CS294-26 Final Project: Quadcopter Building Reconstruction

Mark Jouppi
UC Berkeley
mark.jouppi@berkeley.edu

Sunil Srinivasan
UC Berkeley
sunil.srinivasan@berkeley.edu

Abstract

For our CS294-26 final project, we present a system in which a quadcopter acquires imagery of buildings, which we then show how to use to generate simple textured 3D models. To evaluate our method, we show and discuss results for several buildings on the UC Berkeley campus. Finally we discuss advantages and drawbacks of this approach as well as future work.

1. Introduction

In recent years, small electronics have seen significant price decreases as companies mass produce smart phones and other compact devices. As a result of this, the robotics community has enjoyed a sharp decline in the cost of many parts as well as improved quality for sensors, processors, and to some extent actuators. Today, it is possible to purchase a moderately sized quadcopter complete with an HD camera, inertial measurement unit (IMU), and onboard processing comparable to that of a smartphone for just a few hundred dollars, making it affordable to many students, casual hobbyists, and "hackers". This opens up many interesting new opportunities for experimentation and research. In this project, we utilize one such inexpensive quadcopter, the Parrot AR Drone (see Figure 1).

While photography is great for capturing realism, it is hard to manipulate to construct new views. On the other hand, traditional graphics approaches are great for manipulation but require significant expertise and work to be realistic. The aim of computational photography is to capture both ease of manipulation and realism [1].

In this project, we create a system in which a quadcopter images buildings, and we use the imagery to generate 3D textured building models. The goal is that these resulting textured models will be both easy to manipulate and look reasonably realistic as they are derived from a large set of real photos.

Using a quadcopter gives us a unique capability to image building facades at heights and angles not available to a ground photographer. If a man on the ground takes photos of buildings, they will generally be in the perspective of looking up at the building. These views could be warped using a homography to appear as if they were taken front-on, that is, as if the camera's optical axis was perpendicular to the plane of the façade. However, warping can look aesthetically poor especially if significant warping must occur. Furthermore, important building details not visible from a ground perspective would be lost with warping (e.g., if the building had inset ornate windows that can't be seen well from the ground looking up).

Figure 1: Parrot AR Drone Quadcopter used for this project

A quadcopter allows us to capture what is approximately a scaled orthographic projection of the building façade. That is, by imaging the building façade looking straight at it, we essentially capture weak perspective images of the façade. This means the image projection reduces to simple uniform scaling. Moreover, since we can capture many such images with the quadcopter at various heights and positions, we can make sure to capture the building in high detail that would not be visible from a single perspective.

Using these textured 3D model results produced by our system, people can look at captured buildings in high resolution at various novel viewpoints which are easy to manipulate.

