Abstract

Finding local features that are repeatable across multiple views is a cornerstone of sparse 3D reconstruction. The classical image matching paradigm detects keypoints per-image once and for all, which can yield poorly-localized features and propagate large errors to the final geometry. In this paper, we refine two key steps of structure-from-motion by a direct alignment of low-level image information from multiple views: we first adjust the initial keypoint locations prior to any geometric estimation, and subsequently refine points and camera poses as a post-processing. This refinement is robust to large detection noise and appearance changes, as it optimizes a featuremetric error based on dense features predicted by a neural network. This significantly improves the accuracy of camera poses and scene geometry for a wide range of keypoint detectors, challenging viewing conditions, and off-the-shelf deep features. Our system easily scales to large image collections, enabling pixel-perfect crowdsourced localization at scale. Our code is publicly available at github.com/cvg/pixel-perfect-sfm as an add-on to the popular SfM software COLMAP.

1. Introduction

Mapping the world is an important requirement for spatial intelligence applications in augmented reality or robotics. Tasks like visual localization or path planning can benefit from accurate sparse or dense 3D reconstructions of the environment. These can be built from images using Structure-from-Motion (SfM), which associates observations across views to estimate camera parameters and 3D scene geometry. Sparse reconstruction based on matching local image features [10, 20, 22, 32, 48, 54, 56, 62] is the most common due to its scalability and its robustness to appearance changes introduced by varying devices, viewpoints, and temporal conditions found in crowdsourced scenarios [2, 28, 33, 45, 47, 55].

SfM assumes that sparse interest points [10, 20, 22, 32, 48, 56, 59, 81, 88] can be reliably detected across views. It typically selects such points for each image independently and relies on these initial detections for the remainder of the reconstruction process. However, detecting keypoints from a single view is inherently inaccurate due to appearance changes and discrete image sampling [30]. The advent of convolutional neural network (CNNs) for detection has magnified this issue, as they generally do not retain local image information and instead favor global context.

Multi-view geometric optimization with bundle adjustment [4, 40, 79] is commonly used to refine cameras and points using reprojection errors. Dusmanu et al. [23] proposed to refine keypoint locations prior to SfM via an analogous geometric cost constrained with local optical flow. This can improve SfM, but has limited accuracy and scalability.

In this work, we argue that local image information is valuable throughout the SfM process to improve its accuracy. We adjust both keypoints and bundles, before and after reconstruction, by direct image alignment [18, 25, 49] in a learned feature space. Exploiting this locally-dense informa-
Figure 2: **Refinement pipeline.** Our refinement works on top of any SfM pipeline that is based on local features. We perform a two-stage adjustment of keypoints and bundles. The approach first refines the 2D keypoints only from tentative matches by optimizing a direct cost over dense feature maps. The second stage operates after SfM and refines 3D points and poses with a similar featuremetric cost.

Differently, dense matching [13, 46, 58, 71, 74, 78, 80] considers all pixels in each image, resulting in denser and more accurate correspondences. It has been successful for constrained settings like optical flow [38, 73] or stereo depth estimation [86], but is not suitable for large-scale SfM due to its high computational cost due to many redundant correspondences. Several recent works [44, 57, 75, 92] improve the matching efficiency by first matching coarsely and subsequently refining correspondences using a local search. This is however limited to image pairs and thus cannot create point tracks required by SfM.

Our work combines the best of both paradigms by leveraging dense local information to refine sparse observations. It is inherently amenable to SfM as it can optimize all locations over multiple views in a track simultaneously.

**Subpixel estimation** is a well-studied problem in correspondence search. Common approaches either upsample the input images or fit polynomials or Gaussian distributions to local image neighborhoods [27, 34, 37, 48, 66]. With the widespread interest in CNNs for local features, solutions tailored to 2D heatmaps have been recently developed, such as learning fine local sub-heatmaps [36] or estimating subpixel corrections with regression [14, 77] or the soft-argmax [52, 89]. Cleaner heatmaps can also arise from aggregating predictions over multiple virtual views using data augmentation [20].

Detections or local affine frames can be combined across multiple views with known poses in a least-squares geometric optimization [24, 79]. Dusmanu et al. [23] instead refine keypoints solely based on tentative matches, without assuming known geometry. This geometric formulation exhibits remarkable robustness, but is based on a local optical flow whose estimation for each correspondence is expensive and approximate. We unify both keypoint and bundle optimizations into a joint framework that optimizes a featuremetric cost, resulting in more accurate geometries and a more efficient keypoint refinement.

