A Global Optimization Approach to Robust Multi-Model Fitting

Jin Yu, Tat-Jun Chin, and David Suter

The Australian Centre for Visual Technologies, The University of Adelaide, South Australia

{jin.yu, tjchin, dsuter}@cs.adelaide.edu.au

Abstract

We present a novel Quadratic Program (QP) formulation for robust multi-model fitting of geometric structures in vision data. Our objective function enforces both the fidelity of a model to the data and the similarity between its associated inliers. Departing from most previous optimization-based approaches, the outcome of our method is a ranking of a given set of putative models, instead of a pre-specified number of "good" candidates (or an attempt to decide the right number of models). This is particularly useful when the number of structures in the data is a priori unascertainable due to unknown intent and purposes. Another key advantage of our approach is that it operates in a unified optimization framework, and the standard QP form of our problem formulation permits globally convergent optimization techniques. We tested our method on several geometric multi-model fitting problems on both synthetic and real data. Experiments show that our method consistently achieves state-of-the-art results.

1. Introduction

The task of fitting mathematical models to data, *i.e.* model fitting, underlies numerous applications in computer vision. In practice, model fitting is often non-trivial because real-life vision data typically contain multiple geometric structures, and are almost always contaminated by noise including measurement noise, gross outliers, and *pseudo-outliers* that are in fact inliers to other structures coexisting in the data. In order to be robust to noise, many model fitting methods, *e.g.* the popular RANSAC approach [1] and its many extensions [5, 10, 19], adopt a *hypothesize-and-verify* procedure, which works by repeatedly generating putative models (or hypotheses) from random subsets of the data, and then evaluating these models according to some quality measure. To cope with multiple structures, a sequential *fit-and-remove* approach is commonly used, which excludes inliers of detected structures from further consideration. A major limitation of this approach is that it can not guarantee the optimality of the solution.

A recent trend in multi-model fitting [6, 8, 9, 14] is to pose the selection of multiple models as an optimization problem in such a form that enables the application of standard optimization techniques. These approaches operate in a unified optimization framework, which *in theory* offers a guarantee on solution optimality (although many previous methods deviate from this goal due to computational intractability). To deal with multi-structure data, these methods employ a model selection criterion, *e.g.* the Akaike information criterion [2], to determine the number of models to select. Unfortunately, this typically involves elaborate (and often unintuitive) tuning of regularization parameters.

The proposed approach belongs to this class of methods. However, in contrast to previous approaches, it does not attempt to decide the correct number of models during the optimization. Instead, it allows users to perform a more informed post-processing model selection by providing a *ranked list* of putative models. The ranking is obtained by globally optimizing a convex and quadratic objective function of model weights that balances model quality and diversity. Fig. 1 depicts an example ranking of putative 4D linear subspaces for the application of motion segmentation [17]. As shown in the figure, the subspaces ranked top 3 by our method clearly separate the two moving objects (circles and triangles) from the background scene (stars).

Another major challenge faced by optimization approaches to multi-model fitting lies in the computational tractability of their problem formulations. The majority of current approaches must resort to approximate solutions to remain computationally feasible. For instance, the Uncapacitated Facility Location (UFL) formulation proposed in [9] is, in general, NP-hard. Although a Linear Program (LP) relaxation of UFL exists, due to its large problem size, only relatively few putative models can be handled. A message-passing inference algorithm was later proposed in [8] to solve the original UFL problem, but only approximate solutions are available. Thakoor and Gao [14] introduced a combinatorial optimization formulation for model selection, and tailored a Branch-and-Bound-based approach to solve the resulting problem. Their approach, however, faces a worst case exponential complexity in the number

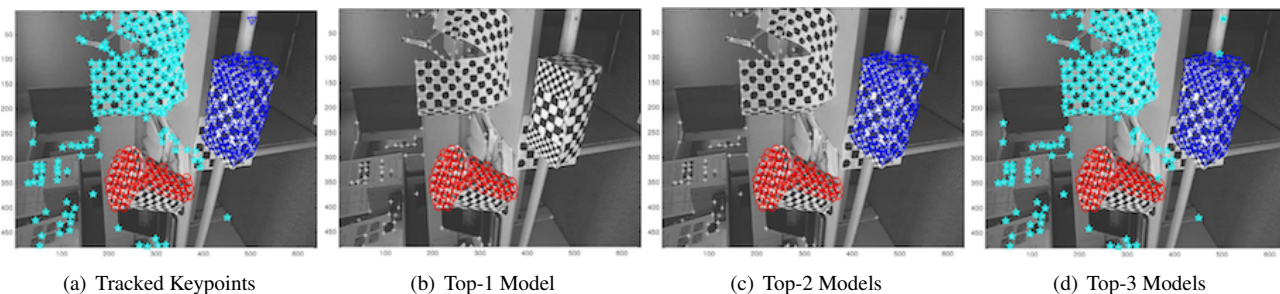| (a) Tracked Keypoints | (b) Top-1 Model | (c) Top-2 Models | (d) Top-3 Models |

Figure 1. (a): Tracked keypoints on a static background (stars) as well as two moving objects of different motions (circles and triangles). They are clearly separated by the 4D linear subspaces (b-d) ranked top 3 by our method among 10^3 putative motion subspace models. Putative models were generated from the "1R2RCT_A" keypoint trajectories in the Hopkins155 benchmark [17]. We assigned keypoints to their most consistent subspace (*c.f.* (13) for our consistency measure).

of putative models and thus is intractable for practical use. More recently, an energy-based approach was proposed in [6], which emphasized a spatial consistency in the clustering results given by the selected models. Since the energy function is NP-hard to minimize, the authors resort to approximate solutions. Departing from previous formulations, our optimization problem is a standard Quadratic Program (QP). The size of our QP is linear in the number of putative models. Large-scale QP solvers such as the MOSEK solver[1] can solve a QP with several thousands of variables in just a few seconds.

