Title:
Texture Mapping Based On T-spline Surfaces

Authors:
Feng Xiao, Beihang University
Aizeng Wang, Beihang University, (Corresponding author) azwang@buaa.edu.cn
Chuan He, Beihang University
Ling Liu, Beijing Institute of Aerospace Systems Engineering
Gang Zhao, Beihang University

Keywords:
T-spline; Texture mapping; NURBS; CAD

DOI: 10.14733/cadconfP.2022.155-159

Introduction:

Curves and surfaces are important objects of geometric modeling, which is the core of computer graphics (CG), computer-aided design (CAD), and computer-aided engineering (CAE). With excellent properties such as local support, affine invariance, and unified mathematical forms of standard analytical bodies and free-form curves and surfaces, NURBS is widely used in graphics processing, CAD and CAE. However, NURBS will encounter serious problems when expressing complex topology. The shortcomings are mainly reflected in two aspects: (1) The distribution of control points of NURBS must form a rectangular topology, which makes it necessary to insert the control points in the whole row and column when inserting knots, resulting in data redundancy; (2) To represent complex models, the method of merging multiple NURBS surface patches is often used. However, in the actual design, the geometric continuity of the model suture is difficult to guarantee, and gaps often appear.

To overcome the limitations of NURBS in surface representation, Sederberg et al. [1] proposed the concept of T-splines in 2003. The T-spline is an extended NURBS with T-junctions. The existence of T-junctions makes control points do not have to go through the whole row or column of the control mesh, which breaks through the limitation of rectangular topology. Based on retaining the excellent properties of NURBS, T-spline also has advantages in local refinement [2], Seamless surface merging [3][4], and data compression [5]. These advantages make T-spline more and more widely used in CAD [6-8] and isogeometric analysis [9-11]. Isogeometric analysis is a new analysis method that takes the basis functions of geometric models as the shape functions and can realize the seamless combination of CAD and CAE. However, the blending functions of T-spline may be linearly dependent [12], which is never allowed in isogeometric analysis. For other research on linear independence of T-spline blending functions, see [13, 14]. To apply T-spline to isogeometric analysis directly, Scott et al. proposed the concept of analysis-suitable T-splines [14-16]. Based on this, Li Xin et al. defined analysis-suitable++ T-spline whose blending functions are always linearly independent [17, 18]. In 2020, Wang Jian et al. proposed a freeform surface reconstruction algorithm based on analysis-suitable T-spline [19], which expanded the application scope of analysis-suitable T-splines.

To improve the realism of objects, texture mapping is often used to add texture to the geometric models in CG. While texture mapping has been widely used in mesh surfaces [20], subdivision surfaces [21] and free-form surfaces [22], its application on T-spline surfaces has not been seen in the literature. To deal with the problem of lack of realism in T-spline modeling, this paper proposes a texture mapping method based on T-spline surfaces, which can improve the visual realism of T-spline models.

The organization of this paper is given as follows. Section 2 briefly introduces the basic concept of T-splines. In section 3, the texture mapping based on T-spline surfaces is discussed and several T-spline surface models are illustrated to verify the effectiveness of the proposed method. Section 4 is the conclusion.

Main Idea:

T-splines

T-spline is a kind of parametric spline surface defined on T-mesh [1][2], which is a topological mesh with T-junctions. Each vertex of T-mesh corresponds to a blending function, which is defined in the local basis function field and is composed of two B-spline basis functions whose parameter directions are orthogonal to each other, and each B-spline basis function is defined over a local knot vector determined by the T-mesh. Taking bicubic T-spline as an example (see Figure 1), the local parametric domain of the blending function corresponding to vertex P is the yellow rectangular region, and the two red edges perpendicular to each other define the knot vectors of the blending function in different parameter directions respectively.

