# CLAM: Coupled Localization and Mapping with Efficient Outlier Handling

Jonathan Balzer
University of California
Los Angeles, CA 90095
USA
balzer@cs.ucla.edu

Stefano Soatto
University of California
Los Angeles, CA 90095
USA
soatto@cs.ucla.edu

## Abstract

*We describe a method to efficiently generate a model (map) of small-scale objects from video. The map encodes sparse geometry as well as coarse photometry, and could be used to initialize dense reconstruction schemes as well as to support recognition and localization of three-dimensional objects. Self-occlusions and the predominance of outliers present a challenge to existing online Structure From Motion and Simultaneous Localization and Mapping systems. We propose a unified inference criterion that encompasses map building and localization (object detection) relative to the map in a coupled fashion. We establish correspondence in a computationally efficient way without resorting to combinatorial matching or random-sampling techniques. Instead, we use a simpler M-estimator that exploits putative correspondence from tracking after photometric and topological validation. We have collected a new dataset to benchmark model building in the small scale, which we test our algorithm on in comparison to others. Although our system is significantly leaner than previous ones, it compares favorably to the state of the art in terms of accuracy and robustness.*

## 1. Introduction

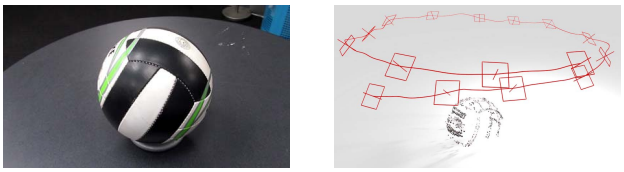
### 1.1. Motivation

We are interested in building models of three-dimensional (3-D) objects for the purpose of manipulation, reconstruction, detection, and recognition. We envision a scenario whereby a video of an object is captured while manipulating or moving around it with a hand-held camera or phone. It is then used to infer, causally and in real time, the coarse geometry and photometry, described in a manner amenable to matching under significant viewpoint and illumination changes. In this context, precision is not critical, but robustness and run-time are. Despite a wealth of work in Structure From Motion (SFM), Simultaneous Localization and Mapping (SLAM) and 3-D reconstruction in general, the goal of a simple yet robust modeling scheme for this scenario remains elusive.

Self-occlusions are dominant in our scenario. As the object moves relative to the camera, different views are revealed, each of which contains only a small number of salient features (Fig. 1). Also, the object occupies a small portion of the visual field, resulting in a small effective field-of-view that presents a challenge for SFM and SLAM. Even with some moderate effort to place the object on a textureless surface, it is often the case that most features are detected in the background, rather than the object.

Global bundle adjustment (BA) methods [13] extract the majority of information of interest in the data, but this comes at a computational cost. Even simpler batch factorization schemes that exploit the weak-perspective nature of small objects can introduce significant latency. Existing real-time SFM systems are too brittle, and SLAM systems that incrementally build a map and localize the viewer relative to it still fail in the presence of significant self-occlusions. Some of these shortcomings have pushed many practitioners to turn to active sensors such as RGB+D systems [19], but these are not well-suited for outdoor lighting.

Although our goal is to eventually use these models for recognition, and therefore our effort naturally relates to [23], here we focus on the reconstruction aspect. We aim for a method that is robust, simple to use, and designed to support classification tasks (Fig. 2), but we do not tackle the



(a) (b)

Figure 1. A close view of an (a) object and (b) its model, visualized as a point cloud. Each point is endowed with a descriptor, and camera motion relative to it comes as a byproduct. Even though it appears that the object dominates the image, it only occupies 20 % of the area of the image.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.204

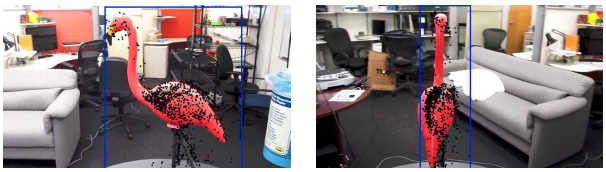


Figure 2. Models acquired by our method can provide support in recognition tasks such as tracking of small-scale objects. In this example, the bounding boxes (blue) are computed from the reprojections of a point cloud (black) segmented in 3-D space.

latter here. In addition, our model can be used as initialization for post-process BA refinement for accurate (sparse) reconstruction and pose estimation, or as one of the building blocks in some of the recent algorithms for dense reconstruction in real time [9, 11, 18, 21, 25, 27].

Perhaps the most closely related work is Klein's and Murray's *Parallel Tracking And Mapping* (PTAM) [15], popular in robotics. It thrives on large, static fronto-parallel scenes, preferably with a dominant plane, but struggles with small objects and significant self-occlusions (Sect. 3). It performs best when the initial motion is orthogonal to the optical axis. We operate under assumptions similar to PTAM: The scene is static, rigid, and for the most part, Lambertian. We assume the camera has been previously calibrated, and both, temporal and spatial scale, are relatively moderate. We aim at real-time performance (w.r.t. the time constant of the acquisition hardware, *i.e.*, video framerate).