The rest of the paper is organized as follows: We first provide a general description of our problem formulation, and list key steps for solving the problem. Those key steps are detailed subsequently in Secs. 3 and 4, where we show how to measure the quality of a model and how to regularize overlapping models, respectively. Experimental results on synthetic and real data are reported in Sec. 5. We conclude the paper with a discussion in Sec. 6.

2. Model Fitting as a Quadratic Program

Our work is inspired by the QP formulation of the classical mean-variance-based portfolio optimization [12, Ch.1] in financial applications, where a set of assets are selected with optimal weights to form an investment portfolio. The optimization problem there reflects a trade-off between expected return and risk, with risk measured as covariance between (returns of) assets. Following this concept, we design a generic optimization framework for the simultaneous selection of multiple geometric models in computer vision. Our method optimizes the ranking (or importance weights) of putative models based on both the qualities of the given models and their correlation (overlap) with each other.

We first give a general description of our problem formulation. Let $\mathcal{X} := \{\boldsymbol{x}_i\}_{i=1}^{N}$ be a set of N input data, and $\mathcal{M} := \{\boldsymbol{\theta}_m\}_{m=1}^{M}$ a series of M putative models that are generated from minimal subsets of $\mathcal{X}$, *e.g.* a set of 4

Algorithm 1 Model Fitting by Quadratic Programming (QP-MF)

1: **input** residuals (5) of all data as measured to the M putative models and parameters $T, \alpha, \beta > 0$ as in (2).
2: **output** a ranking of the M putative models
3: compute similarities between all data via (9)
4: use data similarities to form inlier sets $\mathcal{I}_m$, for $m = 1, \cdots, M$ (Sec. 3.2)
5: compute the model fidelity and inlier similarity terms in (2) via (1) and (12), respectively
6: compute similarities between all models   (Sec. 4)
7: plug results from previous two steps to (2), and solve it to obtain the optimal weight vector $\boldsymbol{t}^*$ via Quadratic Programming
8: **return** $\pi = \texttt{argsort}(-\boldsymbol{t}^*)$ (*i.e.* indices sorted by non-ascending values of $t_m^*, \; m = 1, \cdots, M$)

---

keypoint correspondences for homography estimation. Our goal is to optimally rank putative models in $\mathcal{M}$ such that each of the top-ranked models represents a different structure in the data. Toward this end, we associate with each model a *weight variable* $t_m \in [0, 1], m = 1, \cdots M$. The optimization is then performed over these variables to obtain the optimal weights $\boldsymbol{t}^* := [t_1^*, \cdots, t_M^*]$. The permutation that sorts $\boldsymbol{t}^*$ in non-ascending order provides the ranking.

Our objective function $J : \mathbb{R}^M \rightarrow \mathbb{R}$ makes use of a domain-specific loss function $l(\boldsymbol{x}_i, \boldsymbol{\theta}_m)$, *e.g.* the Sampson distance [7] if the application is homography or fundamental matrix estimation. It computes the residual of $\boldsymbol{x}_i$ as measured to a model $\boldsymbol{\theta}_m$. The discrepancy between a model and the data, *i.e.* model fidelity, is then quantified as

$$L(\mathcal{I}_m, \boldsymbol{\theta}_m) := \frac{1}{|\mathcal{I}_m|} \sum_{i \in \mathcal{I}_m} l(\boldsymbol{x}_i, \boldsymbol{\theta}_m), \qquad (1)$$

where $\mathcal{I}_m$ contains the indices of data that support $\boldsymbol{\theta}_m$, *i.e.* $\mathcal{I}_m$ is a set of inliers for $\boldsymbol{\theta}_m$ (Sec. 3.2).

Recent work in multi-model fitting [3, 4, 15] discovers that data from a coherent structure share similar preferences (given by residual sorting) towards putative models.

---

[1]Available from http://www.mosek.com.

Exploiting this novel feature, we include an inlier similarity term $f(\mathcal{I}_m)$ (defined in Sec. 3.2) in our objective function so as to enforce consistent preferences of data in $\mathcal{I}_m$ to putative models. Moreover, to handle overlapped models (Sec. 4), we incorporate a regularizer into our objective function to encourage diversity in the top-ranked models.

The resulting optimization problem takes the form:

$$\min \; J(\boldsymbol{t}) := \sum_{m=1}^{M} t_m \underbrace{L(\mathcal{I}_m, \boldsymbol{\theta}_m)}_{\text{Model Fidelity}} - \alpha \sum_{m=1}^{M} t_m \underbrace{f(\mathcal{I}_m)}_{\text{Inlier Similarity}}$$

$$+ \beta \underbrace{\sum_{m,n=1}^{M} t_m \, t_n \, s(\boldsymbol{\theta}_m, \boldsymbol{\theta}_n)}_{\text{Regularizer}} \qquad (2)$$

$$\text{s.t. } \boldsymbol{t} \in [0,1]^M, \quad \sum_{m=1}^{M} t_m \geq T, \qquad (3)$$

where $L(\cdot, \cdot)$ is the model fidelity measure defined in (1), $f(\mathcal{I}_m)$ quantifies the similarity between data in $\mathcal{I}_m$, and the parameter $T \in [1, M]$ is a positive integer that guarantees a minimum number of non-zero weights. Free parameters $\alpha, \beta > 0$ trade off various terms in $J(\boldsymbol{t})$, both being automatically set in the experiments of Sec. 5. The function $s : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$ in the quadratic regularizer evaluates the degree of overlap between two putative models, where $\mathcal{A}$ is the domain of model parameters. The regularizer is designed to penalize simultaneously assigning high weights to overlapped models, and hence effectively encourages the top-ranked models to represent diverse structures in data. The model similarity measure $s(\cdot, \cdot)$ is defined in Sec. 4. We will also prove that $s(\cdot, \cdot)$ is a positive-definite kernel function. This means that the regularizer in (2) is non-negative for any values of t_m and t_n [13]:

