3D Building Synthesis Based on

Images and Affine Invariant Salient Features

A Thesis

Submitted to the Faculty

of

Drexel University

by

Chenxi Li

in partial fulfillment of the

requirements for the degree

of

Master of Science in Electrical Engineering

September 2017

ACKNOWLEDGMENTS

This master thesis project was carried out at Electrical and Computer Engineering department at Drexel University. I would like to thank my advisor, Dr. Fernand Cohen, for his patient guidance and constant support during the research and writing of the thesis. The help provided by him was important to the successful completion of my thesis. I'm also thankful to my committee members, Dr. Nagarajan Kandasamy and Dr. James Shackleford for their suggestions they gave throughout the thesis defense which are really helpful for optimization of the thesis.

Thanks also go to Dr. Zexi Liu, the Ph.D. alumni from Drexel, for the great help with the system design, and also to my friends and fellow students at Drexel University for the enjoyable and great time here.

Chenxi Li

September 2017

Table of Contents

ABSTRACT iv

LIST OF TABLES iv

LIST OF FIGURES v

CHAPTER 1 INTRODUCTION 1

1.1 Background of 3D modeling 1

1.2 Introduction to building reconstruction 2

CHAPTER 2 RELATED WORK 5

2.1 Methods for 3D building reconstruction and recognition 5

2.2 Situating our Synthesis Method within the Different Classes of Synthesis 7

CHAPTER 3 SALIENT POINTS 9

3.1 Properties of salient points 9

3.2 Additional salient points 11

3.3 Select salient points 12

3.4 Modeling from stereo views 13

CHAPTER 4 AFFINE TRANSFORMATION 16

4.1 Relative and absolute invariances 16

4.2 Geometric Transformation 16

4.3 Affine transformation and weak perspective projection 20

CHAPTER 5 CONSTRUCT CORRESPONDENCES 22

5.1 Construct correspondence under affine transformation 22

5.2 Imposing order on salient points 23

5.3 Find correspondences using convex hull 23

5.4 Construct nested convex hull 25

CHAPTER 6 3D model synthesizing 27

6.1 Geometric model synthesizing 27

6.2 Texture mapping 28

6.3 Computational complexity associated with our 3D synthesis method 30

CHAPTER 7 EVALUATION 32

7.1 Geometric Error Evaluation 32

7.2 Texture error evaluation 33

CHAPTER 8 APPLICATION 35

CHAPTER 9 DISCUSSION 36

9.1 Conclusion 36

9.2 Future work 36

APPENDIX A: WEAK PERSPECTIVE 38

LIST OF REFERENCES 38

Abstract

3D Building Synthesis Based on
Images and Affine Invariant Salient Features
Chenxi Li
Fernand S. Cohen. Supervisor, Ph.D.

In this thesis, we introduce a method to synthesize and recognize buildings using a set of at least two 2D images taken from different views. Based on a coarse set of affine invariant salient feature points (corner points) on the images, a 3D high-resolution building model is obtained in accordance with the observed images. Corresponding salient points are found using the ratio of triangle areas formed from a set of four consecutive ordered salient corresponding points that form two triangles. The order is obtained by finding the vertices of the convex hull of the salient points. The salient points are tessellated to form a high-resolution triangular mesh with the appearance of a triangular patch in the image imported onto the personalized 3D model. With multiple images, all coordinates and appearances are reconstructed in accordance with the observed images. The 3D model reconstruction method allows for a 3D classification of a test building to one of many possible buildings stored in the database. The classification is based on a geometric 3D point cloud error. For buildings with very close 3D point cloud errors, a further classification is achieved based on the mean squared error (MSE) on the appearance of corresponding points on the test and base models. Our method can also be used in localization when preloaded location information of each model in the database is stored, hence helping an observer navigate without a GPS system.

List of Tables

Table 1: Geometric error values 32

List of Figures

Figure 1: Overall reconstruction process 4

Figure 2: Set of 8 different towers 10

Figure 3: Salient Feature Points on a Tower 11

Figure 4: Process of subdivision 12

Figure 5: Selected salient points on images 13

Figure 6: Overlapping area from different viewing angle 14

Figure 7: Images from stereo views 14

Figure 8: Rigid transformation 17

Figure 9: Similarity transformation 17

Figure 10: Affine transformation 18

Figure 11: Projection process with pin-hole model 19

Figure 12: Perspective projection 19

Figure 13: Weak and strong perspective 20

Figure 14: Process of weak perspective 21

Figure 15: Absolute invariant in affine transformation 22

Figure 16: Ordered vertices of convex hulls and area ratio invariance 24

Figure 17: Convex hull with occlusion 25

Figure 18: Nested convex hull layers 26

Figure 19: Loop subdivision 27

Figure 20: 3D model synthesis without texture 28

Figure 21: Loop subdivision on image 1 of Tower1 29

Figure 22: Increase interation time 29

Figure 23: Compare test Tower1 to base Tower1 and base Tower2 33

Figure 24: Simulated location recognition in experiment scene 35

CHAPTER 1 INTRODUCTION

1.1 Background of 3D modeling

Three-dimensional (3D) modeling can be regarded as a visualization of a set of data interactively displayed on computer. It may seem like a direct and simple way to reconstruct the 3D point cloud from dataset, however, this technology has been a long-lasting and worthwhile topic in computer vision and computer graphics. With essential functions in daily life, the 3D modeling can be applied in many areas such as recognition based on 3D models, navigation and digitalization of the archived documents. Moreover, 3D modeling can also be used as one of the contactless and elaborate methods to reconstruct the ancient architecture for archeology studies as well as applied in medical surgeries with a more realistic and easy-observable environment for doctors.

One of the applications of 3D modeling is used in medical research [1-3]. The straightforward method to get the 3D model in medical image is to get the statistic shape model from a set of known data [4]. There are lots of methods applied based on the statistic shape model. In [3], researchers provide a novel model called Active Appearance Model to learn from the model parameter displacement and their residual error through an iterative scheme. This model can also be used to recognize the displacement in human face reconstruction. Also, the segmentation of MRI images and CT scanners can be used to form 3D model in medical images of brain tissue and vessel [2]. Applying 3D modeling in historical monument allows for digital archive and non-invasively studies with more details of the historical building [5,6]. In [5], the author introduces a novel prototype library of parametric models which is called Historical Building Information Modelling. This library can automatically form the detailed information including 3D documentation, orthographical projections based on laser scanning. While in [7-9], 3D modeling can be used for digitalizing city models. Based on aerial images taken by

mature-airborne laser scanning, the geo-information can be easily acquired. Thus, the model of an urban area can be roughly reconstructed based on the 2D planner and geo-information.

1.2 Introduction to building reconstruction

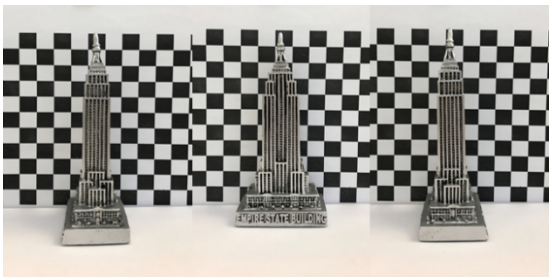
3D building and scene reconstruction based on 2D images and videos is an active and popular research topic in computer vision and computer graphics as it pertains to various applications ranging from non-GPS location identification, architectural studies, and archeological studies for heritage conservation and exhibition [10].3D structure and appearance are features of a building that uniquely lead to its unique characterization. The 3D structure can be had from images of the building through extracting salient corresponding points in the images from which a 3D cloud structure is computed via triangulation, while the building appearance is obtained by importing the appearance of an image patch to its corresponding 3D patch on the building surface.