### 1.2. Contributions

Our system is considerably simpler than PTAM: While the latter employs a full-fledged epipolar geometry pipeline for initialization (feature selection and tracking, epipolar constraint, incremental bundle adjustment etc.), our approach bypasses all that and trivially starts with all points on the image plane. Empirically, we find that this works faster and better for the conditions discussed above[1]. Our first contribution is a *unified optimization criterion* (Sect. 2.2) that addresses both localization and mapping in a coupled fashion. This would at first seem to go against the wisdom of [15], but presents additional benefits in terms of simplification and management of correspondence, that represents our second contribution: In Sect. 2.3, we propose a putative correspondence mechanism based on tracking to generate inlier hypotheses, and a simple photometric validation mechanism based on a contrast-*invariant* descriptor. It accommodates the percentage of outliers the M-estimator [12], favored in [15] over slower combinatorial or acceptance/rejection sampling methods, can tolerate before breakdown. Our third contribution consists of feeding back the motion estimates from local BA to infer the scale- and rotation-*covariant* component of the descriptor, to reduce nuisance variability. A temporal aggregate of such descriptors can then be the basis for a classification scheme that uses our system for detection, recognition, localization of the learned objects in cluttered scenes. Finally, publicly available datasets for evaluating SLAM methods and recognition of 3-D objects are few. Our fourth contribution is to expand and adapt the benchmarks [8, 17] to the task of small-scale object modeling. An experimental assessment of the performance of our system in comparison with offline SFM (at the high-end) and PTAM (at the low-end) is reported in Sect. 3.

[1] This choice is not well-suited for forward motion, where PTAM also fails.

### 1.3. Other related work

Mair et al. developed a system for close-range 3-D reconstruction [16]. Unlike ours, it includes inertial measurements to cope with drift occuring at large time scales. Early examples of real-time monocular SFM include [14, 20], improved by [10]. The latter two rely on accelerated versions RANSAC for hypothesis generation, based on the 5-point algorithm, and separately triangulate new depths. This leads to the decoupling that we find detrimental to performance and hence wish to avoid. In [6], Engels et al. provide a review of BA, whereas the authors of [24] argue that batch processing based on keyframes performs better marginalization than a causal filtering approach, although the conclusions contravene some of the basic tenets of causal data processing. The setup of Zhang et al. described in [28] coincides with ours because correspondences are collected by tracking. Although their algorithm falls under the category "offline SFM", we will include it as a baseline in our comparison in Sect. 3.

## 2. Method

### 2.1. Setting

In the following, matrices are in bold, vectors in bold italic; points in space $\boldsymbol{X} \in \mathbb{R}^3$ are capitalized when possible. The canonical *pinhole projection* $\pi : \mathbb{R}^3 \to \mathbb{R}^2$, $\boldsymbol{X} \mapsto \boldsymbol{x}$, maps a point in space onto the image plane, where $\boldsymbol{X} = \bar{\boldsymbol{x}}\rho$ for some *depth* $\rho > 0$, and $\bar{\boldsymbol{x}} = (\boldsymbol{x}^\top \ 1)^\top$. We can define a function $\pi_\rho^{-1} : D \to \mathbb{R}^3$ that, given some fixed $\rho$, backprojects $\boldsymbol{x}$ onto $\boldsymbol{X}$. A vantage point at time $t \in \mathbb{N}$ is represented by $g_t = (\mathbf{R}_t, \boldsymbol{T}_t)$, $\mathbf{R}_t \in \mathrm{SO}(3)$, $\boldsymbol{T}_t \in \mathbb{R}^3$, element of the special Euclidean group $\mathrm{SE}(3)$, which transforms the world coordinate system $g_0 = \mathrm{id}$ to the reference frame located at $-\mathbf{R}_t^\top \boldsymbol{T}_t$ with axes parallel to the columns of $\mathbf{R}_t^\top$. The inverse of $g_t$ is denoted by $g_t^{-1}$.

Given a collection of images $\{I_\tau : D \to \mathbb{R}^+\}_{\tau=0}^t$ up to time $t$, we wish to estimate the camera motion and the geometry of the scene in a computationally efficient manner. To this end, we focus on a sparse collection of points

$M = \{\boldsymbol{X}^j\}_{j=1}^m$, the *map*, and their corresponding projections $\boldsymbol{x}_s^i$ in each image where $\boldsymbol{X}^j$ is visible. Since $M$ does not include surface topology, visibility boils down to a combinatorial matching problem; we will address it using tools from robust inference and without explicitly determining the inlier/outlier sets.

### 2.2. A unified inference criterion

Two feature points $\boldsymbol{x}_t^i \in D$ at time $t$, and $\boldsymbol{x}_s^k \in D$ at time $s$, are said to correspond if there exists a location in space $\boldsymbol{X}^j$ that projects to *both*: $\bar{\boldsymbol{x}}_t^i \rho_t^i = \boldsymbol{X}^j = \bar{\boldsymbol{x}}_s^k \rho_s^k$. If such image-to-image correspondences $\boldsymbol{x}_t^i \leftrightarrow \boldsymbol{x}_s^k$ were known, we could compute the inter-frame motion and depths by minimizing the *reprojection error*

$$E_{\mathrm{r}}(g_t, \rho_s^i) := \sum_{i \in V(g_t, g_s)} \|\pi(g_t g_s^{-1} \pi_{\rho_s^i}^{-1}(\boldsymbol{x}_s^i)) - \boldsymbol{x}_t^i\|_1. \quad (1)$$

Here, with an abuse of notation, $i \in V(g_s, g_t)$ indexes image correspondences whereby, after a suitable permutation of indices, $\boldsymbol{x}_s^i$ stands for $\boldsymbol{x}_s^{k(i)}$. Analogously, if scene-to-image correspondences were known, we could infer pose $g_t$ by minimizing the *projection* error

$$E_{\mathrm{p}}(g_t) := \sum_{j \in V(g_t)} \|\pi(g_t \boldsymbol{X}^j) - \boldsymbol{x}_t^j\|_1. \quad (2)$$

Here, $V(g_t)$ indexes scene-to-image correspondences, again with an abuse of notation whereby $\boldsymbol{x}_t^j$ stands for $\boldsymbol{x}_t^{i(j)}$.