**Direct alignment** optimizes differences in pixel intensities by implicitly defining correspondences through the motion and geometry. It therefore does not suffer from geometric noise and is naturally subpixel accurate via image interpolation. Direct photometric optimization has been successfully applied to optical flow [8, 49], visual odometry [18, 25, 26, 42], SLAM [5, 69], multi-view stereo (MVS) [19, 21, 87], and pose refinement [70]. It generally fails for moderate displacements or appearances changes, and is thus not suitable for large-baseline SfM. One notable work by Woodford & Rosten [84] refines dense SfM+MVS models with a robust image normalization. It focuses on dense mapping with accurate initial poses and moderate appearance changes. Georgel et al. [29] instead estimate more accurate relative poses by elegantly combining photometric and geometric costs. They show that dense information can improve sparse estimation.

2. Related work

**Image matching** is at the core of SfM and visual SLAM, which typically rely on sparse local features for their efficiency and robustness. The process i) detects a small number of interest points, ii) computes their visual descriptors, iii) matches them with a nearest neighbor search, and iv) verifies the matches with two-view epipolar estimation and RANSAC. The correspondences then serve for relative or absolute pose estimation and 3D triangulation. As keypoints are sparse, small inaccuracies in their locations can result in large errors for the estimated geometric quantities.
but their approach ignores appearance changes. Differently, our work improves the entire SFM pipeline starting with tentative matches and addresses larger, challenging changes. To improve on the weaknesses of photometric optimization, numerous recent works align multi-dimensional image representations. Examples of this featuremetric optimization include frame tracking with handcrafted [6, 53] or learned descriptors [17, 50, 82, 83, 85], optical flow [7, 11], MVS [90], and dense SFM in small scenes [76]. Closer to our work, PixLoc [63] learns deep features with a large basin of convergence for wide-baseline pose refinement. It improves the accuracy of sparse matching but is designed for single images and disregards the scalability to multiple images or large scenes. Here we extend this paradigm to other steps of SFM and propose an efficient algorithm that scales to thousands of images. We show that learning task-specific wide-context features is not necessary and demonstrate highly accurate refinements with off-the-shelf features.

In conclusion, our work is the first to apply robust featuremetric optimization to a large-scale sparse reconstruction problem and show significant benefits for visual localization.

3. Background

Given \( N \) images \( \{ I_i \} \) observing a scene, we are interested in accurately estimating its 3D structure, represented as sparse points \( \{ P_j \in \mathbb{R}^3 \} \), intrinsic parameters \( \{ C_i \} \) of the cameras, and the poses \( \{(R_i, t_i) \in SE(3)\} \) of the images, represented as rotation matrices and translation vectors.

A typical SFM pipeline performs geometric estimation from correspondences between sparse 2D keypoints \( \{ p_u \} \) observing the same 3D point from different views, collectively called a track. Association between observations is based on matching local image descriptors \( \{ d_u \in \mathbb{R}^D \} \), but the estimated geometry relies solely on the location of the keypoints, whose accuracy is thus critical. Keypoints are detected from local image information for each image individually, without considering multiple views simultaneously. Subsequent steps of the pipeline discover additional information about the scene, such as its geometry or its multi-view appearance. Two approaches leverage this information to reduce the noise and refine the keypoints.

Global refinement: Bundle adjustment [79] is the gold standard for refining structure and poses given initial estimates. It minimizes the total geometric error

\[
E_{BA} = \sum_j \sum_{(u,v) \in \mathcal{T}(j)} \| \Pi(R_jp_j + t_i, C_i) - p_u \|_\gamma, \tag{1}
\]

where \( \mathcal{T}(j) \) is the set the images and keypoints in track \( j \), \( \Pi(\cdot) \) projects to the image plane, and \( \| \cdot \|_\gamma \) is a robust norm [31]. This formulation implicitly refines the keypoints while ensuring their geometric consistency. It however ignores the uncertainty of the initial detections and thus requires many observations to reduce the geometric noise. Operating on an existing reconstruction, it cannot recover observations arising from noisy keypoints that are matched correctly but discarded by the geometric verification.

