbol{\theta}}_A^{(i)}$ [13, p.32].

The computation of $\Sigma_A^{(i)}$ is challenging [33]. Therefore, we use the relationship of $\Sigma_A^{(i)}$ and $\Lambda_A^{(i)}$ to overcome this problem. The scene has exactly seven degrees of freedom (i.e. all the parameters can be translated, rotated and scaled without changing the reprojection error). Let us assume that the λ function returns the eigenvalues in ascending order. Therefore the first seven eigenvalues are zero, $\lambda_{1\ldots7}(\Lambda_A^{(i)}) = \lambda_{1\ldots7}(\Sigma_A^{(i)}) = \mathbf{0}$, and other eigenvalues are

$$\lambda_{\hat{k}_A^{(i)}-j}(\Sigma_A^{(i)}) = \frac{1}{\lambda_{8+j}(\Lambda_A^{(i)})} \quad \forall j \in \{0, \ldots, \hat{k}_A^{(i)} - 8\}. \quad (11)$$

Having a degenerate configuration can be detected by the condition number $\Lambda_A^{(i)}$ and the eight eigenvalue $\lambda_8(\Lambda_A^{(i)})$ is related to the most uncertain parameter. This is the reason why we may use $\lambda_8(\Lambda_A^{(i)})$ as a meaningful accuracy criterion. However, we propose to use the trace

$$\text{AC} = \text{tr}(\Lambda_A^{(i)}). \quad (12)$$

instead of $\lambda_8(\Lambda_A^{(i)})$. The sum of all eigenvalues of $\Sigma_A^{(i)}$ for each individual reconstruction $\hat{\boldsymbol{\theta}}^{(i)}$ is the sum of all variances of the parameters, i.e. a smaller number means that the parameters are more accurately determined on average. Since $\text{tr}(\Lambda_A^{(i)})$ equals $1/\text{tr}(\Sigma_A^{(i)})$, larger values correspond to more accurate reconstructions. The advantage of using $\text{tr}(\Lambda_A^{(i)})$ instead of $\lambda_8(\Lambda_A{}^{(i)})$ lies in computational efficiency. It is enough to compute the diagonal of $\Lambda_A^{(i)}$.

6. Camera model selection method (ACS)

Our ACS method selects the camera model with the largest AC (12), see Algorithm 1. To evaluate AC, we need to calculate the parameters $\hat{\boldsymbol{\theta}}^{(i)}$ for each model $\boldsymbol{M_i} \in \mathcal{M}$ and the covariance matrix Σ_u of the observations. According to our experiments, it is sufficient to use $5 \leq K \leq 15$ registered cameras to estimate the camera model reliably. We run the SfM such that it first tries to register given K cameras and only if it fails it moves to other cameras. If the geometry between images fits the model $\boldsymbol{M_i}$, the SfM pipeline will register all given K cameras in the predefined order. If the model does not fit the data, SfM usually tries to register other cameras, which usually takes long time. We denote the fastest reconstruction time needed to register K images as T_1 and stop the other SfM processes when time limit $T_d = \gamma T_1$ is reached. Here γ is set empirically based on the input data and the difference between the simplest and the most complex model from $\mathcal{M}$. The models that were not able to register K images within T_d time are discarded. After the sub-reconstructions are found we select a subset of parameters common to all sub-reconstructions $\boldsymbol{\theta}_A \subseteq \boldsymbol{\theta}^{(i)}, \forall i$.

Input: A finite set of images $\boldsymbol{Z} = \{\mathtt{Z}_1, \mathtt{Z}_2, \ldots, \mathtt{Z}_U\}$; reprojection threshold δ; the number of registered cameras K; time factor γ; a finite set of camera models $\mathcal{M} = \{\boldsymbol{M_1}, \boldsymbol{M_2}, \ldots, \boldsymbol{M_n}\}$
Output: the selected camera model $\boldsymbol{M_b}$; calibration parameters for the selected model $\hat{\boldsymbol{\theta}}_b$