$$(\forall t_m, t_n) \sum_{m,n=1}^{M} t_m \, t_n \, s(\boldsymbol{\theta}_m, \boldsymbol{\theta}_n) \geq 0, \qquad (4)$$

implying that the Hessian matrix $\nabla^2 J \in \mathbb{R}^{M \times M}$ of the objective function is positive-semidefinite. Therefore, $J(\boldsymbol{t})$ is a convex function. Moreover, since $J(\boldsymbol{t})$ is quadratic and the constrains in (3) are all linear in $\boldsymbol{t}$, our optimization problem is a standard QP, thus can be minimized to optimality by various constrained optimization techniques, *e.g.* interior-point methods [11].

Our approach is detailed in Alg. 1, which involves the following three key steps of operations:

1. forming the inlier sets $\mathcal{I}_m$ for all putative models,
2. measuring the degree of overlap between putative models, *i.e.* computing $s(\boldsymbol{\theta}_m, \boldsymbol{\theta}_n)$ in (2), and
3. solving the QP (Eqns. 2 and 3) to obtain the optimal weights, sorting which in non-ascending order gives the optimal ranking of putative models.

In what follows we discuss the first two key steps. The standard QP form of our problem enables the use of off-the-shelf QP solvers to complete the final step.

3. Quantifying Model Quality

We introduce a novel data similarity measure, and show how to utilise it to form the inlier set $\mathcal{I}_m$ for the computation of the model fidelity and inlier similarity terms in (2).

3.1. A Similarity Measure Based on Permutations

As demonstrated in recent work [3, 4, 15], inliers from the *same* structure share very similar preferences to putative models, while cross-structure inliers and gross outliers do not exhibit this property. Building on this observation, we design a robust data similarity measure that is capable of differentiating inliers of the same structure from other data.

To obtain the preference of each datum $\boldsymbol{x}_i$ to putative models, we first compute its absolute residuals as measured to the M putative models to form a residual vector

$$\boldsymbol{r}^{(i)} := [\, r_1^{(i)} \; r_2^{(i)} \; \cdots \; r_M^{(i)} \,]. \qquad (5)$$

The elements in $\boldsymbol{r}^{(i)}$ are then sorted in non-descending order to obtain the permutation

$$\boldsymbol{\pi}^{(i)} := [\, \pi_1^{(i)} \; \pi_2^{(i)} \; \cdots \; \pi_M^{(i)} \,] \qquad (6)$$

such that $p < q \Rightarrow r_{\pi_p^{(i)}}^{(i)} \leq r_{\pi_q^{(i)}}^{(i)}$.

The permutation $\boldsymbol{\pi}^{(i)}$ essentially encodes the preference of $\boldsymbol{x}_i$ to the putative models. To compare preferences of two data, we compute the intersection between leading elements in their corresponding permutations:

$$s_t(\boldsymbol{x}_i, \boldsymbol{x}_j) := |\boldsymbol{\pi}^{(i)}[h_t^{(i)}] \cap \boldsymbol{\pi}^{(j)}[h_t^{(j)}]| / (h_t^{(i)} \times h_t^{(j)})^{\frac{1}{2}}, \quad (7)$$

where $|\cdot|$ denotes the cardinality of a set (or a vector), the notation $\boldsymbol{a}[b]$ refers to a set that consists of the first b elements of a vector $\boldsymbol{a}$. The *bandwidth* $h_t^{(i)}$ is given by

$$h_t^{(i)} := \min(|\boldsymbol{\pi}^{(i)}|, \, t \times h^{(i)}), \qquad (8)$$

where t is a positive integer, $|\boldsymbol{\pi}^{(i)}|$ is the full length of the permutation (M here), and $h^{(i)} \in \mathbb{N}_+$ is a *basis bandwidth*, controlling the discriminative power of the similarity measure (7) for a given t. Since a datum is most well characterized by its preferred models, a natural choice for $h^{(i)}$ is an estimated number of models that $\boldsymbol{x}_i$ strongly agrees with, *e.g.* it was determined by (15) in our experiments. For models that are less preferred by $\boldsymbol{x}_i$ (hence their rankings are less informative), their contributions are discounted. The
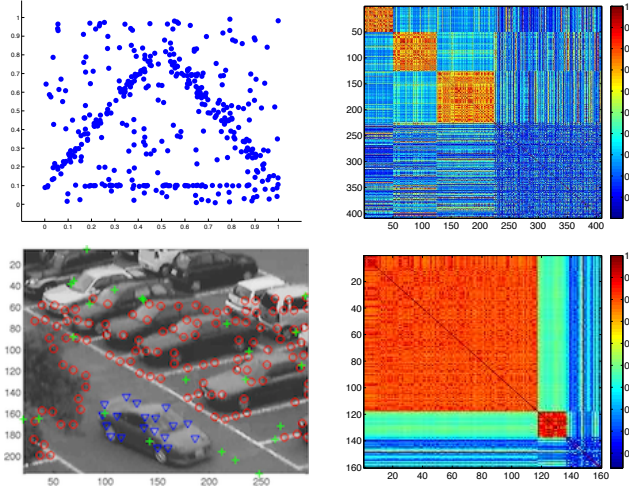
Figure 2. Left: synthetic three-structure line-fitting data with 45% gross outliers (top) and two-structure keypoints on the first image of the "kanatani1" data (bottom) [17] with 15% gross outliers (marked as "+") for 2-view fundamental matrix estimation (Keypoints on the second image are not shown here.). Right: the structures in the data are revealed as high similarity blocks in the data similarity matrix $\boldsymbol{S}$ with $\boldsymbol{S}_{ij} := s(\boldsymbol{x}_i, \boldsymbol{x}_j)$ (9). To aid presentation, we arranged data in $\boldsymbol{S}$ based on their structure membership.
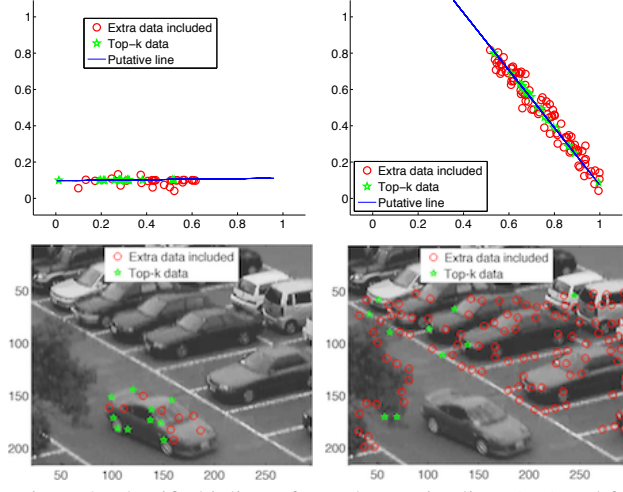

