# A Generalisation of the ICP Algorithm for Articulated Bodies

Stefano Pellegrini[a,b], Konrad Schindler[b], Daniele Nardi[a]

[a] Dip. di Informatica e Sistemistica, Sapienza University Rome
[b] Computer Vision Laboratory, ETH Zürich
{stefpell|konrads}@ee.ethz.ch, nardi@dis.uniroma1.it

### Abstract

The ICP algorithm has been extensively used in computer vision for registration and tracking purposes. The original formulation of this method is restricted to the use of non-articulated models. A straightforward generalisation to articulated structures is achievable through the joint minimisation of all the structure pose parameters, for example using Levenberg-Marquardt (LM) optimisation. However, in this approach the aligning transformation cannot be estimated in closed form, like in the original ICP, and the approach heavily suffers from local minima. To overcome this limitation, some authors have extended the straightforward generalisation at the cost of giving up some of the properties of ICP. In this paper, we present a generalisation of ICP to articulated structures, which preserves all the properties of the original algorithm. The key idea is to divide the articulated body into parts, which can be aligned rigidly in the way of the original ICP, with additional constraints to keep the articulated structure intact. Experiments show that our method reduces the residual registration error by a factor of $\approx 2$.

## 1 Introduction

The ICP algorithm, in its original formulation [1], registers two point sets (the *model* and the *data*) when the point-to-point correspondences among these two sets are unknown. This technique has been extensively used in the computer vision and robotic communities for registration and tracking purposes: a model consisting of 3D points is matched against static or dynamic range data, e.g. coming from a stereo camera or from a laser range finder. Due to its simplicity and efficiency, many different variants and extensions have been proposed [11]. In particular, ICP has been employed by several authors for articulated body matching and tracking. A straightforward generalisation of ICP algorithm to articulated models is achievable through the joint minimisation of all the pose parameters of the structure, for example using the Levenberg-Marquardt (LM) method [4]. Note however that in this generalisation the aligning transformation is estimated in a way, which is conceptually different from the original formulation: vanilla ICP would recover the registration of non-articulated bodies in a *single iteration*, if the correspondences were exact. The optimisation with LM (or similar local optimisers) needs more than one iteration, even with rigid non-articulated bodies and known correspondences.

BMVC 2008 doi:10.5244/C.22.87

We argue that this apparent subtlety makes an important difference: the LM-based algorithm for articulated bodies fails to capture one of the most important characteristics of ICP, and as a consequence suffers more from weak local minima. This problem gets worse when the *distance* between the initial model and the data increases (see Sec. 4). To overcome the difficulties arising from local minima, many authors have extended the original formulation at the cost of giving up additional properties of the ICP method.

A way to use ICP with articulated bodies is not to strictly enforce the joint constraints in the model. Soft constraints rather than hard ones can be used, as in [10]. In this way, the error function to be minimised is a weighted combination of residual distance of the correspondences and a cost term for not respecting the joint constraint. Another solution based on relaxing the joint constraints is in [3], where each rigid part of the model is matched separately to the data. The solution thus obtained is likely not to be a valid one, as the joint constraints are not enforced. Thus, the unconstrained solution is projected to a constrained state space to yield a valid solution.

Another way of tackling the problem is that of employing a different model. In [8] the joint constraints in the articulated model are enforced by adding model-to-model artificial correspondences to the model-to-data ones. The additional correspondences link each pair of connected model parts. This gives rise to constraints for each joint. The strength of each of these constraints is determined by the number of such pairs for a joint. A different model is proposed by [6], where a deformable model is employed rather than an articulated one made of rigid parts. In this way, even though the correspondences are still estimated in an ICP fashion, the model no longer deforms in an articulated way.

Another interesting solution is proposed in [2], where a hand is tracked in stereo image sequences. An improved gradient descent method is used to minimise the error function. This function does not depend on all the correspondences, but only on a randomly sampled subset. In this way, the error function changes from iteration to iteration and so do the local minima, while the global optimum is shared by all the samples.

In [5], it shown that, with a single rigid body, using LM instead of the closed form solution does not worsen the convergence properties of the registration method. We show in the following sections that this is not the case when dealing with many degrees of freedom.

