Rapid simulation of knitted fabrics based on a ribbon yarn model

Song Mingming[1] ⓘ, **Peng Jiajia**[2] ⓘ, **Chang Chenyu**[1], **Liu Feng**[1] and **Lu Zhiwen**[1] ⓘ

Abstract

In the study, an innovative three-dimensional ribbon-based yarn model simulation technique is introduced, aimed at significantly enhancing the realism and speed of knitted fabric simulation through efficient model construction and rendering processes. The method successfully overcomes the limitations of traditional two-dimensional loop models and tubular yarn models, accurately capturing the textural details of knitted fabrics with a vivid three-dimensional representation. Experimental results demonstrate the outstanding performance of this simulation system in simulating the fuzz effects of yarn and in processing speed. This technology offers significant potential in textile design and production, notably in boosting design efficiency and accelerating new product development. By employing low-cost image capture methods, the study effectively reduces technical and financial barriers in knitted fabric simulation, enhancing its practical and industrial applicability.

Keywords

Knitting, knitted fabric simulation, three-dimensional ribbon structure, yarn model, fuzz effect, texture mapping

Compared to woven fabrics, knitted fabrics exhibit a more flexible and relaxed yarn state. This characteristic endows knitted fabrics with unique elasticity and breathability, simultaneously posing higher demands on their simulation techniques. A highly accurate yarn model can capture the key characteristics of knitted fabrics, ensuring the authenticity of simulation results.[1] However, this high level of accuracy often comes with increased computational costs, potentially slowing down the simulation process.[2] Therefore, developing a yarn model that not only precisely simulates the key characteristics of knitted fabrics but also meets the needs for rapid response and iteration in industrial production will enhance the efficiency of design and production and significantly strengthen the market competitiveness of knitted products.[3]

In the simulation of knitted fabrics, employing a two-dimensional (2D) loop model for texture mapping is an efficient method, particularly suited for handling common yarns. The relatively simple structure of regular yarns allows for the rapid simulation of the appearance of knitted fabrics with lower computational resources. By setting up the binding point model of the yarn's central curve on the 2D model,[4] and performing texture mapping, interpolation and brightness processing on the loops fitted with cubic Bézier curves, a clear and realistic knit texture can be effectively simulated.[5] This approach not only improves the rendering efficiency of the simulation under general conditions but also maintains the visual authenticity of the knitted fabric, suitable for quickly and accurately simulating the appearance of regular yarn knitted fabrics.

By creating three-dimensional (3D) models that more closely resemble the actual geometric shape of the yarn,[6] and conducting precise texture mapping in 3D space, a more in-depth simulation of knitted fabrics with complex structures can be achieved.[7] To simulate the surface details of the yarn, the cylindrical structure of the yarn is first flattened to form an elliptical cross-section and projected onto a plane.[8] Then, the yarn image is adjusted to ensure that its size and shape are adapted to the elliptical cross-section.[9] Furthermore, after pre-processing the yarn image and undergoing elastic and geometric transformations, the 3D shape

[1]College of Textile Engineering, Taiyuan University of Technology, China
[2]School of Textile Garment and Design, Changshu Institute of Technology, China

Corresponding author:
Lu Zhiwen, College of Textile Engineering, Taiyuan University of Technology, Jinzhong, China.
Email: luzhiwen@tyut.edu.cn

and lighting effects of the yarn are accurately reproduced on the 2D image using an intensity curve model and Boolean matrix. This accurately reflects the three-dimensionality and lighting changes of the yarn visually.[10] The intertwining and bending of the yarn in the fabric lead to a shortened projection length and this is addressed by using adjustment coefficients for length modification, as well as setting offsets and offset rates to simulate the random bending of the yarn, ultimately presenting a 3D form that more closely matches the actual yarn.[11]

To enhance the realism of knitted fabric simulations, a tubular model method is employed to meticulously simulate the microstructure of yarns. By matching tubular models to different types of fibers, the authenticity of yarn details is significantly improved.[12] The use of tubular models to capture the microstructural details of yarns enhances the visual effect and allows designers to directly adjust the fluffiness of the yarn and the irregularity of loop positions. This results in rendered knitted garments that exhibit a complexity and delicate texture close to reality.[13] The tubular model's limitations in yarn texture mapping can lead to an overly simplified visual appearance. While effective for basic knitted fabric simulations, it exhibits constraints when higher levels of complexity and detail are required. Therefore, to achieve a superior level of realism and detail representation, the adoption of more advanced models may be necessary.

The method of procedural model fitting for yarns utilizes procedural generation technology to accurately simulate the physical and visual characteristics of yarns, especially in simulating complex lighting and shadow effects. This model allows users to adjust various parameters to influence the appearance of the yarn, enhancing the realism of fiber and yarn rendering.[14–16] Based on the Bidirectional Scattering Distribution Function used for textile fibers, the use of a precomputed light transport method improves rendering efficiency, making the generated images more accurately reflect the physical properties of the fibers.[17] Accurate simulation of yarns at the fiber level requires highly detailed yarn data, which typically relies on advanced technologies such as computed tomography (CT) scanning. However, these methods are often costly and operationally complex. In practical applications, the need for high-precision yarn data and the cost of acquiring it constitute significant limitations. Moreover, achieving the visual effects of specific types of yarn may require extensive trial and error, further increasing the computational resource demands of this approach.

In this study, we introduce a revolutionary method for simulating knitted fabrics, the cornerstone of which is the use of 3D ribbon structures for yarn modeling. This advanced 3D model circumvents the complex calculations associated with yarn overlap inherent in traditional 2D loop models, thereby enhancing compatibility and efficiency in texture mapping. Notably, our approach intricately replicates the complex interactions between yarn and light, accurately rendering the surface texture and materiality of the textiles, thus providing a high degree of visual realism. Furthermore, this study employs low-cost image acquisition techniques, offering an efficient and practical solution for knitted fabric design and production.

