Particle Filtering for Registration of 2D and 3D Point Sets with Stochastic Dynamics

Romeil Sandhu Samuel Dambreville Allen Tannenbaum
Georgia Institute of Techology
School of Electrical and Computer Engineering
Atlanta, GA, USA 30322
{rsandhu, samuel.dambreville}@gatech.edu {tannenba}@ece.gatech.edu

Abstract

In this paper, we propose a particle filtering approach for the problem of registering two point sets that differ by a rigid body transformation. Typically, registration algorithms compute the transformation parameters by maximizing a metric given an estimate of the correspondence between points across the two sets of interest. This can be viewed as a posterior estimation problem, in which the corresponding distribution can naturally be estimated using a particle filter. In this work, we treat motion as a local variation in pose parameters obtained from running a few iterations of the standard Iterative Closest Point (ICP) algorithm. Employing this idea, we introduce stochastic motion dynamics to widen the narrow band of convergence often found in local optimizer functions used to tackle the registration task. Thus, the novelty of our method is twofold: Firstly, we employ a particle filtering scheme to drive the point set registration process. Secondly, we increase the robustness of the registration performance by introducing a dynamic model of uncertainty for the transformation parameters. In contrast with other techniques, our approach requires no annealing schedule, which results in a reduction in computational complexity as well as maintains the temporal coherency of the state (no loss of information). Also, unlike most alternative approaches for point set registration, we make no geometric assumptions on the two data sets. Experimental results are provided that demonstrate the robustness of the algorithm to initialization, noise, missing structures or differing point densities in each sets, on challenging 2D and 3D registration tasks.

1. Introduction

A well-studied problem in computer vision is the fundamental task of optimally aligning two point clouds with applications ranging from medical image analysis, quality control, or military surveillance and tracking [10, 1, 2, 5]. In general, point set registration is a two part problem: first, the correspondences between points across the two sets of interest must be established; then, the transformation parameters are estimated. The registration task is further complicated when there exists clutter or when information is missing.

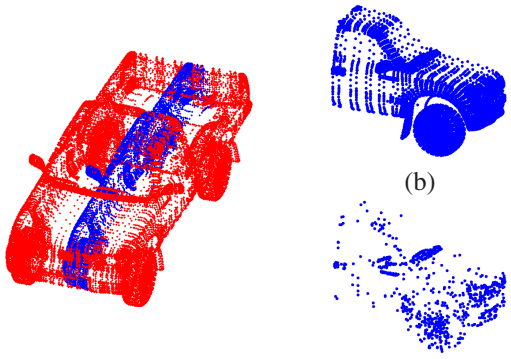


Figure 1. Common problems in point set registration. (a) Initial alignment that can yield an incorrect registration to the "wrong" side of the truck, when using iterative based techniques.(b) Dense point set. (c) Sparse point set.

A popular technique and benchmark for point set registration is the Iterative Closest Point (ICP) algorithm [1]. Given an initial alignment, ICP assigns a set of correspondences based on the L_2 distance, computes the transformation parameters, and then proceeds in an iterative manner with a newly updated set of correspondences. However, the basic approach is widely known to be susceptible to local minima. To address this issue, Fitzgibbon [6] introduces a robust variant by optimizing the cost function with the Levenberg-Marquardt algorithm. Even though this method and variants of ICP [2, 6] do improve the narrow band of convergence, they are still heavily dependent of the initial alignment, and may fail due to the existence of homologies (due to noise, clutter, outliers) within the correspon-

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

dence matrix. For instance, Figure 1 demonstrates a common problem in registration, in which a poor initial alignment can yield an incorrect registration to the "wrong" side of the truck.

The use of robust statistics and measures form a second class of point set registration algorithms [12, 20]. Representing point sets as probability densities, Tsin and Kanade [21] propose a Kernel Correlation (KC) approach using kernel density estimates. The method computes the optimal alignment by reducing the "distance" between sets via a similarity metric. An extension is considered in [13] through the use of a Gaussian mixture model. Both approaches admit solutions which may be interpreted as robust multiply-linked ICP registration schemes where the cost function is defined globally. Reducing these cost functionals, which are independent of point correspondences, allow for a wider basin of convergence. However, these methods are computationally expensive as one point set must interact with each point in the opposing set. In Section 5, we compare the KC algorithm with the particle filtering technique discussed in this paper.

To solve the dependency to the initial alignment as in standard ICP and other iterative based methods, Chui *et. al* [3] introduced a global-to-local technique, the Robust Point Matching (RPM) algorithm. The approach performs an exhaustive search that is reduced over time with an appropriate annealing schedule. However, the authors of [19] demonstrate the failure of RPM in the presence of clutter or when certain structures are missing. This is one of the main problems that is addressed in our work.

