Crossroads
The ACM Student Magazine

The Science of Shape:
Revolutionizing Graphics and Vision with the Third Dimension

by **Justin Solomon**

Almost any popular computer game that has been released recently draws complex three-dimensional figures as part of ordinary game play. From *Warcraft*'s moving figures to the complex but less mobile realism of *Riven* or *Uru*, computer games display and produce shapes on the fly depending on user input. Often, the more realistic (and more complicated) the shapes, the more believable the game. For that reason, significant research has focused on the science of shape, from its generation, to its visualization and analysis.

Beyond computer games, many other applications depend on properties of shape. We live in a three-dimensional world, and most software that uses real-world input must account for that fact. For instance, a robot searching for a tool in a toolbox may need to identify the head of a particular screwdriver, or distinguish the handle of a pair of clippers from the blade. The fact that humans can accomplish these tasks almost without thinking indicates that they are good targets for computer vision research. If humans can analyze and even reproduce shape quickly and accurately, a computer should be able to do the same.

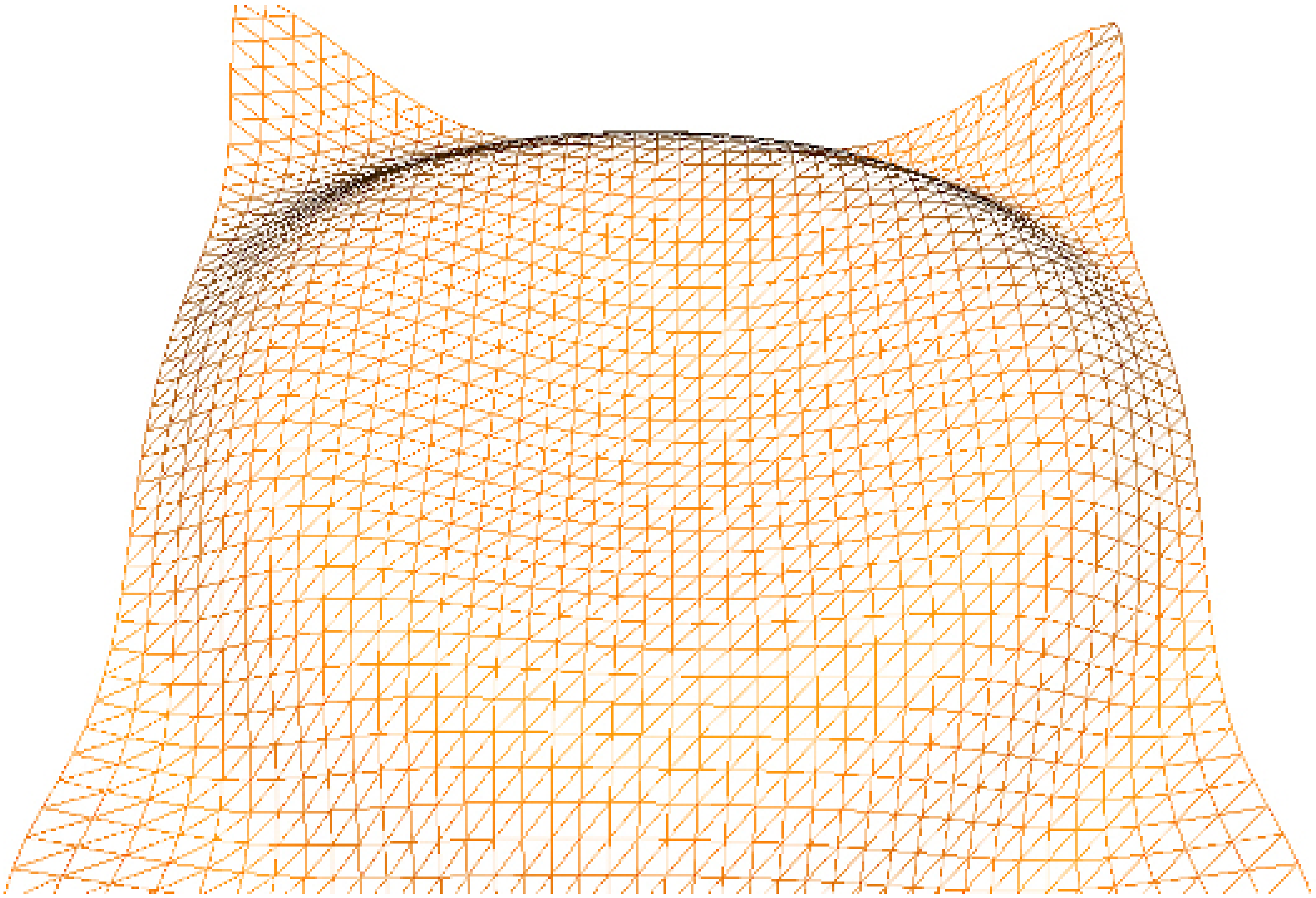

Figure 1. A polygonal mesh.

