## 3D Reconstruction Using the Direct Linear Transform with a Gabor Wavelet Based Correspondence Measure

# **Technical Report**

Daniel Bardsley / Bai Li

1	Introduction	1
2	DLT Camera Calibration	1
2.1	2D DLT Algorithm	1
2.2	3D DLT Algorithm	3
3	DLT Reconstruction	3
4	DLT Data Normalisation	4
5	The DLT Gold Standard Algorithm	4
6	Gabor Stereo Correspondence	5
7	Voronoi Based Propagation Matching	6
8	DLT Accuracy Analysis	7
9	Reconstruction Results	8
10	Outstanding Issues1	0
10.	1 Multi-Source Lighting1	10
10.	2 Input Image Masking1	10
11	Conclusion 1	0
12	References1	11

## 1 Introduction

This report aims to describe the technical details behind the development of a 3D reconstruction system for face recognition. Initially the methods used for constructing a 3D model of the face from several cameras are discussed followed by some consideration of the pitfalls and problems behind some of the techniques.

The reconstruction system uses the Direct Linear Transform (DLT) method for both calibration and triangulation along with the Gabor Jet correspondence measure and Voronoi cell propagation matching strategy for stereo correlation. We also discuss the accuracy of the DLT method and assess its suitability for use in model construction for face recognition.

The final section of the report shows some example reconstructions using the described methods in combination with a six camera capture rig with a random light pattern projector and suggests how some of the problems currently being encountered could be solved.

# 2 DLT Camera Calibration

The most commonly used camera calibration method is perhaps the DLT (direct linear transformation) method originally reported by *Abdel-Aziz and Karara [1]*. The DLT method uses a set of control points whose object space/plane coordinates are already known. The control points are normally fixed to a rigid frame, known as the calibration frame. The problem is essentially to calculate the mapping between the 2D image space coordinates ( $x_i$ ) and the 3D object space coordinates ( $X_i$ ). For this 3D  $\leftarrow \rightarrow$  2D correspondence the mapping should take the form of a 3x4 projection matrix (P) such that  $x_i = PX_i$  for all i.

# 2.1 2D DLT Algorithm

Whilst we will be using the DLT algorithm for camera calibration it is also a suitable technique for finding linear mappings between any two data sets given a certain number of corresponding data points between the sets. The simplest form of the DLT algorithm is described below, however, it should be evident that the only difference between this method and the 3D case is the dimension of

the problem. In the 2D case the solution matrix has dimension 3x3 where as the 3D result produces a 3x4 projection matrix. The algorithm for the 3D DLT case is described in the next section.

The most basic form of the 2D DLT algorithm requires a set of four 2D to 2D point correspondences:  $x_i \leftarrow \rightarrow x'_i$ . The transform is then given by the equation  $x'_i = Hx_i$ . The equation may then be expressed in terms of a vector cross-product:  $x'_i \times Hx_i = 0$ . Expressing the transform in terms of a vector cross-product allows a simple linear solution to H to be calculated.

We denote the j<sup>th</sup> row of the matrix H by  $H^{jT}$ :

$$Hx_i = \begin{pmatrix} h^{1T} x_i \\ h^{2T} x_i \\ h^{3T} x_i \end{pmatrix}$$

Denoting X'<sub>i</sub> as  $(x'_i, y'_i, w'_i)^T$  the cross-product may be given explicitly as:

$$x'_{i} \times Hx_{i} = \begin{pmatrix} y'_{i} h^{3T} x_{i} - w'_{i} h^{2T} x_{i} \\ w'_{i} h^{1T} x_{i} - x'_{i} h^{3T} x_{i} \\ x'_{i} h^{2T} x_{i} - y'_{i} h^{1T} x_{i} \end{pmatrix}$$

Since  $h^{jT}x_i = X_i^T h_j$  for j = 1,2,3, this gives a set of three equations for H which may be written as in the following equation:

$$\begin{bmatrix} 0^{T} & -w'_{i} x_{i}^{T} & y'_{i} x_{i}^{T} \\ w'_{i} x_{i}^{T} & 0^{T} & -x'_{i} x_{i}^{T} \\ -y'_{i} x_{i}^{T} & x'_{i} x_{i}^{T} & 0^{T} \end{bmatrix} \begin{pmatrix} h^{1} \\ h^{2} \\ h^{3} \end{pmatrix} = 0$$

When each of the four coordinates being considered is presented in this form we have a set of equations:  $A_ih = 0$ , where A is a 3x9 matrix and h is a 9-vector made up of entries to the matrix H. This equation is linear in the unknown h.