In this paper we present a generalisation of ICP to articulated structures, which preserves all the properties of the original method. The key idea is to closely follow the basic ICP strategy, and perform a complete rigid alignment of a *part* of the model in each iteration. In Section 2 we will introduce the articulated model, followed by a description of the algorithm in Section 3. In Section 4 we will show the experimental results obtained when applying our algorithm and the conclusions will be drawn in Section 5.

## 2 Model

An articulated body model $\mathtt{M}$ is composed of rigid parts $\mathsf{p}_1, \mathsf{p}_2, \dots \mathsf{p}_{N_P}$. Each part $\mathsf{p}_i$ has a point of articulation or joint $\mathsf{j}_i$ through which the part is connected to another part.

We restrict our attention to *open* kinematic structures. In other words the underlying graph – with a node for each rigid part and an edge between any two connected parts – is an undirected tree. We arbitrarily choose one of our rigid parts to be the root node $\mathsf{p}_r$. By convention, the corresponding joint $\mathsf{j}_r$, which has no parent, is connected to the world

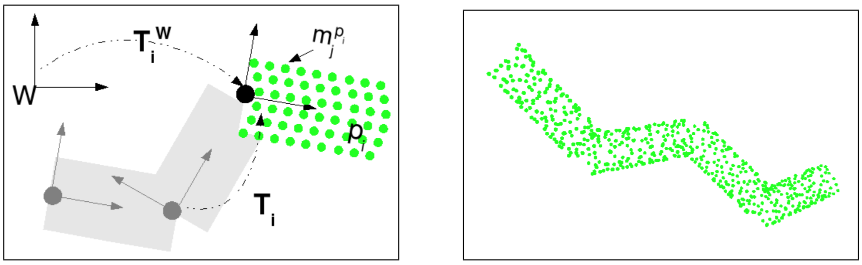


Figure 1: *Left:* The articulated model with part $\mathsf{p}_i$ highlighted. *Right:* An instance of a simple articulated model made of four semi-cylindrical rigid parts.

coordinate system. For ease of notation, we define the operator $\bigwedge(\cdot)$, which takes a part index as input and returns the index of the part's parent as output (so $\mathsf{p}_{\bigwedge(i)}$ is the parent of $\mathsf{p}_i$).

For each part $\mathsf{p}_i$ there is a rigid transformation $\mathbf{T}_i(\theta_i)$, which specifies the part's translation and rotation w.r.t. its parent (for notational simplicity, we will omit the dependence on the parameters $\theta$). A general rigid transformation has 6 parameters, $\theta = [\omega, \phi, \kappa, x, y, z]$. Depending on the type of joint, the number of actual d.o.f.'s of a part can vary. For example, if $\mathsf{j}_i$ is a *spherical* joint, there are only 3 d.o.f.'s $\theta_i = [\omega, \phi, \kappa]$, while the remaining parameters are constant.

For the root $\mathsf{p}_r$, the reference system $\mathbf{T}_r$ specifies the displacement from the coordinate system. We assume a fully unconstrained joint with 6 d.o.f.'s for the root part, even though in some applications it might be useful to model it as a constrained joint. The absolute pose of a part $\mathsf{p}_i$ in the world coordinate system can be obtained by concatenating the transformations along the kinematic chain from the root part to $\mathsf{p}_i$,

$$\mathbf{T}_i^W = \mathbf{T}_r \dots \mathbf{T}_{\bigwedge(i)} \mathbf{T}_i \,. \tag{1}$$

For each $\mathsf{p}_i$ a (possibly empty) set of points $\{\mathbf{m}_1^{\mathsf{p}_i} \dots \mathbf{m}_{N_i}^{\mathsf{p}_i}\}$ is given. These points determine the *shape* of the part (see Fig.1) and are specified in homogeneous coordinates with respect to the part's local coordinate system. This means that the position of a point $\mathbf{m}_j^{\mathsf{p}_i}$ in world coordinates is $\mathbf{T}_i^W \mathbf{m}_j^{\mathsf{p}_i}$.

Finally, we will require the following definition: by removing a single joint $\mathsf{j}_i$, any open articulated structure $\mathtt{M}$ can be split into two branches, one of which contains the root $\mathsf{p}_r$. We will call the branch containing the root the *base* branch $\mathtt{M}_b^i$, and the other one the *outer* branch $\mathtt{M}_o^i$. In the special case of "splitting" at the root, we define $\mathtt{M}_b^r = \mathtt{M}_o^r$, i.e. both branches correspond to the entire model.