$T_d \leftarrow \infty$, $\boldsymbol{Q} \leftarrow \emptyset$, AC $\leftarrow \emptyset$
$\boldsymbol{O} \leftarrow$ get_registration_order($\boldsymbol{Z}$)
`// run in parallel until` T_d `elapses`
for $i \leftarrow 1$ ***to*** n **do**
  $[\hat{\boldsymbol{\theta}}^{(i)}, T_1] \leftarrow$ SfM($\boldsymbol{Z}$, $\boldsymbol{M_i}$, δ, K, $\boldsymbol{O}$)
  if $T_d = \infty$ **then**
    $T_d \leftarrow \gamma T_1$
  end
  $\boldsymbol{Q} \leftarrow \{\boldsymbol{Q}, \hat{\boldsymbol{\theta}}^{(i)}\}$
end
`// finished sub-reconstructions` $\boldsymbol{Q}$
$\bar{\boldsymbol{\theta}} \leftarrow$ get_largest_reconstruction($\boldsymbol{Q}$)
$\boldsymbol{S_A} \leftarrow$ find_common_parameters($\boldsymbol{Q}$)
for $\hat{\boldsymbol{\theta}}^{(i)} \in \boldsymbol{Q}$ **do**
  $\hat{\boldsymbol{\theta}} \leftarrow$align_coordinates($\hat{\boldsymbol{\theta}}^{(i)}$, $\bar{\boldsymbol{\theta}}$)
  $[\hat{\boldsymbol{\theta}}_A, \hat{\boldsymbol{\theta}}_B] \leftarrow$ split_parameters($\hat{\boldsymbol{\theta}}$, $\boldsymbol{S_A}$)
  $[\mathtt{J}_A, \mathtt{J}_B] \leftarrow$ get_derivatives($\boldsymbol{M_i}$, $\hat{\boldsymbol{\theta}}_A$, $\hat{\boldsymbol{\theta}}_B$)
  $[\Lambda_{AA}, \Lambda_{AB}, \Lambda_{BB}] \leftarrow$ get_inform_mat($[\mathtt{J}_A, \mathtt{J}_B]$)
  $\Lambda_A^{(i)} \leftarrow$ get_schur_complement($[\Lambda_{AA}, \Lambda_{AB}, \Lambda_{BB}]$)
  AC $\leftarrow \{$AC, $\text{tr}(\Lambda_A^{(i)})\}$
end
$\boldsymbol{M_b} \leftarrow$ select_model(AC, $\mathcal{M}$)

Algorithm 1: The ACS method: SfM($\boldsymbol{Z}$,$\boldsymbol{M_i}$,δ,K,$\boldsymbol{O}$) applies SfM with camera model $\boldsymbol{M_i}$ and reprojection error threshold δ until K images from the queue $\boldsymbol{O}$ are registered or time limit T_d is exceeded. align_coordinates($\hat{\boldsymbol{\theta}}$,$\bar{\boldsymbol{\theta}}$) specifies the gauge of the coordinates via the transformation between the coordinate systems $\hat{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\theta}}$. split_parameters($\hat{\boldsymbol{\theta}}$,$\boldsymbol{S_A}$) splits the parameters into common parameters $\boldsymbol{S_A}$ for all models $\hat{\boldsymbol{\theta}}_A$ and the remaining parameters $\hat{\boldsymbol{\theta}}_B$. get_derivatives($\boldsymbol{M_i}$,$\hat{\boldsymbol{\theta}}_A$,$\hat{\boldsymbol{\theta}}_B$) computes partial derivatives for a given model $\boldsymbol{M_i}$ and the estimated parameters $\hat{\boldsymbol{\theta}}_A$, $\hat{\boldsymbol{\theta}}_B$. get_inform_mat($[\mathtt{J}_A, \mathtt{J}_B]$) uses equation 7 to compute blocks of the information matrix $\Lambda_\theta{}^{(i)}$. get_schur_complement($[\Lambda_{AA}, \Lambda_{AB}, \Lambda_{BB}]$) applies equation 10 to compute $\Lambda_A^{(i)}$. select_model(AC,$\mathcal{M}$) selects the model with the largest accuracy criterion AC.