It should be noted that whilst each set of coordinate matches leads us to a set of three equations only two of them are linearly independent. Thus, it is standard practice whilst using the DLT algorithm to ignore the third equation whilst solving for H. The set of equations then becomes:

$$\begin{bmatrix} 0^{T} & -w'_{i} x_{i}^{T} & -y'_{i} x_{i}^{T} \\ w'_{i} x_{i}^{T} & 0^{T} & -x'_{i} x_{i}^{T} \end{bmatrix} \begin{pmatrix} h^{1} \\ h^{2} \\ h^{3} \end{pmatrix} = 0$$

This gives us the equation  $A_ih = 0$ , where  $A_i$  is now a 2x9 matrix. This equation holds true for any homogeneous coordinate representation of the coordinates involved.

Each point correspondence gives rise to two independent equations in the entries for H. Given four correspondences we obtain a set of equations Ah = 0 where A is formed from the equation coefficients built from the matrix rows A<sub>i</sub>. Next, in order to solve for A, we obtain the singular value decomposition (SVD) of A and take the smallest singular value as our solution and thus determine the linear transform between x<sub>i</sub> and x'<sub>i</sub>.

If more than four corresponding points are known and the measurements contain noise (as is usual in computer vision processing) then we must find an over-determined solution for the equation Ah = 0. This is achieved simply by stacking the n 2x9 matrices  $A_i$  into a single 2nx9 matrix and using SVD to solve for A. This is known as the basic DLT algorithm.

### 2.2 3D DLT Algorithm

In order to apply the basic 2D  $\leftarrow \rightarrow$  2D DLT algorithm to the 2D  $\leftarrow \rightarrow$  3D case we must simply change the dimension of the problem. In the 3D case for each correspondence  $X_i \leftarrow \rightarrow x_i$  we derive the following equation:

$$\begin{bmatrix} 0^{T} & -w_{i}X_{i}^{T} & y_{i}X_{i}^{T} \\ w_{i}X_{i}^{T} & 0^{T} & -x_{i}X_{i}^{T} \\ -y_{i}X_{i}^{T} & x_{i}X_{i}^{T} & 0^{T} \end{bmatrix} \begin{bmatrix} P^{1} \\ P^{2} \\ P^{3} \end{bmatrix} = 0$$

As in the 2D case the third equation is dependent on the first two and as such can be discounted. This leaves us with the following:

$$\begin{bmatrix} \mathbf{0}^T & -w_i X_i^T & y_i X_i^T \\ w_i X_i^T & \mathbf{0}^T & -x_i X_i^T \end{bmatrix} \begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix}$$

From a set of n point correspondences we now have a 2nx12 matrix A formed by stacking each of the equations from their respective point correspondences. The projection matrix for a given camera can be computed by solving the set of equations Ap = 0, where p is a 3x4 projection matrix.

## 3 DLT Reconstruction

Having utilised the DLT method to calculate the projection matrix for each camera in a stereo rig it then becomes possible to project 2D camera-space coordinates into 3D when the point is visible from more than one camera. Since we know this is a linear mapping (assuming that we ignore radial distortion in the cameras) it is again possible to use the DLT method to calculate a previously unknown mapping. This section describes a simple linear triangulation method that uses the same principles discussed in Section 1.