Track refinement: To improve the accuracy of the keypoints prior to any geometric 3D estimation, Dusmanu et al. [23] optimize their locations over tentative tracks formed by raw, unverified matches. They exploit the inherent structure of the matching graph to discard incorrect matches without relying on geometric constraints. Given two-view dense flow fields \( \{ T_{v \rightarrow u} \} \) between the neighborhoods of matching keypoints \( u \) and \( v \), this keypoint adjustment optimizes, for each tentative track \( j \), the multi-view cost

\[
E_{KA}^j = \sum_{(u,v) \in \mathcal{M}(j)} \| p_v + T_{v \rightarrow u}[p_v] - p_u \|_\gamma, \tag{2}
\]

where \( \mathcal{M}(i) \) denotes the set of matches that forms the track and \( \| \cdot \|_\gamma \) is a lookup with subpixel interpolation. A deep neural network is trained to regress the flow of a single point from two input patches and the flow field is interpolated from a sparse grid. This dramatically improves the keypoint accuracy, but some errors remain as the regression and interpolation are only approximate.

Both bundle and keypoint adjustments are based on geometric observations, namely keypoint locations and flow, but do not account for their respective uncertainties. They thus require a large number of observations to average out the geometric noise and their accuracy is in practice limited.

4. Approach

Summarizing dense image information into sparse points is necessary to perform global data association and optimization at scale. However, refining geometry is an inherently local operation, which, we show, can efficiently benefit from locally-dense pixels. Given constraints provided by coarse but global correspondences or initial 3D geometry, the dense information only needs to be locally accurate and invariant but not globally discriminative. While SFM typically discards image information as early as possible, we instead exploit it in several steps of the process thanks to direct alignment. Leveraging the power of deep features, this translates into featuremetric keypoint and bundle adjustments that elegantly integrate into any SFM pipeline by replacing their geometric counterparts. Figure 2 shows an overview.

We first introduce the featuremetric optimization in Section 4.1. We then describe our formulations of keypoint adjustment, in Section 4.2, and bundle adjustment, in Section 4.3, and analyze their efficiency.

4.1. Featuremetric optimization

Direct alignment: We consider the error between image intensities at two sparse observations: \( r = I_i[p_u] - I_j[p_v] \).
Local image derivatives implicitly define a flow from one point to the other through a gradient descent update:

$$T_{v \to u}[p_v] \propto -\frac{\partial I_j}{\partial p}[p_v] \top r .$$  \hspace{1cm} (3)

This flow can be efficiently computed at any location in a neighborhood around $r$, without approximate interpolation nor descriptor matching. It naturally emerges from the direct optimization of the photometric error, which can be minimized with second-order methods in the same way as the aforementioned geometric costs. Unlike the flow regressed from a black-box neural network [23], this flow can be made consistent across multiple views by jointly optimizing the cost over all pairs of observations.

**Learned representation:** SfM can handle image collections with unconstrained viewing conditions exhibiting large changes in terms of illumination, resolution, or camera models. The image representation used should be robust to such changes and ensure an accurate refinement in any condition. We thus turn to features computed by deep CNNs, which we refine the keypoint locations before geometrically verifying.

**Connected components in the match:** Track separation:

Once local features are detected, described, and matched, we refine the keypoint locations before geometrically verifying the tentative matches.

**Track separation:** Connected components in the matching graph define tentative tracks – sets of keypoints that are likely to observe the same 3D point, but whose observations have not yet been geometrically verified. Because a 3D point has a single projection on a given image plane, valid tracks cannot contain multiple keypoints detected in the same image. We can leverage this property to efficiently prune out most incorrect matches using the track separation algorithm introduced in [23]. This speeds up the subsequent optimization and reduces the noise in the estimation.

**Objective:** We then adjust the locations of 2D keypoints belonging to the same track $j$ by optimizing its featuremetric consistency along tentative matches with the cost

$$E_{\text{FBA}}^j = \sum_{(u,v) \in \mathcal{M}(j)} w_{uv} \left\| F_{i(u)}[p_u] - F_{i(v)}[p_v] \right\|_\gamma ,$$  \hspace{1cm} (4)

where $w_{uv}$ is the confidence of the correspondence $(u,v)$, such as the similarity of its local feature descriptors $d_u \cdot d_v$.

This allows the optimization to split tracks connected by weak correspondences, providing robustness to mismatches. The confidence is not based on the dense features since these are not expected to disambiguate correspondences at the global image level.

