



Pareto Meets Huber: Efficiently Avoiding Poor Minima in Robust Estimation

Christopher Zach, Guillaume Bourmaud

► To cite this version:

Christopher Zach, Guillaume Bourmaud. Pareto Meets Huber: Efficiently Avoiding Poor Minima in Robust Estimation. ICCV, Oct 2019, Seoul, South Korea. hal-02353424

HAL Id: hal-02353424

<https://hal.science/hal-02353424v1>

Submitted on 7 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Pareto Meets Huber: Efficiently Avoiding Poor Minima in Robust Estimation

Christopher Zach

Chalmers University of Technology, Sweden

christopher.m.zach@gmail.com

Guillaume Bourmaud

University of Bordeaux, France

guillaume.Bourmaud@u-bordeaux.fr

Abstract

Robust cost optimization is the task of fitting parameters to data points containing outliers. In particular, we focus on large-scale computer vision problems, such as bundle adjustment, where Non-Linear Least Square (NLLS) solvers are the current workhorse. In this context, NLLS-based state of the art algorithms have been designed either to quickly improve the target objective and find a local minimum close to the initial value of the parameters, or to have a strong ability to avoid poor local minima. In this paper, we propose a novel algorithm relying on multi-objective optimization which allows to match those two properties. We experimentally demonstrate that our algorithm has an ability to avoid poor local minima that is on par with the best performing algorithms with a faster decrease of the target objective.

1. Introduction

Many computer vision problems can be stated as finding the parameters of a generative model that explain best the observed data points. The assumption, that the observed data points are functions of unknown parameters corrupted by known Gaussian noise, leads to non-linear least-squares (NLLS) minimization tasks to obtain a maximum-likelihood estimate of the unknowns. One advantage of NLLS problems is that minimizers can be found efficiently by leveraging second-order methods. In practice not all observations are just corrupted by non-problematic Gaussian noise, but an unknown fraction of data points may be proper outliers subject to an arbitrary (but independent) noise process. In order to cope with such outlier data points, *robust estimation* (a.k.a. *m-estimation* [18]) replaces the squared residuals in the NLLS problem with terms based on so-called *robust kernels* (or *robust functions*), which reduces the impact of large residuals. When the amount of outliers is large compared to the number of inliers, as it is the case for difficult matching scenarios, it is necessary to use a non-convex kernel, such as Tukey’s Biweight, to significantly reduce the influence of outliers. However, summing

such non-convex functions results in an overall highly non-convex cost function with many local minima. As a consequence, standard approaches such as Iterated Reweighted Least Squares (IRLS), which performs well with convex kernels (e.g. Huber kernel), easily get trapped in poor local minima for non-convex robust kernels. In order to avoid those poor local minima, state of the art approaches either smooth or alternatively “lift” the cost function before applying an NLLS solver. Thus, both types of methods effectively modify or distort the target objective to some extent.

In the specific context, where the practitioner seeks a compromise between efficiency and accuracy, as e.g. in real-time applications, it is also important to select an algorithm that quickly decreases the target objective. Such behavior allows the practitioner to stop the optimization process at any time (e.g. when a time budget is exhausted) with the guarantee that the initial cost was significantly reduced. Smoothing approaches do not possess this feature while, as it experimentally shown in the paper, lifting-based approaches sometimes have troubles avoiding poor minima. To overcome these limitations, we propose a novel NLLS-based algorithm that is inspired by Multi-Objective Optimization (MOO). To the best of our knowledge, this is the first robust estimation algorithm that both quickly decreases the target objective and possesses a strong ability to avoid poor local minima. More precisely, our contributions are:

1. we identify a major element of the success of lifting-based approaches and are therefore able to easily construct failure cases,
2. we propose to use an MOO approach to obtain an algorithm able of both avoiding poor local minima and quickly decreasing the target objective,
3. and we derive an efficient Levenberg-Marquardt-MOO method yielding cooperative minimization steps.

The rest of the paper is organized as follows: Sec. 2 discusses the related work while Sec. 3 introduces important definitions and notations regarding robust estimation. Our contributions are detailed in Sec. 4 and Sec. 5. Experimental results and a conclusion are provided in Sec. 6 and 7, respectively.

	IRLS[15], Triggs [28], $\sqrt{\psi}$ [9]	HQ [30], k -HQ [31]	GOM [4]	GOM+ [32]	LM-MOO (ours)
Quickly decreases target cost*					
Avoids poor local minima*					
Never ignores target cost	✓	×	×	×	✓
No extra variables	✓	×	✓	✓	✓

Table 1: State of the art NLLS-based robust estimation algorithms and their corresponding properties. (*) These rankings are observed experimentally on several computer vision problems.

2. Literature review

We discuss the literature related to robust estimation below, but mostly limit the exposition to methods utilizing a second-order NLLS solver¹. In this context, the current workhorse for robust cost estimation is the IRLS algorithm [15]. It directly tries to minimize the target cost function by iteratively fitting a (majorizing) quadratic model to the underlying robust kernel and applying a 2nd-order solver to the resulting NLLS problem. Several variants of this approach have been proposed, *e.g.* [28, 9], that essentially utilize different quadratic models. While these approaches usually converge quickly and have the desirable property of optimizing the target cost function at each step, they lack a mechanism to avoid poor local minima, and therefore easily get stuck in a poor solution.

One way of avoiding poor local minima is to “lift” the target cost function into a higher dimensional space by introducing so called “lifting” variables and apply an NLLS solver to the resulting objective. The construction is based on “half-quadratic” (HQ) minimization [12, 13, 2, 3], and it was experimentally shown that joint optimization of this half-quadratic cost over all unknowns (original and lifting ones) reaches significantly better local minima than IRLS at only a slight increase of run-time [30, 31]. Nevertheless, as it is shown in sec. 4, the ability of HQ-based approaches to avoid poor local minima relies on some assumptions that can be violated in computer vision applications.

Another way of avoiding poor local minima consists in building surrogate cost functions that have fewer local minima than the target cost function. Most of these methods fall under the umbrella term of Graduated Optimization Methods (GOM), but come in a large number of flavors (*e.g.* [20, 4, 26, 29, 8, 24, 23, 25, 32]). GOM-based approaches sequentially optimize a smoothed surrogate cost function (starting from a highly smoothed version of the target cost function), using *e.g.* IRLS, and use its minimizer

as initialization to optimize the next, less-smoothed surrogate cost function. This is done until the algorithm reaches the final optimization problem, *i.e.* the target cost function. In practice, GOM-based algorithms are computationally expensive, since they solve a sequence of optimization problems, but they exhibit a strong ability to reach good local minima. Early stopping can be applied to accelerate the graduated optimization methods (*e.g.* [32]), however even in this case, the ability of the algorithm to quickly decrease the target cost is limited.