This thesis deals with the problem of constructing a 3D textured model of a building or monument from images of the building by extracting corresponding salient points (corner points) on the 2D images. The salient points need to be intrinsic and invariant under the projection transformation, i.e. they need to be preserved under the projection transformation, which means salient points need to appear as salient points after the transformation unless there is occlusion. This step uses absolute affine invariants constructed based on the nested convex hulls of the salient points and yield a coarse set of corresponding points on the images. These coarse corresponding feature points are tessellated to form a high-resolution triangular mesh using a loop subdivision process. The subdivision produces a dense set of corresponding landmarks on the images of the building and result in a high-resolution 3D building model. This adds very little to the total computational cost (adding a few hundred milliseconds to the total processing time). Finally, the appearance of a triangular patch in the image is imported onto the personalized model. With multiple images, all coordinates and appearance are

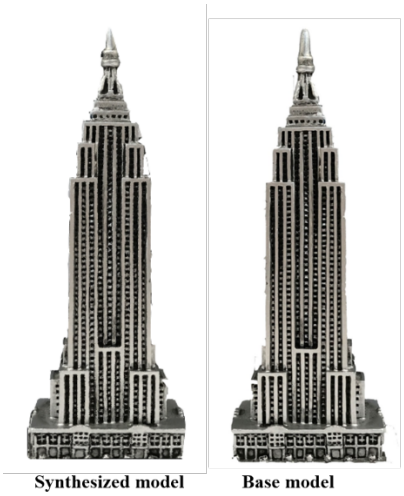
reconstructed in accordance with the observed images. The 3D model reconstruction method allows for a 3D classification of a test building to one of many possible buildings stored in the database. The classification is based on a combination of a geometric 3D point cloud error. The geometric error is obtained by calculating mean squared error (MSE) between one point in test model and its nearest neighbor point in base model. For buildings with similar shape but different appearance, the geometric ambiguity is resolved through calculating the differences in intensity between the test and each base belonging to the ambiguity set (set with very close shape errors). The overall process of a building synthesis/ recognition is shown in Figure 1.

Our method can also be used in image based navigation when preloaded location information of each model in database is provided, hence helping an observer navigate without a GPS system.

The thesis is organized as follows. In Chapter 2 we briefly cover related work, while in Chapter 3 we discuss the properties of salient points and how to enrich the salient point set. The affine transformation theory is discussed in Chapter 4 and we discuss why choose convex hull as well as how to construct nested convex hull in Chapter 5. The detailed process of our method applied on building models is discussed in Chapter 6 and we evaluate the geometric error combined with texture error in Chapter 7. The application and conclusion of the thesis are given in Chapter 8 and 9, respectively.



(a) Overall synthesizing process



(b) Synthesized tower juxtaposed against the true tower

Figure 1: Overall reconstruction process

CHAPTER 2 RELATED WORK

2.1 Methods for 3D building reconstruction and recognition

3D building reconstruction are needed for city mapping where urban centers are mapped and digitized for navigation and urban construction planning for crime prevention and disaster mitigation. This technology is also useful in archeology, where reconstruction of monuments (e.g. palaces, temples, etc.) can be achieved virtually yielding historical detailed information about a monument non-invasively and without physically damaging the monument remains. Finally, this technology lends itself to non-GPS image based navigation, where the position location system is attained from the identification of a building against a possible building stored in a database.

Many methods have been developed to accommodate different conditions in 3D modeling. The most direct method is to use 3D scanning system which all have similar steps. For the models which are scanned directly from a 3D scanner, the steps always contain shape measurement for geometric structure and texture alignment for model surface appearance [11]. For the physical model scan, multiple scanning images must be aligned into the same coordinate and cover the surface of the model. Multiple image registration is done by matching features from 2 overlapping images automatically. In [12], researchers introduce a method to simplify the surface matching problem into a 2D image matching problem through a harmonic map. Also, a method based on pin images of the 3D model is introduced in [13]. The surface is matched based on the highly correlated pin images which are the level descriptors to represent images. Besides the shape registration of the model, the reconstruction will be inadequate when missing the texture part. To align the texture on to 3D model accurately, the camera calibration is needed to provide intrinsic and extrinsic parameters [14]. And for models which have obvious geometric features, the correspondence can be found manually before image texture can be rendered on the 3D model. In [15], the author introduces the way to reconstruct 3D model based

on the RGB-D camera. This novel sensing system take RGB images with depth information for each pixel. With a RGB-D mapping, a dense indoor model can be formalized. While in [16], a hand-held stereo photometric camera is introduced to recover the shape and surface reflectance of a model from a set of images. The photometric constraint is added to the multi-view stereo problem which can help to solve the shape, surface normal and reflectance simultaneously. However, it is inapplicable for laser scanner to get the shape range from objects with shiny, mirroring and transparent texture. And this method requires time accuracy for a time-flight system [17].

Some of the current technologies are based on cartographical maps or 2D planar maps and aerial images based on Geographical Information System (GIS) database. Based on aerial images taken by mature-airborne laser scanning, the geo-information can be easily acquired. With the given aerial photogrammetry in a large area, urban 3D city models and building models can be reconstructed [7-9]. In [7], a 3D city model is reconstructed from detected planes extracted using Hough Transformation in 3D space. In [8], ground plans and airborne LIDAR data are combined using a 2D partitioning method to separate building footprint into 'nonintersecting and mostly quadrangular sections' as cell decomposition. This allows the creation of a level of detail building reconstruction. Building reconstruction based on aerial photogrammetry, however, can sometimes be not as accurate due to the low quality of laser scanning and cannot give a precise description of each building, and requires higher accuracy to achieve localization of a specific spot. In [9], a method to reconstruct 3D scene from video is introduced. This method can reconstruct the 3D scene using multi-view stereo reconstruction and a textured model in real time. This can be used in application like Google Earth and Microsoft Virtual Earth to visualize large scale models. In [18], a model-based building reconstruction based on coarse polyhedral model is proposed by finding the vanishing point then reconstructing more details by detecting features based on the maximum of the density gradient. This method is done using a close range 2D image, the model details, however, are

not enough for recognition. Another method for building reconstruction based on vanishing point is introduced in [19] using different vanishing points in 2D images and assigning points on lines with the same vanishing point to the common hue color. This method is called localized color histogram, which is regarded as an index vector for recognition. The authors also suggested finding SIFT points to refine their results.

There is an abundance of work [20,21] for 3D scene recognition that are based on SIFT point features. SIFT point features are pointed out by Lowe in [20], and these features are regarded as salient points to describe an object. In [21] informative SIFT descriptors are calculated to reduce the dimensionality by rejecting the majority of irrelevant descriptors formed by standard SIFT method. While in variant and invariant patch (VIP) features are generated based on SIFT features to match a 3D scene. This method can also be very effective when matching a 3D scene just from 2D images. However, basically, SIFT features are formulated in a grey-scale images, so it is hard for it to detect the differences between building appearance with the same structure.

2.2 Situating our Synthesis Method within the Different Classes of Synthesis

There are basically three ways for synthesizing a building structure. The first and most basic is using 3D scanners (time of flight) to obtain a 3D cloud of building points that represent a given building or monument, while the second uses acquired images of the building from which the 3D cloud building structure is obtained. The third method starts with a generic building model preferably one that belongs to the set of buildings that need to be synthesized, then personalize the generic model by morphing its 3D structure so it aligns itself with the images of the building that we are trying to synthesize. This approach was used for 3D face synthesis in [11,22]. The first method is more direct but is prone to errors in the presence of metallic surface resulting in reflection during scanning and is definitely more expensive and non-portable technology. The second method, while inexpensive and portable, requires the tracking of corresponding points

on the images to be able to reconstruct the 3D cloud via triangulation and necessitates cameras' calibration. While for the third method, it requires the models in the database have the same number of feature points for generalizing a generic model. For example, in the 3D face synthesis, all faces have the same number of features such as nose, mouth, eyes. However, it is hard to say if every building, even within the same type, will have the same number of floors or pillars. Thus, it will be difficult for us to construct a generic model for our building with this method.

The method used in this thesis belongs to the second class of methods. In this thesis, we use corner points as our salient points. These are invariably present in all buildings and can stand as common descriptors in the same way eyes, nose, cheeks, and mouth are common descriptors for all faces, notwithstanding the fact that their relative positions and shapes vary from one face to another. Moreover, these are intrinsic points that are preserved under the projection model (affine transformation). A coarse set of corresponding corner points on the images are obtained using a set of absolute affine invariants constructed from the set of convex hulls of the corner point set. Note that we have limited our transformation to affine since the transformation obtained from the camera model (perspective transformation) is well approximated locally by an affine transformation. This set is considered an intrinsic affine preserved set and go beyond the scale and rotation preserving property inherent in SIFTs.

