EUROGRAPHICS '0x / N.N. and N.N.
(Editors)

Volume 0 (200x), Number 0

Supplemental Material: FAKIR : An algorithm for revealing the anatomy and pose of statues from raw point sets

Tong Fu[1], Raphaëlle Chaine[1] and Julie Digne[1]

[1]Université de Lyon, UCBL, CNRS, France

Abstract

This supplemental material provides mathematical details for computing the distance and projection of a point onto a bone, and the optimization details for bone and joint parameters as they are performed in the FAKIR algorithm. It also provides additional FAKIR registration results and comparisons between registration obtained through our sequential optimization and a direct simultaneous optimization.

A. One-bone distance computation

In this appendix, we detail the projection of a point p on a bone B given an approximation of the oriented normal of point p. Instead of using the usual orthogonal projection on the bone, we constrain the projection $\tilde{p}$ to have a normal coherent with the one of p. This constraint is helpful when the bone lies far away from its corresponding point set: the point can then be projected on the "right side" of the bone. In the following, without loss of generality, let us assume $r_1 \leq r_2$. All the following computations depend on an angle α defined in Fig. 1 and which can be expressed as $\alpha = \arctan \frac{|r_2 - r_1|}{\sqrt{\|c_1c_2\|^2 - (r_2-r_1)^2}}$. Let us first compute $p^\star$ the projection of p on the oriented line c_1c_2, and two translations of these points along this line: $p^\star_\alpha$ at the distance $\|pp^\star\| \tan\alpha$ of $p^\star$ and $p^\star_{-\alpha}$ at the distance $-\|pp^\star\| \tan\alpha$, as illustrated on Figure 1. Let $\tau_\alpha = \frac{p^\star_\alpha - c_1}{c_2 - c_1}$, so that $p^\star_\alpha$ can be expressed as $\tau_\alpha c_1 + (1-\tau_\alpha)c_2$. Different cases can occur:

- $0 < \tau_\alpha < 1$: the point projects on the cone part of the bone. Let $\tilde{p}_\alpha$ be the intersection of segment $[p^\star_\alpha p]$ with the cone. $\tilde{p}_\alpha$ is the orthogonal projection of p on the bone. If the normal to $\tilde{p}_\alpha$ has a positive scalar product with the normal of p, $\tilde{p} = \tilde{p}_\alpha$. Otherwise, normals are deemed inconsistent and $\tilde{p} = \tilde{p}_{-\alpha}$, *i.e.* the farthest intersection of $pp^\star_{-\alpha}$ with the non-truncated cone. This situation occurs when the point p is on the wrong side of the bone (i.e. its normal is inconsistent with the normal of its closest point on the bone).
- $\tau_\alpha < 0$ (resp. $\tau_\alpha > 1$): $\tilde{p}$ is the projection of p on the sphere centered at c_1 (resp. c_2) with consistent normal direction, except if this normal-constrained projection falls within the bone and not on the envelop. In that case, p is on the wrong side of the bone, and we set $\tilde{p} = \tilde{p}_{-\alpha}$ on the other side of the non-truncated cone.

In any case, the distance between p and its normal-constrained projection $\tilde{p}$ vanishes when p is located near the surface of one bone, with a normal oriented consistently. It may happen that the returned projection does not provide a point belonging to the surface of the bone: on Figure 1, $\tilde{q}_{-\alpha}$ is the normal-constrained projection of point q, but it is not on the surface of the bone. It corresponds to a case where the point is very far from the part of the bone which is coherent with its normal. During the registration process, $\tilde{q}_{-\alpha}$ will attract q on the other side of the bone, so that the projection point will gradually be replaced by a more consistent one.

