Normal transformations for articulated models

Bruce Merry
University of Cape Town

Patrick Marais
University of Cape Town

James Gain*
University of Cape Town

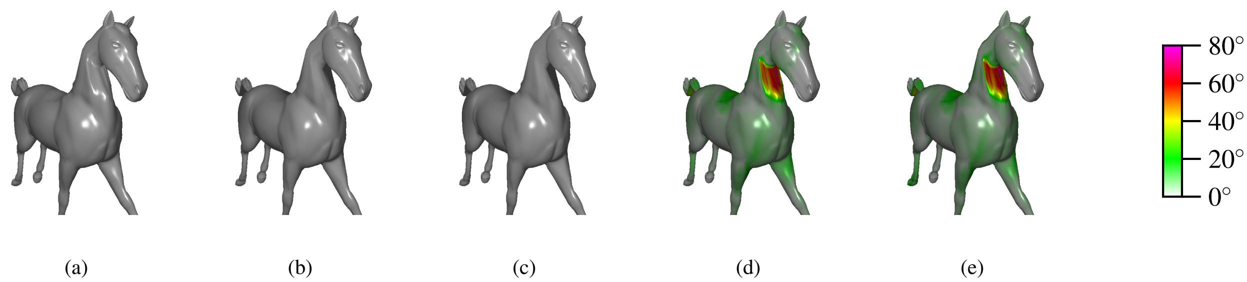

Figure 1: Normal transformation methods and their angle errors. (a) Correct transformation (equation (1)). (b), (c) Equations (2), (3) — note the lack of highlights on the neck and displaced highlights on the body. (d), (e) Errors introduced by equations (2), (3).

1 Introduction

It is well-established that when a matrix is used to transform a rigid object, the normals should be transformed by the inverse transpose of that matrix. For skeletally animated models, it is common to apply this approach to the blended matrix that animates each vertex. This is only an approximation, as it assumes that the blended matrix is locally constant. We derive a correct method for normal transformation in skeletally animated models, and examine the errors introduced by two approximations.

Skeletal subspace deformation (SSD) is a simple method for character animation, based on a linear blend of transformation matrices [Mohr and Gleicher 2003]. We use the following notation: $\mathbf{v}'$ is the rest position of some point on the surface, M_i is the 4×4 matrix that transforms bone i from its rest position to its dynamic position, w_i is the influence of bone i, and $\mathbf{v}$ is the dynamic position. In general, a prime indicates a rest pose value, while an over-bar (e.g., $\overline{N}$) indicates the top-left 3×3 sub-matrix of a matrix. SSD is described by the equation $\mathbf{v} = \left(\sum w_i M_i\right)\mathbf{v}'$.

2 Exact solution

We treat the polygonal mesh as an approximation to a differentiable manifold, with a differentiable weight field for each bone. Consider a local (s,t) parametrization of the rest surface around a point $\mathbf{v}'$:

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial s} &= \frac{\partial}{\partial s}\left(\sum M_i \mathbf{v}' w_i\right) = \left(\sum M_i \frac{\partial \mathbf{v}'}{\partial s} w_i\right) + \left(\sum M_i \mathbf{v}' \frac{\partial w_i}{\partial s}\right) \\ &= \left(\sum w_i M_i\right)\frac{\partial \mathbf{v}'}{\partial s} + \left(\sum M_i \mathbf{v}' \frac{\partial w_i}{\partial \mathbf{v}'}\frac{\partial \mathbf{v}'}{\partial s}\right) \\ &= \left(\sum w_i M_i + M_i \mathbf{v}' \frac{\partial w_i}{\partial \mathbf{v}'}\right)\frac{\partial \mathbf{v}'}{\partial s}. \end{aligned}$$

The normal is transformed by the inverse transpose of the top-left 3×3 sub-matrix of the tangent transformation[1] (further detail may be found in Merry et al. [2006]):

$$\mathbf{n} = \overline{\left(\sum w_i M_i + M_i \mathbf{v}' \frac{\partial w_i}{\partial \mathbf{v}'}\right)}^{-T} \mathbf{n}'. \tag{1}$$

*email: {bmerry,patrick,jgain}@cs.uct.ac.za

[1]Normalization will be omitted from all equations to improve clarity.

3 Approximations

For a given rest-pose normal $\mathbf{n}'$, Mohr and Gleicher [2003] mention two common approximations for the transformed normal $\mathbf{n}$:

$$\mathbf{n} = \overline{\left(\sum w_i M_i\right)}^{-T} \mathbf{n}' \tag{2}$$

$$\mathbf{n} = \left(\sum w_i \overline{M_i}^{-T}\right)\mathbf{n}'. \tag{3}$$

The first is similar to our exact solution, but does not include the second term. The second is typically used for efficiency, as the M_i matrices are independent of the vertices and hence only one inverse operation is needed per bone per frame, rather than one per vertex per frame.

4 Results

For the model shown above, the rest normal at each vertex is estimated by averaging the surrounding face normals. The weight gradients $\frac{\partial w_i}{\partial \mathbf{v}'}$ are similarly computed by averaging the gradients of the surrounding faces, followed by a projection onto the plane perpendicular to $\mathbf{n}'$. Figure 1 shows the results: both approximations introduce large and quite similar errors (over 60° in some places). Equation (3) has only slightly worse error, most noticeably on the outside of joints. In our hardware-accelerated implementation, the two approximations run at the same speed, while our exact solution is 5% slower. The exact solution is thus suitable for most cases, although an extra vector ($\frac{\partial w_i}{\partial \mathbf{v}'}$) must be passed for each influence.

References

Merry, B., Marais, P., and Gain, J. 2006. Correct normal transformations for articulated models. Tech. Rep. CS06-01-00, Department of Computer Science, University of Cape Town.

Mohr, A., and Gleicher, M. 2003. Building efficient, accurate character skins from examples. *ACM Trans. Graphics 22*, 3, 562–568.