In the next chapter, the salient points, which we mentioned several times, will be discussed including the properties of them and the proper salient point set in building models. After that, we will show the way to get a dense and rich salient point set based on original salient point set.

CHAPTER 3 SALIENT POINTS

Salient points can be regarded as the feature points in building models. As for a 3D face reconstruction indicated in [23,24], salient points are relied on features on human face such as eyes, nose, mouth and cheek. These points are treated as salient points based on their common existence on every face. Based on the salient points, we can easily find the correspondence between different faces and morph the generic model into a synthesized one. However, in our case the model which is going to be reconstructed are buildings. Even belongs to the same type, one building will not have the same number of features as others do. Thus, the generic model method is inadequate to be applied here for the generic model can only be well prepared when take the average of different models with same salient points.

The method we are going to use here is to track the corresponding points in different images and align them into a 3D model. As salient points must remain consistence and be preserved during the transformation, corner points on the building models are selected to be served as the salient points in reconstruction. As shown in Figure 3, salient points are marked in to red.

3.1 Properties of salient points.

In order to reconstruct the building model with a rich and dense salient point set, the salient points should have the following properties.

The salient points should be invariant, intrinsic and locally controllable. Firstly, salient points should be preserved under projective transformation and appear in the images as the salient points. Secondly, salient points should be intrinsic. Since we can take images of building from different angles, these points should be remained as salient points even we change the viewing angle. This requires salient points to be independent from the coordinate. Finally, salient points should be locally controllable due to noise and occlusion. For example, when noise occurs during the transformation, only the vicinity of the noise will be affected, leaving the ones not

in their noise vicinity to yield a good reconstruction. This also should apply to occlusion where only the occluded salient points are imparted.

One such set of salient points are corner points. These are invariably present in the set of different of towers shown in Figure 2, and clearly shown as frontal view image of tower 1 for example in Figure 3. Corner points map onto corner points under the affine transformation since intersection points between two lines map into intersection points after the affine transformation. This is also true for the midpoint between the line joining two salient points and any midpoints between those. This is of particular importance when obtaining a dense set of salient points from the corresponding corner points through the loop subdivision method outlined in section 3.2.

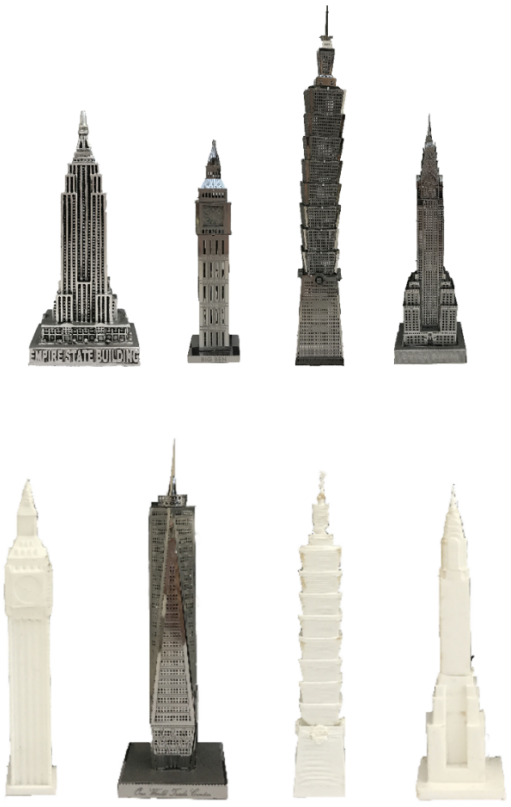


Figure 2: Set of 8 different towers

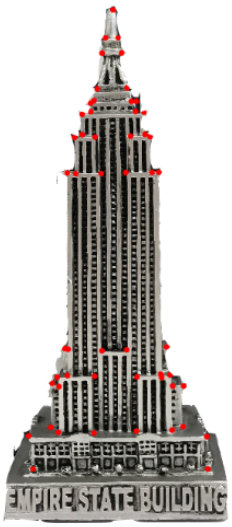


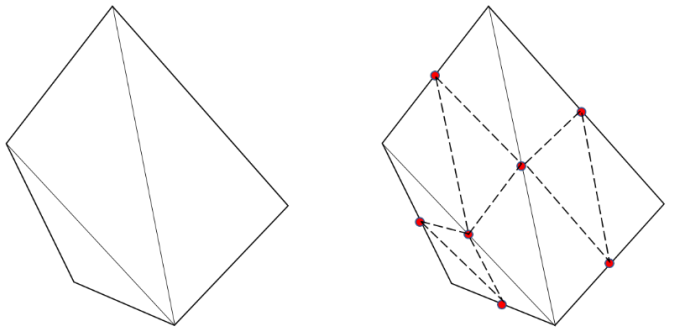
Figure 3: Salient Feature Points on a Tower

3.2 Additional salient points

Though the salient points are critical to present the geometric features on 3D model, only these original salient points are not enough. We need more salient points to build the points cloud of the model in order to express the model more accurately.

For a high-resolution 3D building model, we need more salient points. Under affine transformation, the midpoint of two points along a line will map to midpoint, we consider the midpoint of two adjacent salient points to be an additional salient point. Since midpoints are also salient points, this implies that all points on lines joining midpoints between themselves and the corner points are themselves salient points. By repeatedly selecting the midpoint of 2 adjacent salient points, we can enlarge our salient point set.

Using the coarse set of corresponding salient corner points for images 1 and 2, we form a Delaunay triangulation to tessellate the salient points in a triangular mesh (salient point mesh). The salient meshes have vertex-to-vertex and triangle-to-triangle correspondences in the frontal and profile images. Each mesh is then further subdivided through the loop subdivision routine in [23,24], where for each side of a triangle in the mesh, a midpoint is chosen. The loop subdivision process is shown in Figure 4. Suppose now we take a patch of building model with triangular mesh, the vertexes of the mesh come from the original salient point set. Then we take the midpoint along each triangular edge which are marked as red points shown in Figure 4 (b). Connecting the red points in the same triangular, a new triangular mesh contains original salient points and additional salient points will be formed after 1st time subdivision. By iteratively doing this, a set of rich salient points and dense triangular mesh will be constructed to describe the building model.



(a) Patch of building model (b) Patch after subdivision

Figure 4: Process of subdivision

3.3 Select salient points

The salient points are selected manually and ordered consecutively. Take Tower1 for a quick example. After taking 2 different images of the building from different views, we order the

corner points as the salient points manually on the images as shown in Figure 5. The salient points in frontal view which is shown as Image 1 have been selected and numbered. I just pointed out four of them in order to make the figure clear to be observed. After changing viewing angle into 90 -degree, the salient points will be ordered and have a common set with salient points selected from frontal view. The number of common salient points in the common set is determined by the overlapping area of different viewing angles. When the position where we take the images changes, the set of intersection of salient points also changes. Figure 6 shows how the overlapping area changes based on different viewing angles. We can see from the Figure 6 that the combination of Cams 1 and 2 is better than the combination of Cams 1 & 3 as it has a better view of the whole building.