The next class of point set registration schemes, referred to as shape descriptors, introduces structural information [7, 14, 11]. While not specifically classified as a shape descriptor, the approach in [19] employs a covariance driven correspondence scheme. Although these techniques generally perform well under poor initializations, they make an underlying assumption on the point set densities. Without special consideration, registration may fail if one tries to match a sparse cloud to a dense cloud. See Figure 1 for illustration of "sparsity."

Related works that follow the approach presented in this paper are based on filtering methods [15, 16, 17]. Ma and Ellis [15] successfully pioneered the use of the Unscented Particle Filter (UPF) for point set registration. Although the algorithm accurately registers small data sets, it requires a large number of particles (5000) to perform accurate registration. Because of the large computational costs involved, the method becomes impractical for large data sets. To address this issue, the authors in [17] propose to use an Unscented Kalman Filter (UKF) approach. However, their method suffers the limitation of assuming a unimodal probability distribution of the state vector, and thus, may fail for multimodal distributions. Also, both filtering methods use a deterministic annealing schedule to drive the prediction model of the registration process. Even though the estimates are improved over time and the variance of particles is reduced, the annealing schedule itself does not incorporate any information learned online. Instead, the parameters of the annealing schedule are chosen *a priori*. Moreover, the above factors have the effect of increasing the computational complexity rendering the approaches unsuitable for real-time applications, such as tactical tracking.

As a result, the particle filtering framework presented in this paper introduces a registration scheme that attempts to address the concerns mentioned above: local point-wise accuracy, the ability to handle both sparse and dense points, robustness to noise, initialization, and extraneous structures, while still being computationally efficient. By incorporating information learned online, we propose a prediction model that diffuses particles in the direction of uncertainty of the transformation. This can also be re-interpreted as having stochastic dynamical motion, where motion is defined, in this paper, to be local variations in pose parameters obtained from running a few iterations of ICP.

This paper is organized as follows: In the next section, we discuss the particle filter and the basic local optimizer, ICP. In Section 3, we describe the registration algorithm along with the specifics of the prediction step, measurement model, and the resampling scheme. Section 4 provides numerical implementation details. Experimental results are given in Section 5. Limitations and future work are discussed in Section 6.

2. Preliminaries

In this section, we review some basic notions from the theory of particle filtering, as well as give a brief description of ICP, which we will need in the sequel.

2.1. Particle Filtering

Letting $x \in \mathbb{R}^n$, Monte Carlo methods allow for the evaluation of a multidimensional integral $I = \int g(x)dx$ via a factorization of the form $I = \int f(x)\pi(x)dx$, whereby $\pi(x)$ can be interpreted as a probability distribution. Taking samples from such a distribution in the limit yields the estimate of I that would otherwise be difficult or impossible to compute. However, generating samples from the posterior distribution is usually not possible. Thus, if one can only generate samples from a similar density $q(x)$, the problem becomes one of "importance sampling." That is, the Monte Carlo estimate of I can be computed by generating $N >> 1$ independent samples $\{x^i; i = 1, ..., N\}$ distributed according to $q(x)$ by forming the weighted sum: $I_N = \frac{1}{N}\sum_{i=1}^{N} f(x^i)w(x^i)$, where $w(x^i) = \frac{\pi(x^i)}{q(x^i)}$, represents the normalized importance weight. Thus, by employing Monte Carlo methods in conjunction with Bayesian

filtering, authors in [8] first introduced the Particle Filter. We refer the reader to [18, 4] for an in-depth discussion on Monte Carlo methods and particle filtering schemes. However, it is important to note that the *proposal density* $q(x)$ has the utmost significance in any Monte Carlo method and must be properly approximated. This is further addressed in Section 3.

Now considering $x_t \in \mathbb{R}^n$ to be a state vector, particle filtering is a technique for implementing a recursive Bayesian filter through Monte Carlo simulations. At each time t, a cloud of N particles is produced, $\{x_t^i\}_{i=1}^N$, whose empirical measure closely "follows" $p(x_t|z_{0:t}) = \pi_t(x_t|z_{0:t})$, the posterior distribution of the state given the past observations.

The algorithm starts with sampling N times from the initial state distribution $\pi_0(x_0)$ in order to approximate it by $\pi_0^N(x_0) = \frac{1}{N}\sum_{i=1}^N \delta(x_0 - x_0^i)$, and then implements Bayesian recursion at each step. With the above formulation, the distribution of the state at $t-1$ is given by $\pi_{t-1}(x_{t-1}|z_{0:t-1}) \approx \frac{1}{N}\sum_{i=1}^N \delta(x_{t-1} - x_{t-1}^i)$. The algorithm then proceeds with a **prediction step** that draws N particles from the proposal density $q(x_t|z_{0:t-1})$. With appropriate importance weights assigned to each particle, the *prediction distribution* can now be formed in a similar fashion as above, i.e. $\hat{\pi}_t(x_t|z_{0:t-1}) = \frac{1}{N}\sum_{i=1}^N w_t^i \delta(\hat{x}_t - \hat{x}_t^i)$. Then, in the **update step**, new information arriving online at time t from the observation z_t is incorporated through the importance weights in the following manner:

$$w_t^i \propto w_{t-1}^i \frac{p(z_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t^i|x_{t-1}^i, z_t)}. \tag{1}$$

