A Hybrid Quantum-Classical Algorithm for Robust Fitting

Anh-Dzung Doan [1]   Michele Sasdelli [1]   David Suter [2]   Tat-Jun Chin [1]

[1] School of Computer Science, The University of Adelaide, South Australia
[2] Centre for AI&ML, School of Science, Edith Cowan University, Western Australia

Abstract

Fitting geometric models onto outlier contaminated data is provably intractable. Many computer vision systems rely on random sampling heuristics to solve robust fitting, which do not provide optimality guarantees and error bounds. It is therefore critical to develop novel approaches that can bridge the gap between exact solutions that are costly, and fast heuristics that offer no quality assurances. In this paper, we propose a hybrid quantum-classical algorithm for robust fitting. Our core contribution is a novel robust fitting formulation that solves a sequence of integer programs and terminates with a global solution or an error bound. The combinatorial subproblems are amenable to a quantum annealer, which helps to tighten the bound efficiently. While our usage of quantum computing does not surmount the fundamental intractability of robust fitting, by providing error bounds our algorithm is a practical improvement over randomised heuristics. Moreover, our work represents a concrete application of quantum computing in computer vision. We present results obtained using an actual quantum computer (D-Wave Advantage) and via simulation[1].

1. Introduction

Imperfections in sensing and processing in computer vision inevitably generate data that contain outliers. Therefore, it is necessary for vision pipelines to be robust against outliers in order to mitigate their harmful effects.

In 3D vision, where a major goal is to recover the scene structure and camera motion, a basic task is to fit a geometric model onto noisy and outlier prone measurements. This is often achieved through the consensus maximisation framework [18]: given N data points $\mathcal{D} = \{\mathbf{p}_i\}_{i=1}^N$ and a target geometric model parametrised by $\mathbf{x} \in \mathbb{R}^d$, let P_N be the power set of index set $\{1, \ldots, N\}$. We aim to solve

$$\max_{\mathcal{I} \in P_N, \, \mathbf{x} \in \mathbb{R}^d} \quad |\mathcal{I}| \qquad (1)$$
$$\text{s.t.} \quad r_i(\mathbf{x}) \leq \epsilon \quad \forall i \in \mathcal{I},$$

where $r_i(\mathbf{x})$ is the residual of point $\mathbf{p}_i$ w.r.t. $\mathbf{x}$, and ϵ is a given inlier threshold. The form of $r_i(\mathbf{x})$ depends on the specific geometric model (more details in Sec. 3). A candidate solution $(\mathcal{I}, \mathbf{x})$ consists of a consensus set $\mathcal{I}$ and its "witness" (an estimate) $\mathbf{x}$, where the points in $\mathcal{I}$ are the inliers of $\mathbf{x}$. Problem (1) seeks the maximum consensus set $\mathcal{I}^*$, whose witness $\mathbf{x}^*$ is a robust estimate of the model[2].

Many computer vision systems employ random sampling heuristics, i.e., RANSAC [32] and its variants (e.g., [5, 6, 20, 57, 68, 69]), for consensus maximisation. The basic idea is to repeatedly fit the model on randomly sampled minimal subsets of $\mathcal{D}$, and return the $\tilde{\mathbf{x}}$ with the largest consensus set $\tilde{\mathcal{I}}$. Such heuristics can only approximate (1) and generally do not provide optimality guarantees or error characterisation, e.g., a tight bound on the discrepancy $|\mathcal{I}^*| - |\tilde{\mathcal{I}}|$. Moreover, $\tilde{\mathbf{x}}$ is subject to randomness, and postprocessing or reruns are often executed to vet the result.

Unfortunately, consensus maximisation is provably intractable [4, 16], hence there is little hope in finding efficient algorithms that can solve it exactly. While there has been active research into globally optimal algorithms [13, 17, 44, 54, 55], such techniques are realistic only for small input instances (small d, N and/or number of outliers [17]).

Bridging the gap between exact algorithms that are costly and randomised heuristics that offer no quality assurances is an important research direction in robust fitting with practical ramifications. Towards this aim, deterministic approximate algorithms [14, 41, 42, 59, 73] eschew exhaustive search (e.g., branch-and-bound) and randomisation, and instead adopt deterministic subroutines such as convex optimisation, proximal splitting, etc. These methods avoid the vagaries of random sampling, and some can even guarantee convergence [41, 42, 59]. However, none of them provide error bounds. Indeed, complexity results [4, 16] also preclude efficient approximate solutions with error bounds.

Partly buoyed by the dominance of deep learning in computer vision, learning-based solutions to robust geometric fitting have been developed [10, 60, 70]. Such techniques leverage statistics in large datasets to learn a mapping from

---

[1]Source code: https://github.com/dadung/HQC-robust-fitting

[2]This also depends on using a correct ϵ. The large volume of works that apply consensus maximisation suggest setting ϵ is usually not a concern.

the input instance to the desired solution. Despite showing promising results in benchmark datasets, learning methods do not provide optimality guarantees and error bounds. Whether the learned model can generalise is also a concern.

To summarise, existing algorithms for robust fitting, particularly those targeted at consensus maximisation, have yet to satisfactorily solve the problem. It is thus worthwhile to investigate novel approaches based on new insights.

Our contributions We propose a new approach that leverages quantum computing for consensus maximisation. Our core contribution is a consensus maximisation algorithm that iteratively solves a sequence of integer programs and terminates with either $\mathbf{x}^*$ or a suboptimal solution $\tilde{\mathbf{x}}$ with a known error bound $|\mathcal{I}^*| - |\tilde{\mathcal{I}}| \leq \rho$. The integer programs are amenable to a quantum annealer [64, Chap. 8], which is utilised to tighten the bound efficiently. Since our method employs convex subroutines and random sampling, it is a *hybrid quantum-classical* algorithm [11, 36, 40, 56].

We will present results using an actual quantum computer, the D-Wave Advantage [23], as well as simulation. While our technique does not yet outperform state-of-the-art algorithms, in part due to the limitations of current quantum technology, our work represents a concrete application of quantum computing in computer vision. We hope to inspire future efforts on this topic in the community.

2. Related work

In Sec. 1, we have provided an overview of robust fitting and recent algorithmic advances. We thus focus our survey on quantum computing in computer vision.

Many quantum methods have been proposed for image processing [15, 27, 71, 74], image recognition [26, 50, 51], and object detection [45]. Also, several methods explored the tasks of classification and training a deep neural network [39, 52, 63]. Recently, Golyanik and Theobalt [34] proposed a practical quantum algorithm for rotation estimation to align two point sets. Their basic idea is to discretise rotation matrices to formulate the problem to the quadratic unconstrained binary optimization (QUBO), which can be solved by quantum annealers. Benkner et al. [7] proposed to solve the graph matching problem through formulating the quadratic assignment problem (QAP) to the QUBO using the penalty approach. They conducted the experiment and provided analysis on the quantum computer D-Wave 2000Q. However, the limitation of quantum computers precluded them from solving large problems. To address this issue, instead of enforcing a penalty to QAP, Q-Match [65] was proposed to iteratively select and solve subproblems of QAP, which allows D-Wave annealers to efficiently deal with large problems. Another interesting work is QuantumSync [8], which addresses the synchronisation problem in the context of multi-image matching. This work carefully formulated the synchronisation problem to the QUBO,

which was then validated on D-Wave Advantage.

The closest work to ours is [19], who proposed a quantum solution for robust fitting. However, there are non-trivial differences: first, [19] estimates *per-point influences* (a measure of outlyingness) [66, 67] for outlier removal instead of consensus maximisation. Second, their algorithm is based on the gate computing model, which is fundamentally different from the quantum annealing approach adopted in our work. Third, the results in [19] are only based on simulation; we will compare against [19] on this basis in Sec. 6.

