

Chapter 15

Mesh Morphing and Smoothing by Means of Radial Basis Functions (RBF): A Practical Example Using Fluent and RBF Morph

Marco Evangelos Biancolini
University of Rome, Italy

ABSTRACT

Radial Basis Functions (RBF) mesh morphing, its theoretical basis, its numerical implementation, and its use for the solution of industrial problems, mainly in Computer Aided Engineering (CAE), are introduced. RBF theory is presented showing the mathematical framework for a basic RBF fit, its MathCAD implementation, and its usage. The equations required for a 2D case comparing RBF smoothing and pseudosolid smoothing based on Finite Elements Method (FEM) structural solution are given; RBF exhibits excellent performance and produces high quality meshes even for very large deformations. The industrial application of RBF morphing to Computational Fluid Dynamics (CFD) is covered presenting the RBF Morph software, its implementation, and a description of its working principles and performance. Practical examples include: physical problems that use CFD, shape parameterisation strategy, and modelling guidelines for setting-up a well posed RBF problem. Future directions explored are: transient shape evolution, fluid structure interaction modelling, and shape parameterization in multi-physics, multi-objective design optimization.

DOI: 10.4018/978-1-61350-116-0.ch015

INTRODUCTION

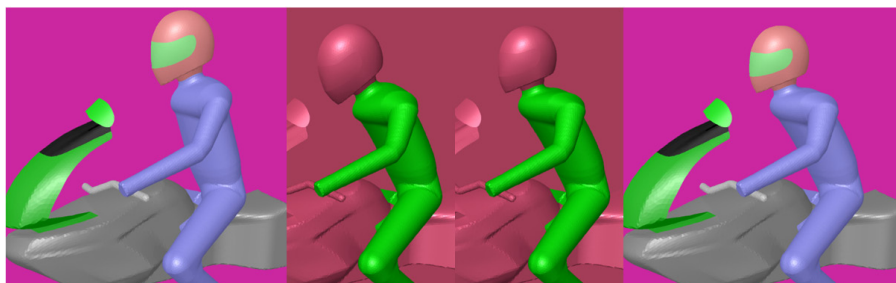
Morphing is the ability to change one thing into another smoothly. In modern computer graphics 3D character animation uses morphing for several reasons. The movement of characters is usually gained through motion capture techniques while the backbone of the character is considered to be constrained to rigid motions. However the characters are not rigid, with the obvious exclusion of characters representing mannequins or humanoid robots. Morphing transforms the surfaces of the original model into a new position or shape (see Figure 1 where the motorbike driver position is adjusted in two steps).

In computer graphics however the accuracy of the movement is not important because it just needs to look good. Morphing the mesh required for a numerical simulation is a more complicated and quite a delicate task, especially for a 3D CFD mesh. In this case morphing, also termed smoothing, is not limited to the surface but has to be extended to the entire volume of the mesh and the solver suffers dramatically. Although the concept is basically the same as morphing in computer graphics, i.e. morphing defines the motion of a set of points and moves them accordingly to the action of a motion field. The way the task is accomplished, depends on which smooth-

ing algorithm is selected and on the definition of the control points criteria which can substantially change the result. A good morpher is one that preserves the exact shape that the user wants (i.e. it undergoes a rigid motion where there is a steady object and a null rigid motion prescribed) and gently deforms the surface and volume elements that are within the deformation field and minimises the distortion of each element (i.e. distributing the motion across the entire domain).

In the literature plenty of smoothing algorithms are available. There are so many that they can be difficult to classify. However a key feature that can be used to subdivide them is how they are related to the original mesh. Some methods are defined on the internal discretised domain adopting some physical rule to propagate the movement of the boundary, as for example in the pseudosolid where an elastic FEM solution is used to propagate the deformation inside. Exotic variations of this concept exist: a solid mesh can be replaced with a network of lumped node to node springs, or an equation different to the elasticity one can be solved. The main advantage of mesh related methods is the ability to exactly prescribe surface movements when the field is known; very good quality can be achieved, especially using the pseudosolid method. However a general implementation of such a method can be

Figure 1. An example of a mesh morphing application. A motorbike driver's position is adjusted in two steps prescribing a rigid motion to the helmet and to the bike while leaving the driver's body free to deform. In the first step a rotation around the ankle is imposed to change the hunching angle and in the second step a rotation around the neck is added to correct the positions of the head so it maintains a look-ahead view.



32 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/mesh-morphing-smoothing-means-radial/60367

Related Content

Soft Computing Techniques in Civil Engineering: Time Series Prediction

Juan L. Pérez, Juan Rabuñal and Fernando Martínez Abella (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 1982-1997).

www.irma-international.org/chapter/soft-computing-techniques-civil-engineering/62557

A Need for Cyber Security Creativity

Harold Patrick and Ziska Fields (2018). *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications* (pp. 192-211).

www.irma-international.org/chapter/a-need-for-cyber-security-creativity/203505

Artificial Intelligence for Extended Software Robots, Applications, Algorithms, and Simulators

Gayathri Rajendran and Uma Vijayasundaram (2020). *AI and Big Data's Potential for Disruptive Innovation* (pp. 71-92).

www.irma-international.org/chapter/artificial-intelligence-for-extended-software-robots-applications-algorithms-and-simulators/236335

Software-Defined Networking Paradigm in Wireless Sensor Networks

Govind P. Gupta (2018). *Innovations in Software-Defined Networking and Network Functions Virtualization* (pp. 254-267).

www.irma-international.org/chapter/software-defined-networking-paradigm-in-wireless-sensor-networks/198202

Managing Tacit Knowledge to Improve Software Processes

Alberto Heredia, Javier García-Guzmán, Fuensanta Medina-Domínguez and Arturo Mora-Soto (2018). *Computer Systems and Software Engineering: Concepts, Methodologies, Tools, and Applications* (pp. 1567-1585).

www.irma-international.org/chapter/managing-tacit-knowledge-to-improve-software-processes/192936