7. Learned threshold (LACS)

The model selected by the proposed ACS method depends on a threshold on the weighted residuals $\bar{e}_{l,m}$, see Fig. 4,

$$\bar{e}_{l,m} = \sqrt{(\hat{\boldsymbol{u}}_{l,m} - \boldsymbol{u}_{l,m})^{\mathsf{T}} \Sigma_{u_{l,m}}^{-1} (\hat{\boldsymbol{u}}_{l,m} - \boldsymbol{u}_{l,m})}. \quad (13)$$

For different thresholds, different camera models may be selected. On the other hand, the values of the AC criterion for different inlier thresholds provide additional information for improving the robustness of the ACS method. Here we use this information and propose a learning-based extension of the ACS method (LACS). Our LACS method will take the values of the AC criteria for different thresholds as input and will output the "best" camera model. To obtain the input data for our network, the parameters (3D reconstructions) $\hat{\boldsymbol{\theta}}^{(i)}$ are estimated for each considered camera model for the largest reasonable threshold, e.g. $\delta = 2$px. Having $\hat{\boldsymbol{\theta}}^{(i)}$ for the threshold δ, we can assign inlier/outlier labels to all input observations for thresholds smaller than δ, e.g. $\{0.5, 1, 1.5, 2\}$px. Then, we can easily evaluate the values of AC for each camera model and each threshold from this set of thresholds.

The values of the AC criterion for each tested camera model and each threshold, i.e. the $\widetilde{n}$ values from the matrix of size (number of tested camera models n) $\times$ (number of assumed thresholds N_{thr}), are the input to our shallow neural network. This network consisting of 4 hidden fully connected layers (with dimensions: $d_0 = \widetilde{n}$, $d_{1,2} = \widetilde{n}/2$, $d_{3,4} = n$), each followed by leaky ReLU [2] activations. The proposed network learns to identify the "best" camera model based on the AC scores obtained for the different reprojection error thresholds, see Fig. 4 and 1.

8. Experimental evaluation

We evaluate the estimation of a camera model on synthetic and real datasets. Run-time experiments were performed on a single computer with the AMD Ryzen 7 1700X processor. We used COLMAP [39] to compute the SfM models and USfM [33] for computing the Jacobians $\mathtt{J}_A^{(i)}, \mathtt{J}_B^{(i)}$ for all considered camera models $\boldsymbol{M_i}$, $i \in \{1, \ldots, n\}$.

Cameras. To generate realistic synthetic experiments we used a set of eight real cameras consisting of low-cost web-cameras, cellphones, fish-eye and DSLR cameras. We calibrated these cameras using a checkerboard and camera models $\boldsymbol{M_{0|0}}$, $\boldsymbol{M_{1|0}}$, $\boldsymbol{M_{2|0}}$, $\boldsymbol{M_{3|0}}$, $\boldsymbol{M_{4|0}}$, $\boldsymbol{M_{1|1}}$, $\boldsymbol{M_{2|2}}$, and $\boldsymbol{M_{3|3}}$. The obtained parameters were used in synthetic experiments. The table with all calibration parameters is in the suppl. material.

The datasets The synthetic datasets were created based on 13 publicly available ETH datasets [40] with 14-76 images, 2k-85k 3D points and 50k-795k observations. The 3D points supplied with the ETH datasets were projected into virtual cameras with poses equal to the provided ground truth camera poses and internal parameters taken from our own calibration of the 8 real cameras. We added random noise with distribution $\mathcal{N}(0, \Sigma_{u_{i,j}})$ to the projections. To obtain realistic covariance estimates for image points we used the scaled inversion of the structure tensor [13] of the affine regions [12] to extract $4,5$M covariances $\Sigma_{u_{i,j}}$ from 454 images of the ETH dataset. These covariances were randomly assigned to projections $\hat{\boldsymbol{u}}_{i,j}$ to generate 100 new datasets (i.e., 2839 images and $10,3$M keypoints). The real datasets comprise of the ETH datasets and the well known KITTI [14] dataset. In the following we provide a detailed analysis of results on one of the ETH datasets (terrains_rig) and one from KITTI (2011_09_26_drive_0001). The remaining results and a detailed summary of all parameters of the datasets are provided in the suppl. material.

