Point cloud densification

Mona Forsman

UMEÅ UNIVERSITY
DEPARTMENT OF PHYSICS
SE-901 87 UMEÅ
SWEDEN

Abstract

Several automatic methods exist for creating 3D point clouds extracted from 2D photos. In many cases, the result is a sparse point cloud, unevenly distributed over the scene.

After determining the coordinates of the same point in two images of an object, the 3D position of that point can be calculated using knowledge of camera data and relative orientation.

A model created from a unevenly distributed point clouds may loss detail and precision in the sparse areas. The aim of this thesis is to study methods for densification of point clouds.

This thesis contains a literature study over different methods for extracting matched point pairs, and an implementation of Least Square Template Matching (LSTM) with a set of improvement techniques. The implementation is evaluated on a set of different scenes of various difficulty.

LSTM is implemented by working on a dense grid of points in an image and Wallis filtering is used to enhance contrast. The matched point correspondences are evaluated with parameters from the optimization in order to keep good matches and discard bad ones. The purpose is to find details close to a plane in the images, or on plane-like surfaces.

A set of extensions to LSTM is implemented in the aim of improving the quality of the matched points. The seed points are improved by Transformed Normalized Cross Correlation (TNCC) and Multiple Seed Points (MSP) for the same template, and then tested to see if they converge to the same result. Wallis filtering is used to increase the contrast in the image. The quality of the extracted points are evaluated with respect to correlation with other optimization parameters and comparison of standard deviation in x- and y- direction. If a point is rejected, the option to try again with a larger template size exists, called Adaptive Template Size (ATS).

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Several automatic methods exist for creating 3D point clouds extracted from sets of images. In many cases, they create sparse point clouds which are unevenly distributed over the objects. The task of this thesis is to evaluate, compare and develop routines and theory for densification of 3D point clouds obtained from images.

Point clouds are used in 3D modeling for generation of accurate models of real world items or scenes. If the point cloud is sparse, the detail of the model will suffer as well as the precision of approximated geometric primitives, therefore the subject of densification methods are of interest to study.

1.2 Aims

The aims of this thesis are to evaluate some methods for generation of point clouds and finding possible refinements, that should result in more detailed 3D reconstructions of for example buildings and ground. Some important aspects are speed, robustness, and quality of the output.

The following reconstruction cases are of special interest:

- Some 3D points on a surface, e.g. a wall of a building , have been reconstructed. The goal is to extract more points on the wall to determine intrusions/extrusions from e.g. window frames.

- A sparse 3D point cloud have been automatically reconstructed on the ground. The ground topography is represented as a 2.5D mesh. The goal is to extract more points to obtain a topography of higher resolution.

1.3   Related Work

Several techniques for constructing detailed 3D point clouds exist. In the aim of documenting and reconstructing detailed heritage objects, the papers by El-Hakim et al. [2004], Grün et al. [2004], Remondino et al. [2008] and Remondino et al. [2009], describe reconstruction of detailed models from image data.

Some papers on methods based on video input are found in Gallup et al. [2007] and Frahm et al. [2009].

The papers by D'Apuzzo [2003] and Blostein and Huang [1987] deal with quality evaluation of the generated point clouds.

A prototype of a computer application for photogrammetric reconstruction of textured 3D models of buildings is presented in Fors Nilsson and Grundberg [2009] where the necessity of point cloud densification is noted.

An overview of literature for 3D reconstruction algorithms is done by Börlin and Igasto [2009] and a deeper evaluation of algorithms can be found in Seitz et al. [2006].

1.4   Organization of Thesis

The focus of this thesis is creating dense point clouds of reliable points extracted from digital images. In Chapter 2 theories of photogrammetry, 3D reconstruction and statistics are introduced. Chapter 3 presents an overview of other methods used for point matching, densification of point clouds and related subjects. The implemented method and the implementation with details are described in Chapter 4. A set of experiments designed to evaluate the implemented methods are presented in Chapter 5. The results of the experiments are presented in Chapter 6 followed by discussion and evaluation of the aims in Chapter 7. Finally, Chapter 8 contains acknowledgments.

Chapter 2

Theory

Photogrammetry deals with finding the geometric properties of objects, starting with a set of images of the object. As mentioned in McGlone et al. [2004], the subject of photogrammetry was born in the 1850:s, when the ability to take aerial photographies from hot-air balloons gave inspiration to ideas of techniques to make measurements in aerial photographs in the aim of making maps of forests and ground.

The technique is today used in different applications like computer and robotic vision, see for example Hartley and Zisserman [2003], in creating models of objects and landscapes, and creating models of buildings for simulators and virtual reality.

2.1 The 3D modeling process

The 3D modeling process can be described in various ways depending on methods and aims. The following way of structure is based on Börlin and Igasto [2009]:

1. **Image acquisition** is the task of planning the camera network, take photos, calibration of cameras and rectification of the images. Different kinds of input images require different handling. Some examples are images from single cameras, images from a *stereo rig*, different angles between the camera positions, video data, and combination with laser scanner data of the objects

2. **Feature points** detection in images. Feature points are points that are likely to be detected in corresponding images.

3. **Matching of feature points** is required to know which points are corresponding to each other in the pair of images.

4. **Relative orientation between images** calculates the relative positions of the cameras where the images were taken.

5. **Triangulation** is used to calculate the 3D point corresponding to each pair of matched points.

6. **Co-registration** is done to organize point clouds from different sets in the same coordinate system.

7. **Point cloud densification** is used to find more details and retrieve more points for better estimation of planes and geometry.

8. **Segmentation and structuring** in order to separate different objects in the images.

9. **Texturing** the model with extracted textures from images makes the model photo realistic and complete.

This thesis focus on step 7, in close connection to steps 2 and 3.

2.2 Projective geometry

Projective geometry is an extension to Euclidean geometry to include e.g. *ideal points* that correspond to the intersection of parallel lines. The following introduction of the subject covers the concepts necessary to understand the pinhole camera model and geometrical transformations. The notation of this section follows Hartley and Zisserman [2000].

2.2.1 Homogenous coordinates

A line in the 2D plane determined by the equation

$$ax + by + c = 0$$

can be represented as

$$\mathbf{l} = [a, b, c]^T,$$

that means the line consists of all points $\mathbf{x} = [x, y]^T$ that satisfies the equation $ax + by + c = 0$. In homogenous coordinates the point will be $[x, y, 1]^T$. Two lines $\mathbf{l}$ and $\mathbf{l}'$ intersect in the point $\mathbf{x}$ if the cross product of the lines equals the point,

$$\mathbf{x} = \mathbf{l} \times \mathbf{l}'.$$

In 3D a space point is in the similar way given by

$$\mathbf{p} = [x, y, z, 1]^T$$

and a plane by

$$\mathbf{l} = [a, b, c, d].$$

2.2.2 Transformations of $\mathbb{P}^2$

Transformations of the projective plane $\mathbb{P}^2$ are classified in four classes, *Isometries*, *Similarity transformations*, *Affine transformations* and *Projective transformations*. A transformation is performed using a matrix multiplication of a transformation matrix H and the points $\mathbf{x}$ to transform,

$$\mathbf{x}' = H\mathbf{x}.$$

Figure 2.1 shows the effects of some different transformations.

Isometries

Isometries are the simplest kind of transformations. They consist of a translation and a rotation of the plane, which means that distances and angles are preserved. The transformation is represented by

$$\mathbf{x}' = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x},$$

where R is a 2D rotation matrix including optional mirroring, and $\mathbf{t}$ is a 2×1 vector determining the translation. This transformation has three degrees of freedom, corresponding to rotation angle and translation.

Figure 2.1: Similarity, affine and projective transform of the same pattern.

Similarity transformations

Combining the rotation of an isometry with a scaling factor s gives a similarity transform

$$\mathbf{x}' = \begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x}.$$

A similarity transform preserves angles between lines, the shape of an object and the ratios between distances and areas. This transform has four degrees of freedom.

Affine transformations

An affine transformation combines the similarity transform with a deformation of the plane, which in block matrix form are

$$\mathbf{x}' = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x},$$

where A is a composition of a rotation matrix and a deformation matrix, which is diagonal and contains scaling factors for x and y

$$D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}.$$

A is then composed as $A = R(\theta)R(-\phi)DR(\theta)$. An affine transformation preserve parallel lines, ratios and lengths of parallel line segments and ratios of areas as well as directions in the rotated plane. The affine transformation has six degrees of freedom.

Projective transformations

Projective transformations give perspective views where objects far away is smaller than close ones. The transformation is represented by

$$\mathbf{x}' = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix} \mathbf{x},$$

The vector $\mathbf{v}^T$ determines the transformation of the ideal point where parallel lines intersect. The projective transform has 8 degrees of freedom, only the ratio between the elements in the matrix are fixed. This makes it possible to determine the transform between two planes from four pairs of points.

2.3   The pinhole camera model
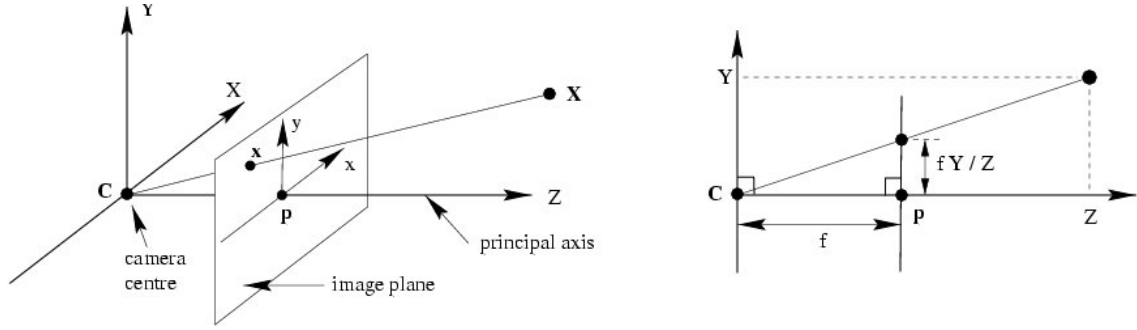
Figure 2.2: Schematic view of a pinhole camera. The image plane is shown in front of the camera centre to simplify the image, in real cameras the image plane, *image sensor*, is behind the centre of the camera.

A simple camera model is the *pinhole camera*. A 3D point $\mathbf{X}$ in world coordinates maps on to the 2D point $\mathbf{x}$ on the *image plane Z* of the camera where the ray between $\mathbf{X}$ and the *camera centre* $\mathbf{C}$ intersects the plane. The *focal distance f* is the distance between the image plane and the camera centre which is the focal point of the lens. Orthogonal to the image plane, the *principal ray* passes through the camera centre along the *principal axis*, originating in the principal point of the image plane. The principal plane is the plane parallel with the image plane through the camera centre. Figure 2.2 shows a schematic view of the pinhole camera model.

The projection $\mathbf{x}$ of a 3D point $\mathbf{X}$ on the image plane of the camera is given by

$$\mathbf{x} = P\mathbf{X},$$

where the camera matrix P is composed by the 3×4 matrix

$$P = KR[I| - \mathbf{C}],$$

The camera matrix describes a camera setup composed by *internal* and *external* camera parameters. The internal parameters are the focal length f of the camera, the principal point $\mathbf{P}$, the resolution m_x, m_y and optional skew s. The focal length and the principal point are converted to pixels using the resolution parameters. $\alpha_x = fm_x, \alpha_y = fm_y$ is the focal length in pixels and $x_0 = m_x P_x, y_0 = m_y P_y$ is the principal point. The internal parameters are stored in the camera calibration matrix

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}. \tag{2.1}$$

The external parameters determine the camera position relative to the world. These are the position of the camera centre $\mathbf{C}$ and the rotation of the camera constructed by a rotation matrix R.

