International Conference on Artificial Reality and Telexistence
Eurographics Symposium on Virtual Environments (2015)
M. Imura, P. Figueroa, and B. Mohler (Editors)

A Projection Simulator to Support the Development of a Spherical Immersive Display

Wataru Hashimoto[1], Yasuharu Mizutani[1] and Satoshi Nishiguchi[1]

[1]Osaka Institute of Technology, Japan

Abstract
The main purpose of this study is to develop a piece of software for the construction of spherical immersive displays. In the process of designing spherical displays, it is necessary to consider about optical design before their construction. A dome-shaped screen usually produces some distortion on the projected image. The pre-distortion method is generally used to compensate that distortion. Our projection simulator aims to easily compute the distortion and to generate the warped image right after the user changes the parameters related to the screen and projectors.

Categories and Subject Descriptors (according to ACM CCS): H.5.1 [Multimedia Information Systems]: Artificial, augmented and virtual realities.

1. Introduction

A Spherical Immersive Display (SID) is a device such as the planetarium and the theater. It is possible to provide the user with a high sense of realism and a huge amount of information. However it is difficult to build a SID due to the large-scale equipment needed. Moreover, a projection on a curved surface causes image distortions. Since the image warping, alignment and calibration techniques are too complex to easily develop the system, it is inconvenient for end-users to use the display system.

The final goal of this study is to simplify the process of building the spherical display system in order to help inexperienced developers. The final product will provide support to professionals, such as teachers, doctors or any other people interested in the system. In this paper, we propose a projection simulator that provides support to end-users for them to design spherical displays and to create their content. Then its implementation is introduced.

2. Development Flow of spherical screen systems

In order to consider the hazard of constructing the spherical display, we will refine the flow of development.

(1). Optical Design

In the first step, we have to consider the most efficient arrangement when projecting the image by using projectors pointing to the curved surface of the screen.

Figure 1 shows some examples of arrangement of the projector faced to single domed screen. There are several possibilities of projector's position/direction to cast the image on the spherical screen. The optical characteristics of projector are also required to examine the arrangement. Also, using a convex mirror to reflect the projection complicates the optical design.

Therefore, making a decision on how to arrange these components is very difficult because of their endless possible combinations. Thus, the simulation of the result of the projection is needed.

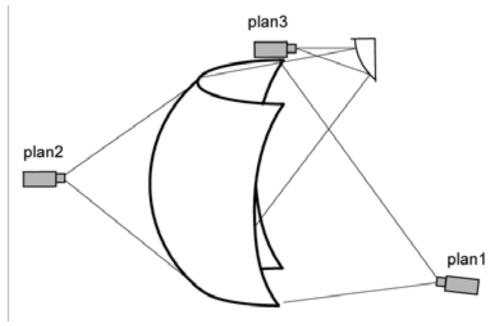

Figure 1: *Different examples of arrangement with a single projector. Plan1 is due to front projection. Plan2 applies rear projection and Plan3 uses a convex mirror for divergent reflection. There are numerous combinations to be examined and simulated.*

(2). Manufacturing/Assembling

Next we have to manufacture the prototype determined by the previous step. In general, the screen should be custom-made since the shape of it becomes expensive. The structure of geodesic dome helps to construct the spherical screen.

Molded acrylic spheres or Styrofoam screens are also available for individual use.

(3). Refine Alignments

It is necessary to detect and correct errors between the ideal design of the system and the actual prototype with practical projection. Image alignment and also some calibrations are required to correct these errors. The left image of Figure 2 shows the state of projection that includes misalignment due to the errors. The right image shows the ideal state of projection after some adjustments. There are some commercial/research pieces of program to do this alignment automatically by using external cameras.

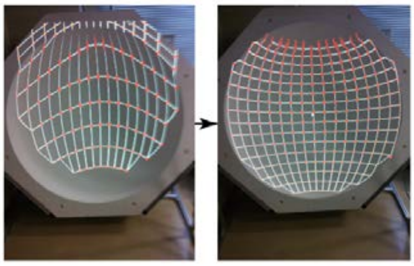

Figure 2: *Result of fine alignments for domed screen in the right image. Manufacturing error causes some misalignments as can be seen in the left image.*

(4). Creating contents with distortion correction

Finally we need to create contents that we want to display on a spherical screen. The final image of the content requires providing the geometry correction according to the result of (3). Some software to provide fish-eye warping is available. [BOU09] performs more advanced geometry correction by using Unity, known as a game engine.

One problem is that a gap between design and actual, as mentioned in (3), is unavoidable because its optical system is delicate. The gap will probably afflict end-users who are unfamiliar with spherical display. In this work we propose to use the result of the alignments to obtain feedback and take it into consideration when designing the spherical display in order to decrease the gap. This means that the feedback could be helpful to acquire the exact design build in (2). The advantage is that it can be adjusted while checking the state of the pixel distribution and distortion.

3. Implementation

The simulator consists of a ray tracing process unit, a calculation of distortion correction process and a rendering process of both results.

The ray tracing process computes the behavior of projector's rays that includes reflecting and intersecting. The focal spot size is also calculated to observe the state of focus on the screen. The number of target pixel for calculation is reduced, since the process can return the result immediately if the parameters of the optical elements are changed.

The distortion calculation and correction process is in charge of choosing the proper image to be returned by the projector attending to the results of ray tracing. Specifically, this process finds the combination of coordinates of the texture that is used to warp image in order to do the distortion correction. The process uses hill climbing to reduce the number of iterations to find the optimal combinations of all projector pixels.

Figure 3 shows an overview of simulations mentioned previously. Several parameters related to the configuration of screen and projectors can be changed by the upper right windows. The result of ray tracing process is reflected immediately on the left window. The result achieved by the distortion correction process is also rendered on the lower right window. The ray tracing process is completed within 3[msec] for each simulation and the distortion correction process need maximally 50[msec] to finish the calculation under a system equipped with an Intel CPU (Core i7-4710MQ, 16GB main memory). This means that the simulator is able to follow the results in almost real time and the user can verify the distribution of pixels in the design or can adjust the alignment at any time. Finally the simulator exports the result of distortion correction as an OBJ file for the sake of importing it to unity.

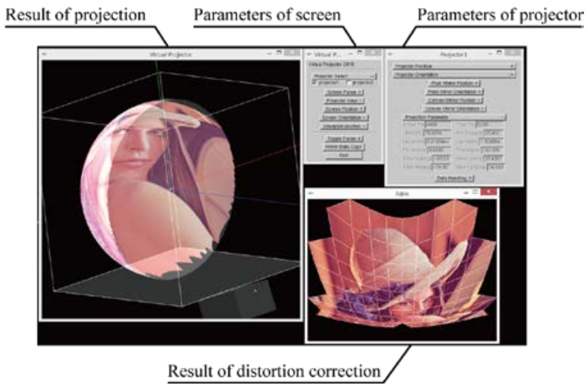

Figure 3: *Overview of the projection simulator. Left window indicates the result of projection of normal image onto spherical screen. Lower right window is the distortion corrected image. If the image is projected onto screen, the viewer can see it properly.*

4. Discussion

The current simulator corresponds only to parametric shapes. The future work aims to develop a simulator capable of supporting non-parametric shapes that captured by 3D depth camera. In addition, further improvements of computation time are expected when GPGPU is applied to this simulator.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Number 15K00290. We also appreciate Prof. Corchado and Dr. De la Prieta in University of Salamanca for assisting with this research.

References

[BOU09] BOURKE.P: Dome: Immersive gaming with the Unity game engine, Proc of CGAT09, pp 265-272, 2009