3. Reformulating consensus maximisation

In this section, we describe our novel reformulation for consensus maximisation and relevant theoretical results, before presenting the usage of quantum annealing in Sec. 4 and the overall algorithm in Sec. 5.

3.1. Preliminaries

Following [19], we consider residuals $r_i(\mathbf{x})$ that are quasiconvex, which encapsulates many geometric models of interest in computer vision [38]. Formally, if the set

$$\{\mathbf{x} \in \mathbb{R}^d \mid r_i(\mathbf{x}) \leq \alpha\} \tag{2}$$

is convex for all $\alpha \geq 0$, then $r_i(\mathbf{x})$ is quasiconvex. Note that assuming quasiconvex residuals does not reduce the computational hardness of consensus maximisation [16].

For $\mathcal{C} \in P_N$, define the minimax problem

$$g(\mathcal{C}) = \min_{\mathbf{x} \in \mathbb{R}^d} \max_{i \in \mathcal{C}} r_i(\mathbf{x}). \tag{3}$$

For quasiconvex $r_i(\mathbf{x})$, (3) is a quasiconvex program [3, 30], which is polynomial-time solvable. Note that $g(\mathcal{C}) \leq \epsilon$ implies that $\mathcal{C}$ is a consensus set, since all the points in $\mathcal{C}$ are within error ϵ to the extremiser of (3).

Define the "feasibility test"

$$f(\mathcal{C}) = \begin{cases} 0 & \text{if } g(\mathcal{C}) \leq \epsilon; \\ 1 & \text{otherwise.} \end{cases} \tag{4}$$

Any $\mathcal{C}$ such that $f(\mathcal{C}) = 0$ implies that $\mathcal{C}$ is a consensus set. Problem (1) can thus be restated as

$$\max_{\mathcal{I} \in P_N} \quad |\mathcal{I}|, \qquad \text{s.t.} \quad f(\mathcal{I}) = 0, \tag{5}$$

with the witness $\mathbf{x}$ for any feasible $\mathcal{I}$ obtainable through computing $g(\mathcal{I})$ to evaluate $f(\mathcal{I})$.

Given a consensus set $\mathcal{I}$ with witness $\mathbf{x}$, the points in the complement $\mathcal{O} = \{1, \ldots, N\} \setminus \mathcal{I}$ are the outliers to $\mathbf{x}$. The "dual" of problem (5) is therefore

$$\min_{\mathcal{O} \in P_N} \quad |\mathcal{O}|, \qquad \text{s.t.} \quad f(\{1, \ldots, N\} \setminus \mathcal{O}) = 0, \tag{6}$$

i.e., find the model with the least number of outliers.

Definition 1 (True inliers and true outliers). *Let $\mathcal{I}^*$ be the maximum consensus set and $\mathcal{O}^* = \{1, \ldots, N\} \setminus \mathcal{I}^*$. We call $\mathcal{I}^*$ the "true inliers" and $\mathcal{O}^*$ the "true outliers".*

Property 1 (Monotonicity). *For* (3) *with quasiconvex residuals, given subsets $\mathcal{P}, \mathcal{Q}, \mathcal{R} \in P_N$ with $\mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{R}$, we have $g(\mathcal{P}) \leq g(\mathcal{Q}) \leq g(\mathcal{R})$. By extension, we also have that $f(\mathcal{P}) \leq f(\mathcal{Q}) \leq f(\mathcal{R})$. See [3, 30] for more details.*

Intuitively, adding points to a feasible subset can only potentially make it infeasible; the converse cannot be true. This leads to the following crucial concept.

Definition 2 (Basis). *A basis $\mathcal{B} \subset \{1, \ldots, N\}$ is a subset such that $g(\mathcal{B}') < g(\mathcal{B})$ for every $\mathcal{B}' \subset \mathcal{B}$.*

Intuitively, removing any point from a basis $\mathcal{B}$ will cause the minimax value of the subset to shrink.

Property 2 (Combinatorial dimension). *The combinatorial dimension δ of minimax problem* (3) *is the upper bound on the size of bases [3, 30]. For quasiconvex $r_i(\mathbf{x})$, $\delta = 2d+1$.*

Claim 1. *If basis $\mathcal{B}$ is infeasible, i.e., $f(\mathcal{B}) = 1$, then $|\mathcal{B} \cap \mathcal{O}^*| \geq 1$, i.e., an infeasible basis $\mathcal{B}$ contains at least one true outlier.*

Proof. See Sec. A in supplementary material. □

3.2. Hypergraph vertex cover

Define the binary N-vector

$$\mathbf{z} = [z_1, \ldots, z_N] \in \{0,1\}^N, \tag{7}$$

where the set of indices corresponding to nonzero z_i's are

$$\mathcal{C}_\mathbf{z} = \{i \in \{1, \ldots, N\} \mid z_i = 1\}. \tag{8}$$

The outlier minimisation problem (6) can be reexpressed as

$$\min_{\mathbf{z} \in \{0,1\}^N} \quad \|\mathbf{z}\|_1, \quad \text{s.t.} \ f(\{1, \ldots, N\} \setminus \mathcal{C}_\mathbf{z}) = 0, \tag{9}$$

where $z_i = 1$ implies the i-th point is removed as an outlier.

Let $\{\mathbf{b}_{(k)}\}_{k=1}^K$ be K binary N-vectors that correspond to *all infeasible bases* of the problem, i.e., for each k,

$$f(\mathcal{C}_{\mathbf{b}_{(k)}}) = 1, \quad \|\mathbf{b}_{(k)}\|_1 \leq \delta, \tag{10}$$

where the latter appeals to the combinatorial dimension (Property 2). Also, the number of infeasible bases $K = O(N^\delta)$. Define the hypergraph H with vertex set $\mathcal{V}$ and hyperedge set E respectively as

$$H = \{\mathcal{V}, E\}, \quad \mathcal{V} = \{1, \ldots, N\}, \quad E = \{\mathcal{C}_{\mathbf{b}_{(k)}}\}_{k=1}^K. \tag{11}$$

Recall that hypergraphs are a generalisation of graphs, where a hyperedge can be incident with more than two vertices [2]. In our hypergraph (11), each hyperedge connects vertices that form an infeasible basis; see Fig. 1.

Claim 2. *A subset $\mathcal{I} \subseteq \mathcal{V}$ is a consensus set iff it is an independent set of hypergraph H.*

Proof. See Sec. B in supplementary material. □

Claim 2 proves that finding the maximum consensus set $\mathcal{I}^*$ is equivalent to finding the maximum independent set of H. Since the complement of an independent set is a vertex cover, it justifies to minimise the vertex cover

$$\begin{aligned} \min_{\mathbf{z} \in \{0,1\}^N} \quad & \|\mathbf{z}\|_1 \\ \text{s.t.} \quad & \mathbf{b}_{(k)}^T \mathbf{z} \geq 1, \quad \forall k = 1, \ldots, K, \end{aligned} \tag{VC}$$

which is an 0-1 integer linear program (ILP). Setting $z_i = 1$ implies removing the i-th vertex, and the constraints ensure that all hyperedges are "covered", i.e., at least one vertex in each hyperedge is removed (cf. Claim 1).

The hypergraph formalism has been applied previously in geometric fitting [2, 46, 47, 58]. However, the target problem in [2, 46, 47, 58] was higher order clustering (e.g., via hypergraph cuts), which is very distinct from our aims.

Formulation (VC) is impractical for two reasons:

- Hypergraph vertex cover is intractable;
- The number of hyperedges in H is exponential.

However, the form (VC) is amenable to a quantum annealer, as we will show in Sec. 4. To deal with the number of hyperedges, we propose a hybrid quantum-classical algorithm in Sec. 5 that incrementally generates hyperedges.

4. Quantum solution

We first provide a basic introduction to quantum annealing, before describing our quantum treatment of (VC).