From the above weight update scheme, the *filtering distribution* is given by $\tilde{\pi}_t(x_t|z_{0:t}) = \frac{1}{N}\sum_{i=1}^N w_t^i \delta(x_t - x_t^i)$. Resampling N times with replacement from $\tilde{\pi}_t$ allows us to generate an empirical estimate of the posterior distribution π_t. Even though $\tilde{\pi}_t$ and π_t both approximate the posterior, resampling helps increase the sampling efficiency as particles with low weights are generally eliminated.

2.2. Iterative Closest Point (ICP) Algorithm

Suppose now that we are given two point sets that reside in $\mathbb{R}^n$. We denote these two sets as *model* and *data*, with their respective elements $\{\mathbf{m}\}_{i=1}^{N_m}$ and $\{\mathbf{d}\}_{j=1}^{N_d}$. Assuming a rigid body transformation, $T(\vec{d}, \phi) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, for a set of data points $\vec{d}_i$ and model points $\vec{m}_j$, ICP computes a rotation matrix R and translation $\vec{t}$ that minimizes the following L_2 distance:

$$E(\vec{d}, \vec{m}, C) = \sum_{i=0}^{N_d} \|\vec{d}_j - C_{ij} R\vec{m}_j - \vec{t}\|^2, \tag{2}$$

where C is the correspondence matrix. Because these correspondences are not known *a priori*, the algorithm chooses m_j that is the closest, in the L_2 sense, to each d_i. After establishing point correspondences, the optimal transformation, $\phi = \{\vec{t}, R\}$, can be computed with minimization of the following energy functional:

$$E(\vec{d}, \vec{m}) = \sum_{i=0}^{N_d} \|\vec{d}_j - R\vec{m}_j - \vec{t}\|^2. \tag{3}$$

For the sake of brevity, we do not provide the derivation of minimizing the above functional, but refer the reader to [5, 1] for details. It is important to note here that after the transformation parameters are computed, a new "image" of the correspondence matrix is formed by applying the transformation $T(\vec{d}, \phi)$. The algorithm proceeds in an iterative manner until convergence or a stopping criterion is reached.

We also note that other local optimizers [2, 6, 9], which include objective functions, may be considered instead of the basic ICP algorithm presented here.

3. Point Set Registration Algorithm

In this section, we cast the problem of pose estimation for point sets within a particle filtering framework. By modeling the uncertainty of the transformation, the resulting approach is substantially less susceptible to local minima, and is robust to noise, clutter, and initialization.

3.1. The State Space Model

Assuming 2D and 3D point sets, we let $x_t \in \mathbb{R}^3$ and $x_t \in \mathbb{R}^6$ represent the respective state space of a rigid body transformation, i.e.

$$x(t) = \begin{pmatrix} \vec{t} \\ \vec{\theta} \end{pmatrix}(t). \tag{4}$$

For the 3D case, the translation and rotation vectors are $\vec{t} = [t_x, t_y, t_z]^T$ and $\vec{\theta} = [R_x, R_y, R_z]^T$, respectively. Similarly, for 2D point sets, the state space is $x(t) = \{t_x, t_y, \theta\}^T$. As stated above, we exploit the uncertainty of the registration in our prediction step. This forms an estimate of the state $\hat{x}_t$ from the stochastic diffusion modeling of the distribution $p(x_t|x_{t-1}, z_{t-1})$. A detailed discussion is provided in Section 3.2, where it is also shown that the basis of this prediction model can be viewed as approximation to the selection of the proposal density. After an estimate is formed, we obtain an observation at time t, which is the "image" formed under the intermediate update of the correspondence matrix, $C(T(\vec{d}, \phi))$.

Thus, the observation space is given as follows:

$$z(t) = \begin{pmatrix} \vec{t}^m \\ \theta^m \end{pmatrix}(t) \tag{5}$$

where θ^m and $\vec{t}^m$ are the measured transformation parameters.

3.2. Prediction Model

We seek a model for the prediction distribution, which can best describe uncertainty of the transformation during the registration process.

