# Reconstruction from Multiple Depth Sensors

by

Dibyendu Mondal & Kunal Agrawal

Roll No: 130050046 & 130050004

under the guidance

of

Prof. Parag Chaudhuri & Prof. Siddhartha Chaudhuri

BTP Project

Department of Computer Science and Engineering

Indian Institute of Technology Bombay

# Abstract

3D model generation from real world data using scanning systems, such as cheap depth-sensors has become widely popular in recent times owing to immense utility of such models in various industries. But such scans are often noisy and incomplete, leading to the fundamental problem of registration of a scan to a full body deformable mesh. We wish to set up a system wherein a subject will be scanned using multiple depth sensors in a room, and its corresponding body mesh will be generated robustly and correctly. This problem of registration has been attempted by many, and we spent a major portion of our time studying their solutions. We were impressed with one particular algorithm of coregistration, which we have studied in depth. To simulate the lab setup and generate synthetic partial point cloud data, we have also developed a visualization tool.

# Contents

# List of Figures

# Chapter 1

# Introduction

Many graphics applications find use of complete surface model for rendering and animation. Complete human mesh models are used extensively in industries like gaming, and film. Medical industry also uses human mesh models for studying and understanding human anatomy. But obtaining a complete model is difficult. We can scan a person standing in front of a camera with depth sensor, but such scans will always be partial, since the human body is opaque. We can mitigate this by taking multiple depth scans from a rotating camera or several cameras placed around the subject. Since we cannot expect a person to be absolutely stable during the capture process, the multiple images captured will be noisy. Extracting a full body complete mesh, from such partial noisy scans is thus not an easy problem.

This registration problem is a fundamental task, and several approaches have been tried in the past, a popular one being the Iterative Closest Point Algorithm (ICP)[3], which tries to align two scans by finding closest corresponding points between them, rigidly transforming the scans to reduce an objective function, and iterating this process till we reach a terminal condition. Since then, several modifications have been suggested to its correspondence algorithm as well as the objective functions to improve its registration accuracy and make it more robust. The SCAPE model[2] suggested using a database of pre-registered scans to train a human shape model that spans variation in both subject pose and shape.

This model was further modified by the coregistration paper [10], by introducing simultaneous model training and scan registration.

Since the literature on this problem is huge, a major portion of our time was spent in literature survey of several past methods, some of which are summarized in Chapter 2. In addition to this, we also studied the coregistration[10] paper in-depth (Chapter 3) and plan to implement it as the starting point of our future work. Obtaining a database of partial point clouds is tricky, hence we also developed a tool that helps us visualize a mesh, arrange depth-sensors around it and capture its partial point clouds from various angles. This also serves as a simulation for the lab that we plan to setup with multiple depth-sensors.

# Chapter 2

# Literature Survey

ICP is one of the most common methods for doing alignment. Many variants of ICP have been tried out in the past. Brown and Rusinkiewicz [7] present a non-rigid alignment algorithm for aligning high resolution range data in the presence of low-frequency deformations such as those caused by scanner calibration error. Traditional iterative closest points (ICP) algorithms, which rely on rigid-body alignment, fail in these cases because the error appears as a non-rigid warp in the data. They use a thin-plate spline to represent the warp, based on feature correspondences computed using a hierarchical iterative closest points (ICP) method. Their algorithm combines the robustness and efficiency of ICP with the expressiveness of thin-plate splines to align high-resolution scanned data accurately. The principal drawback is that very high resolution data is required for accurate warps. This is because closest point computations are made only to vertices in the target mesh; if those vertices are spaced too far apart, the feature correspondence will be skewed. The resulting warp still provides a good alignment, but is less smooth and warps non-overlapping areas of the source mesh more than necessary.

Mitra et al. [13] improved upon ICP by developing an objective function that is a second order approximant to the squared distance between the model and the data. This incorporates higher order information about the surfaces represented by the point clouds, such as local curvatures. When the model and data PCD's are close, their convergence rate is similar to ICP; moreover their algorithm has a relatively stable behavior when the initial displacement is large. Their algorithm mainly improves alignment when the model and the point cloud are far apart, however this is not beneficial to us since we capture the input models using our set-up and therefore can place the model and the obtained point clouds fairly close. Further, they mention the inability of their algorithm to simultaneously register multiple point clouds, a feature that we require since we are using multiple depth sensors, and will obtain multiple partial point clouds of the same person.