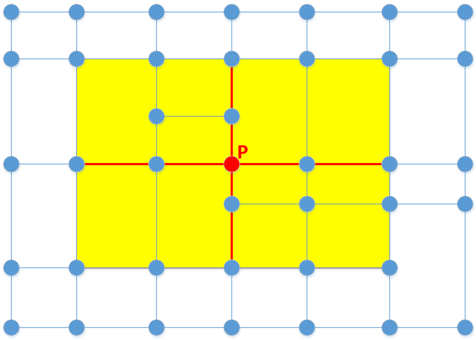


Fig. 1: The parametric domain and knot lines of the blending function corresponding to vertex P on T-mesh.

Let $p_1, p_2, \dots, p_n$ denote the control points of a T-spline surface, and their weights are respectively $w_1, w_2, \dots, w_n$. The blending functions corresponding to these control points are $B_1(u,v), B_2(u,v), \dots, B_n(u,v)$, then the T-spline surface can be expressed as [1][2]:

$$\mathrm{P}(\mathrm{u},\mathrm{v}) = \frac{\sum_{i=1}^{n} w_i p_i B_i(u,v)}{\sum_{i=1}^{n} w_i B_i(u,v)}, \tag{1}$$

where $B_i(u,v) = N_{i,k}(u)N_{i,p}(v)$, and $N_{(i,k)}(u)$, $N_{(i,p)}(v)$ are B-spline basis functions in u direction and v direction of T-mesh, with degree k and p respectively. For simplicity, the following discussion focuses on cubic degree T-splines.

Parametric domain, Texture space, and Cartesian space

The texture is generally defined on the unit square domain ($0 \le U \le 1$, $0 \le V \le 1$), which is called texture space. To paste the two-dimensional texture pattern onto a three-dimensional surface, the corresponding relationship between the points on the surface and the points in the texture space must be established. Because the T-spline surface is a parametric surface, each point on T-spline surfaces in

Cartesian space uniquely corresponds to a topological point in its parametric domain, and its global coordinates in the parametric domain are also unique. For most T-spline surfaces, its parametric domain is rectangular. By normalization, the topological points in the parametric domain have one-to-one correspondence with the points in the texture space, and the mapping relationship between the texture space and the three-dimensional surfaces is established.

As shown in Figure 2, in Cartesian space, there is a point m(x, y, z) on the surface (Figure 2(c)). Our purpose is to determine the corresponding texture point P(U, V) in texture space (Figure 2(a)). Since the topological point corresponding to m in the parametric domain is M(u, v) (Figure 2(b)), and denote the transverse and longitudinal length of the parametric domain by LengthU and LengthV, it is easy to get that the coordinates of M and P in their respective spaces are the same after normalization, thus there is $U = \frac{u}{LengthU}$ and $V = \frac{v}{LengthV}$, and the texture coordinate corresponding to the point m(x, y, z) in three-dimensional space is P($\frac{u}{LengthU}$, $\frac{v}{LengthV}$). In this way, the mapping relationship between the points on the surface and the points in texture space can be correctly established.