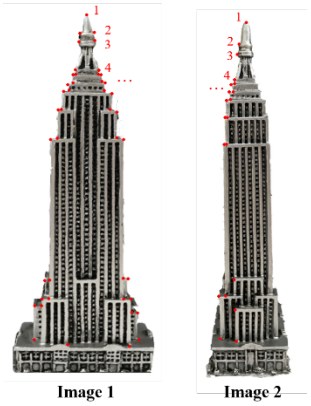


Figure 5: Selected salient points on images

3.4 Modeling from stereo views

In order to get the whole model of the building not just part of the surface, images from stereo viewing angles of one building are needed. Since the common set of the salient points is determined by the overlapping area of the viewing angles, we can choose viewing angles with

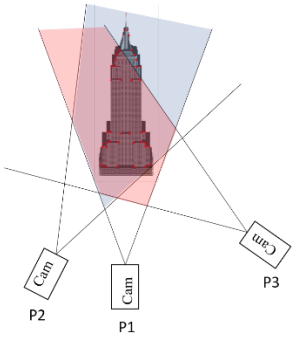


Figure 6: Overlapping area from different viewing angle

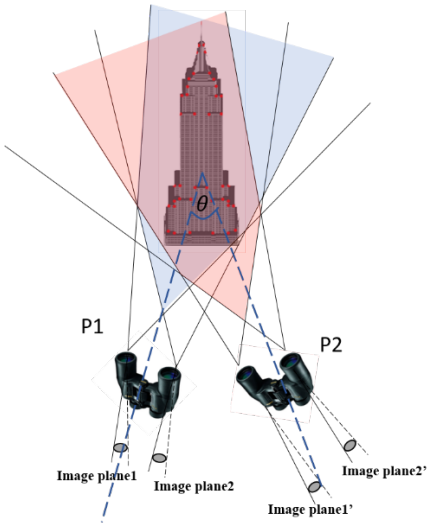


Figure 7: Images from stereo views

the largest overlapping area to reconstruct the whole building model surface when circulating around the model. As is shown in Figure 7, the overlapping area of the binocular viewing angles

contains all the selected salient points. If we rotate the binocular viewing system around the building on a plate with every θ (i.e. $\theta = 30°$ or $\theta = 45°$) degrees, the whole building will be reconstructed from all sides.

In next chapter, we will discuss the relationship (transformation) between different views under pin-hole camera model and invariants under different transformations.

CHAPTER 4 AFFINE TRANSFORMATION

4.1 Relative and absolute invariances

Invariance is an important property in the presence of geometric transformation[25]. There are two types of invariances: relative and absolute. When object's entities before and after the transformation are related through a scale factor that depends only on the transformation, then we have a relative invariance. For similarity transformations (rotation by an angle θ and scaled by a scale factor α in 2D), length is a relative invariance since it is scaled by the scale factor α, i.e., $l' = \alpha l$, where l and $l'$are the lengths between two points before and after the similarity transformation. For an affine transformation, area in 2D or volume in 3D is a relative invariance for it is scaled by the determinant of linear transformation matrix after an affine transformation. i.e., $dA' = \det[L]\, dA$, where dA and dA' are differential areas before and after the affine transformation. Absolute invariance is the invariance where the scale factor is 1. For example, under a similarity transformation, angles are absolute invariants and so is the ratio of lengths. For an affine transformation, the ratio of areas is an absolute invariant.

4.2 Geometric Transformation

An object in 3D when imaged from different views by a camera undergoes a geometric transformation. Depending on the conditions under which the object is imaged, different object image views are related through one of the following transformations: rigid, similarity, affine, or perspective transformation.

In a rigid transformation, a point in one view is mapped in accordance with

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = [R] \begin{bmatrix} x \\ y \end{bmatrix} + t$$

where $[x \quad y]^T$ is the original coordinate of the point, $[x' \quad y']^T$ is the coordinate of the point after the rigid transformation, $[R] = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ is a rotation matrix with θ being the angle of rotation, and t is a translation vector.

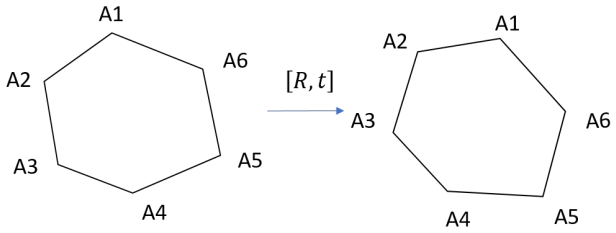


Figure 8: Rigid transformation

Under a rigid transformation, lengths and angles are absolute invariants.

Under a similarity transformation a point in one view is mapped in accordance with

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \alpha\,[R] \begin{bmatrix} x \\ y \end{bmatrix} + t$$

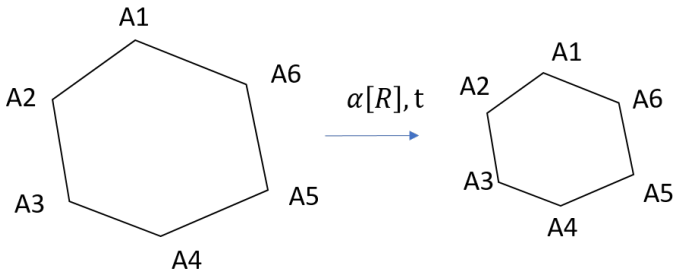


Figure 9: Similarity transformation

A similarity transformation preserves angles and the length ratios.

Under an affine transformation, the object is subjected to a shear and a point on the object is mapped from one view to another in accordance with

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = [L] \begin{bmatrix} x \\ y \end{bmatrix} + t$$

where $[L]$ is a nonsingular matrix. In affine transformation, length and angle are no longer relative invariants, however, area is, and the ratio of areas is an absolute invariant. As shown in Figure 10, we can obviously see that angels and the ratio of length change, but the area ratio $\frac{area\{tr(A1,A2,A3)\}}{area\{tr(A2,A3,A4)\}}$ remains the same after affine transformation.

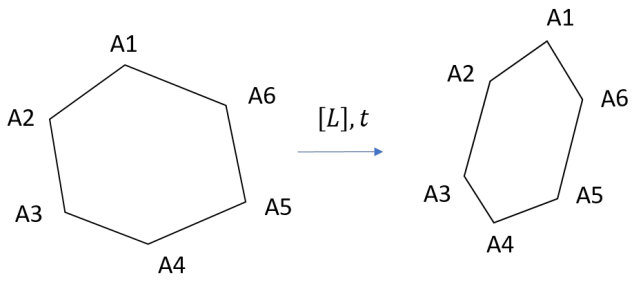


Figure 10: Affine transformation

Under a perspective transformation (pinhole camera) the image projection of a 3D point on an object follows the transformation shown in the Figure 11.

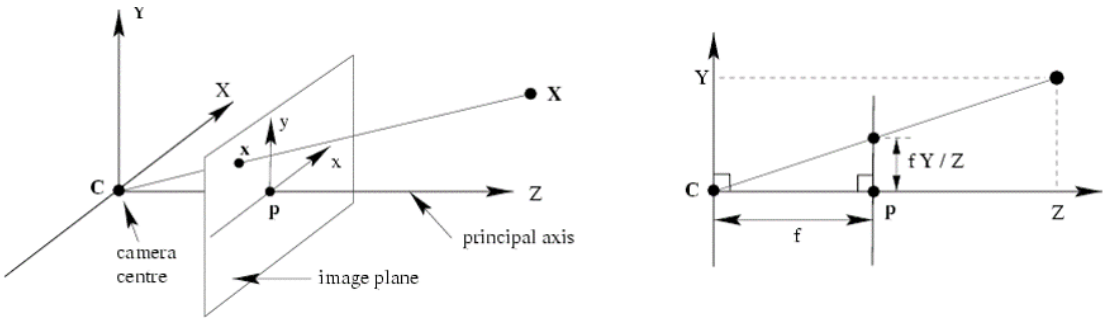


Figure 11: Projection process with pin-hole model

As shown in Figure 11, P is a 3D point with coordinate (x, y, z) with its projection onto the 2D image plane being the point (x, y, f), where f is the focal length of the camera. Note that the image plane is set in front of the camera center is to avoid getting an inverted image. Using the similarity property of triangles, it can be easily observed the relationship between x' and x, y' and y respectively, where $x' = f\frac{x}{z}$ and $y' = f\frac{y}{z}$. Figure 12 shows how an object perspectively maps on the image plane.

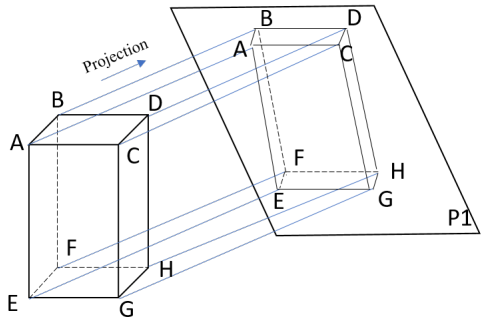


Figure 12: Perspective projection

Points on model are projected onto plane P1 through a perspective projection. It is interesting to see here that in the projective space, the parallel lines are no longer parallel as they converge into a vanishing point.