The reconstruction problem is solved as follows. For each input image we have a measurement x = PX, x' = P'X where x is the 2D camera-space coordinates of a world point, x' is the same point projected into the camera-space coordinates of a second camera. X represents the 3D world-space coordinate that we are attempting to recover. These two equations can be combined into the form AX = 0, which is an equation linear in X. The homogeneous scale factor is eliminated by a cross-product to give three equations for each image point visible in more than one of the cameras in the system. As an example the equation derived for a point in the first image would be given as  $x \times (PX) = 0$ . Expanded, this gives the following set of three equations:

$$x(p^{3T}X) - (p^{1T}X) = 0$$
  

$$y(p^{3T}X) - (p^{21T}X) = 0$$
  

$$x(p^{2T}X) - y(p^{1T}X) = 0$$

Combining equations from both cameras to produce an equation in the form AX = 0 gives us:

$$A = \begin{bmatrix} xp^{3T} - p^{1T} \\ yp^{3T} - p^{2T} \\ x'p^{'3T} - p^{'1T} \\ y'p^{'3T} - p^{'2T} \end{bmatrix}$$

Solving for A using SVD allows us to estimate the value of X and thus the 3D coordinate of any point for which we know the camera-space coordinates from two cameras for which the projection matrix has already been determined.

## 4 DLT Data Normalisation

The result of the DLT algorithm for computing homographies depends on the coordinate frame in which the points are expressed [2]. For this reason it is desirable to normalise the input data before applying the DLT algorithm. Data normalisation improves the accuracy of the results whilst ensuring that the algorithm will be invariant to arbitrary choices in scale and coordinate frame. Image space coordinates for each of the cameras is normalised independently as follows:

- 1. Coordinates are translated so their centroid is located at the origin.
- 2. Coordinates are scaled so that the average distance to the origin is  $\sqrt{2}$  and thus the average point is equal to (1,1,1).

The world space coordinates are normalised in a similar manner except in the 3D case the scaling is computed to ensure the average distance to the origin is  $\sqrt{3}$  therefore making the average 3D point of the input data to be (1,1,1,1).

Once the data has been normalised the DLT algorithm is applied as before. However, applying normalisation to the input data leaves us with a normalised projection matrix for each camera. In order to de-normalise the projection matrix we must reverse the normalisation transform. If T is the similarity transform which normalises the image space coordinates, U is a second similarity transform to normalise the world space coordinates and P' is the normalised projection matrix then the projection matrix for un-normalised coordinates is calculated as follows:

$$P = T^{-1}P'U$$

The recommended form of the DLT algorithm requires the use of the data normalisation step before calculating the homography.

#### 5 The DLT Gold Standard Algorithm

One method of calculating the error in a reconstruction is by measuring the geometric error. Assuming that the world points in our calibration are accurately known then we may define the geometric error as:

$$\sum d(x_i, \hat{x}_i)^2$$

Where  $x_i$  is the re-projected point and  $\hat{x}$  is the exact projection of the world point. Thus the solution to the following minimisation is the maximum likelihood estimate of P.

$$\min_{P} \sum_{i} d(x_{i}, PX_{i})^{2}$$

Minimising the geometric error requires the use of iterative techniques. This increases the computation time but as the calculation only occurs during calibration it is an acceptable loss in

performance. The Levenberg-Marquardt minimisation technique is suitable for our purposes and the initial DLT estimate for P can be used as an initial parameterisation for calculating the maximum likelihood of the projection matrix. When used in conjunction with data normalisation and DLT this calibration method is known as the Gold Standard algorithm for estimating P. The full details of this method are detailed in [2] with the complete algorithm referenced below:

### Objective:

Given  $n \ge 6$  world to image point correspondences  $X_i \leftarrow \rightarrow x_i$ , determine the Maximum Likelihood estimate of the camera projection matrix P.

#### Algorithm:

- 3. Linear Solution: Compute an initial estimate of P using a linear method.
  - a. **Normalisation**: Calculate the similarity transform T to normalise the image coordinates and similarity transform U to normalise the world points as described in the Data Normalisation section.
  - b. DLT: Form the 2n × 12 matrix A by stacking equation 2 as generated by each 2D to 3D correspondence. A solution to Ap=0, subject to ||p||=1, is obtained from the unit singular vector of A corresponding to the smallest singular value.
- 4. **Minimise Geometric error:** Using the linear estimate as starting point minimise the geometric error:

$$\min_{P} \sum_{i} d(\widetilde{x}_{i}, \widetilde{P}\widetilde{X}_{i})^{2}$$

- Over  $\widetilde{P}$ , using an iterative algorithm such as Levenberg-Marquardt.
- 5. **Denormalisation:** The camera matrix for the original (un-normalised) coordinates is obtained from  $\tilde{P}$  as:

$$P = T^{-1}P'U$$

#### 6 Gabor Stereo Correspondence

In order to perform 3D reconstruction using the DLT method we must know the image space coordinates of a given world point in at least two separate 2D projections. In order to produce these 2D coordinate matches a search is carried out over the input images. Initially, we take the point to be reconstructed in the first image and carry out a search of the second image to find a suitable match. Since we have already obtained the projection matrix for each of the cameras, the epipolar constraint allows us to limit the search space to the corresponding epipolar line on the second image. In order to calculate the best candidate match for pixels in the epipolar search space Gabor Jets are used as the similarity measure.

The Gabor wavelet, [3], was originally proposed by Denis Gabor in 1946 in order to represent signals s a combination of elementary functions. The Gabor wavelet has been shown to provide optimal analytical resolution in both the spatial and frequency domains. Later work by Granlund [4] introduced the 2D counterpart (equation 4) of the elementary wavelet. This was closely followed by later work by Daugman [5] who presented evidence that the 2D Gabor wavelet family well represented the receptive fields of the human visual cortex. More recently Okajima studied the Gabor wavelet family from an information theory perspective showing that Gabor type receptive fields can extract maximal information from a local image region [6]. Owing to its array of useful properties the Gabor wavelet has found applications in face recognition [7, 8], texture segmentation [9], finger print recognition [10, 11], hand writing recognition [12, 13] and stereo vision [14, 15].

The 2D form of the Gabor wavelet is as follows:

$$G(x,y) = \frac{1}{2\pi\sigma\beta} e^{-\pi \left[\frac{(x-x_{o})^{2}}{\sigma^{2}} + \frac{(y-y_{o})^{2}}{\beta^{2}}\right]} e^{i[\xi_{o}x+v_{o}y]}$$

Where, where (*xo*, *yo*) is the center of the receptive field in the spatial domain and ([o, Qo) is the optimal spatial frequency of the filter in the frequency domain.  $\sigma$  and  $\beta$  are the standard deviations of the elliptical Gaussian along *x* and *y*.

In order to perform analysis of a particular image region a family of Gabor wavelets is derived from a mother wavelet. Each of these derived filters is then convolved with the image, with the response of each filter being combined into a vector representing all of the filters. This vector of Gabor filter responses is known as a Gabor Jet. Comparisons between different Gabor jets allow a measure of similarity between the image regions to be computed. Equation 5 defines the jet similarity functions for two images (J and J'):



Where  $a_j$ ,  $j=1,...,G_f$  is the magnitude of the result of the convolution between the real and imaginary part of the Gabor Filter, *j*, and the image.

In the described stereo vision system the initial seed points in the reference image are matched to pixels in the corresponding image first by obtaining the gabor jet for filters centered on the reference seed pixel, this jet is then compared with the jet corresponding to each pixel on the corresponding epipolar line. The pixel with the highest similarity is then selected as a match.

Previous work, [16], has shown the Gabor correspondence method to be robust against illumination and perspective distortions which we will encounter within the vision system. Much of the work using Gabor filters, particularly that stemming from research into 2D face recognition similarity metrics, suggests that it would prove a suitable correspondence measure for our work.

# 7 Voronoi Based Propagation Matching

Whilst attempting to correlate feature points between images in a stereo pair various factors such as image noise, occlusion or illumination differences can lead to incorrect matches no matter what correlation algorithm. For this reason it is necessary to constrain the matching process as far as possible using knowledge of the nature of the surface we are attempting to reconstruct. Common matching constrains include: similarity threshold, uniqueness, continuity, ordering, epipolar and relaxation. In order to constrain the way in which the correlation algorithm searches for an appropriate match a search strategy is required. An efficient search strategy will increase the accuracy of a correlation algorithm by reducing the potential search space, whilst usually decreasing the overall search time by requiring fewer comparisons per feature point. An efficient matching strategy is described below, which increase both accuracy and speed within the reconstruction system.

