Fast Simultaneous Gravitational Alignment of Multiple Point Sets

Vladislav Golyanik Soshi Shimada Christian Theobalt

Max Planck Institute for Informatics, SIC

Abstract

The problem of simultaneous rigid alignment of multiple unordered point sets which is unbiased towards any of the inputs has recently attracted increasing interest, and several reliable methods have been newly proposed. While being remarkably robust towards noise and clustered outliers, current approaches require sophisticated initialisation schemes and do not scale well to large point sets. This paper proposes a new resilient technique for simultaneous registration of multiple point sets by interpreting the latter as particle swarms rigidly moving in the mutually induced force fields. Thanks to the improved simulation with altered physical laws and acceleration of globally multiply-linked point interactions with a 2^D-tree (D is the space dimensionality), our Multi-Body Gravitational Approach (MBGA) is robust to noise and missing data while supporting more massive point sets than previous methods (with 10^5 points and more). In various experimental settings, MBGA is shown to outperform several baseline point set alignment approaches in terms of accuracy and runtime. We make our source code available for the community to facilitate the reproducibility of the results[1].

1. Introduction

Rigid point set alignment algorithms conventionally handle two point sets, one of which is a fixed reference, and the other one is a template being transformed [6]. This setting is a well-understood and -studied. With more than two point sets, selecting and fixing one point set as a reference will result in a biased solution towards the selected reference. A global or groupwise approach handling all point sets on par, *i.e.,* enabling each point set to transform during the alignment [16, 19, 35], is a recognised alternative to various pairwise schemes requiring a subsequent consensus of transformations [5, 33]. Groupwise registration arises in different domains of 3D vision and engineering, including 3D reconstruction, CAD modelling, autonomous robot navigation and simultaneous localisation and mapping, among others. While remarkable progress has been achieved in groupwise multi-point set alignment, handling of large point sets (with 10^5 points and more) remains challenging. Subsampling of large point sets can result in loss of high-frequency details and low alignment accuracy. Hence, our goal is a method which is robust to noise and which can deal with a large number of points with no subsampling.

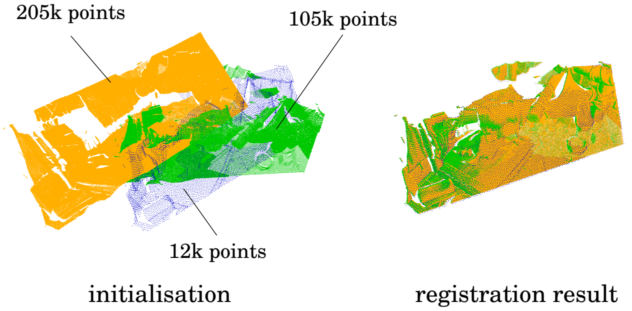

Figure 1: The proposed MBGA simultaneously registers multiple point sets of varying densities into a common reference frame. Thanks to the acceleration with a 2^D-tree, this is the first approach for multi-point-set alignment which performs globally multiply-linked point interactions for large point sets in feasible time (in minutes instead of years).

The core variables for deriving the complexity of groupwise point set alignment approaches are the number of input point sets L, the average cardinality of the point sets N and a model-specific factor Q which reflects how many operations are expected per point of every point set. Thus, we arrive at $\mathcal{O}(QNL)$ complexity in the general case. If point sets are multiply-linked, *i.e.,* every point of one point set interacts with a region of another point set, or nearest-neighbour search is required, then $Q{\sim}N$. If data structures for acceleration such as k-d trees are used, the worst-case complexity for computing nearest neighbours is still linear (which is more and more probable with the growing point dimensionality [22, 51]) in N implying the overall worst-case complexity to be $\mathcal{O}(QN^2)$. Especially in probabilistic approaches, where all incoming point sets are assumed to be sampled from a single unknown mixture model, one can often achieve an accurate alignment with $Q < N$. However, there is no guarantee that the same Q will work well for a broader range of inputs. In some cases, setting $Q > N$

[1] http://gvv.mpi-inf.mpg.de/projects/MBGA/

is necessary, and only $\mathcal{O}(QN^2)$ can be conservatively assumed without any prior knowledge about the scene.

Contributions. Our contributions in this work are:

- We present Multi-Body Gravitational Approach (MBGA) — *the first particle dynamics based method for simultaneous alignment of multiple point sets*, see Fig. 1. It operates in *the reference-free mode* and models point clouds as rigid particle swarms moving in mutually induced and alternating gravitational force fields (Sec. 4).
- We accelerate the calculation of gravitational potentials acting between particles with a Barnes-Hut 2^D-tree (quadtree in 2D, octree in 3D) [4] operating on the union of point clouds, and *guarantee the quasi-linear runtime complexity in the total number of points* NL (Secs. 4.3 and 4.6). We show that this data structure can induce a *new shape signature* relying on fitting a polynomial of degree three and considering the cubic surface component at every point (Sec. 4.5). The new shape signature helps in handling partial overlaps.
- Experimental evaluation with *state-of-the-art results among non-learning based methods*, in which MBGA outperforms several existing point set alignment approaches in terms of accuracy and runtime. *We empirically confirm the optimality criterion of locally minimal gravitational potential energy (GPE) generalises to more than two point sets handled on par, in a wide range of scenarios.* MBGA aligns point set groups more accurately than the recent pairwise gravitational method BHRGA [27] and several variants of ICP [6, 21]; the pairwise methods are tested in the *sequential* and *one-to-many* alignment modes. We also align point set with significantly more points that the tested implementation of the groupwise JRMPC method [19] can support (Sec. 5).

MBGA is a globally multiply-linked approach, *i.e., every point of every point set is interacting with every point of every other point set in the system.* This is a strong property, considering that existing pairwise methods *de facto* restrict point interactions to local neighbourhoods [49, 41]. Thus, the key strength of MBGA is the unification (for the first time, to the best of our belief) of two seemingly contradicting properties — the global multiply-linked point interactions and the $\mathcal{O}(LN\log(LN))$ complexity.

Moreover, MBGA has only a single parameter θ (the cell opening distance) and handles massive point sets on a mainstream CPU. It is easy to implement and contains ∼400 lines of C++ code. Different types of prior information (*e.g.,* point colours, density and prior matches), if available, can be mapped to masses. MBGA can be used for completion of room scans obtained with a depth sensor. All in all, it offers a valuable alternative to current joint multi-point-set registration methods and extends the scope of available techniques to larger point sets.

2. Related Work

Early point set registration algorithms were motivated by emerging 3D scanners that produce partial point clouds that need to be aligned. The seminal *iterative closest point* (ICP) algorithm for aligning two point sets [6, 10] alternates between transformation estimation [31, 32] and local correspondence inference [17]. The heuristic local correspondence search of ICP makes it prone to erroneous local convergence if badly initialised, and sensitive to outliers. Different improvements were subsequently proposed for ICP, ranging from accelerating policies for nearest-neighbour search [30, 43] and relaxation of one-to-one correspondences [24] to more efficient optimisation schemes [45, 20].