In this paper, we propose an approach inspired by MOO, that combines the advantages of both IRLS and GOM-based approaches, *i.e.* it seeks to decrease the target cost at each step of the algorithm while maintaining a strong ability to avoid poor local minima. Relevant properties of the aforementioned methods are summarized in Table 1.

Close in spirit to our work are *multi-objectivization* methods [21, 19, 16], which aim to find better minima of single objective problems via MOO. A single objective can be converted to multiples ones by decomposition [21, 16] or by adding helper functions [19]. Nevertheless, the targeted problem instances in these works are hard combinatorial problems, and the MOO solvers are based on evolutionary algorithms and thus not immediately applicable to our continuous problem instances.

3. Background on robust estimation

We now introduce important definitions as well as our notations regarding robust estimation. In this paper, we are interested in minimizing cost functions of the form:

$$\min_{\mathbf{x}} \Psi(\mathbf{x}) \quad \text{with} \quad \Psi(\mathbf{x}) = \sum_{i=1}^N \psi(\|\mathbf{r}_i(\mathbf{x})\|), \quad (1)$$

where N is the number of data points, $\mathbf{x} \in \mathbb{R}^p$ are the parameters of interest, $\mathbf{r}_i : \mathbb{R}^p \rightarrow \mathbb{R}^n$ is the vectorial residual function corresponding to data point $\mathbf{y}_i \in \mathbb{R}^n$ and $\|\cdot\|$ is the L^2 -norm. In a slight abuse of vocabulary, we call $\|\mathbf{r}_i(\mathbf{x})\|$ the residual corresponding to data point $\mathbf{y}_i$. The function $\psi : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a *robust kernel*, *i.e.* $\psi(0) = 0$,

¹In this work we mostly consider “non-parametric” estimation problem, where the number of unknowns is in the order of available observations. Refining randomly sampled, low-parametric models is *e.g.* addressed in [22].

$\psi''(0) = 1$, and the mapping $z \mapsto \psi(\sqrt{2}z)$ is monotonically increasing and concave. To gain some intuition regarding the previous definition, let us highlight the fact that robust kernels show subquadratic behavior. Consequently, we have $\psi(\|\mathbf{r}_i(\mathbf{x})\|) \leq \|\mathbf{r}_i(\mathbf{x})\|^2/2$, *i.e.* ψ downweights the cost of large residuals but, ψ behaves like $x \mapsto x^2/2$ for $x \approx 0$. We will also need a way to transform a given robust kernel ψ into another (less) robust kernel that is easier to optimize. To do so, we will use the following modification of ψ : for $\tau > 0$ we denote the scaled kernel by ψ_τ , *i.e.* $\psi_\tau(x) = \tau^2\psi(x/\tau)$. It can be verified that ψ_τ is again a robust kernel for all $\tau > 0$, and that ψ_τ is an upper bound of ψ for $\tau \geq 1$. Thus, scaling a kernel ψ , with $\tau \geq 1$, produces another robust kernel ψ_τ that is more sensitive to outliers than ψ . ψ_τ for $\tau > 1$ implies also a *smoothed* cost Ψ_τ with fewer local minima than the original cost function Ψ (illustrated in Fig. 1).

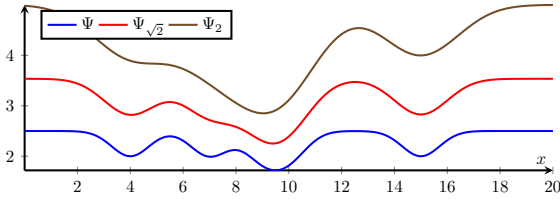


Figure 1: The impact of scaling a robust kernel on the difficulty of a robust cost.

4. The limitations of joint half-quadratic optimization

The aim of this section is to present two observations about joint HQ minimization (*i.e.* lifting-based approaches) that allow to reveal some limitations of these methods, and thus justifies our choice of not basing our method on them: first, we empirically show that initializing with optimistic confidence weights (which correspond to the lifting variables for joint HQ minimization) is critical, and second, we present a clear failure case for joint HQ optimization.

Let us first recall that IRLS can be derived by applying alternating optimization to the HQ cost (*i.e.* freeze the parameters and find the optimal value for the lifting variables, then freeze the (optimal) lifting variables and apply a Levenberg-Marquardt (LM) step to the parameters).

In order to get some insight on the importance of the initial value of lifting variables/confidence weights for joint HQ minimization, we select a subset of the “bundle adjustment in the large” dataset [1], and apply IRLS and joint HQ minimization methods (and variations thereof) on *linearized* bundle adjustment instances. We use linearized residuals to factor out additional effects of using non-linear residuals. We chose the smooth truncated kernel [33] with scale parameters $\sigma \in \{1, 1/2\}$ as underlying robust kernel.

Figure 2 illustrates the objectives reached after 100 iterations of the corresponding LM solver. Here IRLS denotes the standard IRLS approach, IRLS-alt is a version of IRLS, where the order of alternation steps is reversed (given confidence weights update the parameters, given parameters update the weights, and repeat). Further, the initial confidence weights are set to 1, therefore the first iteration of IRLS-alt is a pure non-robust fitting step while the rest of the iterations are standard IRLS steps. HQ refers to joint HQ minimization (with lifting variables also initialized to 1), and HQ* refers to joint HQ with optimal initial lifting variables (*i.e.* set to the IRLS induced ones). It can be seen in Fig. 2 that IRLS-alt outperforms HQ* on the majority of instances and is close to joint HQ minimization in a number of datasets. We conclude, and this is one insight of this section, that initializing confidence weights to one (or at least optimistically) is of higher relevance for reaching a good minimum than the choice of joint or alternating optimization strategies.

