Published in final edited form as:

Comput Methods Biomech Biomed Engin. 2009 October ; 12(5): 599–606. doi:
10.1080/10255840902802883.

A Semi-Automated Method for Hexahedral Mesh Construction of Human Vertebrae from CT-Scans

Yifei Dai and Glen L. Niebur

Tissue Mechanics Laboratory, Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN

Abstract

Generation of finite element meshes of vertebrae from CT scans is labor intensive due to their geometric complexity. As such, techniques that simplify creation of meshes of vertebrae are needed to make finite element analysis feasible for large studies and clinical applications. Techniques to obtain a geometric representation of bone contours from CT scans of vertebrae and construct a hexahedral mesh from the contours were developed. An automated edge detection technique was developed to identify surface contours of the vertebrae, followed by atlas based B-spline curve fitting to construct curves from the edge points. The method was automatic and robust to missing data, with a controllable degree of smoothing and interpolation. Parametric mapping was then used to generate nodes for each CT slice, which were connected between slices to obtain a hexahedral mesh. This method could be adapted for modeling of a variety of orthopaedic structures.

Keywords

FE modeling; computed tomography; geometric modeling; image segmentation; spine

1. Introduction

Subject-specific Finite Element (FE) models are increasingly used in biomechanical studies. In contrast to generic meshes, which employ a single representative geometry and are used to draw general biomechanical conclusions, subject-specific meshes provide a means to explore inter-subject variations in biological response to mechanical load distribution or stress-shielding (Weinans et al., 2000), or to assess biomechanical response to aging, disease, or treatments (Crawford et al., 2003, Keyak et al., 1990, Keyak and Rossi, 2000). Efficient and simple methods enhance the ability to study biomechanical problems in clinical applications and studies that involve a large number of subjects.

Obtaining accurate geometric representations of anatomic structures is critical in creating subject-specific models to maintain fidelity and ensure proper convergence characteristics of the model (Marks and Gardner, 1993). Anatomic geometry is usually derived from computed tomography (CT) data, and approaches of varying degrees of automation have been applied. Early models were created by manually tracing edges and mapping to a regular mesh (Goel et al., 1988, Goel et al., 1993). However, this is not practical for studies requiring multiple subject-specific models. Alternatively, mesh generation by direct voxel conversion (Keyak et al., 1990, Keyak and Skinner, 1992) allows fully automated direct generation from 3-D CT data, and has successfully been applied to assess fracture risk in femora (Keyak et al., 1990, Keyak

and Rossi, 2000) and vertebral bodies (Crawford et al., 2003). Modeling the posterior elements of a vertebral body – which carry approximately 20% of the axial load in the spine (Nachemson, 1960) – using voxel conversion requires that the contact regions be locally smoothed (Grosland and Brown, 2002). As such, voxel-based meshing may not be appropriate for all applications.

The most common approach to create a finite element mesh of a vertebra is extraction of the surface followed by meshing using commercial software. Border-tracing (Testi et al., 2001) has been used to extract closed contours from CT scans, which were subsequently stacked in 3-D to construct a mesh of a human femur (Viceconti et al., 2004). Three-dimensional segmentation methods such as Marching Cubes (Montani et al., 1994, Viceconti et al., 1999) can automatically extract an iso-surface of properly connected triangles that can be used to create solid models and subsequently generate tetrahedral finite element meshes (Viceconti et al., 1999). Isosurfaces extracted by Marching Cubes have been used to fit NonUniform Rational B-spline (NURBS) curves that were stacked to form a solid model of a human femur (Viceconti et al., 1998). However, both marching cubes and border tracing are sensitive to the presence of a discrete, continuous boundary, and indistinct edges due to poor image contrast could present problems. In clinical applications, these deficiencies may make the modeling procedure impractical. A hybrid method used an atlas to generate 2-D grids based on landmarks on the bone contour (Shim et al., 2007). Smooth meshes composed of hexahedral and pentahedral elements on the femur and pelvis were subsequently constructed by connecting the grids. This strategy exploits the topological similarity between subjects, and may be promising for other complex bones like vertebrae.

More robust techniques for finite element mesh construction could improve capabilities for subject-specific biomechanical analysis, and finite element modeling. The goal of this study was to develop a semi-automated technique to obtain geometric models of lumbar vertebrae from clinical CT scans. Specifically, the aims were to develop: 1) a method to extract edges and contours of a vertebra from clinical CT scans; 2) an atlas based semiautomated technique to obtain geometry representations of bone contours; and 3) a parametric mapping technique to create a hexahedron mesh on the geometry.