4.3 Affine transformation and weak perspective projection

The perspective projection can be divided into weak perspective and strong perspective. For the strong perspective, the parallel lines are not preserved and will intersect at the vanishing point (see Figure 13).

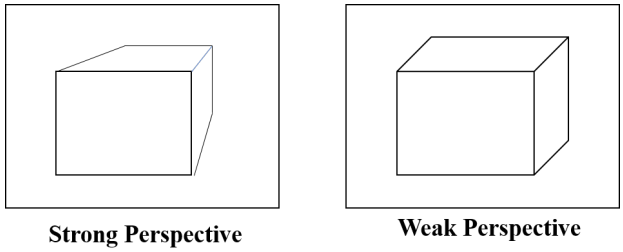


Figure 13: Weak and strong perspective

Under weak perspective shown in Figure 14, a planar patch dP_0 on the object with centroid r_0 is projected on a plane passing through r_0 and parallel to the image plane. This gives rise to an intermediate patch dP_I', which is perspectively projected on the image plane (just a scale factor since the intermediate and image planes are parallel). The combination of these two projections is the weak perspective projection, which is an affine transformation. When the patch size relative to the distance from the object to the camera is relatively small, the perspective transformation is a weak perspective, i.e., is an affine map globally or piecewise locally, rendering views of an object affine maps of one another.

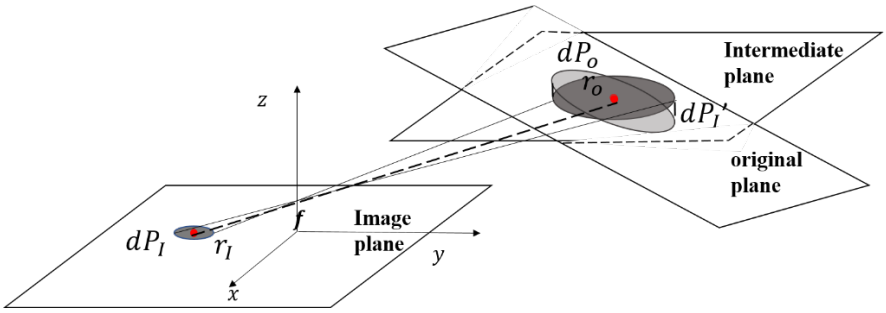


Figure 14: Process of weak perspective

In the next chapter, we show how we can use affine invariant constructed based on the assumption of affine maps to find correspondences between salient points contained in different views (images).

CHAPTER 5 CONSTRUCT CORRESPONDENCES

In this Chapter, we discuss how to find correspondences between salient points in the images from which the 3D model cloud is synthesized using triangulation. The correspondences are established using a set of the absolute affine invariants of affine transformation. Once the correspondences of the salient points are found, loop subdivision (at increasing resolutions) is used at a minimal extra cost to obtain an additional dense set of salient points with already established set correspondences. This is discussed in chapter 6.

5.1 Construct correspondence under affine transformation

Consider the two ordered sets of salient points given in images 1 and 2. We can declare points r_k, r_{k+1}, r_{k+2} and r_{k+3} in image 1 to points r_l, r_{l+1}, r_{l+2} and r_{l+3} as corresponding points if $I_1(k)$ and $I_2(l)$ are the same. Since there might be small errors in locating exactly the corner points we can allow for a small percentage error and use the error function given as $\frac{|I_1(k)-I_2(l)|}{I_1(k)} < \varepsilon$, with ε set to 0.05 for example.

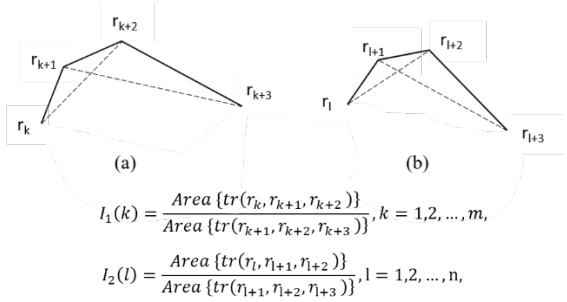


Figure 15: Absolute invariant in affine transformation

5.2 Imposing order on salient points

As we mentioned before, if we want to find the correspondences between 4 points in different salient point set based on the absolute invariant, we have to form all possible area ratios and apply the absolute invariant error metric to find the correspondences which is a time-consuming way. In order to speed up this process, we should impose an order on the salient points.

The first method is to order the areas into an increasing sequence and by taking the ratio of the consecutive elements in the sequence, the absolute invariant of affine transformation is derived. However, if the number of salient points in the point set is great, we still have to form the triangles for all possible point combinations.

The second method is to apply nested convex hull into the salient point set. Since the convex hull is affine invariant and has ordered itself, we can get the area ratio of the consecutive four points on the convex hull into an ordered sequence and the correspondences of the points are found by a circular shift. Since the salient points are divided into several convex hull layers, it would be much easier to find correspondences in each smaller point set.

5.3 Find correspondences using convex hull

Convex hull is the smallest bounding polygon that can contain all the points in dataset [26, 27]. To quickly establish correspondences between the corner points in the two images, we compute the convex hull associated with the extracted salient points in each image. The convex hull is affine preserved, which again means that the convex hull of the affine mapped points is an affine map of the original convex hull constructed from the data before the transformation, with the vertices before and after the affine transformation being an affine map of one another. With extra salient points appearing or disappearing because of the viewing angle of the images, some of the vertices of the two convex hulls will possibly disappear with the new ones possibly appearing. This only occurs when the salient points are near the vertices. In other words, the convex hull has local controllability with added or subtracted points possibly affecting the

convex hull only locally. It is also unique, computational efficient ($O(NlogN$, where N is the number of salient points). Finally, the convex hull has ordered vertices, which helps in the search for corresponding salient point vertices through a simple circular shift.

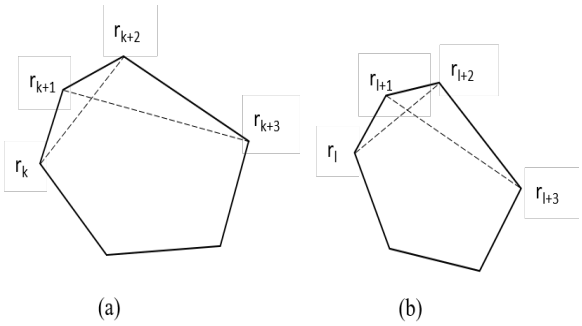


Figure 16: Ordered vertices of convex hulls and area ratio invariance

Every three consecutive vertices on the convex hull form a triangle. The area ratio computed from 2 adjacent neighboring triangles (constructed from 4 consecutive vertices) is an absolute invariant, which means it remains the same before and after the affine transformation as long as we consider the 4 corresponding vertices after the transformation as shown in Figure 16.

If $I_1(k) = \frac{Area\,\{tr(r_k, r_{k+1}, r_{k+2})\}}{Area\,\{tr(r_{k+1}, r_{k+2}, r_{k+3})\}}, k = 1,2, \dots, m,$ is the set of absolute area invariants constructed from the vertices of the convex hull of the salient points of the first image, and $I_1(l) = \frac{Area\,\{tr(r_l, r_{l+1}, r_{l+2})\}}{Area\,\{tr(r_{l+1}, r_{l+2}, r_{l+3})\}}, , l = 1,2, \dots, n,$ that of the second image, then we declare the set of 4 consecutive vertices r_k, r_{k+1}, r_{k+2} and r_{k+3} on image 1 to be correspondent to the set of 4 consecutive vertices r_l, r_{l+1}, r_{l+2} and r_{l+3} on image 2 when $I_1(k) = I_2(l)$. We can allow a nonzero small error margin for declaring correspondences by adopting the following decision: $\frac{|I_1(k) - I_2(l)|}{I_1(k)} < \varepsilon$, where is a small fraction like 0.05 for example.

In the presence of occlusion, as shown in Figure 17 in image (b) with point P occluded, there will be no corresponding point P' to P but the other 5 points will be found as corresponding points [28].

