Quasi-Monte and Data-Driven Monte Carlo Methods for Efficient Human Joint Model Fitting

Sara Greenberg University of Waterloo, ON, Canada
Alexander Wong University of Waterloo, ON, Canada
John McPhee University of Waterloo, ON, Canada

Abstract

Fitting a kinematic model of the human body to an image without the use of markers is a method of pose estimation that is useful for tracking and posture evaluation. This model-fitting is challenging due to the variation in human physique and the large number of possible poses. One type of modeling is to represent the human body as a set of rigid body volumes. These volumes can be registered to a target point cloud acquired from a depth camera using the Iterative Closest Point (ICP) algorithm. The speed of ICP registration is inversely proportional to the number of points in the model and the target point clouds, and using the entire target point cloud in this registration is too slow for real-time applications. This work proposes the use of data-driven Monte Carlo methods to select a subset of points from the target point cloud that maintains or improves the accuracy of the point cloud registration for joint localization in real time. For this application, we investigate curvature of the depth image as the driving variable to guide the sampling, and compare it with benchmark random sampling techniques.

1 Introduction

Markerless pose estimation is a useful tool for tracking applications and posture evaluation in situations in which multiple cameras or markers are considered cumbersome or obtrusive. Simple and unobtrusive pose estimation has potential in physiotherapy applications where a therapist might wish to quantitatively evaluate a patient, or a patient may wish for feedback on their performance for exercises without the need for a therapist's presence. For these applications to be successful, they require a method of pose estimation that is accurate and robust.

The Microsoft Kinect is an RGBD camera that uses trained randomized decision forests [1] to obtain a skeleton from a single depth image, a method which is very fast. Khoshelham et. al [2] evaluated the accuracy and robustness of the Kinect skeletal tracking in the context of physiotherapy exercises, and found that the variability in frame-to-frame pose estimation is about 10cm, concluding that the raw Kinect data could be used to track trends in movement but lacks the accuracy required for quantitative assessment. Wang et. al [3] have obtained smoother joint angles from the Kinect by applying the Unscented Kalman filter to repetitive motions, but no information is yet available as to the accuracy of this approach for pose estimation.

Model-based pose evaluation is an alternative to machine learning that may solve the problem arising from the variation in human physique and the large number of possible poses. The Iterative Closest Point (ICP) algorithm [4] is a method for aligning a model point cloud with a target point cloud that is well-suited for rigid body registration. Hahn et. al [5] propose a 3D model-fitting approach for a robot or a human arm using ICP and the Multiocular Contracting Curve Density (MOCCD) in a multi-camera environment and achieved results averaging less than 1cm of Euclidean distance joint location error, though this system takes up to 20 seconds per image. The speed of ICP registration is inversely proportional to the number of points in the model and the target point clouds, and using the entire target point cloud in this registration is too slow for real-time applications. Sampling the target cloud presents a potential solution.

Rusinkiewicz et. al [6] evaluated the ICP convergence speed and accuracy for a number of sampling techniques: uniform (spatial) sampling, random sampling, and normal-space sampling. Performing normal-space sampling involves bucketing the points by the position of the normal vector, and then sampling evenly from all buckets, with the goal of achieving alignment for surface features. This normal-space sampling achieved much higher accuracy and faster registration for surfaces with many small features. The error in all cases appears to converge by 10 iterations of ICP.

Another method of assessing the level of interest of surface features is curvature. Bhatia et. al [7] determine object saliency using the curvature and silhouette of object point clouds. They evaluate curvature as the difference between geodesic and Euclidean distance between points for all points in the point cloud, and found that this measure is effective and robust to clutter. Therefore, we propose a method of curvature-based sampling to fit a rigid body model to a target point cloud of the human body to achieve a fast and accurate pose estimation.

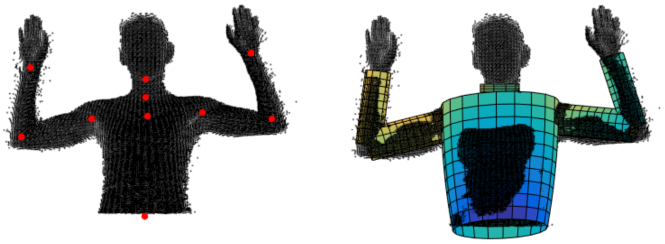

Fig. 1: Nine joints localize the model of the upper body, which consists of six cylindrical rigid bodies.

