Skeleton Subspace Skin Penetration Removal

Tomohiko Mukai[1] and Takafumi Taketomi[2]

[1]Tokyo Metropolitan University, Japan
[2]CyberAgent, Inc., Japan

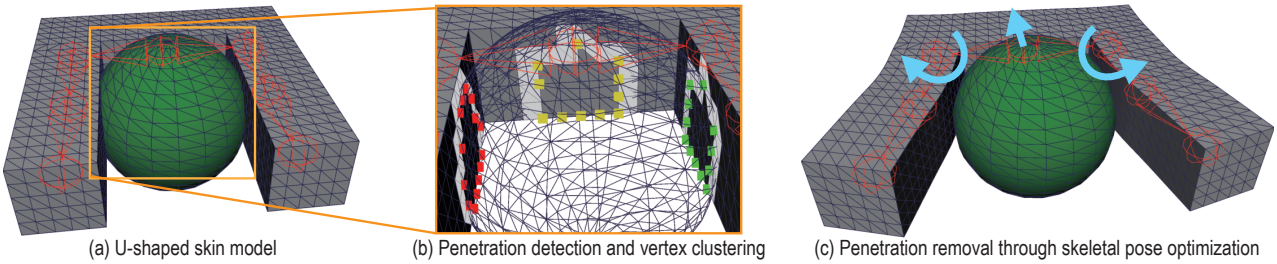

Figure 1: *The overall procedure for the proposed penetration removal algorithm. (a) Penetration between a U-shaped skinned model and a spherical obstacle. (b) Detecting intersecting polygons (white polygons), and clustering the penetrated vertices (red, yellow, and green dots) based on gradient similarity with respect to the skeletal pose. (c) The SSMIK solver optimizes the skeletal pose to offset the mean penetration displacement of each cluster.*

Abstract

This paper presents a kinematics-based method for removing penetration in skinned models based on skeleton subspace deformation. The penetration removal process is formulated as a multi-target inverse kinematics problem, in which penetrated vertices are offset by adjusting the skeletal pose. To ensure numerical stability, penetrated vertices are dynamically clustered based on deformation gradient similarity, reducing the dimensionality of the IK optimization. Experimental results demonstrate that the proposed method robustly eliminates complex penetration in articulated models while achieving interactive performance.

CCS Concepts

• ***Computing methodologies*** → ***Animation;***

1. Introduction

The skin deformation of articulated characters in interactive applications is often generated using a skeleton subspace deformation (SSD) method because of its computational stability and efficiency. One practical issue with this purely kinematic method is physically invalid deformation, particularly penetration between the skin and surrounding obstacles. The skin penetration issue should be resolved by adjusting the skeletal poses rather than by directly moving the skin vertices, as character skin often exhibits near-rigid behavior due to the presence of internal bones. Therefore, animators must adjust the skeletal poses while carefully checking the resulting skin deformation. However, this process is labor-intensive and difficult to perform reliably, especially when penetrations are obscured by other parts.

Several methods address collisions using inverse kinematics (IK) [Gas93, GOT*07, DCAB08]. These methods treat penetrated vertices as end-effectors of the skeleton and move them using an IK solver to offset the penetrations. However, conventional approaches often produce unstable results in complex contact situations. For example, when a U-shaped skin intersects a sphere as shown in Figure 1, there are three areas of penetration, where multiple neighboring polygons penetrate simultaneously. Penetration removal in such cases becomes a densely distributed multi-target IK problem that cannot be solved stably using conventional methods.

In this paper, we propose an IK-based penetration removal method for SSD-based skinned models. Our method resolves mesh penetration by adjusting the skeletal pose to deform the skin into a penetration-free configuration. This task is closely related to mesh inverse kinematics (MeshIK) [SZGP05, SZT*07], which deforms a mesh shape by directly manipulating several vertices. Since our

DOI: 10.2312/egs.20261008

method addresses a MeshIK problem within the skeleton domain, we refer to the proposed technique as *skeleton subspace mesh inverse kinematics* (SSMIK). Additionally, we propose an on-the-fly algorithm to cluster penetrated vertices according to the similarity of their displacement gradients with respect to skeletal pose. By optimizing displacements for each vertex cluster, the densely distributed multi-target IK is reduced to a sparser and lower-dimensional problem that can be computed more stably and efficiently. Our contributions are summarized as follows:

- A skeleton subspace mesh inverse kinematics (SSMIK) for removing penetration in SSD-based skinned models by indirectly adjusting skin shape through skeletal pose deformation.
- A gradient-based vertex clustering method that enables numerically stable computation for complex SSMIK problems.