4.1. Quantum annealing

A quantum annealer solves optimisation problems through energy minimisation of a physical system. A Hamiltonian defines the energy profile of a quantum system, which is composed of a number of interacting qubits. The system's state is initialised at the lowest energy of the initial Hamiltonian and annealed such that the its final state gives the desired solution. At the end of the annealing, the Hamiltonian can be obtained from the following model

$$\sum_n Q_{nn} q_n + \sum_{n<m} Q_{nm} q_n q_m = \mathbf{q}^T \mathbf{Q} \mathbf{q}. \tag{12}$$

The measurement collapses the N-qubit quantum state into $\mathbf{q} = [q_1, q_2, \ldots, q_N]$, where $q_n \in \{0, 1\}$, $\mathbf{Q} \in \mathbb{R}^{N \times N}$. The elements of $\mathbf{Q}$ define the couplings between qubits and their biases; see [64, Chap. 8] for more details.

A quantum annealer solves a problem of the form

$$\min_{\mathbf{q} \in \{0,1\}^N} \mathbf{q}^T \mathbf{Q} \mathbf{q}, \tag{13}$$

(a) Data points with index set $\{1,\ldots,12\}$. Infeasible bases $\bigcirc = \{1,5,6\}$, $\bigcirc = \{2,7,8\}$, $\bigcirc = \{3,9,10\}$, $\bigcirc = \{4,11,12\}$, $\bigcirc = \{1,2,3\}$, $\bigcirc = \{2,3,4\}$. Outlier set $\mathcal{O} = \{1,2,3,4\}$. Consensus set $\mathcal{I} = \{5,6,7,8,9,10,11,12\}$.

(b) Hypergraph with vertex set $\mathcal{V} = \{1,\ldots,12\}$. Hyperedges $\square = \{1,5,6\}$, $\square = \{2,7,8\}$, $\square = \{3,9,10\}$, $\square = \{4,11,12\}$, $\square = \{1,2,3\}$, $\square = \{2,3,4\}$. Vertex cover $= \{1,2,3,4\}$. Independent set $= \{5,6,7,8,9,10,11,12\}$
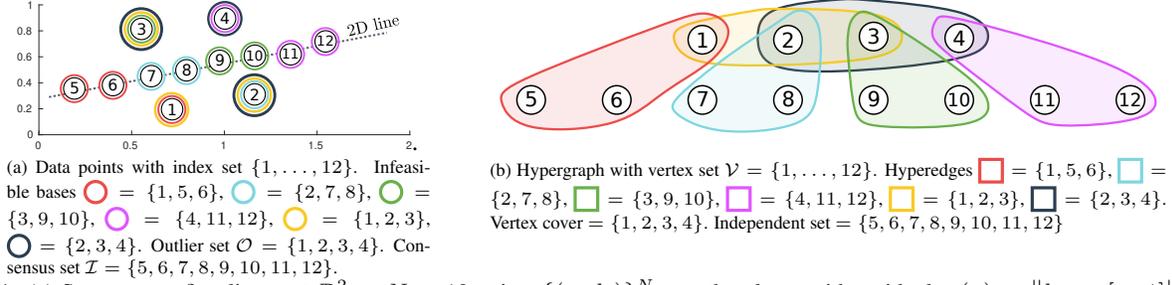
Figure 1. (a) Suppose we fit a line $\mathbf{x} \in \mathbb{R}^2$ on $N = 12$ points $\{(a_i, b_i)\}_{i=1}^N$ on the plane, with residual $r_i(\mathbf{x}) = \left\| b_i - \mathbf{x}[a_i, 1] \right\|_2$. The maximum consensus size is 8. Six infeasible bases are also plotted (colour coded, note that there are in total 281 infeasible bases). (b) Equivalent hypergraph H for (a) based on our construction (Sec. 3.2). The infeasible basis $\equiv$ hyperedge. Claim 2 implies the maximum consensus set $\equiv$ maximum independent set, and the minimum outlier set $\equiv$ minimum vertex cover.

which is the quadratic unconstrained binary optimisation (QUBO). QUBO is intractable on a classical machine, but a quantum annealer, by virtue of the physical processes described above, *may* solve the problem efficiently. It allows N-qubits to evolve through superposed and entangled states (quantum tunnelling), and $\mathbf{q}$ is obtained from the final measurement; see Sec. 4.3 on practical limitations.

4.2. Hypergraph vertex cover as QUBO

To simplify description of the main algorithm in Sec. 5, we first generalise (VC). Let A be a subset of the hyperedges E of the hypergraph H:

$$A = \{\mathcal{C}_{\mathbf{a}_{(m)}}\}_{m=1}^M \subseteq E = \{\mathcal{C}_{\mathbf{b}_{(k)}}\}_{k=1}^K. \qquad (14)$$

Define the 0-1 ILP

$$I(A) = \min_{\mathbf{z} \in \{0,1\}^N} \|\mathbf{z}\|_1, \qquad \text{s.t.} \quad \mathbf{A}^T \mathbf{z} \geq \mathbf{1}_M, \qquad (15)$$

where $\mathbf{A} \in \{0,1\}^{N \times M}$ is obtained by horizontally stacking the binary N-vectors $\{\mathbf{a}_{(m)}\}_{m=1}^M$ corresponding to the hyperedges in A, and $\mathbf{1}_M$ is the vector of M ones.

We can recover (VC) from (15) by setting $A = E$. Moreover, since $A \subseteq E$, it is clear that $I(A) \leq I(E)$.

To formulate (15) as a QUBO, we first convert the inequalities into equalities. Define $\delta' = \delta - 1$. For each constraint $\mathbf{a}_{(m)}^T \mathbf{z} \geq 1$ in (15), we incorporate δ' binary slack variables $\mathbf{t}_{(m)} = \begin{bmatrix} t_{m,1} & \ldots & t_{m,\delta'} \end{bmatrix}^T$ into the constraint

$$\mathbf{a}_{(m)}^T \mathbf{z} - \mathbf{t}_{(m)}^T \mathbf{1}_{\delta'} = 1; \qquad (16)$$

recall that each $\mathbf{a}_{(m)}$ has exactly δ elements with value 1. All M equality constraints can be expressed in matrix form

$$\mathbf{H}_A \begin{bmatrix} \mathbf{v}^T & 1 \end{bmatrix}^T = 0, \qquad (17)$$

where $\mathbf{v} = \begin{bmatrix} \mathbf{z}^T & \mathbf{t}_{(1)}^T & \ldots & \mathbf{t}_{(m)}^T & \ldots & \mathbf{t}_{(M)}^T \end{bmatrix}^T \in \{0,1\}^{N+\delta'M}$, $\mathbf{H}_A = \begin{bmatrix} \mathbf{A}^T & -\mathbf{S} & -\mathbf{1}_M \end{bmatrix} \in$

$\{0,1\}^{M \times (N+\delta'M+1)}$, $\mathbf{S} = \mathbf{I}_M \otimes \mathbf{1}_{\delta'}^T$, the $M \times M$ identity matrix $\mathbf{I}_M$, and Kronecker product $\otimes$. Also, the objective $\|\mathbf{z}\|_1$ can be expressed in the quadratic form

$$\begin{bmatrix} \mathbf{v}^T & 1 \end{bmatrix} \mathbf{J} \begin{bmatrix} \mathbf{v}^T & 1 \end{bmatrix}^T, \qquad (18)$$

with $\mathbf{J} = \begin{bmatrix} \mathbf{I}_N & \mathbf{0}_{N \times (\delta'M+1)} \\ \mathbf{0}_{(\delta'M+1) \times N} & \mathbf{0}_{(\delta'M+1) \times (\delta'M+1)} \end{bmatrix}$, where $\mathbf{0}$ is a zero matrix with the size specified in its subscript, and $\mathbf{I}_N$ is the $N \times N$ identity matrix.