3.2.1 "Motion" Alignment Error

Inspired by [19], let us define the "motion" error for each particle $\{x^i; i = 1, ..., N\}$ that is learned online at time t as

$$e(x_{t-1}, \hat{x}_{t-1}) = x^i_{t-1} - \hat{x}^i_{t-1}, \tag{6}$$

where $\hat{x}_{t-1}$ and x_{t-1} are the predicted and measured state at $t-1$, respectively. Then the covariance of "motion" error is given as

$$S^i_{t-1} = E[e(x^i_{t-1}, \hat{x}^i_{t-1})e(x^i_{t-1}, \hat{x}^i_{t-1})^T]. \tag{7}$$

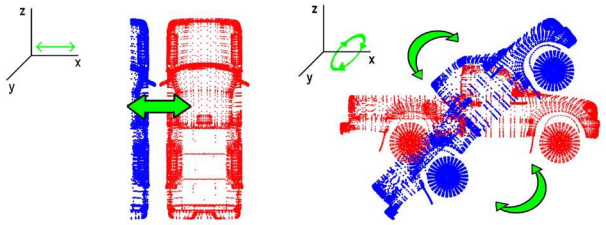
Assuming independence amongst the error parameters, S_{t-1} basically describes the variability or severity of motion in each of the principal axis for a rigid body transformation. This is shown in Figure 2. Here, a displacement is made for the pure translation and rotation case of a truck model. In these simplistic cases, the transformation estimate will be predominantly in the x-direction for (a) or about the rotational x-axis for (b).

By computing these transformation estimates from local variations in the pose parameters, we propose to explore the space described by their principal components of motion in a non-deterministic fashion. It is important to note that the model dynamics employed in conjunction with particle filtering will have an impact on the registration results (see Section 5.2.1). In the context of registration, the use of filtering dynamics enables algorithms to avoid local minima due to poor point correspondences. Depending on the basins of attraction for local minima, certain dynamics, such as constant velocity, may not help. However, by introducing stochastic motion dynamics, we allow for particles to diffuse in a manner that will avoid local minima that have large basins of attraction. In the next section, we incorporate this notion of stochastic motion into particle filtering via the proposal density.

3.2.2 Proposal Density

Now if we assume a translational prior for the proposal density, we can model the multivariate distribution with the use of a mixture of Gaussian. This is given as follows:

$$\begin{aligned} q(x_t|x^i_{t-1}, z_k) &= p(x_t|x^i_{t-1}) \\ &= \sum_{i=1}^{N} \rho_i \mathcal{N}(x_t; \mu_{t-1}; \alpha_{t-1}\sigma_{t-1} + v_{t-1}), \end{aligned} \tag{8}$$



(a) (b)

Figure 2. Simplistic case of the uncertainty in point set registration. (a) For translation, parameter estimates are largest for t_x. (b) For rotation, the estimates are largest in R_x

where ρ_i is the mixture of Gaussian weight, and μ_{t-1} and the term $\alpha_{t-1}\sigma_{t-1} + v_{t-1}$ are the mean and standard deviation describing a Gaussian distribution. The variance is the "weighted diffusion" of particles with dependency on covariance of the alignment error σ^2_{t-1} with weight α_{t-1}, and the process noise v_{t-1}. While the choice of these parameters can be adjusted to suit a particular application, we have assumed the following:

$$\begin{aligned} \rho_i &= \frac{1}{N} \qquad \mu_{t-1} = x^i_{t-1} \\ \alpha_{t-1} &= 1 \qquad v_{t-1} \approx 0 \\ \sigma^2_{t-1} &= E[e(x^i_{t-1}, \hat{x}^i_{t-1})e(x^i_{t-1}, \hat{x}^i_{t-1})^T]. \end{aligned}$$

As stated in Section 2, the selection of the proposal density is a critical issue in the design of any particle filter [18]. Equation (8) describes a prediction that diffuses stochastically in the direction of motion (through the variance term) providing sensible and robust results in the context of point set registration.

A simplification of the weight update scheme can now be made by substituting (8) into Equation (1) yielding:

$$w^i_t \propto w^i_{t-1} p(z_t|\hat{x}_t). \tag{9}$$

Finally, unlike the approach of [15, 16, 17], we have assumed the process noise to be minimal, allowing the registration to be purely driven by information learned online. Through the stochastic diffusion model, we introduce a non-deterministic process. More complex schemes of introducing process noise to help drive registration will be a subject of future work. Note that, as $t \to \infty$, the uncertainty embedded in the diffusion process is naturally reduced ($\sigma^2_{t-1} \to 0$) leading to convergence.

3.3. Measurement Model

The measurement function, $z_t = h\big(\hat{x}_t, C(t)\big)$, where $\hat{x}_t$ is a seed point (corresponding to a transformed point set), and $C(t) = C\big(T(\vec{d}, \phi)\big)$ is the "image" that becomes available at time t, can be described as follows:

1. Run minimization of the functional (3) for R iterations for each of the $\hat{x}_t^i$: the choice of R depends upon the local optimizer and the method of minimization (e.g., gradient descent, Gauss-Newton). This results in a local exploration of both the transformation and the degree of misalignment existing between point sets.