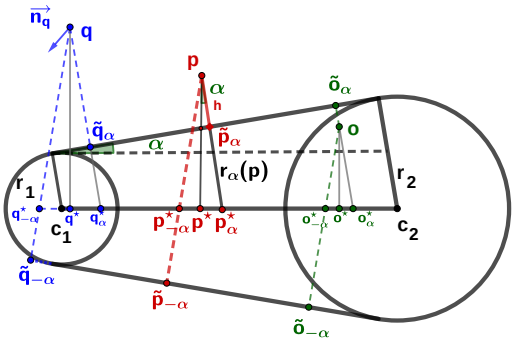


Figure 1: Various projection cases. p has two possible projections $\tilde{p}_\alpha$ and $\tilde{p}_{-\alpha}$ depending on the orientation of the normal at p. Point $p^\star$ is the projection of p on line c_1c_2. If the normal at p is oriented upward $\tilde{p} = \tilde{p}_\alpha$. Otherwise, $\tilde{p} = \tilde{p}_{-\alpha}$. The same strategy is used to project points q and o.

For completeness, let us express unsigned distance $d(p) = \|p - \tilde{p}\|$ in the various cases since they will be required in the following

Levenberg-Marquardt optimization formulations. If $\tau_\alpha < 0$ (resp. $\tau_\alpha > 1$), $d(p) = \|c_1 p\| - r_1$ (resp. $d(p) = \|c_2 p\| - r_2$). If $0 \leq \tau \leq 1$:

$$d(p) = \begin{cases} \|pp^\star_\alpha\| - r_\alpha(p) & \text{if } n_{\tilde{p}} \cdot n_p > 0 \\ \|pp^\star_{-\alpha}\| + r_{-\alpha}(p) & \text{if } n_{\tilde{p}} \cdot n_p \leq 0 \end{cases} \quad (1)$$

Since the radius of the cone varies linearly along line c_1c_2:

$$\begin{aligned} r_\alpha(p) &= \|\tilde{p_\alpha} p^\star_\alpha\| = (1-\tau_\alpha(p))r_1 + \tau_\alpha(p) r_2 \\ r_{-\alpha}(p) &= \|\tilde{p_{-\alpha}} p^\star_{-\alpha}\| = (1-\tau_{-\alpha}(p))r_1 + \tau_{-\alpha}(p) r_2 \end{aligned} \quad (2)$$

with: $\tau_\alpha(p) = \frac{c_1 p^\star_\alpha \cdot c_1 c_2}{\|c_1 c_2\|^2}$ and $\tau_{-\alpha}(p) = \frac{c_1 p^\star_{-\alpha} \cdot c_1 c_2}{\|c_1 c_2\|^2}$. Furthermore $\|pp^\star_\alpha\| = \|pp^\star_{-\alpha}\| = \|pp^\star\| / \cos\alpha$. Hence, for each bone, we first compute the α angle, then, for each point p, we compute its projection $p^\star$ on c_1c_2 and the corresponding $\tau_\alpha(p)$ yielding $r_{\pm\alpha}(p)$ and $\tilde{p}_{\pm\alpha}$.

B. Optimization for one bone

Let us assume that c_1 (Fig. 1) is fixed and let us optimize for the pose and intrinsic parameters of bone B. In a local reference frame centered at c_1 with x-axis aligned with c_1c_2, c_1 has coordinates $(0,0,0)$ and c_2 has initial coordinates $(l,0,0)$. The rotation of the bone can be parameterized by a rotation of angle θ_1 around the y-axis followed by a rotation of angle θ_2 around the z-axis. The one-bone energy is invariant by rotation around the x-axis. After the double rotation, c_2 has coordinates $(l\cos\theta_2\cos\theta_1, l\sin\theta_2, l\cos\theta_2\sin\theta_1)$. Let us call (x,y,z) the coordinates of point p in this local coordinate system and express $d(p)$ with respect to parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$,$l$ and $\boldsymbol{r} = (r_1, r_2)$. We have:

$$\tan\alpha = \frac{r_2 - r_1}{\sqrt{l^2 - (r_2 - r_1)^2}}, \cos\alpha = \frac{\sqrt{l^2 - (r_2 - r_1)^2}}{l}$$

$$\|c_1 p\|^2 = x^2 + y^2 + z^2$$

$$\|c_1 p^\star\| = x\cos\theta_2\cos\theta_1 + y\sin\theta_2 + z\cos\theta_2\sin\theta_1$$

$$\|p^\star p\|^2 = x^2 + y^2 + z^2 - (x\cos\theta_2\cos\theta_1 + y\sin\theta_2 + z\cos\theta_2\sin\theta_1)^2$$

$$\|p^\star p^\star_\alpha\| = \|p^\star p\| \tan\alpha$$

$$\|p^\star_\alpha p\| = \frac{\|p^\star p\|}{\cos\alpha}$$

$$\tau_{\pm\alpha}(p) = \frac{\|c_1 p^\star\| \pm \|p^\star p^\star_\alpha\|}{l}$$

$$\|c_2 p\|^2 = (x - l\cos\theta_2\cos\theta_1)^2 + (y - l\sin\theta_2)^2 + (z - l\cos\theta_2\sin\theta_1)^2$$