With penalty parameter $\lambda > 0$, we lift the constraints (17) into the objective to yield the QUBO

$$Q_\lambda(A) = \min_{\mathbf{v} \in \{0,1\}^{N+\delta'M}} \begin{bmatrix} \mathbf{v}^T & 1 \end{bmatrix} (\mathbf{J} + \lambda \mathbf{H}_A^T \mathbf{H}_A) \begin{bmatrix} \mathbf{v}^T & 1 \end{bmatrix}^T. \qquad (19)$$

Further algebraic manipulation is required to remove the constant 1 from (19) before exactly matching (13); see Sec. C in the supp. material for details. In the following, we will discuss solving (19) using a quantum annealer.

4.3. Practical considerations and limitations

We frame the discussion here in the context of SOTA quantum annealer—the D-Wave Advantage [23].

Challenges Problem (19) is an application of the quadratic penalty method [53, Chap. 17]. While fundamental results exist that allow $Q_\lambda(A)$ to equal $I(A)$, they invariably require λ to approach a large value. However, the precision of D-Wave Advantage is limited to 4-5 bits [22, 28], which precludes the usage of large penalty parameters.

Second, although there are >5000 qubits in D-Wave Advantage, the topology of quantum processing unit (QPU) rules out a fully connected model, i.e., the $\mathbf{Q}$ matrix allowable is not dense [25,49]. Given an arbitrary $\mathbf{Q}$, a *minor embedding* step [9, 12, 62] is required to map the QUBO onto the QPU topology. The embedding consumes extra physical qubits reducing the number of physical qubits available.

As alluded in Sec. 4.1, the annealing process "gradually" transitions (NB: by human scale the transition is rapid) the

quantum system from the initial Hamiltonian to the final Hamiltonian. Current quantum annealers are not able to completely isolate external noise from the process, which affects the quality of the solution.

To obtain a useful solution, during the annealing process, the system must have a non-negligible probability of staying in the lowest energy state. If the system jumps to a higher energy state, it will fail in solving the QUBO (19) optimally. The spectral gap is the minimum gap between the lowest and the second lowest (higher) energy states, which affects the probability of staying in the lowest energy state; see [24, 31] for details. We will investigate the spectral gap issue in Sec. D of the supplementary material.

Why quantum annealing? The above issues limit the scale of problems and quality of solutions attainable with current quantum annealers. However, quantum technology is advancing steadily, and the vision community should be prepared for potential breakthroughs, as like-minded colleagues are also advocating [7, 8, 19, 34, 63, 65]. Moreover, our main algorithm combines quantum and classical computation to leverage the strengths of both paradigms.

5. Main algorithm

Alg. 1 presents our overall algorithm. At its core, our algorithm aims to solve (VC), i.e., find the minimum outlier set, but by incrementally generating the hyperedges $A \subseteq E$. Other main characteristics of the algorithm are:

- At each iteration, the QUBO (19) based on the current hyperedges A is solved using quantum annealing.
- The penalty λ for (19) decays following a schedule defined by hyperparameters $\bar{\lambda}$, γ and $\bar{M}$ (Step 6).
- Hyperedges are sampled from a candidate vertex set $\mathcal{V}'$, which is updated based on the current results (Sec. 5.2).

The algorithm terminates with the best estimate $\mathbf{z}_{\text{best}}$ of the minimum outlier set and the sampled hyperedges A.

In the following, we show how the outputs of Alg. 1 can be used to derive an error bound for consensus maximisation, and the rationale of our hyperedge sampling technique.

5.1. Error bound

Consider the relaxation of (15)

$$LP(A) = \min_{\mathbf{z} \in [0,1]^N} \quad \|\mathbf{z}\|_1, \quad \text{s.t.} \quad \mathbf{A}^T \mathbf{z} \geq \mathbf{1}_M \quad (20)$$

which is a linear program. We must have that

$$LP(A) \leq I(A) \leq I(E). \quad (21)$$

Due to the factors in Sec. 4.3, the solution $\mathbf{z}$ by the quantum annealer on (19) can be suboptimal. Given the best solution $\mathbf{z}_{\text{best}}$, if the $\mathcal{I}_{\text{best}} = \mathcal{V} \setminus \mathcal{C}_{\mathbf{z}_{\text{best}}}$ is a consensus set, by Claim 2, $\mathbf{z}_{\text{best}}$ is a vertex cover of (VC). We must have that

$$LP(A) \leq I(E) \leq \|\mathbf{z}_{\text{best}}\|_1. \quad (22)$$

Algorithm 1 Hybrid Quantum-Classical Robust Fitting. Note: Only Step 8 invokes the quantum annealer.

Require: Data $\mathcal{D} = \{\mathbf{p}_i\}_{i=1}^N$, inlier threshold ϵ, maximum iterations M, penalty λ with decay parameters $\bar{\lambda}, \gamma, \bar{M}$.
1: Initialise hyperedge set $A \leftarrow \emptyset$, candidate vertices $\mathcal{V}' \leftarrow \mathcal{V}$, best outlier set $\mathbf{z}_{\text{best}} \leftarrow \mathbf{1}_N$.
2: **for** $m = 1$ to M **do**
3:      $\mathbf{a}_{(m)} \leftarrow$ Active set of $\mathcal{V}'$ (see Sec. 5.2).
4:      $A \leftarrow A \cup \{\mathcal{C}_{\mathbf{a}_{(m)}}\}$.
5:      **if** m mod $\bar{M} = 0$ **then**
6:          $\lambda \leftarrow \max(\lambda/\gamma, \bar{\lambda})$.
7:      **end if**
8:      $\mathbf{v} = [\mathbf{z} \quad \mathbf{t}] \leftarrow$ Solve (19) using quantum annealing
9:      **if** $f(\mathcal{V} \setminus \mathcal{C}_{\mathbf{z}}) = 0$ **then**
10:         $\mathcal{I} \leftarrow \mathcal{V} \setminus \mathcal{C}_{\mathbf{z}}$ (found a consensus set).
11:         **if** $\|\mathbf{z}\|_1 < \|\mathbf{z}_{\text{best}}\|_1$ **then**
12:            $\mathbf{z}_{\text{best}} \leftarrow \mathbf{z}$.
13:         **end if**
14:         $\mathcal{V}' \leftarrow \mathcal{C}_{\mathbf{z}} \cup \{\text{random subset of } \mathcal{I}\}$ (see Sec. 5.2).
15:      **else**
16:         $\mathcal{V}' \leftarrow \mathcal{V} \setminus \mathcal{C}_{\mathbf{z}}$ (see Sec. 5.2).
17:      **end if**
18: **end for**
19: **return** Outlier set estimate $\mathbf{z}_{\text{best}}$ and hyperedge set A.

Using the fact that $|\mathcal{I}^*| = N - I(E)$, we thus have

$$|\mathcal{I}^*| - |\mathcal{I}_{\text{best}}| \leq \|\mathbf{z}_{\text{best}}\|_1 - LP(A). \quad (23)$$

If the RHS is 0, then $\mathcal{I}_{\text{best}}$ is the globally optimal solution.

5.2. Heuristic for sampling hyperedges

Recall that a hyperedge is an infeasible basis. A simple way to generate hyperedges is to randomly sample δ-subsets from $\mathcal{V}$ until we find an infeasible subset, which will not be efficient. To improve efficiency, our sampling technique maintains a candidate vertex set $\mathcal{V}' \subseteq \mathcal{V}$ where $f(\mathcal{V}') = 1$, and takes the active set $\mathcal{S}$ of $\mathcal{V}'$ [3, 30], where

$$g(\mathcal{S}) = g(\mathcal{V}'), \quad (24)$$

as a hyperedge. Intuitively, the active set of $\mathcal{V}'$ is a basis with equal value with $\mathcal{V}'$. To generate diverse hyperedges, two strategies are employed to maintain $\mathcal{V}'$:

- Take $\mathcal{V}' = \mathcal{V} \setminus \mathcal{C}_{\mathbf{z}}$ if it is not a consensus set (Step 16);
- If a new consensus set $\mathcal{I} = \mathcal{V} \setminus \mathcal{C}_{\mathbf{z}}$ is found (Step 10), set $\mathcal{V}'$ as the union of $\mathcal{C}_{\mathbf{z}}$ with a random subset of $\mathcal{I}$ (Step 14).