# 3 Algorithm

Given a data set $\mathscr{D}$ of world points $\{\mathbf{d}_1, \mathbf{d}_2, \dots \mathbf{d}_{N_\mathscr{D}}\}$ and an articulated body model $\mathtt{M}$ to be aligned with it, the objective function to be minimised is formally defined as

$$E(\theta'_1 \dots \theta'_{N_\mathsf{p}}) = \sum_{s=1}^{N_P} \sum_{i=1}^{N_{\mathsf{p}s}} \min_k \|\mathbf{T}_i^W \mathbf{m}_i^{\mathsf{p}_s} - \mathbf{d}_k\|^2 \,. \tag{2}$$

In words: the sum of squared distances between all pairs of corresponding points shall be minimised. Since the correspondences are unknown, we pick for each model point the closest data point (hence “iterative closest point”). The correspondences can, and in general will, change in each iteration.

The difficulty in the articulated case arises, because there is no closed form solution for jointly estimating all pose parameters of the model, so as to minimise the error in Eq. 2. We propose, instead of estimating *all pose parameters* with an iterative local optimiser, to restrict the attention to a small subset of parameters, which can be solved in closed form, and iterate over different subsets: we split the articulated body into the base $\mathtt{M}_b^i$ and the outer branch $\mathtt{M}_o^i$ as defined above, and align only one of them with a *rigid* transformation, which respects the joint constraints at $\mathtt{j}_i$. Without loss of generality we assume that the part to be aligned is the outer branch. We then minimise the new error function

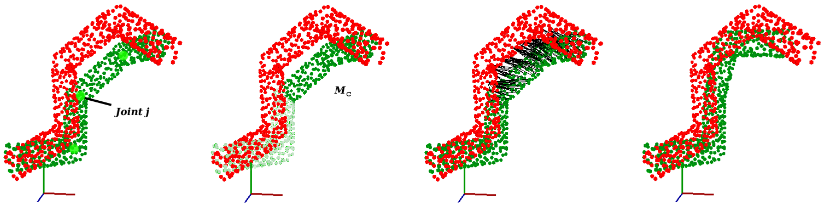
$$E_o(\theta_i) = \sum_{\mathtt{p}_k \in \mathtt{M}_o^i} \sum_{j=1}^{N_{\mathtt{p}_k}} \min_s \|\mathbf{T}_k^W \mathbf{m}_j^{\mathtt{p}_k} - \mathbf{d}_s\|^2 . \tag{3}$$

$E_o$ is the error of $\mathtt{M}_o^i$ w.r.t. the corresponding points in $\mathscr{D}$, and depends only $\theta_i$. Optimising Eq. 3 amounts to an absolute orientation problem [7, 9], which can be solved in closed form. The absolute orientation is constrained by the type of joint $\mathtt{j}_i$ – for example a *prismatic* joint allows only a 1D translation, a *spherical* joint only a 3D rotation, etc. (see App. A). The estimated transformation $\mathbf{H}_o$ defines an update of the parameters $\theta_i$ of the form $\mathbf{T}_i = (\mathbf{T}^W_{\bigwedge(i)})^{-1}\mathbf{H}_o\mathbf{T}^W_i$ (note that if the transformation is applied to the base rather than the outer branch, one also has to update $\theta_r$). The special case where $\mathtt{p}_i$ is the root corresponds to a rigid alignment of the entire model without articulation (i.e. an iteration of standard ICP).

An intuitive illustration of the algorithm is given in Fig. 2. Note that in Step 1 there are no constraints on the joint that must be chosen, neither are there constraints on which branch to select in step 2. Therefore a *selection policy* must be specified. We have experimented with three different policies: RANDOM, MULTIRANDOM and DISTRIBUTED. In the first policy, the joint and the branch are randomly selected. In the second one, a random number $W$ is chosen from an interval $[1 \ldots W_{max}]$ every time we select a joint and branch, and $W$ iterations are performed with the same joint and branch. In the third one, the joints are selected cyclically, and every time a joint is selected, the branch selection is switched. Note that all three policies include the possibility to select the root joint, thus globally aligning the whole structure without articulation with an iteration of traditional ICP. In Section 4 we will experimentally investigate how the selection policy affects the results of the algorithm.