The one-bone energy function is (dropping the k subscript for simplicity):

$$E(P, B(l, \mathbf{r}), \boldsymbol{\theta}) = \sum_{p \in P} d(p)^2 \quad (3)$$

The optimization is performed on three set of parameters in turn: angles $\boldsymbol{\theta}$, bone length l and bone radii $\boldsymbol{r}$.

the optimization for bone B with respect to $\boldsymbol{\theta}$ writes:

$$\hat{\boldsymbol{\theta}} \equiv \underset{\boldsymbol{\theta}}{\operatorname{argmin}} E(P, B(l, \mathbf{r}), \boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_{p \in P} d(p, \boldsymbol{\theta})^2 \quad (4)$$

Following the Levenberg-Marquardt algorithm, at each iteration, parameter $\boldsymbol{\theta}$ is replaced by a new estimate $\boldsymbol{\theta} + \delta\boldsymbol{\theta}$, computed as:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} E(P, B(l, \mathbf{r}), \boldsymbol{\theta}) \approx \underset{\delta\boldsymbol{\theta}}{\operatorname{argmin}} E(P, B(l, \mathbf{r}), \boldsymbol{\theta} + \delta\boldsymbol{\theta}) \quad (5)$$

which is computed by taking:

$$\frac{\partial E(P, B(l, \mathbf{r}), \boldsymbol{\theta} + \delta\boldsymbol{\theta})}{\partial \delta\boldsymbol{\theta}} = \mathbf{0}$$

We finally get $\delta\boldsymbol{\theta}$:

$$\delta\boldsymbol{\theta} = -[J^T J + \lambda diag(J^T J)]^{-1} J^T \mathbf{g}(\boldsymbol{\theta})$$

where $J = [J_1, J_2]$, $J_{i1} = \frac{\partial d(p_i)}{\partial \theta_1}$ and $J_{i2} = \frac{\partial d(p_i))}{\partial \theta_2}$ and $\mathbf{g}(\theta)$ is a column vector whose entries are $d(p, \boldsymbol{\theta})$ for each point p. λ is a damping factor set to 0.01 initially and adapting it throughout iterations.

In the following, we assume $0 < \tau_\alpha(p) < 1$ and $n_{\tilde{p}} \cdot n_p > 0$. In this case, p projects on $\tilde{p}_\alpha$ and $d(p) = \|pp^\star_\alpha\| - r_\alpha(p)$ with $r_\alpha(p) = (1 - \tau_\alpha(p))r_1 + \tau_\alpha(p) r_2$, and $\tau_\alpha(p) = \frac{\|c_1 p^\star_\alpha\|}{l}$. Hence:

$$\frac{\partial d(p)}{\partial \theta_1} = \frac{1}{\cos\alpha} \frac{\partial \|p^\star p\|}{\partial \theta_1} + (r_2 - r_1) \frac{1}{l} \left(\frac{\partial \|c_1 p^\star\|}{\partial \theta_1} + \tan\alpha \frac{\partial \|p^\star p\|}{\partial \theta_1} \right) \quad (6)$$

$$\frac{\partial d(p)}{\partial \theta_2} = \frac{1}{\cos\alpha} \frac{\partial \|p^\star p\|}{\partial \theta_2} + (r_2 - r_1) \frac{1}{l} \left(\frac{\partial \|c_1 p^\star\|}{\partial \theta_2} + \tan\alpha \frac{\partial \|p^\star p\|}{\partial \theta_2} \right) \quad (7)$$

The full expression for the derivatives can be easily derived given the expressions for $\|p^\star_\alpha p\|$, $\|c_1 p^\star\|$, $\|p^\star p\|$ above. The cases $\tau_\alpha(p) < 0$, $\tau_\alpha(p) > 1$ or $n_{\tilde{p}} \cdot n_p < 0$ can be computed similarly.