**Efficiency:** This direct formulation simply compares pre-computed features on sparse points and is thus much more scalable than patch flow regression (Eq. 2), which performs a dense local correlation for each correspondence. All tracks are optimized independently, which is very fast in practice despite the sheer number of tentative matches.

**Drift:** Because of the lack of geometric constraints, the points are free to move anywhere on the underlying 3D surface of the scene. The featuremetric cost biases the updates towards areas with low spatial feature gradients and with better-defined features. This can result in a large drift if not accounted for. Keypoints should however remain repeatable w.r.t. unrefined detections to ensure the matchability of new images, such as for visual localization. It is thus critical to limit the drift, while allowing the refinement of noisier keypoints. For each track, we freeze the location of the keypoint $u$ with highest connectivity, as in [23], and constrain the location $p_v$ of each keypoint w.r.t. to its initial detection $p^0_v$, such that $\left\| p_v - p^0_v \right\| \leq K$.

Once all tracks are refined, the geometric estimation proceeds, typically using two-view epipolar geometric verification followed by incremental or global SfM.

**4.3. Bundle adjustment**

The estimated structure and motion can then be refined with a similar featuremetric cost. Here keypoints are implicitly defined by the projections of the 3D points into the 2D image planes, and only poses and 3D points are optimized.

**Objective** We minimize for each track $j$ the error between its observations and a reference appearance $f^j$:

$$E_{\text{FBA}} = \sum_j \sum_{(i,u) \in T(j)} \left\| F_i \left[ R_j P_j + t_i, C_i \right] - f^j \right\|_\gamma .$$  \hspace{1cm} (5)

The reference is selected at the beginning of the optimization and kept fixed from then on. This reduces the drift of the points significantly, as also noted in [5], but is more flexible than the common ray-based parametrization [25,42,84].

The reference is defined as the observation closest to the robust mean $\mu$ over all initial observations $f^j_{\text{u}}$ of the track:

$$f^j = \arg\min_{r \in \{ f^j_{\text{u}} \} } \left\| \mu - f \right\| ,$$  \hspace{1cm} (6)

with $\mu = \arg\min_{\mu \in \mathbb{R}^p} \sum_{r \in \{ f^j_{\text{u}} \} } \left\| f - \mu \right\|_\gamma .$  \hspace{1cm} (7)

This ensures robustness to outlier observations and accounts for the unknown topology of the feature space.
We form a reduced camera system based on the Schur complement. These can then be combined into a multi-level optimization scheme \cite{63, 63} that sequentially refines based on coarse intermediate maps. For 3D sparse triangulation, we use the ETH3D benchmark \cite{70}.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
\multicolumn{2}{|c|}{SfM features} & \multicolumn{3}{c|}{ETH3D indoor} & \multicolumn{3}{c|}{ETH3D outdoor} & \multicolumn{3}{c|}{ETH3D outdoor} \\
\hline
\multicolumn{2}{|c|}{\textless \textgreater} & 1cm & 2cm & 5cm & 1cm & 2cm & 5cm & 1cm & 2cm & 5cm \\
\hline
\textgreater SIFT \textlt \cite{48} & 75.62 & 85.04 & 92.45 & 0.21 & 0.87 & 3.61 & 57.64 & 71.92 & 85.23 & 0.06 & 0.34 & 2.45 \\
\hline
\textgreater Patch Flow & 80.99 & 89.06 & 95.06 & 0.24 & 0.97 & 3.88 & 64.79 & 78.90 & 90.04 & 0.08 & 0.41 & 2.76 \\
\hline
\textgreater ours & 89.92 & 89.77 & 94.77 & 0.25 & 0.96 & 3.75 & 68.43 & 80.73 & 91.28 & 0.08 & 0.42 & 2.75 \\
\hline
\hline
\textgreater SuperPoint \textlt \cite{20} & 75.76 & 85.61 & 93.38 & 0.59 & 2.21 & 8.89 & 50.45 & 65.07 & 80.26 & 0.10 & 0.55 & 3.92 \\
\hline
\textgreater Patch Flow & 85.77 & 91.57 & 95.85 & 0.72 & 2.51 & 9.59 & 64.94 & 77.65 & 88.86 & 0.15 & 0.77 & 4.93 \\
\hline
\textgreater ours & 89.33 & 93.58 & 96.58 & 0.74 & 2.53 & 9.51 & 71.27 & 82.58 & 92.08 & 0.16 & 0.83 & 5.06 \\
\hline
\hline
\textgreater D2-Net \textlt \cite{22} & 47.18 & 64.94 & 83.37 & 0.47 & 1.87 & 7.07 & 20.87 & 34.55 & 56.53 & 0.03 & 0.19 & 1.78 \\
\hline
\textgreater Patch Flow & 79.10 & 86.64 & 93.26 & 1.45 & 4.53 & 12.95 & 57.34 & 70.71 & 84.12 & 0.21 & 1.06 & 6.02 \\
\hline
\textgreater ours & 82.49 & 88.83 & 94.35 & 1.36 & 4.13 & 11.80 & 65.71 & 77.95 & 89.22 & 0.21 & 1.01 & 5.63 \\
\hline
\hline
\textgreater R2D2 \textlt \cite{56} & 66.30 & 79.21 & 90.00 & 0.53 & 2.06 & 8.62 & 49.32 & 66.10 & 83.10 & 0.11 & 0.55 & 3.63 \\
\hline
\textgreater Patch Flow & 77.94 & 85.82 & 92.48 & 0.66 & 2.32 & 9.07 & 64.14 & 78.10 & 90.18 & 0.16 & 0.71 & 4.09 \\
\hline
\textgreater ours & 80.67 & 87.61 & 93.42 & 0.67 & 2.31 & 8.95 & 67.77 & 80.85 & 91.91 & 0.16 & 0.73 & 4.09 \\
\hline
\end{tabular}
\caption{3D sparse triangulation. Our refinement yields significantly more accurate and complete point clouds than the common geometric SfM pipeline. It is more effective than the existing Patch Flow \cite{23}, especially at 1cm or with SIFT.}
\end{table}