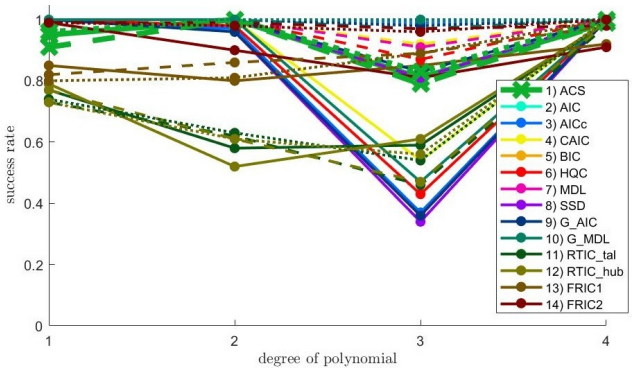

Figure 2: Success rates of different ICs (*c.f.* Table 1) for correctly estimating the degree of a polynomial. Each polynomial was fitted in a statistically optimal way from 100 measurements $\widetilde{y} = \widetilde{f}(\widetilde{x}) + \widetilde{\epsilon}$ with noise $\widetilde{\epsilon} \in \mathcal{N}(0, \widetilde{\sigma}^2)$, where $\widetilde{\sigma}^2 = [10^{-2}, 10^{-3}, 10^{-4}]$ for [solid, dashed, dotted] line.

8.1. Synthetic experiments

This section compares our ACS and LACS with conventional ICs. We start with the task of polynomial fitting to demonstrate the generality of our AC and the properties of ICs on task with a regular statistical model [48]. We generated $120k$ polynomials and let ICs decide the polynomial degree, i.e. $\{1, 2, 3, 4\}$. Our ACS method reached on average 94.1% correctly estimated degrees of polynomials, which is comparable with classical ICs, see Fig. 2. More details about polynomial fitting are in the suppl. material. Further, in this section, we show how employing the covariance matrices of the observations increases the number of correctly identified inliers and the results of camera model selection on synthetically generated datasets.

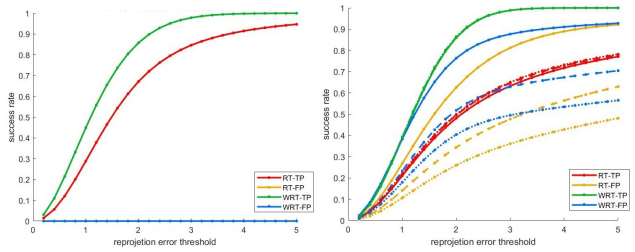

Figure 3: True positive (TP) and false positive (FP) inliers for residual threshold (RT) and weighted residual threshold (WRT). [solid;dashed;doted] lines corresponds to [10%; 30%; 50%] of mismatches.

The outlier filtering in SOTA reconstruction pipelines [39, 45, 29] is done by thresholding the reprojection errors without taking the uncertainty of the observations [49, 13] into account. We empirically show that considering the uncertainty leads to significant increase in the number of correctly identified inliers. We generated mismatches by permuting the 3D point indices in $\boldsymbol{S}$, see Equation 2. We permuted $\{10\%, 30\%, 50\%\}$ of both randomly and systematically chosen ids. For random permutations, we get zero false positive inliers and an increase of about 19.4% for true inliers for a 1.6px reprojection error threshold, see Fig. 3. Real reconstruction errors were simulated systematically by permuting pairs of observations $\boldsymbol{u}_t$, $\boldsymbol{u}_s$ with the smallest distance between each other $||\boldsymbol{u}_t - \boldsymbol{u}_s||$. For all amounts of outliers ($\{10\%, 30\%, 50\%\}$), the number of true positive inliers increased, e.g., for $\delta = 1.6$px the number of true positive inliers increased by 31.7%. The threshold for reprojection error without uncertainty consideration produces approximately the same amount of true inliers for $\delta = 4$px. However, it also increases the false positives by about 9.7%, see Fig. 3.

Camera model estimation was tested on 1k synthetic scenes for each of the camera models $\{\boldsymbol{M_{0|0}}, \boldsymbol{M_{1|0}}, \boldsymbol{M_{2|0}}, \boldsymbol{M_{3|0}}, \boldsymbol{M_{4|0}}\}$ and for $K \in \{5,10,15\}$. These models correspond to the set of most commonly used radial distortion models in SfM pipelines. The synthetic scenes where composed by using $K-1$ neighbouring cameras around one randomly selected camera in our synthetic datasets. Next, we added up to 20% of outliers by systematic permutation (see above) of 3D points ids in $\boldsymbol{S}$ to simulate real reconstruction errors. Each synthetic scene was examined with all ICs from Table 1, ACS and LACS. The success rate of correctly selected camera models is shown in Fig. 4.