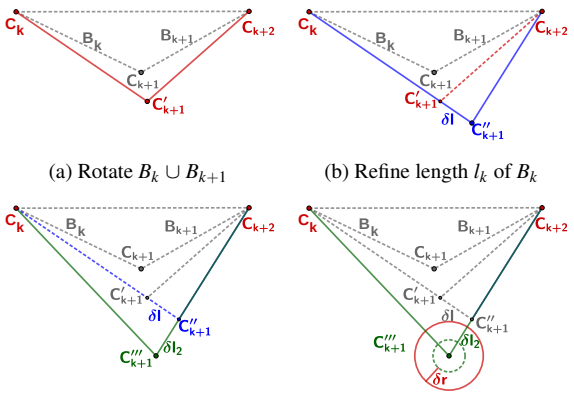
C. Optimization for a joint between two consecutive bones

Let us consider the geometric optimization of the center of the joint between two bones by optimizing the two-bones energy with respect to the lengths l_k and l_{k+1}. Each length is optimized in turn, with a side-effect on the value of the other length. The two-bones energy can be expressed as a function of l_k:

$$E_{(k,k+1)}(l_k) = \sum_{p \in P_k} \|\tilde{p}_k - p\|^2 + \sum_{p \in P_{k+1}} \|\tilde{p}_{k+1} - p\|^2 \quad (8)$$

Following the Levenberg-Marquardt algorithm, at each iteration, each parameter l_k is replaced by a new estimate $l_k + \delta l$:

$$\underset{l_k}{\operatorname{argmin}} E_{(k,k+1)}(l_k) \approx \underset{\delta l}{\operatorname{argmin}} E_{(k,k+1)}(l_k + \delta l) \quad (9)$$



(a) Rotate $B_k \cup B_{k+1}$ (b) Refine length l_k of B_k

(c) Refine length l_{k+1} of B_{k+1} (d) Refine radius r_{k+1}

Figure 2: Pairwise Optimization. With fixed extremities c_k and c_{k+2}, the pair of bones B_k and B_{k+1} is first rotated around axis $c_k c_{k+2}$ in order to minimize the two-bones energy. Then the lengths of the bones B_k and B_{k+1} and their common radius r_{k+1} are optimized successively. After these updates, the point-to-bone assignment is recomputed. As the process is repeated the distances are more accurate since the point-to-bone assignment becomes more meaningful.

By setting $\frac{\partial E_{(k,k+1)}(l_k+\delta l)}{\partial \delta l} = 0$, we get:

$$\delta l = -\frac{\sum_{p\in P_k} d_k \frac{\partial d_k}{\partial l_k} + \sum_{p\in P_{k+1}} d_{k+1}\frac{\partial d_{k+1}}{\partial l_k}}{\sum_{p\in P_k}(\frac{\partial d_k}{\partial l_k})^2 + \sum_{p\in P_{k+1}}(\frac{\partial d_{k+1}}{\partial l_k})^2} \tag{10}$$

where $d_k = \|p-\tilde{p}_k\|$ and $d_{k+1} = \|p-\tilde{p}_{k+1}\|$ are expressed as functions of l_k.

Let us detail the expression of d_k with respect to l_k: during the pairwise optimization c_k and c_{k+2} remain fixed (Figure 2). Let c_k be the origin of a local reference frame with the x-axis aligned with $c_k c_{k+1}$. In this frame, the coordinates write $c_k(0,0,0)$, $c_{k+1}(l_k,0,0)$ and $c_{k+2}(x_2,y_2,z_2)$ while a point P has coordinates (x,y,z). Then $c_{k+1}c_{k+2} = (x_2-l_k, y_2, z_2)$, $c_{k+1}p = (x-l_k, y, z)$.