Another modified version of ICP was developed by J.-D. et al. [11] for registering facial point data obtained using CT scans of a patient, to provide medical assistance and pre-operative training. They modified the correspondence search in ICP by proposing the use of Adaptive Dual Approximate K-D Tree (ADAK-D). Approximate K-D (AK-D) tree was proposed by Greenspan [8] for the same, but it sometimes produces false nearest neighbor points since it uses only one projection plane for partitioning a k-dimensional space to build the node tree. ADAK-D tree uses AK-D tree twice in two different geometrical projection orders for determining the true nearest neighbor point. They also improved the objective function of ICP, by modifying the soft-shape-context ICP (SICP) algorithm proposed by Liu and Chen [12]. In SICP, for each point they generate a bin histogram and a low-pass filter is used to smooth the neighbor histogram values. Since bin histogram generation at each point in 3D point data is computationally expensive, they just generate it for the centroid points of reference and scan data. But since their system uses high quality CT scan data, they do not have a noise analysis, which would be very significant in our data.

Another variation of ICP proposed by Haehnel et al. [9] focuses on registration of non-rigid objects. They transform scans into Markov random fields, where nearby measurements are linked by a (nonlinear) potential function. All links are soft, and bending them incurs penalty. They solve this optimization problem by Taylor series expansion (linearization), and followed by a coarse-to-fine hierarchical optimization technique for carrying out the optimization efficiently. Since human bodies are non-rigid, this seems to be a good approach, but in recent times, much better and relatively simpler ways of capturing the non-rigid aspect of human shape have been developed which use a database of human shapes to learn such a model.

Most variants of ICP alternate between closest point computations to establish correspondences between two data sets, and solving for the optimal transformation that brings these correspondences into alignment. A major difficulty for this approach is the sensitivity to outliers and missing data often observed in 3D scans. Most practical implementations of the ICP algorithm address this issue with a number of heuristics to prune or reweight correspondences. Bouaziz et al. [5] propose a new formulation of the ICP algorithm that avoids these difficulties by formulating the registration optimization using sparsity inducing norms. They propose a sparse optimization problem that automatically learns the separation between data and outliers. The major drawback is that this algorithm assumes that the aligning transformation is rigid.

Vetter et al. [1] show how to extend the ICP framework to nonrigid registration while retaining the convergence properties of the original algorithm. The resulting optimal step nonrigid ICP framework allows the use of different regularisations, as long as they have an adjustable stiffness parameter. They present an algorithm using a locally affine regularisation which assigns an affine transformation to each vertex and minimises the difference in the transformation of neighboring vertices. They have shown that for this regularisation the optimal deformation for fixed correspondences and fixed stiffness can be determined exactly and efficiently.

Pauly et al. [6] introduce 2D and 3D registration algorithms. They formulate registration as the minimization of an energy, which is the sum of matching energy and prior energy. They combine 2D image registration and 3D geometry registration to best utilize the data provided by the RGB-D sensor. The matching energy measures the closeness between the source image and the target image wrapped onto the deformed grid. Prior energy is a combination of global and local rigidity prior.

There are also several methods that use a database to learn a human shape model that spans variation in both subject shape and pose. One such method is introduced by Anguelov et al. [2] in their SCAPE paper. The method represents the human shape model such that it incorporates both articulated and non-rigid deformations. They learn a pose deformation model that derives the non-rigid surface deformation as a function of the pose of the articulated skeleton. They also learn a separate model of variation based on body shape. Both these models together make up the SCAPE model, which can produce 3D surface models with realistic muscle deformation for different people in different poses, when neither appear in the training set. They even demonstrate the model's performance for partial scan completions, which is in line with what we want to achieve.

The SCAPE model uses as its input, a collection of registered meshes, which it obtains by registering scans using other techniques. Another paper by Hirshberg et al. [10] removes this dependency by approaching modeling and registration together.This model serves to regularize how the template mesh can deform, to avoid impossible deformations of the template. They minimize a single objective function, to reliably obtain high quality registration of noisy, incomplete scans, while simultaneously learning a highly realistic articulated body model. This model greatly improves robustness to noise and missing data. This model will be a major portion of what we plan to implement in the future, thus we studied it deeply. It's further details are mentioned in the below sections.