Fabric model

In the study, the construction of the fabric model is based on the yarn model. The core purpose of the fabric model is to accurately reproduce the texture and structural characteristics of the fabric in 3D space.

Yarn model

In the research, the choice was made to construct the 3D geometric model of the yarn using elongated ribbon structures, based on their high compatibility with yarn image texture mapping. This approach greatly accelerates data processing. With the help of the 3D ribbon geometric model, it is possible to realistically display the length, width and depth of the yarn in space, accurately reproducing its physical characteristics. This provides an efficient and cost-effective solution in the field of textile simulation and analysis.

In the study of 3D yarn models, one limitation of the tubular structure is its constraint on texture mapping. It typically requires the use of continuous textures in the quadrilateral form to ensure that the yarn model seamlessly connects at the joint after wrapping around and maintains continuity along the length of the yarn. As shown in Figure 1, the ribbon yarn model, while maintaining a 3D spatial structure, significantly reduces computational load compared to conventional tubular structure models. This accelerates the model construction and rendering processes. More importantly, the model optimizes the texture mapping process, enhancing both efficiency and practicality. Compared to highly realistic procedural models, the method can efficiently render yarn effects with good realism even in resource-limited environments, effectively enhancing the overall quality and practicality of the simulation.

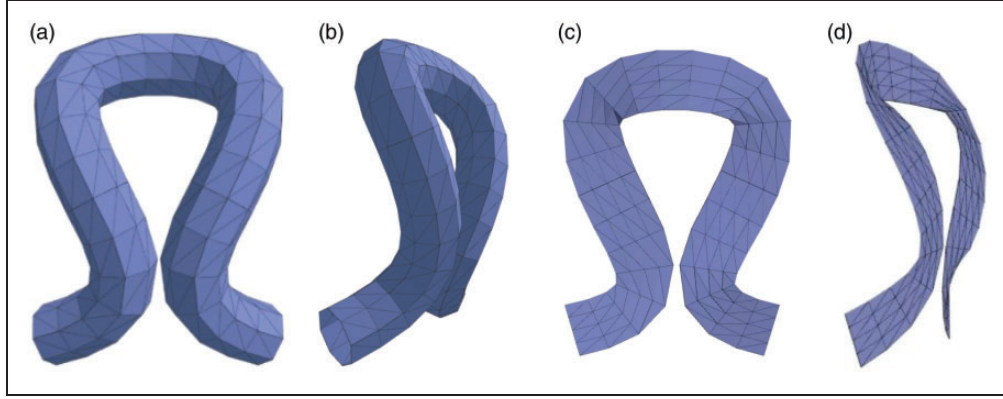The yarn model is based on Non-Uniform Rational B-Splines (NURBS) curves in 3D space as a path

Figure 1. Model comparison: (a) tubular model front view; (b) tubular model left view; (c) ribbon model main view and (d) ribbon model left view.
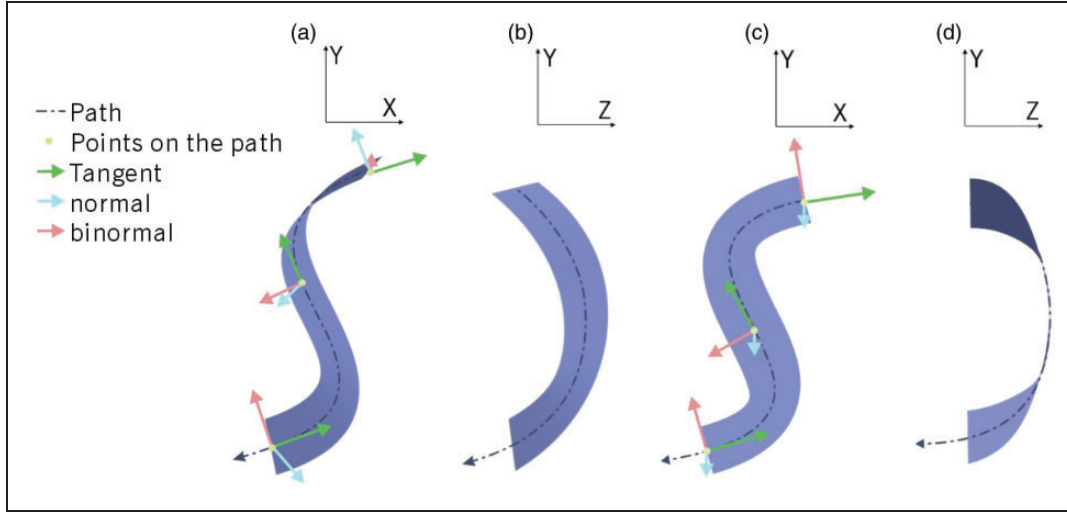

Figure 2. Angle correction of ribbon yarn mode: (a) original ribbon structure front view; (b) original ribbon structure left view; (c) correct ribbon structure main view and (d) correct ribbon structure left view.

for generation.[18] An *k*-th order NURBS curve can be represented as a piecewise rational polynomial vector function:

$$C(u) = \frac{\sum_{i=0}^{n} N_{i,k}(u) w_i A_i}{\sum_{i=0}^{n} N_{i,k}(u) w_i} \quad (1)$$

where $N_{i,k}(u)$ is the k-th order normalized B-spline basis function calculated from the knot vector $U = [u_0, u_1, \ldots, u_{n+k+1}]$, and w_i is the weight factor associated with the control points A_i. The original ribbon structure exhibits geometric distortion as it bends along the curve, as shown in panels (a) and (b) of Figure 2.