2 Methodology

We propose registering a rigid-body model of the human body to a target cloud acquired using a depth camera using ICP with curvature sampling. The focus of this work is the upper body in order to reduce complexity. We obtain the Hessian matrix of the image as a curvature estimate, and sample the target cloud at locations with high curvature at a greater frequency. A baseline percentage of the samples are acquired via Sobol sampling to decrease the risk of neglecting features.

2.1 Kinematic Model

In this multiple rigid-body model of the human upper body, a set of cylinders is created based on the locations of 9 spherical joints (torso base, torso top, top of neck, shoulders, elbows, and wrists).

The kinematic chain of this model consists of the torso to neck and upper arms, followed by the neck to head, upper arms to forearms, and finally forearms to hands. The cylinders of interest are the torso, upper arms, forearms, and neck. The remaining parts of the upper body do not affect the kinematic chain and as such are considered out of scope of this work. Fig. 1 shows the model as it relates to the joint locations and the point cloud obtained from a depth camera.

Cylinder height and radius are considered outside the scope of this work and are determined manually. Points are generated at regular intervals along the visible surfaces of these cylinders to create a model point cloud.

2.2 Iterative Closest Point

The ICP (Iterative Closest Point) algorithm, first proposed by Besl and McKay [4], is used to register a model point cloud to a target. Starting with an initial guess for the model point cloud, the ICP algorithm is as follows:

1. For each point in the model point cloud, find the nearest point in the target point cloud.
2. Estimate the rigid-body transformation that minimizes the mean squared error between point correspondences.
3. Apply the above transformation to the model point cloud.
4. Repeat for either a set number of iterations, or until the error between point correspondences is below a desired threshold.

The speed and accuracy of ICP is affected by the accuracy of the initial guess for the model. Therefore we apply ICP to each of the volumes of the upper body in the sequence of the kinematic chain, using the results of the previous registration to initialize the model of the following body part. The results of the torso registration initialize the models for the neck and upper arms, and the results of the upper arms initialize the models for the forearms.

The speed of ICP is also inversely proportional to the number of points in the model and target point clouds. As the entire target point cloud of the human upper body can contain upwards of several hundred thousand points, using the entirety in this registration is too slow for real-time applications. Instead, we only sample from the target cloud and compute the registration with a subset.

2.3 Sampling

We propose two novel point cloud sampling approaches. First, we propose the application of low-discrepancy quasirandom sampling, where discrepancy is the measure of the spacing between the selected points.

In a Monte Carlo method, we wish to find a sequence x_n in the space $D = [0,1]^s$ that satisfies

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(x_i) = \int_{I^s} f \tag{1}$$

ensuring that the sequence and projections of the sequence onto lower dimensions of D are both evenly spaced. Quasi-Monte Carlo methods produces sequences with greater uniformity across the space by minimizing the discrepancy between points.

Sampling via a Quasi-Monte Carlo method is achieved in our proposed method through the use of a Sobol sequence [8] generator as implemented in Matlab by Bratley et. al [9]. The probability P_{sobol} of selecting a point i is the probability that $i \in x_n$.

In the second sampling approach, we model the probability of sampling point i as a data-driven Quasi-Monte Carlo process. A percentage (β) of the probability is governed by Sobol sampling as described above, and the remaining percentage is the probability of selecting the point i based on the curvature at that point.

$$P(i) = (1-\beta) * P_{curvature}(i) + \beta * P_{sobol}(i) \tag{2}$$

For a depth image Z, we can approximate the curvature of a point i based on the Hessian matrix at the corresponding pixel $\bar{q}$, which is obtained as follows:

$$\Phi(\bar{q}) = \begin{bmatrix} (\Delta_x Z(\bar{q}))^2 & \Delta_x Z(\bar{q})\Delta_y Z(\bar{q}) \\ \Delta_y Z(\bar{q})\Delta_x Z(\bar{q}) & (\Delta_y Z(\bar{q}))^2 \end{bmatrix} \tag{3}$$

where Δ_x and Δ_y are gradients in the x and y directions, respectively. The probability of selecting the point i with corresponding pixel $\bar{q}$ is:

$$P_{curvature}(\bar{q}) = \frac{det(\Phi(\bar{q}))}{trace(\Phi(\bar{q}))} \tag{4}$$

Calculating the Hessian matrix (Eq.3 of the depth image of a pose, we obtain a map of curvature as shown in Fig. 2.