# Chapter 3

# Algorithm : Coregistration

## 3.1 Model : BlendSCAPE

The SCAPE model defines how to deform a triangulated human-shaped template mesh $T^*$ to take on different poses and identities. $T^*$ is pre-segmented into parts, connected by a kinematic tree structure, obtained from the SCAPE dataset. The relative joint angles are represented as Rodrigues vectors. Lets represent all the relative joint rotations collectively as $\theta$. $R(\theta)$ represents the absolute rotation of triangles in a part, which is found using the kinematic tree structure. $D$ represents the deformations that transform $T^*$ into the shape of a specific person, and $Q(\theta)$ represents the non-rigid deformations of the triangles of $T^*$ that capture shape change as a function of pose $\theta$.

In order to deform the template, $T^*$ is decomposed into "un-stitched" triangles, following which we apply pose deformation as a $3 \times 3$ matrix $R_f(\theta)$, a person's body shape deformation as $3 \times 3$ matrix $D_f$, and finally pose dependent shape changes by a $3 \times 3$ matrix $Q_f(\theta)$. Here $f$ indexes the un-stitched triangles. Here is the formula:

$$T_f = R_f(\theta) D_f Q_f(\theta) T^* \tag{3.1}$$

Following this the edges are stitched back together, as is done in [2]. This however can lead to folds and creases near joints. For this reason, the BlendSCAPE model is used, where the triangle rotations are represented by a linear blend, $B_f(\theta)$, given by:

$$B_f(\theta) = \sum_i w_{fi} R^i \tag{3.2}$$

where $i$ indexes the parts of the mesh. $B_f(\theta)$ and $Q(\theta)$, both work simultaneously to reduce the mesh artifacts. $Q$ is represented as as linear function of the pose vector, given by:

$$Q(\theta) = Q^0 + \sum_c \theta_c Q^c \tag{3.3}$$

where $c$ indexes the $c$th element of the pose vector and $Q^c$ is the linear coefficient.

## 3.2 Coregistration

Coregistration aligns a triangulated template mesh to a corpus of 3D scans while simultaneously training a BlendSCAPE model. To train the model, we must estimate a pose $\theta_s$ for each scan in the corpus, a shape $D_p$ for each person in our corpus, and a single linear pose-dependent deformation model $Q(\theta)$. Before training we initialize $D$ and $Q$ to identity. At the start, we roughly align the template to the scan, by posing and scaling the untrained BlendSCAPE model. This can be done using basic ICP or via landmarks.

To fit the deformed template on the surface of scan $s$, we evaluate the data term $E_S$, given by:

$$E_S(T; S) = \frac{1}{a_S} \int_{x_s \epsilon S} \rho \left( min_{x_t \epsilon T} ||x_s - x_t|| \right) \tag{3.4}$$

where, $\rho$ is the Geman-McClure robust error function $\rho(x) = \frac{x^2}{\sigma^2 + x^2}$, $S$ is the scan surface, $a_S$ is the scans surface area, and $T$ is the surface of the aligned template. We approximate the data error using a fixed set of locations xs, uniformly sampled over the surface of the scan $S$.

Next, in a coupling term $E_C$, we penalize the difference between the aligned template and current model:

$$E_C(T,\theta,D,Q) = \sum_f a_f \left|\left| T_f - B_f(\theta) D_f Q_f(\theta) T_f^* \right|\right|_F^2 \tag{3.5}$$

where $T_f$ represents the pair of edge vectors of the un-stitched triangle $f$ of $T$, $B_f(\theta)D_fQ_f(\theta)T_f$ is the corresponding un-stitched triangle of $M(\theta, D, Q)$, and $a_f$ is the area of $f$ on the template mesh, $T$. The squared Frobenius norm is used to measure this difference.

Additionally, we use regularization terms to constrain the body shape deformations, $D$, and the pose-dependent deformation model, $Q$. The first term promotes spatial smoothness of the deformations, D, that map the template mesh to an observed person. The second term penalizes the magnitude of the effect of the pose-dependent deformation model.

$$E_D(D) = \sum_{adjacent\ faces\ i,j} a_{i,j} \frac{||D_i - D_j||_F^2}{h_{ij}^2} \tag{3.6}$$

$$E_Q(Q) = \sum_{faces\ f} a_{i,j} \left( ||Q_f^0 - I||_F^2 + \sum_c ||Q_f^c||_F^2 \right) \tag{3.7}$$

where $h_{ij}$ is the distance between the centroids of template triangles $i$ and $j$, $a_f$ is the area of triangle $f$, and $a_{ij} = \frac{a_i + a_j}{3}$ is the area of the diamond-shaped region defined by the centroids of triangles $i$ and $j$ and the endpoints of their shared edge.