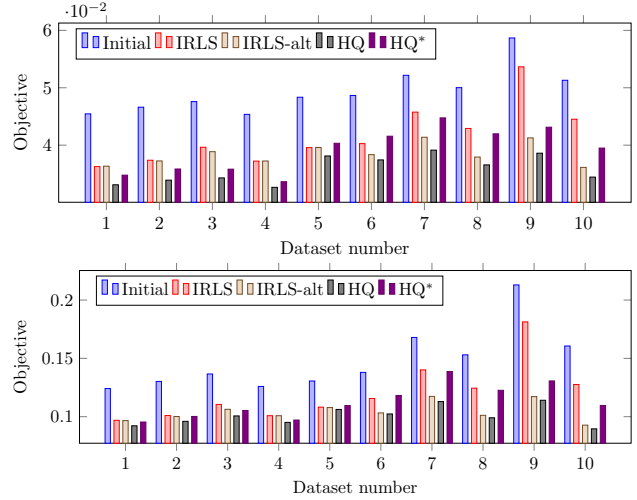


Figure 2: Final objectives for linearized bundle adjustment instances with scale parameter $\sigma = 1$ (top) and $\sigma = 1/2$ (bottom). Observe how IRLS-alt often outperforms HQ*, hence initializing the confidence weights optimistically is of higher relevance for reaching a good minimum than the choice of a joint or alternating minimization strategy.

One important failure case for joint HQ minimization, and this is the second insight of this section, is the presence of mutually exclusive residuals, *i.e.* when related groups of residuals have at most one inlier residual that can be explained by optimally fitted parameters. Such mutually exclusive residuals appear in 3D computer vision problems *e.g.* when allowing multiple, non-unique correspondences. Sparse 3D reconstruction and multi-object tracking are naturally formulated with such multiple matches. Maximum a-posteriori (MAP) estimation tasks can also fall into this

category, if the given clique potentials have multiple local minima. This is usually the case for dense correspondence estimation and for range image fusion. Figure 4 illustrates results for a depth estimation task, where each pixel (i.e. unknown) has $K \geq 1$ mutually exclusive residuals. The exact robust objective is given in Sec. 6, but Fig. 3 illustrates how photo-consistency scores are converted to a robust cost problem by introducing terms for each of the K smallest local minima in the cost profile. The starting point is the same, randomly initialized depth map (to enhance the visual differences). Note that the underlying objective has linear residuals, and therefore the only source of non-convexity are the robust kernels. It can be seen in Fig. 4 that joint HQ optimization returns increasingly poorer answers (compared to the visually more appealing and consistent results of the novel method we present in sec.5), when K is increased. This is also reflected in the final objective values reached by the methods (cf. Fig. 7).

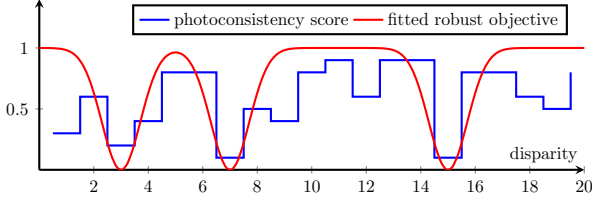


Figure 3: The conversion of photo-consistency scores into a robust objective with multiple local minima.

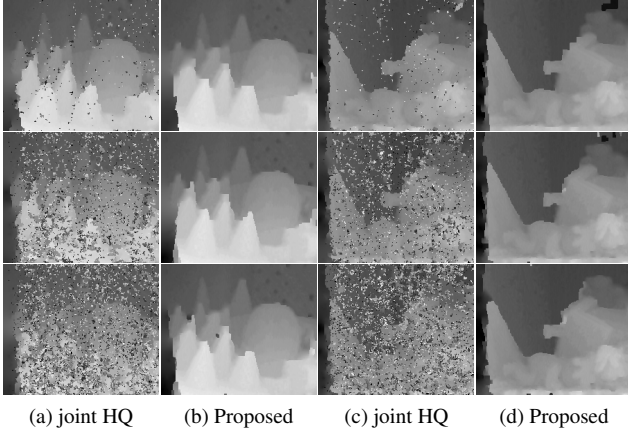


Figure 4: Depth map estimation results using different numbers of residuals per pixel (top $K = 1$, middle $K = 2$, bottom $K = 3$ residuals per pixel).

In summary, in this section we have demonstrated that the good practical performance of joint HQ minimization relies on many residuals actually being inlier residuals (at the optimal solution), and therefore initializing the confidence weights optimistically is reasonable. If this assumption

is violated (e.g. only a small fraction of residuals can be explained even for optimal parameters), then the limitations of joint HQ minimization are revealed. Consequently, our proposed method deviates significantly from joint HQ minimization.

5. Levenberg-Marquardt method for multi-objective optimization

Our optimization algorithm for robust costs relies on multi-objective optimization to be able to (i) avoid poor local minima while (ii) reducing the target cost function at each iteration of the (underlying) NLLS solver. MOO has so far very limited use in computer vision and machine learning, e.g. [27] recently proposes to use gradient-based MOO to tackle multi-task learning. In our application of robustified non-linear least squares problems, we are strongly interested in second-order optimization methods as first-order methods show simply too slow convergence in practice.

In this section, we first briefly introduce MOO in sec.5.1 while our contributions are described in Sec.5.2 and 5.3.

5.1. Background on multi-objective optimization

If multiple differentiable cost functions $(f_1, \dots, f_M)$ are given, then a point $\mathbf{x}$ that minimizes all the objectives simultaneously does usually not exist. Instead, the goal of MOO is to determine one or all (locally) Pareto critical points, *i.e.* points $\mathbf{x}$ such that there is no common direction of strict descent. Stated differently, moving in any direction around a locally Pareto critical point $\mathbf{x}$ may improve several cost functions but necessarily increases the cost of at least one of the objectives $(f_1, \dots, f_M)$.

Methods to find a direction $\mathbf{v}$ (if it exists), that is a descent direction for all costs, are proposed in [11, 6]. In particular, [11] suggests the following quadratic program (QP),

$$\min_{\mathbf{v}} \max_i \{ \mathbf{v}^T \nabla f_i(\mathbf{x}) \} + \frac{1}{2} \|\mathbf{v}\|^2. \quad (2)$$

It is shown that $\mathbf{x}$ is Pareto critical iff the optimal value of the QP is 0, otherwise the optimal $\mathbf{v}$ is a strict descent direction for all f_i . The method described in [6] explicitly distinguishes between an initial cooperative phase to find a Pareto critical solution, which is followed by an (optional) competitive phase refining the solution along the so called Pareto front. If $M = 2$, then a descent direction $\mathbf{v}$ is explicitly given by $\mathbf{v} = \beta \nabla f_1(\mathbf{x}) + (1 - \beta) \nabla f_2(\mathbf{x})$, where

$$\beta = \Pi_{[0,1]} \left(\frac{\nabla f_2(\mathbf{x})^T (\nabla f_2(\mathbf{x}) - \nabla f_1(\mathbf{x}))}{\|\nabla f_1(\mathbf{x}) - \nabla f_2(\mathbf{x})\|^2} \right)$$

for $\nabla f_1(\mathbf{x}) \neq \nabla f_2(\mathbf{x})$. If $\nabla f_1(\mathbf{x}) = \nabla f_2(\mathbf{x})$, then $\beta \in [0, 1]$ arbitrarily. Finally, if $\mathbf{v} = \mathbf{0}$, then $\mathbf{x}$ is Pareto critical.