5.3. Hyperparameter selection

The penalty value and decay schedule play important roles in Alg. 1 to quickly find a consensus set and tighten the error bound, see Sec. E in supplementary material for details. The precise values for the parameters used and/or investigated in our experiments will be provided in Sec. 6.

6. Experiments

6.1. Synthetic data

We first examine the performance of D-Wave Advantage (version 1.1) [23] on our robust fitting formulation via synthetic data. We generated 2D points $\mathcal{D} = \{(a_i, b_i)\}_{i=1}^N$ for 1D linear regression ($x \in \mathbb{R}^1$) with residual $r_i(x) = |a_i x - b_i|$, where $0 \le a_i, b_i \le 1$. For a randomly chosen ground truth x, a proportion of the points are corrupted with Gaussian noise of $\sigma_{in} = 0.1$ to form inliers, with the rest by Gaussian noise of $\sigma_{out} = 1.5$ to simulate outliers.

Due to the cost of accessing the QPU, our results in this subsection were not derived from many data repetitions. However, each QPU input instance was invoked with 10,000 anneals, which typically consumed ≈ 1.3 seconds. Also, see Sec. F in the supp. material for details on embedding (19) (e.g., number of qubits, runtime) onto the QPU.

Comparison between CPU and QPU. We first compare CPU and QPU performance on our QUBO (19) (i.e., independent of Alg. 1), with A containing all hyperedges E. The CPU solver used was Gurobi [35], which solves (19) exactly via exhaustive search, hence practical only for small instances. Fig. 2 plots the number of outliers $\|\mathbf{z}\|_1$ (lower is better) optimised by the solvers as a function of

- Penalty $\lambda \in [0.1, 100]$, with $N = 50$, outlier ratio $= 0.2$.
- Outlier ratio $\in [0.1, 0.6]$, with $N = 20$, $\lambda = 1.0$.
- $N \in [10, 100]$, with $\lambda = 1.0$, outlier ratio $= 0.2$.

As expected (see Sec. 4.3), the gap in quality between the QPU solution and the "ground truth" provided by Gurobi increased with the examined parameters, indicating that the QPU is more reliable on "easier" instances of (19).

Main algorithm Fig. 3 illustrates running Alg. 1 on synthetic 1D linear regression instances with $N = 20$, 50, and 100 points, each with outlier ratio 0.2. The QUBO subroutine (19) in the main algorithm was solved using the QPU with $\lambda = 1.0$ (no λ decay was done). The values $\|\mathbf{z}\|_1$ and $LP(A)$ were plotted as a function of the size of A, i.e., number of hyperedges. The results mainly illustrate the feasibility of solving robust fitting using quantum annealing.

Comparing to simulated annealing In the context of Alg. 1, we compared quantum annealing (QA) and simulated annealing (SA) [37] (on CPU with 10,000 anneals) in solving the QUBO subroutine (Line 8 of Alg. 1). A synthetic 1D linear regression instance with $N = 20$ and outlier ratio 0.2 was generated. The penalty λ was set to 0.5 (no λ decay was done). Fig. 4 shows the runtime of QA and SA across the iterations of Alg. 1 (for QA, the cost of embedding (19) onto the QPU was excluded; again, see Sec. F in the supp. material for details), and the Hamming distance between the $\mathbf{z}$'s found by the methods in each iteration.

(a) Effect of penalty λ ($N = 50$, outlier ratio $= 0.2$)

(b) Effect of outlier ratio ($N = 20$, $\lambda = 1.0$)

(c) Effect of data size N ($\lambda = 1.0$, outlier ratio $= 0.2$)
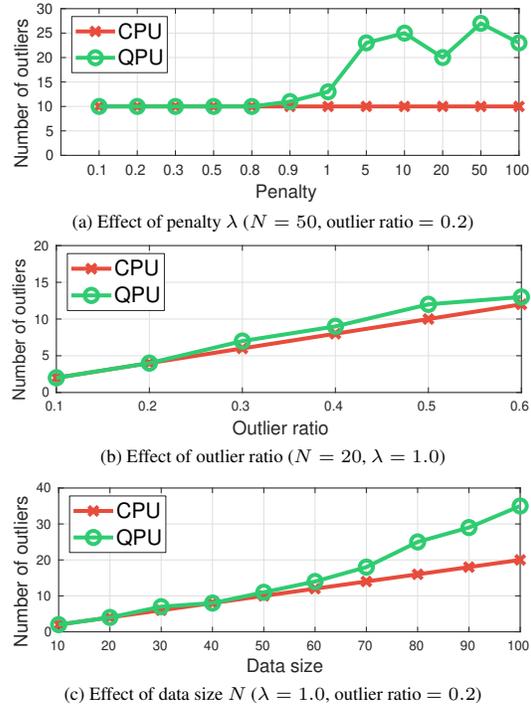
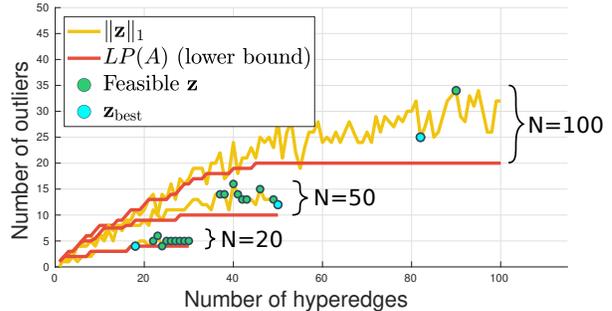Figure 2. Comparisons between CPU and QPU on QUBO (19).

Figure 3. Number of outliers $\|\mathbf{z}\|_1$ optimised by QPU and lower bound $LP(A)$, plotted across the iterations of Alg. 1.

The results illustrate that the runtime of SA (on CPU) grew steadily as the number of sampled hyperedges A increased, whereas the runtime of QA remained largely constant across the iterations, which suggests that the underlying physical processes of QA were not affected significantly by problem size (as long as the problem "fits" on the QPU). Also, Fig. 4 shows the solutions obtained by QA and SA are largely the same; this supports using SA in place of QA to examine the efficacy of Alg. 1 on larger sized real data.

6.2. Real data

We tested our method on real data for fundamental matrix estimation and multi-view triangulation. We used SA (on CPU) in place of QA to allow Alg. 1 to handle bigger problems. Two variants of Alg. 1 were executed:
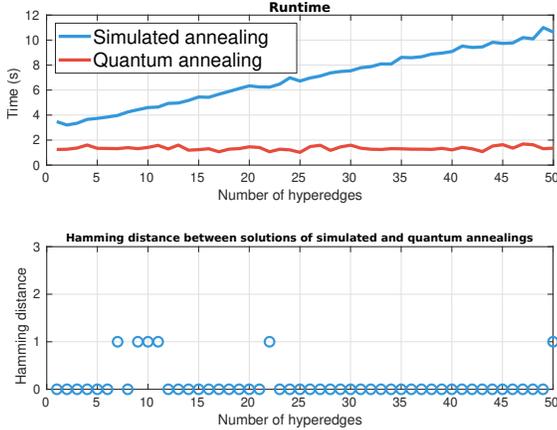
Figure 4. Comparing quantum annealing (on D-Wave Advantage) and simulated annealing (on classical computer).

- Alg. 1-E, where the algorithm was terminated as soon as a consensus set was found (Line 10).
- Alg. 1-F, where the algorithm was run until the maximum iterations M (300 for fund. matrix, 200 for triangulation).