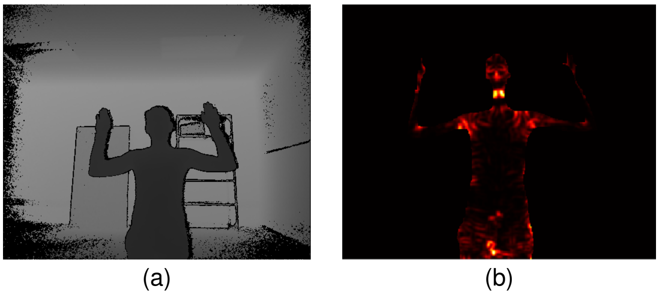

Fig. 2: We evaluate the curvature of a depth image (a) using the Hessian matrix to obtain a curvature heat map (b). Bright areas approach a curvature value of 1.

The above sampling methods are compared with uniform random sampling, as executed by Masuda et. al [10] and Rusinkiewicz et. al [6].

3 Results

To evaluate the efficiency and accuracy of curvature-aided Monte Carlo sampling, we use one pose (Fig. 2 (a)) captured using the Microsoft Kinect 2. The joints and body segments of the target cloud were determined through manual inspection. Uniform random sampling and Sobol sampling are used as a comparison.

As an initial benchmark, we perform the point cloud registration as described in Section 2.2 between the model point cloud and the entire target point cloud (217088 points). After ten repetitions, the average root mean squared error (RMSE) of the joint locations was 8.79cm and the average time was 37.0 seconds.

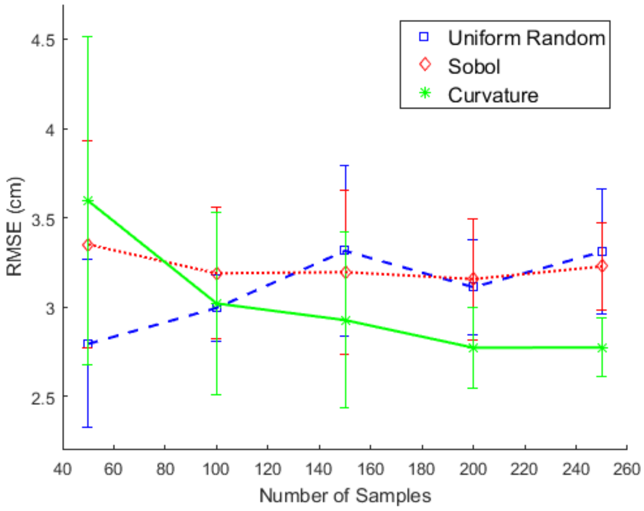

Fig. 3: Total RMSE for all 9 joints and all three sampling methods, where registration with each number of samples was performed 10 times. One standard deviation is shown at each measurement. The RMSE when using all points was 8.79cm.

For all methods of sampling, we performed 10 iterations of ICP for each body part. We vary the number of samples from 50 to 250. We use $\beta = 0.3$ in Eq. 2. Each method is repeated 10 times, and we calculate the average error and standard deviation of each registration. In Fig. 3 it can be seen that above 100 points, the curvature-aided Monte Carlo sampling method outperforms the other methods, achieving an average RMSE of under 3cm. The models generated from the resulting joint estimations at 200 points are shown in Fig. 4.

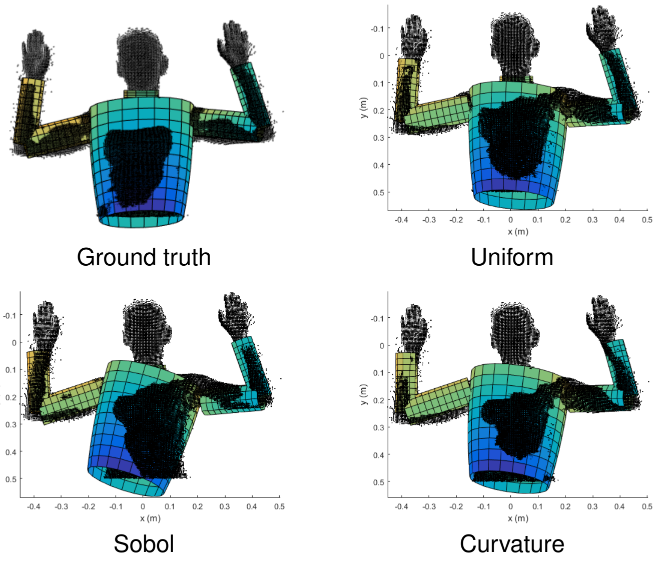

Fig. 4: Ground truth model and the models fit using all three sampling techniques with 200 sample points and 10 iterations of ICP per segment.

