Sparse 3D Model Reconstruction from photographs

Kanchalai Suveepattananont
ksuvee@seas.upenn.edu
University of Pennsylvania

Professor Norm Badler
badler@seas.upenn.edu
University of Pennsylvania

Advisor Aline Normoyle
alinen@seas.upenn.edu
University of Pennsylvania

Abstract

Nowaday, millions photos and videos are being shared and displayed on the internet everyday. Large photograph collections of buildings, statues and various tourist- attraction sites can be found easily on the internet. Ones no longer need to leave computers to "visit" the Notre Dame Cathedral in Paris or the Great Wall in China and many other sites in the world. Simply searching via large collections of photo on Google or Flickr, one can see all these places in large-scale and in details. However, images of these large constructions still lack of visualization technology to give users immerse experiences in browsing through them. Therefore to effectively and efficiently utilize this visual information, more and more efforts are put into developing visualization and user-interfaces tools to help users browsing through it.

In this project, I propose a system for interactively browsing and exploring photographs of sites using 3D-interface. The system will provide interactive users-interfaces to navigate the 3D point cloud. Therefore, the software will enable users to visualize sites through sparse, 3D reconstructed scene giving users more intuitive and creative ways to browse and view their photographs.

Project Blog: cis-497-3dreconstruction.blogspot.com

---

1 Introduction

Large collections of photograph of various tourists sites are available on the internet, giving users' experience to visit these sites. However, users still browse these large collections in traditional methods using grid-based layout, making the experiences less immersive. Ones can see minute details of sites through these collections of photographs; one can also easily find images of places in large scale and see overall pictures of them. However, there is a large discontinuity between these two categories of photographs, making it impossible for users to visualize where all details this photography located on sites and how sites interact with their intricate details. This, then, makes user's experiences less immersive and effective.

I propose a system to enable users to experience sites in exploring photographs through interactive user-interfaces and sparse 3D scene visualization. The system will allow users to explore sets of photographs to visualize details collected by others people while at the same time able to visualize sites in large-scale and see how these intricate details fit with in overall scale of these sites. This system will provide transformation between 3D navigation and photographs exploration.

The project makes following contributions:

- I have learned various computer vision techniques particularly in the field of object recognitions, 3-dimensional model reconstructions from images, and image processing.
- I built a 3D reconstruction pipeline in which each component of the pipeline independently perform computer vision tasks such as feature detecting, feature matching etc. This allows each component to be easily replaced by any future development.
- I have developed a system that construct sparse 3D scene composing of points clouds and the method to visualize the model. The models will be constructed from photographs collections of sites taken from calibrated camera.

1.1 Design Goals

Main audience of the project is everyday computer users who would like to have immersive experience in browsing photographs of various places and being able to feel as if they visit these sites physically. Users will be able to navigation and transition between sparse 3D model and 2D photographs.

Initialy the project is designed to closely follow [Snavely et al. 2006] paper and with goals to extend its interface, rendering, and visualization. However, due to technical and time constriant, the project significantly simplify from the original problems. The project attempts to make initial, simple start which is to reconstruct 3D model from two images from calibrated camera. Although I will, in the future, develop the system which allows multiple photograph from uncalibrated camera to be used, by developing the simpler version like this project, I will have basic framework which will allowme to extend it further as well as learn about computer vision which will significantly help me in developing the framework.

One of the project's goals is to create a 3-dimensions model reconstruction pipeline in which each component independently perform its task; thus, this allows each part of the pipeline to be separatedly tested and replaced by different, better methods in the future. Secondly, the project's goal is to develope a visualization model that provide interactive interface in browsing and exploring 3D reconstructed point cloud; the project aims to provide architectural, historical, and scenic details of various sites.Thirdly, the project serves as both starting point and spring board to learn and be exposed to topics and application in both computer graphic and computer vision.