2.4   Stereo view geometry

2.4.1   Epipolar geometry

The relationship between two images of the same object taken from different points of view, is described by the *epipolar geometry* of the images. The two centres of the cameras, $\mathbf{C}$ and $\mathbf{C}'$, spans the baseline, see figure 2.3 (left). Each camera has an epipole, $\mathbf{e}$ and $\mathbf{e}'$, figure 2.3 (right) which is the projection of the other focal point of the second camera on the image plane of the first camera. Every plane determined by an arbitrary point $\mathbf{X}$ and the baseline between $\mathbf{C}\ \mathbf{C}'$ is an epipolar plane. The line of intersection of the image plane and the epipolar plane is called the *epipolar line*. When the projection point $\mathbf{x}$ of a point $\mathbf{X}$ is known in one image, the projection point $\mathbf{x}'$ is restricted to lie on a line through the projection of the camera centre $\mathbf{C}$ and the point $\mathbf{x}$ in the image plane of the second camera. This line intersects the epipole $\mathbf{e}'$.
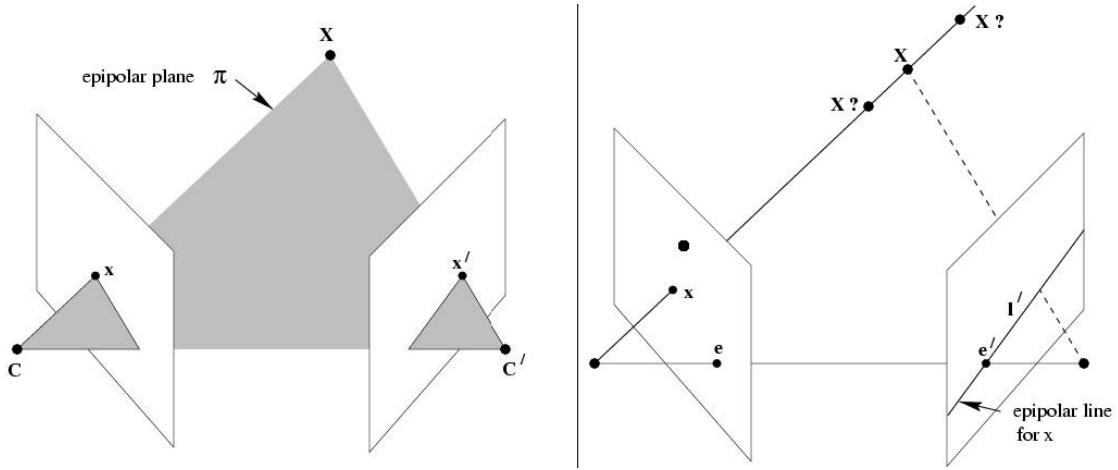
Figure 2.3: The epipolar line connects the the cameras' focal points.

2.4.2   The Fundamental Matrix, F

The fundamental matrix is an algebraic representation of the epipolar geometry. The fundamental matrix F is defined by

$$\mathbf{x}'^{\top} F \mathbf{x} = \mathbf{0}$$

for all corresponding points. The fundamental matrix F is a 3×3 of rank 2. Given at least seven respectively eight pairs of points the fundamental matrix can be calculated using either the seven point algorithm or the eight point algorithm, see Hartley and Zisserman [2000] for details.

The relationship between the fundamental matrix and the camera matrices is given by

$$\mathbf{x} = P\mathbf{X}$$
$$\mathbf{x}' = P'\mathbf{X}$$

$$F = [\mathbf{e}']_{\times} P' P^{+}$$

where $[\mathbf{e}']_{\times}$ is the representation matrix for transforming a cross product to a matrix-vector multiplication and P^{+} is the pseudoinverse of the matrix P.

2.4.3   Triangulation

When the camera matrices are calculated, and the coordinates for a point correspondence are known, the 3D point can be calculated by solving the equation system

$$\mathbf{x} = P\mathbf{X}$$
$$\mathbf{x}' = P'\mathbf{X}$$

for $\mathbf{X}$.

2.4.4   Image rectification

The lens in a camera causes some distortion in the images, making straight lines in the outskirts of the image projected curved, because of the mapping of a 3D world onto a 2D sensor through a spherical lens. This error in the images can be reduced by rectification of the image. In this work, only pre-rectified images are used. Figure 2.4 illustrates the effect of lens distortion and rectifying. The topics of image rectification and lens distortion are throughly explained in Hartley and Zisserman [2000] and Remondino [2006]
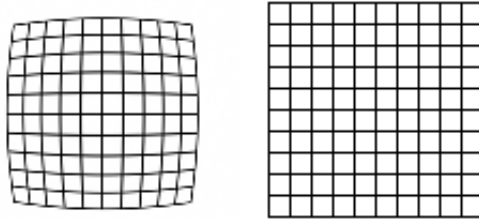
Figure 2.4: The grid to the left is curved as a lens distorted image, the right image is the rectified grid.

2.5   Estimation

This section follows mostly the notation from Montgomery et al. [2004].

2.5.1   Statistics

Origins of errors

In tasks where measurements are done there usually occur some errors. The quality of measurements are affected by systematic errors (bias) and unstructured errors (variance).

In photogrammetry usual origins of errors are the camera calibration, the quality of point extraction, the quality of the model function and numeric errors in triangulation and optimization.

Normal and χ^2 distribution

The *Normal distribution* describes the way many random errors affects the results. The most probable value is close to the expected value μ. Few values are far away. The standard deviation σ (and variance) describes the dispersion of values. The distribution of a normal random variable is defined by the probability density function $N(\mu, \sigma^2)$, giving

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \text{ for } -\infty < x < \infty,$$

where μ is the expectation value of the distribution and σ^2 is the variance.

Figure 2.5: Normal distribution with expectation value $\mu = 5$ and variance $\sigma^2 = 10$.

The χ^2 distribution is defined by

$$p_y(y, n) = \frac{y^{(n/2-1)}e^{-y/2}}{2^{n/2}\Gamma(\frac{n}{2})} \qquad n \in \mathcal{N}, \ y > 0,$$

where $\Gamma(.)$ is the Gamma-function. A particular case is the sum of squared independent random variables, $z_i \sim N(0, 1)$

$$y = \sum_{i=1}^{n} z_i^2$$

which is χ^2 distributed, [Förstner and Wrobel, 2004].

Variance and standard deviation
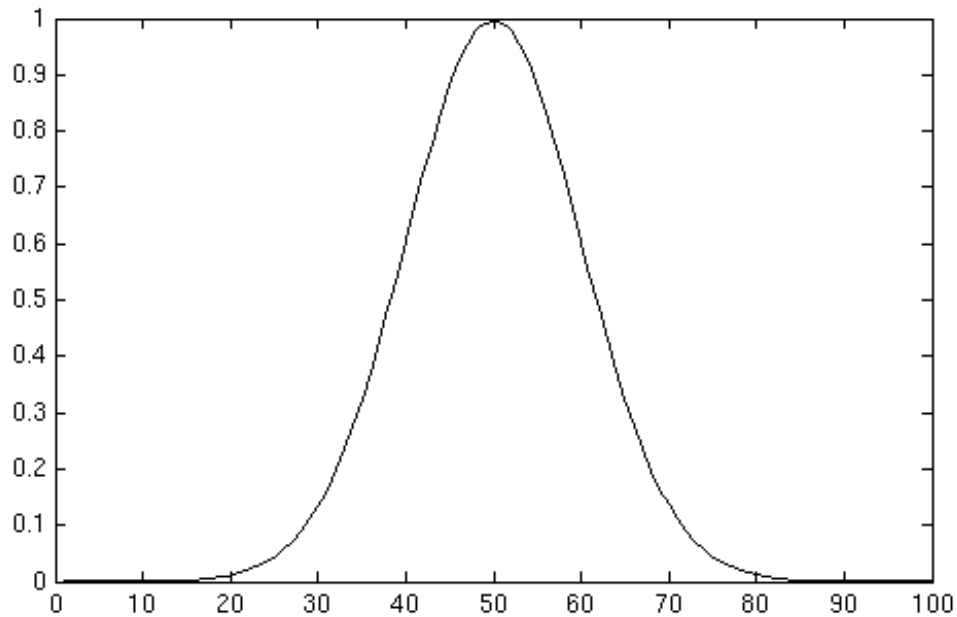
The variance is a measure of the width of a distribution. It is defined as

$$\sigma^2 = \int_{-\infty}^{\infty} x^2 f(x)dx - \mu^2,$$

where $f(x)$ is the probability density function of the distribution and μ is the expected value. The standard deviation is σ, the square root of the variance.

Covariance

Covariance is a measure of how two variables interact with each other. A covariance of zero implies that the variables are uncorrelated. Of two variables x and y with estimation values $E(x)$ and $E(y)$ and mean values μ_x and μ_y the covariance is

$$Cov(x, y) = E(xy) - (\mu_x \mu_y),$$

Correlation coefficients

The correlation coefficient is the normalized covariance and determines the strength of the linear relationship between the variables. The correlation coefficient is determined by

$$\rho_{xy} = \frac{Cov(x, y)}{\sqrt{(\sigma_x^2 \sigma_y^2)}},$$

where S_{xy} is called the corrected sum of cross products, defined by

$$S_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}).$$

Correlated and non-correlated errors

A positive correlation coefficient between **x** and **y** implies that given a small value of x, a small value of y is likely. If the coefficient is zero, there is no linear relationship. If the observations are plotted, $\rho = 0$ if the plotted points are equally distributed. If they are close to a line in positive direction, ρ is close to 1.
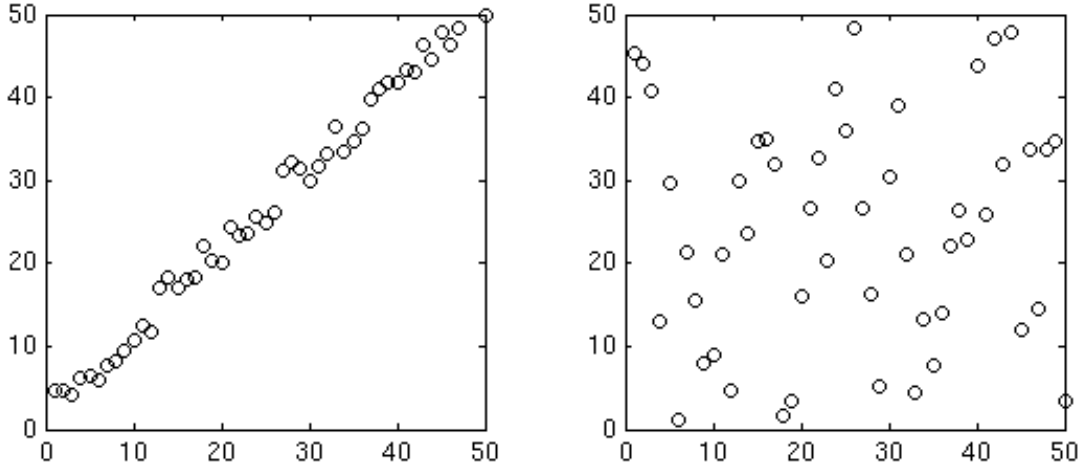
Figure 2.6: Values in the left image are correlated with a correlation coefficient close to 1. Values in the right image are not correlated, and hence the correlation coefficient is close to 0.

The covariance matrix

The covariance matrix is composed by the variance of each variable and their covariances.

$$C = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{bmatrix}$$

Error propagation of linear combinations of random variables

As presented in Förstner and Wrobel [2004, ch.2.2.1.7.3] a set of n normally distributed random variables with covariance matrix $\mathbf{C}_{xx}$ can compose a vector

$$\mathbf{x} = [x_1, x_2, \ldots x_n]^T, \mathbf{x} \in N(\mu_x, \mathbf{C}_{xx})$$

A linear transformation of the vector $\mathbf{x}$ is defined by

$$\mathbf{y} = \mathbf{Mx} + \mathbf{p}.$$

The expected value and the variance transform are defined as follows:

$$E(\mathbf{Y}) = E(\mathbf{Mx} + \mathbf{p}) = \mathbf{M}E(\mathbf{x}) + \mathbf{p} = \mathbf{M}\mu_x + \mathbf{p}$$

$$\sigma^2 = V(\mathbf{y}) = V(\mathbf{Mx} + \mathbf{p}) = \mathbf{M}V(\mathbf{x})\mathbf{M}^T + V(\mathbf{p}) = \mathbf{M}\mathbf{C}_{xx}\mathbf{M}^T.$$

2.5.2   Optimization

Optimization has a set of different applications in photogrammetry. One is in template matching, which is the interesting area for this work, and one is in Bundle adjustment where a model and a point cloud are adjusted to each other. The task of least squares optimization is to minimize the norm $||r(x)||$ of the residual $\mathbf{r}(\mathbf{x})$ between a model function $f(x)$ and the observations $\mathbf{b}$. If the model is linear, the residual becomes

$$\mathbf{r}(\mathbf{x}) = A\mathbf{x} - \mathbf{b}.$$

where A is a constant matrix.

Many problems belong to the class least squares problem which can be solved using different algorithms. The choice of optimization algorithm is a question of efficiency in time and memory, precision, probability to converge and implementation difficulty. Some methods, as Levenberg-Marquardt and Gauss-Newton, [Nocedal and Wright, 1999, ch.10.3] linearize the problem and solve it using linear optimization techniques.