Finding a Pareto critical point can be accelerated by using Newton's method [10]. Using a local (convex) quadratic

model of each objective, a trial solution is determined that maximizes the minimum decrease of each quadratic model. However, even for the case of two objectives this amounts to solve a quadratic program which is impractical for large-scale problems such as bundle adjustment. Hence, in the following sections, we propose a more scalable method that allows to better apply multi-objective optimization to large-scale robust estimation problems.

5.2. Efficient 2nd-order multi-objective solver

In this section we present an efficient second order method for multi-objective minimization when we have exactly two objectives. Thus, we are given objectives Ψ and $\tilde{\Psi}$, and the goal is to find a locally Pareto critical solution efficiently. In our application both Ψ and $\tilde{\Psi}$ will be robustified non-linear least squares objectives, but Ψ and $\tilde{\Psi}$ can be any pair of functions for which a second order method is suitable. The only weak requirement is, that $\tilde{\Psi}$ is in some sense “easier” to minimize, i.e. $\tilde{\Psi}$ is less prone of reaching a poor local minimum. As such $\tilde{\Psi}$ acts as a “guidance” function to navigate around poor local minima of Ψ . The construction of $\tilde{\Psi}$ given Ψ is usually application dependent, but if Ψ is a robustified non-linear least squares problem, then there are standard ways to design $\tilde{\Psi}$ (e.g. the one described in Sec. 3).

In the following, let $\mathbf{H}$ and $\tilde{\mathbf{H}}$ be the (p.s.d.) approximations of the Hessians of Ψ and $\tilde{\Psi}$, respectively, at a current linearization point $\mathbf{x}_0$. Further, let $\mathbf{g}$ and $\tilde{\mathbf{g}}$ be the corresponding gradients. Thus, the local quadratic models at $\mathbf{x}_0$ are given by

$$\begin{aligned}\Psi(\mathbf{x}_0 + \mathbf{v}) &\approx \Psi(\mathbf{x}_0) + \frac{1}{2}\mathbf{v}^T\mathbf{H}\mathbf{v} + \mathbf{g}^T\mathbf{v} =: m_\Psi(\mathbf{v}) \\ \tilde{\Psi}(\mathbf{x}_0 + \mathbf{v}) &\approx \tilde{\Psi}(\mathbf{x}_0) + \frac{1}{2}\mathbf{v}^T\tilde{\mathbf{H}}\mathbf{v} + \tilde{\mathbf{g}}^T\mathbf{v} =: m_{\tilde{\Psi}}(\mathbf{v}).\end{aligned}\quad (3)$$

In the following, we also make the simplifying assumption that m_Ψ and $m_{\tilde{\Psi}}$ are majorizing quadratic surrogates of Ψ and $\tilde{\Psi}$, respectively (i.e. $\Psi(\mathbf{x}_0) + m_\Psi(\mathbf{v}) \geq \Psi(\mathbf{x}_0 + \mathbf{v})$ and $\tilde{\Psi}(\mathbf{x}_0) + m_{\tilde{\Psi}}(\mathbf{v}) \geq \tilde{\Psi}(\mathbf{x}_0 + \mathbf{v})$ for all $\mathbf{v}$). In particular, IRLS-induced local quadratic models for linear residual functions satisfy this requirement.

In [10] it is proposed to determine a search direction $\mathbf{v}$ such that the minimal decrease in the quadratic models is maximal, i.e. one seeks the solution of

$$\min_{\mathbf{v}} \max \{m_\Psi(\mathbf{v}), m_{\tilde{\Psi}}(\mathbf{v})\}. \quad (4)$$

The idea of this convex program is that both objectives shall decrease as much as possible (according to the local quadratic models). Note that the optimal objective value of the above quadratic program is non-positive, since $\mathbf{v} = \mathbf{0}$ is feasible and has an objective value of 0. In contrast to gradient-based MOO methods [11, 6] there is no closed form solution for $\mathbf{v}$ given the Hessians $\{\mathbf{H}, \tilde{\mathbf{H}}\}$ and gradients $\{\mathbf{g}, \tilde{\mathbf{g}}\}$, and optimizing the above quadratic program

requires an iterative solver. Thus, we aim for an approach avoiding the explicit solution of Eq. 4.

Since $\tilde{\Psi}$ acts only as a guidance function, the exact absolute decrease in $\tilde{\Psi}$ is not significant, and therefore we relax Eq. 4 by introducing a non-negative scale factor β ,

$$\min_{\mathbf{v}} \max \{m_\Psi(\mathbf{v}), \beta m_{\tilde{\Psi}}(\mathbf{v})\}. \quad (5)$$

We provide a sufficient condition when Eq. 5 can be solved for some $\beta > 0$ by just solving a single linear system. In order to later handle non-linear residuals by regularizing the update vectors, we consider a damped version of Eq. 5.

Proposition 1. Let $\mu \in (0, 1)$ and $\nu > 0$ be given, and let $\mathbf{v}^*$ be the update vector given by

$$\mathbf{v}^* = -((1 - \mu)\mathbf{H} + \mu\tilde{\mathbf{H}} + \nu\mathbf{I})^{-1}((1 - \mu)\mathbf{g} + \mu\tilde{\mathbf{g}}). \quad (6)$$

If $m_\Psi(\mathbf{v}^*) < 0$ and $m_{\tilde{\Psi}}(\mathbf{v}^*) < 0$, then there exists a $\beta > 0$ and $\nu' > 0$ such that $\mathbf{v}^*$ is also the solution of

$$\min_{\mathbf{v}} \max \{m_\Psi(\mathbf{v}), \beta m_{\tilde{\Psi}}(\mathbf{v})\} + \frac{\nu'}{2}\|\mathbf{v}\|^2. \quad (7)$$

Proposition 1 means, that “strong” steps reducing both surrogate quadratic models correspond to solving a cooperative MOO step. Consequently, our algorithm will rely on solving Eq. 6 to be able to both avoid poor local minima (by reducing $\tilde{\Psi}$) and quickly improve the target cost Ψ .