Further, we trained our LACS classification network (CN) in PyTorch [32] using all synthetic datasets from the previous experiments, see results in Table 2. We split the

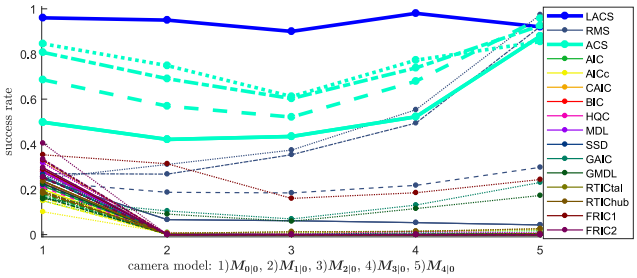

Figure 4: Success rates of the information criteria for camera model selection. The lines {solid, long dashed, short dashed, doted} correspond to a weighted reprojection error threshold of $\{0.5, 1, 1.5, 2\}$px. LACS uses all thresholds.

ACS for $\delta = 2$px					LACS				
0.84	0.13	0.01	0.01	0.01	**0.96**	0.02	0.02	0.00	0.00
0.10	**0.72**	0.12	0.04	0.02	0.00	**0.95**	0.05	0.00	0.00
0.04	0.19	**0.63**	0.09	0.06	0.00	0.10	**0.90**	0.00	0.00
0.00	0.01	0.03	**0.81**	0.14	0.00	0.00	0.00	**0.98**	0.02
0.08	0.02	0.04	0.04	**0.83**	0.00	0.00	0.02	0.06	**0.92**

Table 2: Confusion matrices for ACS and LACS methods evaluated on synthetic data, see Fig. 4. The ACS method selects the camera model with the largest AC for a threshold $\delta = 2$px. The LACS profits from all thresholds of reprojection errors $\delta \in \{0.5, 1, 1.5, 2\}$px. Rows correspond to the ground truth models and columns to the selected camera models from the set $\{\boldsymbol{M_{0|0}}, \boldsymbol{M_{1|0}}, \boldsymbol{M_{2|0}}, \boldsymbol{M_{3|0}}, \boldsymbol{M_{4|0}}\}$.

datasets into training, validation and evaluation parts with respective ratios $[0.8, 0.1, 0.1]$. The input to the CN were the values of ACS normalized by the function

$$f_{norm}(\boldsymbol{x}_j) = \alpha \frac{(\boldsymbol{x}_j - \min(\boldsymbol{x}_j))}{(\max(\boldsymbol{x}_j) - \min(\boldsymbol{x}_j))} + \beta \qquad (14)$$

where $\alpha = 4$, $\beta = 1$ and $\boldsymbol{x}_j$ denotes j-th column of an input sample, i.e. the ACS values for one of the thresholds $\delta \in \{0.5, 1, 1.5, 2\}$px. We used the Adam [22] optimizer with learning rate $lr = 10^{-4}$ and standard Cross Entropy Loss function. To avoid overfitting, we trained for 4k epochs and select the model with lowest validation loss.

The model classifier dependence on the number of images is shown in Table 3. The accuracy is naturally growing with an increasing number of observations [33] and the differences between camera models become more obvious, see Fig. 5. We show, using a large set of simulated datasets, that small sub-reconstructions, e.g. with 15 images, are descriptive enough to make decisions about the camera model. More images and more thresholds can be used to achieve a higher accuracy of camera model classification in practise.

Classifier / K	5	10	15
ACS [0.5px]	0.47	0.53	0.51
ACS [1.0px]	0.55	0.65	0.66
ACS [1.5px]	0.56	0.70	0.70
ACS [2.0px]	0.48	0.68	0.76
LACS	**0.68**	**0.83**	**0.93**

Table 3: The success ratios of the ACS and LACS classifiers w.r.t. an increasing number of registered cameras K.