To meet the requirements of the simulation and ensure the correct deployment of the ribbon structure along the predetermined path, the Frenet-Serret frame was applied to calculate the local reference frame along the curve. The cross product of the path point tangent vector T and the Z axis vector (0, 0, 1) is taken to obtain the normal vector N, which is perpendicular to both the tangent and Z axis. Figure 2 in the study demonstrates the model changes during this calculation process. The key role of this normal vector N is that it defines the angle of the ribbon yarn model along the path, ensuring that the model maintains a consistent orientation even with changes in the path's curvature, as shown in panels (c) and (d) of Figure 2, effectively

avoiding model distortion and geometric distortion that might be caused by the path's complex curvature.

Compared to traditional 2D loop models, the 3D ribbon model directly presents the position of the yarn in space, more realistically reflecting the physical spatial arrangement and interaction of the yarn. This feature eliminates the need to separately calculate the overlapping relationships between yarns, significantly improving the efficiency of data processing and rendering.

Knitted fabric model

The characteristics of knitted fabrics are primarily manifested in their unique manufacturing processes and structural features. The most fundamental characteristic is that knitted fabrics are composed of a series of loops. These loops interlock with each other to form the fabric, endowing knitted fabrics with unique physical and visual properties.

In this study, the pathway of the knitted fabric loops is modeled using third-order NURBS curves, each defined by nine control points, as shown in Figure 3(a). This modeling approach is based on the loop model described by Chang Chenyu, which has been adapted for our analysis.[19] Since the last control point of each loop coincides with the first control point of the next loop, each loop in the fabric model is constituted by eight control points.

Once the model of a single loop is established, the next step is to arrange these loops in a specific way characteristic of plain weft knitting. In plain weft knit stitch, loops are arranged horizontally, with each loop connected not only to its immediate left and right loops but also interlocking with the loops of the rows above and below, thus forming the standard geometric model of plain weft knit stitch as shown in Figures 3(c) and 3(d).

Rendering based on the real image of the yarn

Compared to relying on expensive and operationally complex CT scanning technology to obtain 3D data,[20] the method of using yarn images directly as a data source offers significant convenience and cost-effectiveness,[21] providing a considerable data collection advantage for our research. This approach not only reduces dependence on specialized equipment but also lowers the barrier to commercializing laboratory technologies, making the simulation of yarns' true geometric shapes more aligned with the practical needs of industrial applications. High-quality simulations of knitted fabrics are achieved by preprocessing real yarn images
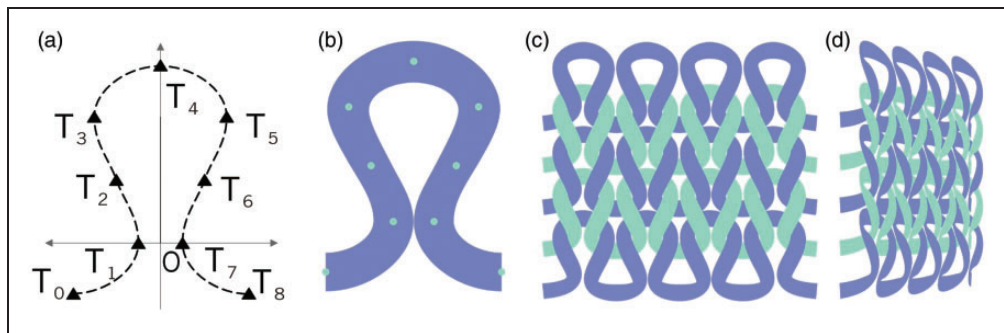

Figure 3. Plain stitch model: (a) loop geometric model; (b) ribbon loop model; (c) plain weft knit stitch standard geometric model front view and (d) plain weft knit stitch standard geometric model side view.
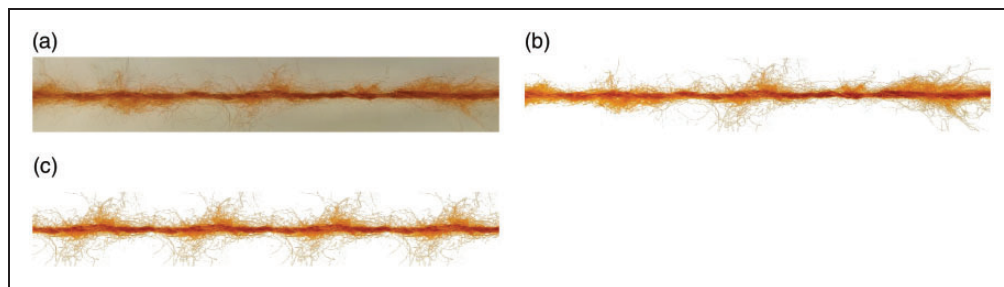

Figure 4. Yarn image processing: (a) cropped yarn image; (b) background-removed yarn image and (c) minimum unit repetition of yarn texture.

and rendering them using physically based rendering (PBR) techniques.

Yarn image processing and texture creation

In the study, a low-cost and easy-to-operate method was adopted to acquire yarn images, namely using consumer-grade smartphone cameras for photography. This method significantly lowers the threshold for image acquisition, ensuring that high-quality simulations are no longer dependent on professional photographic equipment.

