Decoration of distorted 3D objects

Abstract This paper introduces a new method to bind the techniques of texture mapping to a physical process. In the general case of texture mapping, the surface (mesh) adapts to the texture through the parameterization. In our context, the mesh comes from a physical distortion of an initial plane mesh, and our goal is to find the texture attached to the initial object, which gives the wanted result after distortion. To achieve this goal, a specific parameterization is defined and the inverse transformation is applied to the mesh.

Keywords Texture Mapping · Parameterization · Physically-Based Deformation

1 Introduction
The overall goal of this paper is to provide an automated method to design and decorate industrial 3D objects. These objects are built by plastic injection process. A film with a printed picture is placed in the mold, this film and its picture is then mapped on the object during injection. Each material follows his own distortion law. The knowledge of this law enables to simulate the distortion of an object. Then we recover the wanted planar texture from this simulation data.

We focused on the problem of parameterization and inverse deformation. Our method returns the image to be printed from the deformed mesh and the image to place on the object. Both problems are discussed successively in the following method.

2 Previous Work
The problematic of parameterization is very old, because it was discussed as soon as the man tried to map the entire Earth. The texture mapping is studied for decades, and is implemented in all graphic processing chains. Mapping an image on a 3D mesh requires prior calculation of texture coordinates for each vertex of the mesh [1]. To do this, many methods adapting the mesh to the size of the image have been developed. They aim at making the mesh plane and thus at mapping the image directly [3]. But, such a transformation cannot be performed without causing distortions into the mesh, except for developable surfaces, as a cylinder or a cone.

The parameterization technique is well-controlled for simple or regular objects, but becomes more problematic for the complex ones, in particular containing concavities, holes or handles. Conventional methods try to preserve either angles or areas, or, more effectively, a compromise between the distortions related to these two measures [2]. These methods then define a specific energy, corresponding to the chosen requirements that we just have to minimize. Instead of using only geometrical criteria, our approach takes into account physical processes: 3D objects are distorted during their manufacturing (in practice, by plastic injection), the parameterization is calculated from the inverse distortion.

3 Parameterization and inverse distortion
Our method has several steps: from the entry data, a mesh and a picture, first we texture the mesh, then we make the mesh inverse distortion and finally we save the image to print (Figure 2). The mesh is limited for now to an open mesh with rectangular edge. This limitation is not restrictive, because the technical process used in the project transfers the printed decoration on a regular film. We first get the mapping of this image on the mesh, so that the visual rendering is equivalent to the original image. For this, we choose a point of view, and favor it. It is selected for now with the constraint of making the mesh almost projectable. Then the mesh is
parameterized using an orthogonal projection defined by this point of view (Figure 1). Thus the coordinates \((u, v)\) obtained are used to set the texture coordinates to bind to the initial mesh and then reach the first goal, realizing the texture mapping with visual constraints.

![Fig. 1 Visual benefit of a viewpoint choice for the mapping](image)

The type of mesh used upstream for the simulation of distortion is a regular mesh. We took advantage of this property, in order to realize quickly the inverse mesh transformation. We simply redefine each point within the mesh, as the barycenter of its immediate neighbors. This technique is iterated until the sum of the movements of all points falls below a fixed threshold. We thus obtain a nearly regular plane mesh representing the initial simulation mesh, whose associated texture has also been modified by the inverse transformation. We therefore visualize the result to be printed on the film.

We have to save an image close to what we see. We already know which part of the image is associated to each mesh point, thanks to the parameterization. It only remains to browse in a bilinear way the mesh inside and simultaneously the corresponding part of the image, so as to fill the pixels of the image. Finally this enables to obtain the file containing the image to be printed. First the method was validated with data resulting from the simulation of the distortion. This simulation distorts a square grid. Furthermore the edge remains unchanged during deformation. The used texture is a black and white checkered, which enables to visualize the effects of distortion. We also test others textures, including geometric repetitions: text, newspaper extracts, urban landscapes (Figure 3).

![Fig. 2 method overview](image)

**Fig. 2** method overview

**Fig. 3** Images to print on the film

that the whole grid can be projected onto a plane without self-intersection. This characteristic corresponds to a cone of vision as axis a straight orthogonal to the projection plane.

The decoration of more complex objects requires to transfer images printed on several films, each of them applied on a limited area to take into account the mechanical constraints of the films (heat and tear resistance). The edge of the films could be not rectangular. Our work is to define the mesh patches, in order to parameterize locally from an own point of view.

**4 Conclusion and futur work**

We have presented in this paper a method to get exactly the wanted visual result for a texture mapping from a viewpoint, and to obtain the image to print on the film for the industrial process. The first mesh used is a regular square grid, previously artificially deformed, so

**References**