The methods [5, 16] are among the first to address the groupwise case. Bergevin *et al.* minimise registration error between all point clouds in a view network using an extension of ICP assuming one point cloud to be fixed. Eggert *et al.* [16] align point sets into a common, initially unknown, reference frame. The poses are updated iteratively by modelling spring forces between matching points, velocities and accelerations by approximate finite element analysis. The main drawback of these methods is the expensive correspondence search in every iteration and deterministic correspondence assignment, making them sensitive to noise and large initial misalignments. A recent and more robust extension of ICP called Motion Averaged ICP (MAICP) is based on motion averaging on a view graph [28].

Another class of methods models source and target point clouds as probability density functions [12, 41, 37], such as Gaussian Mixture Models (GMMs). They can implicitly model multiply-linked assignments and incorporate noise models [9]. The methods probabilistically find a maximum-a-posteriori solution via an expectation-maximisation algorithm or minimise an explicit probability density distance. Wider convergence basin was obtained by Gaussian mixture decoupling [15] or combining GMM representation and continuous domain mapping with a support vector machine [8]. The multiply-linked Kernel Correlation (KC) approach minimises the Renyi's quadratic entropy of the joint system composed of the reference and the transformed template [49]. In contrast to our method, only local point neighbourhoods are involved in multiply-linked interactions.

Probabilistic multi-point-set alignment methods lead to remarkable progress [19, 13, 35]. The JRMPC algorithm [19] assumes that a collection of point sets is sampled from the same Gaussian Mixture Model and finds the most probable generative GMM [19]. Lately, Danelljan *et al.* extended JRMPC by incorporating colour information [13], and Lawin *et al.* improved it for the case of spatially varying point densities [35]. JRMPC can become impractical for point sets with $35{-}40k$ points and more. In contrast, our method can handle point sets with 10^5 points and more.

Deep learning methods for alignment of two point sets

[3, 50, 11, 44] achieve remarkable accuracy in scenarios with available training datasets, while the generalisability to arbitrary scenarios with different point set characteristics (*e.g.,* varying density or volumetric sampling) is an open question. It is ongoing research on how these methods can be generalised to more than two point sets [23]. We do not assume available training data, and, thus, can handle arbitrary scenes with moderate overlaps. Moreover, MBGA is a correspondence-free approach, *i.e.,* it recovers an alignment in a single joint optimisation and does not require preliminary correspondence extraction.

Recently, approaches relying on analogies to physical processes gained attention [14, 25, 36, 26]. Deng *et al.* cast point sets into the Schrödinger distance transform representation, and align them by minimising a geodesic distance on a unit Hilbert sphere [14]. Golyanik *et al.* [25, 27] and Jauer *et al.* [36] interpret point sets as rigid swarms of particles with masses moving in gravitational or electromagnetic force fields. The optimal alignment corresponds to the state of the minimum potential energy of the system. In [26], laws of quantum physics are used to align point sets which are encoded by magnetic fields acting on qubits.

Our method most closely relates to [16, 25, 27] but substantially differs from and improves over them in terms of performance and scalability for multiple point sets. Eggert *et al.* [16] use Hooke's law and has quadratic worst-case complexity. It relies on nearest neighbour correspondence search. While the gravitational approach [25] is globally-multiply linked, it supports only two point sets with a fixed reference and relies on solving second-order ordinary differential equations which can cause oscillations. Besides, these methods include velocity damping for convergence. Similarly to [27], we abstract away from the physically-accurate modelling and alter physical laws for the sake of faster convergence, stability, enhanced numerical properties and fewer parameters which need to be set. Both [27] and our MBGA employ a Barnes-Hut tree for the accelerated computation of the potentials. We show that MBGA improves the speed and accuracy when handling multiple point sets, over a sequence of point set pair alignments performed by BHRGA [27] and other widely-used methods [6, 20]. Since MBGA handles all samples on par, its performance is path-invariant and has the loop-closure effect.

3. Gravitational Alignment of Two Point Sets

If a system of two point sets $\{\mathbf{X}, \mathbf{Y}\}$ is given, whereby $\mathbf{X}$ is the fixed reference, the input point sets can be aligned by minimising the mutual *gravitational potential energy* (GPE) $\mathbf{U}$ of the corresponding system of particles in the force field induced by $\mathbf{X}$, parametrised by translation $\mathbf{t}$ and rotation $\mathbf{R}$:

$$\mathbf{U}(\mathbf{R}, \mathbf{t}) = -G \sum_{i,j} \frac{m_{\mathbf{y}_i}\, m_{\mathbf{x}_j}}{\|\mathbf{R}\,\mathbf{y}_i + \mathbf{t} - \mathbf{x}_j\|_2 + \epsilon}, \qquad (1)$$

where $m_{\mathbf{y}_i}$ and $m_{\mathbf{x}_j}$ denote masses, $[\mathbf{y}_i] = \mathbf{Y}$, $[\mathbf{x}_j] = \mathbf{X}$, G is the gravitational constant and ϵ is a softening parameter for preventing near-field singularities. In [25], this energy is minimised implicitly by updating the forces $\vec{\mathbf{f}_i}$ acting on particles $\mathbf{y}_i$, accelerations (using Newton's second law of motion $\vec{\mathbf{f}_i} = m_{\mathbf{y}_i}\vec{a}_i$), velocities v_i^{t+1} and individual point displacements d_i^{t+1}:

$$\vec{\mathbf{f}_i} = -Gm_{\mathbf{y}_i} \sum_j m_{\mathbf{x}_j} \left(\|\mathbf{R}\mathbf{y}_i + \mathbf{t} - \mathbf{x}_j\|^2 + \epsilon^2 \right)^{-3/2} \hat{\mathbf{n}}_{i,j} - \eta v_i^t, \qquad (2)$$

$$v_i^{t+1} = v_i + \Delta t \, \frac{\vec{\mathbf{f}_i}}{m_{\mathbf{y}_i}} \text{ and } d_i^{t+1} = \Delta t \, v_i^{t+1}. \qquad (3)$$

In (2), η denotes a dissipation constant which determines the portion of the kinetic energy dragged from the system, per template's particle. In (3), Δt is time or, generally, the forward integration step. In every iteration — once updated — the unconstrained displacements are added to the current positions. A rigid consensus transformation is found using Procrustes analysis [32] relating the previous and current poses. The method converges when the difference in GPE of two last successive system states is below some threshold, or the maximum number of iterations is reached.