Figure 3. Identified inliers of sample putative lines (top) and fundamental matrices (bottom), each model representing a coherent structure in the data depicted in the top and respectively, bottom left panels of Fig. 2. The inliers found for a model include: (1) a limited number of data that strongly support that model, *i.e.* top-k data (stars) and (2) those detected by (11) (circles) that represent a larger population of data that are also consistent with that model.

resulting data similarity measure is then given by an exponentially decayed sum of (7) for increasing values of t:

$$s(\boldsymbol{x}_i, \boldsymbol{x}_j) := \frac{1}{\Lambda} \sum_{t=1}^{t_{\max}} \lambda^{t-1} s_t(\boldsymbol{x}_i, \boldsymbol{x}_j), \text{ where} \qquad (9)$$

$$t_{\max} := \lceil |\boldsymbol{\pi}^{(i)}| / \min(h^{(i)}, h^{(j)}) \rceil \text{ and } \lambda \in (0, 1).$$

The decay factor λ governs the contributions of intersection measurements (7) taken within different bandwidths; and $\Lambda = \sum_{t=1}^{t_{\max}} \lambda^{t-1}$ is a normalization constant. It is easy to verify that $s(\cdot, \cdot)$ takes on values from $(0, 1]$, and is symmetric in its input variables. Moreover, in Appendix A we prove that (9) is in fact a valid kernel function.

Fig. 2 (right) gives sample outputs of (9) using 10^3 putative models for line fitting (top) and 2-view fundamental matrix estimation (bottom). The evident high similarity blocks in the data similarity matrix correspond to coherent structures in the unbalanced and noisy data (Fig. 2, left).

3.2. Identification of Inliers for Each Model

Having measured pairwise similarities between all data, we use a two-step procedure to identify inliers for each putative model, *i.e.* forming the set $\mathcal{I}_m$ as in (2):

1. First, for each model $\boldsymbol{\theta}_m$, initialize $\mathcal{I}_m$ with indices of k data that give the k-smallest residuals as measured to $\boldsymbol{\theta}_m$, we call these data *top-k data*; k is an estimate of the minimum number of inliers of all putative models. We conservatively set $k = 5\% \times N$ in all experiments;

2. then, expand $\mathcal{I}_m$ with data that are "similar" to the top-k data. Denote the average similarity between a datum $\boldsymbol{x}_i$ and the top-k data of $\boldsymbol{\theta}_m$ by $s_m^{(i)}$; and let s_m^{top} be the average similarity among the top-k data, *i.e.*

$$s_m^{(i)} := \sum_{j \in \mathcal{I}_m^{\text{top}} \setminus \{i\}} \frac{s(\boldsymbol{x}_i, \boldsymbol{x}_j)}{z} \text{ and } s_m^{\text{top}} := \sum_{i \in \mathcal{I}_m^{\text{top}}} \frac{s_m^{(i)}}{k}, \quad (10)$$

where $\mathcal{I}_m^{\text{top}}$ contains the indices of the top-k data of $\boldsymbol{\theta}_m$, and z is a normalization constant: $z = k$ if $i \notin \mathcal{I}_m^{\text{top}}$ and $(k-1)$ otherwise. A datum $\boldsymbol{x}_i$ with $i \notin \mathcal{I}_m^{\text{top}}$ is then regarded similar to the top-k data if

$$s_m^{(i)} / s_m^{\text{top}} \geq \epsilon, \qquad (11)$$

where $\epsilon \in (0, 1]$ is a tolerance parameter. The higher the value of ϵ, the stricter the similarity requirement is for a non-top-k datum to be included into $\mathcal{I}_m$. In all experiments we found it sufficient to set $\epsilon = 0.8$.

Fig. 3 shows the inliers identified by the above procedure (with default k and ϵ) on the data shown in Fig. 2 (left). It can be seen that despite the presence of multiple structures of different sizes, heteroscedastic inlier noise and gross outliers in data, our procedure still effectively identifies the inliers that well characterize a given putative model.

