EXTRINSIC CAMERA PARAMETERS ESTIMATION FOR SHAPE-FROM-DEPTHS

Jonathan Ruttle, Claudia Arellano and Rozenn Dahyot

School of Computer Science and Statistics
Trinity College Dublin, Ireland

ABSTRACT

3D reconstruction from multiple view images requires that camera parameters are very accurately known and standard camera calibration techniques [1] often fail to provide the required level of accuracy for the extrinsic camera parameters. Using the Kinect depth camera, we propose to estimate camera parameters by minimising the cross correlation between density functions modelled for each recorded depth images. We illustrate experimentally how this improves the modelling for estimating 3D shape from Depths.

Index Terms— Shape-from-Silhouettes (SfS), Shape-from-Depths (SfD), Multiview geometry.

1. INTRODUCTION

With the availability of cheap depth sensors like the Kinect camera, there is an increasing interest into performing accurate 3D reconstruction using depth images recorded from multiple views. The estimation of extrinsic camera parameters for each captured image is essential for merging the information into a common coordinate system. While in some highly controlled environments and in synthetic environments this level of accuracy is possible, in other situations it can be very expensive, time consuming and hard to achieve. In general a checkerboard pattern is used to determine these parameters. All cameras in the set up should be able to see the one calibration pattern, although multiple calibration patterns can be used if the translation between the patterns is known. However, with the pattern in awkward angles and the mapping from pattern to pattern, a problem arises with the propagation of errors: the data can be misaligned by up to 2 or 3 cms. This means that fine details can be impossible to recover for small objects.

After a short review (section 2), we propose to extend the modelling proposed by Ruttle et al. for shape-from-silhouettes [2] to using depth information (section 3). The cost function is modelling explicitly the noise on the observations (i.e. modelling uncertainties about the pixel resolution and the depth values). The extrinsic camera parameters are estimated using a robust metric between density functions and we show experimentally how this framework allows us to refine the initial camera parameters obtained by calibration to get a better 3D reconstruction (section 4).

Thanks to Trinity College of Dublin and The Government of Chile for funding.

2. CONTEXT

Cui et al. [3] proposed to use sequences of depth data recorded by a time of flight camera for 3D object scanning. Merging several scans together allows to improve the quality of the mesh despites the strong noise in the depth data. The alignment between the depth images is very important for achieving an accurate reconstruction. The rotation and translation are estimated by mapping two depth scans converted as 3D point scans. Their cost function for this estimation corresponds to the cross correlation between two probability density functions (pdf) each of which represents a point cloud [4]. The pdf corresponding to the reference point cloud is modelled as the empirical density function, while the second point set is model with a Gaussian mixture. The resulting cost function to estimate the rigid transformation can also be understood as maximising the likelihood function (the Gaussian mixture) modelled by one point set, while the second corresponds to the observations for computing the likelihood [3].

Cui et al. [3] converted the depth data into a 3D point clouds and the uncertainty (or bias) is modelled by a systematic offset in the direction of the camera ray. This model accounts only for the noise on the depth data but not for the uncertainty associated with the pixel resolution. In section 3, we propose to model both uncertainties explicitly without converting the depth information into 3D point clouds. The extrinsic camera parameters are also estimated using the correlation between two density functions [4]. However while we have explicit expressions for these density functions we do not have independent observations sampled from these distributions to compute directly the correlation. Section 3.2 presents how the correlation is computed for our modelling. In the case of the estimation of a rigid transformation, maximising the correlation between two pdfs can be shown to be equivalent to minimising the Euclidian distance between the pdf [5] . Minimising the distance between two probability density functions is a very robust approach for parameter estimation [6].

3. SHAPE FROM DEPTHS (SFD)

To solve SfD, we define the following random variables:

- $\Theta \in \mathbb{R}^3$ is the 3D spatial latent variable of interest. The cost function proposed here is optimised w.r.t. Θ to extract the shape of the object in view in the depth images.

- $\Psi \in \mathbb{R}^6$ correspond to the extrinsic camera parameters modelled as a nuisance random variable. ψ_1 is the roll component of rotation matrix, ψ_2 is the pitch component of rotation matrix, ψ_3 is the yaw component of rotation matrix and (ψ_4, ψ_5, ψ_6) is the 3D translation vector [1]. The roll, pitch and yaw define a 3D rotation matrix noted $R(\Psi)$. The intrinsic camera parameters, noted f_x, f_y, u_0, v_0 (focal length in horizontal and vertical axis f_x, f_y and the coordinates of the centre pixel (u_0, v_0)), are assumed to be accurately estimated by calibration and they are combined with the extrinsic camera parameters to create the projection matrix $P(\Psi)$:

$$P(\Psi) = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{ccc|c} & & & \psi_4 \\ & R(\Psi) & & \psi_5 \\ & & & \psi_6 \end{array}\right] \tag{1}$$