4. Multi-Body Gravitational Approach

4.1. Notations and Assumptions

Let $\boldsymbol{Y} = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_L\}$ be the set of input point sets of dimensionality D. We call all point sets $\boldsymbol{Y}_l$, $l \in \{1, \ldots, L\}$ templates, since no reference is selected and all $\boldsymbol{Y}_l$ are transformed during the alignment. The objective of MBGA is to recover a set of rigid transformation parameters $\{\mathbf{R}_l, \mathbf{t}_l\} \in \mathbf{T}_l$ aligning all $\boldsymbol{Y}_l$ into a common reference frame $\mathscr{R}$ (which is not known in advance) and correspondences across all $\boldsymbol{Y}_l$ for points which have valid matches, see Fig. 1. $\mathbf{R}_l$ and $\mathbf{t}_l$ denote rotation and translation of $\boldsymbol{Y}_l$, respectively. We denote a rigid transformation operator applying $\mathbf{T}_l$ on single or multiple points by $g(\mathbf{T}_l, \cdot)$.

In MBGA, every $\boldsymbol{Y}_l$ is interpreted as a rigidly transformed particle swarm, and every point induces a gravitational force field and moves in the external (relative to the point set it belongs to) force field. We assume that in the general case, point sets are of different cardinalities, *i.e.,* $|\boldsymbol{Y}_1| \neq |\boldsymbol{Y}_2| \neq \ldots \neq |\boldsymbol{Y}_L|$. Particles in our system possess masses, and their movements are modelled following Newton's second law of motion. The mass of every point is condensed in an infinitesimal volume of space, and collisions are not possible (no particle merging or splitting are allowed) [1]. To align the point clouds, we need to find a system's configuration corresponding to the minimal GPE.

4.2. Our Gravitational Potential Energy Functional

It has been shown in [27] that the GPE (1) can be improved with the negative elementwise reciprocal transform

(NERT), resulting after a series of further simplifications in

$$\mathbf{E}(\mathbf{R},\mathbf{t}) = \sum_i \sum_j m_{\mathbf{y}_i}\, m_{\mathbf{x}_j} \left\| \mathbf{R}\,\mathbf{y}_i + \mathbf{t} - \mathbf{x}_j \right\|_2 . \tag{4}$$

This transform enables a robust optimisation by non-linear least squares (NLLS) with a faster convergence while still preserving the notion of GPE (the higher the distance between the particles, the higher is the GPE). NERT makes multiple parameters of GA obsolete, and the remaining ones cover a much broader range of scenarios. Furthermore, core acceleration techniques applicable to the classical $\mathtt{n}$-body problem, such as Barnes-Hut octree [4] or fast multipole method [29], are also applicable to (4). Thus, we are building upon the currently most accurate method of the gravitational class and propose to minimise the mutual GPE of the system with L point sets $\boldsymbol{Y}_l$, where the latter are constrained to move rigidly:

$$\mathbf{E}(\mathbf{T}) = \sum_{l=1}^{L} \sum_{i=1}^{|\boldsymbol{Y}_l|} \sum_{\substack{\mathbf{p}_j \in \\ \{\boldsymbol{Y} \setminus \boldsymbol{Y}_l\}}} \left(m^l_{\mathbf{p}_i}\, m_{\mathbf{p}_j} \left\| g(\mathbf{T}_l, \mathbf{p}^l_i) - \mathbf{p}_j \right\|_2 \right), \tag{5}$$

with $\mathbf{T} = \{\mathbf{T}_l\} = \{\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_L\}$, $\mathbf{p}^l_i \in \boldsymbol{Y}_l$, $\mathbf{p}_j \in \{\boldsymbol{Y} \setminus \boldsymbol{Y}_l\}$, rigid transformation operator $g(\cdot,\cdot)$, and masses $m^l_{\mathbf{p}_i}$ and $m_{\mathbf{p}_j}$ of $\mathbf{p}^l_i$ and $\mathbf{p}_j$, respectively. There are multiple changes in the GPE for the multi-body case compared to the case with two point sets. In (5), every $\boldsymbol{Y}_l$ is influenced by a transformed gravitational force field induced by all other point sets excluding $\boldsymbol{Y}_l$. For each $\boldsymbol{Y}_l$, there is such $\mathbf{T}_l$ so that $\mathbf{E}(\mathbf{T})$ is minimal. Similar to the two point sets case (see Sec. 3), locally-optimal alignment is achieved when the GPE of the system is locally minimal.

Eq. (5) is minimised iteratively. In every iteration, we alternatingly update $\mathbf{T}_l$ by changing the roles between the current transformed point set $\boldsymbol{Y}_l$ and force inducing point sets $\{\boldsymbol{Y} \setminus \boldsymbol{Y}_l\}$, *i.e.,* a union of the remaining templates. Compared to the pairwise case, there are also substantial differences in the way how the 2^D-tree is used for acceleration of all particle interactions.

4.3. Point Clustering with a 2^D-Tree

We approximate globally multiply-linked interactions between $\boldsymbol{Y}_l$ and $\{\boldsymbol{Y} \setminus \boldsymbol{Y}_l\}$ by grouping the points of the latter into clusters. Instead of exhaustively summing up interactions between every $\mathbf{p}^l_i$ and every $\mathbf{p}_j$, we cluster $\mathbf{p}_j$ — so that every cluster is represented by its mass located at its centre of mass — and compute a single interaction between $\mathbf{p}^l_i$ and the cluster. The further away from $\mathbf{p}^l_i$ are $\mathbf{p}_j$, the larger clusters can be built and more runtime can be saved, eventually leading to the number of evaluations for every template point $\mathbf{p}^l_i$ proportional to $\log(|\{\boldsymbol{Y} \setminus \boldsymbol{Y}_l\}|)$.

We cluster $\mathbf{p}_j$ with a 2^D-tree, *i.e.,* a quadtree in 2D and an octree in 3D. 2^D-tree is a representation allowing for point agglomeration depending on the desired granularity level [4]. Building a 2^D-tree requires quasi-linear time, and extracting a clustered representation takes logarithmic time in the total number of points. The granularity level is determined by the depth at which the clusters are fetched. Note that with every level, the cell is partitioned into 2^D subcells, and all points falling into a cell at any level belong to that cell. The maximum density of the points determines the tree's depth because finally, each point occupies a leaf. For more details, the reader can refer to [4] as we use the algorithm exactly as proposed in [4]. The difference lies in the points belonging to the 2^D-tree and how the tree is used.

We build a 2^D-tree on the combined set $\boldsymbol{Y}$. To prevent inter-point-set interactions, the masses m^l_i of all points $\mathbf{p}^l_i \in \boldsymbol{Y}_l$ are set to zero, whereas while extracting a clustered representation from the tree for every $\mathbf{p}^l_i$, we restore its original non-zero mass. Cluster composition and extraction require a single parameter θ. Among other things, it determines the amount of the nearest points which are not clustered during GPE calculation. Suppose l is the length of the current examined cell, and μ is the distance from $\mathbf{p}^l_i$ to the centre of the cell. If $\frac{l}{\mu} < \theta^{-1}$, then the cell's influence on $\mathbf{p}^l_i$ is accumulative. Otherwise, the subcells are visited recursively until the condition is satisfied or the tree is entirely traversed.

