Reconstructing Historical San Francisco from Aerial Imagery

CS231A Final Project

Justin Manley and Joseph Bolling
justinjm@stanford.edu jbolling@stanford.edu

Abstract

While modern aerial imagery is frequently used to produce accurate 3D reconstructions of buildings, cities, and natural features, there has been little work to do the same with historical photographs. Though these datasets exist, they present their own unique challenges as the imaging systems used are rarely designed with computer-aided reconstruction in mind. We present a processing pipeline that accepts coarsely-georectified aerial photographs and produces a 3D point cloud of any region appearing in at least two images. Our methods leverage the existing georectifications (available for most historical aerial photo collections) to reduce the search space and save processing time, and to provide an initial estimate of structure. We employ AKAZE features to sparsely match images and perform a bundle adjustment using the Levenberg-Marquardt algorithm to produce an accurate point cloud. While our techniques were developed specifically for use on the set of aerial photos of San Francisco captured in 1938 by Harrison Ryker, we believe they will generalize well to any set of georectified historical photos.

Introduction

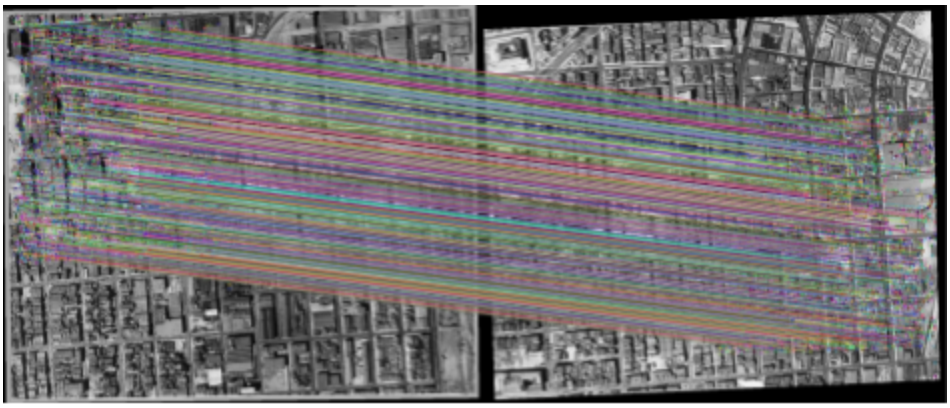
Historical photographs hold a wealth of information about the visual and spatial contours of the recent past. When buildings are demolished, artworks destroyed in violent conflict, and landscapes transformed by human settlement, photographs preserve precious fragments of our cultural heritage and natural history. However, the unique challenges posed by historical imagery have limited the utility of the photographic archive. Historical photographs are typically taken for human viewers, not computers, and so they lack the redundancy that is required by computational methods which exploit overlap disparities. Age, wear, and damage introduce artifacts which make the problem more difficult. Historical images also lack metadata like camera calibration and location information. The small size of these photographic collections make it difficult to recover missing metadata (projects like the PhotoTourism leverage the large size of the dataset to estimate the placement and calibration of uncalibrated images).

We design a system to perform 3D reconstruction of a landscape from uncalibrated, historical aerial imagery. This system is applied to the particular case of reconstructing 1938 San Francisco from a set of historical aerial photographs.

Problem Statement

We propose to generate a 3D reconstruction of the topography of San Francisco in 1938 from a set of 164 black-and-white aerial photographs of the city taken in 1938 and digitized in 2011 by the David Rumsey Map Collection at Stanford. The photographs have been orthorectified and mosaiced, but the topography has not been reconstructed.

There are a number of challenges to reconstructing the topography of the city from these historical aerial images. Chief among these challenges is that there is limited multi-image coverage of the city. Approximately 50% of the city is covered by only a single image; small patches of the city are covered by two, three, and occasionally four images where the



Matched AKAZE Features from overlapped regions of two images after pruning with a RANSAC fit

aerial photographs overlap (see overlap map). In addition, there is no calibration or altitude information for the camera. Conventional methods for processing aerial imagery rely on rational polynomial coefficient (RPC) sensor models provided by the camera manufacturer, as well as altitude data from the camera. The sparsity of the images and the lack of camera calibration make such methods unsuitable for this problem.