Since $D$ and $Q$ are not known, coregistration seeks to align all scans in parallel while simultaneously solving for $D$ and $Q$ across scans. Summing over all scans and adding the model regularization yields the following coregistration optimization problem:

$$\begin{aligned} \min_{T^k,\theta^k,D^P,Q} & \sum_{scans\ k} [E_S(T^k; S^k) + \lambda_C(E_C(T^k, \theta^k, D^P, Q))] \\ & + \lambda_C[\lambda_D \sum_P E_D(D_P) + \lambda_Q(E_Q(Q))] \end{aligned} \tag{3.8}$$

Here $p$ indexes people, $k$ indexes scans, and $p_k$ identifies the person in each scan. The $\lambda$s control the relative influence of terms. $\lambda_C$ is particularly important; it controls how much the alignments can deviate from the model.

## 3.3 Optimization

By fixing $D^p$, and $Q$, we can decouple the scans, and we can minimize 3.8, by solving the following non-linear problem for each scan:

$$\min_{T^k,\theta^k} E_S(T^k; S^k) + \lambda_C(E_C(T^k, \theta^k, D^{p_q}, Q)) \tag{3.9}$$

These sub-problems can be solved using MATLABs lsqnonlin (MathWorks, Natick MA).

If we fix all $T^k$ and $Q()$, minimization with respect to each persons $D^p$ is an independent linear least squares problem for each person $p$. Similarly, with all $T^k$ and $D^p$ fixed, minimization with respect to $Q_f()$ is an independent linear least squares problem for each triangle $f$.

The paper suggests performing optimization in stages, so as to move from a crude fitting to tight fitting, by gradually increasing $\lambda_C$.

# Chapter 4

# Our Work

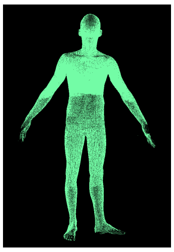
We used MPI-FAUST dataset [4] as synthetic data for our models. It contains 200 depth scans as testing data and 100 depth scans as training data. Initially, we tried out the PCL library [1] to load a model in PCD (Point Cloud Data) format. We then made filters to create partial point clouds of the model. We applied the Iterative Closest Point (ICP) algorithm [3] to register the partial point cloud to the whole model. Figure 4.0.1 shows the template, the partial point cloud and the aligned point cloud.

We then tried the 2D/3D Registration algorithm [6] using the code provided by them [2]. It performs non-rigid registration of the models. Figure 4.0.2 shows step-wise registration process of a human.

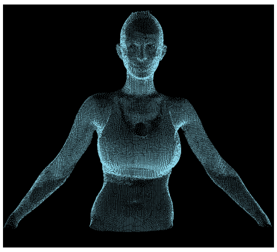
We created a visualization tool in which we can load any ".obj" file from the dataset. We added 4 view-ports through which we can view the loaded model. Figure 4.0.3a shows the model in our tool. We have added Menu Actions (Figure 4.0.3b) to add features like loading a new model, saving the partial point clouds, from the 4 view-ports, of the currently loaded model. We also added an action through which one can view the saved partial point clouds (Figure 4.0.3d)

[1] `http://pointclouds.org/`

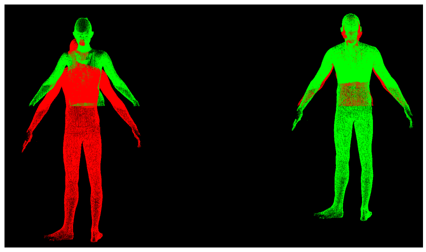
[2] `http://lgg.epfl.ch/publications/2014/2d3dRegistration/code.zip`