These two terms can be combined to provide *coupled* localization and mapping, by minimizing

$$E(g_t, \rho_s^i) := E_{\mathrm{r}} + \alpha E_{\mathrm{p}}, \quad (3)$$

where $\alpha \in \mathbb{R}_+$ is a positive scalar that weighs off the influence of the two separate error terms according to the ratio of $\#V(g_s, g_t)$ and $\#V(g_t)$. Note that (3) covers *all* aspects of a SLAM algorithm: At initialization, when the map is empty, we have $V(g_t) = \emptyset$; $E_{\mathrm{p}} = 0$, and $E = E_{\mathrm{r}}$ is equivalent to the classical BA functional. When image-to-image correspondence fails, $V(g_s, g_t) = \emptyset$ but so long as $\#V(g_t) \geq 3$, minimizing (3) yields a camera pose $g_t$ relative to the now nonempty map. Finally, the general case where $V(g_t), V(g_s, g_t)$ are both nonempty covers the two subproblems of map expansion and motion estimation by minimizing (1) (in lieu of simply triangulating new depths) respectively (2). Note that both are coupled through the variable $g^t$, and such coupling is critical to avoid gauge ambiguities beyond the initialization stage.

It is easy to underestimate the novelty of (3). After all, each of its terms is well-known, and joining them in a linear combination is not a revolutionary idea. Indeed, the two terms reflect the *same* model, and could be further coupled by imposing that $g_s^{-1} \pi_{\rho_s^i}^{-1}(\boldsymbol{x}_s^i)$ in (1) be equal to $\boldsymbol{X}^j$ in (2). Unfortunately, $i \neq j$, and the two energies are summed over different domains $V(g_s, g_t), V(g_t)$.

### 2.3. Correspondence

Of course, the sets $V(g_s, g_t)$ and $V(g_t)$ are *not* known a-priori, and SFM/SLAM methods differ primarily on how they handle the unknown correspondences. One could determine the set of features $V(g_s, g_t)$ that are *co-visible* between $s$ and $t$ by combinatorial matching and voting schemes such as [7] during minimization of the reprojection error. Similarly, one could determine the set of features $V(g_t)$ in the map that are *visible* at time $t$ using the iterative closest-point method [1] or one of its variants. However, this becomes prohibitively expensive when the cardinality of these sets increases. Alternatively, like [15], one could forgo explicit determination of the correspondence sets and use a robust statistical estimator to minimize (3), cf. [12]. Unfortunately, such techniques have a low breakdown point (percentage of outliers) and can still fail in practice.

We adopt an intermediate criterion, where image-to-image putative correspondence is established by short-baseline tracking, and verified using a local contrast-invariant, rotation- and scale-covariant descriptor. If $s$ is the instance when a feature first appears, and the feature is tracked through $\{\boldsymbol{x}_\tau^i\}_{\tau=s}^t$, ideally we would have that $i \in V(g_s, g_\tau)$ for all $\tau = s+1, \ldots, t$. In practice, however, short-baseline trackers are subject to *drift*, and therefore, as time goes by, the track may continue to exist but fail to correspond to a stationary point on the map. Therefore, we design a photometric consistency test based on a local contrast-invariant descriptor. That is, a function of the image in a neighborhood of the tracked point, $\phi(I_\tau | \boldsymbol{x}_\tau^i, g_\tau)$ that is invariant to contrast changes (monotonic continuous transformations $h$ of the image intensity, $h(I_\tau)$, *i.e.*, $\phi(h(I_\tau)|\boldsymbol{x}_\tau^i, g_\tau) = \phi(I_\tau|\boldsymbol{x}_\tau^i, g_\tau)$). We choose BRIEF [3], and test the Hamming distance $d_{\mathrm{H}}$ between the descriptor computed at the time of first appearance $s$ and the current time $t$ against a threshold $\theta > 0$. So,

$$V(g_s, g_t) = \{i \mid d_{\mathrm{H}}\left(\phi(I_s|\boldsymbol{x}_s^i, g_s), \phi(I_t|\boldsymbol{x}_t^i, g_t)\right) \leq \theta.$$

A shortcoming of this initial naive approach is that the neighborhood around $\boldsymbol{x}_s^i$ usually undergoes nonrigid transformations, and the above condition is violated even though the feature remains visible. While in the absence of surface topology, analyzing the map for occlusions is infeasible, analyzing the camera motion relative to it is not. Thus, we feed-back and compensate for domain transformations using portions of the estimated motion $g_t$ (Fig. 3). In this sense, the test is co-variant with respect to scale and in-plane rotation[2]. We can do the same for map-to-image matching, by augmenting each point $\boldsymbol{X}^j$ in the map $M$ with

[2]Notice that in the reprojection error, $g_t$ and $g_s$ only appear as a product, which may suggest that the co-visibility only depends on inter-frame motion $g_t g_s^{-1}$. However, this is not the case, since the dependency on the absolute pose $g_t$ is reflected in the dependency on the scene.

the descriptor(s) from the image taken at the time it first appears.