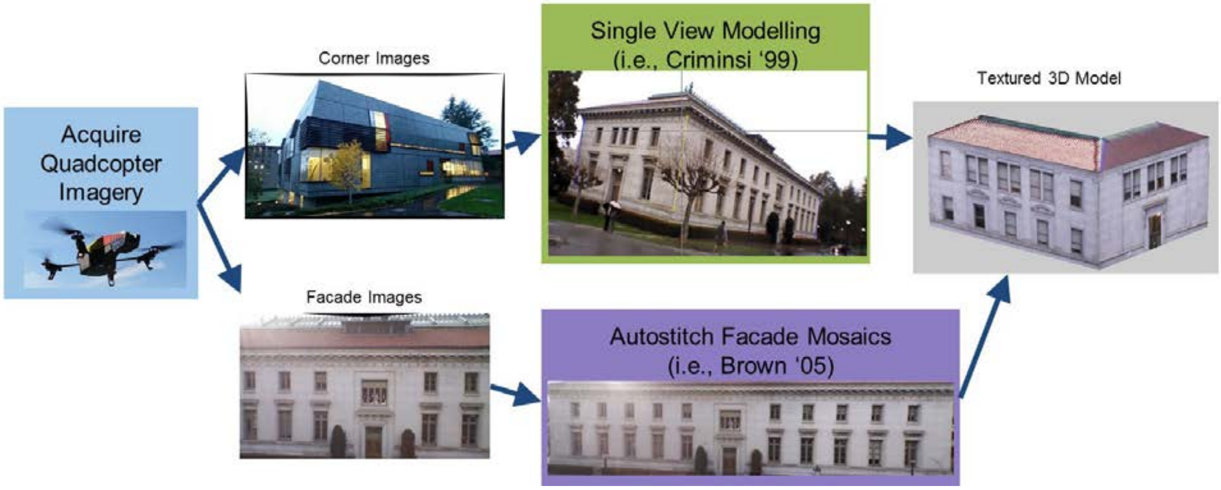


Figure 2: Overview of our system

2. Methods

2.1. Overview

Our overall system is depicted in Figure 2. Our system begins with quadcopter image acquisition, in which we capture two kinds of photos. The first type of photo we refer to as "corner images". These pictures show multiple facades of a building at once. We use these corner images for single view modeling to determine building dimensions with a bit of user input. This tells us what sizes to make our 3D model.

The second type of image we capture is façade images. These constitute the majority of the imagery we collect. Façade images are used to create a high resolution mosaic of the façade, which we use to texture our 3D model.

Rather than using the raw image data directly in our single view modeling computation and mosaic generation, we must first perform preprocessing to correct the significant radial distortion.

2.2. Image acquisition

Of the two types of images we take, corner images are relatively simple. We merely take a picture showing multiple facades in one view to use in single view modeling.

For façade images, as described in the Introduction, we generally image building facades with the quadcopter at multiple positions, looking straight ahead at the façade. This captures what is approximately a scaled orthographic projection of the façade. Note that we can move the quadcopter, changing the camera center position and still have a valid homography to relate images because we assume the façade is a planar surface. In situations where it is not feasible to image the façade of a building as

(a) original calibration target (b) corrected image

(a) original building façade (b) corrected image

Figure 3: Lens distortion correction

described above (e.g., if a tree is blocking part of the area in front of a building so we can't fly in front of that part), we resort to flying the quadcopter to a height that is at about half way up the building's height and staying at a single position. We then spin the quadcopter to get views of the building at different angles. These views are warped using homographies. This approach suffers from the drawback of missing details not visible from that position, but sometimes this is necessary due to flight obstacles.

Quadcopter image acquisition must be performed in favorable flight conditions with weak winds and lack of inclement weather to maintain the safety of the vehicle.

In terms of lighting, we found that overcast lighting conditions were optimal for our purposes. This is because in overcast weather, the clouds act to diffuse sunlight, producing a relatively uniform lighting with no strong shadows on building facades. See the Durant Hall result for an example taken under these conditions. We also tried imaging at the photography "golden hour", about an hour before sunset, on day with relatively clear skies, but it didn't look as good due to harsh shadows and significant lighting differences between facades that were front lit and back lit. However, one advantage was that the colors appeared more vibrant whereas the overcast images tended to appear more washed out. See the California Hall result for an example showing this.

2.3. Lens distortion correction

The quadcopter has a 92 degree wide-angle lens. The wide-angle lens captures more of the scene in photos compared to a normal lens, which also serves to help the pilot be more aware of the quadcopter's surroundings during flight. However, the wide-angle lens results in significant radial and tangential distortion of the scenes for which images were taken. In this case, the images experienced "barrel distortion" or a fisheye effect. Using the method described by Zhang [4] and implemented in OpenCV, we can obtain the five distortion parameters for the radial and tangential distortion of the camera. To accomplish this, we acquired imagery of a chessboard pattern at various locations and orientations (relative to the camera) and used the OpenCV implementation of the method to retrieve both a set of distortion coefficients and a camera matrix containing the horizontal and vertical focal lengths and optical center of the camera. We then used the OpenCV implementation for undistorting images to rectify our images according to these parameters. See Figure 3 for examples of this.

