CEIG - Spanish Computer Graphics Conference (2022)
J. Posada and A. Serrano (Editors)

Interactive visualization of historical fabrics at yarn level using Sphere Tracing

M. Pérez-Aixendri[1], J. Gimeno[1], C. Portalés[1] P. Casanova-Salas[1]

[1]Universitat de València, Spain

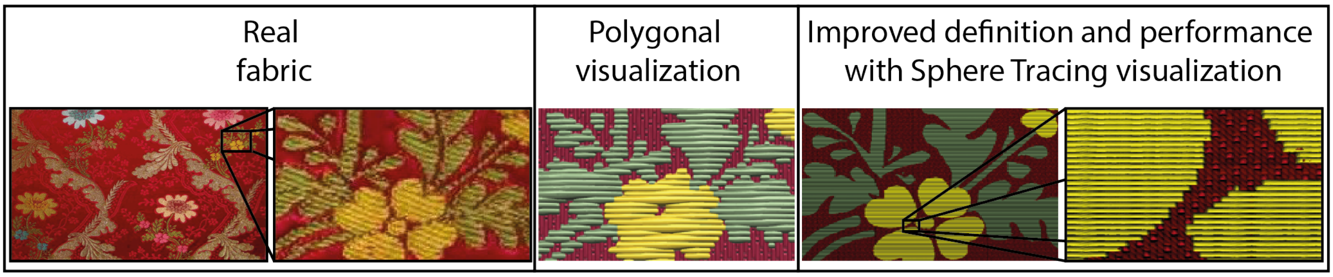

Real
fabric
Polygonal
visualization
Improved definition and performance
with Sphere Tracing visualization

Figure 1: *Historical fabric reconstruction using a rasterized model with a low level of detail (few yarns) and the proposed approach using Sphere Tracing with a very high number of threads*

Abstract

*In this paper we present preliminary results of a new approach for interactive visualization of historical fabrics at yarn level based on models using Signed Distance Functions. They are compared with the results of Virtual Loom, an interactive application for rendering historical fabrics from the 15th to the 19th century, resulting from the European SilkNow project [PSA*18]. Yarns used on these fabrics has an special complexity due to a metallic cover that appears in some elements of the fabric. This new approach is based on rendering using Sphere Tracing. We will propose a signed distance function that allows both the visualization of a single yarn and a complete fabric in real time.*

CCS Concepts

• ***Computing methodologies*** → ***Ray tracing;***

1. Introduction

The three-dimensional graphic representation of fabrics is a challenge that has been tried to be solved by different methods. We can make a representation of the fabrics that represent their surface by means of a planar polygonization and a set of textures that provide illumination information. Other methods take into account the macroscopic visualization of the fabric that allow users to appreciate the internal yarn structure. The yarn-level representation is essential when the purpose of the visualization is to know how the fabrics have been weaved. If we focus on real-time visualization, the number of techniques that we can choose from becomes smaller. [LMMCO17] presents a rasterized method based in GPU voxelization that allows a very high detailed preview and a faster offline raytracing than classic methods. [WY17] presents a fiber-level cloth rendering framework with GPU geometry creation using tessellation. This technique describes a LOD approach that allows real-time rates. The polygonal geometry used in Virtual Loom [PPCSG21] is based on using geometric primitives such as the extrusion of a cylinder along the path that each of the yarns will take. In order to draw a more realistic fiber without increasing the polygonization, an ad hoc shader was developed. This shader together with textures and a set of parameters add effects such as color, yarn twist, internal fibers, etc. Due to the graphical power of current systems, it is possible to perform a representation using polygonal geometry as has been done in Virtual Loom. Even so, this method just allow a limited number of fibers, since a high detail representation of the fabric requires, in general, a large amount of geometry.