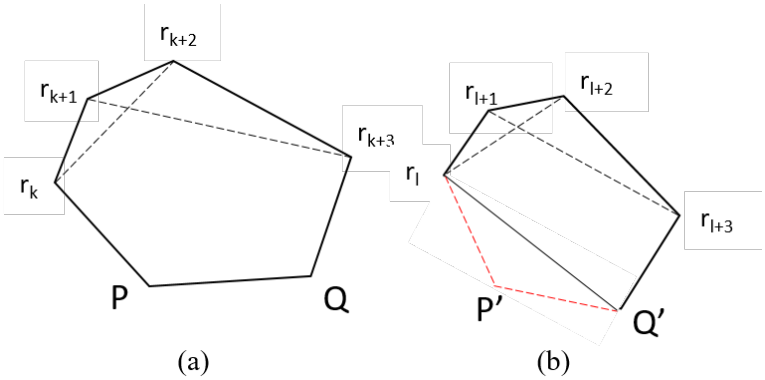


Figure 17: Convex hull with occlusion

5.4 Construct nested convex hull

To enlarge the set of corresponding salient corner points, we find repeat the process on a set of nested convex hulls as shown in Figure 18. This is done by removing the vertices of the outer layer convex hull and finding the inner layer convex hulls in the two images and establish correspondence on those. The process can be repeated until the salient point data is exhausted. Note that finding correspondences based on each convex hull layer is much easier than finding correspondences in a larger dataset. The end of this process results into a coarse set of salient corner points correspondences. This set serves as the basis for tessellation resulting in a dense set of corresponding salient points.

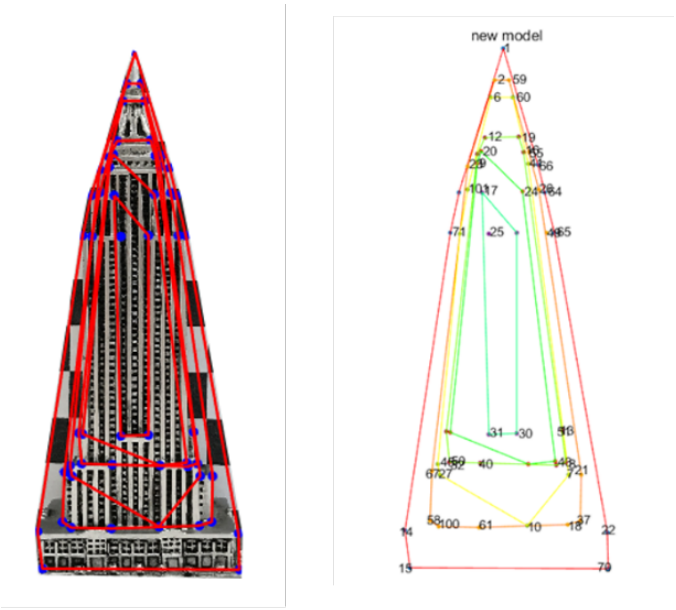


Figure 18: Nested convex hull layers

CHAPTER 6 3D model synthesizing

6.1 Geometric model synthesizing

As stated in section 3.2, since under the affine transformation a midpoint on a line is affine preserved, this means that the new generated set of points are also corresponding in the two images. Based on the correspondences found through the nested convex hull method, the corresponding salient point set can be enlarged through subdivision without finding correspondences, i.e. they are already declared as correspondences. The subdivided meshes maintain the vertex-to-vertex correspondences. This mesh loop subdivision is shown in Figure 19 for images 1 and 2 of tower1.

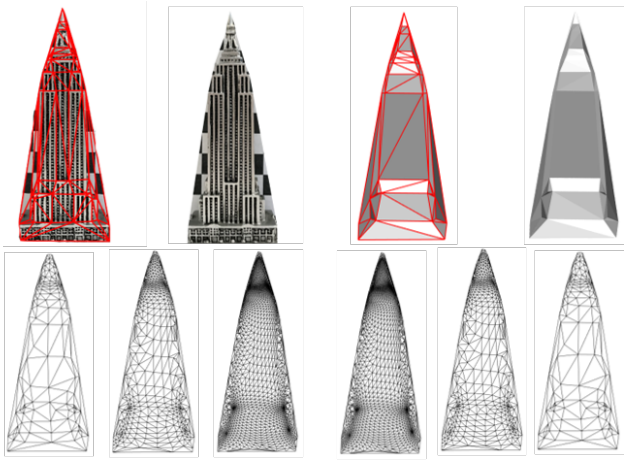


Figure 19: Loop subdivision

The 3D building model without appearance (texture) from the dense salient point set is synthesized as is shown in Figure 20.

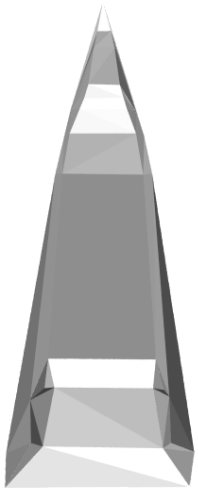


Figure 20: 3D model synthesis without texture

6.2 Texture mapping

A complete 3D model for the tower is obtained when considering images taken all around the building. We can add a textured appearance to the 3D tower model by importing the appearance of any triangle on the images to the corresponding triangle using tray racing on the 3D structure. This is shown in Figure 21 for tower 1. For each patch on the high-resolution 3D building mesh, its intensity is imported as the average intensity from 3 vertexes. When the iteration time of the subdivision increase, the appearance of the building will be smoother and consistent as shown in Figure 22.

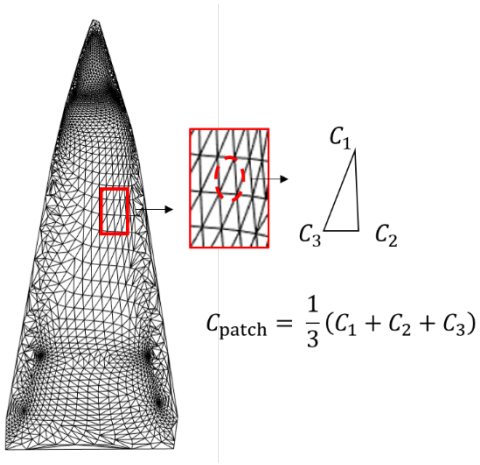


Figure 21: Loop subdivision on image 1 of Tower1.

Figure 22: Increase iteration time

6.3 Computational complexity associated with our 3D synthesis method

Computational complexity is an important index to describe the computational time or storage needed in executing our method. In this paper the corner points in the images are selected using a GUI system. As this part of the algorithm is not yet automated and is beyond the scope of this thesis, the computational complexity commences after the acquisition of the corner points. Note that in a fully automated system, a corner edge detector [29] will be used to extract the corner points in the images. Suppose we have a set of N salient points which are selected. Then the first step is to construct the nested convex hull, then find the correspondences for points on each convex hull, respectively. We then establish a dense salient point set by subdivision, finally map the intensities of points on the images onto corresponding points on the 3D model. Firstly, the computational complexity for generating the convex hull of the N points dataset is O(N log N) [30]. Since each time we generate the inner convex hull we have a lesser number of data points, the total complexity is bounded from above by $\alpha O(N \log N)$, where α represents the number of layers in the nested convex hulls. Secondly, the finding of the correspondences of the points on each convex hull layer has is of order M_k, where M_k is the number of vertices on each convex hull $k$since the convex hull has ordered vertices and the correspondences therefore are found through a circular shift. This is followed by a Delaunay triangulation where the corresponding corner salient points in each image are connected as a mesh with non-intersectional edges. The computational complexity of Delaunay triangulation for each image is of order N log N [30], for N is the number of corner points with established correspondences. The additional salient points are selected as midpoints between two adjacent points. For a point set containing N points, there will be no more than 3N edges in any triangulation. Thus, finding the midpoints on edges will have the computational complexity of O(3N), with increasing resolution (loop subdivision iteration), the computational complexity is of order $3^k N$ where k

is the iteration stage. Finally, to import the intensities from image to the model, generally each vertex on the triangle will be touched 3 times to calculate the intensities of the around 3 triangles except for the points on the boundary. Thus, the computational complexity is $O(3^{k+1}N)$. For example, for test Tower1 for an example with N=47 corner points, the time for constructing nested convex hull is 0.193 sec; for finding the correspondences between salient points is 0.211 sec; and for obtaining additional salient points to dense point set per loop subdivision is 0.003 sec; whereas the time for texture mapping is 0.043sec; whereas for 7 times subdivision the time for finding additional salient point is 12.881sec. Thus, the computational complexity for selecting midpoint on each edge will be much smaller than that of constructing the convex hulls and the Delaunay triangulation. Hence, the method is able to add an additional salient point set at minimal cost. However, with the increase of iteration time, the time for finding the midpoints will increase since there is a 3^{k+1} term in it.