2. Methods

The L2 vertebra of a 53 year old female and the whole lumbar spine of an 86 year old female were obtained from anonymous donors. They were scanned using a clinical CT scanner (GE light Speed Ultra, General Electric). The spines were wrapped in saline soaked gauze during scanning. The X-ray voltage was 120 kV and current was 300 mA. The data sets had a voxel size of $0.28 \times 0.28 \times 1.25$ mm for the 53 year old and $0.31 \times 0.31 \times 1.25$ mm for the 86 year old. The CT slices were exported as grey scale TIFF images with 256 grey levels.

2.1 Edge Detection

A 2-D edge detection algorithm was developed in Matlab (Mathworks Inc., Natick, MA) to extract the bone contours on transverse slices (Fig. 1). First, noise was reduced by median filtering with a 5×5 neighborhood. The edge contour was then obtained by using a gradient sensitive edge detection method (Canny method) in Matlab. The Canny method uses two thresholds to detect edges by identifying local maxima of the gradient of the image calculated using the derivative of a Gaussian filter. Image pixels with a value greater than the high maximum were taken as edges, along with any pixels that were connected to this edge pixel and that had a value greater than the low maximum. The Canny method was selected because it is effective for images that contain step edges corrupted by white noise (Canny, 1986). The two thresholds and the standard deviation of the Gaussian filter were determined by trial and error on one slice for each vertebra. The standard deviation of the Gaussian filter was set to be 1.5 for both of the vertebrae. The values of threshold were consistent within a set of CT data.

For the L2 vertebra, the low maximum was set to be 0.118 and the high maximum was 0.12, while for the L3 vertebra the low maximum was 0.03 and high maximum was 0.12.

The Canny method detected numerous edges inside and outside the main contour. To remove these spurious edges, grayscale dilation was applied to the original image with a 4×4 square shape operator. The value of each pixel of the dilated image was then summed with the associated pixel in the image following edge detection. Only those edge points with the highest 15% of pixel values were retained.

Next, a 5×5 median filter was applied 15 times iteratively, which blurred the image. This was followed by Canny edge detection with the same parameters, grayscale dilation and thresholding of the top 15% of pixel values. The increased smoothing reduced the number of detected edges. However, the positions of the edges were slightly offset. These edges were dilated using a 4×4 square shape operator to obtain a set of coarse edges. These edges were then combined with the original edges using a bitwise AND operator resulting in a series of 2-D coordinates of edge points for each slice.

2.2 Curve Fitting

B-spline curves were fit to the edge points for each slice by least squares (Fig. 2a). The initial parametric values for the data points were obtained from an approximate B-spline curve that was defined by manually placing control points (Fig. 2b). The closet projection of each edge point to the approximate curve was found, and the associated parametric value was assigned to the edge point. Some points were initially associated with incorrect parametric values. To detect these, the points were sorted according to their parametric values, and the straight-line distance to the two neighbors was calculated. Points more than five times the in-plane resolution (about 1.6 mm) from their two parametric neighbors were assumed to be out of order or noisy data, and were excluded from the fit for one iteration. The remaining data points were fit to a B-spline curve by least squares, which was subsequently used to redefine the parametric values for the data points. The algorithm was applied iteratively until the movement of all of the control points for one iteration was less than $1/3$ of the in plane resolution, resulting in a smooth curve representation (Fig. 3). The algorithm was implemented in Matlab on a 1.4GHZ AMD Opteron workstation.

The curve fitting algorithm was automated by creating an atlas of initial approximating curves (Fig. 4). The method was then applied to the L3 vertebra from the 86 year old subject. First, five curves from the fit to the L2 vertebra and three simple geometric shapes were chosen for the atlas. The initial approximating curve for each slice was interactively chosen from the atlas. The position of the atlas curve was determined by placing the geometric centroid of the atlas curve at the associated centroid of the edge points. The atlas curves were oriented with respect to the edge points by calculating the eigen vectors of the 2-D area moment of inertia tensor of 1000 evenly sampled points, and rotated to the same orientation as the eigen vectors of the edge point moment of inertia. Finally, the curve was rescaled based on a minimax bounding box. The transformed atlas curve had approximately the same orientation and size as the edge data points. The iterative fitting algorithm was then used to obtain the final curve fit as described above.