From a computational and mathematical standpoint, the efficient storage and manipulation of three-dimensional figures can be a difficult problem. The standard structure of choice for storing 3-D shape is the polygonal mesh, such as the one shown in Figure 1, which consists of several flat faces linked together to form a surface. Although each individual face is planar, and thus totally flat, taken together the faces can be used to approximate smooth surfaces, linking together smaller and smaller polygons that represent different parts of their shape. Just as the resolution of a digital photograph describes how many individual color values make up a certain area of the image, the number of faces on a mesh gives a rough idea of how detailed that mesh can be: the more faces (used in a reasonable way), the better the shape can be expressed as long as it is not totally flat.

While the concept of a mesh has been around since the inception of 3-D computer graphics, recent developments have changed mesh processing from an art to a science. Before, meshes were simply the medium – perhaps the "virtual clay"– of digital artists wishing to add the third dimension to their work. Anybody seeking to verify the merits of this type of work should view *Shrek*, *Cars*, or any other 3-D animated films which can be appreciated not solely based on plot or voicing, but also based on lifelike and fluid computer animation.

New developments, however, allow us to take mesh processing one step further. Now, even the least artistic can create high-quality digital models of their face, or dub their facial movements onto another character or person with minimal manual intervention. Regions of interest of a surface can be separated from the ordinary. We can even morph one shape into another without wasting significant time choosing "key points" to guide the process.

Although it is not possible in this article to give a complete survey of recent developments in shape analysis, in the following sections we will take a look at a few areas of research that are taking basic mesh processing to the next level. These research topics lie at the intersection of graphics, computer vision, artificial intelligence, and even discrete mathematics, allowing developers from many fields within computer science to participate.

New Shapes from Old: Morphable Models

In 1999, Volker Blanz and Thomas Vetter, from the German Max Planck Institute for Biological Cybernetics, published a research paper that would permanently change the way many researchers look at mesh geometry [2]. The paper, entitled "A Morphable Model for the Synthesis of 3D Faces," describes a method for synthesizing new faces from old. That is, with the algorithms the paper introduced, software could be created to make realistic face models by combining scans of actual humans in a particular way. These new models are almost indistinguishable from actual human models scanned using complex "structured light" equipment.

Fundamental to Blanz and Vetter's research is the concept of a **morphable model**. In general, morphable models make it possible to add and subtract surfaces, essentially allowing a system to create a basis for "shape space" out of which any shape in a particular class can be built. By limiting the number of basis vectors we are willing to use, the morphable model can become a very efficient method for representing shape.

Meshes can consist of tens or even hundreds of thousands of vertices. Adjusting the positions of each of these vertices to represent a particular face can be an unreasonable computational challenge if they are all treated separately. Using the morphable model, however, we can group changes in vertex position together for representing common changes in shape among several surfaces. For instance, Blanz and Vetter's system can make lips smile and add corresponding dimples, make cheeks chubbier, or give a face a crooked nose simply by adding the appropriate basis vector for each change.