1.2 Project Proposed Features and Functionality

- Developed a 3-dimensional model reconstruction pipeline which can be generalized and easily extensible. The pipleine is created such that each component can be tested, extended,

and replaced independently from each other. The goal is that to enalbe future development and extensions if needed

- Reconstructed sparse 3D model from two photographs obtained from the calibreated cameras.
- Created interactive, intuitive user-interace which allows users to explore the point cloud in 3-dimensions.
- Visualized 3D point cloud using basis, interactive and intuitive user-interface which allows users to explore the model in the 3-dimensions.

2 Related Work

Works in the area relied upon performing image processing, image analysis, rendering and visualization. They also relied upon advancement in technology to identify location, date, and many metadata information of photographs. All of these are to be used to answer questions such as where images are taken (in the sense of GPS location and points of views), what cameras look at, what relationship between each image etc.

There are various work to organize large collection of photographs. For example, [Aris et al. 2004], which exploits location and time information of photographs to organize and create storytelling. Another similar works by [Naaman et al. 2004] also proposes an algorithm to automatically organize digital photograph based on geographic coordinate and time when photographs are taken to provide an intuitive way which mimics how people perceive their photo collections. Earlier in 2003 [Naaman et al. 2003] also proposes sharing geographic metadata information of digital photographs to facilitate such organization.

In addition to organize sets of images based on geolocation and time, works have also been done in visualizing the collections in order to mimic real physical and geographical locations where images are taken. For instance, in 1995, [McMillan and Bishop 1995] proposes image-based rendering by sampling, reconstructing the plenoptic function to generate surface based on cylindrical projection. There are also various late works which similar but focus more on 3D applications. [Kadobayashi and Tanaka 2005], develops a method using 3D viewpoints to search and organize sets of photographs, as well as browse groups of images from similar points of views. Another one is [McCurdy and Griswold 2005] proposes a system to reconstruct architecture from video footages.

In 2006, [Snavely et al. 2006] propose another visualization and organization of photograph by creating sparse 3D scene constructed from collection of photograph from different points of view of the sites. As the project is mainly based on [Naaman et al. 2004], some related technical details mentioned in the papers are included.

Through out the development of the project, I have learned various concepts of computer vision both from online resources and books. Resources which I particularly found it to be very helpful for my self-taught learning experience are : Al Shack's computer vision blog, Jason Clemons's lecture slide on *Scale Invariant Feature Transform by David Lowe*, Richard Szeliski's Introduction to Computer Vision slideon Structure from Motion, and Mastering OpenCV with Practical Computer Vision Projects by Daniel Lelis Baggio et al.

2.1 Keypoints feature and Image Matching

As [Snavely et al. 2006] proposes methods to reconstructions of sparse 3D scene, the first step in organizing and visualizing these images collections are to find feature points in each image. LOWE proposed an approach to perform this task of keypoint detector and localization, Scale Invariant Feature Transform (SIFT). However, to find similarity between images, [Snavely et al. 2006], proposes by [Arya et al. 1998], [Arya et al. 1998] develops algorithm to robustly approximate nearest neighbors. This general algorithm can then be applied in more specific context of the problem.

2.2 Bundle Adjustment

In order to reconstruction 3D model, [Snavely et al. 2006] also suggested using a software package developed [Lourakis and Argyros], which based on using levenberg-marquardt algorithm [Nocedal and Wright 1999], to perform bundle adjustment which is common process in 3D reconstruction to refine optical characteristic of camera and 3D coordinates of points.

3 Project Proposal

The objective of the project is to provide a method to visualize photographs (currently support two photographs), create well organized, easily extensible reconstruction pipeline, and provide users with an intuitive interface to explore the model in 3-dimensions. I propose a system which reconstructs sparse 3D models from photographs taken by calibrated camera and visualize of the 3D models using point cloud structure. Through out the project, I hope to gain knowledge regarding computer vision and learn about the field through experimenting with various different methods.