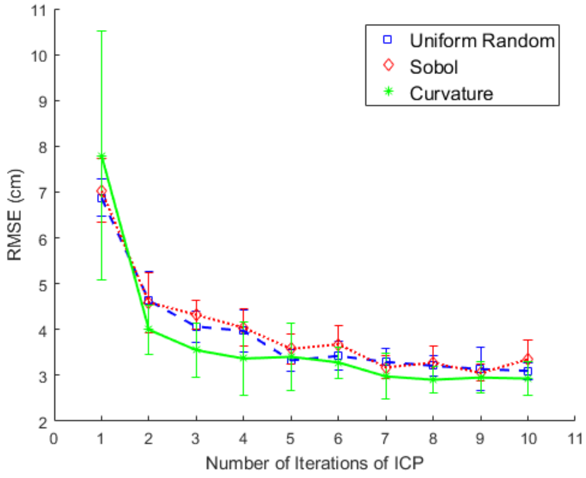

Fig. 5: RMSE after 10 iterations using 200 points shows that on average, the Curvature sampling method converges first, and after 6 iterations performs better than Uniform and Sobol sampling.

In Fig. 5, the number of points for each body part is constrained to 200 and the number of iterations of ICP is varied from 1 to 10. The average error for all methods decreases drastically after only 2 iterations, however, the Curvature sampling method achieves a lower error for all iterations proceeding the initial round. For 10 iterations per body part, there exists an approximately linear relation of 0.75 seconds per 50 points.

4 Conclusion

The results of this work indicate that data-driven sampling using curvature has potential as an approach to real-time ICP registration for multiple rigid-body systems such as the human upper body. While the current results are not real-time, additional tuning of parameters of both the model and the sampling technique is possible, and it is likely that fewer iterations of ICP are necessary than what was used to evaluate accuracy.

Future work includes testing on a larger dataset of depth images where the ground truth has been determined using a more robust technique than manual inspection, such as via a markered motion capture system. We would also like to investigate the feasibility of automatically segmenting the point cloud based on curvature to initialize the rigid-body model, as well as to automatically identify parameters such as cylinder radii and limb lengths. A database of example poses or motions may be useful to this end, such as the methods investigated by Urtasun et. al [11]. Additional processing of the data may also provide smoother and more accurate results, like the Kalman Filtering that Wang et. al [3] applied to raw Kinect data. Lastly, sampling was only applied to the target point cloud, and it is worth investigating whether applying this sampling to the model point cloud can increase the algorithm's performance.

References

[1] M. Qiao, “Report on âĂIJ Real-Time Human Pose Recognition in parts from Single Depth Images âĂİ,” *Compute*.

[2] K. Khoshelham and S. O. Elberink, “Accuracy and resolution of Kinect depth data for indoor mapping applications.” *Sensors (Basel, Switzerland)*, vol. 12, no. 2, pp. 1437–54, 2012.

[3] Q. Wang, G. Kurillo, F. Ofli, and R. Bajcsy, “Remote Health Coaching System and Human Motion Data Analysis for Physical Therapy with Microsoft Kinect,” no. 1111965, 2015.

[4] P. Besl and N. McKay, “A Method for Registration of 3-D Shapes,” pp. 239–256, 1992.

[5] M. Hahn, B. Barrois, L. Krüger, C. Wöhler, G. Sagerer, and F. Kummert, “3D pose estimation and motion analysis of the articulated human hand-forearm limb in an industrial production environment,” *3D Research*, vol. 1, no. 3, pp. 1–23, 2011.

[6] S. Rusinkiewicz and M. Levoy, “Efficient variants of the ICP algorithm,” *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pp. 145–152, 2001.

[7] S. Bhatia and S. Chalup, “Segmenting Salient Objects in 3D Point Clouds of Indoor Scenes Using Geodesic Distances,” *Journal of Signal and Information Processing*, vol. 04, no. August, pp. 102–108, 2013.

[8] I. M. Sobol, “On the distribution of points in a cube and the approximate evaluation of integrals,” *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86–112, 1967.

[9] P. Bratley and B. L. Fox, “ALGORITHM 659: implementing Sobol's quasirandom sequence generator,” *ACM Transactions on Mathematical Software*, vol. 14, no. 1, pp. 88–100, 1988. [Online]. Available: http://portal.acm.org/citation.cfm?doid=42288.214372

[10] T. Masuda, K. Sakaue, and N. Yokoya, “Registration and integration of multiple range images for 3-D model construction,” *Proceedings of 13th International Conference on Pattern Recognition*, pp. 879–883 vol.1, 1996. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=546150

[11] R. Urtasun and J. Fleet, “Implicit Probabilistic Models of Human Motion for Synthesis and Tracking,” vol. 22, no. 11, pp. 389–409.