The mean squared distance from the edge points to the final fit is often used as a measure of the error in the curve fit. However, in the case of CT data, which is pixilated and noisy, the exact surface geometry is unknown. As such, a curve fit with low error may not be desirable due to its incorporation of noise. A better measure of the error is the curvature (Kjellander, 1983). The dependence of the curvature on the number of control points was investigated for one slice of the L2 vertebra. B-spline fits using 30, 50, 70, 90, and 100 control points were

created, and the curvature and the number of sign changes in the curvature was evaluated for 1000 points that were equally spaced by arc length.

2.3 Mesh Generation

A meshing procedure was developed to obtain hexahedral meshes from the B-spline curves. First, each transverse curve of the L3 vertebra was divided into eight geometric regions: the vertebral body, pedicles, facet joints, spinal process, and transverse processes (Fig. 5c). The resulting 2-D contour curves were closed and written into an Initial Graphics Exchange Specification (IGES 5.3) file.

The IGES files were imported into PATRAN (MSC Software Corp., Santa Ana, CA) and stacked to form a 3-D surface. A quadrilateral isomesh was generated on the boundary to ensure the proper position of nodes. The cross-section defined by the bottom contour in each geometric region was then meshed using the paver feature of PATRAN to obtain a quadrilateral mesh (Fig. 6a). The nodal coordinates and element connectivity information for the cross-section and the isomesh of the bounding surface were written to a file for further processing.

Isoparametric mapping of the nodes and elements was used to generate the remaining mesh. First, mappings were created by manually selecting eight or twelve points on each curve to define a parametric space (Fig. 6b–f). Serendipity functions were used to map the location of interior nodes of the meshed cross-section into parametric coordinates. The associated nodes in the adjacent cross-section were then obtained by mapping the nodes back to Euclidean space using the serendipity functions of that curve. The process was repeated to generate the interior nodes on all slices. Element Connectivity was defined by connecting points with the same parametric coordinates between slices. By carefully selecting the initial paver mesh, the same number of nodes was obtained at the boundary between regions within each slice. The meshes from each region were then merged together to form the final mesh by associating the nodes on the boundary between regions. The quality of the resulting mesh was assessed by the minimum and maximum edge angles of the elements.

2.4 Material properties

Material properties were assigned to the elements based on density information from the CT scans. The plane equations of the six faces of the hexahedron were constructed with the normal of the plane pointing inside the element. For each hexahedral element, a minimax bounding box was defined. Each voxel in the box was tested to decide if it was inside the hexahedron. A voxel was defined as inside the element when substituting the coordinates of this voxel into all six of the plane equations resulted in non-negative values. The mean grayscale value of the hexahedral element was then calculated by averaging all the voxels inside the hexahedron. A reported calibration between elastic modulus and the Hounsfield number in the lumbar spine was used to calculate the elastic properties for the element (Rho et al., 1995):

$$E = -94 + 6.68 \times HU \quad (1)$$

3. Results and Discussion

The goal of this study was to develop a semi-automated method for the creation of hexahedral finite element meshes of human vertebrae from clinical CT scan data. With a minimum of user intervention, the technique combined novel image processing techniques and a least-squares algorithm to obtain objective representations of the surface contours. A hexahedral mesh was constructed using isoparametric mapping to locate nodes on adjacent slices (Fig. 7).

The edge detection algorithm effectively extracted the edges of the L2 and L3 vertebrae. The average CPU time for each slice was 5 seconds. This technique was fully automated when applied on the single L2 vertebra. Manual separation of adjacent vertebra was needed for the L3 vertebra, where the facet joints were intact and there was not a clear boundary on the image. Thresholding, border tracing, or marching cubes would also be affected by this situation. The edge detection technique was less sensitive to the quality of the images than gradient-based edge detection alone. For example, gradient-based edge detection could not fully eliminate spurious edges produced from cancellous bone or surrounding tissues (Fig. 1). The ability of the B-spline fitting algorithm to interpolate missing data allowed the edge detection algorithm to ignore regions where edges were blurred or difficult to detect (Fig. 3).

The B-spline contours defining the geometry of the L2 vertebra (Fig. 5a) resulted in the selection of five contours representing specific geometric features to construct an atlas (Fig. 4). Atlas based fitting was successfully applied to the L3 vertebra (Fig. 5b). With use of the atlas curves as initial approximating curves, estimation of the parametric coordinates was achieved in a maximum of six iterations for each slice. Less than ten minutes of user time was needed to segment and curve fit each slice.