In the image preprocessing stage of the study, the process starts with the original yarn images taken by a smartphone. The first step involves cropping the image to focus on the yarn itself and exclude unnecessary background elements or marginal parts, as shown in panel (a) of Figure 4. This ensures that subsequent steps only process image areas directly related to the yarn. After cropping, the brightness and contrast of the image are adjusted to ensure the clarity of the yarn's color and details in the image. The next step is to identify and isolate the main body of the yarn from the background, ensuring that the final texture only includes information about the yarn itself. This is achieved by applying binarization processing to highlight the main part of the yarn. This is done by setting an appropriate threshold T, assigning a value of 1 to pixel values greater than or equal to T, and a value of 0 to those less than T. For each pixel location (x, y) and corresponding color channel z, the original image I and binarized image B are combined through point multiplication to generate a new image Y. Mathematically, this operation can be represented as:

$$Y(x, y, z) = B(x, y) \cdot I(x, y, z) \qquad (2)$$

In this expression, $I(x, y, z)$ represents the pixel value of the original image at color channel z for coordinates (x, y), and $B(x, y)$ is the value at the same coordinates in the binarized image. The result Y contains pixels from the original image corresponding to non-zero values in the binarized image while setting the pixel values corresponding to the background to zero. This operation ensures that only the area of the yarn is extracted, and the background is eliminated, as shown in Figure 4(b).

The minimal repeatable unit of the yarn texture is determined based on the ability to reuse certain yarn texture portions in the rendering process while maintaining texture integrity. This selection criterion is established according to the structural features and repetitiveness of the yarn, ensuring continuity and consistency in yarn texture mapping during rendering. This step is crucial for achieving continuity and consistency

of the yarn texture mapping, especially in large-area rendering. The subsequent step involves ensuring the horizontal continuity of the image's minimal repeating unit, a concept referred to as 'bilinear continuity' in graphics, as demonstrated in panel (c) of Figure 4. This ensures that during the rendering process, when the texture is applied to the 3D model, the texture of the image seamlessly joins at the seams, without obvious breaks or repetitions, thereby visually creating a more continuous and realistic surface effect. These steps collectively ensure that the physical and visual properties of the yarn are accurately and realistically represented during the rendering process.

In contrast to relying on simplified lighting models and texture mapping to simulate textiles, the PBR approach is based on the physical principles of the real world, taking into account the effect of the material's microsurface structure on light scattering.[22] It can precisely simulate the reflection, refraction and absorption of light on the surface of textiles based on real yarn images, thereby capturing the unique luster and texture details of textiles. Various key maps transformed from real yarn images, including base color maps, roughness maps and metalness maps, are used. They respectively represent the material's basic color, surface roughness and metallic properties. Additionally, a height map is used to simulate the microstructure of the material's surface, enhancing the texture's three-dimensionality. Ambient occlusion maps enhance the local shadow effects of the model, providing a more natural shadow representation based on the model's geometric shape. These maps collectively participate in the rendering process, enhancing the realism of the material's response to light and shadow, thereby increasing the realism of the entire scene.

Simulation of light behavior in rendering

To ensure the correctness of the lighting effects, two key strategies were adopted in the study. Firstly, all the normals of the vertices were uniformly directed towards the positive direction of the Z-axis. This method simplifies the computation of lighting and ensures that under any lighting conditions, the model maintains the correctness of lighting effects, thereby providing consistent and accurate visual effects.

Secondly, smoothing of the model surface normals was performed.

$$\hat{N} = \frac{\text{prevNormal} + \text{currentNormal}}{\| \text{prevNormal} + \text{currentNormal} \|} \qquad (3)$$

By calculating the average of the normals of adjacent vertices, the occurrence of hard edges and incoherent lighting on the surface was reduced, making the

yarn model appear smoother and more visually natural in the study.

To achieve precise simulation of the fabric material and optical properties, this study employed the Bidirectional Reflectance Distribution Function (BRDF).[23] This function integrates information from various texture maps to modulate optical properties such as reflectance and surface roughness, thereby precisely simulating the interaction between light and material surfaces to produce more realistic and detailed visual effects.

The use of the Fresnel-Schlick approximation for simulating the Fresnel effect, which describes the variation in reflectance as light strikes the material surface at different angles, was implemented. This approximation formula provides visually satisfying results while maintaining computational efficiency.

$$F_r(u) = F_0 + (1 - F_0)(1 - \cos(u))^5 \quad (4)$$

In this formula, $F_r(u)$ represents the reflectance at a specific angle u. Here, F_0, which is the reflectance at normal incidence (i.e., when the light is perpendicular to the surface), is derived from both the albedo and metallic maps. The variable u is the cosine of the angle between the direction of the incident light and the surface normal, modified by the normal map.

Considering the microstructure of the material surface, the Beckmann distribution is used to simulate how light reflects off microscale roughness, controlled by the roughness map. This map adjusts the α parameter in the Beckmann formula, which is given as:

$$D(\hat{m}) = \frac{\exp\left(-\frac{\tan^2(\theta_m)}{\alpha^2}\right)}{\pi\alpha^2\cos^4(\theta_m)} \quad (5)$$

where $\hat{m}$ is the observation direction, θ_m is the angle between the microsurface normal and the macrosurface normal, again influenced by the normal map.

Considering the occlusion effect between microstructures, functions like the Schlick-GGX geometry function G are used for calculation, with the formula:

$$G(\hat{v}, \hat{l}, \hat{m}) = \frac{G_1(\hat{v}, \hat{m})G_1(\hat{l}, \hat{m})}{4(\hat{n} \cdot \hat{v})(\hat{n} \cdot \hat{l})} \quad (6)$$

where $\hat{v}$ is the observation direction, $\hat{l}$ is the light direction, $\hat{m}$ is the microsurface normal and $\hat{n}$ is the macrosurface normal, with the normal map fine-tuning these vector directions.

The BRDF is the overarching framework that connects these different factors to calculate the final reflected light, used to synthesize various optical effects in computing the reflected light on the material surface. The specific formula can be represented as:

$$f_r(\lambda, \hat{v}, \hat{l}) = \frac{DF_rG}{4(\hat{n} \cdot \hat{l})(\hat{n} \cdot \hat{v})} \quad (7)$$

where f_r is the BRDF function, λ is the wavelength, $\hat{v}$ and $\hat{l}$ are the observation and light source directions, respectively, and $\hat{n}$ is the surface normal.