Related Work

3D reconstruction as a field has seen a wealth of work in the last 30 years. Most applicable to our pipeline is the Phototourism project, which reconstructed famous landmarks from around the world using uncalibrated datasets of images from the internet. The Phototourism project relies on the SBA (Sparse Bundle Adjustment) library developed by Lourakis and Argyros, which we use in our implementation and which is remarkable for its flexible implementation of the Levenberg-Marquardt nonlinear optimization algorithm.
Sevara et Al present a similar pipeline for extracting geographic contours from historic flight imagery. Their implementation contains useful pre- and post-processing steps for working with historic photos, but relies on proprietary software packages for the majority of the 3D reconstruction task. We seek to extend this work and make it more accessible by using only open-source libraries that are available to any developer.

One of the challenges in developing this system is that of finding a feature detector that was robust to the unique noise sources in the Ryker image dataset. We find the accelerated version of Alcantarilla et al's KAZE features to be very effective. The nonlinear scale space used in KAZE features avoids smoothing out higher-frequency detail that is lost to traditional gaussian feature detectors like SIFT and ORB. For the AKAZE (Accelerated KAZE) feature implementation and for many of the other traditional vision techniques we use, we turn to the OpenCV library.

Technical Approach

Image Partitioning

Because Ryker's aerial photographs were taken before the development of computational methods for 3D reconstruction, successive images overlap only in small areas. Matching and cropping these areas of overlap limits the scope of the keypoint identification problem and increases the accuracy of the keypoint matches.

We use GDAL and CGAL to match and crop the images. The boundaries of the orthorectified images are arbitrary quadrilaterals (rectangles with a projective transformation applied). We extract these orthorectified boundaries using gdalinfo.

We insert the boundaries into a CGAL arrangement. For each image, we output a set of masks corresponding to the faces of the arrangement which overlap with the image. We use the resulting masks to crop the images. We also perform image fragment matching during this step. Once the masks are extracted, we map them back to the original images by calculating a projective transformation between the orthorectified geographic coordinates and the image coordinates.

The benefit of treating the set of images as an arrangement is that, unlike a triangulation or polygon partitioning approach, the resulting partition respects the boundaries of the images.

Keypoint Detection & Matching

One of the challenges of the Ryker dataset is the condition of the photos. There are numerous small tears and abrasions that, while they don't significantly impact or distort the structure of the photographs, provide sharp edges that draw many Gaussian-based keypoint detectors. What's more, at some point in its history the photo collection was hand-labeled with street names in black ink. These labels present additional strong edges and are particularly challenging for keypoint finding, as the strongest keypoints are formed by the labels. We investigated the possibility of using the Neumann-Matas text localization algorithm to bound and mask out these labels, but determined in testing that direct modifications to our keypoint finding algorithm provided a less-expensive and more elegant alternative.

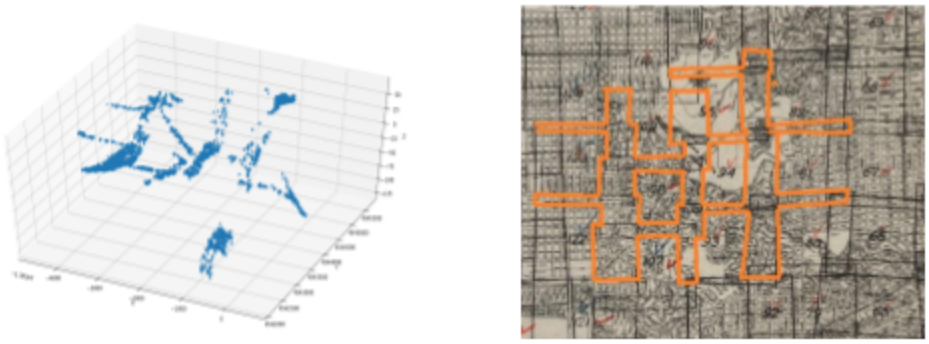
Rather than using a gaussian-based keypoint detector such as ORB, we found that AKAZE features were extremely robust to sharp edges, remaining evenly distributed across the image and providing plenty of "real" features aside from the outliers produced by the street labels. After a brute-force search for keypoint matches, we select pairs of keypoints that pass the ratio test as proposed by Lowe. These keypoint matches are further pruned using a RANSAC homography fit to remove any remaining outliers. Finally, we discard the surviving matches if there are less than 8 total, since there is a high probability of finding a successful RANSAC fit that includes outliers with such small numbers of points.