2. Compute an update of the importance weight by Equation (9) by defining $p(z_t|\hat{x}_t) \triangleq e^{\sum_{i=1}^{N_d} \|m_i - T(\vec{d},\phi)\|^2}$.

3. Build a cumulative distribution function from these importance weights. Using the generic method in [18], resample N times with replacement to generate N new samples.

4. Select the transformed point set with the minimum ICP energy (smallest L_2 distance) as the measurement. Update the path of transformation for each particle, which is used by Equation (6) to describe the "motion" alignment error.

As can be noticed from above, the posterior distribution and transformation parameters can vary drastically depending on the set of correspondences obtained at each step. Hence, we must not model the distribution as unimodal. Thereby, this justifies the use of a mixture of Gaussian to capture the wide variety of particle motions. Next, we discuss the resampling scheme, and the importance of doing gradient descent for R iterations.

3.4. Resampling

The resampling step is introduced into particle filtering schemes as a solution to "sampling degeneracy," which is unavoidable in sequential importance sampling. That is, the authors in [4] show that the variance of importance weights are only allowed to increase over time. This results in particles that are not concentrated in a region of high likelihood of the posterior distribution. Aside from the computational cost, the phenomenon of degeneracy creates poor filtering results. Thus, in this paper, we adopt the general resampling scheme, and we refer the reader to [18] for details.

Although resampling attempts to solve "sampling degeneracy," it induces another problem known as "sample impoverishment," whereby all of the particles are *only* concentrated in a single region. Because this leads to a loss of diversity for a given set of particles, an approximation to the posterior distribution is not accurate, and the registration may fail.

To address both of the above problems, careful consideration of the number of ICP iterations R must be made. By choosing R too large, we would be converging towards the local minima. This is not desirable since the state at t and $t-1$ would lose dependency. Indeed, "sample degeneracy" will occur if all of the particles tend toward one region. Likewise, if R is chosen to be too small, then most of the particles may never be associated with the high likelihood region of the posterior resulting in "sample impoverishment." In other words, *the choice of R depends on how much one trusts the system model versus the obtained measurement*. For our experiments, we have found that a choice $R = 7$ for gradient descent and $R = 3$ for Gauss-Newton's method has given robust results, as will be seen in Section 5.

4. Implementation

Here, we provide implementation and numerical details along with the pseudo-code of the algorithm described in Section 3.

4.1. Numerical Details

Experiments performed on 2D data sets are implemented by minimizing the ICP function with the gradient descent method, but we opted for the Gauss-Newton approach when working with 3D data sets. This is done to ensure the algorithm's independence with regards to minimization technique chosen. For a fast calculation of the correspondence matrix, we use a "KD tree" for the model points. Nearest neighbor searches are then easily performed for the varying data sets, as we proceed through the algorithm.

4.2. Pseudo-Code

Point Set Registration Algorithm

For t = 0

- Initialize by drawing N particles from $\pi_0(x_0)$
- Propagate particles through $z_0 = h\big(\hat{x}_0^i, C^i(0)\big)$, i.e.
  - Perform R steps of $f_{icp}(\vec{d}, \vec{m}, \hat{x}_0^i)$
  - Update Weights via $w_0^i \propto p(z_0|\hat{x}_0)$
  - Resample according to [18]
  - Compute $z_0 = \min\big\{\arg\{f_{icp}(\vec{d}, \vec{m}, \hat{x}_0^i)\}\big\}$

For t = 1,2,3,...

- Draw N particles from $q(x_t|x_{t-1}^i, z_k)$ using (8)
- Propagate particles through $z_t = h\big(\hat{x}_t^i, C^i(t)\big)$, i.e.
  - Perform R steps of $f_{icp}(\vec{d}, \vec{m}, \hat{x}_t^i)$
  - Update Weights via $w_t^i \propto w_{t-1}^i p(z_t|\hat{x}_t)$
  - Resample according to [18]
  - Compute $z_t = \min\big\{\arg\{f_{icp}(\vec{d}, \vec{m}, \hat{x}_t^i)\}\big\}$