2. Related work

SSMIK can be regarded as a subclass of MeshIK problems that deform a mesh by directly positioning a small subset of its vertices. Example-based approaches synthesize a new shape by interpolating the deformation gradients of the surface polygons of example shapes [SZGP05, DSP06]. These techniques require multiple example shapes and do not explicitly model internal skeletons. Mesh puppetry [SZT*07] jointly optimizes skeletal pose and mesh deformation under multiple constraints; however, its nonlinear optimization is computationally expensive when correcting dense penetration. A marker-based 3D character reconstruction method [KPK24] can also be regarded as a form of SSMIK, as it optimizes skeletal pose based on the gradients of per-vertex positional errors with respect to the skeletal pose. In contrast, our method avoids per-vertex optimization by using on-the-fly clustering to group vertices with similar deformation gradients.

Penetration-free animation is often achieved using physics-based approaches that map contact forces applied to the skin surface onto joint torques of the skeleton [Gas93]. The soft character technique [GOT*07] computes both skeleton and surface responses to collision forces using a layered representation of articulated body dynamics and skinning. The contact skinning method [DCAB08] employs a two-step process: first, contact forces are transmitted to a small number of influential joints, and then the torques of all skeletal joints are derived using inverse dynamics. However, these physics-based approaches are often unsuitable for removing deep penetration due to numerical instability. By contrast, our method adopts a purely kinematic SSMIK formulation combined with on-the-fly vertex clustering, enabling more stable penetration removal.

3. Algorithm

The proposed SSMIK method iteratively adjusts the skeletal pose of an SSD-based model to reduce skin penetration. Our basic formulation treats each penetrated vertex as an end-effector, resulting in a multi-target IK problem. The desired vertex displacements are mapped to variations in the skeletal pose using vertex position gradients with respect to the skeletal pose. As this naive formulation often leads to numerical instability, we introduce an on-the-fly vertex clustering strategy based on deformation gradients. This clustering approximates the densely distributed multi-target IK problem with a lower-dimensional and more stable formulation.

3.1. Skeleton subspace deformation

A skinned model is constructed using the SSD method. Let $\mathbf{y}_v \in \mathbb{R}^3$ be the position of the v-th vertex, $\overline{\mathbf{y}}_v \in \mathbb{R}^3$ be its rest position, $\mathcal{J} = \{1, \cdots, J\}$ be the set of skeletal joints, and $\mathcal{V} = \{1, \cdots, V\}$ be the set of vertex indices of the skinned model. The SSD model is defined as:

$$\mathbf{y}_v = \sum_{j \in \mathcal{J}} w_{v,j} \mathbf{M}_j \overline{\mathbf{M}}_j^{-1} \overline{\mathbf{y}}_v \,, \tag{1}$$

where $\mathbf{M}_j \in \mathbb{R}^{4\times4}$ and $\overline{\mathbf{M}}_j \in \mathbb{R}^{4\times4}$ represent the world transformation of the j-th joint and its rest state, respectively. The skinning weight $w_{v,j}$ specifies the influence of the j-th joint transformation on the v-th vertex position. The world transformation $\mathbf{M}_j$ is calculated using forward kinematics as $\mathbf{M}_j = \mathbf{M}_{\rho(j)}\mathbf{L}_j$, where $\mathbf{L}_j \in \mathbb{R}^{4\times4}$ and $\rho(j)$ represent a local transformation and the parent joint index of the j-th joint, respectively. The local transformation $\mathbf{L}_j$ is parameterized by nine variables of scale, rotation, and translation as $\mathbf{x}_j = \{x_{j,1}, \cdots, x_{j,9}\} \in \mathbb{R}^9$. The skeletal pose is therefore represented by the pose vector $\mathbf{X} = [\mathbf{x}_1 / \cdots / \mathbf{x}_J] \in \mathbb{R}^{9J}$ where $/$ denotes concatenation.