For every current point set $\boldsymbol{Y}_l$ with the corresponding $\mathbf{R}_l$ and $\mathbf{t}_l$ being updated, we denote the set of fetched clusters by $\mathbf{C}_l$. As a set of fetched clusters for every $\mathbf{p}^l_i \in \boldsymbol{Y}_l$ is different, we use the specified notation $\mathbf{C}_{l,i}$ for every $\mathbf{p}^l_i$ when necessary. Thus, the acceleration with a 2^D-tree alters the GPE (5) as follows:

$$\mathbf{E}_{\mathbf{C}}(\mathbf{T}) = \sum_{l=1}^{L} \sum_{i=1}^{|\boldsymbol{Y}_l|} \sum_{\mathbf{c}_j \in \mathbf{C}_{l,i}} \left(m^l_{\mathbf{p}_i}\, m_{\mathbf{c}_j} \left\| g(\mathbf{T}_l, \mathbf{p}^l_i) - \mathbf{c}_j \right\|_2 \right), \tag{6}$$

with the fetched clusters $\mathbf{c}_j \in \mathbf{C}_{l,i}$ and cluster masses $m_{\mathbf{c}_j}$.

4.4. Defining Boundary Conditions through Masses

In gravitational methods, including MBGA, mass distribution serves as an additional alignment cue. We propose a policy for the integration of prior matches into our approach. Let N_c be the subset of points for which it is known beforehand that they match among each other across all L inputs. By setting point masses in N_c larger than the default masses, the penalty increases for the cases when the points with higher masses do not coincide. Thus, we assign to the points from N_c a higher weight which is proportional to the confidence of prior matches (usually, two-three orders of magnitude higher compared to the default masses). Such GPE pre-conditioning improves the alignment accuracy under challenging scenarios with noise and missing data in cases when prior matches are available or can be obtained in a pre-processing step. Other cues such as colours or geometric descriptors (such as the one proposed in Sec. 4.5) can be likewise mapped to masses.

4.5. Shape Signature Induced by a 2^D-Tree

Structure variability is the primary registration cue in point set alignment. In scenarios with partially-overlapping data and outliers, uniform mass initialisation can result in suboptimal registrations. We are encouraged to counterbalance increasing correspondence uncertainties with mass assignments relying on geometric features and propose a new shape signature induced by the surface fitting. For every $\mathbf{p}_i^l \in \boldsymbol{Y}_l$, we fit a surface of degree three in its vicinity.

Suppose M_i^l is a local point neighbourhood of $\mathbf{p}_i^l$ and $K = |M_i^l|$, *i.e.,* the number of points in M_i^l. We find the best approximation of M_i^l by a third degree polynomial of two variables which in general form reads:

$$\begin{gathered} z = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + \\ a_6 y^2 + a_7 x^3 + a_8 x^2 y + a_9 x y^2 + a_{10} y^3, \end{gathered} \tag{7}$$

with the shortcut notation $\mathbf{p}_i^l = (x, y, z)$, and the unknown coefficients $\boldsymbol{a} = \{a_1, \ldots, a_{10}\}$. Following the standard theory of surface approximation [42], we take partial derivatives of the sum of residuals

$$E = \sum_{i=1}^{K} \left(r_i^l\right)^2 = \sum_{i=1}^{K} \left(z_i - (a_1 + a_2 x_i + \ldots + a_{10} y_i^3)\right)^2 \tag{8}$$

and equate them to zero. As a result, the following linear system is obtained:

$$\begin{bmatrix} 1 & x_1 & y_1 & x_1 y_1 & \ldots & y_1^3 \\ 1 & x_2 & y_2 & x_2 y_2 & \ldots & y_2^3 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_K & y_K & x_K y_K & \ldots & y_K^3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{10} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_K \end{bmatrix}. \tag{9}$$

We estimate $\boldsymbol{a}$ in (9) by solving the corresponding normal equations. Finally, we take the sum of the absolute values of a_7 and a_{10} coefficients as a shape descriptor, *i.e.,* the indicator of the cubic surface component. For highly curved surfaces, the cubic component is strong, and for more flat surfaces it either weak or vanished. During the mass initialisation, points with high descriptor values are assigned tenfold masses (this value is scenario-specific), whereas the remaining masses are initialised uniformly with smaller values. Our shape descriptor suppresses areas with probable outliers in flat regions as well as noise, and emphasises more descriptive geometric cues reflected in higher masses.

For every $\mathbf{p}_i^l$, computing the shape descriptor requires finding the local neighbourhood M_i^l. To find M_i^l, we use a 2^D-tree described in Sec. 4.3. First, we build a separate 2^D-tree for the given $\boldsymbol{Y}_l$. Next, we use the property that every point inserted into a 2^D-tree corresponds to a leaf. For a given $\mathbf{p}_i^l$, we query the closest points (the leaves) from $\mathbf{C}_{l,i}$ by choosing a sufficiently large $\theta > 8.0$. Thus, the proposed light-weight shape descriptor executes in $\mathcal{O}(|\boldsymbol{Y}_l| \log(|\boldsymbol{Y}_l|))$ time. Its additional advantages are the continuity, *i.e.,* the property that every point is assigned a shape signature value unless thresholded (and not only sparse points which is the case with widespread shape descriptors for approximate alignment [46, 52, 47]), and code reusability (the 2^D-tree is used unaltered).

4.6. Computational and Memory Complexity

Proposition 1. *The computational complexity of MBGA is $\mathcal{O}(L\bar{N}\log(L\bar{N}))$, where L is the total number of point sets, and $\bar{N}$ is the average number of points in each point set.*

Proof. We assume, without loss of generality, that every point set has $\bar{N}$ points on average. For $L\bar{N}$ points, a 2^D-tree is built in $\mathcal{O}(L\bar{N}\log(L\bar{N}))$ time, as each point insertion results in a leaf, and all point clusters on the path to the leaf need to be updated.

For computing potentials between the points, each template point requires a different set of clusters in every iteration. The complexity of fetching a set of clusters for one point in the two point sets case is $\mathcal{O}(\log \bar{N})$. For the two point set case, the clusters for $\bar{N}$ points in the joint 2^D-tree are fetched in $\mathcal{O}(\bar{N}\log\bar{N})$ time. The complexity of fetching clusters for one point set, when the clusters are fetched from $L-1$ point sets is $\mathcal{O}(\bar{N}\log((L-1)\bar{N}))$. In every iteration[2], the clusters have to be fetched for every point set out of L, and we arrive at $\mathcal{O}(L\bar{N}\log((L-1)\bar{N}))$. Thus, the total complexity of MBGA is $\mathcal{O}(L\bar{N}\log(L\bar{N}) + L\bar{N}\log((L-1)\bar{N}))$ which further simplifies to $\mathcal{O}(L\bar{N}\log(L\bar{N}))$. □

Proposition 2. *The memory complexity of MBGA handling $L\bar{N}$ points is $\mathcal{O}(L\bar{N}\log(L\bar{N}))$, with the factor $\log(L\bar{N})$ attributable to the nodes in the 2^D-tree [4, 27].*

In practice, we restrict the depth of the 2^D-tree to 20. In addition to providing the practical upper bound on the memory, this also filters out duplicated particles which would otherwise lead to infinite splits. This implies that even for point sets with cardinalities going beyond tractable in this paper, the memory available on modern workstations will be sufficient for the proposed variant of the Barnes-Hut tree.