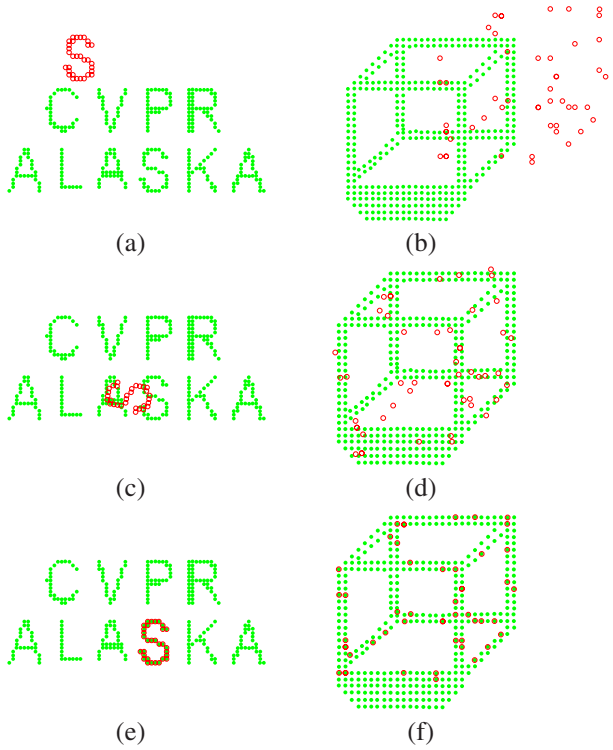


Figure 3. Examples of estimating the pose with points sets having clutter or sparseness. (a) Initial letter off-set. (b) Initial cube off-set. (c) KC word search result. (d) KC cube result. (e) Particle filtering word search result. (f) Particle filtering cube result.

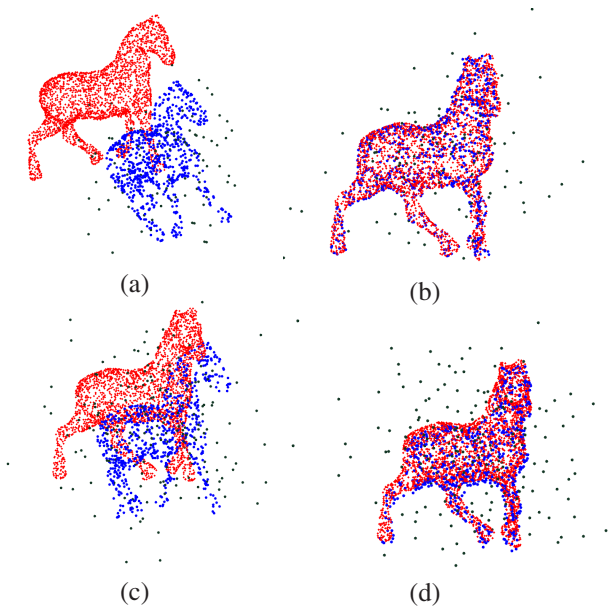


Figure 4. Examples of robustness to noise and initialization with a 3D model of a horse (Note: Noise is colored black for visualization). (a) 10% (Substitution) Gaussian zero mean isotropic noise level. (b) Registration result for 10% case. (c) 25% (Substitution) Gaussian zero mean isotropic noise level. (d) Registration result for 25% case

5. Experimental Results

We provide experimental results for both 2D and 3D point sets that undergo a rigid body transformation. Several specific tests are undertaken that demonstrate the robustness of the algorithm to **initialization**, **partial structures** (or clutter), and **noise**. Moreover, we highlight the importance of the **stochastic motion dynamics** described in Section 5.2.1. In addition to the above tests, we show that the method is robust to differing **geometric** structures among point sets.

5.1. Comparative 2D Rigid Registration

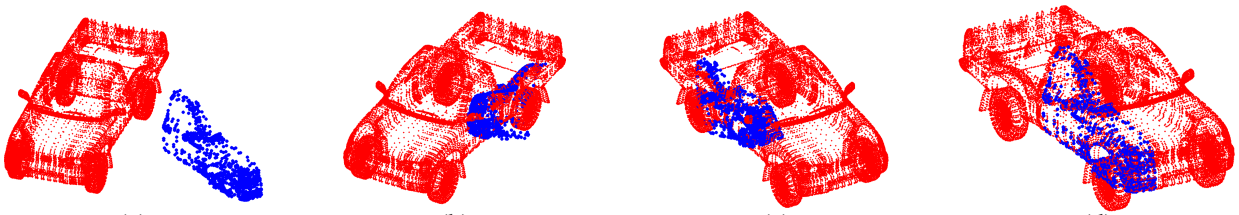
In the first set of experiments, we compare the Kernel Density Correlation (KC) approach [21] with our algorithm here. The MATLAB code of the KC algorithm is made available on the authors' website (http://www.cs.cmu.edu/~ytsin/KCReg/). In this algorithm, a global cost function is defined such that the method can be interpreted as a multiply-linked ICP approach. Rather than define a single pair of correspondences, one point set must interact with *each* point in the opposing set, thereby eliminating point correspondences altogether. Our algorithm can also be re-interpreted as a switching stochastic ICP approach where one point set interacts with only a handful of correspondences. Interestingly enough, we show next that this switching approach out performs the multiply-linked method (at least in the examples tested).