Let us assume that p projects on $\tilde{p}_\alpha$ (the case $\tilde{p}_{-\alpha}$ can be deduced with minor changes). Using the same notation as in Figure 1 and appendix B, recall that $d_k = \|p-\tilde{p}_k\| = \|pp^\star_\alpha\| - r_\alpha(p)$. Since when optimizing l_k the orthogonal projection on $c_k c_{k+1}$ does not change, $\|pp^\star\|$ remains the same. However both α and $r_\alpha(p)$ change. Since $r_\alpha(p) = (1-\tau_\alpha(p))r_k + \tau_\alpha(p) r_{k+1}$ with $\tau_\alpha(p) = \frac{\|c_k p^\star_\alpha\|}{l_k}$, we get:

$$\frac{\partial d_k}{\partial l_k} = -\frac{\|pp^\star\|}{\cos^2\alpha}\frac{\partial \cos\alpha}{\partial l_k} - (r_{k+1}-r_k)\frac{\partial \tau_\alpha(p)}{\partial l_k} \tag{11}$$

Simple geometric considerations give $\cos\alpha = \sqrt{1-\frac{(r_{k+1}-r_k)^2}{l_k^2}}$, $\tau_\alpha(p) = \frac{\|c_k p^\star\| + \|pp^\star\|\tan\alpha}{l_k}$ and $\tan\alpha = \frac{r_{k+1}-r_k}{\sqrt{l_k^2-(r_{k+1}-r_k)^2}}$, whose differentiation with respect to l_k is easy.

One must also express distances d_{k+1} as functions of l_k. In that case, the projection on bone B_{k+1} is slightly different, since the position of point c_{k+1} changes with l_k. The formulas are only slightly modified by it, but this time $\|pp_\star\|$ also depends on l_k. We get:

$$\frac{\partial l_{k+1}}{\partial l_k} = \frac{1}{\cos\alpha}\frac{\partial \|pp^\star\|}{\partial l_k} - \frac{\|pp^\star\|}{\cos^2\alpha}\frac{\partial \cos\alpha}{\partial l_k} - (r_{k+2}-r_{k+1})\frac{\partial \tau_\alpha}{\partial l_k} \tag{12}$$

The full expression for the derivatives can be easily computed using the following formulas:

$$\cos\alpha = \sqrt{1-\frac{(r_{k+2}-r_{k+1})^2}{(x_2-l_k)^2+y_2^2+z_2^2}}$$

$$\tau_\alpha(p) = \sqrt{\frac{(x-l_k)^2+y^2+z^2}{(x_2-l_k)^2+y_2^2+z_2^2}}$$

$$\|c_{k+1}p^\star\| = \frac{c_{k+1}p\cdot c_{k+1}c_{k+2}}{\|c_{k+1}c_{k+2}\|} = \frac{(x-l_k)(x_2-l_k)+yy_2+zz_2}{(x_2-l_k)^2+y_2^2+z_2^2}$$

Plugging all the derivatives in Equation 10 yields δl, and l_k can be updated as $\hat{l}_k = l_k + \delta l$. This impacts the position of c_{k+1}, whose new position is computed as $\hat{c}_{k+1} = c_k + \hat{l}_k \frac{c_k c_{k+1}}{c_k c_{k+1}}$, and l_{k+1} is recomputed as : $l_{k+1} = \|\hat{c}_{k+1}c_{k+2}\|$.

The two-bones energy $E_{k,k+1}$ is then optimized with respect to l_{k+1}. This optimization is symmetric to the l_k case above and can be easily adapted. Finally, the optimization of the radius of the common joint and rotation angle around axis $c_k c_{k+2}$ are done in a similar manner.

D. Importance of the optimization order for registering a chain of bones.

During our optimization process our approach takes advantage of the articulated property of our model by processing bones in a specific order. Here, we run an experiment to illustrate that the order in which the optimizations are made is crucial.

To do so, we replace our iterations of sequential optimizations followed by point to bone reassignment by iterations of simultaneous parameter optimizations followed by point reassignment. At each iteration, the positions of all the joints are simultaneously optimized by minimizing their two-bone energies as if the adjacent joints remained fixed. The free joints at the extremities are also optimized at the same time by minimizing their one-bone energy as if their non free joint remaining fixed. However the parameter and pose change is not applied right away after each optimization but simultaneously once all updates have been computed.