3.1 Anticipated Approach

3.1.1 Keypoints Selection and Feature Matching

The first step to utilize sets of images is to detect key features in each image. The process of doing so is well studied as mentioned in 2.1. In order to perform robust keypoint detection which is invariant to image transformations, I will utilize existed SIFT package which will provides both locators and descriptors of keypoints. Once keypoints are selected, the next step is to find similarity between keypoint descriptors using approximate nearest neighbors as mentioned in 2.1.

3.1.2 Structure of Motions and Bundle Adjustment

Because the project utilizes images from calibrated cameras, all cameras' parameters, such as focal length, image format, and principal point, can be gained from calibration process. With all these data, I am able to utilize them to reconstruct sparse 3D model, Structure of Motions (SfM), by triangulating points between two

images. IF time permits, The process is an iterative process of continuously adding camera information to perform optimization. The core part of the SfM, performing Bundle Adjustment, will be done using the Sparse Bundle Adjustment Library as mention in **??**.

3.1.3 Rendering and Visualization

Main rendering technique of the project is to visualize the sparse 3D model as points clouds. If applicable and time permitting, I will develope non-photorealistic rendering style of watercolor. In addition to visualize the model,I will also organize collections of photographs using geographic locations and estimated, relative camera positions resulted from SfM.

3.1.4 User-Interfaces

I will provide both 3D navigation of the model which is a standard motion controls to allow users to explore the model in the 3-dimensional space. If time permits, I would also like to implement seamless transformation between 3D control and viewing of photographs. I will attempt to provide intuitive, effective, and user-friendly ways to visualize photographs and their relationship with 3D representation of the sites. If time permitting, it will also be in my interest to employ similar navigation techniques used by [Snavely et al. 2006] in moving between two photographs and their related views.

3.2 Target Platforms

I will mainly develop the project in C++ and use various libraries to perform certain tasks: ANN library for Approximate Neares Neighbor Searching, SIFT++ for Scale-Invariant Feature Transform detector and descriptor, jHead for reading EXIF of JPEG, Sparse Bundle Adjustment library for bundle adjustment, as well as refering to SfM libraries and frameworks. I also will use Eigen library for matrix computation if needed.

3.3 Evaluation Criteria

To evaluate the project, I will compare the sparse 3D model in relative to its original sites and whether it successfully provide accurate visualization (i.e users can identify the sites). I will compare to existed works of [Snavely et al. 2006] and see how close and accurate my pplication. Another criteria is to use database of photographs and compare them to similar areas of the model and observe how closely they link together.

Another evaluation method is to use test images and camera paramter which is avialable online as input then compare the result 3D model to alread existed model.

4 Research Timeline

Below are my project timeline. I also have included a Gantt chart in the final page with more detail.

4.1 Project Milestone Report (Alpha Version)

- Complete all background reading
- Collect all necessary images
- Developed software framework is functioning with simple base case such taking data set of images and performing keypoint detection
- Built keypoint detection and matching

4.2 Project Milestone Report (Beta Version)

- Built basic but functional SfM procedure although more tweeking may still require
- Devloped rendering system which will aid reconstruction process
- Built standard 3D navigation tools

4.3 Project Final Deliverables

- Complete sparse 3D recontruction from collections of photographs
- Video to demonstrate users interact with the application
- Complete rendering of sparse 3D model in point clouds
- Compete final report

4.4 Project Future Tasks

If I have extra 6 months to work on the project, I would like to add following features

- 2D Navigation of views related to each others
- Non-photorealistic rendering in style of watercolor
- Providing options for quering database of image from social media
- Develop alternative organization models of images collection such as using time
- Provide functionality which users can manually select features on their images and see how such features interact with reconstruted model

5 Method

The project is designed with the pipeline (figure 2) in mind. The pipeline is composed of five separate different components which can be independently architected, implemented, and tested. The pipeline allows also each component to be easily replaced by new implementation in the future. In this section, I will break down each component of the pipeline and summerize both technical and implementation details of each piece.