Proposition 2. Let $\mathbf{g}$ and $\tilde{\mathbf{g}} \neq \mathbf{0}$. If $\mathbf{g} + \gamma\tilde{\mathbf{g}} = \mathbf{0}$ for some $\gamma \geq 0$, then $\mathbf{x}_0$ is locally Pareto critical for $(\Psi, \tilde{\Psi})$. Otherwise there exists a $\nu > 0$ and a $\mu \in (0, 1)$ such that $\mathbf{v}^*$ given by Eq. 6 satisfies $m_\Psi(\mathbf{v}^*) < 0$ and $m_{\tilde{\Psi}}(\mathbf{v}^*) < 0$. In particular, a universally admissible choice for μ is given by $\mu = \frac{\|\tilde{\mathbf{g}}\|}{\|\mathbf{g}\| + \|\tilde{\mathbf{g}}\|}$. This in turn implies $\Psi(\mathbf{x}_0 + \mathbf{v}^*) < \Psi(\mathbf{x}_0)$ and $\tilde{\Psi}(\mathbf{x}_0 + \mathbf{v}^*) < \tilde{\Psi}(\mathbf{x}_0)$.

Proposition 2 essentially implies that sufficiently large damping of $\mathbf{v}$ is guaranteed to yield a cooperative update unless $\mathbf{x}_0$ is already locally Pareto critical. Thus, proposition 2 justifies the use of a stopping criterion that tests whether the two gradients $\mathbf{g}$ and $\tilde{\mathbf{g}}$ are opposing or not.

Proposition 3. If $\Psi(\mathbf{x}_0 + \mathbf{v}^*) > \Psi(\mathbf{x}_0)$, then $m_\Psi(\mathbf{v}^*) > 0$. If $\tilde{\Psi}(\mathbf{x}_0 + \mathbf{v}^*) > \tilde{\Psi}(\mathbf{x}_0)$, then $m_{\tilde{\Psi}}(\mathbf{v}^*) > 0$.

Proposition 3 implies that the surrogate models m_Ψ and $m_{\tilde{\Psi}}$ will not disagree with their respective objectives Ψ and $\tilde{\Psi}$ for strong steps. As a consequence there is no need to test $m_\Psi(\mathbf{v}^*) > 0$ and $m_{\tilde{\Psi}}(\mathbf{v}^*) > 0$ explicitly in our algorithm. The proofs of these propositions are given in the supplementary material.

Algorithm 1 Multi-objective Levenberg-Marquardt method

Require: Target Ψ and guidance costs $(\Psi^1, \dots, \Psi^{K_{\max}})$

Require: Initial solution $\mathbf{x}_0$ and damping parameter $\nu > 0$

```
1:  $k \leftarrow K_{\max}$ 
2: repeat
3:    $\mu \leftarrow \frac{\|\nabla \Psi(\mathbf{x}_0)\|}{\|\nabla \Psi(\mathbf{x}_0)\| + \|\nabla \Psi^k(\mathbf{x}_0)\|}$ 
4:    $F^k \leftarrow (1 - \mu)\Psi + \mu\Psi^k$ 
5:    $\triangleright$  Gauss-Newton / IRLS model
6:    $\mathbf{g}_F \leftarrow \nabla F^k(\mathbf{x}_0)$     $\mathbf{H}_F \leftarrow \nabla^2 F^k(\mathbf{x}_0)$ 
7:    $\mathbf{v} \leftarrow -(\mathbf{H}_F + \nu \mathbf{I})^{-1} \mathbf{g}_F$     $\triangleright$  Search direction
8:    $\mathbf{x}^+ \leftarrow \mathbf{x}_0 + \mathbf{v}$ 
9:   if  $F^k(\mathbf{x}^+) < F^k(\mathbf{x}_0)$  then  $\triangleright$  Success to reduce  $F^k$ 
10:    strong  $\leftarrow \Psi(\mathbf{x}^+) < \Psi(\mathbf{x}_0) \wedge \Psi^k(\mathbf{x}^+) < \Psi^k(\mathbf{x}_0)$ 
11:    stop  $\leftarrow$  TEST-STOPPING( $\Psi, \Psi^k, \mathbf{x}_0, \mathbf{x}^+$ )
12:    if strong and not stop then
13:       $\mathbf{x}_0 \leftarrow \mathbf{x}^+$     $\triangleright$  Update  $\mathbf{x}_0$ 
14:    else    $\triangleright$  Failure to reduce  $\Psi$  and  $\Psi^k$ 
15:       $k \leftarrow k - 1$   $\triangleright$  Go to next guidance function
16:    end if
17:     $\nu \leftarrow \nu / 10$   $\triangleright$  Decrease the damping parameter
18:  else    $\triangleright$  Failure to reduce  $F^k$ 
19:     $\nu \leftarrow 10\nu$   $\triangleright$  Increase the damping parameter
20:  end if
21: until  $k = 0$ 
22: return the solution of a standard Levenberg-Marquardt
    method given current point  $\mathbf{x}_0$ 
```

5.3. The algorithm

We are now able to state the proposed multi-objective Levenberg-Marquardt method, which we call “LM-MOO.” We will allow a sequence of “guidance” functions $(\Psi^1, \dots, \Psi^{K_{\max}})$, which are assumed to have decreasing difficulty of finding a global minimum (similar to graduated optimization setting). The input of the method are the initial solution $\mathbf{x}_0$ and the initial value of the damping parameter ν . The pseudo-code is given in Alg. 1. The algorithm sequentially aims to cooperatively minimize both Ψ and Ψ^k until reaching an approximately Pareto critical solution (which leads to a decrease of k). Ψ^k is usually more similar to Ψ for smaller k . Once these guidance functions are exhausted, a standard solver is applied on Ψ (which in our implementation is just setting $\Psi^0 := \Psi$).

The function TEST-STOPPING determines whether the current solution is sufficiently close to a stationary point of $F^k = (1 - \mu)\Psi + \mu\Psi^k$. In principle, testing for a strong step decreasing both Ψ and Ψ^k is sufficient, but in practice this condition alone is too conservative and leads to poor use of computation time. Thus, we incorporate an additional test to detect sufficient convergence.

We use two implementations for TEST-STOPPING. The first one is a normalized reduction suggested in [32], which

in our setting can be written as follows

$$\frac{F^k(\mathbf{x}_0) - F^k(\mathbf{x}^+)}{\sum_j |F_j^k(\mathbf{x}^+) - F_j^k(\mathbf{x}_0)|} < \varepsilon_0, \quad (8)$$

where $F_j^k(\mathbf{x}) = (1 - \mu)\psi(\|\mathbf{f}_j(\mathbf{x})\|) + \mu\psi^k(\|\mathbf{f}_j(\mathbf{x})\|)$. F_j^k is the contribution of the j -th residual to the combined objective F^k . We will denote the algorithm using this stopping criterion by LM-MOO † .