DOI: 10.2312/ceig.20221146

2. Sphere Tracing for fabric visualization at yarn level

Rasterization of 3D surfaces for their representation takes advantage of the classic graphic pipeline, although it has several drawbacks. The main one is the large amount of graphic resources needed when a very detailed model is required. When this happens, it becomes difficult to maintain enough frame rate for real-time visualization. A yarn-level representation of a fabric is usually a very complex model and requires a large amount of geometry. For these reasons, it seems a good option to look for alternative visualization methods. For the representation of figures such as cylinders, spheres, extrusions or in general, more complex figures with a known implicit surface function, we can use other drawing methods different from rasterization. Hart in [Har96] already points out these problems and indicates the suitability of ray tracing for the representation of implicit surfaces. We can represent the ray equation as eq.1.

$$r(t) = r_o + tr_d \tag{1}$$

Where r_o corresponds to the camera position, r_d is the normalized ray direction and t is the intersection distance between the camera and the surface in the direction of vector r_d. Given the implicit function of the surface f, the ray can intersect the surface at different points. Intersection are produced when $f(r(t)) = 0$. The root searched should be the one with the smallest positive t value. As per pixel techniques, these algorithms have a cost proportional to the size of the final image and not to the polygonal complexity of the scene. In these cases, complexity depends of how many operations per pixel are performed. On the other hand, figures visualized using ray tracing are smooth since they do not have the inherent error of polygonization technique.

2.1. Sphere Tracing

The Sphere Tracing algorithm [Har96] optimises the ray tracing calculation when dealing with implicit surfaces by making the forward step in the ray β non-constant. In this case, the algorithm works with functions known as Signed Distance Functions or SDF [BS91]. SDF returns how far a point in euclidean space is from the surface. We can calculate this distance from the point r_o. The value d returned by the function allows to define a sphere of radius d with origin at r where no point of the surface will fall inside this sphere. This sphere intersects with the vector r_d so that we can make a forward step of distance d and perform the calculation iteratively again. This calculation will terminate when the distance approaches a given error or when the accumulated distance from r_0 is greater than a limit (fig.2). Current graphic hardware allows this ray tracing algorithm as an alternative for real time visualization.

3. Current polygonal implementation and its limitations

Threads represented in Virtual Loom are made up of twisted yarns and, occasionally, covered by a metallic covering, something characteristic of some embroideries on historical fabrics (fig. 4). To represent a yarn, we have taken into account the parameters presented in [PPCSG21]. These parameters can be grouped into parameters of the yarns and parameters of the metallic covering. In the following, we will briefly describe each of them:

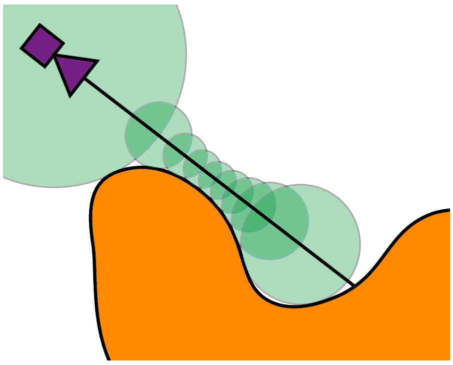
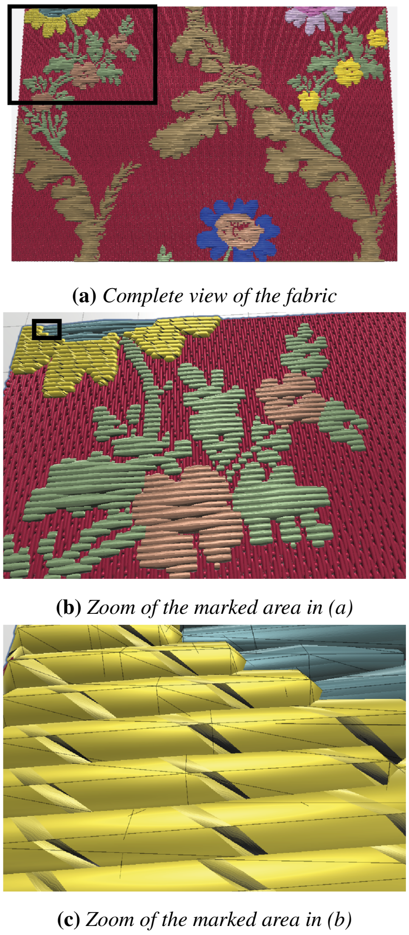