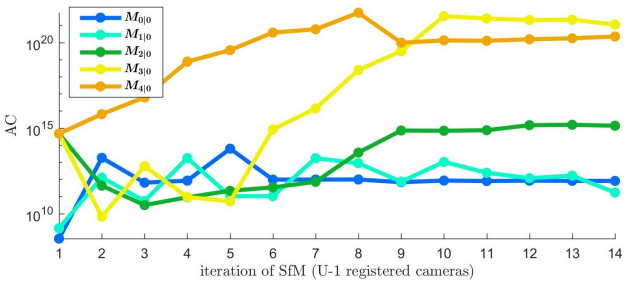

Figure 5: The dependence of AC criteria on iterations for the *terrains_rig* dataset [40] for 1-15 registered cameras.

$\mathcal{M}$	T_1	T_{all}	U	V	N	$\frac{\sqrt{R}}{N}$	Q
$\boldsymbol{M_{0\|0}}$	**7.0**	1070.2	34	6.7	64.1	0.7	2.5
$\boldsymbol{M_{1\|0}}$	32.0	626.8	**165**	**17.5**	210.5	0.9	7.0
$\boldsymbol{M_{2\|0}}$	16.9	212.7	**165**	**17.5**	**210.9**	0.8	6.4
$\boldsymbol{M_{3\|0}}$	17.6	175.7	**165**	17.3	210.2	0.7	3.1
$\boldsymbol{M_{4\|0}}$	29.5	215.1	**165**	17.2	209.6	0.7	3.7
$\boldsymbol{M_{1\|1}}$	12.0	**172.3**	**165**	17.2	209.6	0.7	3.5
$\boldsymbol{M_{2\|2}}$	94.7	1443.8	**165**	17.3	210.1	0.8	3.7
$\boldsymbol{M_{3\|3}}$	83.4	1545.0	18	4.3	27.0	**0.5**	**0.8**
$\boldsymbol{M_{0\|0}}$	113.5	1323.6	**114**	34.4	305.4	0.8	–
$\boldsymbol{M_{1\|0}}$	91.7	1401.3	**114**	52.3	424.2	0.6	–
$\boldsymbol{M_{2\|0}}$	**84.2**	1407.1	**114**	64.7	502.7	0.6	–
$\boldsymbol{M_{3\|0}}$	105.7	**1272.2**	**114**	**66.2**	**504.5**	0.6	–
$\boldsymbol{M_{4\|0}}$	-	2238.4	0	0	0	–	–
$\boldsymbol{M_{1\|1}}$	206.1	1628.0	**114**	64.7	496.7	0.6	–
$\boldsymbol{M_{2\|2}}$	-	**431.2**	12	7.6	104.8	**0.4**	–
$\boldsymbol{M_{3\|3}}$	-	1543.2	0	0	0	–	–

Table 4: Evaluation on the *terrains_rig* [40] (with GT) and *2011_09_26_drive_0001* KITTI [14] (without GT) datasets. COLMAP [40] was used with a 2px reprojection error threshold. $T_1[sec]$ is the time for registering K cameras, $T_{all}[sec]$ is the run-time of SfM, U is the number of registered cameras, V and N are the number of 3D points and number of observations (in thousands), $\frac{\sqrt{R}}{N}[px]$ is the mean reprojection error, and $Q[cm]$ is the mean distance of the estimated camera centres to the GT. Camera models $\boldsymbol{M}$ which exceed the time limit $T_d = \gamma T_1$ (where $\gamma = 5$) for registering $K = 15$ cameras are red, the best values are bold, and the model selected by LACS is inside a rectangle.