3.2. Penetration removal

Let $C(\mathbf{y}_v)$ be the signed distance between the vertex position $\mathbf{y}_v$ and the nearest point on the obstacle's surface, where $C(\mathbf{y}_v) > 0$ represents the penetration depth, while $C(\mathbf{y}_v) \leq 0$ corresponds to the shortest distance to a non-penetration surface. Penetration removal is formulated as a constrained least-squares optimization as:

$$\begin{aligned} &\min_{\mathbf{X}} |\mathbf{X} - \overline{\mathbf{X}}|_2 \\ &\text{subject to } \forall v \in \mathcal{V}, C(\mathbf{y}_v) \leq 0 \,. \end{aligned} \tag{2}$$

The objective function evaluates the variation between the rest skeletal pose $\overline{\mathbf{X}}$ and the deformed pose $\mathbf{X}$.

We use a standard collision detection pipeline to compute the penetration depth $C(\mathbf{y}_v) > 0$. First, intersecting polygons between the skin and the obstacle meshes are detected using a bounding volume hierarchy. A set of penetrated vertices $\mathcal{P} = \{v_1, \cdots, v_P\} \subset \mathcal{V}$ is detected in the intersected polygons by checking whether each vertex is on the back side of the plane spanned by the polygon of the penetrated obstacle. Finally, the vertex displacement $\Delta\mathbf{y}_{v_p}$ to offset the penetration depth is calculated as the shortest displacement from the skin vertex to the polygon plane. Note that we initially employed a minimum translation vector for this purpose, whereas it was unsuccessful in our preliminary experiment. A more sophisticated algorithm is required for accurate detection [KOv06].

3.3. Skeleton subspace mesh inverse kinematics

The SSMIK algorithm uses iterative optimization to adjust the pose vector $\mathbf{X}$ to eliminate penetrations. At each iteration, the penetrated vertices and their offset displacement $\{\Delta\mathbf{y}_{v_p} | v_p \in \mathcal{P}\}$ are first computed. Next, the pose variation $\Delta\mathbf{X}$ is optimized to reduce the penetration depth using a gradient descent approach. The skeletal pose is then updated by adding the variation as $\mathbf{X} \leftarrow \mathbf{X} + \Delta\mathbf{X}$. This process is repeated until no penetrated vertices remain, or the Frobenius norm $|\Delta\mathbf{X}|_F$ is less than the given threshold.

Our gradient descent method uses a first-order approximation of the mapping from the pose variation $\Delta\mathbf{X}$ to the offset displacement $\Delta\mathbf{Y}_p = [\Delta\mathbf{y}_{v_1}/\cdots/\Delta\mathbf{y}_{v_P}] \in \mathbb{R}^{3P}$ as

$$\Delta\mathbf{Y}_p = \Phi\Delta\mathbf{X}\,, \tag{3}$$

$$\Phi = \frac{d\mathbf{Y}_p}{d\mathbf{X}} = \begin{bmatrix} \Phi_1 \\ \vdots \\ \Phi_P \end{bmatrix} = \begin{bmatrix} \Phi_{1,1} & \cdots & \Phi_{1,J} \\ \vdots & \ddots & \vdots \\ \Phi_{P,1} & \cdots & \Phi_{P,J} \end{bmatrix} \in \mathbb{R}^{3P\times 9J}\,, \tag{4}$$

where Φ is the Jacobian matrix that maps small pose variations to the vertex displacements. A submatrix $\Phi_{p,j} \in \mathbb{R}^{3\times 9}$ is defined as

$$\Phi_{p,j} = \frac{\partial \mathbf{y}_{v_p}}{\partial \mathbf{x}_j} = \begin{bmatrix} \phi_{p,j,1} & \phi_{p,j,2} & \cdots & \phi_{p,j,9} \end{bmatrix}\,, \tag{5}$$