**Efficiency**: Compared to the keypoint adjustment (Eq. 4), using a reference feature reduces the number of residuals from $O(N^2)$ to $O(N)$. On the other hand, all tracks need to be updated simultaneously because of the interdependency caused by the camera poses. To accelerate the convergence, we form a reduced camera system based on the Schur complement and use embedded point iterations \cite{40}. The refinement generally converges within a few camera updates.

### 4.4. Implementation

**Dense extractor**: Our refinement can work with any off-the-shelf CNN that produces feature maps that are locally discriminative. These should be of the same resolution as the input (stride 1) to enable subpixel accuracy. The radius of convergence, or context, of such features depends on the amount of noise in the keypoints. Most detectors like SIFT have at most a few pixels of error, while others like D2-Net exhibit a much larger detection noise. In our experiments, we use S2DNet \cite{30} for dense feature extraction, as it computes fine features very efficiently in only 4 convolutions, but also required, deeper features with a larger context. These can then be combined into a multi-level optimization scheme \cite{25, 63, 82} that sequentially refines based on coarse to fine features. The convergence can thus be adjusted depending on the detector and on the image resolution. We show in Section 5.4 that other dense features work well too.

**Optimization**: The optimization problems of both keypoint and bundle adjustments are solved with the Levenberg-Marquardt \cite{43} algorithm implemented using Ceres \cite{3}. Feature maps are stored as collections of $16 \times 16$ patches centered around the initial keypoint detections. We thus constrain points to move at most $K=8$ pixels. The feature lookup is implemented as bicubic interpolation. We use the Cauchy loss $\gamma$ with a scale of 0.25. The robust mean in Eq. 7 is computed with iteratively reweighted least squares \cite{35}.

Simultaneously storing all high-dimensional feature patches incurs high memory requirements during BA. We dramatically increase its efficiency by exhaustively precomputing patches of feature distances and directly interpolate an approximate cost $E_{ij} = \| \mathbf{F}_i - \mathbf{F}_j \| \| \mathbf{p}_{ij} \|$. To improve the convergence, we store and optimize its spatial derivatives $\partial E_{ij}/\partial \mathbf{p}_{ij}$. This reduces the residual size from $D$ to 3 with no loss of accuracy. See Supplemental C for more details.

**Run time and memory**: S2DNet can extract 3-5 dense feature maps per second and both featuremetric adjustments run in less than 5 minutes for 100 images. As these features are 128-dimensional, the memory consumption can be a bottleneck. We believe that much fewer dimensions are actually required for refinement, and retraining a compact feature extractor would improve the efficiency of the optimization.