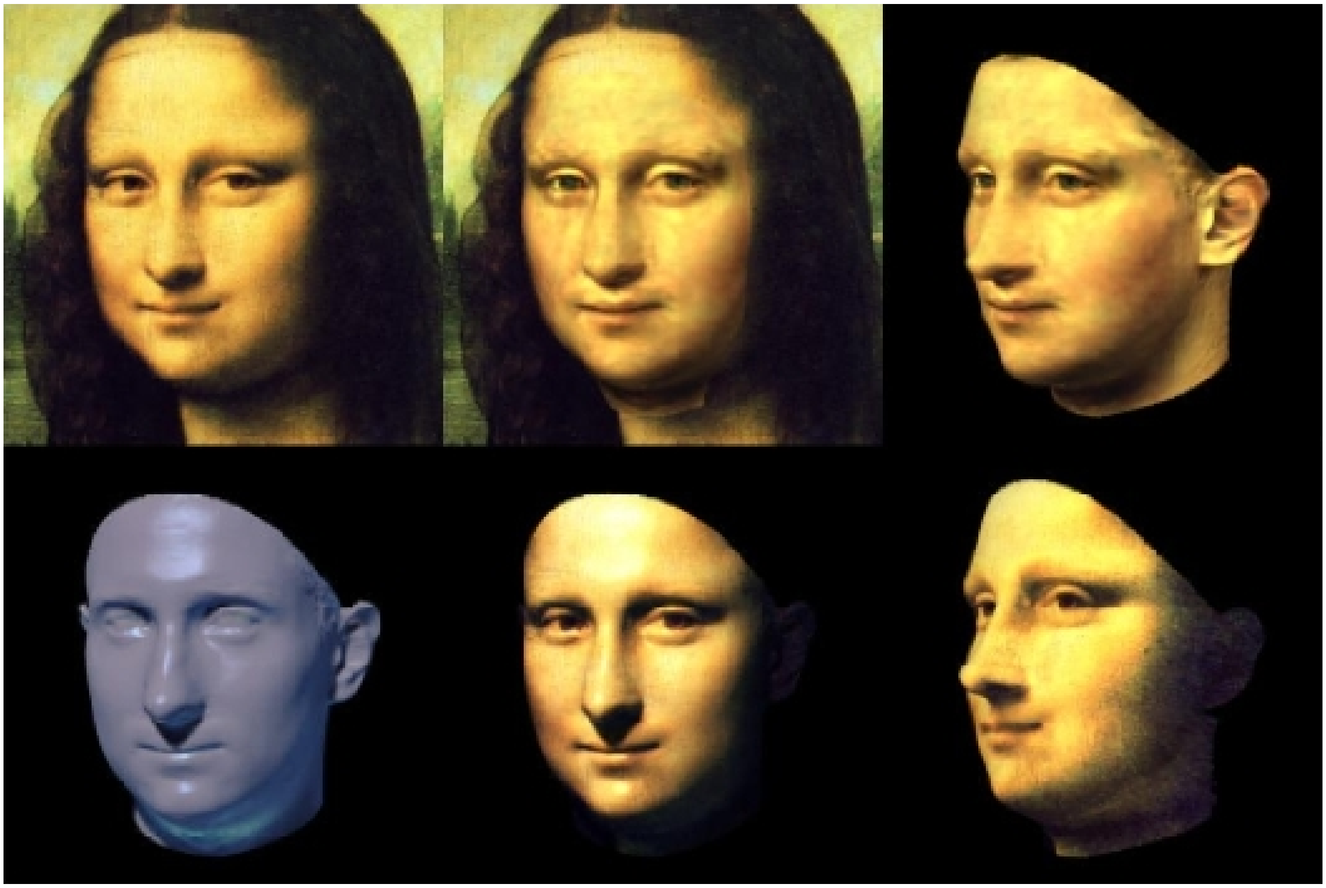

Figure 2. A 3-D reconstruction of the Mona Lisa's face.

Although the general concept of using a morphable model to represent certain statistical variations predates Blanz and Vetter's paper, the paper showed that morphable models are capable of representing shapes as complicated as a human face accurately and efficiently. Blanz and Vetter constructed their morphable model by taking images of several faces using a 3-D scanner and putting them into "one-to-one" correspondence by reexpressing each shape using the same mesh structures. After the faces were put into correspondence, they had the same number of vertices in approximately the same positions on each face; for that reason, combining face shapes became as easy as combining the position of each individual point on the faces. Blanz and Vetter then applied a basic operation from linear algebra, called principal component analysis, to find a basis for expressing shape changes between faces. The resulting basis was so accurate that they even could estimate the shape of Mona Lisa's face, as shown in Figure 2.

Since Blanz and Vetter's initial work was published in 1999, several other researchers have extended the concept of morphable face models to make even more advanced systems. For instance, researchers from MIT and the Mitsubishi Electric Research Laboratories (MERL) published a method for expressing changes in face shape using a multilinear model, accounting for shape changes not only based on a person's identity but also based on what they are saying [10]. Also, as it turns out, morphable shape models are by no means usable only for describing face shape. Researchers in France even managed to apply the concept of a morphable model to animating three-dimensional animal motion [9].