4.7. Gravitational Potential Energy Minimisation

MBGA finds a stable system state in a time-varying force field, *i.e.,* the state with a minimal GPE. Every change in the poses of $\boldsymbol{Y}_l$ causes changes both in the gravitational force field and $\mathbf{C}$ (cluster numbers and compositions). The alternating role of the transformed point set and force-inducing point set union makes all $\mathbf{T}_l$ mutually dependent. Since we

[2] MBGA converges within ∼20−30 iterations

Algorithm 1 MBGA: Simultaneous Point Set Alignment

```
Input: $L$ point sets $\boldsymbol{Y} = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \dots, \boldsymbol{Y}_L\}$
Output: rigid transformations $\mathbf{T}_l = \{\mathbf{R}_l, \mathbf{t}_l\}$ aligning all $\boldsymbol{Y}_l$
    into a common unknown reference frame $\mathscr{R}$
Initialisation: $\mathbf{R}_l = \mathbf{I}$, $\mathbf{t}_l = \mathbf{0}$, unit masses (per default)
while not converged do   ($t$ is iteration index)
    apply current $\mathbf{T}_l^t$ to the respective $\boldsymbol{Y}_l$ (for all $\mathbf{Y}_l$)
    build a joint $2^D$-tree on the union $g(\mathbf{T}_l^t, \boldsymbol{Y}_l)$ (Sec. 4.3)
    for every $\boldsymbol{Y}_l$ do
    minimise the GPE (6): $\mathbf{T}^{t+1} = \mathbf{T}^t \oplus \arg\min_{\mathbf{T}} \mathbf{E}_{\mathbf{C}}(\mathbf{T}) = \sum_{l=1}^{L} \sum_{i=1}^{|\boldsymbol{Y}_l|} \sum_{\mathbf{c}_j \in \mathbf{C}_{l,i}} \left( m_{\mathbf{p}_i}^l \, m_{\mathbf{c}_j} \left\| g(\mathbf{T}_l, \mathbf{p}_i^l) - \mathbf{c}_j \right\|_\epsilon \right)$
    with the Levenberg-Marquardt method [38, 39, 2]
    end for
end while
```

optimise $\mathbf{T}_l$ alternatingly in every iteration, we need to linearise in the vicinity of the current solution and iterate. The right side of (6) is a sum of residuals, and $\mathbf{E}_{\mathbf{C}}(\mathbf{T})$ can be relaxed and minimised by iteratively reweighted non-linear least squares with a Huber-norm [34]. We repeat the following transformation updates until convergence:

$$\mathbf{T}^{t+1} = \mathbf{T}^t \oplus \arg\min_{\mathbf{T}} \sum_{l,i,j} w_{i,j,l}(\mathbf{T}^t) \left\| g(\mathbf{T}_l, \mathbf{p}_i^l) - \mathbf{c}_j \right\|_\epsilon, \quad (10)$$

with the weights $w_{i,j,l}(\mathbf{T}^t) = m_{\mathbf{p}_i}^l \, m_{\mathbf{c}_j}$, the robustness threshold $\epsilon \in [10^{-4}; 0.1]$ and $\oplus$ denoting the composition of rigid poses. After denoting the residuals by $r_{l,i,j}(\mathbf{T}) = w_{i,j,l} \left\| g(\mathbf{T}_l, \mathbf{p}_i^l) - \mathbf{c}_j \right\|_2$ and collecting them into a single vector-valued function $\mathbf{F}(\mathbf{T}) : \mathbb{R}^{6L} \rightarrow \mathbb{R}^{\mathcal{O}(\bar{N}\log(L\bar{N}))} = [r_1(\mathbf{T}), r_2(\mathbf{T}), \dots, r_{\mathcal{O}(\bar{N}\log(L\bar{N}))}(\mathbf{T})]^\mathsf{T}$, the overall objective can be compactly written as $\mathbf{T} = \arg\min_{\mathbf{T}} \|\mathbf{F}(\mathbf{T})\|_\epsilon$. We minimise it with the Levenberg-Marquardt method [38, 39] provided by the *ceres* library [2], with the rotations parametrised by axis-angles.

Solution Initialisation. We initialise $\mathbf{R}_l$ as identities and $\mathbf{t}_l$ as zero vectors. Any other values can be used if available from an external process. Translation can be approximately resolved in the pre-processing step by bringing all point set centroids into coincidence (this is not compulsory). Points are initialised with unit masses if a point set is uniformly sampled. Otherwise, a variant of a mass normalisation technique or integrating additional alignment cues counterbalancing varying point densities can be applied (see Secs. 4.4, 4.5). For the convergence reasons, we keep the number of Gauss-Newton solver iterations per every update of the 2^D-tree low (one-two iterations). Otherwise, the gravitational force field will be severely deprecated while point sets are approaching each other. MBGA is summarised in Alg. 1.

5. Experiments

This section summarises the experimental results. MBGA is implemented in C++ and tested on Intel *i7-6700K* (quadcore) under the operating system Ubuntu 16.04.

5.1. Quantitative Comparisons

First, we compare MBGA to the ICP [6] (with no acceleration of correspondence search), LM-ICP [20] and BHRGA [27] approaches for the alignment of two point sets. We decimate one frame from the *sleeping 2* sequence of the SINTEL dataset [7] to 5045 points and create two rotated and translated copies of it. The point sets are composed so that each of the methods can resolve the rotation in the noise-free setting (the maximum transformation angle amounts to 24°). The correspondences across the transformed point sets are known, and we compute the average 3D error e_{3D} for L point sets defined as

$$e_{3D} = \binom{L}{2}^{-1} \sum_{\{i,j\} \in \Phi} \frac{\|g(\mathbf{T}_i, \boldsymbol{Y}_i) - g(\mathbf{T}_j, \boldsymbol{Y}_j)\|_\mathcal{F}}{\|g(\mathbf{T}_i, \boldsymbol{Y}_i)\|_\mathcal{F}}, \quad (11)$$

with Φ denoting all combinations of two point sets out of L, and $\binom{L}{2} = |\Phi|$ is the total number of combinations.