$$\phi_{p,j,d} = \sum_{k\in\mathcal{K}(j)} w_{v_p,k}\mathbf{M}_{\rho(j)}\frac{\partial \mathbf{L}_j}{\partial x_{j,d}}\tilde{\mathbf{L}}_{j\to k}\overline{\mathbf{M}}_k^{-1}\overline{\mathbf{y}}_{v_p}\,, \tag{6}$$

where $\mathcal{K}(j)$ denotes the set of the j-th joint and its descendants down to all end-effectors, and $\tilde{\mathbf{L}}_{j\to k}$ represents the product of local matrices along the skeletal chain from the j-th joint's child to the k-th joint. The optimal pose variation is approximated by solving the linear equation 3 using the weighted damped Jacobian pseudoinverse [ALCS18] as

$$\Delta\mathbf{X} = \mathbf{S}\Phi^T(\Phi\mathbf{S}\Phi^T + \epsilon\mathbf{I})^{-1}\Delta\mathbf{Y}_p\,, \tag{7}$$

where the diagonal weight matrix $\mathbf{S} \in \mathbb{R}^{9J\times 9J}$ determines joint stiffness and ϵ is a damping coefficient.

3.4. Gradient-based vertex clustering

The proposed method adopts the Jacobian pseudoinverse because it generally provides more stable solutions than the Jacobian transpose [DCAB08, ALCS18]. However, it still suffers from singularity issues: the Jacobian Φ becomes rank deficient when similar sub-Jacobians Φ_p are present. We explored the singular value decomposition and other numerically robust solvers, but these approaches failed to produce stable results and incurred a computational cost that was too high for interactive processing.

To address this numerical issue, penetrated vertices are classified into several clusters of similar sub-Jacobians using an on-the-fly clustering technique. The skeletal pose variation is then optimized using the mean offset displacement and the mean Jacobian of each cluster. We use an agglomerative hierarchical clustering approach for its efficiency. Initially, each penetrated vertex is assigned to an individual cluster, i.e., $\forall p \in \{1,\cdots,P\}, g_p = \{p\}$ and $\mathcal{G} = \{g_1,\cdots,g_P\}$. In the first step, the similarity matrix $\Lambda \in \mathbb{R}^{|\mathcal{G}|\times|\mathcal{G}|}$ is computed, where each element $\Lambda_{a,b}$ measures the similarity between the sub-Jacobians Φ_a and Φ_b of clusters a and b as:

$$\Lambda_{a,b} = \mathrm{tr}\left(\frac{\Phi_a^T\Phi_b}{|\Phi_a|_F\,|\Phi_b|_F}\right)\,, \tag{8}$$

which corresponds to the sum of cosine similarities between the vertex position gradients. In the second step, clusters whose similarity exceeds a user-defined threshold $\lambda_{\min}$ are merged into a single cluster. Specifically, our method greedily selects the pair of clusters a and b with the highest similarity and merges them if $\Lambda_{a,b} \geq \lambda_{\min}$. The corresponding sub-Jacobian and offset displacement are then updated by taking the mean within each merged cluster as follows:

$$\begin{aligned} a^*, b^* &= \underset{a,b\in\mathcal{G},\Lambda_{a,b}\geq\lambda_{\min}}{\arg\max}\ \Lambda_{a,b}\,, \quad g_c = g_{a^*}\cup g_{b^*}\,, \\ \Phi_c &= \frac{1}{|g_c|}\sum_{g\in g_c}\Phi_g\,, \qquad \Delta\mathbf{y}_c = \frac{1}{|g_c|}\sum_{g\in g_c}\Delta\mathbf{y}_g\,, \\ \mathcal{G} &\leftarrow \mathcal{G}\cup\{g_c\}\setminus\{g_{a^*}\}\setminus\{g_{b^*}\}\,. \end{aligned} \tag{9}$$

These two steps are repeated until the similarity between all remaining clusters falls below the threshold $\lambda_{\min}$. This on-the-fly clustering reduces the size of the Jacobian Φ from $3P\times 9J$ to $3|\mathcal{G}|\times 9J$, while ensuring that Φ remains full rank.