The novelty of our approach hinges on the joint optimization of (3). This may seem to go counter the results of [15] and others. However, equation (3), combined with tracking and the pre-rejection of inconsistent features, enables us to operate without combinatorial or sequential outlier rejection, significantly decreasing run-time. To the best of our knowledge, nobody has addressed the determination of the visibility sets $V$ by a combination of tracking and contrast-invariant validation. This is the innovation that enables faster outlier handling with standard robust-statistical tools. Also notice that the joint energy functional enables us to expand the map in a robust fashion, unlike [15] that performs triangulation in a separate stage, while lacking outlier filtering altogether.

## 2.4. Implementation

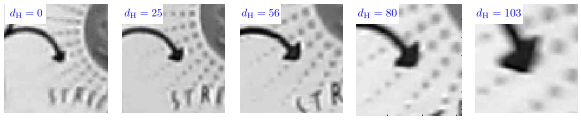
### 2.4.1 Tracking

To warrant sufficient parallax, structure and motion estimation is customarily performed on a subset of the input image sequence called *keyframes*. We rely on *features from accelerated segment tests* (FAST) in such keyframes [22]. These are tracked with the help of the Kanade-Lucas-Tomasi method (KLT) at the original video frame rate. One is not bound to this particular choice of tracker because the central part of our algorithm makes no assumption on how correspondences are established. In contrast, PTAM utilizes a more elaborate, affine deformation model [15]. Two points in two keyframes *correspond* if they are connected by a track. There are two situations in which a track expires: The associated feature either leaves the field of view, or it is about to merge with another tracked feature[3]. We check for occurrences of the second event in the following way: Denote by $\delta(\boldsymbol{x})$ the Dirac distribution and $\boldsymbol{x}^i \in D$ the locations of $n \in \mathbb{N}$ interest points in the image, some of which may already be tracked and others, which have been qualified by the detector as candidates for addition. Then,

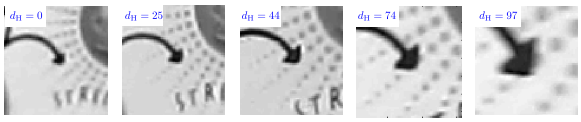
$$I(\Omega) := \int_{\Omega} \sum_{i=1}^{n} \delta(\boldsymbol{x} - \boldsymbol{x}^i) d\boldsymbol{x} \tag{4}$$

counts the number of features in the image region $\Omega \subseteq D$. When this region has the shape of an rectangular neighborhood $B_r(\boldsymbol{x}) := \{\boldsymbol{y} \mid \|\boldsymbol{y} - \boldsymbol{x}\|_\infty < r\}$, we only need to compute $I(D)$, then (4) can be evaluated very efficiently with the help of the integral image trick [4]. At detection, we only admit features $\boldsymbol{x}^i$ for which there exists neighborhoods $B_{r_\mathrm{d}}(\boldsymbol{x}^i)$ such that $I(B_{r_\mathrm{d}}) = 1$. Concurrently, a verification routine removes all tracks that are not unique in a neighborhood $B_{r_\mathrm{v}}$ around their current state. Note that the radii $r_\mathrm{d}$ and $r_\mathrm{v}$ may be chosen differently under the constraint however that $r_\mathrm{d} \geq r_\mathrm{v}$.

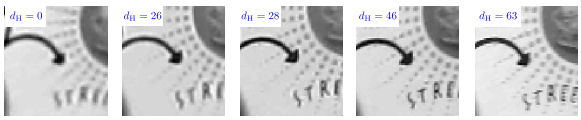
[3]The residual of the KLT model is a bad adviser in this matter because it stipulates brightness conservation which is not given in the vicinity of occlusions. But occlusions and only occlusions are precisely what we wish to cause the death of a track.




(a) Translation-covariant frame




(b) Translation-rotation covariant frame




(c) Translation-rotation-scale covariant frame

Figure 3. Local perspective on the image sequence of different Lagrangian observers attached to a tracked feature: (a) Translation invariance comes naturally with KLT tracking. (b)-(c) In-plane rotations and scale changes are compensated online with the help of motion estimates from local BA. This improves imaged-based occlusion detection. The value of $d_\mathrm{H}$ (see text) appears respectively in the upper left corner.

**Remarks** Pairwise distance comparisons are efficiently implemented with the aid of tree-like data structures. However, they provide no guarantee that the true nearest neighbor of a feature point is found, whereas the integral image construction yields exact results. Hedborg et al. also enforce a distance constraint but only before the pose estimation step [10]. We do so immediately during tracking for two reasons: First, we believe that a tracker prone to drift such as KLT must monitor inter-feature distances to avoid track duplication. Second, convergence of tracks indicates an impending self-occlusion.

### 2.4.2 Numerical optimization

We take the standard reweighted least-squares approach to minimizing (3), which consists of three nested optimization loops: The outermost loop discounts the contribution of each data point to the functional value depending on whether they are outliers with respect to a robust covariance estimate. By default, the resulting weighted nonlinear least-squares problem is then solved with the Levenberg-Marquardt (LM) method, which in turn is equivalent to solving a sequence of linear least-squares problems.