We compared our method to i) random sampling methods: RANSAC (RS) [32], LO-RANSAC (LRS) [20], and Fixing LO-RANSAC (FLRS) [43], ii) deterministic algorithms: Exact penalty (EP) [41], and Iterative biconvex optimization (IBCO) [14], and iii) quantum robust fitting (QRF) [19]. Each method was run 100 times and average results were reported. All experiments were conducted on a system with 2.6 GHz processor and 16 GB of RAM.

6.2.1 Fundamental matrix estimation.

We evaluated our method on linearised fundamental matrix fitting [61, Chapter 4], where $\mathbf{x} \in \mathbb{R}^8$. We used inlier threshold $\epsilon = 0.03$ for the algebraic residual (convex in $\mathbf{x}$, hence also quasiconvex), and penalty parameters $\lambda = 1.0$, $\gamma = 0.5$, $\bar{M} = 50$, and $\bar{\lambda} = 0.01$.

We used three image pairs from VGG [72] (Castle, Valbonne, and Zoom) and three image pairs from sequence 00 of KITTI odometry [33] [3] (frame indices 104-108, 198-201, and 738-742). In each pair, SIFT features [48] were extracted and matched using VLFeat [1]; Lowe's second nearest neighbour test was also applied to prune matches.

Fig. 5 shows the intermediate outputs of Alg. 1-F on Castle, KITTI 198-201 and KITTI 738-742, particularly the lower bound of the solution. See Sec. G in the supplementary material for the plots for the other image pairs.

Table 1 compares our method with the others. Overall, the quality of our method was comparable to the others, with Alg. 1-F providing higher quality and tighter bound than Alg. 1-E. Note that only our method returned error bounds (Sec. 5.1), which allowed to deduce that Alg. 1-F found consensus sets that were close to the optimum. As

expected, the fastest methods were the random sampling approaches. Our method was much slower than the others, mainly due to the usage of SA. However, our experiments in Sec. 6.1 shows that QA can improve the speed of SA up to a factor of 10 without affecting solution quality (see Fig. 4). Hence, we expect Alg. 1 to be more competitive as quantum annealer capacity improves. Fig. 6 qualitatively illustrates the results of Alg. 1-E.
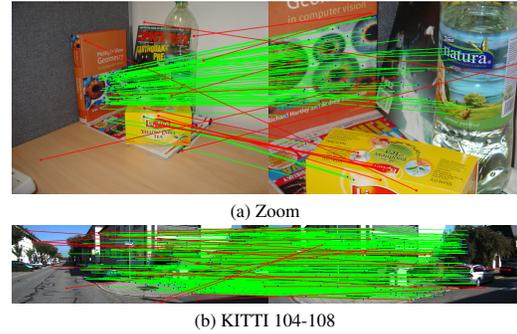

(a) Zoom

(b) KITTI 104-108

Figure 6. Qualitative results of Alg. 1-E on fundamental matrix estimation. Green and red lines represent inliers and outliers found.

6.2.2 Multi-view triangulation

Points 134 & 534 from Nikolai, points 1 & 14 from Linkoping, and points 3 & 132 from Tower [29] were used. In this task, 3D coordinates of those 3D points ($\mathbf{x} \in \mathbb{R}^3$) were estimated using reprojection error (which is quasiconvex [38]) under outliers. The inlier threshold and penalty were respectively $\epsilon = 1$ pixels and $\lambda = 5$. The decay parameters were $\gamma = 0.5$, $\bar{M} = 50$, and $\bar{\lambda} = 0.03$.

Fig. 7 shows the intermediate outputs of Alg. 1-F on Nikolai point 134, Linkoping point 1 and Tower point 132, particularly the lower bound of the solution. See Sec. G in the supplementary material for the plots for the other points. Interestingly, the results show that it was more difficult to find a tight lower bound here (especially Nikolai point 134 and Linkoping point 1). This could be due to numerical inaccuracies in solving the minimax problem (3) for quasiconvex residuals [3, 30], which affected the efficacy of hyperedge sampling. Table 2 shows the quantitative results; a similar conclusions as that of Table 1 can be drawn. In particular, note that only our method was able to provide error bounds; in the case of Tower point 132, the global solution was provably found by the algorithm (gap is zero).

7. Weaknesses and conclusions

There are two main shortcomings: First, Alg. 1 was validated on an actual quantum computer only for small scale synthetic data (for reasons covered in Sec. 4.3). To fully realise the potential of the algorithm, testing with real data on a quantum computer is needed. Second, our results reveal that the hyperedge sampling procedure is also crucial to Alg. 1. Developing a more effective way of sampling hyperedges is an interesting research direction.
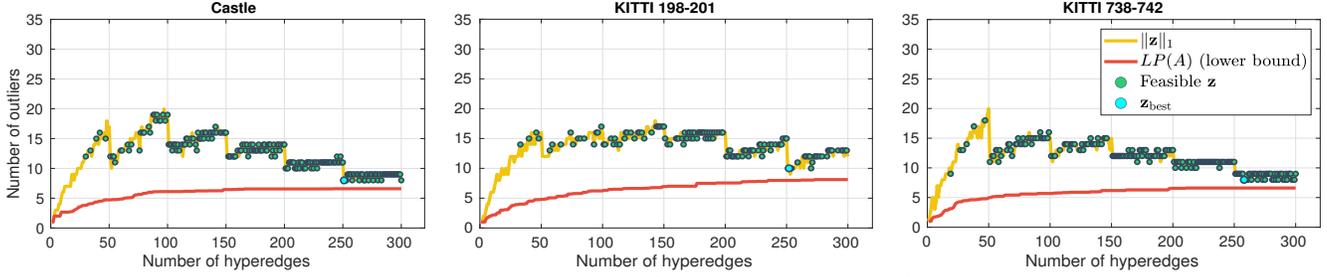
Figure 5. Fundamental matrix estimation, where number of outliers $\|\mathbf{z}\|_1$ and lower bound $LP(A)$, plotted across the iterations of Alg. 1-F.

	Method	RS [32]	LRS [20]	FLRS [43]	EP [41]	IBCO [14]	QRF [19]	Alg. 1-E	Alg. 1-F
Castle	$|\mathcal{I}|$ (Error bound)	74 (−)	74 (−)	74 (−)	70 (−)	76 (−)	73 (−)	72 (8.17)	76 (1.41)
$N = 84$	Time (s)	0.20	0.11	0.20	0.25	0.34	199.48	18.07	1998.87
Valbonne	$|\mathcal{I}|$ (Error bound)	34 (−)	36 (−)	36 (−)	33 (−)	38 (−)	29 (−)	36 (6.00)	36 (4.00)
$N = 45$	Time (s)	0.21	0.20	0.31	0.34	0.44	110.30	6.71	1915.82
Zoom	$|\mathcal{I}|$ (Error bound)	90 (−)	91 (−)	91 (−)	92 (−)	95 (−)	89 (−)	93 (9.91)	94 (3.64)
$N = 108$	Time (s)	0.31	0.29	0.14	0.21	0.35	257.03	92.35	2109.13
KITTI 104-108	$|\mathcal{I}|$ (Error bound)	309 (−)	313 (−)	312 (−)	318 (−)	321 (−)	256 (−)	320 (9.91)	324 (2.30)
$N = 337$	Time (s)	0.04	0.04	0.07	0.28	0.39	799.33	137.26	2408.04
KITTI 198-201	$|\mathcal{I}|$ (Error bound)	306 (−)	308 (−)	307 (−)	308 (−)	312 (−)	309 (−)	308 (10.00)	312 (1.89)
$N = 322$	Time (s)	0.05	0.13	0.07	0.23	0.42	774.06	36.15	2350.39
KITTI 738-742	$|\mathcal{I}|$ (Error bound)	481 (−)	483 (−)	483 (−)	491 (−)	492 (−)	447 (−)	492 (5.88)	493 (1.39)
$N = 501$	Time (s)	0.05	0.18	0.23	0.53	0.61	1160.12	22.46	2506.04