Using the MOO perspective allows us to refine the above criterion to also detect when the objectives Ψ and Ψ^k are (nearly) conflicting at $\mathbf{x}_0$. This is indicated by the vectors $\mathbf{u} = \nabla \Psi(\mathbf{x}_0)$ and $\mathbf{v} = \nabla \Psi^k(\mathbf{x}_0)$ being (approximately) opposing. The second implementation of TEST-STOPPING uses a regularized cosine for this additional test,

$$\frac{\mathbf{u}^T \mathbf{v} + \min\{0, \min\{\|\mathbf{u}\|, \|\mathbf{v}\|\} - \varepsilon_1\}}{\|\mathbf{u}\| \|\mathbf{v}\| + \max\{0, \varepsilon_1 - \min\{\|\mathbf{u}\|, \|\mathbf{v}\|\}\}}, \quad (9)$$

to avoid undesired behavior when $\|\mathbf{u}\| \approx 0$ or $\|\mathbf{v}\| \approx 0$. This quantity is exactly $\frac{\mathbf{u}^T \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$ if $\|\mathbf{u}\| \geq \varepsilon_1$ and $\|\mathbf{v}\| \geq \varepsilon_1$. If $\|\mathbf{u}\| \ll \varepsilon_1$ or $\|\mathbf{v}\| \ll \varepsilon_1$, then this ratio approaches -1 . The function TEST-STOPPING returns true if this ratio is less than $1 - \varepsilon_2$ or Eq. 8 is satisfied (false otherwise). Thus, this version of TEST-STOPPING is also true if either sufficiently large input vectors are close to opposing directions, or if one (or both) of the vectors is approximately vanishing (which indicates that the current solution is almost a stationary point of Ψ or Ψ^k). We denote the induced algorithm by LM-MOO. In the numerical experiments we set $\varepsilon_0 = 1/10$, $\varepsilon_1 = 10^{-3}$ and $\varepsilon_2 = -0.95$.

The stopping criterion in line 10 of Alg. 1 is guaranteed to be eventually satisfied as the algorithm reduces Ψ and the current Ψ^k as long as the gradients of Ψ and Ψ^k at the current solution are not opposing. Observe that $(1 - \mu)\mathbf{H} + \mu\mathbf{H}^k + \nu\mathbf{I}$ is strictly positive definite and therefore has full rank. Thus, $\mathbf{v}^* = \mathbf{0}$ iff $\mathbf{g} = -\mu(1 - \mu)^{-1}\mathbf{g}$, but this means that the current solution $\mathbf{x}_0$ is locally Pareto critical. If $\mathbf{g}$ and $\mathbf{g}^k$ are not opposing, then due to Proposition 2 there exists a $\nu > 0$ such that $\mathbf{v}^*$ decreases both Ψ and Ψ^k .

For robust NLLS objectives Ψ and Ψ^k analogous to Eq. 1 the local quadratic models at the current solution $\mathbf{x}_0$ are obtained by (i) linearizing the residuals $\mathbf{f}_i(\mathbf{x}_0 + \delta) \approx \mathbf{f}_i(\mathbf{x}_0) + \mathbf{J}_i \delta$, and (ii) using the IRLS approach to convert the mappings $\mathbf{f} \mapsto \psi(\|\mathbf{f}\|)$ and $\mathbf{f} \mapsto \psi^k(\|\mathbf{f}\|)$ to quadratic functions. We implemented Alg. 1 in C++, and the update step given in line 6 of Alg. 1 is determined by a direct sparse solver [5] (which dominates the runtime).

6. Results

We run numerical experiments for two sets of problem instances: first, we evaluate the different methods on bundle adjustment problems. We consider bundle adjustment

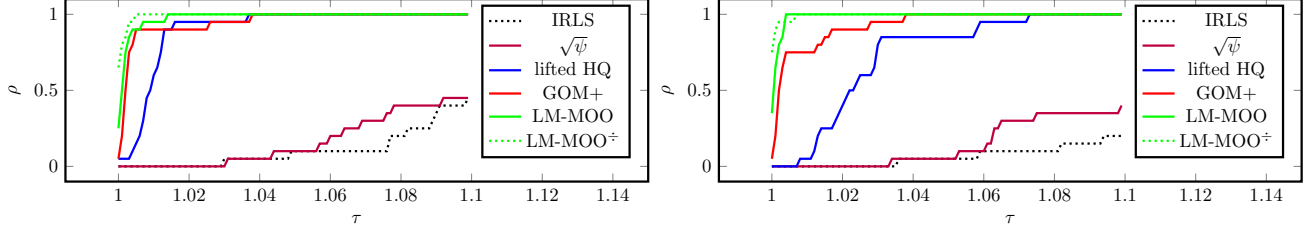


Figure 5: Performance profiles for linearized and metric bundle adjusted computed from 20 bundle adjustment instances.

as one of the main applications of robust cost minimization in computer vision. Second, we assess the ability of the algorithms to avoid poor local minima in the multiple data-association setting by maximum a posteriori inference for dense correspondence estimation. In all experiments we limit the number of iterations (i.e. the number of times line 6 of Alg. 1 or the respective normal equation is solved) to 100. Especially bundle adjustment instances will not be converged in terms of standard stopping criteria after this number of iterations, but the achieved objectives and solutions are effectively stable. In the experiments the target kernel is chosen to be the smooth truncated kernel ψ^{ST} [33], since it is a close approximation to an “ideal” truncated quadratic cost. As guidance functions we choose its scaled versions ψ_{τ}^{ST} with $\tau = 2^k$ for $k = 1, \dots, 4$. More results are provided in the supplementary material.