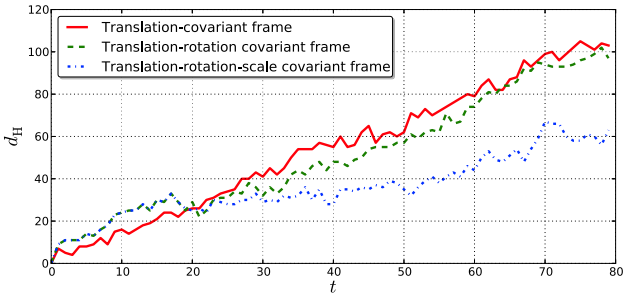
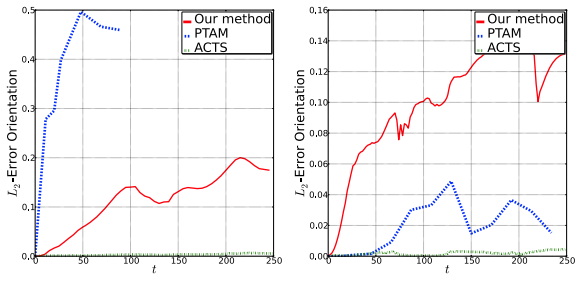



Figure 4. Distance between BRIEF descriptors computed around a tracked feature at time $t$ and the location it occupied at detection. In this example, a gain of almost 50 % is achieved by the feedback of in-plane rotation and forward motion to the track validation module.




(a) Experiment 64: "Occlusion" (b) Experiment 65: "Landscape"

Figure 5. Orientation error over time: (a) Especially in the presence of (self)occlusions, our method outperforms PTAM. (b) Roles are reversed in PTAM's native application scenario.

Let $\mathbf{J} \in \mathbb{R}^{2(m+n)\times(n+6)}$ denote the Jacobian of the squared version of (3), and $\boldsymbol{r} \in \mathbb{R}^{2(m+n)}$ the underlying residual vector. Each LM step seeks the minimum-norm solution to the non-square linear system $\mathbf{J}^\top \boldsymbol{h} = \boldsymbol{r}$, in which $\boldsymbol{h} \in \mathbb{R}^{(n+6)}$ is the update that takes the set of unknown variables closer to the minimizer. The conjugate-gradient least-squares (CGLS) solver achieves this *without* explicitly forming the normal equation $\mathbf{J}^\top\mathbf{J}\boldsymbol{h} = \mathbf{J}^\top\boldsymbol{r}$. Krylov subspace methods have been shown to outperform direct ones, *e.g.*, based on Cholesky factorization, in the realm of large-scale BA. Our numerical studies confirm that the observations made by Byröd and Åström in [2] carry over seamlessly to the present scenario, especially when LM iterates are executed in an *inexact* fashion, *i.e.*, the linear solver is forced to terminate before convergence depending on the value of (3) (as opposed to the residual of the normal equation). In contrast, we found preconditioning of the linear system superfluous because at initialization, the map is relatively small and the amount of time consumed uncritical, then, once the system is online, initial estimates are reasonably close to the desired solution. We set the regularization parameter enforcing the trust region size in a manner according to [2].

Extensive numerical tests have shown that initializing all visible points to be at unit depth (on the image plane) converges rapidly and robustly provided the initial motion is not directly towards or away from the scene. In the latter case, other systems also fail. This can be explained using the analysis in [26], that shows that fronto-parallel motion (perpendicular to the optical axis) yields a residual surface with a large global minimum and an isolated local minimum corresponding to a mirror-symmetric translational direction (the "rubbery motion" ambiguity), quite far from the global minimum. The epipolar geometric pipeline customary in most SFM/SLAM systems like [15] occasionally produces a twisted pair and places the initialization in the wrong basin of attraction, producing worse results than a trivial initialization. To recap, everything here evolves around the central functional (3).

For reweighting, we use the bisquare function, which induces a soft partitioning of data points depending on the covariance of the statistical model for data formation. To obtain a robust estimate of the covariance, one usually forms the medium absolute deviation (MAD) over all entries of $\boldsymbol{r}$. In our opinion, this is not entirely adequate: The regression problem underlying (3) compares measured pieces of data with those predicted by the (re)projection model $\pi$; and the latter is indeed bivariate, a fact which previous works on SFM have failed to acknowledge. In their defense, the MAD, which is the workhorse of robust covariance estimation for univariate data, does not easily generalize to dimension greater than one. To address the issue and effectively reduce the number of inlier/outlier misclassifications, we propose to compute it restricted respectively to the two components of projection and reprojection residuals. This yields a crude approximation to the *Stahel-Donoho estimator* [5].

## 3. Experimental evaluation

### 3.1. A new data set

Ground truth is indispensable in the analysis and evaluation of any SFM/VSLAM system and its subcomponents, yet, obtaining it with complementary sensing modalities may require significant efforts all the while the relationship between data from alternative sources and the "true" scene remains unclear. On the other hand, modern rendering engines – even those available under generous license conditions – allow us to simulate arbitrarily complex real scenes, cf. [8, 10]. We created 18 video sequences of varying degrees of difficulty (occlusions, background clutter, specularities, motion blur, etc.) with the open-source software Blender. The set of synthetic examples is completed by 50 videos of small objects captured by a digital consumer camera. Unlike [17], we refrain from using lab equipment

such as a turntable to best replicate the circumstances under which a user would acquire an object model in real life, *e.g.*, with a mobile phone, and also to discourage the use of motion priors. To obtain a baseline for the entire dataset not just the synthetic part, we processed it by the Automatic Camera Tracking System (ACTS), a state-of-the-art offline SFM algorithm [28]. For the convenience of researchers in the field of SFM/SLAM, besides the raw image sequences, we will distribute the ACTS results as well as the Blender scene files through the first author's website.