Exploring the Structure Behind a Mesh: Discrete Differential Geometry

While morphable models seek to group certain global properties of shape across several surfaces, the field of discrete differential geometry seeks to reveal properties of a single surface represented by a particular mesh. The field of classical differential geometry dates back to the eighteenth century or earlier as a tool for analyzing shapes in physics and other basic scientific disciplines. Discrete differential geometry, on the other hand, has been developed only recently for the purpose of analyzing 3-D meshes as if they were smooth surfaces.

The two principal themes of discrete differential geometry are "convergence and structure preservation" [4]. Stated more simply, the aim of discrete differential geometry is to apply standard operators from calculus, which normally depend on surfaces being smooth rather than faceted like a mesh, to discrete shapes stored by computers. As meshes become more and more refined or detailed, values obtained from these operators must *converge* upon those predicted using calculus on smooth surfaces. Additionally, some global properties of surfaces that do not depend on what happens point-to-point (in 2-D, one such quantity would be the number of loops in a closed curve—this is called the **winding number** of the curve) should be preserved exactly when they are computed for meshes.

Although the technically inclined reader may prefer a more mathematical introduction to the field of discrete differential geometry, such as that in [4] or [8], even a basic introduction to the discipline reveals its value.

Before the introduction of discrete differential geometry, little information was known about the structure of a 3-D mesh. For the most part, meshes were treated as simple descriptions of a shape, valuable only as far as the visual images that could be rendered from their structure. Differential geometry, however, reveals that even the simplest meshes can be used to estimate properties of a particular surface, such as how much it bends at a given point and the regions in which it is concave or convex.