### 5. Experiments

We evaluate our featuremetric refinement on various SfM tasks with several handcrafted and learned local features and show substantial improvements for all of them. We first evaluate its accuracy on the tasks of triangulation and camera pose estimation in Sections 5.1 and 5.2, respectively. We then assess in Section 5.3 the impact of the refinement on two-view and multi-view pose estimation for end-to-end reconstruction in challenging conditions. Lastly, Section 5.4 analyzes the validity and scalability of our design decisions through an ablation study.

#### 5.1. 3D triangulation

We first evaluate the accuracy of the refined 3D structure given known camera poses and intrinsics. **Evaluation**: We use the ETH3D benchmark \cite{70}, which
is composed of 13 indoor and outdoor scenes and provides images with millimeter-accurate camera poses and highly-accurate ground truth dense reconstructions obtained with a laser scanner. We follow the protocol introduced in [23], in which a sparse 3D model is triangulated for each scene using COLMAP [67] with fixed camera poses and intrinsics. Following the original benchmark setup, we report the accuracy and completeness of the reconstruction, in %, as the ratio of triangulated and ground-truth dense points that are within a given distance of each other.

**Baselines:** We evaluate our featuremetric refinement with the hand-crafted local features SIFT [48] and the learned ones SuperPoint [20], D2-Net [22], and R2D2 [56], using the associated publicly available code repositories. We compare our approach to the geometric optimization of [23], referred here as Patch Flow. We re-compute the numbers provided in the original paper using the code provided by the authors.

**Results:** Table 1 shows that our approach results in significantly more accurate and complete 3D reconstructions compared to the traditional geometric SfM. It is more accurate than Patch Flow, especially at the strict threshold of 1cm, and exhibits similar completeness. The improvements are consistent across all local features, both indoors and outdoors. The gap with Patch Flow is especially large for SIFT, which already detects well-localized keypoints. This confirms that our featuremetric optimization better captures low-level image information and yields a finer alignment. Patch Flow is more complete for larger thresholds as it partly solves a different problem by increasing the keypoint repeatability with its large receptive field, while we focus on their localization.

**5.2. Camera pose estimation**

We now evaluate the impact of our refinement on the task of camera pose estimation from a single image.

**Evaluation:** We again follow the setup of [23] based on the ETH3D benchmark. For each scene, 10 images are randomly selected as queries. For each of them, the remaining images, excluding the 2 most covisible ones, are used to triangulate a sparse 3D partial model. Each query is then matched against its corresponding partial model and the resulting 2D-3D matches serve to estimate its absolute pose using LO-RANSAC+PnP [15] followed by geometric refinement. We compare the 130 estimated query poses to their ground truth and report the area under the cumulative translation error curve (AUC) up to 1mm, 1cm, and 10cm.

**Baselines:** Patch Flow performs multi-view optimization over each partial model independently as well as over the matches between each query and its partial model. Similarly, we first refine each partial model as in Section 5.1. We then adjust the query keypoints using its tentative matches, estimate an initial pose, and refine it with featuremetric BA.

**Results:** The AUC and its cumulative plot are shown in Table 2. Our refinement substantially improves the localization accuracy for all local features, including SIFT, for thresholds, featuremetric optimization is consistently more accurate and complete 3D reconstructions.

**5.3. End-to-end Structure-from-Motion**

While the previous experiments precisely quantify the accuracy of the refinement, they do not contain any variations

<table>
<thead>
<tr>
<th>Task 1: Stereo</th>
<th>Task 2: Multiview</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SfM features</strong></td>
<td><strong>AUC@K°</strong></td>
</tr>
<tr>
<td><strong>Refinement</strong></td>
<td></td>
</tr>
<tr>
<td>1mm</td>
<td>1cm</td>
</tr>
<tr>
<td>SfM features</td>
<td></td>
</tr>
<tr>
<td>SuperPoint</td>
<td>58.78</td>
</tr>
<tr>
<td>ours</td>
<td>58.78</td>
</tr>
<tr>
<td>SIFT</td>
<td>38.09</td>
</tr>
<tr>
<td>ours</td>
<td>40.59</td>
</tr>
<tr>
<td>D2-Net (4k)</td>
<td>16.83</td>
</tr>
<tr>
<td>ours</td>
<td>25.89</td>
</tr>
</tbody>
</table>

**Table 3:** The proposed refinement improves the accuracy of poses estimated by epipolar geometry (stereo) or a complete SfM pipeline (multiview) with crowd-sourced imagery. Improvements are substantial for both standard (SIFT) and recent (SuperGlue) matching configurations, especially when few images N observe the scene.
of appearance or camera models. We thus turn to crowd-sourced imagery and evaluate the benefits of our featuremetric optimization in an end-to-end reconstruction pipeline.