### 3.2. Results

All experiments were run on a commodity computer with 12 GB of main memory and four-core processor running at 3.4 GHz. Our C++ code and PTAM was compiled under Linux, ACTS is available for the Windows platform only. To maintain a fair comparison, we have spent a considerable amount of time for finding stable parameter configurations individually for all three methods applied to each example in the dataset. For the sake of reproducibility, a documentation of these configurations will be included in the supplemental material (together with an exhaustive presentation of experimental results). Figs. 8 (b)-(d) depict some of the reconstructed point clouds and trajectories traversed by the origin of the camera coordinate system as well as its orientation – to keep the presentation clear – only at selected locations. The lighting effects these plots contain were added deliberately to attenuate the recovered 3-D structures. Transparent overlays foreshadow ground truth where it is available.

The precision of the BRIEF descriptor used in the experiments was 256 bit. The threshold $\theta$ was set to 96 throughout. Fig. 3 demonstrates the visual effect of motion feedback on the series of image patches supporting the computation of the covariant descriptor. Obviously, the success of stabilization depends fundamentally on the type of input motion. In the displayed case, undoing the impact of in-plane rotation and forward motion of the camera on the occlusion detection mechanism eventually prevents the track from premature deletion (Fig. 4).

For the synthetic examples, we can quantify accuracy independent of gauge by the orientation error in axis-angle representation w.r.t. a common world coordinate system. Fig. 5 illustrates how this error develops over time for two chosen examples, in the first of which the camera moves around two objects mutually occluding each other (and themselves) at two instances in time, and in the second of which, the scene remains fronto-parallel for the duration of the video, also see the bottom rows in Fig. 8(a). Example 65 is a situation in which PTAM excels. Since in example 64, there are no other disturbances present, we can conclude that PTAM is particularly prone to failure when being exposed to occlusions. Fig. 7(b) further supports this claim.

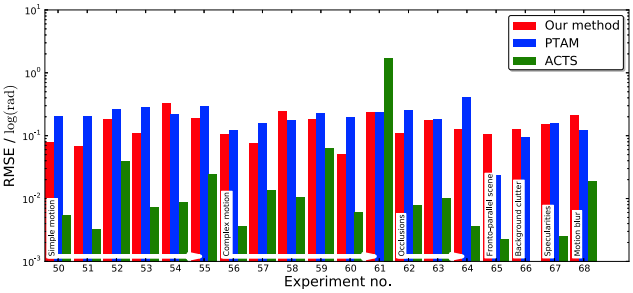
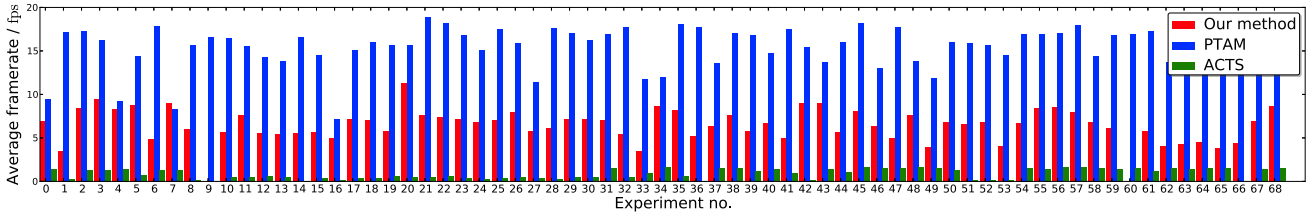



Figure 6. Accuracy on the synthetic part of the dataset (view in color): As it is to be expected, offline SFM by ACTS, serving as baseline for our comparison, outperforms our method and PTAM, which are generally on par with each other. Better outlier management, however, furnishes our method with robustness advantages, see Fig. 7(b).

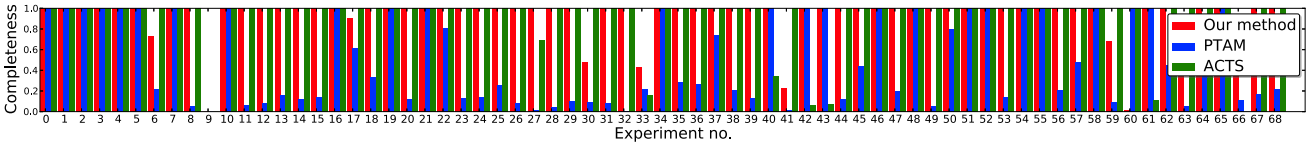
Fig. 6 presents a time aggregate of the error. It is not surprising that, in terms of accuracy, ACTS exhibits by far the best performance: First, offline operation allows for adopting a global view of each track and scrutinizing its utility for reconstruction. Second, correspondences arising from SIFT matching are expected to be significantly more reliable. Third, the underlying consensus-based optimization method has an up-to-now unchallenged break-down point. It is remarkable that ACTS fails on sequences that one would deem particularly simple, see row five of Fig. 8(d). Looking at Fig. 8(b), our method generates results that are just as visually pleasing, however, at a fraction of computation time, see Fig. 7(a). In comparison, where PTAM works stably, the maps it delivers are slightly more cluttered, see the second and fifth row of Fig. 8(c).

The root-mean-square error (RMSE) is somewhat misleading in the sense that normalization by time conceals the completeness of reconstruction. On this account, we manually recorded the time at which tracking failure occurs relative to total video length. The result is shown in Fig. 7(b) and reflects the robustness of each of the compared methods. Our method completed 85.5 % of all experiments successfully, *i.e.*, with a score of 1.0, PTAM 36.2 %, and ACTS 88.4 %.