Weighted optimization

If the covariances of the variables are known and not zero, the problem is considered weighted and the optimization problem can be formulated as

$$\min_{\mathbf{x}} ||\mathbf{r}(\mathbf{x})||_{\mathbf{W}}^2 = \min_{\mathbf{x}} ||\mathbf{A}\mathbf{x} - \mathbf{b}||_{\mathbf{W}}^2,$$

The index $\mathbf{W}$ indicates that the norm $||.||_{\mathbf{W}}^2$ is weighted and defined as

$$||\mathbf{x}||_{\mathbf{W}}^2 = \mathbf{x}^T\mathbf{W}\mathbf{x}.$$

The weight matrix W is defined by

$$\mathbf{W} = \mathbf{C}_{bb}^{-1},$$

where $\mathbf{C}_{bb}$ is the covariance matrix of the observations. The structure of the covariances may be enough, and then the covariance matrix can be decomposed to

$$\mathbf{C_{bb}} = \sigma_0^2 \mathbf{Q_{bb}}$$

where σ_0^2 is a scaling parameter and $\mathbf{Q_{bb}}$ is a structure parameter.

2.5.3 Rank N-1 approximation

Rank N-1 transform is often, for historical reasons, called *Direct Linear Transform*. In this work, the aim of the rank N-1 transform is to find the homography $H : \mathcal{P}^2 \rightarrow \mathcal{P}^2$ from a set of point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ in a pair of images. The homography can be exactly determined from four point correspondences or estimated from a larger amount of points using the *Singular Value Decomposition* (SVD), described in e.g. Strang [2003].

The transformation is given by

$$\mathbf{x}_i' = H\mathbf{x}_i,$$

where H is a 3×3-matrix. $\mathbf{x}_i'$ and $H\mathbf{x}_i$ are parallel in $\mathcal{R}^3$ giving $\mathbf{x}_i' \times H\mathbf{x}_i = \mathbf{0}$. Rewriting the system, using $h^1 = [h_{11}h_{12}h_{13}]^T$ and $\mathbf{x}_i' = (x_i', y_i', w_i')^T$, gives in matrix form

$$\begin{bmatrix} \mathbf{0}^T & -w_i'\mathbf{x}_i^T & y_i'\mathbf{x}_i^T \\ w_i'\mathbf{x}_i^T & \mathbf{0}^T & -x_i'\mathbf{x}_i^T \\ -y_i'\mathbf{x}_i^T & x_i'\mathbf{x}_i^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{bmatrix} = \mathbf{0},$$

also written as $A_i'\mathbf{h} = \mathbf{0}$. This system has only two linearly independent rows, which makes it possible to remove one row. By assembling the 2×9 equation for all known points, a $2n \times 9$ system is built. This gives an over-determined equation system to solve, and according to noise, it will probably not have an unique solution. Instead, the optimization problem

$$\min_{h} ||A\mathbf{h}||$$

$$\text{s.t.} ||\mathbf{h}|| = 1.$$

is solved.

This can be solved by using the SVD of A

$$A = UDV^T.$$

The solution is found in the right singular vector $\mathbf{v}_n$

This transformation matrix P is usually ill-conditioned, because the centres of gravity of the set of points are far from zero. Making a *normalization* of the points, as described in Hartley and Zisserman [2003, ch.4.1], gives a less perturbation sensitive system, which implies smaller variance in the transformed points. By transforming the points to have their centre of gravity at the origin and mean distance $\sqrt{2}$ to the origin, before applying the Rank N-1 transform, the system becomes becomes normalized.

2.6 Least Squares Template Matching

Least Squares Template Matching (LSTM), presented by Gruen [1996], searches for the best position for a template in a search patch. The method is closely related to Adaptive Least Squares Matching (ALSM) described in Gruen [1985]. The template and the search patch are described by two discrete two-dimensional functions, $f(x, y)$ and $g(x, y)$. A noise function $e(x, y)$ contains the difference between the image functions

$$f(x, y) - e(x, y) = g(x, y).$$

The aim of the optimization is to reduce the noise function $e(x, y)$. The template function $f(x, y)$ is transformed by an affine transformation approximating the projective difference between the images to match the search patch function $g(x, y)$. Optionally, the transformation is combined with radiometric parameters to compensate for lighting differences. The optimization parameters are combined in a vector $\mathbf{x}_o$, based on the homography matrix H. The Gauss Newton optimization is then applied on the elements of $\mathbf{x}$ to minimize the resulting noise.

$$\min_{s.t.\mathbf{x}} e(\mathbf{x}) = \min_{s.t.\mathbf{x}} f(\mathbf{x}) - g(\mathbf{x})$$

The output from LSTM is primary the position of the template in the both images. Optional, the implementations also return the results of the other optimization parameters, the step lengths used by Gauss Newton and statistics.

Chapter 3

Overview of methods for densification

3.1 Introduction

There are many different methods used in the aim of improving the density of point clouds, with different pros and cons. For image acquisition video data or still images can be used, sometimes in combination with laser scanner data of the object. Both feature based methods and area based methods are used in various implementation. Much of the work with densification is done on small objects with symmetrical camera network all around the object.

The task of finding general methods for densification is still an open task.

3.2 Various kinds of input

3.2.1 Video data

Working with video data has the advantage of very short base lines between sequential image, and with knowledge of the camera motion and calculated vanishing lines, which are vertical, an up-vector can be determined. A disadvantage of this kind of input data is the lower resolution of most video cameras which probably gives less details, and also to handle the large amount of data created from the short base line images to handle. In Gallup et al. [2007] and Frahm et al. [2009] a system for 3D-reconstruction of architectural scenes is presented. The input data used are uncalibrated video data. For faster handling of the data the GPU is used for calculations. The introduction of the Viewpoint Invariant Patch (VIP) gives a descriptor of features looking similar from a variety of directions.

3.2.2 Laser scanner data

A laser scanner device can make accurate measurements of the distance to points in an object. Combining laser scanner data from different points of view gives the opportunity to create detailed model of the hull of an object. In Wendt [2007] laser scanner data are combined with still images by describing features in both data sets and then matching these together. That makes it possible to create accurate textured models. However, a laser scanning device is expensive, and not commonly available.

3.2.3   Still images

For still images, a lot of different methods are used. There are differences in methods used if the input are single images or stereo images, if the base lines are short or long, whether the camera positions are known or not. As mentioned in Remondino [2006] the camera network is important to improve the precision of the reconstructed model. Knowledge of the requirements of the reconstruction method are important to decide the choice of cameras used, their positions and their directions.

3.3   Matching

There are two main classes of methods for matching of images, feature based methods and area based methods, in some cases combined with each other to improve results. Here are some different approaches briefly presented.

3.3.1   SIFT, Scale-Invariant Feature Transform

SIFT is a widely used feature based method for object recognition and was first presented by Lowe [1999]. It uses a class of local image features which are invariant to image scaling, translation and rotation, and partially invariant to changes in lighting and projection.

3.3.2   Maximum Stable Extremal Regions

Maximum Stable Extremal Regions (MSER) is a method for feature detection proposed by Matas et al. [2002] where regions which are little affected by various points of view are chosen as feature points. Wide baseline image pairs are possible to use for point matching using this method

3.3.3   Distinctive Similarity Measure

In Yoon and Kweon [2008] a method to measure the distinguishable of a feature point is presented. A point which is unique in the image is treated as more valuable than a point that is similar to many others.

3.3.4   Multi-View Stereo reconstruction algorithms

A family of algorithms useful for high precision reconstruction of small objects. The requirements of dense camera networks in exact positions makes them not suitable for outdoor work, but they are used for detailed reconstructions of smaller objects. An evaluation of a set of these algorithms is done in Seitz et al. [2006].

3.4   Quality of matches

A denser point cloud is not useful if the matched points are of low quality due to wrong matchings or low precision. D'Apuzzo [2003] suggests a couple of measurements to detect the quality of a point match. Some of the values suggested to use are the σ_0 from template matching, the differences shifts in x and y- direction, the scale factors in x and y and step lengths used by the optimization.

Chapter 4

Implementation

The objective for this thesis is template matching in the purpose of point cloud densification. In this section, an implementation of least squares template matching is described, combined with a set of different methods for refinements in preprocessing of the images, analysis of resulting parameters and successive improvement.

Four corners in one image and a homography between the particular plane to match in the pair of images is given to the method. The images have known epipolar geometry and camera positions.

The implemented methods shall be evaluated with respect to

- Point cloud quality:

  - Completeness — what is the density of the generated point cloud?
  - Robustness — how many matches are correct?
  - Precision — what is the reconstruction error for the correct matches?

- Method sensitivity to:

  - Object geometry — how large deviations from the basic shape can be reconstructed?
  - Camera network geometry — how well do the methods work with a long baseline, i.e. the images were taken far apart?

4.1   Method

The choice of method is Least Squares Template Matching (LSTM) as described in ch. 2.6. A dense grid of seed points is constructed for the area of interest in the left image. The grid is transformed to the right image using the homography to generate initial guesses for the points. This gives a more evenly distributed set of seed points than feature-based methods usually generates. The density of points is possible to adjust by changing the grid.

A number of extensions to LSTM have been implemented. These are; Wallis Filtering for contrast enhancement, Transformed Normalized Cross Correlation (TNCC) to improve the initial pairs of seed points, Multiple Seed Points (MSP) to ensure stability of the match and Adaptive Template Size (ATS) to detect matches where the scale of gradients in the image is too large for the first one used. The matches found are evaluated in respect to matching parameters to detect and reject possible false matchings.

4.2   Implementation details

4.2.1   Algorithm overview

1. Create homography $\mathbf{H}$ for a plane $\mathbf{P}$ representing a wall, ground or equivalent surface in the set of images.

2. Place a rectangular grid over the area in the left image for point generation as in figure 4.1.

3. Create initial seed points by transforming the grid points with the homography to the right image as in figure 4.2.

4. Optional: Improve the contrast by Wallis Filtering (section 4.2.3)

5. For each grid point:

   (a) Optional: Improve the seed point by Transformed Normalized Cross Correlation, see section 4.2.4

      – If the maximum value of the cross correlation is too low, set an error code.

   (b) Cut out a template centered in the grid point of the left image and a search patch centered in the improved seed point in the right image which is three times larger than the template, see figure 4.3.

   (c) Optional: If Multiple seed points is used, put eight extra seed points around the grid point in the left image.

      – Perform Least Squares Template Matching for all nine points.
      – Choose coordinates the most start points converge to as the found point. If less than three converges to the same point, restart from the normalized cross correlation with a larger template.
      – If less than four seed points converge to the same point, set an error code.

   (d) If Multiple seed points is not used, perform least squares template matching on the point.

   (e) Calculate the covariance matrix for the found point. If it not passes the analysis, set an error code.

6. Calculate 3D points for the accepted point pairs using calibration data for the cameras.

4.2.2   Adaptive Template Size (ATS)

A small template size of seven pixels is used initially, and the size grows up to a maximum size of 31 pixels in steps of four if no point is found using smaller templates. For the least squares template matching to converge to the right point it requires that the point is inside the area overlapped by the template. If adaptive template size is used, a check for error codes is evaluated when step 5 in the algorithm is finished for each point. If an error code is found the step is re-executed using a larger template size until an error-free run is done, or maximum template size is reached.

4.2.3 Wallis filtering

Wallis filtering is a method for contrast enhancement. The filter function is represented by

$$J_{i,j} = \alpha\mu + \frac{(1-\alpha)(I_{i,j} - \mu)}{\sigma}$$

where $J_{i,j}$ is the new value of the processed pixel in the image, $I_{i,j}$ is the old value, α is a blending parameter, μ is the mean intensity of the pixels in the filter window and σ is the standard deviation. If α is close to 1one, the Wallis filter is an averaging filter and if α is close to zero it normalizes the intensity of the pixels.

This process is time consuming, so it is wise to only filter the interesting parts on an image. In images 4.4 and 4.5, a grass area is displayed in gray scale and its Wallis filtered version.

4.2.4 Transformed Normalized Cross Correlation (TNCC)

The normalized cross correlation is used to improve the initial seed point created by transforming the grid point with the homography. Compared to classical Normalized Cross Correlation, described by Gonzalez and Woods [2008], this process pretransforms the image by the homography.

The procedure is as follows:

– Take a template from the left image centered in the grid point.

– Find the corners of an area three times as large as the template centered in the same point in the left image.

– Transform the corner points using the homography to the right image.

– Pick the rectangular area including the corner points and transform the sub image with `imtransform` to reshape it as the left image.