Like standard ICP, the algorithm is guaranteed to converge to a local minimum. This is easy to show: if ICP is applied to all the model points (i.e. we “split” at the root), the reduction of the error is guaranteed [1]. Otherwise a joint $\mathtt{j}$ and a branch $\mathtt{M}_o$ are selected, while the other substructure is left untouched. The transformation applied to $\mathtt{M}_o$ is obtained by least squares minimisation [7], and hence must reduce the error in Eq. 2, unless the algorithm has converged. Which local minimum is reached depends on the initialisation and on the selection policy, but all the policies lead to a minimum.

**AICP Algorithm**



**Step 1:** Pick up a joint j in the articulated model

**Step 2:** Pick up either $\mathtt{M}_b$ or $\mathtt{M}_o$. Call the chosen branch $\mathtt{M}_c$ (solid green in figure)

**Step 3:** For each point $\mathbf{m}_i^{\mathsf{p}_s}$ such that $\mathsf{p}_s \in \mathtt{M}_c$, estimate the closest point $\mathbf{d}_v \in \mathscr{D}$.

**Step 4:** Minimise Eq. 3 and update $\theta_i'$

Figure 2: Steps of the Articulated ICP algorithm. The green chain represents the articulated model, while the red one represents the data. The black lines in step 3 show the correspondences for $\mathtt{M}_c$. The last step show the articulated model at the end of the iteration. These steps are repeated up to convergence.

# 4 Experimental Results

We evaluate our articulated ICP algorithm on both synthetic and real data. We used the synthetic data to compare our algorithm with a LM optimisation, to compare the implemented policies, and to test registration of a complex articulated structure with 54 degrees of freedom. During LM optimisation, the correspondences are updated at every single iteration of the innermost loop (i.e. every single time the linearised equation system is solved). We also evaluated our algorithm for tracking human parts with two kinds of real range data. The detailed description of the results is given in the rest of this section.

## 4.1 Visual Analysis

A visual inspection on the results of the experiments, suggests that our algorithm is more robust to local minima than the LM based one. In many of the experiments, the LM based optimisation got stuck in local minima quite soon. A typical example is reported in Fig. 3. In this example, the data and the model structure differ only for the value of one degree of freedom. Computing the correspondences in an ICP fashion leads to a *force* that pushes the displaced rigid part perpendicularly to the data. Our algorithm performs better in such situations: if a part of the model is caught in such a position, changes to other joints in the following iterations are very likely to perturb it and free it from the weak minimum. Therefore, our algorithm performs better, particularly with poor initialisation.

## 4.2 Synthetic data experiments

In the synthetic data experiments, both the model and the data are made of 3D points arranged to resemble a (regular or noisy) sampling of a parametric surface, such as a

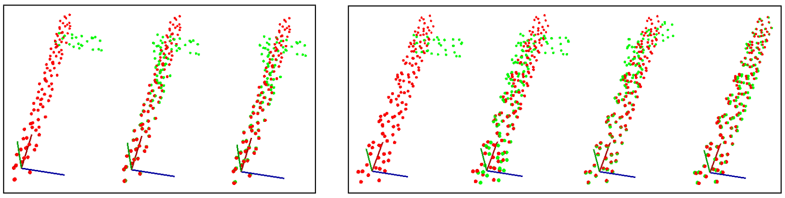


Figure 3: *Left*: the LM based algorithm gets stuck in a local minimum. *Right*: the local minimum is avoided in the proposed AICP because the perpendicular orientation of the last body is perturbed by the optimisation of some other joint.

semi-cylinder (see Fig. 1).

We used synthetic data to carry out a quantitative comparison between our algorithm and the LM-based articulated ICP and a quantitative comparison among the different policies implemented. Since with synthetic data we know the ground truth, we were able to measure the true residual error. In order to do this, both in the model and in the data, we placed marked points on the joints and on the outer extremities of the *leaf* bodies. As an error measure, we used the sum of squared distances (SSD) between the extremity points in the model and the corresponding points of the data.

