A Virtual Environment for 3D-Photography

VICTOR BOGADO, ASLA SÁ, LUIZ VELHO
IMPA - Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina, 110, Rio de Janeiro, Brasil

Abstract. This poster describes a virtual environment for 3D photography developed at the VISGRAF Lab.

1- Introduction

3D Photography is one of the "hot" areas of research due to its wide range of applicability and the diversity of problems it involves, unifying computer vision, image processing, and geometric modeling in the same framework. Here, we are interested in coded structured light techniques, which allow the use of off-the-shelf hardware, reducing the costs of the scanner.

Shape from structured light is based on triangulation principle. Measurement of depth values is carried out with a system that resembles a system for stereovision except that a projection unit is used instead of a second camera. Stereo vision has to overcome the problem of finding correspondent points between two different views. The structured light approach makes surface measurements by projecting patterns. Thus, it uses the illumination to mark points in the scene to help solving the correspondence problem.

In this work we show a 3D-photography virtual environment used to simulate the pair camera/projector. Although it was developed as a teaching tool, we have found many utilities of its use in research; one of the most promising is its capability of analyzing error of recovered points comparing it to the object itself used in rendering. It then becomes a powerful instrument in comparing different processing methods. It also can be used as a planning software to help constructing real structured-light acquisition set-ups.

This poster describes all steps of the virtual environment pipeline from positioning camera and projector to recovering and visualization of the data

2 -The Virtual Environment

The coded structured light 3D-photography pipeline consists of three main steps:

- Setting up the system: i.e., positioning camera, projector, object and choosing the patterns to be projected.
- Acquiring images (which in our virtual environment mean rendering synthetic images).
- Processing the obtained images in order to recover object's geometry and other attributes such as color and photometric properties.

The virtual environment is different from a real environment in the first two steps. For the third step it doesn't matter if the images were generated from a real or a virtual camera.

2.1 - Positioning Camera and Projector

When positioning camera / projector relative to the object we have two conflicting goals: to recover depth, an angle has to be formed between projector planes and camera rays. On the other hand it would be desirable to minimize projector shadows on the object from the camera point of view. We have implemented an interface considering the restriction that camera and projector are on the same plane. The area viewed by the camera is marked in yellow and the projection area in black. The working volume is the intersection of the two areas. Doing so we have reduced the problem of positioning camera and projector to a bi-dimensional problem, what simplifies user's interface.

2.2 - Coding and Decoding

The problem of coding structured light has been well studied in recent years. An overview of the problem can be found in [Sa02]. In our virtual ambient we have implemented three codes and their respective decoders: Gray Code, color Gray Code, and (b,s)-BCS. The user can choose the code or insert its own coded pattern together with the respective decoder.

2.3 - Rendering Images

To simulate the photographic process we use a 3D graphics pipeline named S3D. The object is described as a triangle mesh and the projector as a textured light source. To achieve a realistic result with shadows, we needed to use a ray-tracing algorithm. For efficiency, when a ray is traced the lighting is calculated simultaneously for all patterns.

2.4 - Generating and Visualizing Point Sets

An image-processing step is carried out to find the stripes transitions in the rendered images. Those points are decoded to match which projector stripe is correlated to each transition. By intersecting the camera ray that produced this pixel with the stripe projector plane we obtain the spatial position of this point. The set of points obtained is then visualized using a point-could viewer.

2.5 -Error Analysis

A simple error analysis was incorporated into the triangulation step. For each point of the cloud we can find the correspondent point in the mesh by simply re-casting the camera ray into the scene. The distance between those two points is used as an error measure. In future works a statistical analysis can be made to help comparison between different methods.

3 - Conclusion and Future Work

Our 3D photography virtual environment is a useful tool that models camera / projector hardware. It was designed to be open source and freely available. In the future, we plan to include simulation of calibration on the system. Sensor noise can be also incorporated. This would be useful to achieve more realistic error analysis.

References

[Sa02] A. Sá, E. Filho, P. Carvalho, L. Velho. Coded Light 3D-Photography: Overview. p.203-219,RITA 2002.