Figure 2: *Sphere Tracing with variable step*



(a) *Complete view of the fabric*

(b) *Zoom of the marked area in (a)*

(c) *Zoom of the marked area in (b)*

Figure 3: *Full model of an spolin with 450x300 yarns using polygonal representation*

- Fibre parameters
    - Section radii: The section of each fibre is described by two values R1 and R2. R1 is the radius of the fibre section. R2 is the radius of the fibre compressed when woven. This

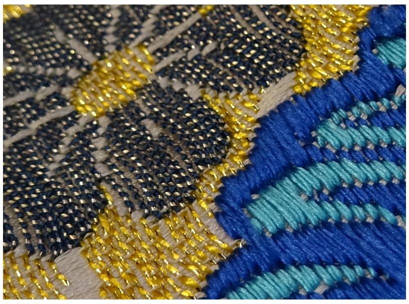


Figure 4: *Detail of an historical fabric weaved in silk and metallic yarns (golden ones)*

compression is only applied in the axis perpendicular to the weave.

- Composition: Each yarn is made up of a set of threads. This value indicates how many threads form the outside of the yarn. We do not need to know the inner threads as they are not necessary in the surface model.
- Fibre twisting: Yarns twist the threads to increase mechanical strength. This twist can be Z-twist or S-twist, depending on the direction of the twist. Virtual Loom uses positive values for S-twist and negative values for Z-twist.
- Colour: This parameter indicates the albedo of the yarn.
- Metallicity: This parameter modifies the light response of the fibre. It represents how much metallic it seems.

- Metallic cover parameters
  - Bandwidth: Indicates the width of the metallic band around the fibres. Its value is normalised. 0 means no band and 1 means a complete band that hides the entire core.
  - Twist: Indicates how many turns do the metallic cover per linear unit in the yarn.
  - Colour: This parameter indicates the albedo of the metallic strip.
  - Metallicity: This parameter modifies the light response of the metallic strip based on its response to light, giving it a more or less metallic appearance.

In addition to these parameters in the application, Virtual Loom defines the number of vertices to polygonise the yarn section and the number of sections along the fibre path. These two parameters, the selected weaving technique and the masks of the pictorial parts of the reconstructed fabric, define the final geometry created in the application (fig. 3). The tests performed during the results phase of the project showed that generating geometrical models above 400 yarns by the corresponding number of horizontal yarns (weft plus embroidery) increased mesh generation time over 2 seconds and framerates were below 30 fps in medium specs computers. Given this limitation, a new visualization mechanism based on Sphere Tracing and SDF-based fabric modelling was sought.

4. Thread modeling using SDF

This new visualization model is implemented as a combination of a fragment shader with a vertex shader. We use a proxy geometry in order to reduce the fragment shader calculation. If our final target was to draw a single line we would try to find a proxy volume that reduce fragment shader calculations using something similar to a pipe with a geometry similar to our first visualization approach. However, we want to visualize a full fabric. So, in our case, a simplier proxy volume can be used: the bounding box of the fabric. Width and heigth of the box are determined by the source image of the real fabric. The depth of this bounding box is defined by a minimum yarns axial volume plus a Z gap defined by user that allows to separate different yarns in order to examine the different weaving techniques. To model a yarn for rendering by Sphere Tracing we can use SDFs that approximate its shape. In our case we have relied on the functions of [Qui08]. These functions allow several actions: drawing primitives, CSG operations. Another functions in the catalogue allow to perform twist **(T)** on a given axis, so we cover all the operations to be performed for the modelling of a yarn with a set of parameters similar to those of the Virtual Loom polygonal model. Since our model consists of the fibres together with the metallic cover, we will briefly describe the SDFs created in both elements (fibres and cover) to obtain the final model. Once computed, we will obtain the union of both figures by calculating the minimum between the two SDFs. The Figure 5 indicates graphically the functions used and how they are performed.