– Find the best starting point with `normxcorr2` for the template in the transformed sub image.

– Transform the coordinates for the best point to absolute coordinates in the right image. This is the new seed point.

– If the maximum cross correlation value is to low, try with larger template size (and larger search patch).

Image 4.6 shows the Normalized cross correlation. This improvement is expensive in terms of execution time, especially if the template sizes are large, and is not recommended to use in combination with Adaptive Template Size.

4.2.5 Multiple Seed Points (MSP)

Eight extra seed points are generated around the seed point improved by normalized cross correlation. Least squares template matching is applied on all seed points. If at least three of them converges to within two pixels, this point is accepted as the result. Figure 4.7 shows a template, the multiple seed points and the different points of convergence.

4.2.6   Acceptance criteria

The question of which points should be accepted has many different answers. Higher precision requirements give less dense point clouds but higher precision in the matched points. The chosen criteria are correlation between optimization parameters, the scale adjustment s_x, s_y in x- and y- direction and the result from the optimization. In this implementation a correlation value larger than 0.80 implies rejection as well as a quotient of x- and y- scale factors larger than two.

4.2.7   Error codes

A set of error codes is implemented to prepare for evaluation of why points are rejected. The codes are

1. The max value from Normalized cross correlation was lower than 0.8.

2. Less than 4 of multiple seed points converged to the same result.

3. Maximum adaptive template size reached without any accepted point found.

4. The optimization did not converge.

5. The scale adjustment quotient s_x/s_y is too high (over 2) or too low (less than 0.5).

6. Too high correlation between position and any other optimization parameter (over 0.80).

7. Too high standard deviation σ_0 from the optimization.

An accepted point has the code 0. Codes 1-3 appears only when their respective method is applied. Code 3 for maximum adaptive template size overrides the other codes.

Figure 4.1: A 10×10 grid of points.

Figure 4.2: Seed points generated by transforming a 10×10 grid of points with the homography H.
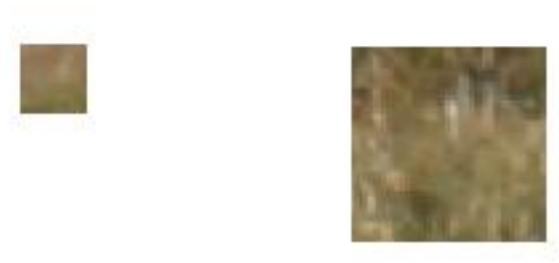
Figure 4.3: Left image is an example of a template 15×15 pixels, right image is the corresponding search patch.
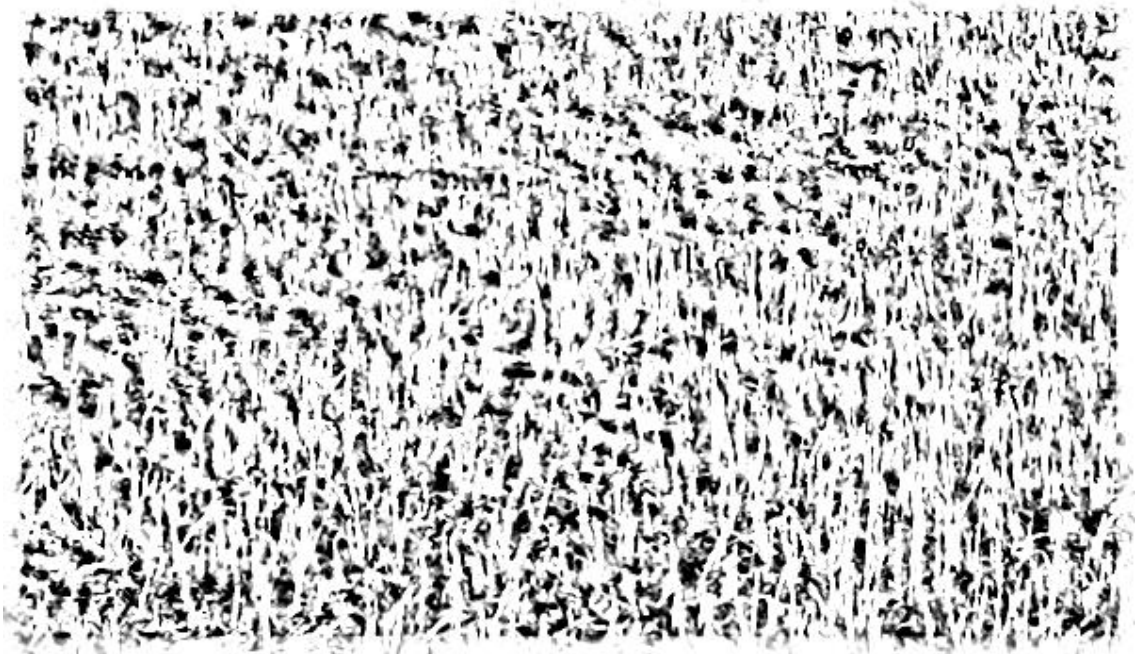
Figure 4.4: Grass without Wallis filtering

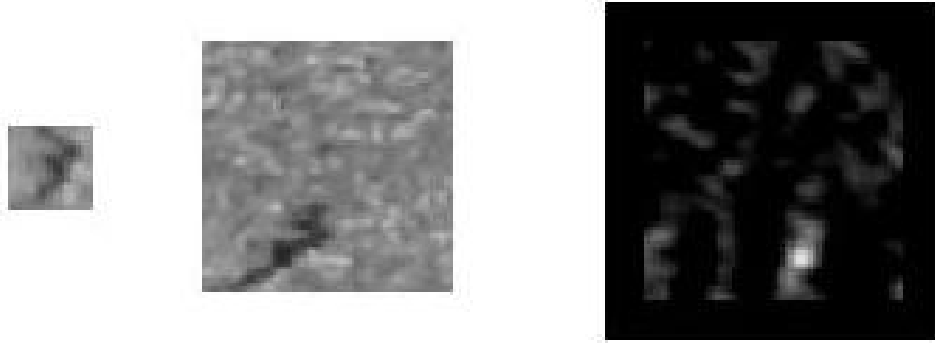Figure 4.5: Grass with Wallis filtering for contrast enhancement

Figure 4.6: Normalized cross correlation of a template and the corresponding search patch in area A.5. The left image is the template of 21×21 pixels, the middle image is the search patch, and the right image is the resulting cross correlation where light pixel implies high correlation.
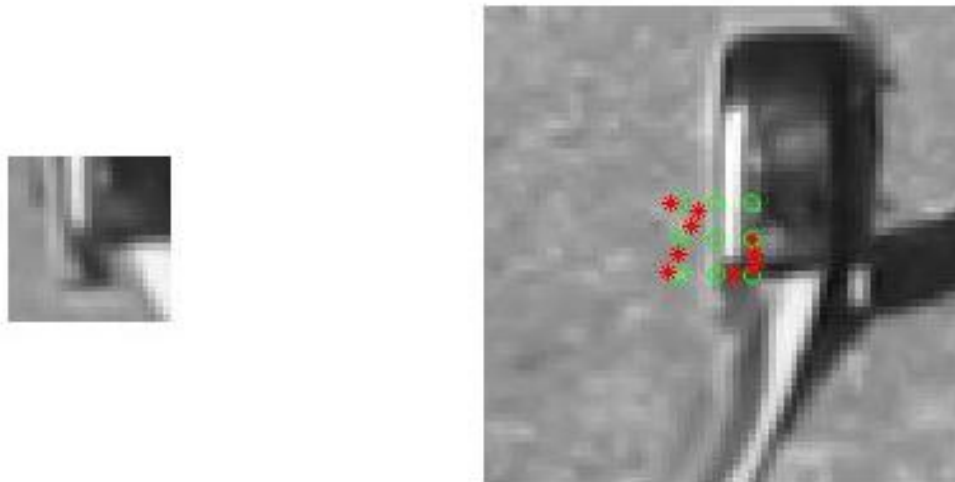

Figure 4.7: Example of a search patch for multiple seed points in area A.5. Green circles are seed points, red stars are the found points.

4.3 Choice of template size

To estimate appropriate template size, calculations are done to determine the need for precision in point coordinates and required template size.

4.3.1 Calculation of z-coordinate from perturbed input data

The depth of the signs on the wall in area 2 in image 5.3 on page 30 is estimated to 0.2 meters. Taking a point in the 3D cloud of this image set as an example,

$$X = [2.85, 2.68, -39.02]$$

is projected in camera 1 (left image) on the point

$$\mathbf{x} = [2203.3, 1101.6]^T$$

using the camera equation (eq.2.1). A shift of the z-coordinate of 0.2 meters gives the 2D point

$$\mathbf{x}' = [2204.5, 1100.5]^T.$$

The difference in projection on the image plane is

$$(\Delta_x, \Delta_y) = [-1.2, 1.1]^T$$

pixels.

This tells us that a 0.2 meter detail is projected less than two pixels from its corresponding point in the plane. Least Squares Template Matching requires that the true point is within a half template size from the seed point, which is fulfilled by a template size of seven pixels.

Chapter 5

Experiments

A number of experiments were designed to investigate the properties of LSTM and the extensions described in chapter 4. In this chapter the examples are described, followed by their results in chapter 6.

The tests were run under MATLAB 7.9.0 (R2009b) on Intel Core2 Quad CPU Q9300 2.500GHz with 4096 MB memory and Intel Core2 Quad Q9400 2.66GHz with 4096 MB memory. The second type has about 5% higher performance.

5.1 Image sets

Three image sets were used, image set A - the loading dock of the MIT building, figures 5.1 and 5.2, image set B - the building "Sliperiet" at Strömpilen , figures 5.3 and 5.4, and image set C - the building "Elgiganten" at Strömpilen, figures 5.5 and 5.6. These different areas have been selected for evaluation based on their different properties and structures.

The experiments are set up in the purpose of evaluating how the different extensions applied to LSTM works on image parts with different properties. For every area, a 100×100 grid of seed points were distributed to create a dense set of possible point correspondances.

The results are presented in tables with comments and figures in the following results chapter where the amount of accepted point matchings and the execution times are presented for all experiments, and other properties where they are of interest.

5.1.1 Image pair A, the loading dock

This set of images contains planes in different viewing angles and different types of texture.

The used areas are the following (numbered as in figure 5.1 on page 29 and figure 5.2 on page 29):

A.1 A brick wall at an oblique angle in shadow.

A.2 A brick wall from almost orthogonal angle in direct sunlight.

A.3 A painted door with a window and some signs, very dark area of the image.

A.4 Brick wall from almost orthogonal angle in shadow.

A.5 Asphalt in skew angle with a lot of random structure.

27

5.1.2   Image pair B, "Sliperiet"

This set of images contains three planes on different distances with signs and windows protruding from the main wall. There is also an asphalt plane at a skew angle.

The used areas are the following (numbered as in images 5.3 on page 30 and 5.4 on page 30):

B.1 Plastered wall with doors, windows, including cracks in the plaster (not used in final version).

B.2 Plastered wall with windows, signs and obstacles at the bottom.

B.3 Plastered wall with some fine cracks in the plaster and windows (not used in final version).

B.4 Asphalt in skew angle with a lot of random structure.

5.1.3   Image pair B, "Elgiganten"

This set of images contains a grass area, and two wall areas of different structure.

The used areas are the following (numbered as in images 5.5 and 5.6):

C.1 Grass in skew angle with a lot of random structure.

C.2 Wall of corrugated plate with mostly horizontal gradients.