Through the above discussion, it can be seen that BRDF provides a powerful tool for simulating the optical properties of complex material surfaces. Particularly crucial in fabric simulation is that it allows us to maintain computational efficiency while reproducing highly realistic visual effects.

Spatial accuracy and transparency effects in rendering

For the complex interlacing and occlusion among yarns, depth testing and depth writing techniques are employed to ensure realistic and accurate visual effects. This technique allows for the precise rendering of each yarn's position, whether on the surface or obscured by other yarns. By recording pixel depth information in the depth buffer, the overlapping relationships between yarns are effectively managed. This approach visually realizes the correct representation of nearer yarns covering farther ones and ensures that subsequent rendering of pixels correctly processes occlusion relationships based on the latest depth information. This method not only enhances the realism of the rendering effect but also ensures rendering efficiency and accuracy.

For the rendering process that involves removing the background texture of the yarn, transparency processing becomes a crucial step. By setting an Alpha threshold, the renderer can determine which pixels are opaque enough to be rendered and which should be considered transparent and ignored. This processing is vital for creating realistic transparency effects in the yarn. Additionally, standard Alpha blending modes are used:

$$C_{\text{final}} = \alpha C_{\text{src}} + (1 - \alpha)C_{\text{dst}} \quad (8)$$

C_{final} is the final color value, C_{src} is the source color value (i.e. the foreground color) and C_{dst} is the destination color value (i.e. the background color alpha is the Alpha value of the source color, used to determine the opacity of the foreground color). Through these calculations, the yarn color is smoothly combined with the yarn behind it or other backgrounds, thereby visually creating a natural and coherent transition effect.

In summary, the precise application of depth testing and depth writing techniques, combined with refined

transparency processing, ensures the accurate representation of the occlusion relationships between yarns and the natural transition of color blending. This makes the simulated fabric not only visually close to reality but also significantly improved in optical simulation accuracy.

Variance shadow maps

Ribbon geometries, due to their volumeless nature, may more easily produce rippling or other rendering artifacts when processing shadows. This is because the surface details and edge processing of ribbon geometries may not be as smooth as those of volumetric geometries.[24] During the rendering process, especially in light and shadow calculations, these unsmooth surfaces and edges may lead to the appearance of rippling effects. The situation becomes particularly complex when ribbon geometries need to receive shadows produced by themselves. Self-shadowing calculations need to consider the position of the light source relative to the geometry and the shape of the geometry itself. Due to the thinness of ribbon geometries, correctly calculating self-shadows and maintaining their naturalness is a challenge.

Variance shadow maps (VSM) technology uses depth values and depth variance to estimate whether a pixel is in shadow, thus producing a more natural gradient effect at shadow edges.[25] The VSM technique effectively addresses the shadow issues of ribbon geometries, particularly in mitigating rippling effects and rendering artifacts. This is particularly important for handling volumeless ribbon geometries, as the surface details and edges of such geometries may not be as smooth as those of volumetric geometries.

VSM first renders the scene from the perspective of the light source. For each pixel, a depth value is calculated, and the square of the depth d^2 is also computed and stored. By sampling the blurred depth map and the squared depth map, the expected depth $E[d]$ and the expectation of the squared depth $E[d^2]$ are obtained. The variance $\text{Var}[d]$ is then calculated as:

$$\text{Var}[d] = E[d^2] - (E[d])^2 \quad (9)$$

For each pixel in the observer's perspective, the depth value d_{frag} is compared with the corresponding depth value $E[d]$ from the light source perspective using the Chebyshev inequality, to calculate the adjusted depth value:

$$\text{light}_{\text{depth}} = E[d] + \sqrt{\frac{\text{Var}[d]}{d_{\text{frag}}}} \quad (10)$$

If $d_{\text{frag}} > \text{light}_{\text{depth}}$, then the pixel is considered to be in shadow. This method allows for a more natural gradient effect at shadow edges. Not only does it improve the smoothness of the shadow edges, but it also enhances the overall realism of the shadows.

Furthermore, VSM also aids in more effectively simulating self-shadows, especially in complex scenes with ribbon geometries. Through intelligent processing of depth information, VSM can produce more realistic and continuous shadow effects at different depth levels. As VSM supports standard 2D texture filtering, it can utilize GPU hardware acceleration to achieve smooth shadow edges without additional computational overhead. Although VSM requires higher precision textures, it typically provides high-quality shadow effects without sacrificing too much performance.

Results and discussion

The simulation and rendering of yarn and fabric in this study were conducted on a Windows 11 system equipped with an Intel Core i7-9750H CPU and 16 GB RAM. All processes were programmed in JavaScript and implemented in a WebGL environment using the THREE.JS library.

The core innovation of this study is a yarn simulation method that leverages 3D ribbon structures to enhance both the detail and computational efficiency of fabric simulations while improving the realism of yarn edge details. This method incorporates physically based rendering (PBR) techniques to more accurately depict the interaction of light with textile surfaces, enhancing texture and material portrayal. Additionally, the model addresses shadow generation challenges in ribbon-based structures through precise calculations of depth values and variances. Panels (b) and (c) of Figure 5 show the front and side views of the simulation results, providing a comprehensive visual representation of the simulated fabric. Panels (c) and (d) compare the initial color mapping effect with the final rendering, highlighting the enhancements in realism and detail of the simulated fabrics.

To illustrate the advancements achieved by the method in this study, a comparison was made between the simulation results of this study and those of Lu Z[5]. As shown in Figure 6, this comparison highlights the superior detail and realism achieved by our method, demonstrating significant enhancements in the visual and structural depiction of fabric texture.