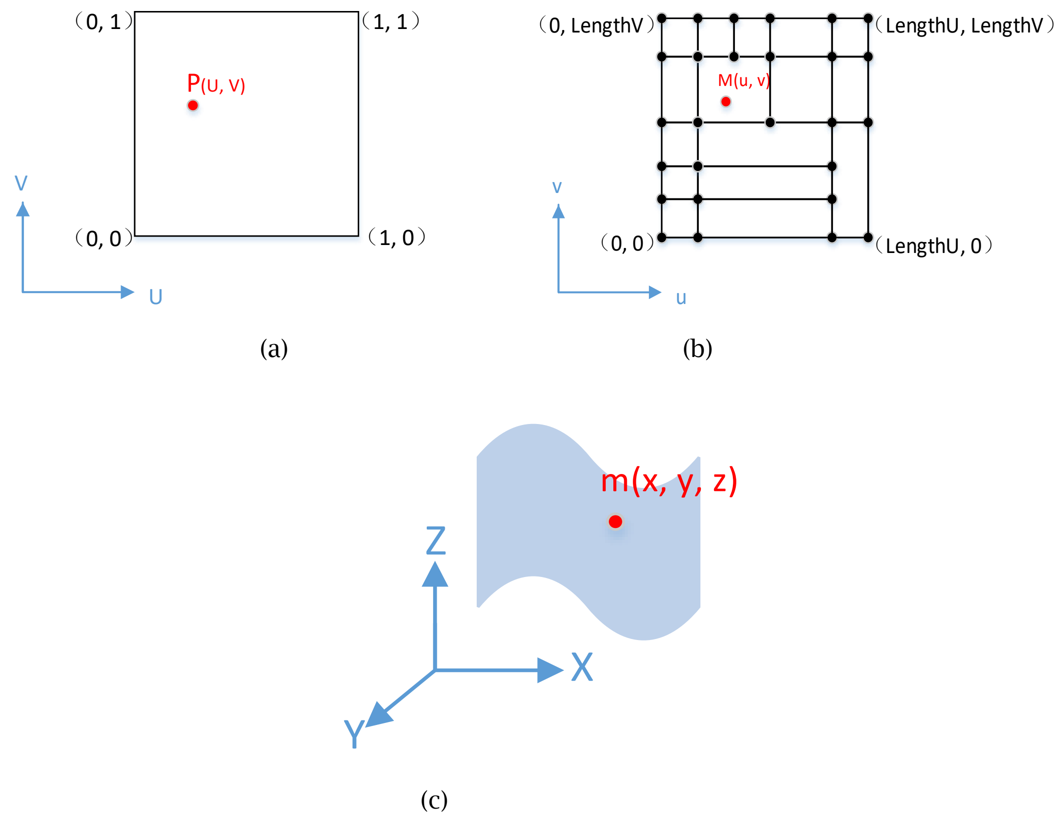


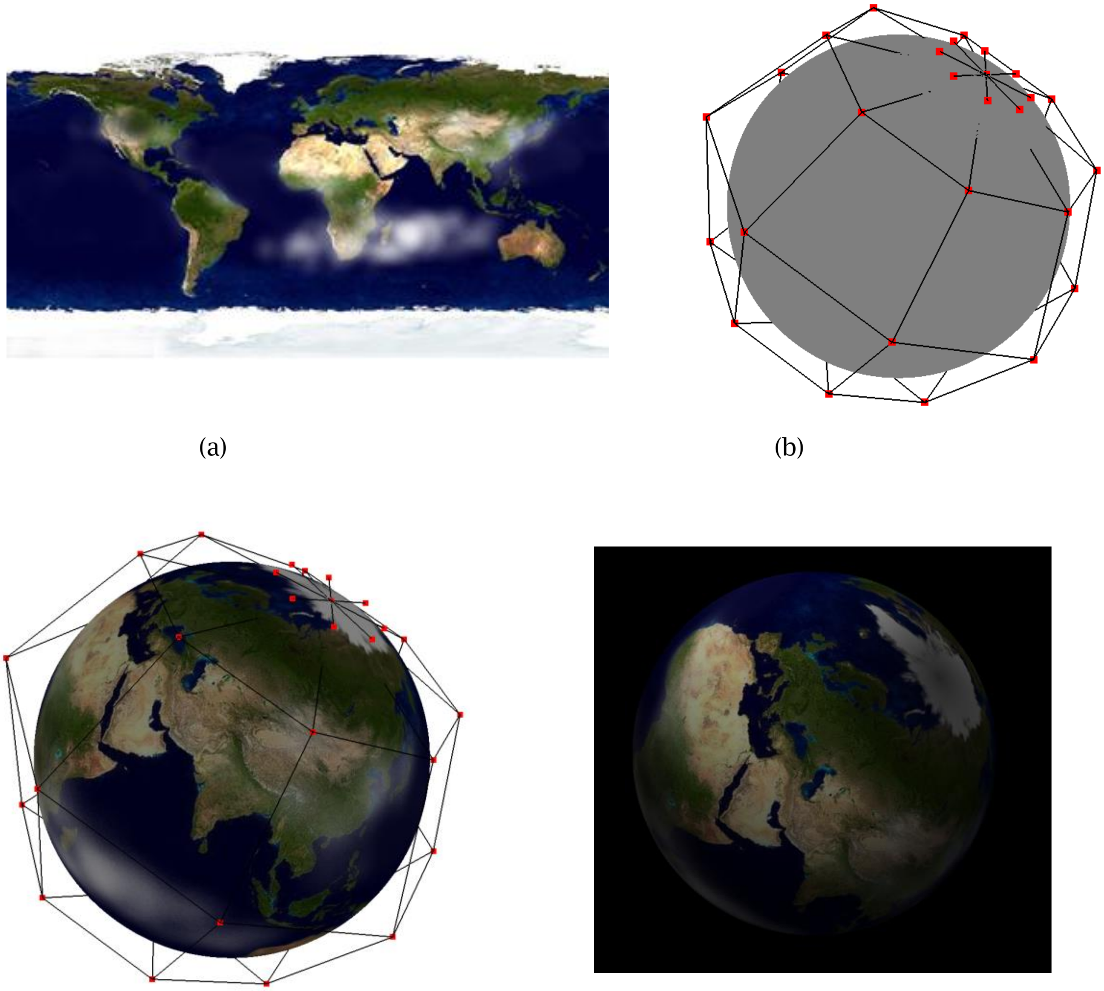
Fig. 2: Correspondence of each space: (a) Texture space, (b) T-spline parameter space, and (c) Cartesian space.