CHAPTER 7 EVALUATION

7.1 Geometric Error Evaluation

In this section, we test the efficacy of our synthesis model for 3D classification of a test building (synthesized using two images) to one of 8 possible buildings shown in Figure 3 stored in the database. A geometric error is obtained by calculating the mean square error (MSE) between one point in the test model and its nearest neighbor point in the base model. The test models are reconstructed and centered in the same coordinate as the base models. The classification errors are shown in Table I, with the table entries showing the MSE distance between the point clouds of every test model and base model.

Table 1: Geometric error values

Geometric Error (mm)	Base Tower1	Base Tower2	Base Tower3	Base Tower4	Base Tower5	Base Tower6	Base Tower7	Base Tower8
Test Tower1	**0.6496**	8.3792	4.0086	5.5078	8.8455	8.8113	7.0114	8.6035
Test Tower2	2.6926	**0.7513**	4.6711	2.4358	0.7924	2.6991	1.1122	2.3341
Test Tower3	5.0746	8.8628	**1.7758**	6.3684	9.3125	8.6343	7.5001	9.4213
Test Tower4	2.5334	3.4174	2.6862	**1.0454**	3.6499	3.7493	2.8735	2.8559
Test Tower5	3.1153	0.7093	5.2709	2.3646	**0.3621**	2.0273	0.7926	1.7380
Test Tower6	5.0505	2.0244	5.9182	3.2491	2.0032	**0.7978**	1.7468	2.3306
Test Tower7	3.0211	1.3401	4.4444	2.8135	1.0950	2.1639	**0.7632**	1.4341
Test Tower8	3.6117	1.4247	4.9204	2.2466	1.4591	2.1220	1.3995	**0.4382**

A larger value refers to a greater difference between two models while a smaller value represents a higher similarity. As expected entries along the diagonal in Table I, point to the smallest error values indicating a 100% correct classification. It is interesting to note that the

errors are much larger for dissimilar towers. We also note that the errors for two towers with similar shapes but different facades will have very close geometric errors (e.g. towers 2 and 5 have similar shapes but different textures), hence the importance of incorporating appearance as well as geometry for these cases for correct classification.

When comparing test model with base Tower1 and base Tower2, we can easily find from the point error in Figure 23 that the error distribution is close to 0 when the test tower is compared to base tower1, whereas it is non-uniformly distributed away from 0 when compared to tower base 2.

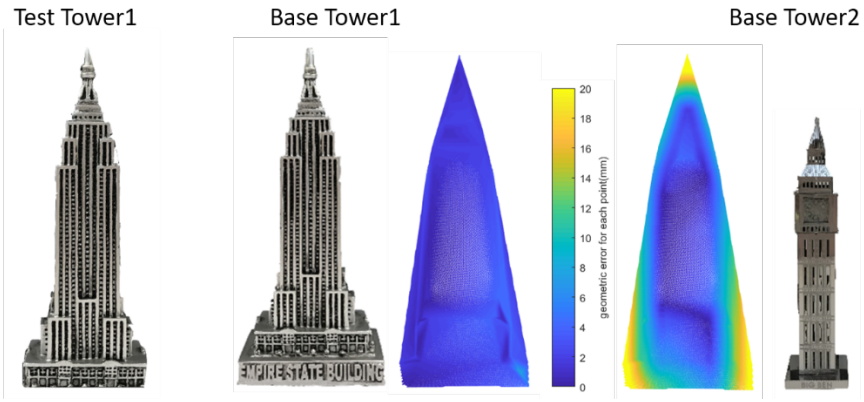


Figure 23: Compare test Tower1 to base Tower1 and base Tower2

7.2 Texture error evaluation

As indicated in the previous section when two towers have similar shapes but different textures or appearances, the classification geometric errors will be almost identical. To resolve this ambiguity, we compute the MSE in intensity between corresponding points on the test and base towers belonging to the ambiguity set with very close geometric errors, and pick the base with

the smallest intensity MSE. The intensity error is calculated under the normalized gray scale with 0 representing black and 1 representing rather than on a 0 to 255 scale. The errors between test tower 2 and base tower2 and base tower5 were found to be 0.1398 and 0.7380, with the minimum value pointing to the right classification.

CHAPTER 8 APPLICATION

Based on the evaluation results shown in Chapter 7, buildings can be reconstructed and well identified through this method. If the location information for each base tower is apriori known then using our reconstruction and classification method, navigation and localization is possible without connecting into GPS. For example, if an observer is navigating in the scene shown in Figure 24 with known coordinates for the 8 towers appearing in the scene, then the observer would be able to locate him/herself based on identifying the building the observer is looking at by taking a couple of images for the building s/he is looking at, then grab the location information from a database of landmarks. Based on the location information, the observer could then be aware of his/her location.

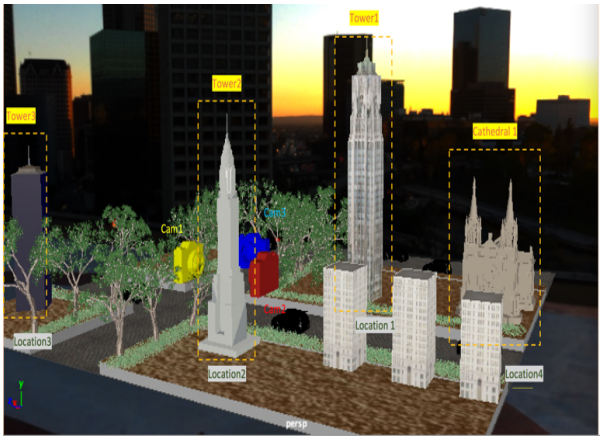


Figure 24: Simulated location recognition in experiment scene

CHAPTER 9 DISCUSSION

9.1 Conclusion

We constructed a 3D textured model of a building or monument from images of the building by extracting corresponding intrinsic and invariant (under the projection transformation) salient points (corner points) on the 2D images. We constructed a set of absolute affine invariants based on the nested convex hulls of the salient points to yield a coarse set of corresponding points on the images. These coarse corresponding feature points are tessellated to form a high-resolution triangular mesh using a loop subdivision process. The subdivision produces a dense set of corresponding landmarks on the images of the building and results in a high-resolution 3D building model. Finally, the appearance of a triangular patch in the image is imported onto the personalized model. With multiple images, all coordinates and appearances are reconstructed in accordance with the observed images. The 3D model reconstruction method allows for a 3D classification of a test building to one of many possible buildings stored in the database. The classification is based on geometric 3D point cloud error. The geometric error is obtained by calculating mean square error (MSE) between one point in test model and its nearest neighbor point in base model. For buildings with similar shape but different appearance, the geometric ambiguity is resolved through calculating the differences in intensity between the test and each base belonging to the ambiguity set (set with very close shape errors). Our method can also be used in image based navigation when preloaded location information of each model in the database is provided, hence helping an observer without a GPS system.

9.2 Future work

(1) We will consider a richer set of salient points under the weak perspective and beyond. Since the common set of salient points from different viewing angles will be maximized if we choose

an appropriate viewing angle, we can formulate the 3D surface of the whole building model if we enlarge the salient point set of the whole building.

(2) In this thesis, we approximate the weak perspective into affine transformation. In order to get more precise correspondences, we can consider a set of perspective invariants to establish correspondences.

(3) We are now selecting the salient points from images manually with a GUI. In order to get a more precise location of the salient points without distortion cased in selection, we are considering extract the salient points from images automatically with corner point detector.