2.4. Single view modeling

Our implementation starts with the assumption of the building's basic model: that of a rectangular prism (a box) with unknown dimensions. On the "corner images" described previously, we manually designate the line segments corresponding to the three visible vertical edges of the box corresponding to the building. Because we have now defined a set of points on the ground plane and on a parallel plane above the ground plane, we have the option of using the method detailed by Criminisi et al. [2] to determine the world coordinates of the corners of the building. However, we make several key assumptions that allow us to simplify our method to avoid explicit use of vanishing lines or world reference coordinates. First, we assume that we do not need to retrieve the actual world positions or dimensions of our box model, but rather we only require that the ratio between the dimensions is correct in order to generate an aesthetically pleasing reconstruction. Second, we assume that the world Z positions of the top of all three edges are all the same (and that the three edges are parallel in both the world and the image). Third, we assume that the camera focal length and the camera distance from the building are approximately the same. Fourth, for buildings that are not rectangular

Figure 4: Single view modeling user interface

prisms, we restrict our facade planes to rectangular regions that crop any parts of the building that to not conform to the rectangular prism model (for an example of this, see Figure 5c). These assumptions allow us to view our image as a sort of scaled orthographic projection, which allows us to retrieve the approximate length of the building using the ratio between the lengths of the projections of the designated edges. As in Figure 4, we take a as the distance between two designated edges in the image, v as the length of the edge further from the camera in the image, and x as the length of the edge closer to the camera in the image. We first establish that the height of our box is the same as x, and calculate the distance d of the physical length of the facade as:

$$d = ax(v^{-1})$$

We repeat this for the other facade visible in the corner image, so that we now have the height, width, and length of the box for our model, where the actual values may be arbitrary but where the ratios between the three dimensions are approximately correct. This is similar to use of a triangle similarity in conjunction with the distance from a vanishing point, but our assumptions about the focal length allow us to disregard it while still obtaining a reasonable approximation.

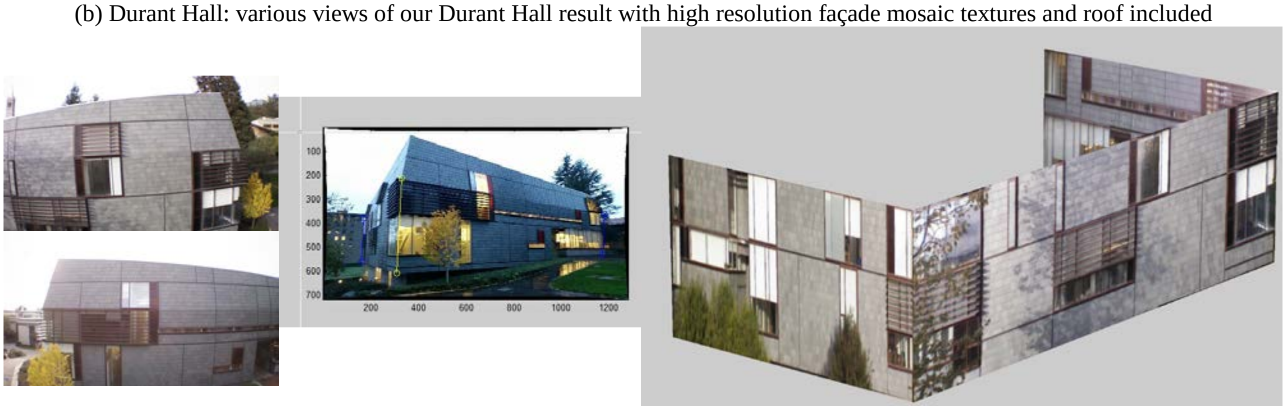
2.5. Autostitched façade mosaics

The various façade images are automatically stitched together using a MOPs based approach which we briefly outline below [3]. For each image, we detect Harris corners and use Adaptive Non-Maximal Suppression (ANMS) to produce a set of the strongest points over a uniform spatial distribution. For each of these points, we use image patch feature descriptors, subsampled from a blurred image and then bias-gain normalized. The features are matched and only feature matches for whose best match (first nearest neighbor) is significantly better than its second best match (second nearest neighbor) are kept.