5.1.1 Partial Structures (Letter Search)

In this example, we create the words "CVPR ALASKA," and off-set the letter S with a rather large pose transformation as seen in Figure 3(a). Running the KC algorithm and the proposed approach, we attempt to recover this transformation. The task of finding a letter within a set of words is a typical partial matching problem. We performed the KC method for several varying kernel bandwidths, and found that $\sigma_{KC} = 2$ provided the most successful result. This is shown in Figure 3(c). In particular, as one increases the bandwidth σ_{KC}, the algorithm tends to align distributions spatially, which makes it particularly ill-suited for partial matching. The result of the particle filtering approach (number of particles is 200 with $R = 7$) described in this paper is shown in Figure 3(e). The transformation is recovered.

5.1.2 Geometric Assumptions (Cube)

The next experiment deals with the case of differing densities across the two point sets. While not usually stating it, many point set registration algorithms make some tacit assumptions on the point set density. In another words, they



(a) (b) (c) (d)

Figure 5. The importance of stochastic motion. (a) Initialization. (b) Result with no dynamical model. (c) Result with constant velocity model. (d) Result with stochastic motion model.

assume point sets that have a similar density or geometry around a local neighborhood for each point within their respective sets. We refer the reader again to Figure 1(b) and (c) for an illustration of differing densities. In particular, the KC algorithm uses kernel density estimates to describe the (dis)similarity between points across the two sets. To overcome poor initialization, noise, or clutter, the kernel bandwidth must be increased. However, in doing so, the kernel smoothes the point sets, which makes it increasingly difficult to discriminate among individual points when working with sparse and dense sets. To demonstrate this, we generate 50 points from the model cube, which is itself composed of 400 points. A transformation $T(\vec{d}, \phi)$ is then applied to the extracted data set. Similar to the preceding section, we tested several kernel bandwidths, and found $\sigma_{KC} = 3$ to be the optimal choice. The result is shown in Figure 3(d), where a suboptimal registration is obtained. A successful alignment is recovered using the proposed method (number of particles = 100, $R = 7$) as seen in Figure 3(f).

5.2. Rigid Registration of 3D Point Sets

Our experiments with 3D rigid registration of point sets use a 3D model of a truck and a horse. The main focus here is the importance of stochastic motion dynamics and the algorithm's inherent robustness to noise and initialization.

5.2.1 Motion Dynamics (Truck)

In this experiment, we demonstrate the importance of stochastic motion. As stated in Section 3.2, dynamics have a significant impact on the registration results. In contrast to a constant velocity model, diffusing by Equation (8) enables some particles to avoid local minima that have a large basin of attraction. It also follows that without any dynamics, registration will fail due to poor initialization, and will tend to suffer the normal fate of local optimizers.

To demonstrate this, we first extract a cloud of points from the front side of the truck as seen in Figure 5(a). A translation displacement places the point set on the opposite side with a rotation of 60° about the z-axis. Figure 5(b) shows that using no dynamics registers the localized structure to the "wrong" side of truck, while Figure 5(c) shows that a constant velocity yields an incorrect registration to the middle portion of the "right" side of the truck. Figure 5(d) presents a successful registration result of our method with diffusion. Note that the same initial conditions are used for each test, including the initial distribution of particles, highlighting the influence of the diffusion model.

5.2.2 Robustness to Noise and Initialization (Horse)

In this example, we extensively test the algorithm's performance in the case of large misalignment and large levels of noise. First, we generate 1000 random transformations, and apply them to a data set that is uniformly sampled from the model. In particular, we generate translations $\vec{t} = [t_x, t_y, t_z]$ from a normal distribution with each component having a standard deviation of 45, i.e. $\mathcal{N}(0, 45)$. This value is chosen according to the range of model points, $([-36, 33], [-67, 74], [-75, 82])$. The rotation angle θ is also chosen randomly along the z rotation axis, but from a uniform distribution $\mathcal{U}(0, \frac{\pi}{3})$.

After a transformation is applied, the data set is sampled with replacement with Gaussian zero mean noise. The applied noise is $\mathcal{N}(0, 40)$, which is again chosen with respect to the dimensions of the horse. In this experiment, we generated noise levels of 5, 10, 25, and 35 percent substitution, and then we performed 1000 tests at each of the several varying noise levels. Further, the number of particles used is 40 with $R = 3$, and the initial distribution has the same spread for each random transformation. Two examples with their registration results are shown in Figure 4.

We computed the off-set of the measured transformation with respect to the ground truth. Table 1 below presents average, standard deviation, and maximum off-sets that were computed across the 1000 tests. Notice the results are low, demonstrating the robustness of the algorithm to noise and initialization.