(a) Template Mesh

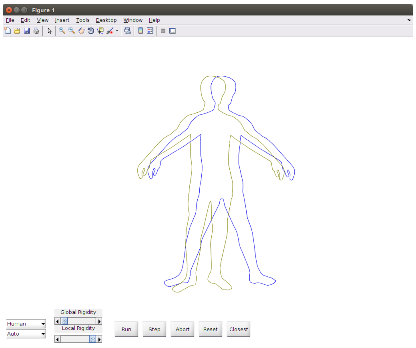


(b) Partial Point Cloud

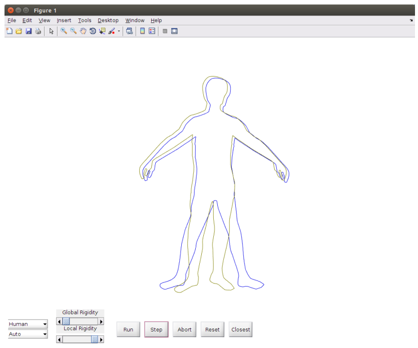


(c) Aligned Point Cloud

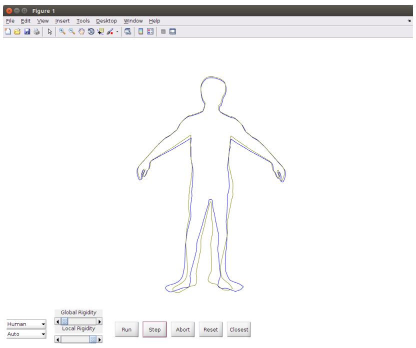
Figure 4.0.1: PCL ICP Alignment



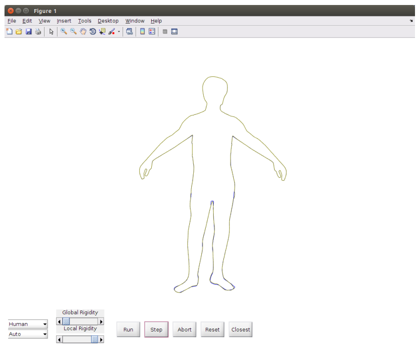

(a) Initial Point Clouds




(b) Registration Step

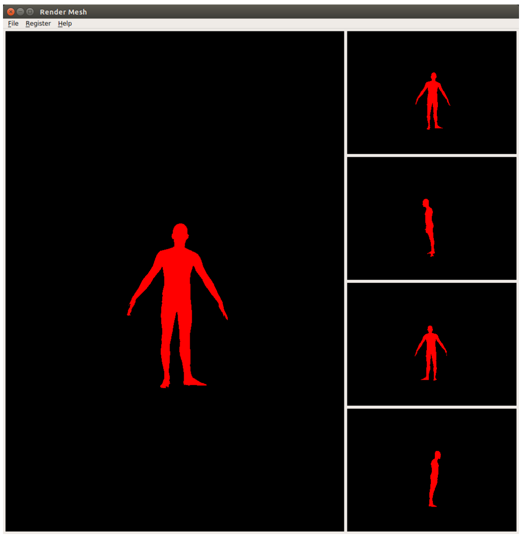



(c) Registration Step

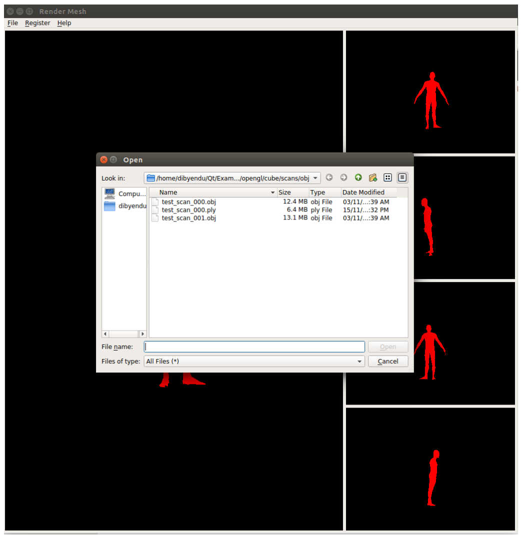



(d) Registered Point Cloud

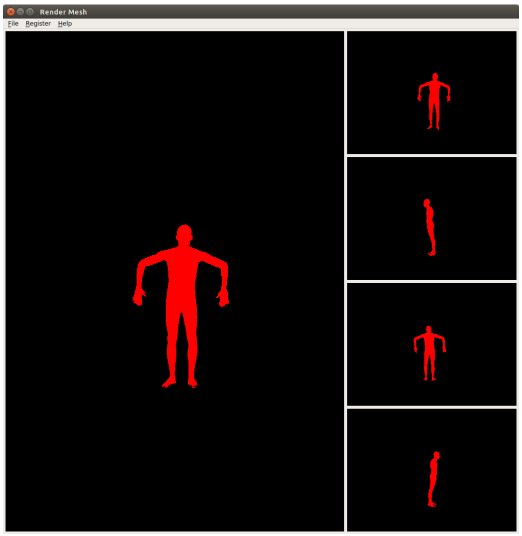
Figure 4.0.2: Registration by Pauly et al. [6]