Sparse SfM & Bundle Adjustment

Once an accurate set of matched keypoints in each overlapped region of the dataset has been found, we use a Structure from Motion processing pipeline to estimate both the locations of the image points in 3D space and the camera model used to capture each image. Our pipeline differs from many SfM processes in that we do not have an accurate description of the camera intrinsic properties a priori. Furthermore, given the coarse placement of the analog images on the scanner glass during digitization, the camera intrinsics may vary significantly from image to image. We use the Levenberg-Marquardt algorithm as implemented by Lourakis and Argyros in the SBA library to minimize the reprojection error of the points into the image plane as the camera model and point locations are varied, subject to a square pixel constraint.

Our camera model as used in the bundle adjustment pipeline has 11 free parameters: focal length (1) aspect ratio (1) skew (1) image center (2) rotation relative to the world frame (3) and translation relative to the world frame (3). While the rotation is stored as three parameters, during calculations it is represented as a quaternion with an extra degree of freedom. This avoids the problem of gimbal lock that occurs in Euler-rotating systems. Each image has a unique and separately-optimized camera. Each point in the world fram is three-dimensional, and is represented on a 1 meter scale.

To initialize the parameters in the model, we assume that all the 3D points lie on a perfect plane with altitude Z = 0, and all cameras are oriented directly down at the ground at an altitude of 2000m. We use



Right: Pointcloud reconstruction of Twin Peaks, San Francisco with vertical scale enhanced. Slopes and altitudes appear realistic to the modern peaks and the surrounding valleys
Left: Overlay regions used in twin peaks reconstruction

the georegistered pixel coordinates for image coordinates, meaning that the (u,v) coordinate axes in the image plane are aligned with the cardinal axes in the world plane. For simplicity of initialization, we set the world coordinate system to have positive X in an eastward direction, positive Y in a southern direction, and positive Z oriented towards the ground. This ensures that the initial rotation between the world frame and the camera frame is zero. The X and Y coordinates of points in the world frame are initialized using the georegistration transform provided with each digitally registered image in the collection.

We run the Levenberg-Marquardt bundle adjustment routine on all points and images simultaneously, instead of on one pair of overlapped images at a time. This allows information about camera intrinsics derived from one overlapped area to inform the optimization routine in another overlapped region of the same image.

In our initial implementation, we do not bother tracking individual world points across more than two images at a time. If a point appears in more three photos, say, it is optimized separately, as if it were three points, one appearing in images one and two, one appearing in images 2 and 3, and one appearing in images 1 and 3. Empirically the routine still performs well despite the extra degrees of freedom, and point clouds in multiple-overlap regions appear to effectively coregister.

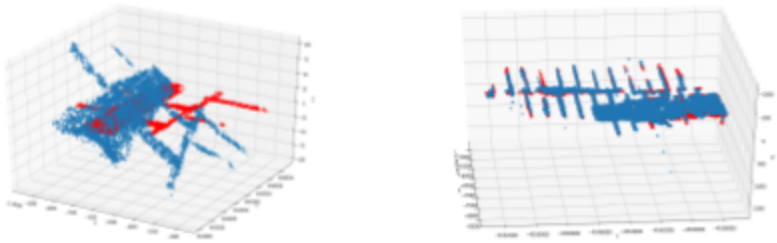
Results

The full pipeline can process a set of 10 overlapped images in approximately 15 minutes. The time grows faster than linearly for larger numbers of images; because there are multiple overlapping polygons created with each additional image, it takes many hours to process the entire set of 154 images. Interestingly, the iterative bundle adjustment is one of the fastest parts of the process, taking approximately 10% of processing time. The majority of time is consumed by the AKAZE feature calculation process, followed by the geometric segmentation of the images.