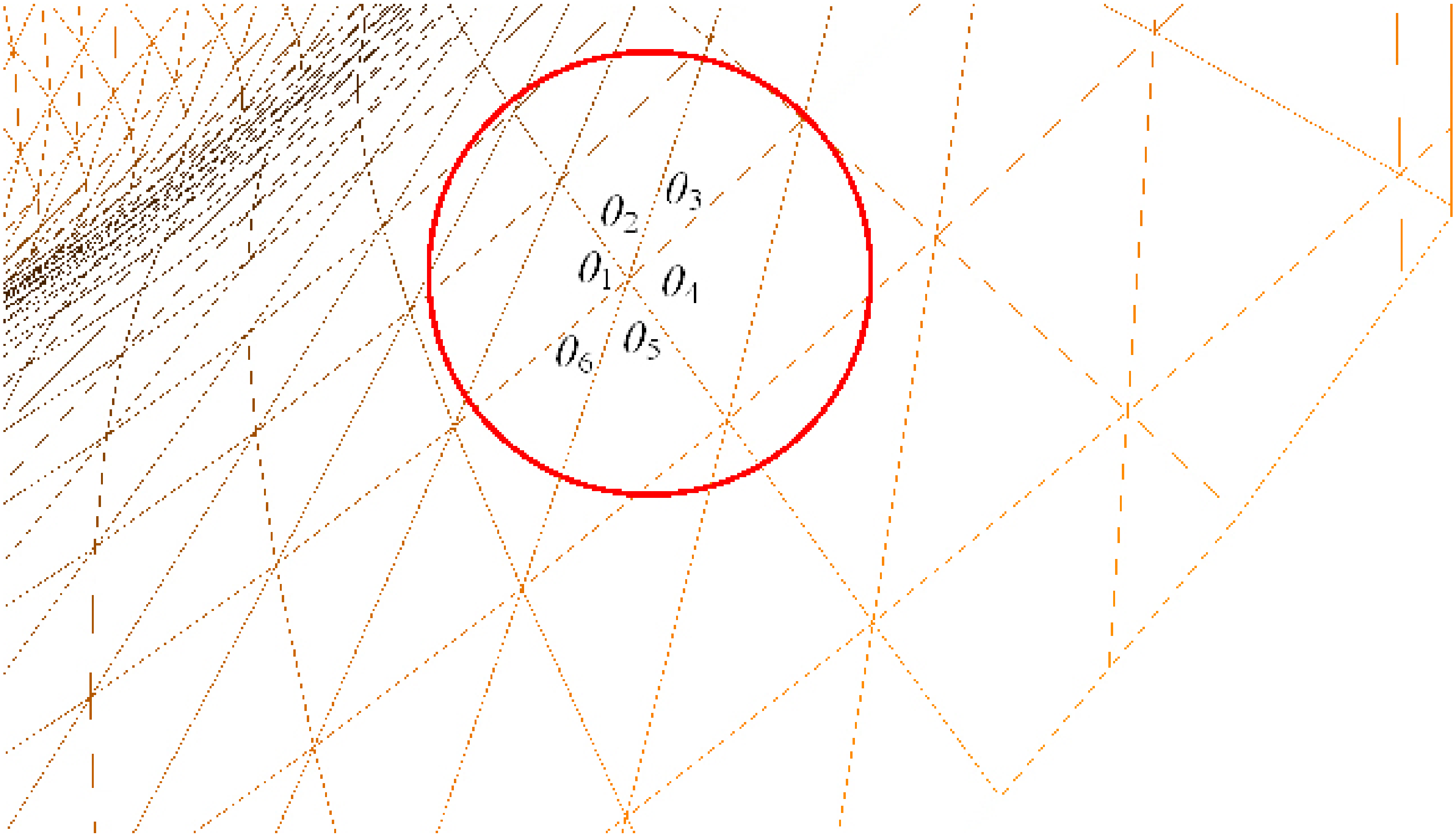

Figure 3. Interior angles around the vertex of a simple shape. Estimating discrete curvature on a figure as simple as this one simply involves taking the sum of these angles.

As a simple example, consider the problem of estimating the convexity of a surface at a particular point. If the surface bends inward, then it is concave, and if it bends outward, it is convex. It turns out that this problem can be solved by computing the **Gaussian curvature** at each point on the surface. This value describes how much a surface bends at each point, and in what direction. Positive Gaussian curvature indicates that a region is convex, and negative Gaussian curvature indicates that a region is concave. From a classical standpoint, Gaussian curvature can be difficult to compute—a surface **x**(*u*,*v*) has Gaussian curvature given by the following formula [5]:

$$K = \frac{\det(\mathbf{x}_{\mathrm{uu}}\mathbf{x}_{\mathrm{u}}\mathbf{x}_{\mathrm{v}})\det(\mathbf{x}_{\mathrm{vv}}\mathbf{x}_{\mathrm{u}}\mathbf{x}_{\mathrm{v}}) - [\det(\mathbf{x}_{\mathrm{uv}}\mathbf{x}_{\mathrm{u}}\mathbf{x}_{\mathrm{v}})]^2}{[|\mathbf{x}_{\mathrm{u}}|^2|\mathbf{x}_{\mathrm{v}}|^2 - (\mathbf{x}_{\mathrm{u}}\cdot\mathbf{x}_{\mathrm{v}})^2]^2}$$

Clearly, this formula is probably too complicated to be useful from a practical standpoint. Using discrete differential geometry, however, we can estimate Gaussian curvature on a mesh vertex very simply [4]:

$$K_p = 2\pi - \sum_i \theta_i$$

This formula reduces computing Gaussian curvature on a mesh to taking the sum of the interior angles θ_i, shown in Figure 3. This formula allows for an efficient and geometrically straightforward curvature computation method. As you can see, discrete differential geometry often can lead to very simple expressions for relatively complicated concepts.

Even though it is a relatively young field, discrete differential geometry has found applications within computer vision, computational physics, and other fields. For example, computer vision applications that collect shape data in the form of polygonal meshes also can analyze that data for interesting features, such as ridges and valleys, represented by maxima and minima in curvature. Applications that reduce the resolution of, or decimate meshes can do so intelligently, focusing on the largest number of polygon surfaces in high-curvature areas where detail is important [**2**]. Discrete differential geometry has even been used in simulating the way "thin shells," such as papers and certain cloths, interact with a physical environment [**4**].

Understanding the Shape of a Face: Three-Dimensional Face Recognition

Face recognition systems today hold the potential to revolutionize applications in security, personalization, and other fields. At this point, however, many of the most common recognition methods are hampered by a single piece of equipment—the digital camera. As you might imagine, images of the same face can look quite different from varying angles, poses, and lighting configurations. For this reason, many recognition systems designed for "active" applications where the user participates voluntarily in the digital photography process fail for "passive" situations in which the computer must be able to not only recognize a particular face but also find it without user interaction (e.g., finding a somebody in a crowd).