The highest frame rates are achieved by PTAM, however, as Fig. 7(b) suggests, at the expense of robustness. A major advantage of the parallel tracking and mapping paradigm is the separability into two independent threads. The nature of (3) clearly prohibits a similar computational architecture. Our implementation achieves a certain degree of parallelism at the level of matrix multiplication but other than that, it is not as tuned towards performance as the latest very well-developed PTAM version is (*e.g.*, by exploiting graphics hardware and/or specialized processor instruction sets for parallelization). What matters in this context is the fact that both algorithms possess the same computational complexity.




(a) Efficiency: Average number of frames per second.




(b) Completeness: Time of tracking failure over total length of the image sequence.

Figure 7. Overview of the performance achieved on the entire dataset by the three investigated methods in comparison (view in color).

## 4. Conclusion

We have presented a computationally efficient numerical scheme to perform simultaneous reconstruction (map-building) and localization. The key to achieving fast and robust performance lies in the outlier management process during correspondence. Rather than resorting to combinatorial matching or random sampling techniques, we employ a simpler M-estimator, and design a simple topological validation test to reject points that slide on the surface of the object (*e.g.*, at occluding boundaries), and a photometric test based on a coarse contrast-invariant descriptor. Such descriptors are associated to each point in the map, and could support detection, localization and recognition of objects, which is our long-term goal. Indeed, one could interpret the localization and (implicit) inlier selection of our approach as a method for object detection and localization in a straightforward manner. However, in this manuscript we have not addressed the use of our maps for classification purposes, which is well beyond our scope here. Instead, we have shown that our approach fills an important vacant niche in the state of the art by providing a method to reconstruct models of small scale objects despite significant self-occlusions and a small effective field of view. Our method achieves a satisfactory compromise between computational complexity, simplicity, and robustness, as demonstrated experimentally.

## Acknowledgements

The authors gratefully acknowledge D. Davis, J. Dong, and J. Hernandez for their help in collecting data and running large-scale experiments. This research was supported by DARPA's program MSEE FA8650-11-1-7156 and AFOSR FA9550-12-1-0364.

## References

[1] P. Besl and H. McKay. A method for registration of 3-D shapes. *IEEE T. Pattern Anal.*, 14(2):239–256, 1992. 3

[2] M. Byröd and K. Åström. Conjugate gradient bundle adjustment. *Proc. ECCV*, 1:114–127, 2010. 5

[3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. *Proc. ECCV*, 1:778–792, Sept. 2010. 3

[4] F. Crow. Summed-area tables for texture mapping. *Proc. SIGGRAPH*, 18(3):207–212, Jan. 1984. 4

[5] D. Donoho. *Breakdown properties of multivariate location estimators*. Phd, Harvard University, 1982. 5

[6] C. Engels, H. Stewénius, and D. Nistér. Bundle Adjustment Rules. *Proc. PCV*, 1:266–271, 2006. 2

[7] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 3

[8] J. Funke and T. Pietzsch. A Framework For Evaluating Visual SLAM. *Proc. BMVC*, 1, 2009. 2, 5

[9] G. Graber, T. Pock, and H. Bischof. Online 3D reconstruction using convex optimization. *Proc. ICCV*, 1:708–711, 2011. 2

[10] J. Hedborg, P.-E. Forssén, and M. Felsberg. Fast and Accurate Structure and Motion Estimation. *Proc. ISVC*, 1:211–222, 2009. 2, 4, 5

[11] C. Hernandez, F. Perbet, M.-T. Pham, G. Vogiatzis, O. J. Woodford, A. Maki, B. Stenger, and R. Cipolla. Live 3D shape reconstruction, recognition and registration. *Proc. ICCV Workshops*, 1:729–729, 2011. 2

[12] P. Huber. *Robust Statistics*. Wiley, 2nd edition, 2004. 2, 3

[13] Y. Jeong, D. Nistér, D. Steedly, R. Szeliski, and I.-S. Kweon. Pushing the envelope of modern methods for bundle adjustment. *IEEE T. Pattern Anal.*, 34(8):1605–1617, 2012. 1

[14] H. Jin, P. Favaro, and S. Soatto. Real-time 3D motion and structure of point features: a front-end system for vision-based control and interaction. *Proc. CVPR*, 2:778–779, 2000. 2

[15] G. Klein and D. Murray. Parallel Tracking and Mapping for Small AR Workspaces. *Proc. ISMAR*, 1:225–234, 2007. 2, 3, 4, 5