Having obtained the inliers of a model $\boldsymbol{\theta}_m$, we assess its quality by jointly taking into account the fidelity of $\boldsymbol{\theta}_m$ to the identified inliers as well as the similarity among these inliers. It is straightforward to quantify the model fidelity

via (1); using (9), we compute the inlier similarity as the average median similarities between inliers in $\mathcal{I}_m$:

$$f(\mathcal{I}_m) := \frac{1}{|\mathcal{I}_m|} \sum_{i \in \mathcal{I}_m} \text{median}_{j \in \mathcal{I}_m} s(\boldsymbol{x}_i, \boldsymbol{x}_j). \qquad (12)$$

The higher the value of $f(\mathcal{I}_m)$, the more likely the data in $\mathcal{I}_m$ are from a genuine structure, and thus the more promising $\boldsymbol{\theta}_m$ is to be a "good" model. Note that unlike the standard RANSAC, our method does not consider the number of inliers as a measure of model quality. Therefore, it is not contingent on the identification of all inliers.

4. Regularization of Overlapped Models

It is very common in practice to have several similar putative models that essentially represent the same structure in data, but differ slightly in their parameters. To resolve such an ambiguity, we include a model overlapping regularizer in (2) to penalize overlaps between top-ranked models.

Our regularization makes use of a model similarity measure which is designed by the same principle as introduced in Sec. 3.1 for the computation of data similarity: model similarities are quantified by their preferences to the data. Such preferences are collectively determined by residuals of the data as measured to a given model as well as their similarities to the top-k data of that model:

$$c_i^{(m)} := r_m^{(i)} - \alpha s_m^{(i)}, \qquad (13)$$

where all the quantities defining the consistency $c_i^{(m)}$ of $\boldsymbol{x}_i$ with a model m are readily available from the operations discussed in previous sections: $\alpha > 0$ is a trade-off parameter used in (2), all the other quantities are from Sec. 3. Since the smaller the value of $c_i^{(m)}$, the more consistent $\boldsymbol{x}_i$ is with a model m, we sort $c_i^{(m)}$, for $i = 1, \cdots, N$, in non-descending order to obtain the preference of $\boldsymbol{\theta}_m$ to the N input data to obtain the permutation $\boldsymbol{\tau}^{(m)} := [\tau_1^{(m)}, \cdots, \tau_N^{(m)}]$ such that $p < q \Rightarrow r_m^{(\tau_p^{(m)})} \leq r_m^{(\tau_q^{(m)})}$, where the residuals $r_m^{(\tau_p^{(m)})}$ are indexed as in (5). Comparing the similarity between two models, *i.e.* computing $s(\boldsymbol{\theta}_m, \boldsymbol{\theta}_n)$ in (2), then involves applying the similarity measure (9) on permutations given by $\boldsymbol{\tau}^{(m)}$ and $\boldsymbol{\tau}^{(n)}$. Since each $\boldsymbol{\theta}_m$ is best represented by the data in its associated inlier set $\mathcal{I}_m$, we set its basis bandwidth $h^{(m)}$ to $|\mathcal{I}_m|$, *i.e.* the estimated number of its inliers.

Let $\boldsymbol{S} \in (0, 1]^{M \times M}$ be the model similarity matrix with

$$\boldsymbol{S}_{mn} := s(\boldsymbol{\theta}_m, \boldsymbol{\theta}_n). \qquad (14)$$

By Theorem A.1, $\boldsymbol{S} \succeq 0$. This ensures that the objective (2) is convex, hence allowing for global optimization. Note that each diagonal element of $\boldsymbol{S}$ takes on the maximal value of

one. The form of the regularizer in (2), however, suggests that an $\boldsymbol{S}$ with zeros on its diagonal would further encourage the concentration of weights on distinct models. Unfortunately, this makes the objective non-convex. Without losing the convexity, we added to $\boldsymbol{S}$ a non-negative diagonal penalty matrix $\boldsymbol{D}$ to discourage over 50% overlaps between top-ranked models, thus further ensuring model diversity. Alg. 2 gives a simple tree-traversal-based scheme that generates such a $\boldsymbol{D}$. Note that $\boldsymbol{S} \succeq 0$ and $\boldsymbol{D} \succeq 0$ imply $(\boldsymbol{S} + \boldsymbol{D}) \succeq 0$, hence the convexity of (2) remains.

5. Experiments

We evaluated the performance of the proposed method (QP-MF, Alg. 1) on various model fitting problems on both synthetic and real data. Our method was compared against two recent optimization approaches[2] to multi-model fitting: the energy-based approach [6] (denoted Energy) and the approximate UFL approach FLoSS [8]. For all experiments, we generated 10^3 putative models by using a guided sampling method [4] that is recently developed for efficient hypothesis generation on multi-structure data.

We tuned the free parameters of Energy and FLoSS to ensure their good performance; specifically, we use the ground-truth knowledge about the number of structures in data to guide the tuning of their regularization parameters. As suggested in [6], we also introduced an outlier model into the optimization of our competing methods for the experiments that involve gross outliers (Secs. 5.1 and 5.2). The residuals associated with this pseudo-model were tuned to 0.01. Such a scheme was designed to handle outliers.

For QP-MF, we fixed all its parameters throughout as follows: The decay factor λ in (9) was set to 0.5; the basis bandwidth $h^{(i)}$ (8) for our model and data similarity measures was set to $|\mathcal{I}_m|$ and respectively, the number of models that a datum is consistent with. The consistency is determined by the following check

$$r_m^{(i)} \leq r_{\text{inlier}}, \qquad (15)$$

where $r_{\text{inlier}} \geq 0$ is an estimate of inlier residual, we set it to the maximum residual among the residuals that are ranked not lower than $2p$ by each model (*i.e.* the $2p$ smallest residuals of each model), p being the size of a minimum subset of data required to estimate a particular geometric model, *e.g.* $p = 4$ for homography estimation. Using more sophisticated methods for estimating r_{inlier} (*e.g.* [16, 18]) may improve the performance of QP-MF. Both α and β in (2) were set to the average residual of all data as measured to all putative models. This ensures that the three terms in (2): model fidelity, inlier similarity, and regularizer, are on the same scale. The lower bound T on the number of active models was set to 2, meaning that at least 2 putative models

---

[2]Code available from the respective authors' webpages.