Note that this greedy strategy does not guarantee a globally optimal clustering. However, we did not observe any significant issues in our experiments. This is because the penetration configuration changes at each iteration of SSMIK, which mitigates the impact of suboptimal greedy clustering.

4. Experiments

We implemented the proposed algorithm as a C++ plug-in for Autodesk Maya 2023 (https://github.com/TomohikoMukai/ssmik/) and conducted experiments using three different models. We set the damping coefficient to $\epsilon = 1.0\times 10^{-6}$ for all experiments. The clustering threshold was set to $\lambda_{\min} = 0.8$ because a comparative study demonstrated that it should be set to a relatively large value, as demonstrated in the supplemental video. The computational performance was evaluated on a laptop equipped with an Intel Core i7-13700H processor and 64 GB of RAM, and the results are summarized in Table 1. As shown in the table, the proposed method achieved interactive performance in all experiments.

The first experiment involved the deformation of a stick-like skinned model interacting with a cylindrical obstacle, as shown in Figure 2(a). Each skeletal joint was parameterized with rotation and translation, as illustrated in Figure 2(b). When the cylinder contacted the model from the right side, the stick exhibited a shrinking deformation. In contrast, when the skeletal joints were parameterized using rotation only, the model showed bending behavior without shrinkage, as shown in Figure 2(c). The bending behavior was further controlled through the weight matrix $\mathbf{S}$, which encouraged joints near the penetrated region to undergo larger adjustments as shown in Figure 2(d).

The second experiment was conducted using an H-shaped model

Table 1: *Statistics of the experiments. The computation time of SSMIK was measured separately for clustering and pose optimization.*

	# Skin vertices	# Obstacle vertices	# Joints	Collision test [ms]	Cluster [ms]	Pose [ms]
Fig. 1	1212	382	7	3.73	0.91	0.26
Fig. 2	90	42	4	0.46	0.52	0.34
Fig. 3	1170	382	9	1.72	0.97	0.66
Fig. 4	1518	382	26	2.55	4.79	1.81

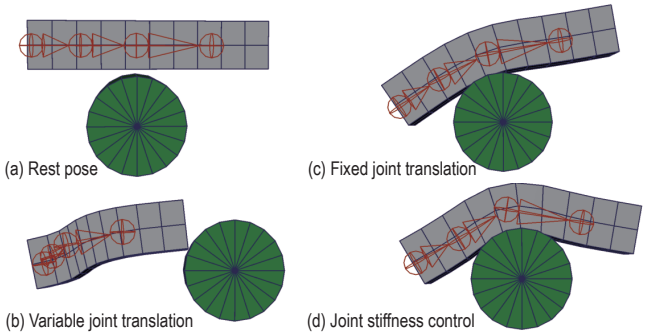

Figure 2: *(a) Stick-like skinned model and cylinder obstacle. (b) Shrinking behavior caused by the joint translation optimization. (c) Non-shrinking deformation due to rotation-only deformation. (d) Joint stiffness control to produce further bending around the penetrated area.*

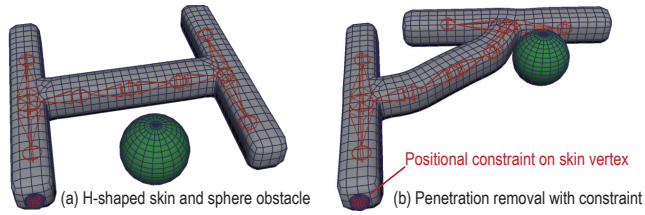

Figure 3: *(a) H-shaped skinned model, in which the position of a vertex is constrained by the red sphere. (b) Penetration removal while satisfying the positional constraint.*

as shown in Figure 3 (a). In this experiment, a positional constraint was imposed on a vertex at the model edge by directly computing an offset displacement to maintain its specified position, as indicated by the red sphere. The proposed method successfully produced a bending deformation at the model branch while satisfying the positional constraint as shown in Figure 3 (b).

The final experiment used a quadrupedal horse model. Figure 4 shows the deformed configuration in which the feet were moved under positional constraints while obstacles were applied to the head. The proposed method remained effective even for models with many vertices and joints.