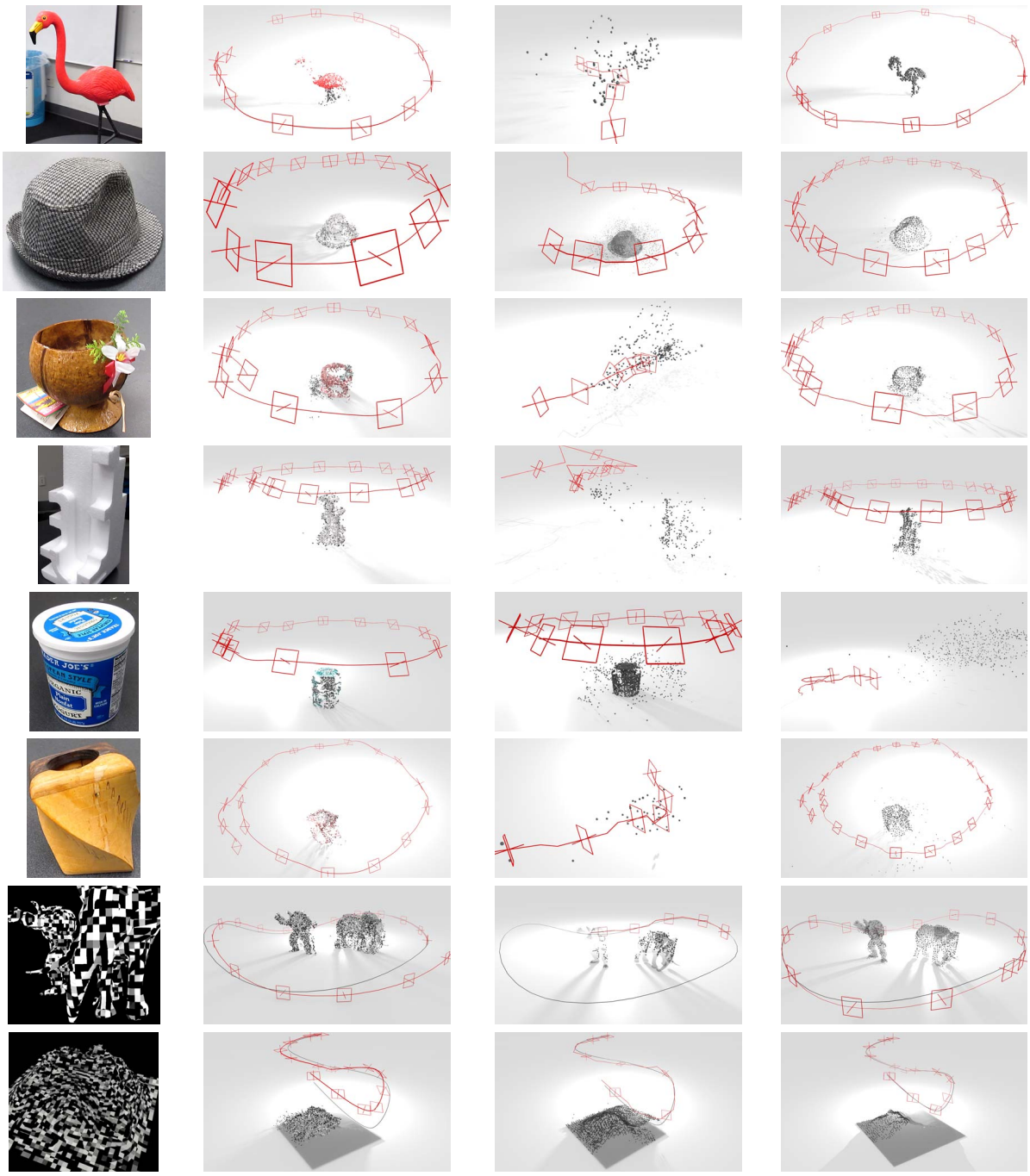
[16] E. Mair, K. H. Strobl, T. Bodenmüller, M. Suppa, and D. Burschka. Real-time Image-based Localization for Hand-held 3D-modeling. *KI*, 24(3):207–214, 2010. 2

[17] P. Moreels and P. Perona. Evaluation of Features Detectors and Descriptors based on 3D Objects. *Int. J. Comput. Vision*, 73(3):263–284, 2006. 2, 5

[18] R. a. Newcombe and A. J. Davison. Live dense reconstruction with a single moving camera. *Proc. CVPR*, 1:1498–1505, 2010. 2

[19] R. A. Newcombe, D. Molyneaux, D. Kim, P. Koli, A. J. Davison, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-Time Dense Surface Mapping and Tracking. *Proc. ISMAR*, 1:127–136, 2011. 1

[20] D. Nister. Preemptive RANSAC for live structure and motion estimation. *Proc. ICCV*, 1:199–206, 2003. 2



(a) Object (b) Our method (c) PTAM (d) ACTS

Figure 8. Selected reconstructions of structure and motion from video. The last two rows show results obtained on synthetic input data.


[21] Q. Pan, G. Reitmayr, E. Rosten, and T. Drummond. Rapid 3D modelling from live video. *Proc. MIPRO*, 1:252–257, 2010. 2

[22] E. Rosten, R. Porter, and T. Drummond. Faster and better: a machine learning approach to corner detection. *IEEE T. Pattern Anal.*, 32(1):105–119, 2010. 4

[23] S. Savarese and L. Fei-Fei. 3D generic object categorization, localization and pose estimation. *Proc. ICCV*, 1:1–8, 2007. 1

[24] H. Strasdat, J. Montiel, and A. J. Davison. Visual SLAM: Why filter? *Image. Vision. Comput.*, 30(2):65–77, 2012. 2

[25] J. Stühmer, S. G. And, and D. Cremers. Real-Time Dense Geometry from a Handheld Camera. *Lect. Notes Comput. Sc.*, 6376:11–20, 2010. 2

[26] A. Vedaldi, G. Guidi, and S. Soatto. Moving Forward in Structure From Motion. *Proc. CVPR*, 1:1–7, 2007. 5

[27] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof. Dense reconstruction on-the-fly. *Proc. CVPR*, 1:1450–1457, 2012. 2

[28] G. Zhang, X. Qin, W. Hua, T.-T. Wong, P.-A. Heng, and H. Bao. Robust Metric Reconstruction from Challenging Video Sequences. *Proc. CVPR*, 1:1–8, 2007. 2, 6