Algorithm 2 Generate a Diagonal Overlapping Penalty Matrix

1: **input** $L(\mathcal{I}_m, \boldsymbol{\theta}_m)$ and $f(\mathcal{I}_m)$, $\forall m$ and α as in (2), the model similarity matrix $\boldsymbol{S}$ (14), and a penalty $\gamma > 0$

2: **output** a diagonal penalty matrix $\boldsymbol{D} \in \mathbb{R}^{M \times M}$

3: compute model quality scores: $q_m = L(\mathcal{I}_m, \boldsymbol{\theta}_m) - \alpha f(\mathcal{I}_m)$ for all m

4: **for** $m = 1, \cdots, M$ **do**

5:    **if** exist $\boldsymbol{\theta}_n \neq \boldsymbol{\theta}_m$ with $q_n \geq q_m$ and $\boldsymbol{S}_{mn} \geq 0.5$ **then**

6:       link $\boldsymbol{\theta}_m$ to $\boldsymbol{\theta}_n$: $\boldsymbol{\theta}_m \rightarrow \boldsymbol{\theta}_n$

7:    **else**

8:       link $\boldsymbol{\theta}_m$ to itself: $\boldsymbol{\theta}_m \rightarrow \boldsymbol{\theta}_m$

9:    **end if**

10: **end for**

11: perform tree traversals to associate each $\boldsymbol{\theta}_m$ with its corresponding root node $\boldsymbol{\theta}_{r_m}$

12: set $\boldsymbol{D}_{mm} = \gamma \boldsymbol{S}_{mr_m}$ if $r_m \neq m$; $\boldsymbol{D}_{mm} = 0$ otherwise

---

will be assigned non-zero weights. To impose a penalty on over 50% model overlap, we generated a diagonal penalty matrix $\boldsymbol{D}$ via Alg. 2 with the penalty constant $\gamma > 0$ set to M (Such a choice of γ is not essential.). The matrix $\boldsymbol{D}$ was then added to the similarity matrix (14) at Step 6 of Alg. 1. For the optimization, we used the QP solver in MOSEK.

In terms of speed, Energy requires the least CPU time, costing less than 0.5 second in most of our experiments on a machine with 2.6GHz Intel quad core processors with 4GB of RAM. Our method takes about 2 seconds to solve its QP, while its overall running time is typically between 10 to 25 seconds, most of which were spent in computing the data and model similarity matrices (Steps 3 and 6 in Alg. 1). Such a speed is comparable to FLoSS, but is slower than Energy. However, in return, QP-MF provides globally optimal solutions (compared to the approximate solutions given by the other two methods), which eliminates potential problems caused by the inaccuracy of approximation optimization. Moreover, since the elements of the data (resp. model) similarity matrix (9) can be computed independently, it is easy to devise a parallel implementation to accelerate QP-MF for large-scale problems.

5.1. Homography Estimation

Our first set of experiments were conducted on synthetic keypoint correspondences for planar homography estimation. We used the "spinning wheels" data obtained from the web.[3] The data contain 5 frames of tracked keypoints on 4 rotating planar objects, each having 50 inliers (Fig. 4, left). We obtained keypoint correspondences by matching the keypoints on the first and the last frames. Each putative homography was computed from 4 correspondences via Di-

---

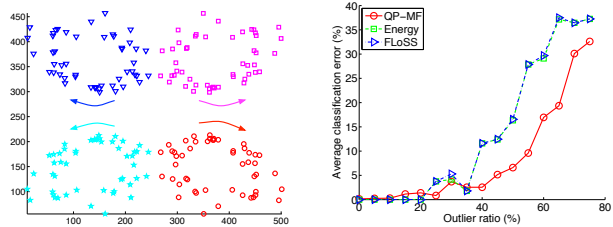[3]http://www.iu.tu-darmstadt.de/datasets.

Figure 4. Average classification error under the influence of various outliers ratios (right) on the "spinning wheels" data (left).
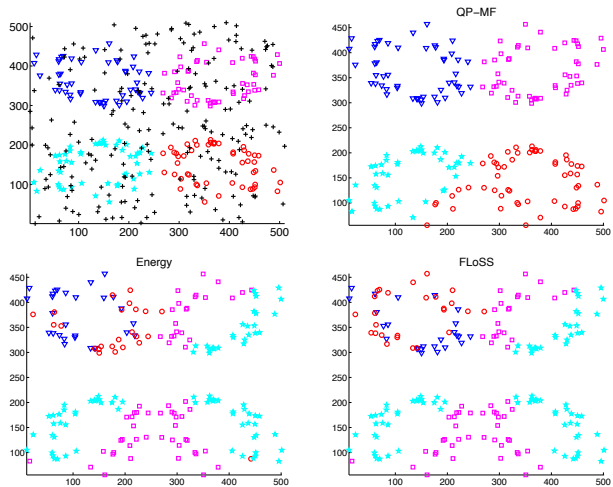

Figure 5. Sample classification results on the corrupted "spinning wheels" data (top left) with 50% gross outliers (marked as "+").

rect Linear Transformation [DLT, 7]. Residuals were measured as the Sampson distances.

