2D to 3D Image Visualizer

P.B.M.M.A. Bandara *and* S.D.H.S. Wickramarathna
Uva Wellassa University, Badulla, Sri Lanka

Introduction

The ordinary photographs are only considered about just an individual photograph itself. Since these photographs act individual character in a scene there is no way to visualize a connection between them. Also browsing of each captured photographs is not relative to other though these photographs are in the same scene. But there is a possible way to reorganize the relative position and orientation of ordinary photographs which are captured by a digital camera device. That is, reorganize the important features on each photographs using feature detection algorithm, then match each photograph by matching these features, then estimate the relative camera pose against each photograph in order to calculate relative replacement and orientation of each photographs on 3D environment. It is useless to calculate the 3D replacement without visualizing them, because the real world scenes are in 3D environment. Therefore, 3D visualizer, to visualize all calculated photographs is developed with very user friendly and interactive way. This project is to develop a system for interactively browsing and exploring unstructured collections of photographs of a scene using a three-dimension interface.

Materials and Methodology

The image processing techniques for feature extraction, feature matching techniques, structure and motion (SaM) estimation techniques and 3D geometry techniques for estimate the relative 3D motion of each photograph, structure from motion (SfM) techniques to minimize the re-projection error and finally 3D drawing techniques to visualize the results are used to complete this project. Following materials were used to implement the complete system.

- C++ programming language
- OpenCV image processing library[3]
- Bundler structure from motion system[6]
- OpenGL 3D graphics API[5]
- Gtkmm graphical user interface API[2, 3]

Feature extractions were done using *Scale-invariant feature transform* (SIFT) feature extraction algorithm. Then feature that extracted were matched using *K*-nearestneighbor (KNN) feature matching algorithm against each and every photograph. Once the extraction and matching were done, bundle adjustment is involved to estimate the relative camera pose [1] and minimize the re-projection error in order to create optimized 3D environment to visualize the photographs with user friendly and interactive manner to users. Finally the 2D to 3D image visualizer system is accepted a collection of unstructured 2D photographs and produces a 3D environment by considering relative 3D position and orientation of the photographs.

When creating a user friendly and user interactive user interface, many 3D geometrical and GUI creating techniques and algorithms were involved such as,

- 3D transformation using transformation matrices with texture mapping to a 3D rectangle by extracting image file's binary data and providing those data to OpenGL.[5]

- Images blur to enable more user interactive effects.
- Image transparency effect with OpenGL.
- First person camera movement by capturing mouse movement and key board inputs and transform the camera with camera axis to enable user to explore the photographs in interactive and meaningful manner.
- Easily navigate to a specific photograph in the 3D environment by implementing 3D ray intersection algorithm to identify which object on the 3D scene was user clicked.
- Provide an easy and user friendly Graphical User Interface to hide the complexity of the overall process from the user by creating an user friendly GUI with Gtkmm GUI components and specially OpenGL capable drawing are to draw OpenGL graphics on a Gtkmm Drawing Area GUI component without move to a different OpenGL window.[5, 2]

Results and Discussion

The resultant overall system can be used to create a 3D environment from given 2D photographs with the following key features.

- The system accepts the 2D photographs to make it 3D.
- The system will calibrate the camera which is the photographs taken from. Therefore this system can be used to calibrate a camera with the OpenCV chess board method for computer vision applications. Therefore the produced camera intrinsic matrix can be used for any other computer vision, 3D reconstruction or stereo vision related computer programs.
- Extract feature points of each photograph. This involved SIFT key point detection algorithm which is one of the fastest and more accurate method used to extract key points on the image.
- Match these extracted key points using KNN feature matching algorithm also a one of the fastest method to match the key points. Therefore it takes only a few seconds to create match between each and every image.
- Produce a file with all parameters to re-create 3D environment using calibrated camera parameters and extracted key features.
- User does not want to re-run all the above steps for the same collection of photographs, if the photographs and the file which is already created with all parameters are exists.
- The ability to store camera calibration information for further usage is also included in the system to enable more user friendly manner with fewer steps to produce the 3D visualization.

Conclusion

The system is capable to create a three-dimension interface to interactively browsing and exploring unstructured collections of photographs of a scene by accepting 2D unstructured collections of photographs which are captured from a digital camera device. But there are some limitations on the system,

- The orientation should be landscape for all images.

- The system cannot produce the 3D visualize without the focal length (pixel) of images. But still user can follow calibration steps on the system to calculate the focal length. It is impossible to create 3D visualize, if user cannot calibrate the camera using the system also. For example if user has the image files (without focal length information) only without camera device, user cannot take photographs of the chess board to calibrate it.
- Unavailability of sufficient key points may lead to produce an identity matrix for camera transformation matrix. Therefore that image is invalid to visualize it on 3D space.

References

Ma Y., S. Soatto, J. Kosecka and S. Sastry, An Introduction to 3D Vision, (MASKS).

Gtkmm Documentation which is supplied with Gtkmm-dev package.

ANON., http://www.gtk.org/documentation.php

ANON., http://docs.opencv.org/

ANON., http://www.opengl.org/documentation/

ANON., http://phototour.cs.washington.edu/bundler/