C.3 Sign with a lot of strong gradients combined with gradient free areas.
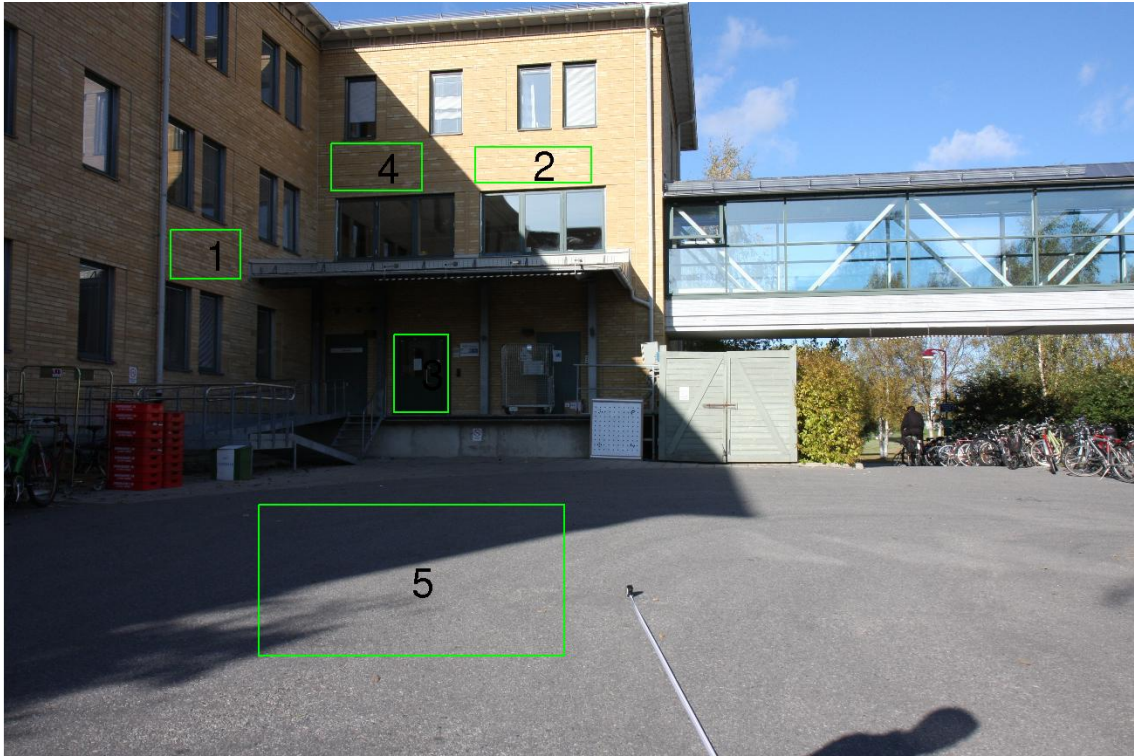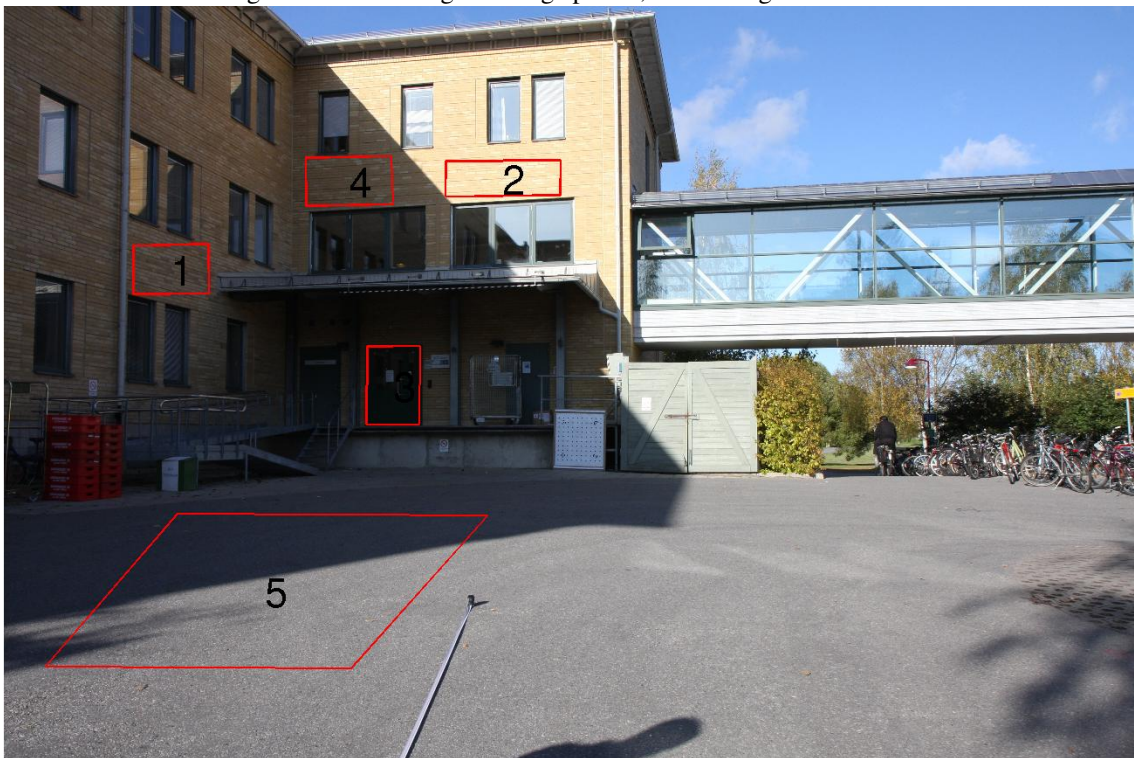
Figure 5.1: Left image of image pair A, the loading dock.

Figure 5.2: Right image of image pair A, the loading dock.
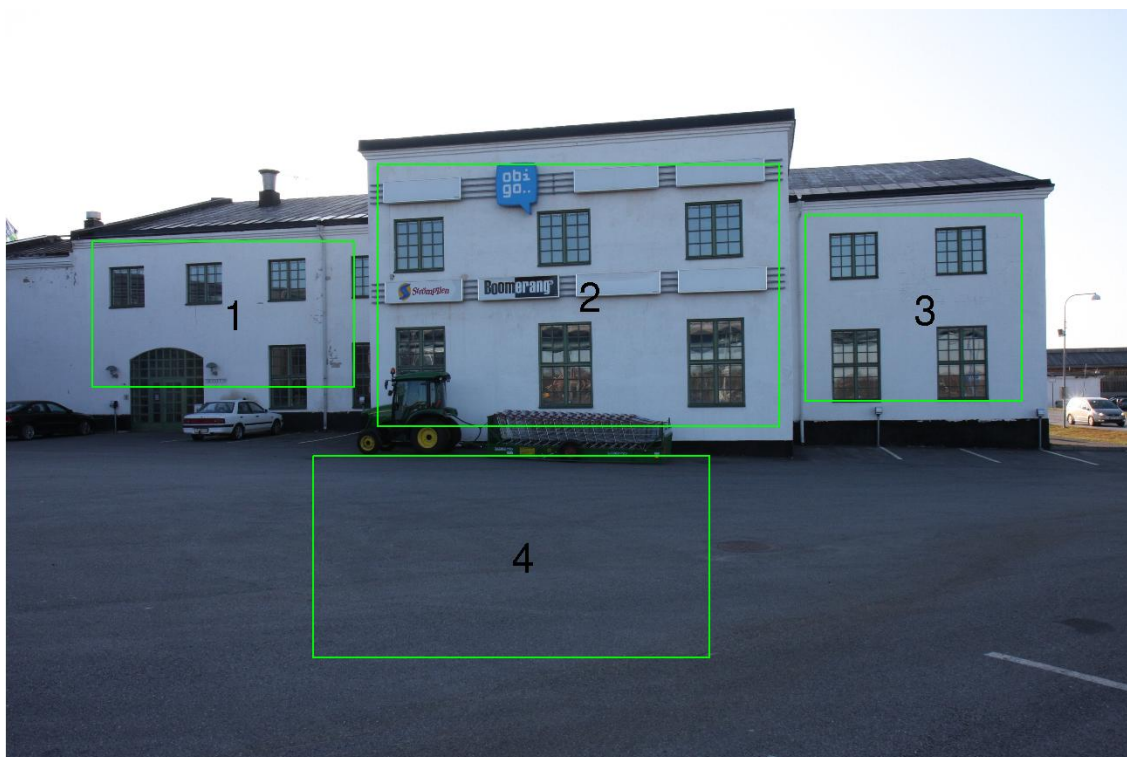
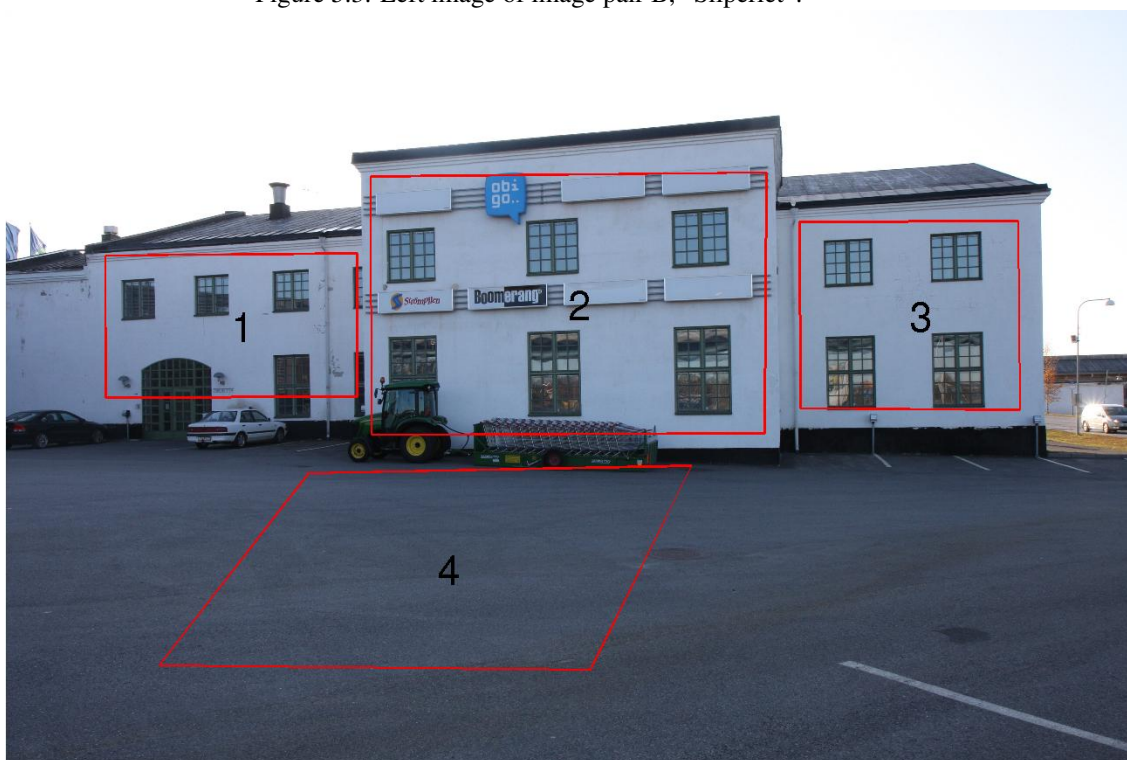Figure 5.3: Left image of image pair B, "Sliperiet".

Figure 5.4: Right image of image pair B, "Sliperiet".

Figure 5.5: Left image of image pair C, "Elgiganten".

Figure 5.6: Right image of image pair C, "Elgiganten".

5.2 Experiments

5.2.1 Experiment 1, Asphalt

Two asphalt areas were chosen to study the effects of different lightning in the same kind of area, the effect of a gradient from a shadow, and how multiple seed points works in an area with a lot of small, irregular structure. The chosen areas are area A.5 and area B.4. For both areas, TNCC and Wallis filtering are used, and in the second run also combined with MSP. For reference, a run without extensions is done.

5.2.2 Experiment 2, Brick walls

Three brick wall areas are studied to evaluate effects of skew in similar structure and different light settings. Two of them were facing the camera almost directly, one in direct sunlight and the other one in shadow. The third one was orthogonal to the other two and was in shadow. The areas used are in image A, area A.1, area A.2 and area A.4. The extension MSP was compared to baseline LSTM.

5.2.3 Experiment 3, Door

The door in image area A.3 is very dark and contain little information detectable for the human eye. This area is studied to see if LSTM is capable of matching under these circumstances. In this case, the effects of adaptive patch size are analyzed and the patch sizes used are presented. The area used is area A.3. The first run for reference is using TNCC and Wallis filtering, to the second run ATS is added.

5.2.4 Experiment 4, Lawn

The grass area C.1, is chosen to study the effects of Wallis filtering on an image with irregular gradients and little contrast. Two runs are performed, one baseline LSTM, and one with Wallis filtering.

5.2.5 Experiment 5, Corrugated plate

Area C.2 consists of a facade of corrugated plate with horizontal gradients but very few vertical gradients. This area is studied to determine if it is possible to detect points in this kind of area with the help of Wallis filtering or/and ATS. As comparison, area C.3 is used which contains signs with irregular, man-made gradients. The methods to test are baseline LSTM, only Wallis, and Wallis combined with ATS.

Chapter 6

Results

6.1 Experiments

In this section the results from the experiments are presented in detail.