5. Discussion

We have presented an SSMIK algorithm that removes skin penetration by adjusting the skeletal pose. Our approach uses a gradient-based optimization to map the offset displacement of penetrated vertex clusters onto the skeletal pose variations. One area for further improvement is the acceleration of on-the-fly clustering. The current implementation computes Jacobian matrices for all penetrated vertices and merges them into a smaller set of clusters at runtime. Since Jacobian similarity depends on both skinning weights and skeletal poses, it is impractical to precompute all possible clustering patterns. A promising direction would be to incorporate a machine learning-based similarity predictor or a partial precomputation strategy.

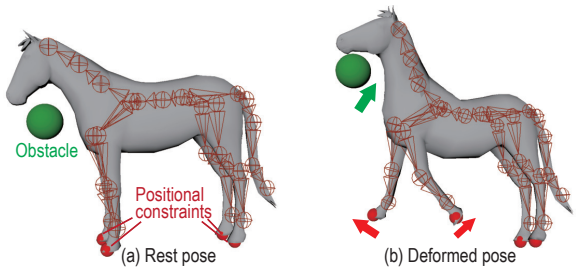

Figure 4: *(a) A skinned horse model whose foot positions are constrained. (b) The collision with the head was avoided while ensuring that the four feet were in the specified position.*

Although the proposed method performs well in many scenarios, it may produce temporally unsmooth results. To address this issue, future work should consider temporal coherence in penetration-free skin animation. A possible solution is a spacetime formulation that jointly minimizes spatial and temporal variations in the skeletal motion. Another interesting avenue for future research is the adoption of alternative IK solvers, such as FABRIK [ALCS18]. We also plan to incorporate volume preservation constraints into the optimization framework.

References

[ALCS18] ARISTIDOU A., LASENBY J., CHRYSANTHOU Y., SHAMIR A.: Inverse kinematics techniques in computer graphics: A survey. *Comput. Graph. Forum. 37*, 6 (2018), 35–58. doi:https://doi.org/10.1111/cgf.13310. 3, 4

[DCAB08] DURIEZ C., COURTECUISSE H., ALCALDE J.-P. D. L. P., BENSOUSSAN P.-J.: Contact skinning. In *Eurographics 2008 Short Papers* (2008). doi:10.2312/egs.20081025. 1, 2, 3

[DSP06] DER K. G., SUMNER R. W., POPOVIĆ J.: Inverse kinematics for reduced deformable models. *ACM Trans. Graph. 25*, 3 (2006), 1174–1179. doi:10.1145/1141911.1142011. 2

[Gas93] GASCUEL M.-P.: An implicit formulation for precise contact modeling between flexible solids. In *SIGGRAPH '93* (1993), pp. 313–320. doi:10.1145/166117.166157. 1, 2

[GOT*07] GALOPPO N., OTADUY M. A., TEKIN S., GROSS M., LIN M. C.: Soft articulated characters with fast contact handling. *Comput. Graph. Forum. 26*, 3 (2007), 243–253. doi:https://doi.org/10.1111/j.1467-8659.2007.01046.x. 1, 2

[KOv06] KAVAN L., O'SULLIVAN C., ŽÁRA J.: Efficient collision detection for spherical blend skinning. In *GRAPHITE '06* (2006), pp. 147–156. doi:10.1145/1174429.1174452. 2

[KPK24] KANG D., PARK H., KWON T.: Efficient inverse-kinematics solver for precise pose reconstruction of skinned 3D models. *Comput. and Graph. 125* (2024), 104125. doi:10.1016/j.cag.2024.104125. 2

[SZGP05] SUMNER R. W., ZWICKER M., GOTSMAN C., POPOVIĆ J.: Mesh-based inverse kinematics. *ACM Trans. Graph. 24*, 3 (2005), 488–495. doi:10.1145/1073204.1073218. 1, 2

[SZT*07] SHI X., ZHOU K., TONG Y., DESBRUN M., BAO H., GUO B.: Mesh puppetry: Cascading optimization of mesh deformation with inverse kinematics. *ACM Trans. Graph. 26*, 3 (2007), 81:1–81:10. doi:10.1145/1276377.1276479. 1, 2