ICP, LM-ICP and BHRGA accept two point sets at a time, and we align three point sets either sequentially with the *growing reference* (*GRef*) strategy, or select one reference and align the remaining point sets to it with the *one-to-many* (*1-M*) policy. For the pairwise methods, we test all possible combinations of references and templates and report the average e_{3D} and σ, *i.e.,* the standard deviation of e_{3D}, over all runs to avoid the bias towards one of the point sets. *1-M* and *GRef* policies require six and twelve alignments for each point cloud triple, respectively. We evaluate all methods with the growing level of noise and decreasing degree of partial overlap between the input samples. In total, there are eleven different noise levels and six different degrees of partial overlaps (*i.e.,* 17 tests per method). We repeat each experiment for every noise level and degree of partial overlapping ten times. Consequently, the pairwise methods require 3060 alignments each. MBGA performs 170 alignments of point set triples since it does not have to alternate between the choice of the reference.

Uniform Noise. We add uniformly distributed noise to each point set triple in the bounding sphere ranging in the portion from 5% to 100% of the initial amount of points[3]. BHRGA [27] is a state-of-the-art gravitational rigid point set alignment method at the moment of submission. Since BHRGA and our MBGA use a 2^D-tree for the acceleration, we fix $\theta = 12.0$ for both methods in all experiments. BHRGA allows for multiple internal iterations per each external NLLS solver iteration, and we set this value to 10, as recommended in [27] for faster convergence.

The results of the pairwise methods and our MBGA are shown in Fig. 2-(A), left. The accuracy of MBGA is only slightly affected by noise, and $e_{3D} < 0.1$. MBGA outperforms LM-ICP and BHRGA starting from ≈65% of

[3] 100% of added noise means that the point set contains 50% of outliers

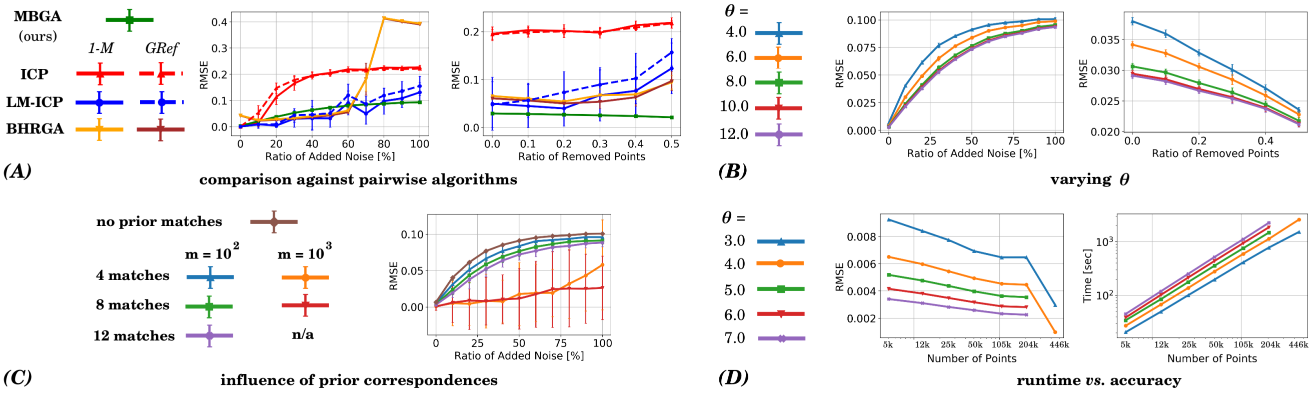

Figure 2: The summary of the quantitative results. ***(A):*** Comparison of MBGA to the pairwise point set alignment methods ICP [6], LM-ICP [20] and BHRGA [27] executed in the modes *one-to-many* (*1-M*) and *growing reference* (*GRef*). ***(B):*** The accuracy of MBGA with different θ. ***(C):*** The accuracy of MBGA with added prior correspondences. ***(D):*** Runtime *vs* accuracy experiment, *i.e.,* RMSE and the runtime as the functions of θ and point set cardinalities.

added noise. For the lower noise levels, LM-ICP achieves smaller average e_{3D} compared to ICP and MBGA, although it shows the highest $\sigma \leq 0.05$ among all methods. At the same time, our approach has higher accuracy compared to ICP already at the noise level of 20%. Moreover, its σ is orders of magnitude lower compared to the competing methods. This implies that MBGA is highly robust to noise, *i.e.,* the latter has virtually no effect on its performance. This result agrees with previous observations about the strength of the gravitational model [25, 27]. Additionally, treating all samples on par extra contributes to the accuracy because the final reference frame is obtained automatically.

Missing Data. Next, we systematically remove points from the same three input point sets and test the methods on their ability to handle missing data. The points are removed randomly from each point set, and then the intersection of points with valid correspondences across all point sets is taken to compute e_{3D}. The portion of the removed points from all three point sets varies in the range $[0.1; 0.5]$. On top of that, we add 40% of uniform noise to the samples. The results are given in Fig. 2-(A), right. MBGA shows the smallest e_{3D} and σ in all experiments. Similarly to the case with uniform noise, the accuracy of MBGA is barely affected by the missing data ratio. LM-ICP is severely affected by missing data and shows the highest variation in the mean $e_{3D} \in \{0.05; 0.14\}$ among all methods, while BHRGA is the second most accurate method.

Runtime *vs* Accuracy. We repeat the experiments with noise and missing data with different θ, see Fig. 2-(B). As expected, larger θ results in a lower e_{3D}. At the same time, increasing θ implies a higher runtime. In Fig.2-(D), we report e_{3D} and the corresponding runtime of MBGA for multiple subsampling factors and $\theta \in [3.0; 7.0]$. The dependency of the runtime from point set cardinality is very close to linear, *i.e.,* we see that with doubling the number of points, the runtime doubles likewise, and this pattern holds for all θ. e_{3D} consistently decreases with the increasing number of points in the point sets. This confirms that *all points* guide the registration and are significant.

Prior Correspondences. We repeat the test with the uniform noise with prior correspondences. In total, three modes with four, eight and twelve prior matches each are tested. Every prior correspondence in every repetition of the experiment for every noise level is drawn randomly with equal probability from the entire set of points with a valid correspondence (as one point index out of 5045). For the statistical significance, correspondences are drawn randomly ten times for each of ten repetitions of the experiment for each noise level. We report e_{3D} and σ for each mode with two different masses (weights) of prior matches, *i.e.,* 10^2 and 10^3. The results and summarised in Fig. 2-(C). With the increasing number of prior matches of the same weight, mean e_{3D} decreases. On the other hand, we observe a rise of σ with the higher weight. This suggests that the choice and distribution of prior correspondences affects e_{3D} stronger with higher weights of prior matches (since prior matches are drawn randomly, they can build a suboptimal configuration when distributed unevenly or degenerately). For too large prior weights and a growing number of prior matches, the result becomes strongly influenced by choice of prior matches (which is the case with twelve prior matches and the prior weight of 10^3).