For the comparison with the LM-based ICP, the synthetic structures consist of a chain of cylinders connected by spherical joints (we evaluate for both 3 and 4 cylinders). The data was generated by initialising each pose parameter with a random value sampled from $[-\frac{\pi}{2}, \frac{\pi}{2}]$. The articulated model initial pose was computed as the data initial pose plus a displacement $\mathbf{d}$. This displacement $\mathbf{d}$ takes into account both translation and rotation of the bodies in the model. The translation $\mathbf{t}$ to be applied to the whole model was sampled from three-dimensional uniform distribution with each dimension in the range $[-l/2, l/2]$, where $l$ is the major axis length of a cylinder in the chain. The displacement $\mathbf{r}$, made of the whole model rotation and of the angular displacement for each joint degree of freedom, was sampled from a multidimensional uniform distribution with each dimension in the range $[-f, f]$, for various $f$. Isotropic Gaussian noise with varying standard deviation $\sigma$ was added at each point in the data. For each $f$, 100 data configurations were created. Each of these configurations was used 3 times at each noise level.

For comparison with LM, we tested both our own implementation, and the publicly available code of [5], which uses the distance transform instead of explicit nearest-neighbor search. The results of both versions are similar. In Fig. 4 we display the results of our own implementation, which were marginally better. For the comparison, we used the DISTRIBUTED policy in our method. For each displacement value $f$, 300 SSD values were averaged. One can clearly see that our approach reduces the residual error by a factor of $\approx 2$.

With the same experimental setup, we also compared the three simple selection policies that we implemented on the chain made of three cylinders, with variable amount of noise. Fig. 4 reports some of these results. The DISTRIBUTED policy performs best in this case. The number of iterations needed to reach the convergence slightly favours the DISTRIBUTED policy as well, however further research is needed to understand better the selection policy and how it affects the algorithm in different situations.

Finally, we created the articulated *spider* data-set, with 54 degrees of freedom, in order to test the robustness of the algorithm to different initialisations on an extremely

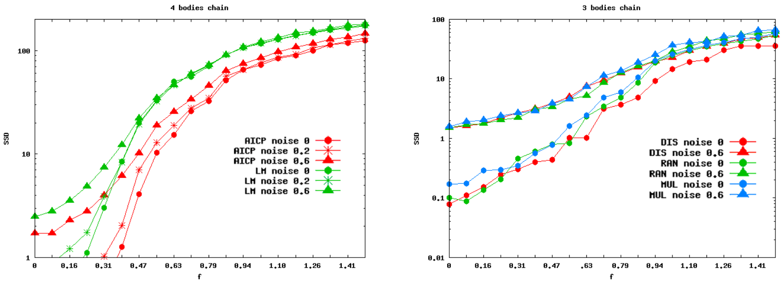


Figure 4: *Left:* residual errors of our method and the LM-based ICP for noise levels $\sigma$ (0, 0.2, 0.6) and initial displacements f. *Right:* residual errors for different selection policies. Note the logarithmic scale along the *y*-axis.

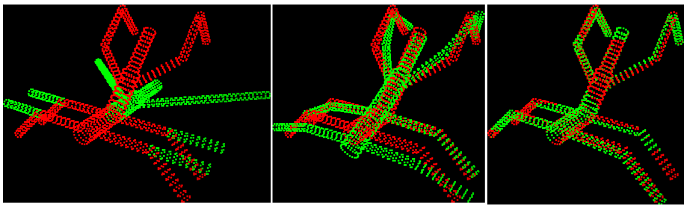


Figure 5: A registration experiment with a 54 degrees of freedom structure. The red points show the data while the green points show the model.

complicated articulated structure. Even with such a complex object, the algorithm reaches the global optimum from a relatively wide range of initialisations. An example, which shows convergence from a quite bad initialisation, is shown in Fig. 5

## 4.3 Real Data Experiments

For the real data, we set up two different tracking experiments: one human upper body tracking experiment and one hand tracking experiment. In the former experiment the data were acquired with a stereo camera Videre Design VTH-MDSC2, with 9 cm baseline. The acquisition was at 640x480 at 15 fps. A model of the upper part of the human body, comprising torso, head, arms and forearms was used. All the joints of this model are spherical joints.

In the latter experiment higher quality data were acquired with a real-time structured light system [12]. The acquisition was at 780x560 at 20 frames per second. The model of the hand is realised with spherical and hinge joints. Indeed 9 hinge joints are used to model the points of articulation of the fingers. These joints are constrained to the range of 0-90 degrees. Fig. 6 show some frames of the tracking experiments. The results are accurate both with the noisier data coming from the stereo camera and, with the more precise output of the structured light system. Note that in the hand tracking experiment, the algorithm also proves to be robust to small occlusions.