6. Limitations and Future Work

In this paper, we cast the problem of pose estimation for point sets within a particle filtering framework that exploits the underlying variability in the registration process. The proposed method was shown to be able to deal with partial

Noise	$\vec{t}$	θ	$\vec{R}$
5 %.	$\mu = 0.73$ $\sigma = 0.31$ max = 1.86	$\mu = 0.68\,^\circ$ $\sigma = 0.57\,^\circ$ max = 4.39 °	$\mu = 0.11$ $\sigma = .22$ max = 1.91
10 %	$\mu = 1.23$ $\sigma = 0.49$ max = 3.04	$\mu = -0.29\,^\circ$ $\sigma = 1.42\,^\circ$ max = 3.99 °	$\mu = .14$ $\sigma = 0.22$ max = 1.91
25 %	$\mu = 2.83$ $\sigma = 0.87$ max = 5.71	$\mu = -0.93\,^\circ$ $\sigma = 2.61\,^\circ$ max = 7.36 °	$\mu = .19$ $\sigma = 0.26$ max = 1.92
35 %	$\mu = 3.82$ $\sigma = 1.11$ max = 7.53	$\mu = 2.87\,^\circ$ $\sigma = 2.05\,^\circ$ max = 8.24 °	$\mu = .20$ $\sigma = 0.28$ max = 1.95

Table 1. Average, Standard Deviation, and Maximum Offset Errors for Each Noise Level

structures, poor initialization, and noise without making any geometric assumption on the point set density. Unlike [15, 16, 17], the method does not require an annealing schedule and drives the registration with information that is learned online through a stochastic diffusion model.

As compared to the KC algorithm [21], our approach only considers a set of correspondences in a switching like fashion. This enables the algorithm to correctly align point sets when dealing with partial structures or with the matching of dense and sparse sets.

The above framework can be further improved by using robust estimation techniques [12]. Specifically, taking the L_1 norm (within the measurement function) is expected to reduce the effect of noise and outliers even more (as shown e.g., in [6]). Also, our future work will focus on modifying the proposed approach to address non-rigid transformations.

7. Acknowledgments

This work was supported in part by grants from NSF, AFOSR, ARO, MURI, as well as by a grant from NIH (NAC P41 RR-13218) through Brigham and Women's Hospital. This work is part of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149. Information on the National Centers for Biomedical Computing can be obtained from `http://nihroadmap.nih.gov/bioinformatics`.

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *PAMI*, 14(2):239–256, 1992. 1, 3

[2] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *Image Vision and Computing*, 10(3):145–155, 1992. 1, 3

[3] H. Chui, A. Rangarajan, J. Zhang, and C. M. Leonard. Unsupervised learning of an atlas from unlabeled point-sets. *PAMI*, 26(2):160–172, 2004. 2

[4] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001. 3, 5

[5] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications*, 9:272–290, 1997. 1, 3

[6] A. W. Fitzgibbon. Robust registration of 2d and 3d point sets. *Image Vision and Computing*, 21(13). 1, 3, 8

[7] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Proc. Symp. Geom. Processing*, 2005. 2

[8] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/nongaussian bayesian state estimation. *IEEE Proceedings on Radar and Signal Processing*, 140(2):107–113, 1993. 3

[9] S. Granger and X. Pennec. Multi-scale em-icp: A fast and robust approach for surface registration. In *ECCV*, pages 418–432, 2002. 3

[10] B. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America Association*, 4(1):629–634, 1987. 1

[11] D. F. Huber and M. Hebert. Fully automatic registration of multiple 3d data sets. *Image Vision and Computing*, 21(7), 2003. 2

[12] P. J. Huber. *Robust Statistics*. John Wiley & Sons, New York, 1981. 2, 8

[13] B. Jian and B. C. Vemuri. A robust algorithm for point set registration using mixture of gaussians. In *ICCV*, pages 1246–1251, 2005. 2

[14] A. Johnson and M. Herbert. Using spin-images for efficient object recogntion recognition in cluttered 3-d scenes. *PAMI*, 21(5):433–449, 1999. 2

[15] B. Ma and R. E. Ellis. Surface-based registration with a particle filter. In *MICCAI*, 2004. 2, 4, 8

[16] B. Ma and R. E. Ellis. Unified point selection and surface-based registration using a particle filter. In *MICCAI*, 2005. 2, 4, 8

[17] M. Moghari and M. Abolmaesumi. Point-based rigid-body registration using an unscented kalman filter. *Transactions on Medical Imaging*, 26(12):1708–1728, Dec. 2007. 2, 4, 8

[18] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004. 3, 4, 5

[19] M. Sofka, G. Yang, and C. V. Stewart. Simultaneous covariance driven correspondence (cdc) and transformation estimation in the expectation maximization framework. In *CVPR*, volume 14, pages 234–778, 2007. 2, 4

[20] C. V. Stewart. Robust parameter estimation in computer vision. *SIAM Rev.*, 41(3):513–537, 1999. 2

[21] Y. Tsin and T. Kanade. A correlation-based approach to robust point set registration. In *ECCV*, pages 558–569, 2004. 2, 6, 8