Bundle adjustment We use the camera parametrization suggested in [1], and therefore the bundle adjustment objective is given by

$$\sum \psi(f_i \eta_i(\pi(\mathbf{R}_i \mathbf{X}_j + \mathbf{t}_i)) - \hat{\mathbf{p}}_{ij}), \quad (10)$$

where $\hat{\mathbf{p}}_{ij} \in \mathbb{R}^2$ is the image point associated with the 3D point $\mathbf{X}_j \in \mathbb{R}^3$ in the i -th image, $\mathbf{R}_i \in SO(3)$ and $\mathbf{t}_i \in \mathbb{R}^3$ are the orientation parameters of the i -th camera, $\pi: \mathbb{R}^3 \rightarrow \mathbb{R}^2$, $\pi(\mathbf{X}) = \mathbf{X}/\mathbf{X}_3$ is the projection function, and f_i is the focal length of camera i . η_i is the lens distortion function with $\eta_i(\mathbf{p}) = (1 + k_{i,1}\|\mathbf{p}\|^2 + k_{i,2}\|\mathbf{p}\|^4)\mathbf{p}$, and ψ is the smooth truncated kernel.

We use 20 instances from the dataset in conjunction with [1] (listed in the supplementary material, with the number of cameras ranging from 73 to 744). We use the achieved outlier fraction as performance measure (since it is more meaningful than e.g. objective values). Fig. 5 visualizes the obtained performance profiles [7] for different methods applied on linearized and metric bundle adjustment. The graphs have to be read as follows: for a multiplicative factor $\tau \geq 1$, the value ρ is the fraction of instances such that the performance measure (in our case the outlier ratio) is less than τ times of the result achieved by the best algorithm. At $\tau = 1$ it reports the relative frequency of an algorithm being the best method. Although

performance profiles are widely used to compare software for numerical optimization, some care is required when interpreting the resulting graphs [14, 17]. Nevertheless, we use performance profiles to visualize a summary of the obtained numerical results. According to the profiles illustrated in Fig. 5, LM-MOO with either stopping criterion is highly competitive.

The speed of convergence is illustrated in Fig. 6. It can be seen in these figures that the proposed MOO-based method is state-of-the-art in terms of the reached objective, but also highly competitive in terms of how quickly the target objective is reduced. We refer to the supplementary material for a complete set of convergence graphs.

Dense correspondence We chose dense correspondence as multiple data association task, since the solution quality is very easy to assess visually. The underlying objective is given by

$$\sum_{p \in \mathcal{V}} \left(\lambda \sum_{k=1}^K \psi_{\text{data}}(d_p - \hat{d}_{p,k}) + \sum_{q \in \mathcal{N}(p)} \psi_{\text{reg}}(d_p - d_q) \right),$$

where $\hat{d}_{p,k}$ is the position of the k -th local minimum of the matching cost profile at pixel p (recall Fig. 3). ψ_{data} , ψ_{reg} and λ are chosen to yield visually sensible results. Fig. 7 summarize the results for the “teddy” and “cones” pair. In contrast to Fig. 4 the starting point for all methods is initialized with the more reasonable best-cost solution. Joint HQ minimization improves the initial solution minimally, and GOM+ and our proposed method reach similar minima, but the convergence speed of GOM+ is significantly slower.

7. Conclusion

In this paper we derive a novel NLLS-based robust optimization algorithm that both quickly decreases the target objective and possesses a strong ability to avoid poor local minima. We identify that the optimistic initialization of the confidence weights in half-quadratic lifting approaches is of higher importance to avoid poor local minima than the joint optimization strategy. This experimental analysis allows us to demonstrate failure cases of joint HQ for diffi-

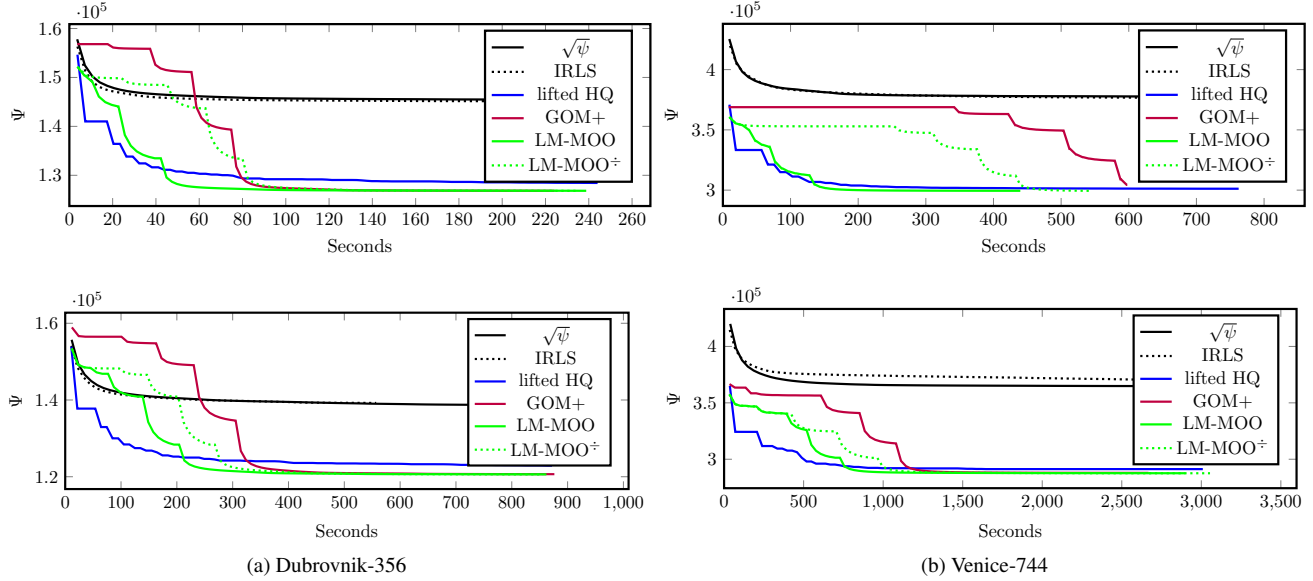


Figure 6: Best encountered objective values obtained versus wall clock time as reported by different methods for linearized (top) and metric (bottom) bundle adjustment instances.