Analyzing the shape of a face rather than the colors it reflects into a camera can remove many of these issues. Three-dimensional shape does not depend on the angle from which you view it and is not affected by lighting. Additionally, morphable models and other geometric algorithms allow for effective representation of variation in face shape due to changes in expression or **viseme**, the shape of one's mouth and surrounding features when pronouncing a particular sound [**10**].

Despite its promise, three-dimensional face recognition remains largely theoretical. Perhaps the largest factor preventing this field from faster progress is the fact that few practical methods for inputting face shape exist. Laser and structured light interfaces used to read in the shapes of objects can be uncomfortable and inefficient for inputting human face shape. On the other hand, algorithms that infer face shape from digital video can be highly inaccurate.

Some organizations have tried more creative methods for reading in face shape. Researchers at the Mitsubishi Electric Research Labs use a dome with 16 cameras and 146 lights to read in not only face shape but also "reflectance" data, which describes how the skin reflects light [7]. The pursuit of an accurate and efficient method for face shape input suitable for use in practical situations is still an open problem within 3-D face recognition algorithm design.

Even if we can provide the computer with a description of face shape, three-dimensional face recognition systems still must identify which features of that shape uniquely identify a person. Like the problem of inputting the face shape, there is little consensus as to the best way to analyze a 3-D face. Interestingly, both morphable models and discrete differential geometry have been put to the test as methods for recognizing a 3-D face [6, 3].

Other methods for 3-D face recognition run the gamut from computational geometry, where faces are identified based on unique curves and regions that distinguish them from other models, to statistical analysis, where huge amounts of data are collected about each face as a whole in order to find certain identifying characteristics. No single method for 3-D face recognition has proved to be totally effective, leaving many research opportunities at all levels to contribute to this exciting new field.

Getting Involved

The three areas of research described above are by no means the only topics in shape analysis open to new contributions and ideas. Since this link between programming and geometry is relatively new, opportunities abound for the interested and creative programmer with or without significant mathematical background beyond calculus and geometry. To get involved, explore the papers cited below or other resources online and elsewhere to get an idea of open problems and hot research topics, and write your own code to test out your ideas. Some of the most sophisticated results in shape analysis are represented by the shortest programs, so the best way to get started is to try it yourself.

References

1

Alliez, P. et al. 2003. Anisotropic polygonal remeshing. In *Proceedings of ACM SIGGRAPH 2003*. San Diego, California (July). ACM Press, New York, NY, 485-493.

2

Blanz, V. and Vetter, T. 1999. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*. Los Angeles, California (Aug.). ACM Press, New York, NY, 187-194.

3

Blanz, V. and Vetter, T. 2003. Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25. 1063-1074.

4

Desbrun, M. et al. 2006. Discrete differential geometry: An applied introduction. *SIGGRAPH 2006 Course Notes*. Boston, Massachusetts (July/Aug.). ACM Press, New York, NY, 1063-1074.

5

Gray, A. 1997. *Differential Geometry of Curves and Surfaces with Mathematica*. CRC, Boca Raton, FL.

6

Hallinan, P. et al. 1999. *Two- and Three-Dimensional Patterns of the Face*. AK Peters, Natick, MA.

7

Lee, J. et al. 2005. A bilinear illumination model for robust face recognition. In *Proceedings of the Tenth IEEE International Conference on Computer vision*. Beijing, China (Oct.). IEEE Computer Society, Washington, DC, 1177-1184.

8

Meyer, M. et al. 2002. Discrete differential-geometry operators for triangulated 2-manifolds.

9

Reveret, L. et al. 2005. Morphable model of quadrupeds skeletons for animating 3D animals. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Los Angeles, CA (July). ACM Press, New York, NY, 135-142.

10

Vlasic, D. et al. 2005. Face transfer with multilinear models. In *Proceedings of ACM SIGGRAPH 2005*. Los Angeles, CA (July). ACM Press, New York, NY, 426-433.

Biography

Justin Solomon is a freshman at Stanford University majoring in computer science. His ongoing research interests, which he has pursued at the Naval Research Labs, MIT, the Mitsubishi Electric Research Labs, and other research organizations, include graphics and three-dimensional computer vision.