Table 1. Fundamental matrix estimation results. Alg. 1 employed simulated annealing (quantum annealing could be faster by 10 times). Only Alg. 1 amongst all methods compared here returned error bounds.
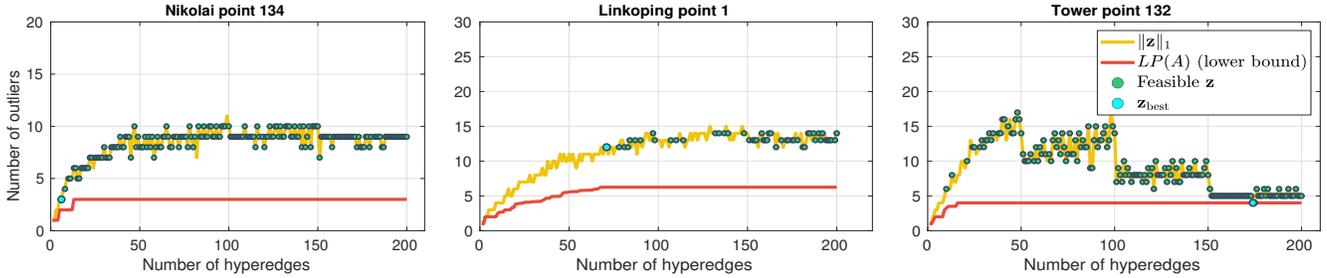

Figure 7. Multi-view triangulation, where number of outliers $\|\mathbf{z}\|_1$ and lower bound $LP(A)$ plotted across the iterations of Alg. 1-F.

	Method	RS [32]	LRS [20]	FLRS [43]	EP [41]	IBCO [14]	QRF [19]	Alg. 1-E	Alg. 1-F
Nikolai point 134	$|\mathcal{I}|$ (Error bound)	21 (−)	21 (−)	21 (−)	21 (−)	21 (−)	21 (−)	21 (1.00)	21 (0.00)
$N = 24$	Time (s)	0.24	0.32	0.30	0.34	0.36	158.39	6.12	159.28
Nikolai point 534	$|\mathcal{I}|$ (Error bound)	16 (−)	16 (−)	16 (−)	15 (−)	17 (−)	17 (−)	16 (2.00)	16 (1.00)
$N = 20$	Time (s)	0.27	0.35	0.25	0.29	0.32	154.63	8.71	147.14
Linkoping point 1	$|\mathcal{I}|$ (Error bound)	15 (−)	15 (−)	15 (−)	14 (−)	16 (−)	14 (−)	13 (5.75)	13 (5.75)
$N = 25$	Time (s)	0.25	0.30	0.34	0.38	0.47	175.83	20.05	153.79
Linkoping point 14	$|\mathcal{I}|$ (Error bound)	36 (−)	36 (−)	36 (−)	35 (−)	37 (−)	32 (−)	37 (4.67)	37 (4.27)
$N = 52$	Time (s)	0.27	0.44	0.38	0.53	0.64	360.37	130.10	194.46
Tower point 3	$|\mathcal{I}|$ (Error bound)	73 (−)	73 (−)	73 (−)	73 (−)	73 (−)	73 (−)	72 (3.00)	72 (1.00)
$N = 79$	Time (s)	0.28	0.64	0.32	0.36	0.43	555.27	27.26	177.43
Tower point 132	$|\mathcal{I}|$ (Error bound)	79 (−)	79 (−)	79 (−)	79 (−)	81 (−)	81 (−)	79 (2.75)	81 (0.00)
$N = 85$	Time (s)	0.30	0.62	0.42	0.51	0.51	563.43	26.33	163.32

Table 2. Multi-view triangulation results. Alg. 1 employed simulated annealing (quantum annealing could be faster by 10 times). Only Alg. 1 amongst all methods compared here returned error bounds.

Conclusions Our work illustrates the potential of quantum annealing for robust fitting. It outperforms (in simulation) the only other quantum approach in robust fitting [19] as well as offers an error bound to mitigate the weakness of current QPU. We hope that our work helps trigger further development on applying quantum computers in robust fitting and computer vision applications.

References

[1] VLFeat. https://www.vlfeat.org/. Accessed: 2021-11-02.

[2] Sameer Agarwal, Jongwoo Lim, Lihi Zelnik-Manor, Pietro Perona, David Kriegman, and Serge Belongie. Beyond pairwise clustering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[3] Nina Amenta, Marshall Bern, and David Eppstein. Optimal point placement for mesh smoothing. *Journal of Algorithms*, 1999.

[4] Pasquale Antonante, Vasileios Tzoumas, Heng Yang, and Luca Carlone. Outlier-robust estimation: Hardness, minimally-tuned algorithms, and applications. *IEEE Transactions on Robotics*, 2021.

[5] Daniel Barath and Jiří Matas. Graph-cut ransac. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[6] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. Magsac++, a fast, reliable and accurate robust estimator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[7] Marcel Seelbach Benkner, Vladislav Golyanik, Christian Theobalt, and Michael Moeller. Adiabatic quantum graph matching with permutation matrix constraints. In *International Conference on 3D Vision*, 2020.

[8] Tolga Birdal, Vladislav Golyanik, Christian Theobalt, and Leonidas J Guibas. Quantum permutation synchronization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[9] Tomas Boothby, Andrew D King, and Aidan Roy. Fast clique minor generation in chimera qubit connectivity graphs. *Quantum Information Processing*, 2016.

[10] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[11] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Hybrid quantum-classical algorithms for approximate graph coloring. *arXiv preprint arXiv:2011.13420*, 2020.

[12] Jun Cai, William G Macready, and Aidan Roy. A practical heuristic for finding graph minors. *arXiv preprint arXiv:1406.2741*, 2014.

[13] Zhipeng Cai, Tat-Jun Chin, and Vladlen Koltun. Consensus maximization tree search revisited. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[14] Zhipeng Cai, Tat-Jun Chin, Huu Le, and David Suter. Deterministic consensus maximization with biconvex programming. In *Proceedings of the European Conference on Computer Vision*, 2018.

[15] Simona Caraiman and Vasile Manta. Image processing using quantum computing. In *16th International Conference on System Theory, Control and Computing*, 2012.

[16] Tat-Jun Chin, Zhipeng Cai, and Frank Neumann. Robust fitting in computer vision: Easy or hard? In *Proceedings of the European Conference on Computer Vision*, 2018.

[17] Tat-Jun Chin, Pulak Purkait, Anders Eriksson, and David Suter. Efficient globally optimal consensus maximisation with tree search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[18] Tat-Jun Chin and David Suter. The maximum consensus problem: recent algorithmic advances. *Synthesis Lectures on Computer Vision*, 2017.

[19] Tat-Jun Chin, David Suter, Shin-Fang Ch'ng, and James Quach. Quantum robust fitting. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[20] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Joint Pattern Recognition Symposium*. Springer, 2003.

[21] Creative Commons. Attribution-NonCommercial-ShareAlike 3.0 License. https://creativecommons.org/licenses/by-nc-sa/3.0/. Accessed: 2021-11-16.

[22] D-Wave Systems, Inc. Error Sources for Problem Representation. https://docs.dwavesys.com/docs/latest/c_qpu_ice.html. Accessed: 2021-11-08.

[23] D-Wave Systems, Inc. QPU-specific physical properties: Advantage_system1.1. https://docs.dwavesys.com/docs/latest/_downloads/9a1b594d84370df6c0a09d00a5b72661/09-1237A-A_QPU_Properties_Advantage_system1_1.pdf. Accessed: 2021-11-02.