6.1.1 Asphalt

The asphalt area contains lots of small irregular gradients as seen in image 6.2. In the Wallis filtered image the gradient from the shadow is distinct as well as parts of the others, see figure 6.3, but some smaller gradients are suppressed. In this experiment, an outlier is defined as a point whose 3D coordinate is more than 5 cm away from a manually determined plane in the point cloud. As seen in image 6.1, the combination of TNCC and Wallis only detects points along the gradient conjured by the shadow. The effect of Wallis filtering of asphalt is shown in image 4.6.

Table 6.1: Example 1, Area A.4 and area B.4. No improvements vs. TNCC + Wallis vs. TNCC + Wallis + MSP vs. Wallis vs. Wallis + MSP.

Area + method	A.5 + 0	A.5 + cw	A.5 + cwm	A.5 + w	A.5 + wm
Number of accepted points	5601	79	79	6363	6423
Runtime (s)	184.3	7646	9412	335.7	474.4
Median residuals	0.01	0.009	0.008	0.006	0.006
Std.deviation	0.015	0.01	0.012	0.012	0.012
No. outliers	30	0	0	0	0

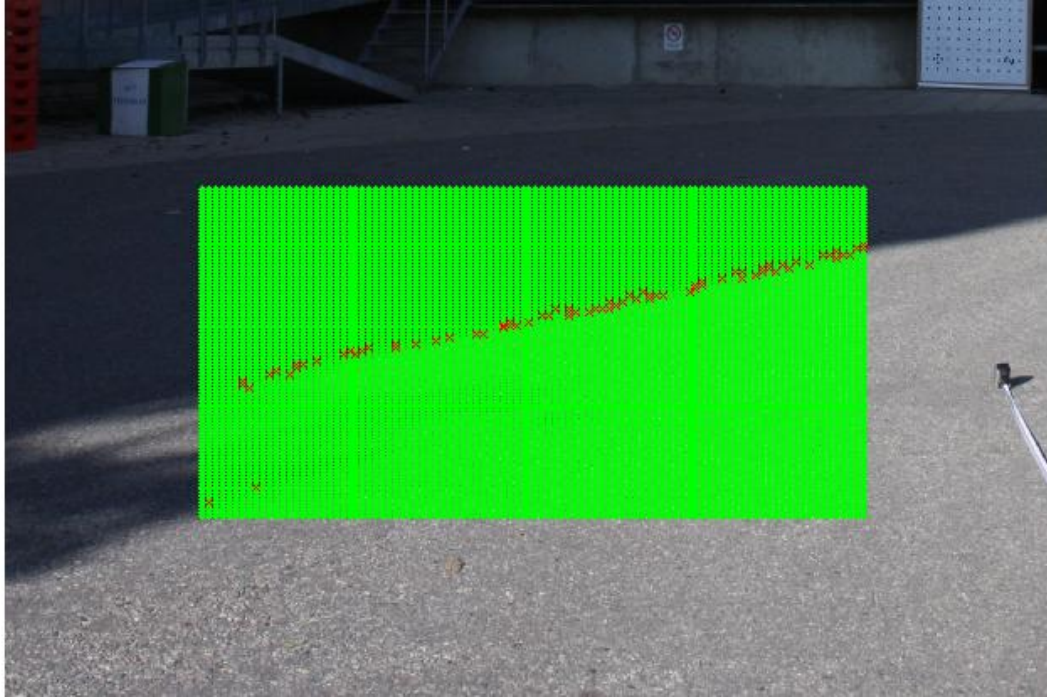| Area + method	B.4 + 0	B.4 + cw	B.4 + cwm	B.4 + w	B.4 + wm
Number of accepted points	5481	79	79	4922	4948
Runtime (s)	166.78	7850	9623	484.2	566.3
Residuals	0.15	0.21	0.20	0.12	0.15
Std.deviation	0.20	0.24	0.20	0.19	0.19
No. outliers	4690	6	15	4183	4205

Figure 6.1: Red points are points detected in area A.5 with TNCC and Wallis filtering, green points are seed points.
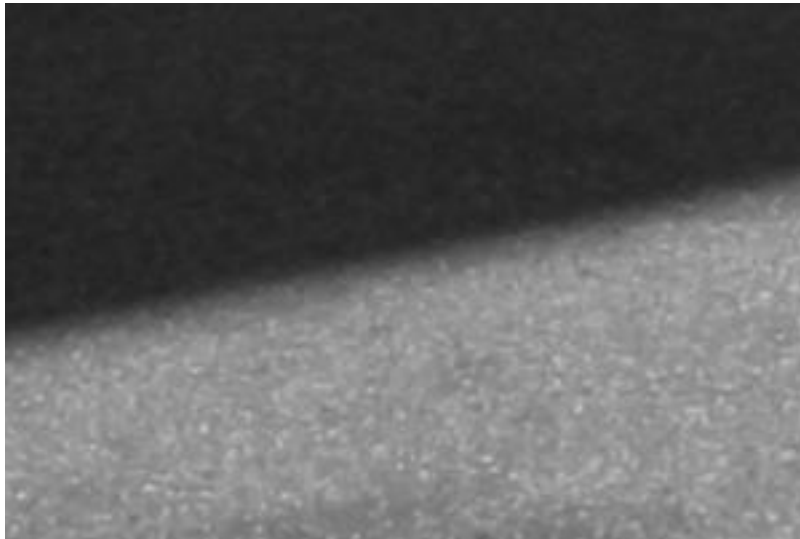

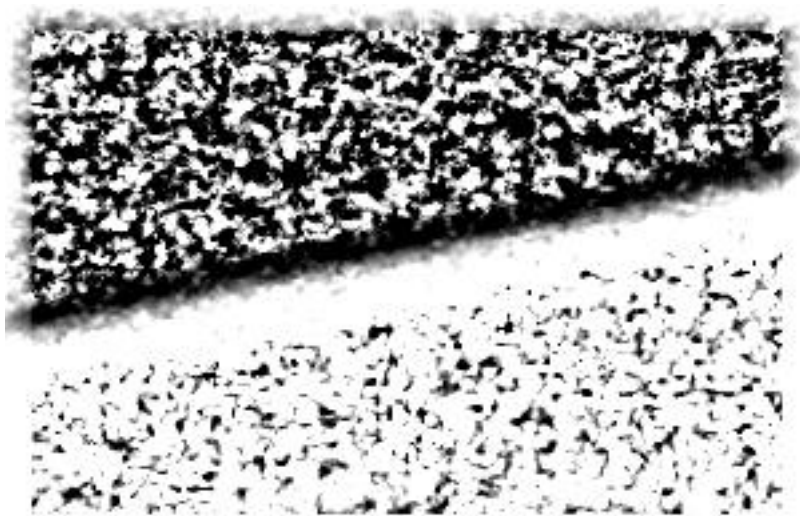Figure 6.2: A patch of asphalt with a strong gradient from a shadow.

Figure 6.3: Wallis filtered image of asphalt.

6.1.2   Brick walls

In the brighter area A.2, less points are matched than in the shadow areas. A possible cause for this is the less prominent gradients in the first case. The number of accepted points with using LSTM extended with MSP is close to those matched by standard LSTM.

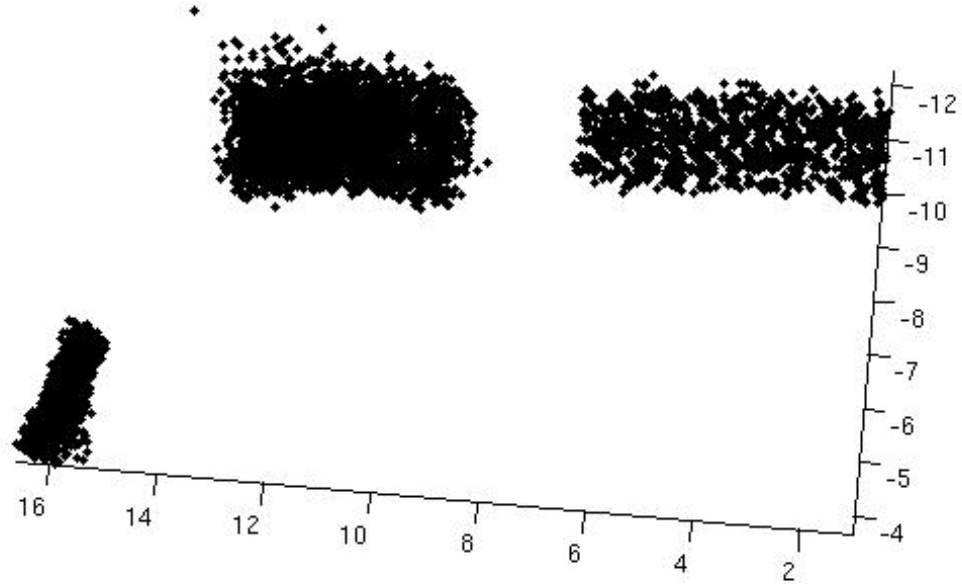Figure 6.4: 3d point cloud of matched points using LSTM with MSP in areas A.1, A.2 and A.4.

Table 6.2: Example 2, Area A.1, area A.2 and A.4. No improvements versus MSP.

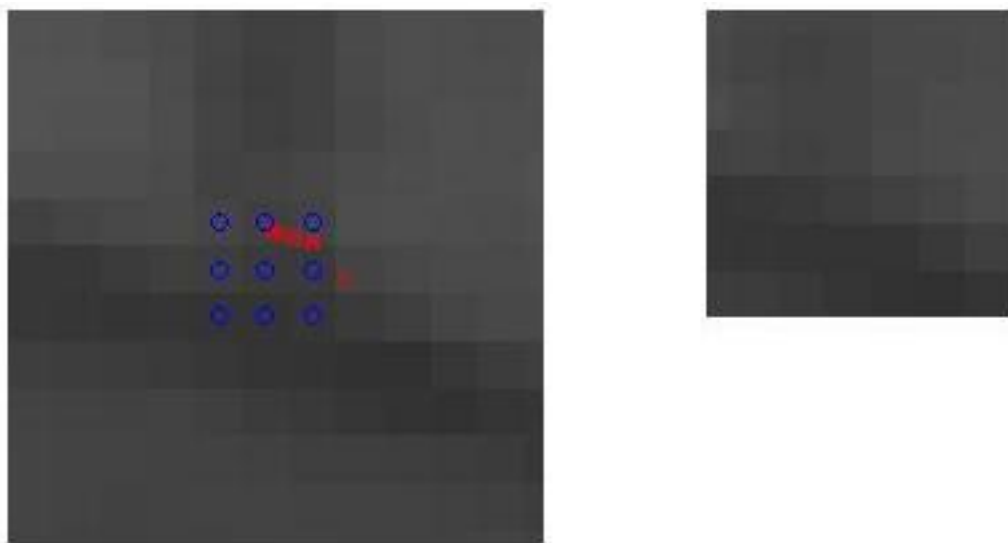| Area + method	A.1 + 0	A.1 + m	A.2 + 0	A.2 + m	A.4 +0	A.4 +m
Number of accepted points	3674	3683	1517	1484	2926	2902
Runtime (s)	105.2	845.4	138.9	1327	175.3	1428

Figure 6.5: Result of MSP in area A.1 in left image, template in right image. Blue circles are the multiple seed points, red dots are found points.

6.1.3   Door

On the dark door, improvements with Wallis filtering respectively Wallis with ATS give over 100 times as many hits as improvement with Wallis filtering and TNCC, see table 6.3 and figure 6.6, 6.7, and 6.8 for a comparison. A seen in the histogram in figure 6.9, larger template sizes only make a minor increase to the amount of accepted points.

Table 6.3: Example 3, Area A.3. TNCC + Wallis vs. TNCC + Wallis + ATS.

Area + method	A.3 + cw	A.3 + wa	A.3 w
Number of accepted points	41	9545	6182
Runtime (s)	7375	374.68	117.4

Figure 6.6: Seed points giving accepted results in part of Area A.3 using Wallis and TNCC.

Figure 6.7: Seed points giving accepted results in part of Area A.3 using Wallis.