Regardless of the source of data, the proposed curve fitting algorithm is robust to missing edges. Missing or blurred data can occur due to the condition of the bone or the quality of the images, especially in a clinical setting where conditions such as cartilage degeneration, fracture, or tumors can result in poor image contrast. In this method, blurred or indistinct edges were left out during edge detection and later interpolated. The B-spline fitting method automatically spanned the gaps with a smooth curve (Fig. 3). However, if the missing data is too extensive, this method would fail due to lack of sufficient data to fit a given B-spline segment. This approach automatically and objectively smoothes noisy and pixilated edges (Fig. 3). Compared to voxel conversion and border tracing that incorporate surface noise, which could propagate to errors in later finite element simulations, least-squares B-spline fitting produced a smooth geometric representation.

Increasing the number of control points improved the fidelity to the data, but resulted in increased curvature fluctuation. With the least squares fitting technique, both the curvature and the degree of fidelity to the CT data can be controlled by changing the number of B-spline control points. For the CT slice used in the smoothness investigation, a B-spline curve with 50 control points exhibited smooth and steady changes in the values and directions of curvature (Fig. 8). The number of fluctuations increased three-fold for 100 control points, while 30 control points could not adequately capture the contour geometry. As such, the number of control points was set to be around 50 for this particular CT slice. The number of control points ranged from 15 to 50 for each bone contour depending on the specific shape.

A hexahedral mesh was successfully obtained from B-spline curves of the L3 vertebra (Fig. 7). The mesh contained 4450 elements. The mesh quality analysis found that 0.4% of the elements had a face corner angle less than 10°, while 5% had angles larger than 160°. Among the large angles, only 0.8% of the elements had face corner angles greater than 165°. Most of the elements that contained large corner angles were located at the vertebral surface where the curvature had sharp changes, which reflected the poor bone condition of the vertebra that was modeled. Incorporation of wedge elements at these locations or using a finer mesh by interpolating both in plane and between slices might further improve the mesh quality.

The meshing technique was semi-automated, and required only minor user intervention. It generated a hexahedral mesh, which allows the element size to be easily modified, or refined in areas of interest. This capability may be limited when using voxel conversion. Although the

curves could be used to generate a solid model and meshed automatically, the construction of the solid model from cross-sections is complicated and usually results in a tetrahedral mesh.

The elastic modulus was higher in the posterior elements than in the vertebral body. The average element modulus for the posterior elements was 1.7 GPa, compared to 694 MPa for the vertebral body.

In a study that contains multiple subjects, generating geometrical models of vertebrae separately would require considerable effort. The methods developed here could be used to automatically construct a set of similar meshes for a large numbers of subjects. The atlas-based fitting algorithm enables fast and automatic generation of contours of a vertebra regardless of which level and subject it is from once the associated atlas has been constructed. The parametric mappings of nodal coordinates from one vertebra could similarly be used as an atlas, and applied to a large group of bones to generate finite element meshes with consistent element and node locations. Such similarity of meshes would facilitate comparisons between subjects by ensuring that element and nodal results correspond to similar anatomic locations in each subject. Using artificial intelligence techniques, this method could be further enhanced to select appropriate initial approximating curves from the atlas, partition the contour curves and automatically apply isoparametric mapping between adjacent slices, offering fully automated mesh generation. Although the method was developed for reconstruction of finite element models of lumbar vertebrae, it could be used to model a variety of orthopedic structures once a corresponding atlas of contour shape and initial quadrilateral meshes has been constructed.

Acknowledgments

This study was supported by the U.S. Centers for Disease Control (CDC CE000789) and the U.S. National Institutes of Health (NIH AR052008). Computed tomographys scans were performed by Memorial Lighthouse Medical Imaging Center, Mishawaka, IN