5.1 Camera Calibration

5.2 Feature Selection

The feature in computer vision is a piece of information which is used to solve computer vision problem. The task of this component of the pipeline is to find features in the given collections of images. To better understand the output of the process, I will briefly describe the algorithm focusing on David Lowe's Scale Invaraint Feature Transform. Firstly, the process generates the scale space of input images; one can view the scale space as input images undergoing various Gaussian smoothing which is to emphasize the local variation , and local minima. Secondly, the process will compute Difference of Gaussian (DOG) for each pair of image in different octave. Thirdly, the algorithm will compute location of potential features, detect edge, and filter low contrast points. Forthly, it will assign keypoints orientation and build keypoint descriptors. The keypoint descritptors will be used in the next part of the pipeline, feature matching. Keypoint descriptors should not be confused with keypoint location. Keypoint descriptors encode a unique finger print of the feature; it represents normalized gradient magnitudes and orientations around the keypoint location by constructing a 16 x 16 window around the keypoint which will be broken into 16 of 4x4 windows (figure 3a). In each of this 4x4 window, gradient magnitudes and orientations are calculated and store in histogram of 8 bins. There the total storage of feature descriptor is a vector of 128 elements. The process of storing information into histogram bin is done by utilizing Gaussian weight function shown in blue circle of (figures 3b). The kyepoing, on the other hand, is simply a cirresponding location of a feature descriptor in an image.

Using OpenCV library, the pipeline currently supports two feature selection techniques: Scale Invaraint Feature Transform (SIFT) by David Lowe and Speeded-Up Robust Features (SURF). There is slightly difference in performace in speed. SURF is slightly faster than SIFT. However, results of both algorithm look very similar. At the end, I decided to use SURF going forward. Nevertheless,switching to SIFT is simply a commenting/uncommenting of C++. pound define flag. Sample images of the algorithm are shown in figure **??**

5.3 Feature Matching

Feature matching is a next step in the process. The idea behind feature matching is to match keypoint descriptors between a pair of image. The pipeline provides two different options in performing feature matching. The first one is simply using OpenCV library to do featuer matching. Another method is to use Approximate Nearest Neighbor (ANN) algorithm. The ANN algorithm requires the data structure to be converted from standard C++ vector to ANN point array. The algorithm, then, creates KD-tree using keypoint descriptors vector of size 128. In addition, I also implemented threshold cutoff to filter mismatching features. An example of feature matching using ANN algorithm is shown in 5a. The results are quite accurate. However, because the pipeline currently only supports two image reconstructiion, I decided to use OpenCV feature matching whose results are more dense than ANN (as shown in 5b). The ANN results will be great to use if more than two images are used for reconstruction.

5.4 Fundamental Matrix Computation

Fundamental matrix (F-Matrix) is a 3x3 matrix encoding relationship beween 3D points and locations in 2D Image. One can considered F-matrix as transformation matrix between image A and image B. The process to compute such matrix is non-trivial in uncalibrated camera case. In the uncalibrated camera scenario, computing fundamental matrix requires Random Sample Consensus (RANSAC) algorithm which is an interative, optimization process which attempts to fit line into data points which are considered as inliners. It will also require the pipeline to figure out focal length from each image before being able to compute the fundamental matrix. Using calibrated camera (currently used in the project) significantly simplifies the problem. We can compute instrinsic and extrinsic camera parameters. Then, we can simply utilize OpenCV's function to retrieve back fundmental matrix which will then allows the program to compute essential matrix. The essential matrix is similar to fundamental matrix in that both encode a mapping constraint between two image; however, the essential matrix can be used in reconstruction of the whole collection of images taken by the calibrated camera. This is not true in the case of fundamental matrix which can only be apply between a pair of image. Therefore, with large collection of uncalibrated cameras' images, solving the fundmental matrix requires interative process and and optimization to minimize errors. Once the program is able to compute essentail matrix, it can then use a factorization process, Singular Value Decomposition (SVD) to solve for a camera matrix which maps between 2-dimensional point of images and 3-dimensional of the model. The other mapping matrix is encoded as fixed and canonical camera matrix, no rotation and translation.