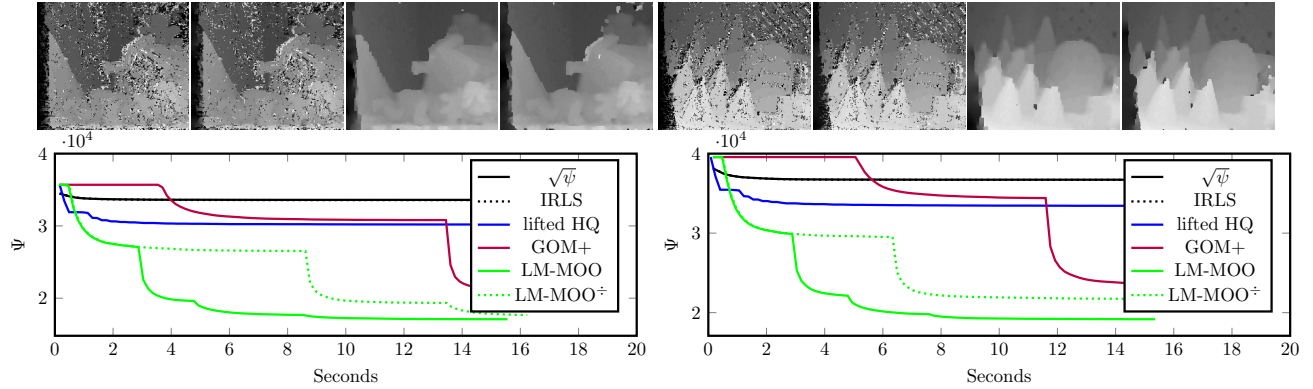


Figure 7: Top: Initial best-cost depth and solutions of joint HQ, GOM+ and LM-MOO, respectively, for the “teddy” and “cones” stereo pair. Bottom: best objectives reached vs. runtime for different methods.

cult problems with mutually exclusive residuals. As a consequence, we decide to rely on a smoothing-based scheme instead of half-quadratic lifting to reach our goal. The question at hand therefore is: how can we combine an algorithm that uses a smoothing mechanism to avoid poor local minima while maintaining an ability to quickly decrease the target objective? In order to answer that question, we propose to leverage a multi-objective optimization framework that allows us to obtain an algorithm matching the two aforementioned properties. We believe that our algorithm will be useful in many computer vision applications where a trade-off between computational time and ability to avoid poor local minima is required.

Future work will address the application of MOO-inspired methods to other difficult, continuous optimiza-

tion problems. In particular, highly non-linear residuals still pose an important challenge.

Acknowledgements: This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

- [1] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *Proc. ECCV*, pages 29–42. Springer, 2010. 3, 7
- [2] Michael J Black and Paul Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow

- fields. *Computer vision and image understanding*, 63(1):75–104, 1996. 2
- [3] Michael J Black and Anand Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *IJCV*, 19(1):57–91, 1996. 2
- [4] Andrew Blake and Andrew Zisserman. *Visual reconstruction*. 1987. 2
- [5] Timothy A Davis and Yifan Hu. The university of florida sparse matrix collection. *ACM Transactions on Mathematical Software (TOMS)*, 38(1):1, 2011. 6
- [6] Jean-Antoine Désidéri. Multiple-Gradient Descent Algorithm (MGDA). Research Report RR-6953, INRIA, June 2009. 4, 5
- [7] Elizabeth D Dolan and Jorge J Moré. Benchmarking optimization software with performance profiles. *Mathematical programming*, 91(2):201–213, 2002. 7
- [8] Daniel M Dunlavy and Dianne P O’Leary. Homotopy optimization methods for global optimization. Technical report, Sandia National Laboratories, 2005. 2
- [9] Chris Engels, Henrik Stewénus, and David Nistér. Bundle adjustment rules. In *Photogrammetric Computer Vision (PCV)*, 2006. 2
- [10] Joerg Fliege, LM Grana Drummond, and Benar Fux Svaiter. Newton’s method for multiobjective optimization. *SIAM Journal on Optimization*, 20(2):602–626, 2009. 4, 5
- [11] Jörg Fliege and Benar Fux Svaiter. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, 51(3):479–494, 2000. 4, 5
- [12] Donald Geman and George Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):367–383, 1992. 2
- [13] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing*, 4(7):932–946, 1995. 2
- [14] Nicholas Gould and Jennifer Scott. A note on performance profiles for benchmarking software. *ACM Transactions on Mathematical Software (TOMS)*, 43(2), 2016. 7
- [15] Peter J Green. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 149–192, 1984. 2
- [16] Julia Handl, Simon C Lovell, and Joshua Knowles. Multi-objectivization by decomposition of scalar cost functions. In *International Conference on Parallel Problem Solving from Nature*, pages 31–40. Springer, 2008. 2
- [17] Rasoul Hekmati and Hanieh Mirhajianmoghadam. Nested performance profiles for benchmarking software. *arXiv preprint arXiv:1809.06270*, 2018. 7
- [18] Peter J Huber. *Robust statistics*. Wiley, 1981. 1
- [19] Mikkel T Jensen. Guiding single-objective optimization using multi-objective methods. In *Workshops on Applications of Evolutionary Computation*, pages 268–279. Springer, 2003. 2
- [20] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983. 2
- [21] Joshua D Knowles, Richard A Watson, and David W Corne. Reducing local optima in single-objective problems by multi-objectivization. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 269–283. Springer, 2001. 2
- [22] Huu Le, Tat-Jun Chin, and David Suter. An exact penalty method for locally convergent maximum consensus. In *Proc. CVPR*, 2017. 2
- [23] Hossein Mobahi and John W Fisher. On the link between gaussian homotopy continuation and convex envelopes. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 43–56. Springer, 2015. 2
- [24] Hossein Mobahi and John W Fisher III. A theoretical analysis of optimization by gaussian continuation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015. 2
- [25] Pulak Purkait, Christopher Zach, and Anders Eriksson. Maximum consensus parameter estimation by reweighted ℓ_1 methods. In *EMMCVPR*, pages 312–327. Springer, 2017. 2
- [26] Kenneth Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proceedings of the IEEE*, 86(11):2210–2239, 1998. 2
- [27] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *NIPS*, 2018. 4
- [28] Bill Triggs, Philip McLauchlan, Richard Hartley, and Andrew Fitzgibbon. Bundle adjustment – A modern synthesis. In *Vision Algorithms: Theory and Practice*, volume 1883 of *LNCS*, pages 298–372, 2000. 2
- [29] Ming Ye, Robert M Haralick, and Linda G Shapiro. Estimating piecewise-smooth optical flow with global matching and graduated optimization. *IEEE transactions on pattern analysis and machine intelligence*, 25(12):1625–1630, 2003. 2
- [30] Christopher Zach. Robust bundle adjustment revisited. In *Proc. ECCV*, pages 772–787, 2014. 2
- [31] Christopher Zach and Guillaume Bourmaud. Iterated lifting for robust cost optimization. In *Proc. BMVC*, 2017. 2
- [32] Christopher Zach and Guillaume Bourmaud. Descending, lifting or smoothing: Secrets of robust cost optimization. In *Proc. ECCV*, 2018. 2, 6
- [33] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, and Marc Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. In *SIGGRAPH*, 2014. 3, 7