References

- Canny J. A computational approach to edge edtection. IEEE Trans on Patt Analysis and Machine Intell PAMI-8 1986;(6):679–98.
- Crawford RP, Rosenberg WS, Keaveny TM. Quantitative computed tomography-based finite element models of the human lumbar vertebral body: Effect of element size on stiffness, damage, and fracture strength predictions. J Biomech Eng 2003;125(4):434–8. [PubMed: 12968567]
- Goel VK, Kim YE, Lim TH, Weinstein JN. An analytical investigation of the mechanics of spinal instrumentation. Spine 1988;13(9):1003–11. [PubMed: 3061028]
- Goel VK, Lim TH, Gilbertson LG, Weinstein JN. Clinically relevant finite element models of a ligamentous lumbar motion segment. Seminars in Spine Surgery 1993;5(29–41)
- Grosland NM, Brown TD. A voxel-based formulation for contact finite element analysis. Comput Methods Biomech Biomed Engin 2002;5(1):21–32. [PubMed: 12186731]
- Keyak JH, Meagher JM, Skinner HB, Mote CD Jr. Automated three-dimensional finite element modelling of bone: A new method. J Biomed Eng 1990;12(5):389–97. [PubMed: 2214726]
- Keyak JH, Rossi SA. Prediction of femoral fracture load using finite element models: An examination of stress- and strain-based failure theories. J Biomech 2000;33(2):209–14. [PubMed: 10653034]
- Keyak JH, Skinner HB. Three-dimensional finite element modelling of bone: Effects of element size. J Biomed Eng 1992;14(6):483–9. [PubMed: 1434570]
- Kjellander JAP. Smoothing of cubic parametric splines. Comput-Aded Des 1983;15(3):175–79.
- Marks LW, Gardner TN. The use of strain energy as a convergence criterion in the finite element modelling of bone and the effect of model geometry on stress convergence. J Biomed Eng 1993;15(6):474–6. [PubMed: 8277751]
- Montani, C.; Scateni, R.; Scopigno, R. Discretized marching cubes. In: BERGERON, RD.; AEK, editors. Visualization '94' Congress. IEEE Computer society Press; 1994.

- Nachemson A. Lumbar intradiscal pressure. Experimental studies on post-mortem material. *Acta Orthop Scand Suppl* 1960;43(1–104)
- Rho JY, Hobatho MC, Ashman RB. Relations of mechanical properties to density and ct numbers in human bone. *Med Eng Phys* 1995;17(5):347–55. [PubMed: 7670694]
- Shim VB, Pitto RP, Streicher RM, Hunter PJ, Anderson IA. The use of sparse ct datasets for auto-generating accurate fe models of the femur and pelvis. *J Biomech* 2007;40(1):26–35. [PubMed: 16427645]
- Testi D, Zannoni C, Cappello A, Viceconti M. Border-tracing algorithm implementation for the femoral geometry reconstruction. *Comput Methods Programs Biomed* 2001;65(3):175–82. [PubMed: 11339979]
- Viceconti M, Davinelli M, Taddei F, Cappello A. Automatic generation of accurate subject-specific bone finite element models to be used in clinical studies. *J Biomech* 2004;37(10):1597–605. [PubMed: 15336935]
- Viceconti M, Zannoni C, Pierotti L. Tri2solid: An application of reverse engineering methods to the creation of cad models of bone segments. *Comput Methods Programs Biomed* 1998;56(3):211–20. [PubMed: 9725647]
- Viceconti M, Zannoni C, Testi D, Cappello A. A new method for the automatic mesh generation of bone segments from ct data. *J Med Eng Technol* 1999;23(2):77–81. [PubMed: 10356679]
- Weinans H, Sumner DR, Igloria R, Natarajan RN. Sensitivity of periprosthetic stress-shielding to load and the bone density-modulus relationship in subject-specific finite element models. *J Biomech* 2000;33(7):809–17. [PubMed: 10831755]

Figure 1.
A flow chart of the edge detection algorithm showing its application to a single CT slice.

Figure 2.

(a) Flowchart of the iterative B-spline fitting method. (b) B-spline hull (---) with control points (o) of the curve overlaying the edge points.

Figure 3.

(a) Unordered points in the shaded area. (b) The unordered points were temporarily excluded to obtain an improved B-spline curve approximation (c and d).

Figure 4.
B-spline vertices of atlas shapes: Five from the fit of the L2 vertebra, and three simple geometric shapes.

Figure 5.

Cross-section contours of (a) an L2 vertebra and (b) an L3 vertebra. (c) Each cross-section was partitioned into the 1. vertebral body; 2. pedicles; 3. facet joints; 4. spinal process; and 5. transverse processes.

Figure 6.

(a) A quadrilateral mesh was generated on the entire boundary defined by the stacked b-spline curves. Interior nodes of the facet joint (b) and vertebral body (c) regions were mapped into parametric space (d) using 12-node or 8-node serendipity functions. The corresponding nodal coordinates were obtained in the adjacent slices of the facet joint (e) and vertebral body (f).

Figure 7. Nodes with the same parametric coordinates were connected between slices to form the final mesh.

Figure 8.

The effect of the number of control points on smoothness. Curvature vectors were plotted in the same parametric range for (a) 50 control points; and (b) 100 control points. (c) The number of sign changes increased as more control points were used.