If this optimization is run after our forward step (which is useful to bring each bone close to relevant data), it takes 43 iterations to converge, against 9 iterations only with our approach (See Figure 3 for an illustration of the stages). On the contrary, if the simultaneous optimization is run directly from the initial position, the method fails to converge. Figure 4 shows the result after 50 simultaneous optimizations steps.

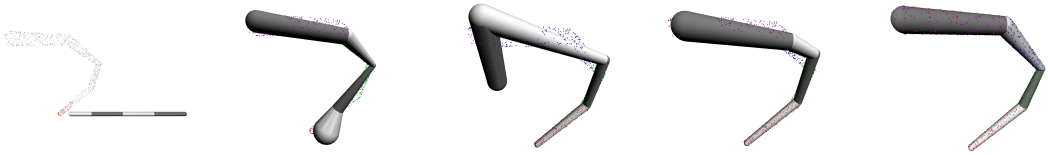


Figure 3: Simultaneous Optimization for a chain of bones applied after the first Fakir forward pass. From left to right: initial position (after our forward step), position after 10 iterations, position after 35 iterations, position after convergence (43 iterations).

(a) Initial position for a simultaneous optimization (b) Position after 50 iterations

Figure 4: Simultaneous Optimization for a chain of bones applied directly from the initial position.

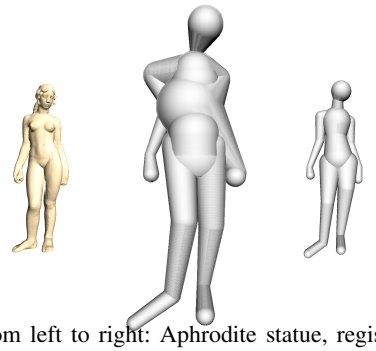


Figure 5: From left to right: Aphrodite statue, registration result without the normal-constrained projection, registration result with our normal-constrained projection

E. Importance of the normal-constrained projection

In this section we demonstrate that the normal-constrained projection both improves the result of the registration and the computation time. It is especially true in the case of the Aphrodite statue (Figure 5). Indeed, for this statue, the arms cling to the body which leads to wrong assignment of points when no normal information is used, yielding an unrealistic statue pose. The normal-constrained projection, on the contrary, permits to recover a good pose of the arms. Furthermore, it takes 16.9s and 2 iterations for the algorithm using normal-constrained projection, against 32.8s and 8 iterations – to converge to a wrong registration – otherwise (number of points: 38954). On simpler cases, like the Dancing Faun (see the main paper for a rendering of the shape), both methods manage to converge to the correct registration, but the computation time and number of iterations are still lower for the normal-constrained projection (8.7s, 6 iterations) than for the simple orthogonal projection (10.4s, 10 iterations).

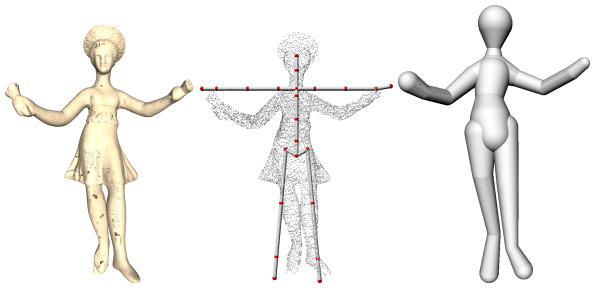


Figure 6: From left to right: the Dancer with Crotales statue, initialization position at bounding box center of point cloud, automatic registration result.

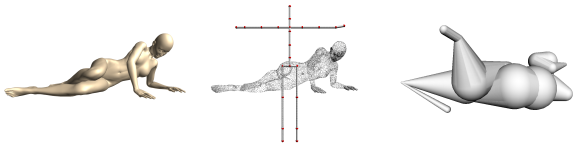


Figure 7: A failure case for automatic initialization. From left to right: the Victoria from TOSCA data set, initialization position at bounding box center of point cloud, automatic registration result.