8.2. Real experiments

This section compares results of the COLMAP pipeline [39] with and without the proposed camera model selection methods, see Table 4 and Fig. 1, 5. The COLMAP reconstruction process follows several steps. First, we select the initial pair and the order of the images for registration into the partial reconstructions. Second, COLMAP registers new cameras, triangulates 3D points and optimizes the partial reconstruction using Bundle Adjustment (BA) [3]. Third, the partial reconstruction is checked for degeneracies and 3D points are filtered if they do not satisfy several conditions, e.g. a minimal triangulation angle, an absolute reprojection error. Fourth, the registered cameras are filtered if they do not contain enough 2D-3D correspondences. We observed that the run-time for sub-optimal camera models is larger than in the optimal case. This is caused mostly either by overfitting of unnecessary camera parameters in the case of more complex camera models or by repetitive cycles of adding, optimizing, and removing of 3D points in the case of too simple camera models. If the camera model is too overparameterized, we observe that all the 3D points are usually removed after the registration of few (e.g., < 15) cameras and the reconstruction starts from scratch. These unsuccessful trials increased the reconstruction time from 175.7sec to 1545sec in case of the *terrains_rig* dataset [40]. Our methods ACS and LACS overcome these problems by automatic selection of the camera model. The time overhead for preprocessing (e.g., $T_d = 40$sec for *terrains_rig*) is negligible in comparison with the increase of the reconstruction time, e.g., 451.1sec for $\boldsymbol{M_{1|0}}$ or 894.5sec for $\boldsymbol{M_{0|0}}$. Note, this speedup was measured for 165 cameras and will be much larger in case of thousands of images. The ACS and LACS methods are developed to select the model with the most accurate reconstruction parameters, and they also provide the intrinsic calibration parameters. We can see that these parameters lead to more accurate reconstruction, Table 4. We observed the same properties on all 13 ETH datasets [40]. These experiments are visualised and summarised in the suppl. material. The actual AC values for $\delta = 2$ are visualised in Fig. 5 showing larger AC values for the more accurate reconstructions from Fig. 1.

9. Conclusion

We have presented a new practical method for the automatic selection of camera models in SfM pipelines. Our approach combines principled design with machine learning-based fine-tuning to achieve good results that go beyond the state of the art. We show that our approach achieves superior performance on publicly available data sets for 3D reconstruction. Our data and codes are available at https://github.com/michalpolic/unc_model_selection.

References

[1] Capturing Reality. https://www.capturingreality.com/. 1

[2] A. F. Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018. 6

[3] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 8

[4] H. Akaike. A new look at the statistical model identification. In *Selected Papers of Hirotugu Akaike*, pages 215–222. Springer, 1974. 2, 3, 4

[5] C. Albl, A. Sugimoto, and T. Pajdla. Degeneracies in rolling shutter sfm. In *European Conference on Computer Vision*, pages 36–51. Springer, 2016. 1

[6] AliceVision. Meshroom: A 3D reconstruction software., 2018. 1, 3

[7] H. S. Alismail, B. Browning, and M. B. Dias. Evaluating pose estimation methods for stereo visual odometry on robots. In *the 11th International Conference on Intelligent Autonomous Systems (IAS-11)*, January 2011. 1

[8] P. Bouthemy, B. M. T. Acosta, and B. Delyon. Robust model selection in 2d parametric motion estimation. *Journal of Mathematical Imaging and Vision*, pages 1–15, 2019. 2, 3, 4

[9] H. Bozdogan. Model selection and akaike's information criterion (aic): The general theory and its analytical extensions. *Psychometrika*, 52(3):345–370, 1987. 2, 3, 4

[10] M. Brown, G. Hua, and S. Winder. Discriminative learning of local image descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):43–57, 2011. 3

[11] K. P. Burnham and D. R. Anderson. A practical information-theoretic approach. *Model selection and multimodel inference, 2nd ed. Springer, New York*, 2002. 2

[12] J. Dong and S. Soatto. Domain-size pooling in local descriptors: Dsp-sift. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5097–5106, 2015. 3, 6

[13] W. Förstner and B. P. Wrobel. *Photogrammetric Computer Vision*. Springer, 2016. 3, 4, 5, 6, 7

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 6, 8

[15] R. K. Hamad, B. Hamed, and H. Hassonny. The automatic selection of radial distortion models. *International Journal of Computer Applications*, 975:8887. 2

[16] E. J. Hannan and B. G. Quinn. The determination of the order of an autoregression. *Journal of the Royal Statistical Society: Series B (Methodological)*, 41(2):190–195, 1979. 2, 3, 4

[17] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 2

[18] J. Heinly, E. Dunn, and J.-M. Frahm. Comparative evaluation of binary features. In *European Conference on Computer Vision*, pages 759–773. Springer, 2012. 3