In both experiments the correspondences were computed between the points in the models and the depth map points directly from the stereo camera or the structured light

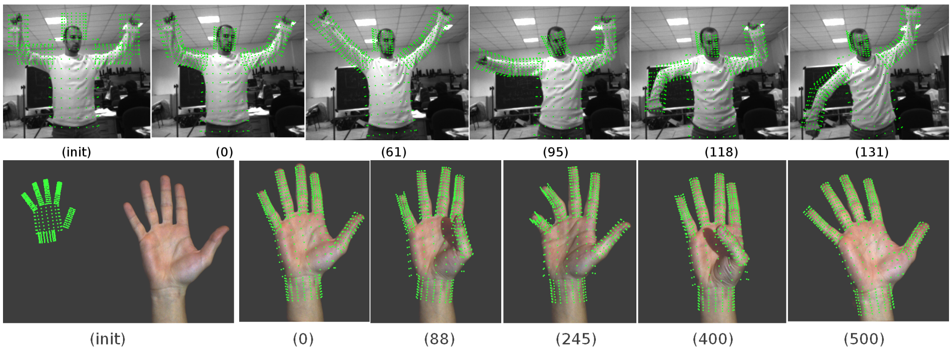


Figure 6: The green points represent the projection of the model 3D points on one of the images of the acquisition system. The first picture on the left shows the initialisation. The frame number is reported for the other images. *Top*: The human upper body tracking experiments. *Bottom*: The hand tracking experiment.

system.

We were able to process 1.5 frames per second in the human upper body experiments and about 3 frames per second in the hand tracking experiment. A considerable amount of time was used to load the data for each frame, that was bigger in the case of the stereo camera. We estimate that further code optimisation would increase the frame rate by at least a factor of 5. Moreover, the main bottleneck is nearest-neighbour search, so further speed-ups are possible through trivial parallelisation.

## 5 Discussion

In this paper we have presented a generalisation of the ICP algorithm for articulated bodies. The key ideas are to iterate the alignment over different parts of the articulated structure, and to use a rigid transformation in each iteration, which can be estimated directly, as in the classical rigid ICP. Experiments were carried out both for synthetic and real data. The results show the robustness of the algorithm and its effectiveness for registration and tracking of articulated bodies.

From a domain independent perspective, our algorithm can be seen as an implementation of the *divide et impera* strategy: at each iteration, one joint is selected and a simpler sub-problem is solved. From iteration to iteration, the joint selection changes, and thus all parameters are eventually optimised. The simpler problem can be solved in a least-squares sense in closed form, which is not the case when estimating all pose parameters of an articulated structure at once. Furthermore, each iteration on a single joint changes the pose of a large part of the structure through the kinematic chain, which allows the method to escape from local minima, in which a part of the structure is caught.

From a domain dependent perspective, our algorithm allows one to specify different selection policies (and is guaranteed to converge independent of the chosen policy). This decoupling from the core of the optimisation problem makes it possible to customise the selection policy for a specific domain. We are currently studying new selection strategies and investigating which are the elements that determine the choice of a policy in a specific domain.

Finally, we note that we have only applied our generalisation to vanilla ICP in this paper. However, it can be combined with any variant or extension of the ICP algorithm – e.g. one could register articulated objects with non-rigid parts, by plugging the ICP variant for non-rigid deformations into our framework. Future work will address different combinations, which can potentially result in very powerful registration methods. Furthermore, we plan to extend the method to closed kinematic chains. In principle one can in every iteration select two joints and break the chain. This will however require additional constraints to ensure that after the iteration the structure can be reassembled in a valid way, while still lowering the total registration error.

# A Appendix

In this Appendix we illustrate how to resolve the absolute orientation problem for two joint types: *prismatic* and *spherical*. The solutions are not proved and are not the most efficient to implement. For further details about the absolute orientation problem, see [7, 9].

In the following, we will assume that we want to align a point set $\mathscr{M} = \mathbf{m}_1 \ldots \mathbf{m}_{\mathscr{N}}$ to a point set $\mathscr{D} = \mathbf{d}_1 \ldots \mathbf{d}_{\mathscr{N}}$ in some world coordinates, when the former point set is associated to some joints and its movements are therefore constrained according to the joint type. We assume that we know the correspondences between the two point sets. Furthermore, we assume that the ordering within the point sets is such that $\mathbf{d}_j$ is the corresponding point of $\mathbf{m}_k$ only if $k = j$. We want find the the rigid transform $\mathbf{H}$ that minimise the error

$$E = \sum_{i=1}^{N} \|\mathbf{H}\mathbf{m}_i - \mathbf{d}_i\|^2 \tag{4}$$

According to the type of joint, the rigid transform $\mathbf{H}$ will be a translation or a rotation.