(a) California Hall:

from clockwise starting at upper left: Result using only corner image, result with façade mosaic texture, façade mosaic texture

(b) Durant Hall: various views of our Durant Hall result with high resolution façade mosaic textures and roof included



(c) Hargrove Music Library: various acquired façade images, labelled corner image, resulting façade mosaic textured 3D model

Figure 5: Results

Lastly, we employ Random Sample Consensus (RANSAC) to robustly compute our homographies in the presence of spurious feature matches that survived the aforementioned steps.

After the façade mosaic is automatically produced, we manually crop out the façade from the rest of the image.

2.6. Putting it all together

The final step in our system is to integrate the high resolution façade mosaics with the 3D model deduced from single view modeling. To do this, we simply texture each plane of the 3D model with the corresponding façade mosaic.

3. Results

We show results for three buildings in Figure 5. The advantages of using facade mosaics for texturing rather than just the corner image and warping are clear from the California Hall result in Figure 5a. Notice that the corner image derived texture result is much lower resolution, has many more artifacts from occluding objects like trees, suffers from warp distortion (e.g., the bushes are significantly wider than they should be), and misses details (like the windows towards the back).

The Durant Hall result shown in Figure 5b is our favorite. The facade mosaics turned out especially well due to being able to avoid all occluding objects with the quadcopter, good uniform lighting from an overcast sky, and the inclusion of roofs in our model. Note that we manually tweaked the corner positions of the pieces forming the roof for aesthetics.

Lastly, we show a result on a non-rectangular prism building, which is the Hargrove Music Library, in Figure 5c. The upper section of the building is fairly non-rectangular, so we do not include that in our model. We simply use the first story of the building which can be approximated as a box more easily.

4. Future Work

To extend the capabilities of this system, we would like to implement more complex models such as Debevec's approach where the user specifies a basic model and stereo is used to refine the model with more detail [5].

Furthermore, it is sometimes not possible to avoid including occluding objects in façade images even with the quadcopter. To get rid of artifacts like trees, we would implement a feature like Sinha where the user can specify trees to remove, and texture is generated to fill the hole [6].

5. Acknowledgements

The authors would like to thank the fantastic course staff including Professor Efros and GSIs Shiry and Shubham for teaching us so much and offering such a fun class. The components of this project were all based on topics covered in lecture or projects and constitute some of our favorite techniques learned from the course.

References

[1] Efros, Alexei. “Introduction” [PPT document]. Retrieved from CS194-26 Image Manipulation & Computational Photography Lecture Notes. 2014. http://inst.eecs.berkeley.edu/~cs194-26/fa14/

[2] Criminisi, Antonio, Ian Reid, and Andrew Zisserman. "Single view metrology." International Journal of Computer Vision 40.2 (2000): 123-148.

[3] Brown, Matthew, Richard Szeliski, and Simon Winder. "Multi-image matching using multi-scale oriented patches." Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Vol. 1. IEEE, 2005.

[4] Zhang, Zhengyou. "A flexible new technique for camera calibration." Pattern Analysis and Machine Intelligence, IEEE Transactions on 22.11 (2000): 1330-1334.

[5] Debevec, Paul E., Camillo J. Taylor, and Jitendra Malik. "Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach." Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. ACM, 1996.

[6] Sinha, Sudipta N., et al. "Interactive 3D architectural modeling from unordered photo collections." ACM Transactions on Graphics (TOG). Vol. 27. No. 5. ACM, 2008.