We investigated the perfomance of each method under the influence of various gross outlier ratios (0-75%); false correspondences were established by matching keypoints that were randomly generated over the range of inliers. We recorded the average classification error of each method on the inliers across 10 random runs, each run used a new batch of 10^3 putative models generated by our chosen sampling method [4]. For Energy and FLoSS, each datum was assigned to its most consistent (as measured by residual) *non-outlier* model; recall that our parameter tuning ensures that Energy and FLoSS return exactly 4 non-outlier models for this task, while for our method, each datum was assigned to its most consistent model among the returned top-4 models with the consistency quantified by (13).

Fig. 4 (right) shows the minimum inlier classification error among all possible outcomes given by different permutations of data labels. We can see that QP-MF tolerates high percentages of gross outliers, maintaining a less than 10% classification error for up to 55% outliers (inlier rate of only 11.25% for each spinning wheel), whereas the performance of Energy and FLoSS deteriorates rapidly as the outlier ratio surpasses 35%, resulting in noticeably higher classification errors than our method.

It can be seen from the sample classification results in

Figure 6. The "cars1", "cars2", and "three-cars" data with 10% gross outliers (marked as "+") for fundamental matrix estimation.
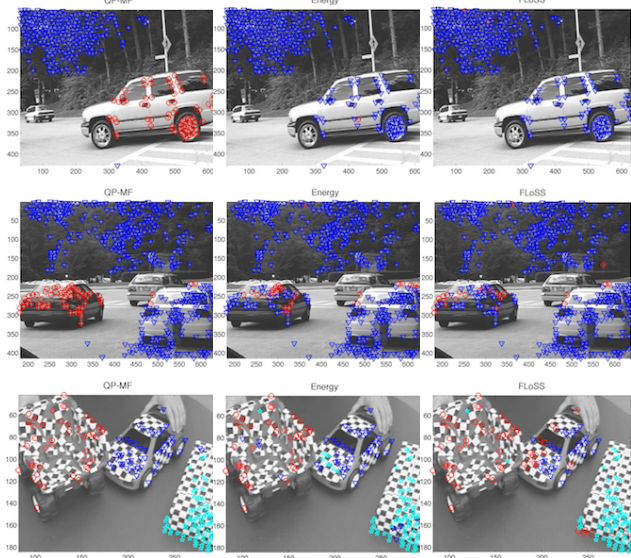

Figure 7. The classification results of QP-MF (left), Energy (center), and FLoSS (right) on the inlying data shown in Fig. 6.

Fig. 5 that even in the presence of a large amount of gross outliers (top left), our method (top right) still correctly clusters the majority of the inlying data into 4 classes (denoted by 4 different markers), while under such noise, the performance of the other two methods appears almost random (bottom row). This set of experiments indicate that model quality measure by residuals alone, as in FLoSS, is susceptible to outliers; and the spatial smooth constraint enforced by Energy can be misleading if gross outliers spatially mix with inliers. In fact, we had to significantly tune down the contribution of the smooth cost of Energy to ensure its good performance; however, in this case it essentially reduced to FLoSS, which explains their similar performance in Fig. 4. QP-MF's quality measure, *i.e.* residual complemented with inlier similarity, is designed to alleviate these problems.

5.2. Fundamental Matrix Estimation

We also applied our method to 2-view fundamental matrix estimation on images of multiple moving objects from the Hopkins155 benchmark [17]. We used the first and the last frames of the tested video sequences to form image pairs. Keypoint correspondences were obtained by matching keypoints on each image pair. The standard 8-point estimation method [7] was used to generate putative funda-

Table 1. Average and median classification error (%) on the Hopkins155 sequences of 2 objects.

	#Seq.	Error	FLoSS	Energy	QP-MF
Checker.	78	Avg.	7.70	**5.28**	9.98
		Med.	**1.23**	1.83	1.38
Traffic	31	Avg.	0.14	1.15	**0.12**
		Med.	0.00	0.00	**0.00**
Articul.	11	Avg.	4.69	3.30	**2.38**
		Med.	1.30	1.22	**0.00**

Table 2. Average and median classification error (%) on the Hopkins155 sequences of 3 objects.

	#Seq.	Error	FLoSS	Energy	QP-MF
Checker.	26	Avg.	16.45	21.38	**15.61**
		Med.	16.79	21.14	**8.82**
Traffic	7	Avg.	**0.29**	11.19	0.29
		Med.	**0.00**	14.02	0.00
Articul.	2	Avg.	8.51	13.04	**5.85**
		Med.	8.51	13.04	**5.85**

mental matrices from sets of 8 correspondences. Residuals were measured as the Sampson distances. We added 10% gross outliers to the original noise-free Hopkins155 data to create a more realistic test setting. Fig. 6 shows the data.

As in the previous set of experiments, we evaluated the classification error of each method on the inliers of each data. Fig. 7 shows that in all experiments the fundamental matrices ranked high by QP-MF consistently lead to the best classification results than that of the other two methods.

5.3. Subspace Segmentation

Our final set of experiments consider the task of identifying multiple moving objects via subspace segmentation. In this case each datum x_i is the trajectory of a keypoint tracked over several frames. Assuming an affine projection model, point trajectories of each moving object lie in a linear subspace of dimensionality at most 4. The task of subspace segmentation is to cluster point trajectories into different linear subspaces. We used the entire 155 video sequences in the Hopkins155 benchmark [17] in our experiments, and followed [17] in first projecting point trajectories onto a subspace of dimensionality 5; putative 4D subspace models were then generated from sets of 4 projected trajectories. Residuals were calculated as the squared normal distances between data and subspaces. We evaluated the classification errors of all methods on three categories of the Hopkins155 data: "checkerboard", "traffic", and "articulated objects", respectively. Again, Energy and FLoSS require the tuning of their regularization parameters in order to output the correct number S of subspaces; in addition, FLoSS invokes a subspace merging scheme if too many subspace models are selected. As reported in [8], FLoSS with such a scheme produces competitive results (though the requirement of ground-truth knowledge about the num-

ber of motions renders it impractical). Since the code for the merging scheme of FLoSS is not released, we took the results in [8] for our comparison. For QP-MF, we simply used its top-S models for the evaluation.