The proposed matching strategy is based on the Voronoi propagation method proposed by Tang, Tsui and Wu in [17]. A number of



Figure 1

modifications to their original design have been made in order to produce a more robust strategy. Initially N seed points are selected in the initial image. These seed points should, ideally, be the most salient points in the input image since errors at this stage will produce catastrophic results later in the process. The original seed points are then matched to their corresponding locations in the image pair. Since it is imperative at this stage to correctly match the seed points, the Gabor correlation algorithm is used and performs a full epipolar line search for each of the seed points.

The Gabor algorithm is used since it is often more robust to changes in illumination and perspective than other alternatives.

Once the seed points have been selected and matched the Voronoi diagram of the original seed points is calculated (figure 1). The Voronoi diagram of a collection of seed feature points is a partition of an image space into cells, each of which consists of those image points which are closer to one feature point than to any other. Voronoi diagrams are involved in situations where a space needs to be partitioned into "spheres of influence" [17], hence it is a good choice for use in this propagation algorithm. Once the Voronoi diagram has been calculated, matches are propagated from the seed points towards boundaries of the Voronoi cells until all of the matched regions are merged together. Strong matches in the propagation process are used to guide further matches within the same cell.

This method of propagation inherently enforces a continuity constraint into the matching process. This makes the assumption that object surfaces will be smooth and continuous. This assumption is not always valid for real world objects and will certainly break down at large discontinuities in the image, however, it is a suitable constraint given the advantages in speed that can be obtained through its use. Furthermore, additional processing steps could be employed and the constraint dynamically withdrawn at image locations where it does not hold true. Propagation provides a convenient method of producing dense correlation maps whilst also reducing the computational cost of the matching process. The reduction in computation stems from the fact that once the match for the initial seed point has been calculated the search for points within the same cell can be guided by the relative position of the matched seed point. This reduces the search space by an order of magnitude from a full scan line search to a small localized area.

Matches propagate outwards from the initial seed points in each cell in a standard breadth first search pattern. As a match for each pixel is found its neighbours are then added to the queue of pixels waiting to be matched. Pixels with high match strengths are used to produce an initial estimate for the position of neighbouring pixels reducing the potential search space to a window only a few pixels wide, resulting in a smooth surface with only a low number of comparisons between candidate matches. The algorithm cycles until every pixel within the given Voronoi cell has been matched to its corresponding point. The entire process is then repeated for each initial seed point until a dense disparity map has been produced.

# 8 DLT Accuracy Analysis

In order to test the correctness of the DLT algorithm for both calibration and reconstruction a synthetic calibration object was created using a 3D modelling package. A synthetic scene was used to ensure that both the world space coordinates and image space coordinates are accurately known.

Algorithm	Rig Configuration	3D Geometric Error: <i>d(X, X')</i>		2D Geometric Error: <i>d(x, PX)</i>	
		Sum	Average	Sum (P / P')	Average (P / P')
DLT	R45 / T20	2.990047	0.025556	73.40056 / 88.56606	0.62735 / 0.75697
Gold Standard	R45 / T20	3.00303	0.025667	73.17187 / 88.55565	0.62540 / 0.75688
DLT	R25 / T0	2.765166	0.023634	52.78314 / 52.34621	0.45113 / 0.44740
Gold Standard	R25 / T0	2.768046	0.023659	52.79225 / 52.37272	0.45121 / 0.44763

Rig Config. Key:	А	45 deg. Angle between cameras / 20 Unit horizontal separation				
	В	25 deg. Angle between cameras / 0 Unit horizontal separation				
Table 1						

Once reconstruction has been carried out the geometric error between the reconstructed object and the original object are compared. We also compare the re-projected 2D coordinates with the original image space coordinates in order to test the 2D geometric error. Several different camera configurations were also tested to ensure the results reflect several real world scenarios. The

synthetic object being reconstructed is a standard calibration object with 117 control points placed along two separate planes at 90 degrees to each other.

As can be seen from the accuracy test results (table 1) 2D geometric error is approximately 0.5 pixels per reconstructed point. The 3D geometric error is less then 0.03 units in all cases. The Gold Standard algorithm outperforms the standard DLT algorithm using rig configuration A, however, under configuration B the standard DLT algorithm has very slightly more accurate results. This is surprising since the Gold Standard algorithm seeks to minimise the 2D geometric error. This is most likely a result of the data normalisation step which despite the claim that "data normalisation should not be considered an optional step when implementing the DLT algorithm", [2] in certain situations seems to decrease the accuracy of the results, this is despite claims of consistent improvements in accuracy using the Gold Standard algorithm by Hartley and Zisserman, [2]. Overall the accuracy of either the DLT or the Gold Standard algorithms should be suitable for our reconstruction purposes.