As illustrated in Figure 6, our simulation of a 32*36 cable stitch demonstrates the practicality of our yarn model. During the simulation process, the modeling took 168 milliseconds, and rendering was accomplished in 24 milliseconds, achieving instantaneous simulation
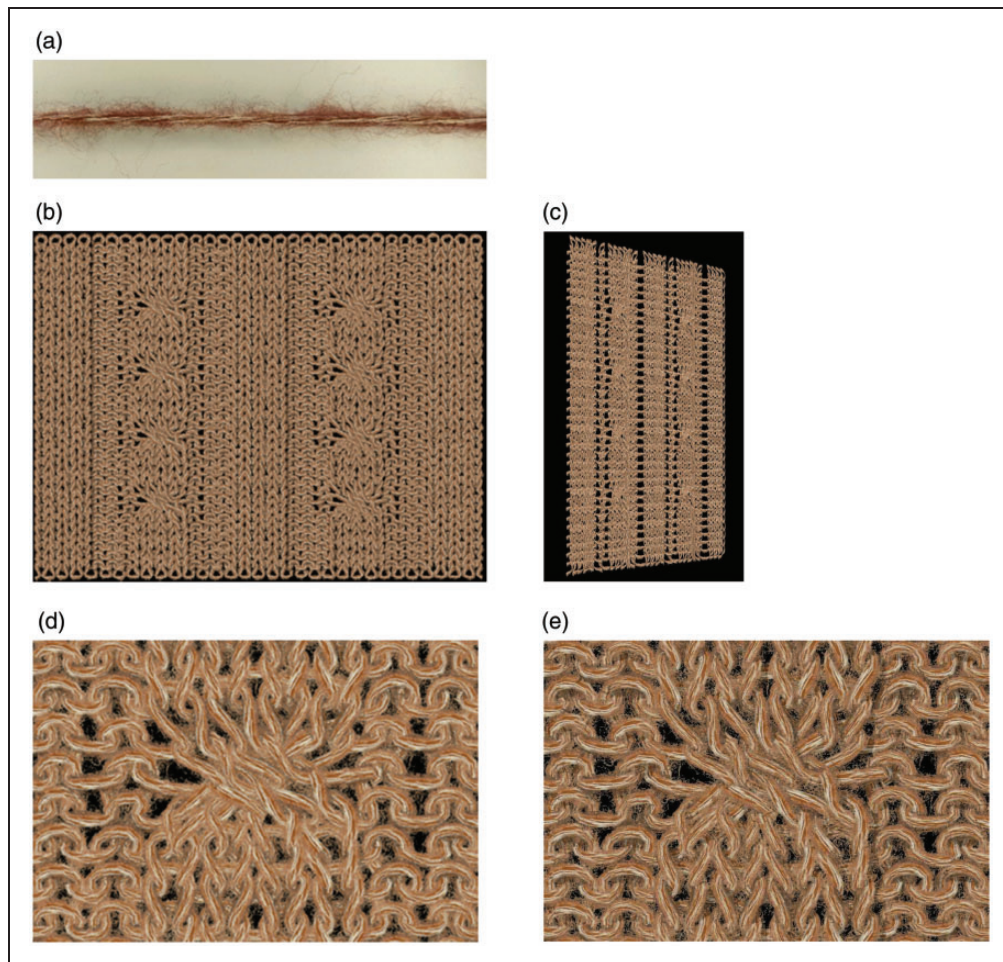
Figure 5. Cable stitch simulation: (a) original yarn image; (b) front view of simulation results; (c) side view of simulation results; (d) initial color mapping of fabric detail and (e) final rendering effect on simulated fabric.
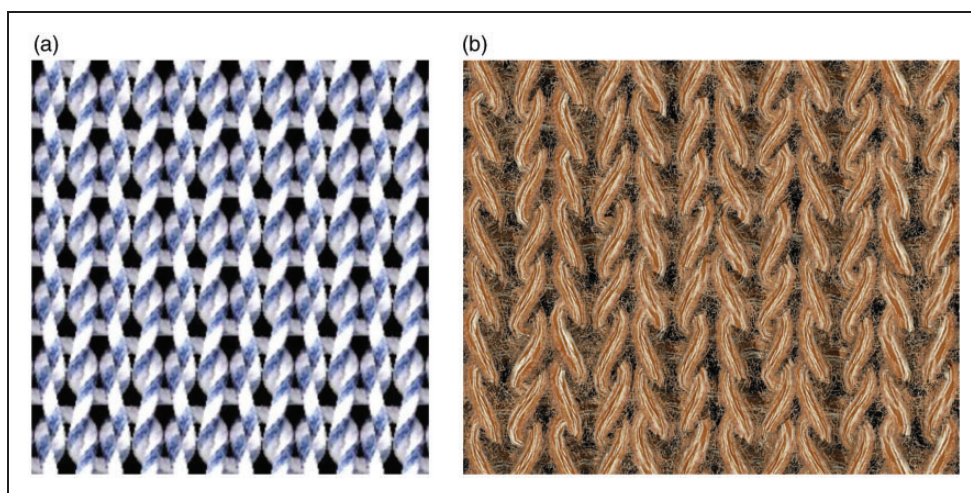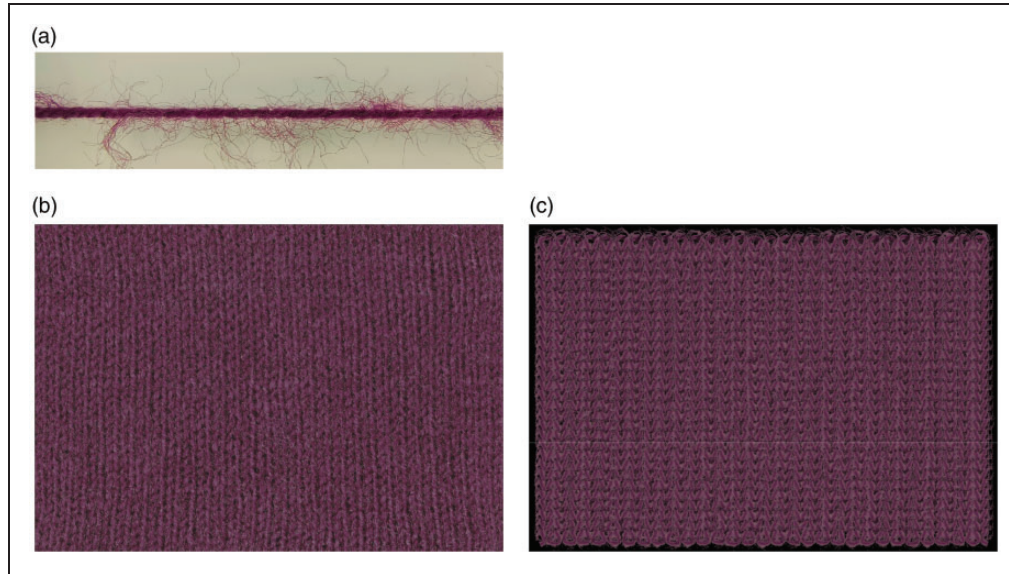