Tabs. 1 and 2 summarize the performance of the three methods on the Hopkins155 data of 2 and 3 objects, respectively. We can see that QP-MF outperforms its competing methods on almost all sequences except for the 2-object "checkerboard" sequences where Energy performs the best in terms of the average classification error, while FLoSS slightly outperforms others in terms of the median error.

6. Conclusions

We proposed a novel QP-based global optimization approach that directly optimizes the ranking of putative models such that top-ranked ones represent a possibly unknown number of structures in data. We also introduced a data (resp. model) similarity measure that is robust to noise and effective in its own right in quantifying the correlations between data (resp. models), which enables its use in other robust fitting methods. Moreover, the proposed approach is the first that integrates both data and model similarities into a unified optimization framework for multi-model fitting, and provides superior results over recent optimization approaches on a range of computer vision applications.

Acknowledgements

References

[1] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2041

[2] H. Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974. 2041

[3] T.-J. Chin, H. Wang, and D. Suter. Robust fitting of multiple structures: The statistical learning approach. In *ICCV*, Kyoto, Japan, 2009. 2042, 2043

[4] T.-J. Chin, J. Yu, and D. Suter. Accelerated hypothesis generation for multi-structure robust fitting. In *ECCV*, 2010. 2042, 2043, 2045, 2046

[5] O. Chum, J. Matas, and J. Kittler. Locally optimized RANSAC. In *Pattern Recognition: 25th DAGM Symposium*, pages 236–243. Springer, 2003. 2041

[6] A. Delong, A. Osokin, H. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. In *CVPR*, 2010. 2041, 2042, 2045

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 2042, 2046, 2047

[8] N. Lazic, I. Givoni, B. Frey, and P. Aarabi. FloSS: Facility location for subspace segmentation. In *ICCV*, 2009. 2041, 2045, 2047, 2048

[9] H. Li. Two-view motion segmentation from linear programming relaxation. In *CVPR*, 2007. 2041

[10] D. Nistér. Preemptive RANSAC for live structure and motion estimation. In *ICCV*, 2003. 2041

[11] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, 1999. 2043

[12] B. Scherer. *Portfolio construction and risk budgeting*. Risk Books, London, 2002. 2042

[13] B. Schölkopf and A. J. Smola. *Learning with kernels*. MIT Press, 2002. 2043, 2048

[14] N. Thakoor and J. Gao. Branch-and-bound hypothesis selection for two-view multiple structure and motion segmentation. In *CVPR*, 2008. 2041

[15] R. Toldo and A. Fusiello. Robust multiple structures estimation with j-linkage. In *ECCV*, 2008. 2042, 2043

[16] R. Toldo and A. Fusiello. Automatic estimation of the inlier threshold in robust multiple structures fitting. In *ICIAP*, 2009. 2045

[17] R. Tron and R. Vidal. A benchmark for the comparison of 3-D motion segmentation algorithms. In *CVPR*, 2007. 2041, 2042, 2044, 2047

[18] H. Wang and D. Suter. MDPE: A very robust estimator for model fitting and range image segmentation. *International Journal of Computer Vision*, 59(2):139–166, 2004. 2045

[19] M. Zuliani, C. Kenney, and B. S. Manjunath. The multi-RANSAC algorithm and its application to detect planar homographies. In *ICIP*, 2005. 2041

A. A Positive-Definite Similarity Measure

Theorem A.1 *The similarity measure $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined as* (9) *is a positive-definite kernel on $\mathcal{X}$.*

Proof Since $s(\cdot, \cdot)$ is a positive combination of $s_t(\cdot, \cdot)$ (7) for various t, it suffices to show that $s_t(\cdot, \cdot)$ is a positive-definite kernel. For $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathcal{X}, \forall n \in \mathbb{N}_+$, let $\boldsymbol{S} \in \mathbb{R}^{n \times n}$ be a square matrix with $\boldsymbol{S}_{ij} = s_t(\boldsymbol{x}_i, \boldsymbol{x}_j)$, then $s_t(\cdot, \cdot)$ is a positive-definite kernel if $\boldsymbol{S} \succeq 0$ [13]. We prove this by showing that $\boldsymbol{S}$ can be decomposed as $\boldsymbol{S} = \boldsymbol{A}^\top \boldsymbol{A}$, a form of matrix decomposition that guarantees $\boldsymbol{S} \succeq 0$.

First, for each permutation $\boldsymbol{\pi}^{(i)}$ (6) of length M, we define a column vector $\boldsymbol{a}^{(i)} \in \{0, 1\}^M$ with its jth element set to 1 if j is contained in the sub-permutation $\boldsymbol{\pi}^{(i)}[h_t^{(i)}]$, *i.e.* $\boldsymbol{a}^{(i)}$ is the bitvector representation of $\boldsymbol{\pi}^{(i)}[h_t^{(i)}]$. It follows from (7) that $s_t(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{a}^{(i)}, \boldsymbol{a}^{(j)} \rangle / (\|\boldsymbol{a}^{(i)}\| \|\boldsymbol{a}^{(j)}\|)$. We prove our claim by constructing the $\boldsymbol{A}$ matrix as $\boldsymbol{A} := [\boldsymbol{a}^{(1)}/\|\boldsymbol{a}^{(1)}\|, \cdots, \boldsymbol{a}^{(n)}/\|\boldsymbol{a}^{(n)}\|] \in \mathbb{R}^{M \times n}$. ∎