**Evaluation:** We use the data, protocol, and code of the 2020 Image Matching Challenge [1, 41]. It is based on large collections of crowd-sourced images depicting popular landmarks around the world. Pseudo ground truth poses are obtained with SfM [67] and used for two tasks. The stereo task evaluates relative poses estimated from image pairs by decomposing their epipolar geometry. This is a critical step of global SfM as it initializes its global optimization. The multiview task runs incremental SfM for small subsets of images, making the SfM problem much harder, and evaluates the final relative poses within each subset. For each task, we report the AUC of the pose error at the threshold of 5°, where the pose error is the maximum of the angular errors in rotation and translation. As the evaluation server accepts at most correspondences, we cannot evaluate our method using the test data. We instead test on a subset of the publicly available validation scenes, and tune the RANSAC and matching parameters on the remaining scenes. More details on this setup are provided in the Supplemental.

**Baselines:** We evaluate our refinement in combination with SIFT [48], D2-Net [22], and SuperPoint+SuperGlue [20, 62]. We limit the number of detected keypoints to 2k for computational reasons, but increase this number to 4k for D2-Net as it otherwise performs poorly. In the stereo task, we adjust the keypoints using the entire exhaustive tentative match graph (4950 pairs per scene). We use LO-DEGENSAC [15, 16] for match verification, the ratio test for SIFT, and the mutual check for SIFT and D2-Net. In the multiview task, we adjust keypoints for each subset independently, considering only the matches between images in the subset, and run our bundle adjustment after SfM.

**Results:** Table 3 summarizes the results. For stereo, our featuremetric keypoint adjustment significantly improves the accuracy of the two-view epipolar geometries across all local features and despite the challenging conditions. In multiview setting, it also improves the accuracy of the SfM poses, especially for small sets of images. Featuremetric optimization is particularly effective in this situation, as geometric optimization cannot fully suppress the detection noise due to the small number of observations. We visualize tracks of a 5-image reconstruction in Figure 4 and highlight the accuracy of the refined SfM model.

### 5.4. Additional insights

**Ablation study:** Table 4 shows the performance of several variants of our featuremetric optimization on ETH3D in terms of triangulation (scene Facade only) and localization (all scenes). We compare both types of adjustments, minor tweaks, and different image representations, including NCC-normalized intensity patches with fronto-parallel warping. Our final configuration, based on the dense features of S2DNet [30], performs best across all metrics. We will now show that it is also fairly efficient.

<table>
<thead>
<tr>
<th>SuperPoint</th>
<th>Acc. (%)</th>
<th>Compl. (%)</th>
<th>track length</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>KA vs. BA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>unrefined</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patch Flow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F-KA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F-BA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F-KA+BA (full)</td>
<td>46.46</td>
<td>65.41</td>
<td>0.19</td>
<td>1.14</td>
</tr>
<tr>
<td>w/ F-BA drift</td>
<td>47.93</td>
<td>66.52</td>
<td>0.20</td>
<td>1.17</td>
</tr>
<tr>
<td>Patch Flow + F-BA</td>
<td>46.30</td>
<td>65.22</td>
<td>0.19</td>
<td>1.13</td>
</tr>
<tr>
<td>higher resolution</td>
<td>47.67</td>
<td>65.39</td>
<td>0.21</td>
<td>1.21</td>
</tr>
<tr>
<td>dense feats</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>photometric BA [84]</td>
<td>28.43</td>
<td>45.87</td>
<td>0.11</td>
<td>0.72</td>
</tr>
<tr>
<td>VGG-16 ImageNet</td>
<td>36.86</td>
<td>54.99</td>
<td>0.15</td>
<td>0.90</td>
</tr>
<tr>
<td>DSIFT [46]</td>
<td>38.78</td>
<td>56.46</td>
<td>0.16</td>
<td>0.96</td>
</tr>
<tr>
<td>PixLoc [63]</td>
<td>29.49</td>
<td>46.60</td>
<td>0.12</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Table 4: Ablation study on ETH3D. i) Featuremetric keypoint and bundle adjustments (KA and BA) both largely improve the triangulation and localization accuracy. Patch Flow produces a longer track length because of its larger receptive field but is less accurate. ii) Letting the BA drift by updating reference features or increasing the image resolution both improve the triangulation, at the expense of poorer localization and increased run time, respectively. iii) Different image representations are better than the unrefined detections but S2DNet (our default) works best.