5.5 3D Reconstruction Using Triangulation

The reconstruction of 3D model can be done using linear triangulation process by solving the following equation $x = PX$ where x is a 2D point on an image and X is a 3D point of the scene. Then, by solving the linera equation system of the form $AX = B$ the program will be able to retrieve back the 3D points. The program, then, stores the result in form of standard C++ vector.

5.6 Visualization

The pipeline utilize Point Cloud Library(PCL) for rendering and visualization. As part of the process, the pipeline must convert data structure from standard vector C++ , which stores point struct-the struct contains 3-dimenional location, rgb color vector, and a set of indexs of images which the point is derived from-, to PCL internal point cloud data structure, PointXYZRGB. (see figure **??**) The following link is the video showing 3D navigation : http://youtu.be/zAu-6P5ltlc

6 Results

The program is currently able to reconstruct a 3-dimensional point cloud from two images taken by calibrated camera. It also provides

simple, intuitive 3-dimensions mouse navigation for users to explore the model in 3D space. Because input images must be taken by calibrated cameras, the program also supports calibrated camera component which has to only be be ran once by provide a list of images to be used for calibration in the form of .xml text file. In general, the software at each stage of the pipeline will output results such as detected features overlaying on original images or matched features overlaying on original images. The program currently only takes input in the form of a folder containing images.

7 Conclusions

Through out the project, I have learned mach new knowledge about the field of compute vision. I have an opportunity to learn and familiarize myself with mathematical concepts related to computer vision particularl in the field of optimization, geometry, and linear algebra. In the end, I am able to develope a 3D reconstruction software and pipeline which constructs 3D model from two images taken by calibrated cameras. Although the project is very limited, it is a great starting point of my journey in learning computer vision. The project certainly allows me to learn about the field of computer vision in depth and gain valuable mathematical and analytical skills and experiences.

8 Future Work

- Expand types of input images to include collections of images from uncalibrated camera
- Improve multiple views reconstruction to be more robust and efficient by implement Bundle Adjustment which is an optimization process to minimize least-square errors
- Create additional piece into the pipeline which supports photographs from the Internet
- Employ rendering technique such non-photo realistic to produce better visualization
- Implement user-friendly interface which allows users to upload input images as well as download the 3D reconstructed model. This will hopefully allow users to create a customized system to support what they need
- Create better user interfaces which allows users to browse between 2-dimensions images and 3-dimensions model
- Optimize reconstructed point cloud to reduce errors such as minimizing least square

References

ARIS, A., GEMMELL, J., AND LUEDER, R. 2004. Exploiting location and time for photo search and storytelling. Tech. Rep. MSR-TR-2004-102, Microsoft Research, One Microsoft Way, Redmond, WA, 98052.

ARYA, S., MOUNT, D. M., NETANYAHU, N. S., SILVERMAN, R., AND WU, A. Y. 1998. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *J. ACM 45*, 6 (Nov.), 891–923.

KADOBAYASHI, R., AND TANAKA, K. 2005. 3d viewpoint-based photo search and information browsing. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, 621–622.

LOURAKIS, M., AND ARGYROS, A. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Institution of Computer Science-FORTH. Available from www.ics.forth.gr/ lourakis/sba.

MCCURDY, N. J., AND GRISWOLD, W. G. 2005. A systems architecture for ubiquitous video. In *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, ACM, New York, NY, USA, MobiSys '05, 1–14.

MCMILLAN, L., AND BISHOP, G. 1995. Plenoptic modeling: an image-based rendering system. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '95, 39–46.

NAAMAN, M., PAEPCKE, A., AND GARCIA-MOLING, H. 2003. From wher to what: Metadata sharing for digital photogrphs with geographic coordinates. In *Proceedings of the Int. Conference on Cooperative Information Systems*, 196–217.

NAAMAN, M., SONG, Y. J., PAEPCKE, A., AND GARCIA-MOLINA, H. 2004. Automatic organization for digital photographs with geographic coordinates. In *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, ACM, New York, NY, USA, JCDL '04, 53–62.