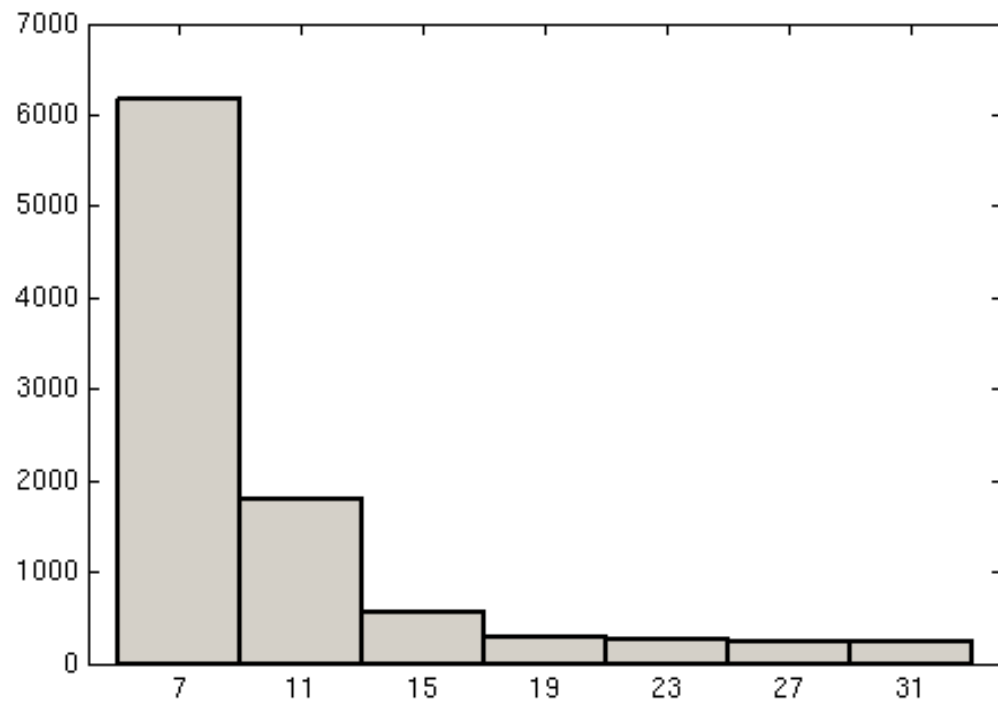Figure 6.8: Seed points giving accepted results in part of Area A.3 using Wallis and ATS.

Figure 6.9: Histogram over used template sizes in Experiment 3, A.3, with Wallis filtering and ATS.

6.1.4   Lawn

On the irregular gradients of a lawn, the extension of Wallis filtering found about 75% more points than standard LSTM, fulfilling the purpose of densification, as seen in table 6.4. Figure 6.11 compared to 6.10 shows, the Wallis filtering helped for accepting points in areas where the baseline LSTM failed.

Figure 6.10: Seed points giving accepted results in part of area C.1 using baseline LSTM.

Table 6.4: Results from example 4, Area C.1 baseline LSTM vs Wallis filtering

Area + method	C.1 + 0	C.1 + w
Number of accepted points	2283	3844
Runtime (s)	389	459

Figure 6.11: Seed points giving accepted results in part of area C.1 with Wallis filtering.

6.1.5   Corrugated plate

The ATS and Wallis extension to LSTM gives six times as many accepted points as plain LSTM, see table 6.5. Only Wallis gives less number of points this time, and the ATS is the significant improver. As seen in figure 6.14, the number of accepted points with template size 7 is the same as all accepted points with the Wallis extension. More points is accepted when larger templates is used. In area C.3 with the signs containing gradients in different directions, many more points are accepted in all cases.

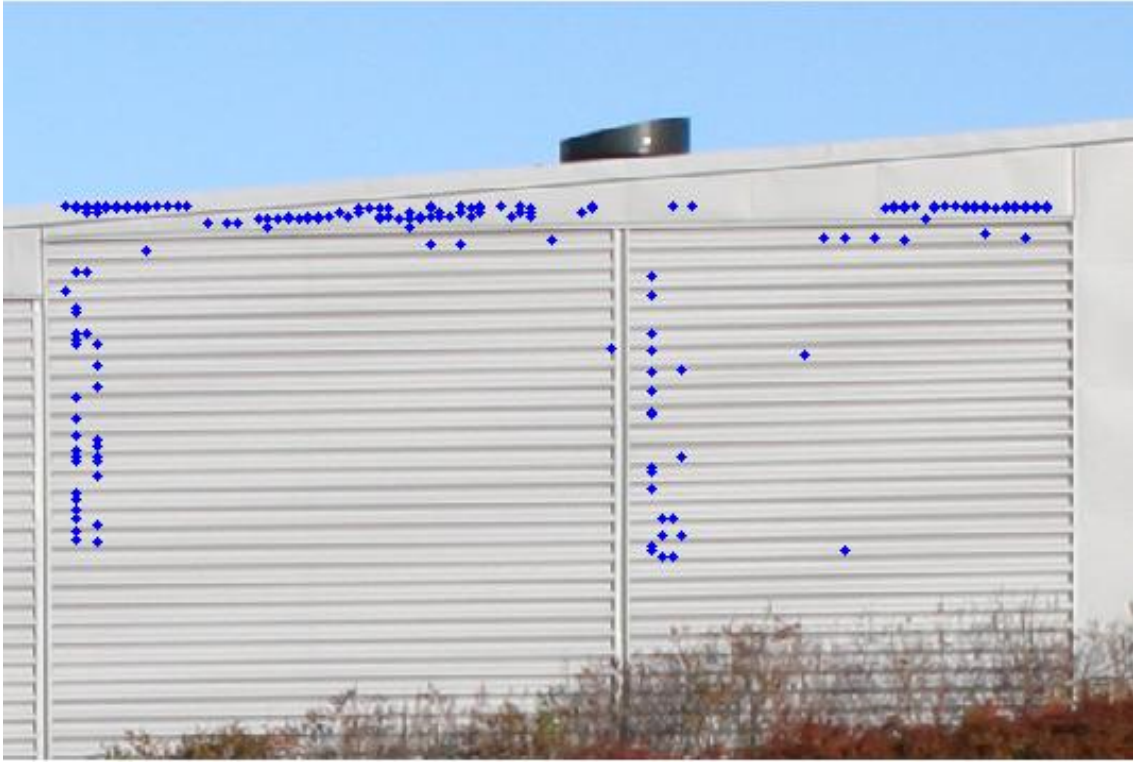Figure 6.12: Seed points giving accepted results in part of area C.2 with baseline LSTM.

Table 6.5: Results Example 5, Area C.2 and C.3 no improvements, Wallis vs. Wallis and ATS.

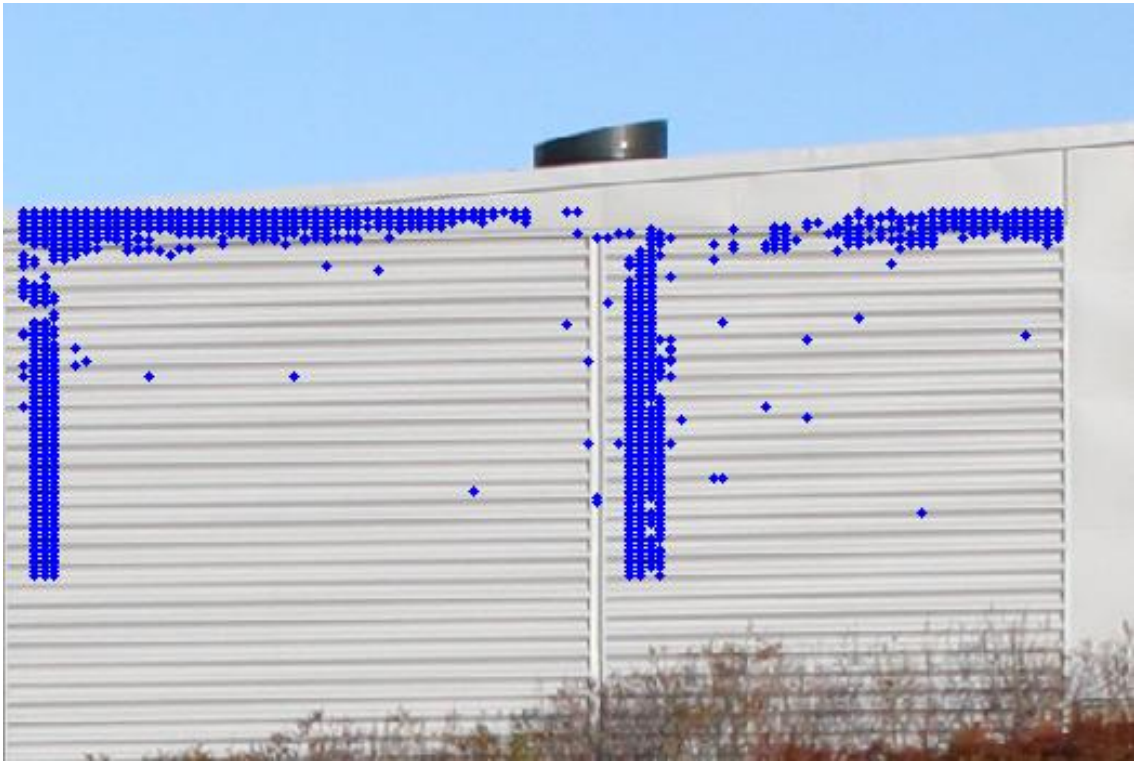| Area + method	C.2 + 0	C2 + w	c.2 + wa	C.3 + 0	C.3 + w	C.3 + wa
Number of accepted points	172	148	1116	1139	2771	6037
Runtime (s)	278.5	150.6	1392	408.9	129.9	642.4

Figure 6.13: Seed points giving accepted results in part of area C.2 with Wallis and ATS.
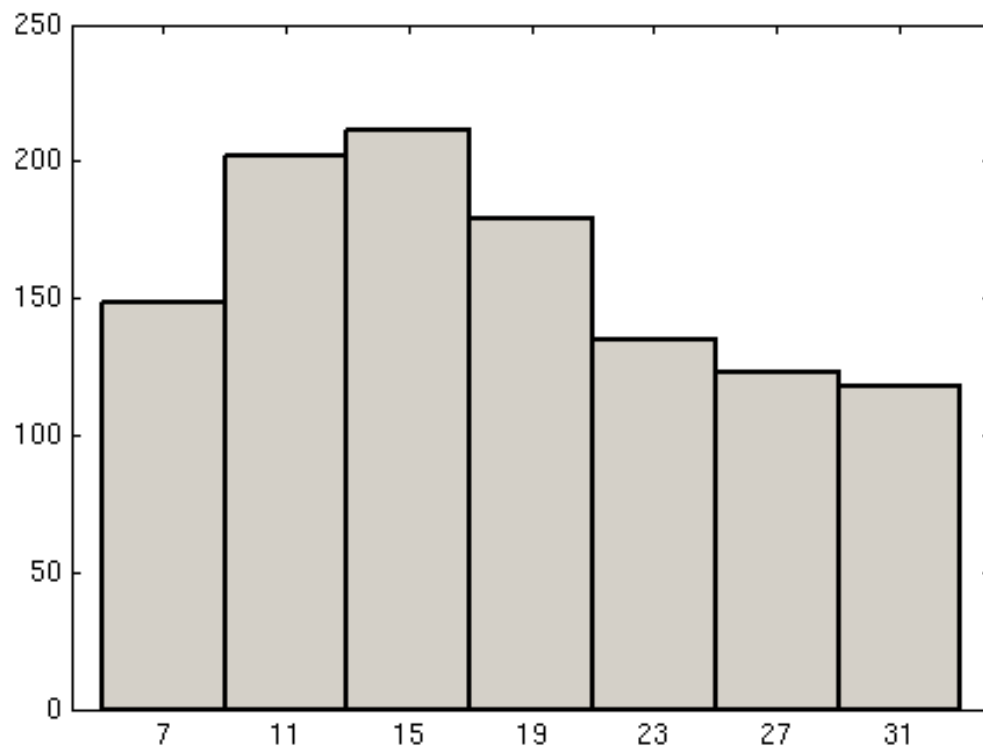
Figure 6.14: Histogram over required template sizes for acceptance of points in area C.2.

Chapter 7

Discussion

7.1 Evaluation of aims

In chapter 4, a set of properties to evaluate the methods were stated. All of these did not fit this implementation, however here are they commented.

- Point cloud quality:

  - Completeness - The densities of point clouds are varying, some extensions gives denser point clouds, some gives sparser but more exact clouds

  - Robustness - Some of the extensions rejects many points due to possible large errors, and hence their robustness is considered good.

  - Precision - Is not generally computed due to no knowledge of ground truth.

- Method sensitivity to:

  - Object geometry - Calculation gives that points with 0.2 meter deviation from the plane are easy to match.

  - Camera network geometry - Since only images obtained by a stereo rig of two cameras with fixed base line are used, the evaluation of method sensitivity with respect to camera network geometry is not applicable.

7.2   Additional analysis

7.2.1   Point cloud density

As seen in the tables and figures of the results chapter, the density of the point clouds varies between different choice of extensions to LSTM and type of area it is used on.