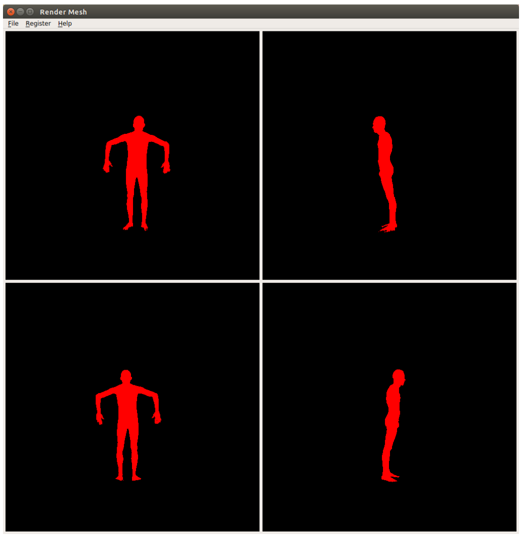

(a) Model 1




(b) Open Action




(c) Model 2




(d) Partial Point Clouds

Figure 4.0.3: Visualization Tool

# Chapter 5

# Future Work

Our next step is to implement the coregistration algorithm given by [10] and register the point clouds generated by our tool. The coregistration algorithm works specifically for the individuals present in the database. We will try to generalize it further. One method is to use PCA to learn body models of several individuals using multiple scans of a person. The current SCAPE [2] model learns a shape space via PCA focusing on single scans of individuals only.

We are also setting up a room in ViGIL lab with 4 (or possibly more) Kinects with proper fixed stands. We will take partial scans from the Kinect device and feed them as inputs to our registration algorithm. Here our main aim will be correctness and robustness of our system.

# Bibliography


[1] B. Amberg, S. Romdhani, and T. Vetter. Optimal step nonrigid icp algorithms for surface registration. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 408–416. ACM, 2005.

[3] P. J. Besl and N. D. McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[4] F. Bogo, J. Romero, M. Loper, and M. J. Black. Faust: Dataset and evaluation for 3d mesh registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3794–3801, 2014.

[5] S. Bouaziz, A. Tagliasacchi, and M. Pauly. Sparse iterative closest point. In *Computer graphics forum*, volume 32, pages 113–123. Wiley Online Library, 2013.

[6] S. Bouaziz, A. Tagliasacchi, and M. Pauly. Dynamic 2d/3d registration. In *Eurographics (Tutorials)*, page 7. Citeseer, 2014.

[7] B. J. Brown and S. Rusinkiewicz. Non-rigid range-scan alignment using thin-plate splines. In *3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Proceedings. 2nd International Symposium on*, pages 759–765. IEEE, 2004.

[8] M. Greenspan and M. Yurick. Approximate kd tree search for efficient icp. In *3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. Fourth International Conference on*, pages 442–448. IEEE, 2003.

[9] D. Haehnel, S. Thrun, and W. Burgard. An extension of the icp algorithm for modeling nonrigid objects with mobile robots. In *IJCAI*, volume 3, pages 915–920, 2003.

[10] D. A. Hirshberg, M. Loper, E. Rachlin, and M. J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *European Conference on Computer Vision*, pages 242–255. Springer, 2012.

[11] J.-D. Lee, C.-H. Huang, L.-C. Liu, S.-S. Hsieh, S.-P. Wang, and S.-T. Lee. A modified soft-shape-context icp registration system of 3-d point data. In *International Conference on Neural Information Processing*, pages 723–732. Springer, 2007.

[12] D. Liu and T. Chen. Soft shape context for iterative closest point registration. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 2, pages 1081–1084. IEEE, 2004.

[13] N. J. Mitra, N. Gelfand, H. Pottmann, and L. Guibas. Registration of point cloud data from a geometric optimization perspective. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 22–31. ACM, 2004.