Comparison to JRMPC. Finally, we align the clean triple of SINTEL frames with JRMPC [19] and MBGA. This experiment tests the registration accuracy on clean data. For JRMPC, $e_{3D} = 1.58 \cdot 10^{-5}$, and the e_{3D} of MBGA amounts to 0.01 with $\theta = 8.0$. It is known that methods of the gravitational class are not designed for clean data [25]. MBGA can repeat the same experiment with the non-subsampled point sets with $464k$ points each, whereas JRMPC, ver-

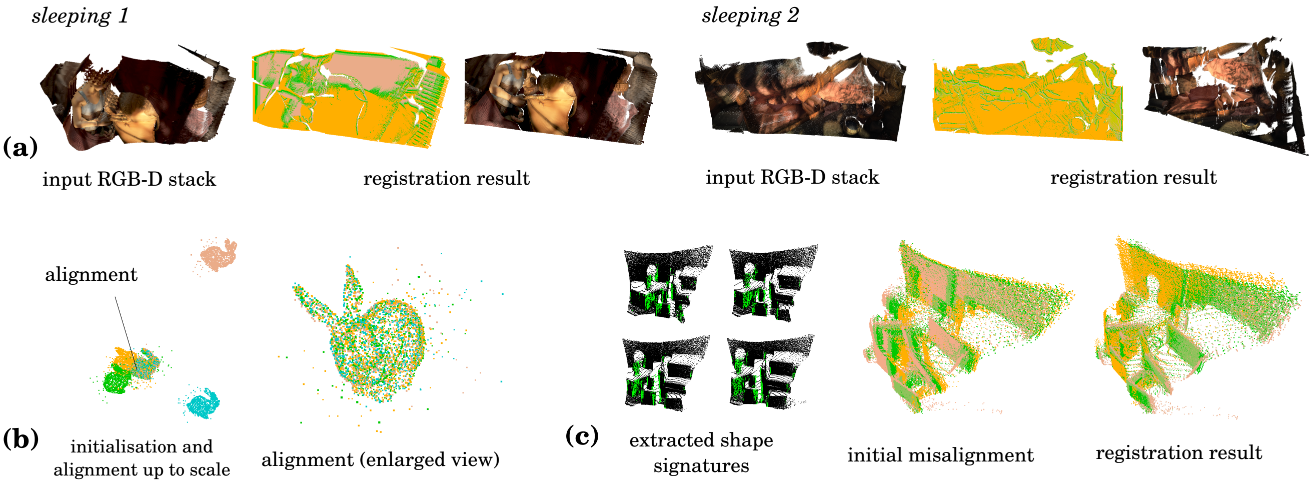

Figure 3: (a): Qualitative results on the rigid *SINTEL* [7] scenes, *i.e., sleeping 1* (4 frames, taken every 3 frames) and *sleeping 2* (5 frames, taken every other frame). Due to the camera movement, either frontal or lateral, the covered areas as well as their sampling densities differ, leading to clustered outliers. (b): This experiment with four bunnies [19] with different sampling rates and uniform noise demonstrates the robustness of our method to noise and the ability to recover translation without a pre-processing step. (c): Qualitative results on the time-of-flight data [19]. The shape signatures used for mass initialisation are shown on the left. The green colour stands for the emphasised regions, and black marks the shadowed regions with scarce geometric features.

		ICP [6]	LM-ICP [20]	BHRGA [27]*	JRMPC [19]*	MBGA (ours)*
N	e_{3D}	0.2244	0.1435	0.392	**1.5*E*-4**	**9.4*E*-2**
	σ	6.4E-3	3.7E-2	1.4E-3	**5.6*E*-5**	**1.1*E*-3**
R	e_{3D}	0.2181	0.1403	**9.6*E*-2**	**1.3*E*-3**	**2.1*E*-2**
	σ	8E-3	4.1E-2	**1.7*E*-2**	**3.6*E*-4**	**4.8*E*-4**

Table 1: Comparison of tested methods on the data with the highest noise (100% of added noise, denoted by *"N"*) and missing data (50% of randomly removed points, denoted by *"R"*) ratios. For the pairwise methods [6, 20, 27], the metrics are averaged over the *1-M* and *GRef* policies. "*" marks methods achieving $e_{3D} < 0.1$ at least once (highlighted in bold).

sion 0.9.4 [18] (in *Matlab* [40]) cannot launch due to the memory issues and high computational complexity.

Next, we repeat the experiments with high levels of noise and missing data, see Table 1. Only BHRGA [27] (missing data), JRMPC [19] and MBGA accurately resolve the transformations. Even though the proposed MBGA is not the most accurate method compared to probabilistic approaches such as JRMPC, it is the only approach which can execute on large point sets. Due to the approximation of potentials with a 2^D-tree, MBGA achieves a slightly higher e_{3D}.

5.2. Qualitative Examples

Fig. 3-(a) shows two registration examples of coloured point cloud batches from the *SINTEL* dataset [7]. We assign masses based on point intensities with a linear mapping. MBGA accurately registers batches of four and five RGB-D images. Fig. 3-(b) illustrates the further experiment with four *Stanford bunnies* [48] with added uniform noise followed [19]. The fusion result for time-of-flight data from [19] (four views) is visualised in Fig. 3-(c). In this experiment, we use the proposed in Sec. 4.5 shape signature (the emphasised areas are shown on the left). The region of interest with a head statue and surfaces are aligned, and the gravitational influence of the wall (a massive clustered outlier with barely any geometric cue) is suppressed.

6. Conclusion

We introduced the first approach for groupwise point set alignment with gravitational particle dynamics which alternates between force inducing union of point sets and the transformed point set. To the best of our knowledge, it is for the first time that a multiset alignment algorithm operates in the globally multiply-linked manner across all points in all samples. At the same time, the quasi-linear time is guaranteed thanks to a 2^D-tree which is built — due to the proposed mass shadowing policy — jointly on all point sets. As a result, we can align $5 \cdot 10^5$ points in several minutes.

MBGA outperforms several widely-used and recent pairwise methods, including BHRGA, when aligning several point clouds with large amounts of noise and missing data. Since all inputs are handled on par, our approach is alignment-path-invariant and has a loop-closure effect. The experiments have also shown that the number of points matters for the alignment accuracy. In the absence of boundary conditions, scenarios with partial overlaps with a predominant common central part work well. For partially-overlapping data, we further proposed a new shape signature for the particle mass initialisation relying on the available 2^D-tree. The GPE landscape can be studied analytically in future work to categorise all cases which MBGA can accurately resolve.

Acknowledgement. This work was supported by the ERC Consolidator Grant 4DReply (770784).