## A.1 Prismatic Joint

When the joint is prismatic, only translations along the vector $\mathbf{t}$ are allowed for the points in $\mathscr{M}$. The transformation $\mathbf{H}$ depends on the amount of translation $k$ along $\mathbf{t}$. To minimise Eq. 4, $k$ is such that the projections on $\mathbf{t}$ of the centres of mass of $\mathscr{M}$ and $\mathscr{D}$ are aligned. Therefore

$$k = \frac{1}{N}\mathbf{t}^\top \sum_{i=1}^{N} (\mathbf{d}_i - \mathbf{m}_i) \tag{5}$$

## A.2 Spherical Joint

When the joint is spherical, i.e. the rotation can be carried out around whatever axis passing through the joint, then a standard absolute orientation technique can be used. If the rotation is to be carried out with respect to the origin of the world reference system, then given the matrix

$$A_i = \begin{pmatrix} 0 & \mathbf{m}_i{}^\top - \mathbf{d}_i{}^\top \\ \mathbf{d}_i - \mathbf{m}_i & [\mathbf{d}_i + \mathbf{m}_i]_\times \end{pmatrix} \tag{6}$$

where $[\mathbf{v}]_\times$ is the 3x3 skew-symmetric matrix of vector $\mathbf{v}$, it is easy to show that the eigen-vector associated with the smallest eigenvalue of $\mathbf{B} = \sum_{i=1}^{N} \mathbf{A}_i^T \mathbf{A}_i$ is the quaternion that specifies the rotation to be applied to $\mathscr{M}$ in order to minimise the error in Eq. 4. Let $\mathbf{R}$ be this rotation. Note that the rotation must be carried out around the joint location $\mathbf{p}$. A translation $\mathbf{Q}$ can be applied to both $\mathscr{M}$ and $\mathscr{D}$ in order to align the $\mathbf{p}$ with the world origin. After estimating $\mathbf{R}$, the model is shifted back, resulting in an overall transformation $\mathbf{H} = \mathbf{Q}^{-1}\mathbf{R}\mathbf{Q}$.

# References

[1] P. J. Besl and N. MacKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992.

[2] Matthieu Bray, Esther Koller-Meier, Nicol N. Schraudolph, and Luc Van Gool. Fast stochastic optimization for articulated structure tracking. *Image and Vision Computing*, 2007.

[3] D. Demirdjian, T. Ko, and T. Darrell. Constraining human body tracking. *ICCV*, 2003.

[4] Guillaume Dewaele, Frédéric Devernay, Radu P. Horaud, and Florence Forbes. The alignment between 3-d data and articulated shapes with bending surfaces. In *ECCV*, 2006,

[5] A. W. Fitzgibbon. Robust Registration of 2D and 3D Point Sets. In *BMVC '01: Proceedings of the 1995 British conference on Machine vision* , pages 662–670, 2001.

[6] D. Hähnel, S. Thrun, and W. Burgard. An extension of the ICP algorithm for modeling nonrigid objects with mobile robots. *IJCAI*, 2003.

[7] B.K.P. Horn. Closed form solutions of absolute orientation using orthonormal matrices. *JOSA-A*, pages 1127–1135, 1987.

[8] Steffen Knoop, Stefan Vacek, and Ruediger Dillmann. *A Human Body Model for Articulated 3D Pose Tracking*, pages 505–520. I-Tech Education and Publishing, 2007.

[9] A. Lorusso, D. W. Eggert, and R. B. Fisher. A comparison of four algorithms for estimating 3-d rigid transformations. In *BMVC '95: Proceedings of the 1995 British conference on Machine vision (Vol. 1)*, pages 237–246, 1995.

[10] L. Mündermann, S. Corazza, and T. P. Andriacchi. The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. *Journal of NeuroEngineering and Rehabilitation*, 2006.

[11] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. *Proc. of 3rd International Conference on 3D Digital Imaging and Modeling*, 2001.

[12] T. Weise, B. Leibe and L. Van Gool. Fast 3D Scanning with Automatic Motion Compensation. *CVPR*, 2007