4.1. SDF modelling of fibres

The basic primitive in this case is a modification of the capsule SDF that allows n capsules to be drawn on the perimeter of a circle of radius r. This can be done using the repetition operator in polar coordinates **(PR)**. After repetition we apply twist **(T)**. This function is used twice in a nested way with different parameters as the yarns are formed by twisted threads which in turn are formed by plys distributed in the same way and twisted again.

4.2. SDF modelling of the metal covering

To model the metal cover we will start again from a capsule to which we will apply a torsion. In this case the torsion will be applied from an axis far from the capsule axis to produce a figure similar to a spring. This figure does not correspond exactly to the covering of the metallic wires as this covering is flat. To achieve this effect we calculate the intersection of the spring with a hollow capsule. This hollow capsule is obtained by applying the onion function **(O)** on a capsule.

5. Yarn level modelling of weaved fabrics

Starting from the model of a yarn we can draw the complete fabric using the modulus function described in [Qui08]. If we apply this function to the point of the evaluated ray, the shader will draw infinite yarns on the selected axis. If we use the modulus function with interval we can limit the repetition to the required number of yarns. This method will be used to draw the fibres of each of the layers of the fabric. There is always at least two layers (weft and warp). To these ones we can add other pictorial elements that will repeat the same process for each of the motifs. The thread interlacing logic in Virtual Loom is based on calculating whether the thread travels along the top or the bottom of the fabric based on the values of a

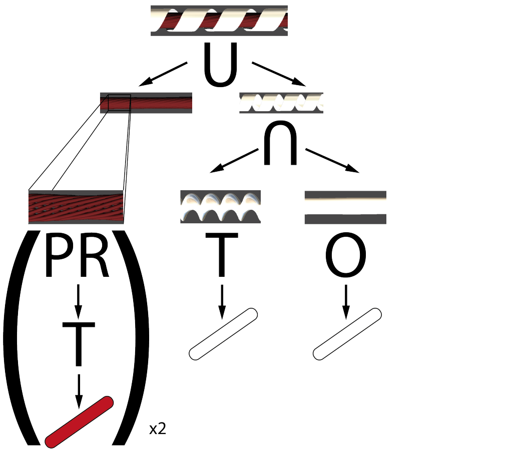
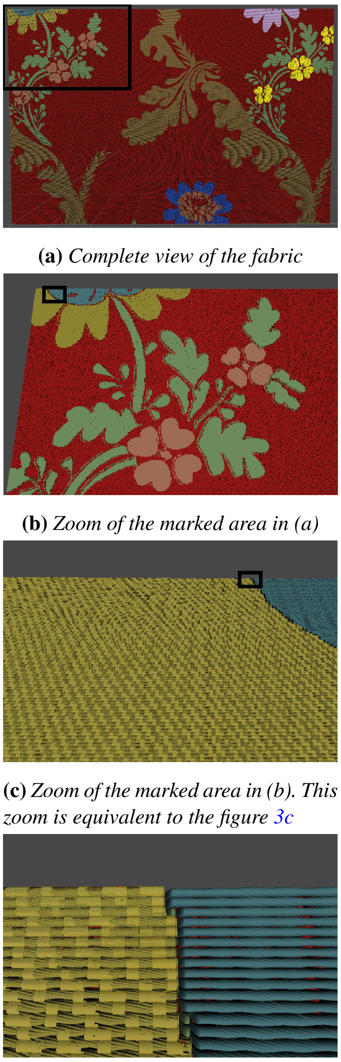


Figure 5: *This tree shows how diferent signed distance functions are performed. **PR**: Polar Repetition, **T**: Twist, **O**: Onion function (carving interior of a shape)*

set of masks. In this new approach, the logic used is similar. The advantage of the SDF model is that moving yarns up or down just requires the displacement of the calculated ray. There is no need to recalculate any poligonization as the current poligonal implementation.