The point cloud produced by our process effectively maps the macro-features of the San Francisco landscape. Hills and valleys are realistic to their modern topography, as can be seen in the figure showing Twin Peaks. Finer features, such as buildings and streets, are not visible in the 3D structure except as variations in the point cloud density because of the relative availability of good features on different surfaces. It is difficult to gauge at what scale the cloud begins to resolve 3D features,

as there area few structures in 1938 San Francisco between the size of a four-story building and a hill; perhaps the Salesforce tower or the Transamerica Pyramid would have been resolvable had they been constructed in the 1930s.

It is remarkable how important the number of images is to reliable reconstruction, even when those images do not overlap significantly. A reconstruction using just three or four images will often show significant affine distortion, which is greatly reduced as the image set is expanded beyond ten photos.



Left: A section of the Mission (a largely flat neighborhood) displays significant affine ambiguity between the reconstructed points (blue) and the zero-elevation initial estimates (red)when reconstructed using 9 images.
Right: The same region, when including an additional 14 surrounding images, shows flat contours where appropriate and diverges from the red initialization where there are hills, accurate to the physical geography

Evaluation

We propose (though we do not implement) a method of evaluation against digital elevation models (DEMs) that can be used to assess the accuracy of such reconstructions. First, the reconstructed points must be brought into the same geographic coordinate reference system as the DEM (this can be done with widely-available tools like the Geo Data Abstraction Library, or GDAL). Second, each reconstructed point should be associated with a pseudo-point in the DEM (the pseudo-point can be, for example, the center latitude and longitude of the bin containing the reconstructed point, and the elevation of that bin). The associations between the two sets of points can be used to fit an affine transformation between the reconstruction and the digital elevation model. This affine transformation is important because the reconstruction given by structure-from-motion is only known up to an affine transformation, so even a high-quality reconstruction need not match up at all with the DEM. This affine transformation, once obtained, can be applied to the reconstructed points to bring them into alignment with the DEM. Next, the reconstructed point cloud should be binned at the resolution of the digital elevation model, and the elevation averaged over each bin. The difference between the binned reconstructed points and the DEMs can be visualized as a heatmap to show the accuracy of the reconstruction. Aggregate statistics of this binning (e.g. RMSE) can be used to provide a quantitative summary of the quality of the reconstruction.

Future Work

While our point cloud is accurate at the large scale, we believe there is work that can be done to improve its accuracy at the small scale. This might rely on more accurate or careful keypoint finding, or perhaps on supplementing the aerial dataset with contemporary photographs shot from the ground, in the style of the Phototourism project.

We believe that one of the most severe obstacles to practical use for our reconstruction is its sparseness, and the fact that significant sections of the city only appear in individual images, foiling the use of

conventional binocular stereo reconstruction techniques to obtain 3D information. Though difficult, we don't believe this problem is insurmountable, especially given the recent developments in deep learning for image processing. We envision an extension to this pipeline that uses a deep neural network trained on overlapped sections to estimate the 3D structure of non-overlapped regions of the city. The structure of this agent might be supplemented with traditional monocular metrological techniques such as vanishing point analysis.

We are most excited by the potential for this technique to be applied to other datasets and other regions of the globe. At its best, such a system might become a 3D software suite similar to Google Street View, but which allows the user to explore an area through time as well as space. We think the possibilities are very exciting.

Appendix

Code:
https://github.com/garlic-guardian-and-the-crouton-kid/ryker-sf