# 9 Reconstruction Results

In this section we demonstrate some of the results of using the outlined techniques in real world situations. The rig used for the example reconstructions consists of six cameras (4 black and white and 2 colour) along with a projector for producing a structured light pattern. Reconstruction takes place using the four black and white cameras for stereo matching whilst the two colour cameras are used for texture mapping the resultant object. Figure 2 shows the input images acquired from the rig.



Figure 2: Top row left to right 1A, 2A, 2C. Bottom row left to right 1B, 2B, 1C.

The rig is calibrated using a standard calibration object and projection matrices were determined using the Gold Standard algorithm discussed previously. Stereo matches were calculated first between cameras 1A and 1B and then between cameras 2A and 2B. Since all six cameras are calibrated simultaneously from a single image of the calibration object they each project into the same world space coordinate system and thus no registration between the left / right camera reconstructions is required.

Once a set of reconstructed points have been calculated and merged into a single point cloud we construct a surface using the powercrust [18, 19] algorithm. Using some of the available noise reduction and smoothing parameters we produce a robust closed surface for the reconstructed face object. The texture coordinates are then generated by back projecting the object vertices to the colour input images (cameras 1C and 2C). The texture mapping process is complicated due to the presence of multiple texture images and thus we must decide which of the two texture images should be used for which object facet. In order to choose the appropriate texture source the angle of each facet normal to each of the two texture cameras is calculated. The texture camera with the lowest normal to camera angle is selected for a given facet since this texture image



Figure 3

can be shown to be viewing that particular object region with the least amount of distortion. Figure 3 shows a fully reconstructed face with full texture mapping.

The accuracy of the rig used for constructing the face model in Figure 3 was tested in a similar manner to the accuracy analysis carried out in the previous section: after calibration the world points are re-projected to the initial camera space coordinates and the geometric error calculated. Table 2 shows the error rates for each of the cameras in the reconstruction rig.

Camera	3D Geometric	Error: <i>d(X, X')</i>	2D Geometric Error: <i>d(x, PX)</i>	
	Sum	Average	Sum	Average
1A	2.592395	0.046294	14.991159	0.267699
1B			15.108214	0.26979
2A	2 961105	0.051093	19.607978	0.350142
2B	2.001193		19.891001	0.355196

Table 2

As shown by the above results the four camera rig calibration results show less geometric error than the synthetic scene used to test the algorithm. The average geometric error over the whole system is 0.31 pixels per reconstructed point; where as the 3D geometric error is on average 0.097 units per reconstructed point.

## 10 Outstanding Issues

Several outstanding issues remain with the reconstruction system. This section will discuss these problems in more detail along with potential solutions.

## 10.1 Multi-Source Lighting

During reconstruction two independent cameras are used to capture texture information from the face. In the reconstruction rig the two cameras are located either side of the face. Despite both texture images being captured simultaneously the angle difference between cameras causes variable lighting conditions on each side of the face causing an obvious "join" in the final texture map. Figure 4 shows a clear example of changes in lighting conditions between the two texture sources.

Two possible solutions to this problem are firstly to automatically correct the lighting levels for each of the cameras or secondly to blend the two texture images at the join in the texture map. The first (lighting correction) method is probably the most desirable since this would ensure both halves of the reconstruction are displayed under

matching solution may be appropriate for solving

A histogram

Figure 4

this issue; however, the process is complicated due to background objects in each of the texture images preventing a true histogram of face colours to be computed correctly.

The issue of correcting face textures in the 3D model may also be unnecessary since the recognition stage is likely to use only 3D shape data and thus correct texture mapping of the model is probably not a requirement for recognition.

# 10.2 Input Image Masking

the appropriate illumination.