(a) *Complete view of the fabric*

(b) *Zoom of the marked area in (a)*

(c) *Zoom of the marked area in (b). This zoom is equivalent to the figure 3c*

(d) *Zoom of the marked area in (c)*

Figure 6: *Full model of an spolin with 8192x5346 yarns using Sphere Tracing*

6. Preliminary results

Firstly, it should be noted that the new model implemented has the great advantage that there is no prior geometry creation time. This allows any modification in the fabric creation parameters to be applied instantaneously: weaving techniques, yarn types, colours, etc. Regarding the interaction capacity, we must point out that the limitations in the case of the polygonal model are marked by the geometric load. In the case of the new model based on Sphere Tracing and SDF, the load will depend on the resolution of the image we generate. In the performance tests carried out during the development of Virtual Loom, geometry of 450x300 yarns (135k) was generated. In some cases, on mid-range and low-end computers, 30 fps were not achieved on fabrics such as an spolin with 10 pictorial areas. On high-end computers, refresh rates of 60 fps could be achieved, but the number of threads was not as high as desired. In the preliminary test of the new visualization system is possible to draw 8192x5346 yarns (43.8M) on a high-end computer in 720p resolution at 30fps. In figure 6 can be observed different zoom levels of a full spolin. User can explore the full fabric in real time and observe the weaving technique over its entire surface. This visualization system is only penalised by the resolution of the final image. The number of yarns to be drawn only affects the amount of memory reserved to store the textures with the necessary information. In addition, a LOD system has started to be implemented within the SDF, which allows the calculation of the minimum distance to be accelerated by reducing the number of calculations per pixel. This points to the possibility of increasing resolution and/or framerate in the future.

References

[BS91] Bloomenthal J., Shoemake K.: Convolution surfaces. *ACM SIGGRAPH Computer Graphics 25*, 4 (July 1991), 251–256. URL: https://doi.org/10.1145/127719.122757, doi:10.1145/127719.122757. 2

[Har96] Hart J. C.: Sphere tracing: a geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer 12*, 10 (1996), 527–545. URL: https://www.academia.edu/156766/Sphere_tracing_A_geometric_method_for_the_antialiased_ray_tracing_of_implicit_surfaces. 2

[LMMCO17] Lopez-Moreno J., Miraut D., Cirio G., Otaduy M. A.: Sparse GPU voxelization of yarn-level cloth. 22–

34. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12782. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12782, doi:10.1111/cgf.12782. 1

[PPCSG21] PORTALÉS C., PÉREZ M., CASANOVA-SALAS P., GIMENO J.: Virtual Loom: a tool for the interactive 3D representation of historical fabrics. *Multimedia Tools and Applications 80* (Apr. 2021), 1–26. doi:10.1007/s11042-020-10294-w. 1, 2

[PSA*18] PORTALÉS C., SEBASTIÁN J., ALBA E., SEVILLA J., GAITÁN M., RUIZ P., FERNÁNDEZ M.: Interactive tools for the preservation, dissemination and study of silk heritage—an introduction to the SILKNOW project. 28. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. URL: https://www.mdpi.com/2414-4088/2/2/28, doi:10.3390/mti2020028. 1

[Qui08] QUILEZ I.: Inigo Quilez, 2008. URL: https://iquilezles.org. 3

[WY17] WU K., YUKSEL C.: Real-time fiber-level cloth rendering. In *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, Feb. 2017), I3D '17, Association for Computing Machinery, pp. 1–8. URL: https://doi.org/10.1145/3023368.3023372, doi:10.1145/3023368.3023372. 1