References

[1] S. J. Aarseth. *Gravitational N-Body Simulations: Tools and Algorithms (Chapter: The N-body problem).* Cambridge University Press, 2003. 3

[2] S. Agarwal, K. Mierle, and Others. Ceres solver. http://ceres-solver.org. 6

[3] Y. Aoki, H. Goforth, R. A. Srivatsan, and S. Lucey. Pointnetlk: Robust efficient point cloud registration using pointnet. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[4] J. Barnes and P. Hut. A hierarchical O(n-log-n) force calculation algorithm. *Nature*, 324, 1986. 2, 4, 5

[5] R. Bergevin, M. Soucy, H. Gagnon, and D. Laurendeau. Towards a general multi-view registration technique. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 18(5):540–547, 1996. 1, 2

[6] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 14(2):239–256, 1992. 1, 2, 3, 6, 7, 8

[7] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (ECCV)*, pages 611–625, 2012. 6, 8

[8] D. Campbell and L. Petersson. An adaptive data representation for robust point-set registration and merging. In *International Conference on Computer Vision (ICCV)*, 2015. 2

[9] Y.-P. Cao, L. Kobbelt, and S.-M. Hu. Real-time high-accuracy three-dimensional reconstruction with consumer rgb-d cameras. *ACM Trans. Graph.*, 37(5):171:1–171:16, 2018. 2

[10] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing - Special issue: range image understanding*, 10(3):145–155, 1992. 2

[11] C. Choy, W. Dong, and V. Koltun. Deep global registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[12] H. Chui and A. Rangarajan. A feature registration framework using mixture models. In *Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 190–197, 2000. 2

[13] M. Danelljan, G. Meneghetti, F. Shahbaz Khan, and M. Felsberg. A probabilistic framework for color-based point set registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[14] Y. Deng, A. Rangarajan, S. Eisenschenk, and B. C. Vemuri. A riemannian framework for matching point clouds represented by the schrödinger distance transform. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3756–3761, 2014. 3

[15] B. Eckart, K. Kim, A. Troccoli, A. Kelly, and J. Kautz. Mlmd: Maximum likelihood mixture decoupling for fast and accurate point cloud registration. In *International Conference on 3D Vision (3DV)*, 2015. 2

[16] D. Eggert, A. W. Fitzgibbon, and R. B. Fisher. Simultaneous registration of multiple range views satisfying global consistency constraints for use in reverse engineering. In *Computer Vision and Image Understanding (CVIU)*, pages 253–272, 1996. 1, 2, 3

[17] J. Elseberg, S. M. Rol, and S. A. Nüchter. A.: Comparison of nearest-neighbor-search strategies and implementations for efficient shape registration. *Journal of Software Engineering for Robotics*, pages 2–12, 2012. 2

[18] G. Evangelidis, D. Kounades-Bastian, R. Horaud, and E. Psarakis. Joint registration of multiple point sets, v. 0.9.4. https://team.inria.fr/perception/research/jrmpc/, 2018. [accessed on 05.10.2020]. 8

[19] G. D. Evangelidis and R. Horaud. Joint alignment of point sets with batch and incrementalexpectation-maximization. In *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018. 1, 2, 7, 8

[20] A. Fitzgibbon. Robust registration of 2d and 3d point sets. volume 21, pages 1145–1153, 2003. 2, 3, 6, 7, 8

[21] A. W. Fitzgibbon. Robust registration of 2d and 3d point sets, 2001. 2

[22] J. H. Friedman, J. L. Bentley, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. 3(3), 1977. 1

[23] Z. Gojcic, C. Zhou, J. D. Wegner, L. J. Guibas, and T. Birdal. Learning multiview 3d point cloud registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[24] S. Gold, A. Rangarajan, C.-P. Lu, and E. Mjolsness. New algorithms for 2d and 3d point matching: Pose estimation and correspondence. *Pattern Recognition*, 31:957–964, 1997. 2

[25] V. Golyanik, S. Aziz Ali, and D. Stricker. Gravitational approach for point set registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 3, 7

[26] V. Golyanik and C. Theobalt. A quantum computational approach to correspondence problems on point sets. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[27] V. Golyanik, C. Theobalt, and D. Stricker. Accelerated gravitational point set alignment with altered physical laws. In *International Conference on Computer Vision (ICCV)*, 2019. 2, 3, 5, 6, 7, 8

[28] V. M. Govindu and A. Pooja. On averaging multiview relations for 3d scan registration. *IEEE Transactions on Image Processing*, 23(3):1289–1302, 2014. 2

[29] L. Greengard and R. Vladimir. A fast algorithm for particle simulations. *J. Comput. Phys.*, 1987. 4

[30] M. A. Greenspan and G. Godin. A nearest neighbor method for efficient icp. In *International Conference on 3-D Digital Imaging and Modeling (3DIM)*, pages 161–168, 2001. 2

[31] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987. 2

[32] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America*, 5(7):1127–1135, 1988. 2, 3

[33] D. F. Huber and M. Hebert. Fully automatic registration of multiple 3d data sets. *Image and Vision Computing*, 21:637–650, 2001. 1

[34] P. J. Huber. Robust estimation of a location parameter. *Ann. Math. Statist.*, 35(1):73–101, 1964. 6

[35] F. Järemo Lawin, M. Danelljan, F. Shahbaz Khan, P.-E. Forssén, and M. Felsberg. Density adaptive point set registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[36] P. Jauer, I. Kuhlemann, R. Bruder, A. Schweikard, and F. Ernst. Efficient registration of high-resolution feature enhanced point clouds. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 41(5):1102–1115, 2019. 3

[37] B. Jian and B. C. Vemuri. Robust point set registration using gaussian mixture models. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(8):1633–1645, 2011. 2

[38] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics*, II(2):164–168, 1944. 6

[39] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441, 1963. 6

[40] MATLAB. *Version R2016b*. The MathWorks Inc., 2016. 8

[41] A. Myronenko and X. Song. Point-set registration: Coherent point drift. *T-PAMI*, 2010. 2

[42] A. Nealen. An as-short-as-possible introduction to the least squares, weighted least squares and moving least squares methods for scattered data approximation and interpolation. *Report of Discrete Geometric Modeling Group, TU Darmstadt*, 2004. 5

[43] A. Nüchter, K. Lingemann, and J. Hertzberg. Cached k-d tree search for icp algorithms. In *Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)*, pages 419–426, 2007. 2

[44] G. D. Pais, S. Ramalingam, V. M. Govindu, J. C. Nascimento, R. Chellappa, and P. Miraldo. 3dregnet: A deep neural network for 3d point registration. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[45] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001. 2

[46] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz. Aligning point cloud views using persistent feature histograms. In *International Conference on Intelligent Robots and Systems*, pages 3384–3391, 2008. 5

[47] I. Sipiran and B. Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27(11), 2011. 5

[48] Stanford University Computer Graphics Laboratory. The stanford 3d scanning repository. `http://graphics.stanford.edu/data/3Dscanrep/`. 8

[49] Y. Tsin and T. Kanade. A correlation-based approach to robust point set registration. In *European Conference on Computer Vision (ECCV)*, pages 558–569, 2004. 2

[50] Y. Wang and J. M. Solomon. Deep closest point: Learning representations for point cloud registration. In *International Conference on Computer Vision (ICCV)*, 2019. 3

[51] R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *International Conference on Very Large Data Bases*, page 194–205, 1998. 1

[52] Y. Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *International Conference on Computer Vision (ICCV) Workshops*, pages 689–696, 2009. 5