Prior to performing stereo matching on the input images it is necessary to segment face/non-face pixels. Currently this part of the reconstruction process is carried out manually. Development of an automatic segmentation method is complicated since the input images are captured as the structured light pattern is projected onto the face. This renders the methods such as skin colour segmentation useless. One possible solution would be to carry out the face segmentation on the colour camera images and then project the mask onto the non-colour (structured light) input. A second option would be to develop a method capable of segmenting the structured light images directly, however, this would require the implementation of a novel segmentation method where it would be preferable to use a more tired and tested technique.

# 11 Conclusion

The above technical solution presents the methodology required to successfully reconstruct 3D scenes given a suitably calibrated camera rig and input images. 3D output could be improved through the use of a more sophisticated surface construction algorithm and by addressing issues in the previous section, however, the above information should provide a starting point for researchers wishing to perform multi-view 3D reconstruction.

# 12 References

- 1. Abdel-Aziz, Karara., *Direct linear transformation into object space coordinates in closerange photogrametry.* In Proc. Symp. Close-Range Photogrametry, 1971: p. 1-18.
- 2. Zisserman, Richard Hartley and Andrew, *Multiple View Geometry in Computer Vision*. Second ed. 2004: Cambridge University Press.
- 3. Gabor, Denis, *Theory of communications.* Journal of Institution of Electrical Engineers, 1946. **93**: p. 429-457.
- 4. Granlund, G. H., *Search for a General Picture Processing Operator.* Computer Graphics and Image Processing, 1978. **8**: p. 155-173.
- 5. Daugman, J. G., Uncertainty Relation for Resolution Space, Spatial-Frequence and Orientation Optimised by Two-Dimensional Visual Cortical Filters. Journal of the Optical Society of America A-Optics Image Science and Vision, 1985. **2**: p. 1160-1169.
- 6. Okajima, K., Two-dimensional Gabor-type receptive field as derived by mutual information maximization. Neural Networks, 1998. **11**: p. 441-447.
- 7. Bai Li, Dylan Shen. Combining wavelet and HMM for face recognition. in 23rd Artifical Intelligence Conference. 2003. Cambridge, UK.
- 8. LeiZhang, Stan Z. Li, ZhiYiQu, Xiangsheng Huang, *Boosting Local Feature Based Classifiers for Face Recognition.* 2001.
- 9. Levesque, Vincent, Texture Segmentation Using Gabor Filters. 2000.
- 10. Chih-Jen Lee, Sheng-De Wang, Kuo-Ping Wu. *Fingerprint Recognition Using Principal Gabor Basis Functions*. in *International Symposium on Intelligent Multimedia, Video and Speech Processing*. 2001. Hong Kong.
- 11. Fernando Alonso-Fernandez, Julian Fierrez-Aguilar, Jaview Ortega-Garcia, An Enhanced Gabor Filter-Based Segmentation Algorithm for Fingerprint Recognition Systems. 2004.
- 12. Yefeng Zheng, Huiping Li, David Doermann, The Segmentation and Identification of Handwriting in Noisy Document Images. 2002.
- 13. Yong Zhu, Tieniu Tan, Yunhong Wang, *Biometric Personal Identification Based on Handwriting.* 2000.
- 14. A.D. Calway, H. Knutsson, R. Wilson. *Multiresolution Estimation of 2-d Disparity Using a Frequency Domain Approach*. in *British Machine Vision Conference*. 1992. Leeds.
- 15. Yuzhi Chen, Ning Qian, A Coarse-to-Fine Disparity Energy Model with Both Phase-Shift and Position-Shift Receptive Field Mechanisms. Neural Computation, 2004. **16**: p. 1545-1577.
- 16. Osadchy, Margarita, *What Makes Gabor Jets Illumination Insensitive?* 2002.
- 17. Li Tang, H.T. Tsui, C.K. Wu, Dense Stereo Matching Based on Propagation with a Voronoi Diagram. 2003.
- 18. Nina Amenta, Sunghee Choi, Ravi Kolluri. *The Power Crust.* in *Sixth ACM Symposium on Solid Modeling and Applications.* 2001.
- 19. Nina Amenta, Sunghee Choi, Ravi Kolluri, *The power crust, unions of balls, and the medial axis transform.* Computational Geometry: Theory and Applications: special issue on surface reconstruction, 2001. **19**(2-3): p. 127-153.