Examples

In CG, once the texture value and normal vector of the object surface are obtained, the strong realism can be easily constructed. There are two purposes to map texture to T-spline surfaces, and we will verify it through the following examples:

(1) Texture mapping can significantly improve the realism of a T-spline model and enhance the graphic display effect (see Figure 3).

Proceedings of CAD'22, Beijing, China, July 11-13, 2022, 155-159



(a) (b)

(c) (d)

Fig. 3: Texture mapping improves the realism of T-spline surfaces: (a) Texture picture, (b) T-spline model without texture, and red points are control points, (c) Add texture, and (d) Add lighting.

Conclusion:

In this paper, a method of texture mapping based on T-spline surfaces is proposed, and an approach of calculating texture value and normal vector at visible points of the T-spline surface is given. Finally, several examples demonstrate the effectiveness of our method. Texture mapping can enrich the surface details of T-spline surfaces, greatly enhance the visual realism of T-spline models, and help to expand the application of T-splines in 3D modeling.

References:

[1] Sederberg, T. W.; Zheng, J.; Bakenov, A.; et al.: T-splines and T-NURCCs, ACM SIGGRAPH 2003 Papers. San Diego, California, ACM, 477-484. https://doi.org/10.1145/882262.882295

[2] Sederberg, T. W.; Cardon, D. L.; Finnigan, G. T.; et al.: T-spline simplification and local refinement, ACM SIGGRAPH. ACM, 2004, 276-283. https://doi.org/10.1145/1015706.1015715
[3] Sederberg, T. W.; Finnigan, G. T.; Li, X.; et al.: Watertight trimmed NURBS, ACM Transactions on Graphics, 27(3), 2008, 1-8. https://doi.org/10.1145/1360612.1360678
[4] Wang, J.; Lu, Y.; Ye, L.; et al.: Efficient analysis-suitable T-spline fitting for freeform surface reconstruction and intelligent sampling, Precision Engineering, 2020, 66, 417-428. https://doi.org/10.1016/j.precisioneng.2020.08.008
[5] Xiao, W.; Liu, Y.; Li, R.; et al.: Reconsideration of T-spline data models and their exchanges using STEP, Computer-Aided Design, 79(C), 2016, 36-47. https://doi.org/10.1016/j.cad.2016.06.004
[6] Harmel, M.; Sauer, R.; Bommes, D.: Volumetric mesh generation from T-spline surface representations, Computer-Aided Design, 2017, 82, 13-28. https://doi.org/10.1016/j.cad.2016.07.017
[7] Kang, H.; Chen, F.; Deng, J.: Modified T-splines, Computer Aided Geometric Design, 30(9), 2013, 827-843. https://doi.org/10.1016/j.cagd.2013.09.001
[8] Ginnis, A.I.; Kostas, K.V.; Kaklis, P.: Construction of smooth branching surfaces using T-splines, Computer-Aided Design, 92, 2017, 22-32. https://doi.org/10.1016/j.cad.2017.06.001
[9] Hughes, T. J. R.; Cottrell, J. A.; & Bazilevs, Y.: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering, 194, 2005, 4135-4195. https://doi.org/10.1016/j.cma.2004.10.008
[10] Cottrell, J. A.; Hughes, T. J. R.; Bazilevs, Y.: Isogeometric Analysis: Toward Integration of CAD and FEA, Wiley, Chichester, 2009. https://doi.org/10.1002/9780470749081
[11] Bazilevs, Y.; Calo, V. M.; Cottrell J A, et al.: Isogeometric analysis using T-splines, Computer Methods in Applied Mechanics and Engineering, 199(5-8), 2010, 229-263. https://doi.org/10.1016/j.cma.2009.02.036
[12] Buffa, A.; Cho, D.; Sangalli, G.: Linear independence of the T-spline blending functions associated with some particular T-meshes, Computer Methods in Applied Mechanics and Engineering, 199(23-24), 2010, 1437-1445. https://doi.org/10.1016/j.cma.2009.12.004
[13] Wang, A.; Zhao, G.: The Analysis of T-spline Blending Functions Linear Independence, Computer Aided Design and Applications, 8(5), 2011, 735-745. https://doi.org/10.3722/cadaps.2011.735-745
[14] Li, X.; Zheng, J.; Sederberg, T.W.; Hughes, T.; Scott, M.: On linear independence of T-spline blending functions, Computer Aided Geometric Design, 29(1), 2012, 63-76. https://doi.org/10.1016/j.cagd.2011.08.005
[15] Scott, M. A.: T-splines as a design-through-analysis technology, 2011.
[16] Scott, M. A.; Li, X.; Sederberg, T. W.; et al.: Local refinement of analysis-suitable T-splines, Computer Methods in Applied Mechanics and Engineering, 213, 2012, 206-222. https://doi.org/10.1016/j.cma.2011.11.022
[17] Li, X.; Zhang, J.: AS++ T-splines: Linear independence and approximation, Computer Methods in Applied Mechanics and Engineering, 333, 2018, 462-474. https://doi.org/10.1016/j.cma.2018.01.041
[18] Zhang, J.; Li, X.: Local refinement for analysis-suitable++ T-splines, Computer Methods in Applied Mechanics and Engineering, 342, 2018, 32-45. https://doi.org/10.1016/j.cma.2018.07.024
[19] Wang, J.; Lu, Y.; Ye, L.; et al.: Efficient analysis-suitable T-spline fitting for freeform surface reconstruction and intelligent sampling, Precision Engineering, 66, 2020, 417-428. https://doi.org/10.1016/j.precisioneng.2020.08.008
[20] Anqing.: Curve Surface Texture Synthesis by Local Texture Mapping, Journal of Computer-Aided Design & Computer Graphics, 2008.
[21] Weng, Y.; Li, D.; Tong, Y.: Texture mapping subdivision surfaces with hard constraints, The Visual Computer, 29(11), 2013, 1231-1241. https://doi.org/10.1007/s00371-013-0794-4
[22] Arad, N.; Elber, G.: Isometric Texture Mapping for Free-form Surfaces, Computer Graphics Forum, 16(5), 2010, 247-256. https://doi.org/10.1111/1467-8659.00192
[23] Scott, M. A.; Simpson, R. N.; Evans, J. A.; et al.: Isogeometric boundary element analysis using unstructured T-splines, Computer Methods in Applied Mechanics and Engineering, 254, 2013, 197-221. https://doi.org/10.1016/j.cma.2012.11.001

Proceedings of CAD'22, Beijing, China, July 11-13, 2022, 155-159