**Figure 3: Run-times.** We show the duration, in logarithmic scale, of the refinement for varying numbers of images. Our refinement is more than ten times faster than Patch Flow [23], whose run-time is dominated by the computation of the pairwise flow, which scales quadratically. Thanks to our precomputed cost patches, the featuremetric BA is fast. The KA amounts for the majority of the refinement time.
Figure 4: **Refined SfM tracks.** We show patches centered around reprojections of 3x 3D points observed in 4 images of the **St. Peter’s Square** scene. Deep features and their correlation maps with a reference are robust to scale or illumination changes, yet preserve local details required for fine alignment. Points refined with our approach (in **green**) are consistent across multiple views while those of a standard SfM pipeline (in **red**) are misaligned because the initial keypoint detections (in **blue**) are noisy.

**Scalability:** We run SfM on subsets of images of the Aachen Day-Night dataset [64, 65, 91]. Figure 3 shows the run times of the refinement for subsets of 10, 100 and 1000 images. The featuremetric refinement is an order of magnitude faster than Patch-Flow [23]. Precomputing distance maps reduces the peak memory requirement of the bundle adjustment from 80 GB to less than 10 GB for 1000 images. As storing feature maps only requires 50 GB of disk space, this refinement can easily run on a desktop PC. We thus refined the entire Aachen Day-Night v1.1 model, composed of 7k images, in less than 2 hours. Scene partitioning [67] could further reduce the peak memory. See Supplemental D for more details.

6. **Conclusion**

In this paper we argue that the recipe for accurate large-scale Structure-from-Motion is to perform an initial coarse estimation using sparse local features, which are by necessity globally-discriminative, followed by a refinement using locally-accurate dense features. Since the dense feature only need to be locally-discriminative, they can afford to capture much lower-level texture, leading to more accurate correspondences. Through extensive experiments we show that this results in more accurate camera poses and structure; in challenging conditions and for different local features.

While we optimize against dense feature maps, we keep the sparse scene representation of SfM. This ensures not only that the approach is scalable but also that the resulting 3D model is compatible with downstream applications, e.g. mapping for visual localization. Since our refinement works well even with few observations, as it does not need to average out the keypoint detection noise, it has the potential to achieve more accurate results using fewer images.

We thus believe that our approach can have a large impact in the localization community as it can improve the accuracy of the ground truth poses of standard benchmark datasets, of which many are currently saturated. Since this refinement is less sensitive to under-sampling, it enables benchmarking for crowd-sourced scenarios beyond densely-photographed tourism landmarks.

**Acknowledgements:** The authors thank Mihai Dusmanu, Rémi Pautrat, Marcel Geppert, and the anonymous reviewers for their thoughtful comments. Paul-Edouard Sarlin was supported by gift funding from Huawei, and Viktor Larsson by an ETH Zurich Postdoctoral Fellowship.
References


[37] Andres Huertas and Gerard Medioni. Detection of intensity changes with subpixel accuracy using laplacian-gaussian masks. TPAMI, 1986. 2


[46] Liu Liu, Hongdong Li, and Yuchao Dai. Efficient global 2D-3D matching for camera localization in a large-scale 3D map. In ICCV, 2017. 1


[48] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In IJCAI, 1981. 1, 2


[52] Seonwook Park, Thomas Schöps, and Marc Pollefeys. Illumination Change Robustness in Direct Visual SLAM. In ICRA, 2017. 3


[54] Filip Radenovic, Johannes L Schönberger, Dinghuang Ji, Jan-Michael Frahm, Ondrej Chum, and Jiri Matas. From dusk till dawn: Modeling in the dark. In CVPR, 2016. 1


[68] Thomas Schöps, Torsten Sattler, and Marc Pollefeys. BAD SLAM: Bundle Adjusted Direct RGB-D SLAM. In CVPR, June 2019. 2


[78] Engin Tola, Vincent Lepetit, and Pascal Fua. DAISY: An efficient dense descriptor applied to wide-baseline stereo. TPAMI, 2009. 2
[86] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. MVSNet: Depth inference for unstructured multi-view stereo. ECCV, 2018. 2
[88] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. LIFT: Learned invariant feature transform. In ECCV, 2016. 1