The main tendencies are

- TNCC reduces the density of the point cloud, sometimes more than ten times,

- Wallis filtering improves usually the density of the point cloud significantly, however there are some exceptions, area C.2, B.2 and B.4, where the densities decreased some.

- ATS increases the density, how much depends on structure of the surface.

- MSP neither increase or decrease the density more than a few percent.

7.2.2   Runtime

The runtime varies a lot between the different methods, and is also depending on the type of area. Some things to notice:

- Wallis filtering takes time directly depending on the size of the filtered area. When matching large amounts of points, the overhead for Wallis filtering isn't a problem.

- TNCC works fast on small areas, but in combination with APS it may generate too long runtimes. A couple of runs in this configuration are not presented because the runs exceeded a week and therefore aborted. With a smaller maximum template size, for example 21×21 pixels, this would not have been a problem.

- MSP gives the predicted nine times longer runtimes than base line LSTM.

- ATS runtime varies depending on how often large template sizes are used. It has acceptable speed if many points are accepted on small templates, but if the templates grows it slows down significantly.

7.2.3   Error codes

The error codes are analyzed for all cases except ATS, this is because the ATS error code, 3, over rules every other error code.

The error code 0 is set for the accepted points. This code applies to 30 % of the points total points.

As displayed in figure 7.1, code 5, the code for to high correlation between position and any other optimization parameter is the most common one. This one implies that the optimization may have gone wrong, and the returned point is not reliable. This code is possible to get with all the extensions.

The second most common error code is 4, indicating that the optimization did not converge followed by error code 1 from the TNCC. As noted in the results chapter, TNCC gave much less points than any of the other codes, and that is indicated by this common code only possible to set in the TNCC cases. Those points accepted with TNCC are intended to be more exact than the others.

Error code 7 is also set from the optimization, indicating to high standard deviation of the optimization parameters.
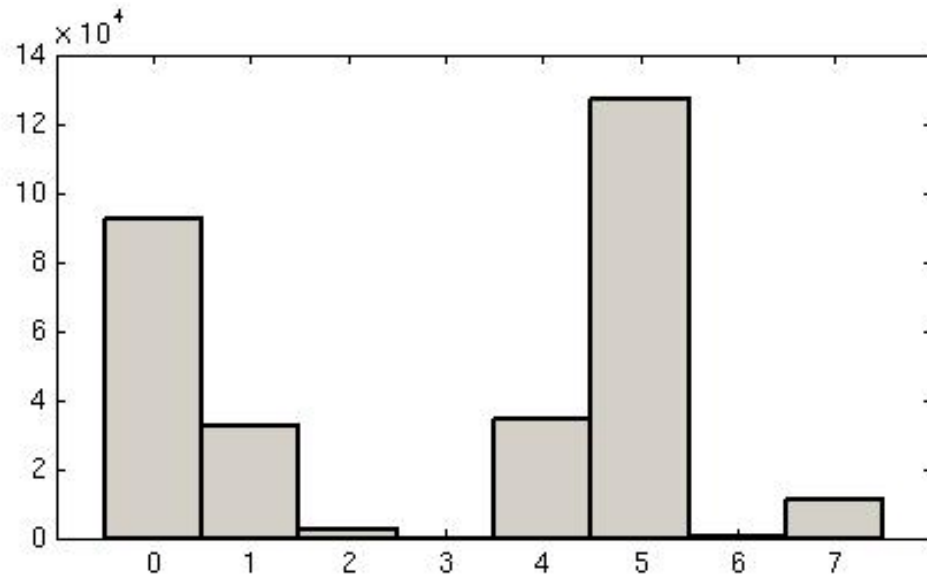
Figure 7.1: Histogram over error codes

The MSP rejects very few points by its error code 2. It may be possible to set a higher threshold on number of points converging to the same point here.

The error code 6 is possible to be reset by 7 in the error code logic, and may in reality fit on more points.

APS is required to set the error code 3, and resets then all other codes, so that one is not used in this evaluation.

7.2.4 Homographies

The quality of the homography is very relevant for the result, since a bad fitting hompgraphy gives bad seed points for the optimization. The code implemented for assisted creation of homographies gives a warning if it is plausible that the quality is low, but it is recommended to always generate a couple of homography matrices and compare them as well as test them on the image.

Chapter 8

Conclusions

It is possible to densify a point cloud using extensions to LSTM. Aspects of point quality and time consumptions need to be considered in comparison to the need of densification of the point cloud. The task of densification of point clouds suffer from the same difficulty as most image analysis tasks; generally working methods aren't existing, and different kinds of objects take advantage of different methods.

Chapter 9

Future work

During the work on this thesis a lot of questions possible to evaluate have risen. A selection of them are:

– **Template sizes** It would be possible to write an algorithm for finding the optimal patch size in different parts of an image. This would probably include frequency analysis of the image using Fourier transform or wavelets, or using the SIFT-detectors radius information.

– **TNCC.** The threshold for accepting a seed point from TNCC needs a deeper evaluation.

– **Precision of point clouds.** The point clouds generated from LSTM with the different extensions would be compared to ground truth of the object, determining the precision of the points.

– **Bench mark.**Constructing a set of images including camera calibration data, ground truth, homographies and sets of points to match and detect would give the opportunity to compare different methods regarding precision, robustness and time consumed in a re-usable way.

– **MSP** A deeper study of distribution of the multiple seed points, the threshold for the definition of the same convergence point, and number of agreeing points can be interesting.

Chapter 10

Acknowledgements

References

Steven D. Blostein and Thomas S. Huang. Error analysis in stereo determination of 3-d point positions. *IEEE T Pattern Anal*, 9(6):752–765, November 1987. doi: 10.1109/TPAMI.1987.4767982.

Niclas Börlin and Christina Igasto. 3d measurements of buildings and environment for harbor simulators. Technical Report UMINF 09.19, Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden, October 2009.

Nicolas D'Apuzzo. *Surface measurement and tracking of human body parts from multi station video sequences*. PhD thesis, Institute of Geodesy and Photogrammetry, ETH Zürich, Zürich, Switzerland, October 2003.

S.F. El-Hakim, J.-A. Beraldin, M. Picard, and G. Godin. Detailed 3d reconstruction of large-scale heritage sites with integrated techniques. *IEEE Comput Graphics Appl*, 24(3):21–29, May 2004. ISSN 0272-1716. doi: 10.1109/MCG.2004.1318815.

Håkan Fors Nilsson and David Grundberg. Plane-based close range photogrammetric reconstruction of buildings. Master's thesis, Department of Computing Science, Umeå University, Technical report UMNAD 784/09, UMINF 09.18 2009.

Wolfgang Förstner and Bernhard Wrobel. *Mathematical Concepts in Photogrammetry*, chapter 2, pages 15–180. IAPRS, 5 edition, 2004.

Jan-Michael Frahm, Marc Pollefeys, Brian Clipp, David Gallup, Rahul Raguram, ChangChang Wu, and Christopher Zach. 3d reconstruction of architectural scenes from uncalibrated video sequences. *International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences*, XXXVIII(5/W1):7 pp, October 2009.

D. Gallup, J.-M. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *Proc. CVPR*, pages 1–8, Minneapolis, Minnesota, USA, June 2007. IEEE. doi: 10.1109/CVPR.2007.383245.

Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley, 3rd edition, 2008.

A. Gruen. Least squares matching: a fundamental measurement algorithm. In K. B. Atkinson, editor, *Close Range Photogrammetry and Machine Vision*, chapter 8, pages 217–255. Whittles, Caithness, Scotland, 1996.

A. W. Gruen. Adaptive least squares correlation: A powerful image matching technique. *S Afr J of Photogrammetry*, 14(3):175–187, 1985.

Armin Grün, Fabio Remondino, and Li Zhang. Photogrammetric reconstruction of the great buddha of bamiyan, afghanistan. *Photogramm Rec*, 19(107):177–199, 2004.

R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, 2nd edition, 2003.

David G. Lowe. Object recognition from local scale-invariant features. In *Proc Intl Conf on Computer Vision*, pages 1150–1157, Corfu, Greece, September 1999.

J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In A. David Marshall and Paul L. Rosin, editors, *Proc British Machine Vision Conference*, pages 384–393, Cardiff, UK, September 2002. British Machine Vision Association.

Chris McGlone, Edward Mikhail, and Jim Bethel, editors. *Manual of Photogrammetry*. ASPRS, 5th edition, July 2004. ISBN 1-57083-071-1.

Douglas C. Montgomery, George C. Runger, and Norma Faris Hubele. In *Engineering Statistics*, 2004. ISBN 0-471-45240-8.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer-Verlag, 1999. ISBN 0-387-98793-2.

F. Remondino, S. El-Hakim, S. Girardi, A. Rizzi, S. Benedetti, and L. Gonzo. 3d virtual reconstruction and visualization of complex architectures - the 3d-arch project. *International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences*, XXXVIII(5/W1):9 pp, October 2009.

Fabio Remondino. *Image-based modeling for object and human reconstruction*. PhD thesis, Institute of Geodesy and Photogrammetry, ETH Zürich, ETH Hoenggerberg, Zürich, Swizerland, 2006.

Fabio Remondino, Sabry F. El-Hakim, Armin Gruen, and Li Zhang. Development and performance analysis of image matching for detailed surface reconstruction of heritage objects. *IEEE Signal Proc Mag*, 25(4):55–64, July 2008.

Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR'06*, volume 1, pages 519–528, June 2006. doi: 10.1109/CVPR.2006.19.

Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge, 3rd edition, 2003.

Axel Wendt. A concept for feature based data registration by simultaneous consideration of laser scanner data and photogrammetric images. *ISPRS J Photogramm*, 62(2):122 – 134, 2007. ISSN 0924-2716. doi: DOI: 10.1016/j.isprsjprs.2006.12.001.

Kuk-Jin Yoon and In So Kweon. Distinctive similarity measure for stereo matching under point ambiguity. *Comp Vis Imag Under*, 112(2):173 – 183, 2008. ISSN 1077-3142. doi: DOI: 10.1016/j.cviu.2008.02.003.

Appendix A

Homographies

Here is the homography matrices used for the different areas presented.

A.1 Loading dock

$$H1 = \begin{bmatrix} 0.9591 & -0.1217 & -68.80 \\ 0.0052 & 0.7070 & 158.1 \\ 0 & -0.0002 & 1 \end{bmatrix}$$

$$H2 = \begin{bmatrix} 0.912 & -0.055 & 12.70 \\ -0.0158 & 0.9234 & 75.42 \\ -0 & -0 & 1 \end{bmatrix}$$

$$H3 = \begin{bmatrix} 1.024 & 0.021 & -95.45 \\ 0.008 & 1.083 & -31.09 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H4 = \begin{bmatrix} 0.8794 & -0.251 & 78.94 \\ 0.0011 & 0.772 & 96.15 \\ 0 & -0.0002 & 1 \end{bmatrix}$$

$$H5 = \begin{bmatrix} 1.0 & -0.9 & 1194.1 \\ 0. & 1.0 & 36.4 \\ 0 & 0 & 1 \end{bmatrix}$$

A.2   Building Sliperiet

$$H1 = \begin{bmatrix} 0.9430 & 0.0028 & 58.96 \\ -0.0175 & 0.971 & 71.08 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H2 = \begin{bmatrix} 0.9617 & 0.0133 & 12.37 \\ -0.0131 & 0.989 & 57.58 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H3 = \begin{bmatrix} 0.9075 & -0.0182 & 120.61 \\ -0.0154 & 0.941 & 69.30 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H4 = \begin{bmatrix} 0.7 & -0.6 & 1067 \\ -0.1 & .4 & 640.2 \\ 0 & 0 & 1 \end{bmatrix}$$

A.3   Building Elgiganten

$$H1 = \begin{bmatrix} 1.077 & 0.602 & -872.95 \\ 0.052 & 0.935 & 1.80 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H2 = \begin{bmatrix} 1.021 & -0.003 & -46.81 \\ 0.012 & 1.013 & -49.32 \\ 0 & 0 & 1 \end{bmatrix}$$

$$H3 = \begin{bmatrix} 1.046 & -0.011 & -67.00 \\ 0.014 & 1.025 & -56.98 \\ 0 & 0 & 1 \end{bmatrix}$$

Appendix B

Abbreviations

LSTM	Least Squares Template Matching
ALSM	Adaptive Least Squares Matching
GPU	Graphics Processing Unit
VIP	Viewpoint Invariant Patch
SVD	Singular Value Decomposition
MSP	Multiple Seed Points
TNCC	Transformed Normalized Cross Correlation
ATS	Adaptive Template Size
SIFT	Scale Invariant Feature Transform