(4) We will study how the loop subdivision and resolution impacts the between class variance (test and base coming from the different classes) and the within variance (test and base coming from the same class) and define signal-to-ratio measure that decides on far one should go into finer resolution.

(5) We will look at the scalability of the approach in the presence of large database and find ways of efficiently search for the best using a divide and conquer hierarchical approach where the observer decides on the specific kind of building s/he is trying to synthesize and recognize so the matching is done against that specific kind of buildings in the database.

(6) Since the idea of this thesis comes from the non-GPS navigating, this algorithm can be implemented on smart phones for navigation and location.

List of References

1. Krissian, K., et al., Model-based detection of tubular structures in 3D images. Computer vision and image understanding, 2000. 80(2): p. 130-171.

2. Clarke, L., et al., MRI segmentation: methods and applications. Magnetic resonance imaging, 1995. 13(3): p. 343-368.

3. Cootes, T.F., G.J. Edwards, and C.J. Taylor, Active appearance models. IEEE Transactions on pattern analysis and machine intelligence, 2001. 23(6): p. 681-685.

4. Heimann, T. and H.-P. Meinzer, Statistical shape models for 3D medical image segmentation: a review. Medical image analysis, 2009. 13(4): p. 543-563.

5. Murphy, M., E. McGovern, and S. Pavia, Historic Building Information Modelling–Adding intelligence to laser and image based surveys of European classical architecture. ISPRS journal of photogrammetry and remote sensing, 2013. 76: p. 89-102.

6. Chevrier, C., et al., Parametric documenting of built heritage: 3D virtual reconstruction of architectural details. International Journal of Architectural Computing, 2010. 8(2): p. 135-150.

7. Overby, J., et al., Automatic 3D building reconstruction from airborne laser scanning and cadastral data using Hough transform. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2004. 34: p. 296-301.

8. Kada, M. and L. McKinley, 3D building reconstruction from LiDAR based on a cell decomposition approach. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2009. 38(Part 3): p. W4.

9. Suveg, I. and G. Vosselman, Reconstruction of 3D building models from aerial images and maps. ISPRS Journal of Photogrammetry and remote sensing, 2004. 58(3): p. 202-224.

10. Bruno, F., et al., From 3D reconstruction to virtual reality: A complete methodology for digital archaeological exhibition. Journal of Cultural Heritage, 2010. 11(1): p. 42-49.

11. Wand, M., et al., Efficient reconstruction of nonrigid shape and motion from real-time 3D scanner data. ACM Transactions on Graphics (TOG), 2009. 28(2): p. 15.

12. Zhang, D. and M. Hebert. Harmonic maps and their applications in surface matching. in Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference On. 1999. IEEE.

13. Johnson, A.E. and M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Transactions on pattern analysis and machine intelligence, 1999. 21(5): p. 433-449.

14. Rocchini, C., et al., Multiple textures stitching and blending on 3D objects, in Rendering Techniques' 99. 1999, Springer. p. 119-130.

15. Henry, P., et al., RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments. The International Journal of Robotics Research, 2012. 31(5): p. 647-663.

16. Higo, T., et al. A hand-held photometric stereo camera for 3-d modeling. in Computer Vision, 2009 IEEE 12th International Conference on. 2009. IEEE.

17. Bernardini, F. and H. Rushmeier. The 3D model acquisition pipeline. in Computer graphics forum. 2002. Wiley Online Library.

18. Schindler, K. and J. Bauer. A model-based method for building reconstruction. in Higher-Level Knowledge in 3D Modeling and Motion Analysis, 2003. HLK 2003. First IEEE International Workshop on. 2003. IEEE.

19. Zhang, W. and J. Kosecka. Localization based on building recognition. in Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on. 2005. IEEE.

20. Lowe, D.G., Distinctive image features from scale-invariant keypoints. International journal of computer vision, 2004. 60(2): p. 91-110.

21. Fritz, G., et al., Building detection from mobile imagery using informative SIFT descriptors. Image Analysis, 2005: p. 284-287.

22. Tomasi, C. and T. Kanade, Shape and motion from image streams under orthography: a factorization method. International Journal of Computer Vision, 1992. 9(2): p. 137-154.

23. Zhang, C. and F.S. Cohen, 3-D face structure extraction and recognition from images using 3-D morphing and distance mapping. IEEE Transactions on Image Processing, 2002. 11(11): p. 1249-1259.

24. Liu, Z. and F.S. Cohen, 3D face reconstruction from image(s) based on gender and ethnicity models, in CGVCVIP2017 2017: Lisbon, Portugal

25. Mundy, J.L. and A. Zisserman, Geometric invariance in computer vision. Vol. 92. 1992: MIT press Cambridge, MA.

26. De Berg, M., et al., Computational Geometry: Introduction. 2008: Springer.

27. Preparata, F.P. and M.I. Shamos, Convex hulls: basic algorithms. Computational geometry, 1985: p. 95-149.

28. Yang, Z. and F.S. Cohen, Image registration and object recognition using affine invariants and convex hulls. IEEE Transactions on Image Processing, 1999. 8(7): p. 934-946.

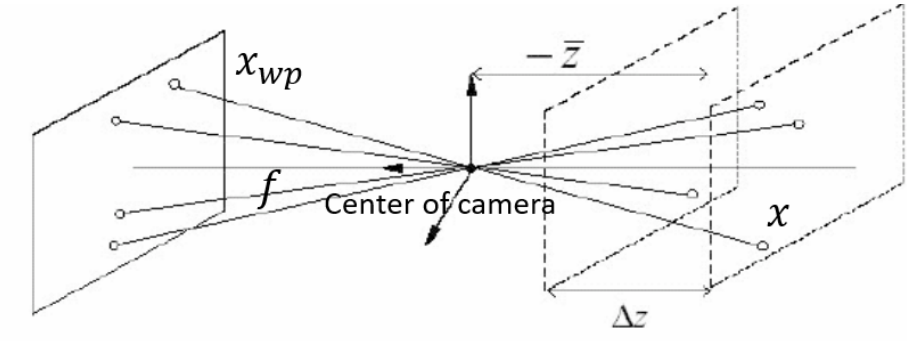
29. Willis A, Sui Y. An algebraic model for fast corner detection, Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009: 2296-2302.

30. Clarkson, K.L. and P.W. Shor, Applications of random sampling in computational geometry, II. Discrete & Computational Geometry, 1989. 4(5): p. 387-421.

APPENDIX A: WEAK PERSPECTIVE

Proposition:

For approximation, if the depth of the object is far more less than the average distance between the object to camera center, the perspective projection can be regarded as weak perspective.



Proof:

Suppose the coordinate of point **X** on object is $\begin{bmatrix} x \\ y \end{bmatrix}$, the depth of the object is ΔZ, the average distance between object and camera center is Z_{av} and the focal length of the pin-hole camera is f.

With perspective projection, the coordinate of $\boldsymbol{X}_p$ on image plane will be:

$$\boldsymbol{X}_p = \left(\frac{f}{Z_{av}+\Delta Z}\right)\begin{bmatrix} x \\ y \end{bmatrix},$$

with Taylor-series expansion:

$$\boldsymbol{X}_p = \left(\frac{f}{Z_{av}+\Delta Z}\right)\begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z_{av}}\left(1 - \frac{\Delta Z}{Z_{av}} + \left(\frac{\Delta Z}{Z_{av}}\right)^2 - \cdots\right)\begin{bmatrix} x \\ y \end{bmatrix},$$

when $|\Delta Z| \ll Z_{av}$ only the zero-order term remains and we get

$$\boldsymbol{X}_{wp} = \left(\frac{f}{Z_{av}}\right)\begin{bmatrix} x \\ y \end{bmatrix},$$

which is the weak perspective.

The error between the point of perspective projection and weak perspective will be:

$$\boldsymbol{X}_{err} = \boldsymbol{X}_p - \boldsymbol{X}_{wp} = -\frac{f}{Z_{av}}\left(\frac{\Delta Z}{Z_{av}+\Delta Z}\right)\begin{bmatrix} x \\ y \end{bmatrix}.$$