[24] D-Wave Systems, Inc. What is Quantum Annealing? https://docs.dwavesys.com/docs/latest/c_gs_2.html. Accessed: 2021-11-09.

[25] Nike Dattani, Szilard Szalay, and Nick Chancellor. Pegasus: The second connectivity graph for large-scale quantum annealing hardware. *arXiv preprint arXiv:1901.07636*, 2019.

[26] Aditya Dendukuri, Blake Keeling, Arash Fereidouni, Joshua Burbridge, Khoa Luu, and Hugh Churchill. Defining quantum neural networks via quantum time evolution. *Quantum Techniques in Machine Learning*, 2019.

[27] Aditya Dendukuri and Khoa Luu. Image processing in quantum computers. *Quantum Techniques in Machine Learning*, 2019.

[28] John E Dorband. Extending the d-wave with support for higher precision coefficients. *arXiv preprint arXiv:1807.05244*, 2018.

[29] Olof Enqvist, Fredrik Kahl, and Carl Olsson. Non-sequential structure from motion. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011.

[30] David Eppstein. Quasiconvex programming. *Combinatorial and Computational Geometry*, 2005.

[31] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106*, 2000.

[32] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

[33] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE conference on computer vision and pattern recognition*, 2012.

[34] Vladislav Golyanik and Christian Theobalt. A quantum computational approach to correspondence problems on point sets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[35] Gurobi. https://www.gurobi.com/. Accessed: 2021-11-02.

[36] Laszlo Gyongyosi and Sandor Imre. A survey on quantum computing technology. *Computer Science Review*, 2019.

[37] Joseph T. Iosue. https://qubovert.readthedocs.io/en/stable/. Accessed: 2021-11-02.

[38] Fredrik Kahl and Richard Hartley. Multiple-view geometry under the L_∞-norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[39] Amir Khoshaman, Walter Vinci, Brandon Denis, Evgeny Andriyash, Hossein Sadeghi, and Mohammad H Amin. Quantum variational autoencoder. *Quantum Science and Technology*, 2018.

[40] Donggyu Kim, Pureum Noh, Hyun-Yong Lee, and Eun-Gook Moon. Advancing hybrid quantum-classical algorithms via mean-operators. *arXiv preprint arXiv:2107.07527*, 2021.

[41] Huu Le, Tat-Jun Chin, and David Suter. An exact penalty method for locally convergent maximum consensus. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[42] Huu Le, Anders Eriksson, Michael Milford, Thanh Toan Do, Tat Jun Chin, and David Suter. Non-smooth m-estimator for maximum consensus estimation. In *Proceedings of the British Machine Vision Conference*, 2018.

[43] Karel Lebeda, Jiri Matas, and Ondrej Chum. Fixing the locally optimized ransac–full experimental evaluation. In *British Machine Vision Conference*, 2012.

[44] Hongdong Li. Consensus set maximization with guaranteed global optimality for robust geometry estimation. In *IEEE 12th International Conference on Computer Vision*, 2009.

[45] Junde Li and Swaroop Ghosh. Quantum-soft qubo suppression for accurate object detection. In *European Conference on Computer Vision*, 2020.

[46] Shuyuan Lin, Guobao Xiao, Yan Yan, David Suter, and Hanzi Wang. Hypergraph optimization for multi-structural geometric model fitting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

[47] Hairong Liu and Shuicheng Yan. Efficient structure detection via random consensus graph. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[48] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 2004.

[49] Harmut Neven, Vasil S Denchev, Marshall Drew-Brook, Jiayong Zhang, William G Macready, and Geordie Rose. Nips 2009 demonstration: Binary classification using hardware implementation of quantum annealing. *Quantum*, 2009.

[50] Hartmut Neven, Vasil S Denchev, Geordie Rose, and William G Macready. Qboost: Large scale classifier training withadiabatic quantum optimization. In *Asian Conference on Machine Learning*, 2012.

[51] Nga TT Nguyen and Garrett T Kenyon. Image classification using quantum inference on the d-wave 2x. In *IEEE International Conference on Rebooting Computing*, 2018.

[52] V Nguyen, SB Orbell, Dominic T Lennon, Hyungil Moon, Florian Vigneau, Leon C Camenzind, Liuqi Yu, Dominik M Zumbühl, G Andrew D Briggs, Michael A Osborne, et al. Deep reinforcement learning for efficient measurement of quantum devices. *npj Quantum Information*, 2021.

[53] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[54] Carl Olsson, Olof Enqvist, and Fredrik Kahl. A polynomial-time bound for matching and registration with outliers. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[55] Alvaro Parra Bustos and Tat-Jun Chin. Guaranteed outlier removal for rotation search. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[56] Alejandro Perdomo-Ortiz, Marcello Benedetti, John Realpe-Gómez, and Rupak Biswas. Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. *Quantum Science and Technology*, 2018.

[57] Trung Pham, Tat-Jun Chin, Jin Yu, and David Suter. Simultaneous sampling and multi-structure fitting with adaptive reversible jump mcmc. *Advances in Neural Information Processing Systems*, 2011.

[58] Pulak Purkait, Tat-Jun Chin, Alireza Sadri, and David Suter. Clustering with hypergraphs: the case for large hyperedges. *IEEE transactions on pattern analysis and machine intelligence*, 2016.

[59] Pulak Purkait, Christopher Zach, and Anders Eriksson. Maximum consensus parameter estimation by reweighted $\ell1$ methods. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2018.

[60] René Ranftl and Vladlen Koltun. Deep fundamental matrix estimation. In *Proceedings of the European conference on computer vision*, 2018.

[61] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[62] Neil Robertson and Paul D Seymour. Graph minors. xiii. the disjoint paths problem. *Journal of combinatorial theory, Series B*, 1995.

[63] Michele Sasdelli and Tat-Jun Chin. Quantum annealing formulation for binary neural networks. *International Conference on Digital Image Computing Techniques and Applications*, 2021.

[64] Wolfgang Scherer. *Mathematics of Quantum Computing*. Springer, 2019.

[65] Marcel Seelbach Benkner, Zorah Lähner, Vladislav Golyanik, Christof Wunderlich, Christian Theobalt, and Michael Moeller. Q-match: Iterative shape matching via quantum annealing. In *International Conference on Computer Vision*, 2021.

[66] David Suter, Ruwan Tennakoon, Erchuan Zhang, Tat-Jun Chin, and Alireza Bab-Hadiashar. Monotone boolean functions, feasibility/infeasibility, lp-type problems and maxcon. *arXiv preprint arXiv:2005.05490*, 2020.

[67] Ruwan Tennakoon, David Suter, Erchuan Zhang, Tat-Jun Chin, and Alireza Bab-Hadiashar. Consensus maximisation using influences of monotone boolean functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[68] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 2000.

[69] Quoc Huy Tran, Tat-Jun Chin, Wojciech Chojnacki, and David Suter. Sampling minimal subsets with large spans for robust estimation. *International journal of computer vision*, 2014.

[70] Giang Truong, Huu Le, David Suter, Erchuan Zhang, and Syed Zulqarnain Gilani. Unsupervised learning for robust fitting: A reinforcement learning approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[71] Salvador E Venegas-Andraca and Sougato Bose. Storing, processing, and retrieving an image using quantum mechanics. In *Quantum Information and Computation*. International Society for Optics and Photonics, 2003.

[72] Visual Geometry Group. https://www.robots.ox.ac.uk/~vgg/data/. Accessed: 2021-11-02.

[73] Fei Wen, Rendong Ying, Zheng Gong, and Peilin Liu. Efficient algorithms for maximum consensus robust fitting. *IEEE Transactions on Robotics*, 2019.

[74] Fei Yan, Abdullah M Iliyasu, and Salvador E Venegas-Andraca. A survey of quantum image representations. *Quantum Information Processing*, 2016.