NOCEDAL, J., AND WRIGHT, S. J. 1999. Numerical optimization. *Springer Series in Operations Research*.

SNAVELY, N., SEITZ, S. M., AND SZELISKI, R. 2006. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph. 25*, 3 (July), 835–846.

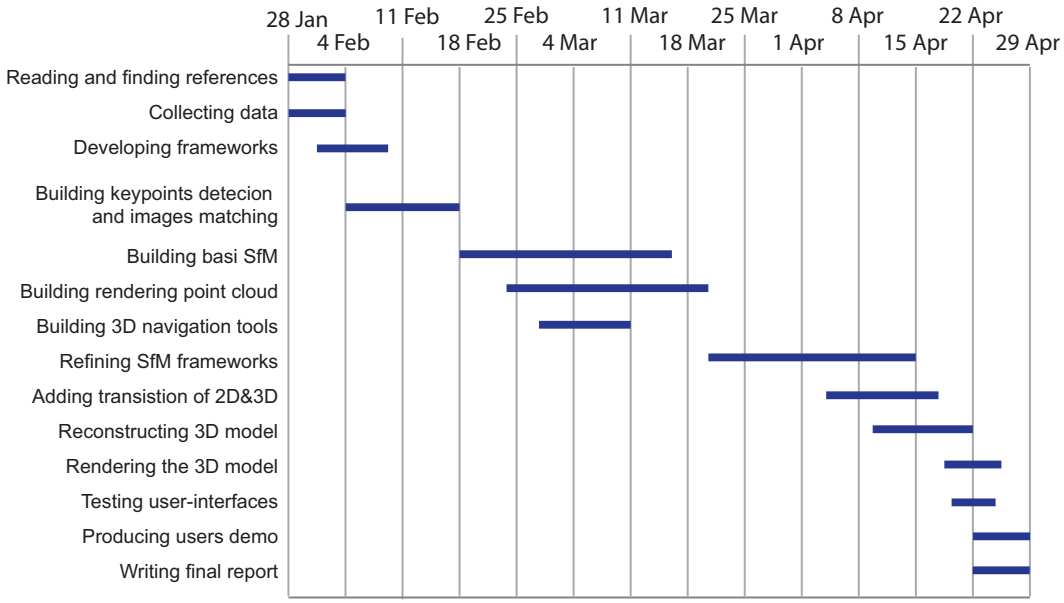


Figure 1: *Ghantt Chart.*

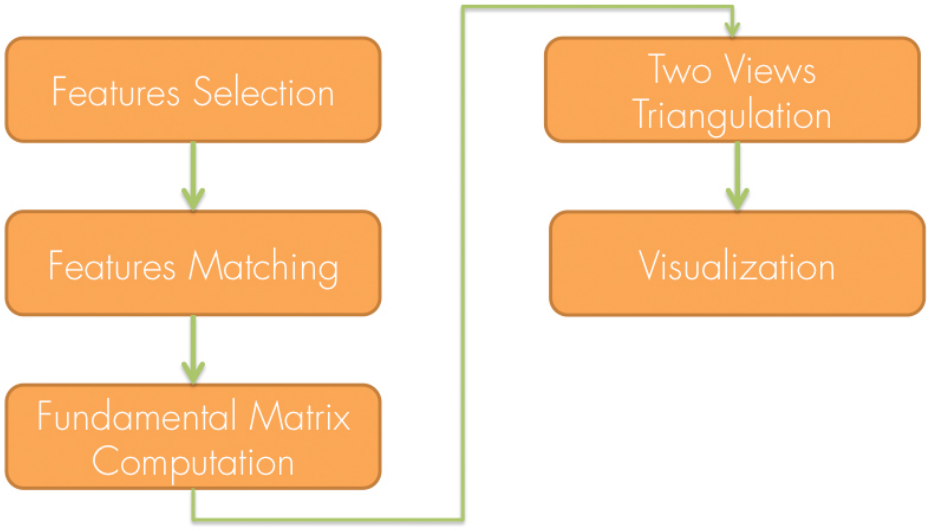
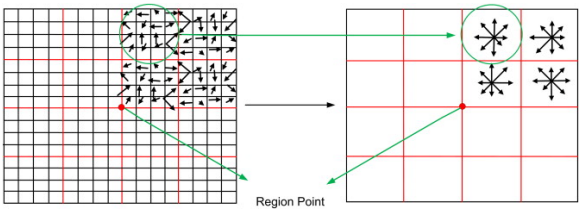
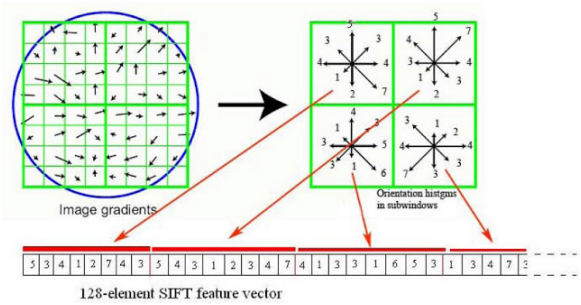