Figure 6. Comparison of fabric simulation results: Lu Z[5] versus this study: (a) fabric simulation results by Lu Z and (b) fabric simulation results from this study.

Table 1. Plain stitch modeling time

Fabric size	Control points	Each loop is subdivided into 15 segments (ms)	Each loop is subdivided into 20 segments (ms)	Each loop is subdivided into 20 segments (ms)
20*20	3220	163	188	382
50*50	20050	910	1167	3534
100*100	80100	4536	7708	23544

Figure 7. Comparison of yarn simulation with actual fabric textures: (a) images of original yarns; (b) images of real fabrics and (c) simulation rendering.
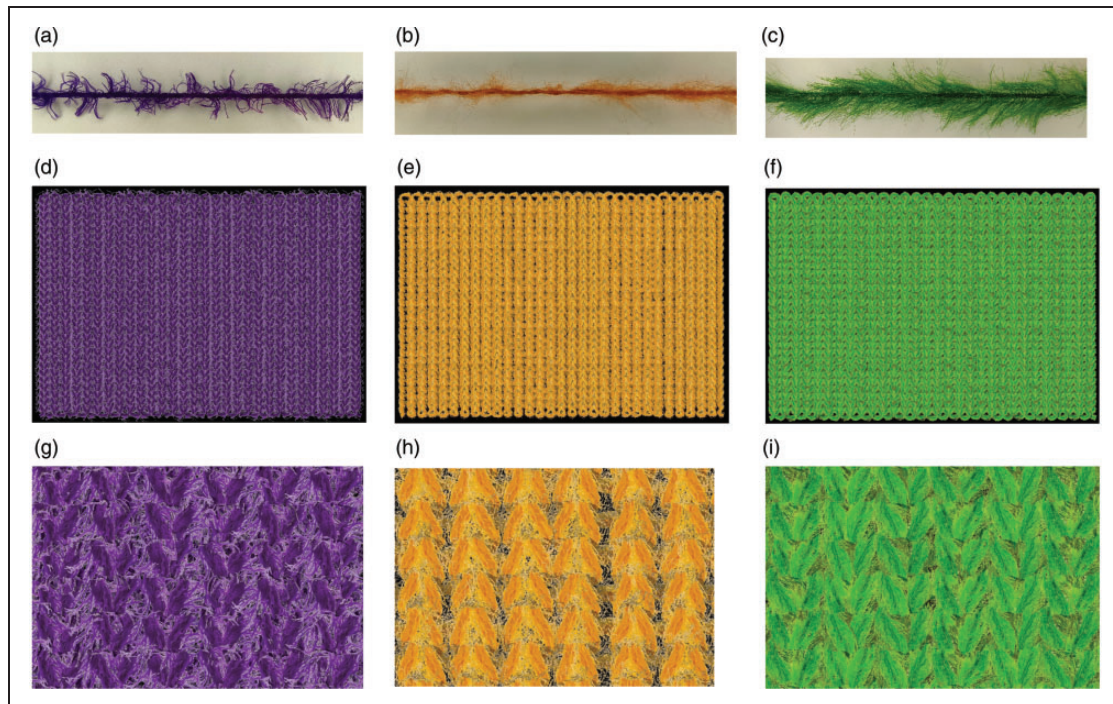
Figure 8. 30*30 plain stitch simulation: (a), (b), (c) original yarn image; (d), (e), (f) simulation rendering and (g), (h), (i) simulation rendering fabric detail.

speeds while maintaining high-quality simulation results.

In terms of simulation speed, as shown in Table 1, our modeling of the loop structure at different subdivision levels, including 15, 20 and 50 segments, reveals that subdivisions of 15 and 20 segments significantly enhance the simulation results without substantially increasing modeling time. This finding suggests that moderate subdivision levels effectively balance simulation realism with computational efficiency. On the other hand, a 50-segment subdivision shows limited visual enhancement but requires considerably more time, indicating a reduced return on computational effort at higher subdivision levels. Notably, the rendering time within our simulation system exhibits remarkable efficiency. Regardless of the size of the fabric model, the rendering time consistently maintains around just 24 milliseconds, a notably rapid performance in the field of knitted fabric simulation. This not only signifies our method's capability to efficiently process complex models but also suggests its vast potential in real-time rendering and dynamic simulation applications.

To demonstrate the effectiveness of this method, images of actual yarns and their corresponding fabrics are compared with the simulation outputs in Figure 7. These comparisons show that the simulation closely replicates the intricate details of real fabrics, validating our approach for accurate textile modeling. Additionally, Figure 8 illustrates the results of simulations using different yarns, further showcasing the versatility and robustness of our simulation techniques in textile modeling.