[19] C. M. Hurvich and C.-L. Tsai. A corrected akaike information criterion for vector autoregressive model selection. *Journal of time series analysis*, 14(3):271–279, 1993. 2, 3, 4

[20] K.-i. Kanatani. Uncertainty modeling and model selection for geometric inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1307–1319, 2004. 1, 2, 3, 4

[21] K.-i. Kanatani and D. D. Morris. Gauges and gauge transformations for uncertainty description of geometric structure with indeterminacy. *IEEE Transactions on Information Theory*, 47(5):2017–2028, 2001. 1

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[23] K. Kinoshita and L. Lindenbaum. Camera model selection based on geometric aic. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 514–519. IEEE, 2000. 2

[24] Z. Kukelova, J. Heller, M. Bujnak, A. Fitzgibbon, and T. Pajdla. Efficient solution to the epipolar geometry for radially distorted cameras. In *Proceedings of the IEEE international conference on computer vision*, pages 2309–2317, 2015. 1

[25] V. Larsson, T. Sattler, Z. Kukelova, and M. Pollefeys. Revisiting radial distortion absolute pose. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1062–1071, 2019. 1, 3

[26] J. A. Machado. Robust model selection and m-estimation. *Econometric Theory*, 9(3):478–493, 1993. 2, 4

[27] C. L. Mallows. Some comments on c p. *Technometrics*, 15(4):661–675, 1973. 4

[28] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005. 3

[29] P. Moulon, P. Monasse, R. Marlet, and Others. Openmvg. an open multiple view geometry library. https://github.com/openMVG/openMVG. 7

[30] D. Nistr, O. Naroditsky, and J. Bergen. Visual odometry. In *Computer Vision and Pattern Recognition (CVPR)*, pages 652–659, 2004. 1

[31] V. Orekhov, B. Abidi, C. Broaddus, and M. Abidi. Universal camera calibration with automatic distortion model selection. In *2007 IEEE International Conference on Image Processing*, volume 6, pages VI–397. IEEE, 2007. 2

[32] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 7

[33] M. Polic, W. Forstner, and T. Pajdla. Fast and accurate camera covariance computation for large 3d reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 679–694, 2018. 5, 6, 7

[34] J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978. 2, 3, 4

[35] J. Rissanen. Universal coding, information, prediction, and estimation. *IEEE Transactions on Information theory*, 30(4):629–636, 1984. 2, 3, 4

[36] I. Rocco, M. Cimpoi, R. Arandjelovi, A. Torii, T. Pajdla, and J. Sivic. Neighbourhood consensus networks, 2018. 1

[37] E. Ronchetti. Robust model selection in regression. Technical report, PRINCETON UNIV NJ DEPT OF STATISTICS, 1984. 2, 4

[38] T. Sattler, B. Leibe, and L. Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization.

IEEE Trans. Pattern Anal. Mach. Intell., 39(9):1744–1756, 2017. 1

[39] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 3, 6, 7, 8

[40] T. Schöps, J. L. Schönberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 6, 8

[41] G. Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978. 2, 3, 4

[42] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. In *ACM SIGGRAPH'06*, 2006. 1

[43] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision (IJCV)*, 80(2):189–210, 2008. 1

[44] L. Svärm, O. Enqvist, F. Kahl, and M. Oskarsson. City-scale localization for cameras with known vertical direction. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1455–1461, 2017. 1

[45] C. Sweeney. Theia multiview geometry library: Tutorial & reference. http://theia-sfm.org. 3, 7

[46] H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii. InLoc: Indoor visual localization with dense matching and view synthesis. In *cvpr*, 2018. 1

[47] T. Tuytelaars, K. Mikolajczyk, et al. Local invariant feature detectors: a survey. *Foundations and trends® in computer graphics and vision*, 3(3):177–280, 2008. 3

[48] S. Watanabe. A widely applicable bayesian information criterion. *Journal of Machine Learning Research*, 14(Mar):867–897, 2013. 1, 2, 6

[49] B. Zeisl, P. F. Georgel, F. Schweiger, E. G. Steinbach, N. Navab, and G. Munich. Estimation of location uncertainty for scale invariant features points. In *BMVC*, pages 1–12, 2009. 7