F. Automatic initialization test

Our attempt to automatically initialize FAKIR by placing the pelvis in the center of the bounding box is effective for many statues. The Figure 6 illustrates that this works with a vertically oriented statue, while the Figure 7 shows a failure result, due to the fact that the center of the bounding box does not give any information about the orientation of the body. Thus the chains of bones are not aligned with the appropriate points, which leads to a local minimum. An improved version using principal component analysis of the points to initialize the orientation of the pelvis and adding loose constraints on the length of the bones could improve the registration.

G. Additional registration results

We show on Figures 8 and 9 four additional results, including one with a missing part (the arm) and some garments which do not hinder the registration. The Goddess Parvati and the Gorilla show the good performance of FAKIR on data which do not have realistic human proportions, while still using the standard human skeleton presented in the paper.

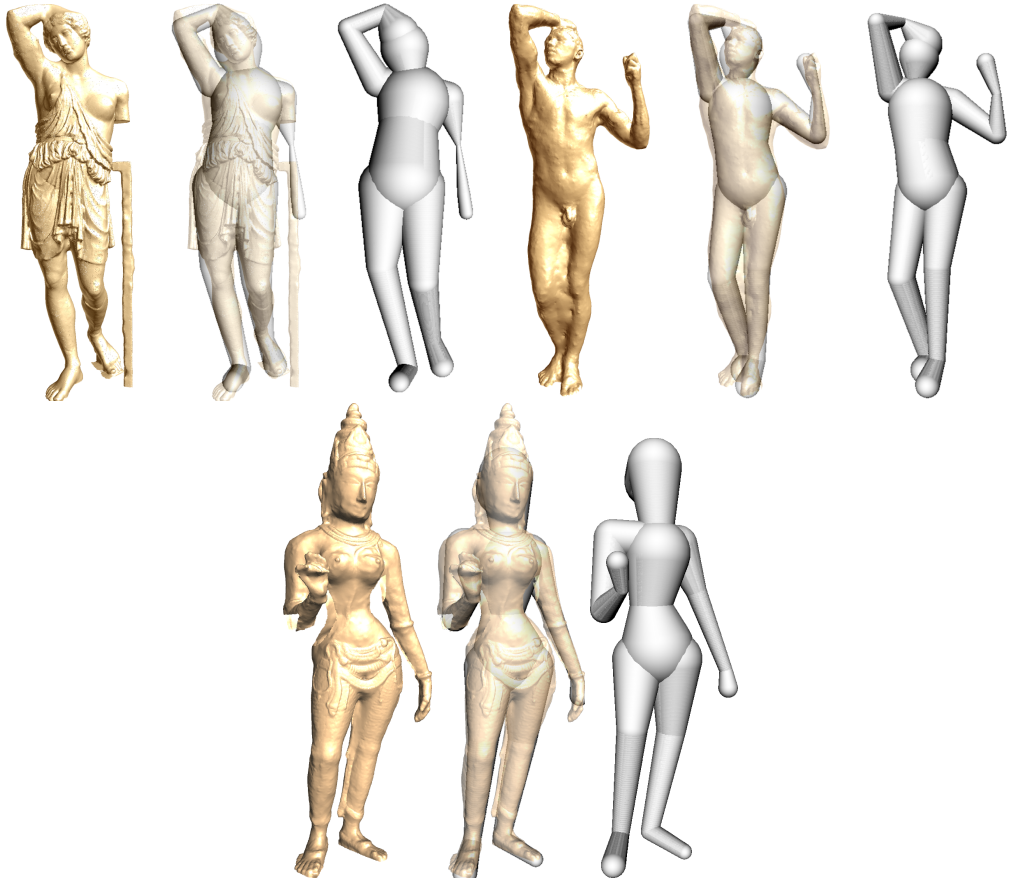


Figure 8: Additional statue registration results. Top-left: *Wounded Amazon* (Rome, 150 A.D., Nye Carlsberg Glyptotek, Copenhagen, Denmark); Top-right: *Age of Bronze* (Auguste Rodin), top-right; Bottom: *The Goddess Parvati* (South India, Circa 1200 A.C.). For each result, we show the initial point set, overlay of the data and the registered model, and the registered model alone.

Figure 9: Additional result on the Gorilla from the TOSCA dataset (initial point set; overlay of the data and the registered model; registered model alone).