Conclusion

The simulation system developed in this study achieves high realism with quick and efficient task completion. Its key innovation is a yarn model simulation technique using 3D ribbon structures, enhancing the simulated fabric fuzz effect and speeding up the simulation. The rendering technology, including PBR and advanced shadow mapping algorithms, allows for rapid speeds and detailed presentation of light and fabric texture interactions. Experimental results show that this method effectively handles diverse and complex knitted fabric structures and textures. This not only demonstrates the technical sophistication of the method but also highlights its immense potential in practical applications, particularly in enhancing design efficiency, accelerating product development and contributing to innovative textile design. The application of this technology in real-world and industrial settings, especially in terms of improving design efficiency and speeding up

product innovation, showcases its substantial practical value and promising application prospects.

Author contributions

Song Mingming: Responsible for experimental work, data collection and analysis, and manuscript drafting.
Peng Jiajia: Responsible for research topic and objective determination, research plan formulation and manuscript refinement.
Chang Chenyu: Responsible for literature review.
Liu Feng: Responsible for creating physical fabric samples for the experiments.
Lu Zhiwen: Responsible for all aspects of the figures in the manuscript.

Conflict of interest statement

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ORCID iDs

Song Mingming ⓘ https://orcid.org/0009-0008-5421-8650
Peng Jiajia ⓘ https://orcid.org/0000-0001-9423-6847
Lu Zhiwen ⓘ https://orcid.org/0000-0001-9644-5749

References

1. Rong W, Jie C and Yu X. The characteristics of dyed fancy yarn formation pattern research. *Adv Mat Res* 2014; 599: 1149–1153.
2. Song M, Chang C, Sun Y, et al. Progression realistic modeling and simulation of weft knitted fabrics. *Adv Text Technol* 2023; 31: 255–266.
3. Castillo C, López-Moreno J and Aliaga C. Recent advances in fabric appearance reproduction. *Comput Wkly News* 2019; 84: 103–121.
4. Jiang G, Lu Z, Cong H, et al. Flat knitting loop deformation simulation based on interlacing point model. *Autex Res J* 2017; 17(4): 361–369.
5. Lu Z and Jiang G. Rapid simulation of flat knitting loops based on the yarn texture and loop geometrical model. *Autex Res J* 2017; 17(2): 103–110.
6. Zheng P and Jiang G. Modeling and realization for visual simulation of circular knitting transfer-jacquard fabric. *Text Res J* 2021; 91(19–20): 2225–2239.

7. Peng J, Jiang G, Xia F, et al. Deformation and geometric modeling in three-dimensional simulation of fancy weft-knitted fabric. *Text Res J* 2020; 90(13–14): 1527–1536.

8. Ozdemir H and Baser G. Computer simulation of plain woven fabric appearance from yarn photographs. *J Text I* 2009; 100(3): 282–292.

9. Özdemir H. Computer simulation of woven fabric appearances based on digital video camera recordings of moving yarns. *Text Res J* 2008; 78(2): 148–157.

10. Zhongjian L, Fei Y, Ning Z, et al. Automatic construction of digital woven fabric by using sequential yarn images. *Autex Res J* 2019; 19(2): 147–156.

11. Song B, Pan R, Zhang N, et al. Visual similarity simulation of slub denim based on image processing. *Text Res J* 2024; 94(1–2): 120–135.

12. Haisang L, Yordan K, Gaoming J, et al. Realistic fabric rendering with yarn models. *Text Res J* 2023; 93(15–16): 3552–3563.

13. Zhong H, Xu YQ, Guo B, et al. Realistic and efficient rendering of free-form knitwear. *Visual Comp Animat* 2001; 12(1): 13–22.

14. Zhao S, Luan F and Bala K. Fitting procedural yarn models for realistic cloth rendering. *ACM Trans Graph* 2016; 35(4): 1–11.

15. Saalfeld A, Reibold F and Dachsbacher C. Image-based Fitting of Procedural Yarn Models. In: *MAM@ EGSR* 2018; 19–22.

16. Kui W and Cem Y. Real-time cloth rendering with fiber-level detail. *IEEE transactions on visualization and computer graphics* 2019; 25(2): 1297–1308.

17. Aliaga C, Castillo C, Gutierrez D, et al. An appearance model for textile fibers. *Comput Graph Forum* 2017; 36(4): 35–45.

18. Dimas E and Briassoulis D. 3D geometric modelling based on NURBS: a review. *Adv Eng Softw* 1999; 30: 741–751.

19. Chang, C, Song, M, Liu, F, et al. Structural deformation dynamic simulation of weft knitted fabric basic stitch. *J Text I*, 2024; 1–10.

20. Zhao S, Jakob W, Marschner S, et al. Building volumetric appearance models of fabric using micro CT imaging. *Commun ACM* 2014; 57(11): 98–105.

21. Alcain R, Heras C, Salinas I, et al. Microscale optical capture system for digital fabric recreation. In: *Phot* 2019; 114–119.

22. Dana K, Van Ginneken B, Nayar SK, et al. Reflectance and texture of real-world surfaces. *ACM Trans Graph* 1999; 18: 1–34.

23. Li T-M, Aittala M, Durand F, et al. Differentiable Monte Carlo ray tracing through edge sampling. *ACM Trans Graph.* 2018; 37: 1–11.

24. Crow F. Shadow algorithms for computer graphics. In: *Siggraph '77: Proceeding of the 4th annual conference on computer graphics and interactive techniques* 1977; 11: 242–248.

25. Donnelly W and Lauritzen A. Variance shadow maps. In: *Proceedings of the 2006 symposium on interactive 3D graphics and games* 2006; 161–165.