Figure 2: *Pipeline Diagram*

Figure 3: *Feature Detection*



(a) *Feature Descriptor 4x4 Windows*



(b) *Feature Descriptor Histogram Bin*

Figure 4: *Sample Feature Detection Using SURF*

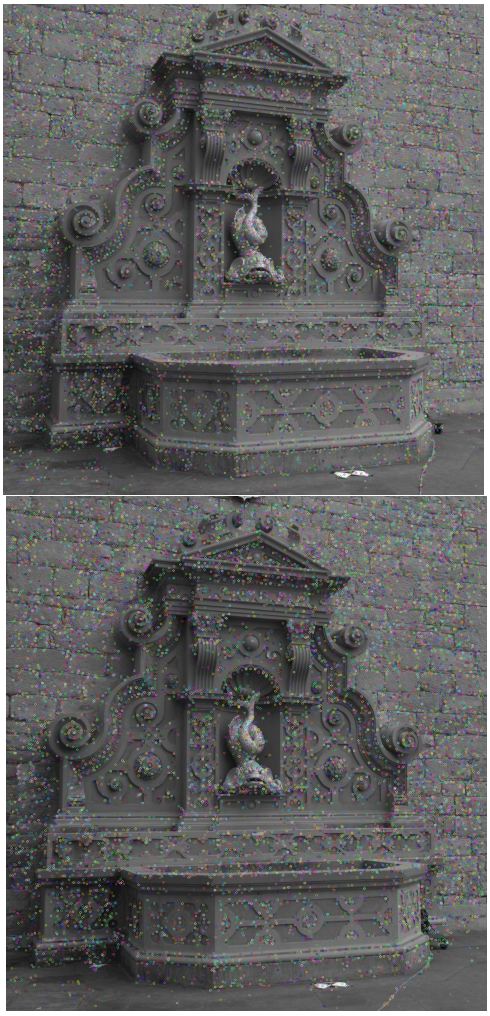
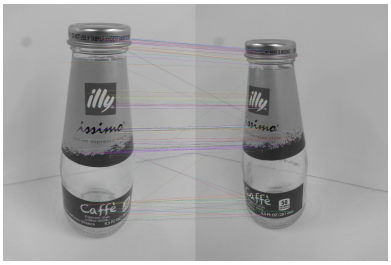
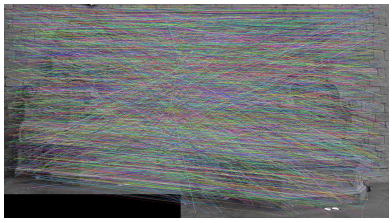


Figure 5: *Sample Feature Detection Using SURF*



(a) *ANN Feature Matching*



(b) *OpenCV Feature Matching*

Figure 6: *Result*

(a) *Original Input Images*

(b) *Screen Shot of the Model*

(c) *Screen Shot of the Model*