  The coordinate of the centre of the camera $C(\Psi)$ can then be computed with:

$$C(\Psi) = -\begin{bmatrix} & & \\ & R(\Psi) & \\ & & \end{bmatrix}' \begin{bmatrix} \psi_4 \\ \psi_5 \\ \psi_6 \end{bmatrix} \tag{2}$$

- $\mathbf{x} \in \mathbb{R}^3$ is an observed random variable corresponding to the pixel spatial positions and the depth value. For each camera, a set of observations for $\mathbf{x}$ has been collected. These sets are noted $\mathcal{S}_1 = \{\mathbf{x}_1^{(i)}\}_{i=1,\cdots,N_1}$ for the depth image recorded by camera 1, $\mathcal{S}_2 = \{\mathbf{x}_2^{(i)}\}_{i=1,\cdots,N_2}$ for the depth image recorded by camera 2, and so on up to $\mathcal{S}_C = \{\mathbf{x}_C^{(i)}\}_{i=1,\cdots,N_C}$ recorded by camera C.

- $\boldsymbol{\epsilon} \in \mathbb{R}^3$ is the random variable modelling the noise on depth images. Its distribution p_ϵ is assumed normal with mean zero and diagonal covariance matrix of bandwidth $h_1 = h_2 = 1$ for the uncertainty on the pixel positions, and $h_3 = 0.002$ the uncertainty about the depth values (obtained by calibration).

3.1. Link function F for SfD

The function $F(\mathbf{x}, \Psi, \Theta)$ that links the random variable $\mathbf{x}, \Theta, \Psi$ is defined as:

$$F(\mathbf{x}, \Psi, \Theta) = \begin{pmatrix} F_1(\mathbf{x}, \Psi, \Theta) \\ F_2(\mathbf{x}, \Psi, \Theta) \\ F_3(\mathbf{x}, \Psi, \Theta) \end{pmatrix} = \begin{pmatrix} x_1 - \frac{\theta_1 P(\Psi)_{11} + \theta_2 P(\Psi)_{12} + \theta_3 P(\Psi)_{13} + P(\Psi)_{14}}{\theta_1 P(\Psi)_{31} + \theta_2 P(\Psi)_{32} + \theta_3 P(\Psi)_{33} + P(\Psi)_{34}} \\ x_2 - \frac{\theta_1 P(\Psi)_{21} + \theta_2 P(\Psi)_{22} + \theta_3 P(\Psi)_{23} + P(\Psi)_{24}}{\theta_1 P(\Psi)_{31} + \theta_2 P(\Psi)_{32} + \theta_3 P(\Psi)_{33} + P(\Psi)_{34}} \\ x_3 - \sqrt{(C(\Psi)_1 - \theta_1)^2 + (C(\Psi)_2 - \theta_2)^2 + (C(\Psi)_3 - \theta_3)^2} \end{pmatrix} \tag{3}$$

and the stochastic equation used in our modelling is:

$$\lambda + F(\mathbf{x}, \Theta, \Psi) = \boldsymbol{\epsilon} \sim p_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon}) \tag{4}$$

The variable $\lambda \in \mathbb{R}^3$ is an auxiliary random variable that is added to help the modelling of the cost function (cf. section 3.2) and we are only interested in inferring information about Θ in the case $\lambda = 0$. Note that this modelling links explicitly the observed quantity $\mathbf{x}$ from the cameras with the additive perturbation $\boldsymbol{\epsilon}$. The first two functions (F_1, F_2) link the pixel positions to the latent 3D locations and was used in Ruttle et al. modelling to infer shape-from-silhouettes [2]. As an extension to [2], the last function F_3 relates the depth values to the latent 3D locations.

3.2. Cost function

From the stochastic equation (4), the conditional density of λ given $\mathbf{x}$, Θ and Ψ is:

$$p_{\lambda|\Theta\Psi\mathbf{x}}(\lambda|\Theta, \Psi, \mathbf{x}) = p_{\boldsymbol{\epsilon}}(\lambda + F(\mathbf{x}, \Theta, \Psi)) \tag{5}$$

Assuming independence between $\mathbf{x}$, Θ and Ψ, the joint density function $p_{\lambda\Theta\Psi}$ can be computed by:

$$\begin{aligned} & p_{\lambda\Theta\Psi}(\lambda, \Theta, \Psi) \\ & \quad = p_\Theta(\Theta)\, p_\Psi(\Psi) \int p_{\boldsymbol{\epsilon}}(\lambda + F(\mathbf{x}, \Theta, \Psi))\, p_{\mathbf{x}}(\mathbf{x})\, d\mathbf{x} \\ & \quad = p_\Theta(\Theta)\, p_\Psi(\Psi)\, \mathbb{E}\left[p_{\boldsymbol{\epsilon}}(\lambda + F(\mathbf{x}, \Theta, \Psi))\right] \end{aligned} \tag{6}$$

λ is an auxiliary variable and we focus now on the case of interest when $\lambda = 0$. $p_\Theta(\Theta)$ and $p_\Psi(\Psi)$ are the priors for the latent variable Θ and the nuisance variable Ψ. We define our cost function only using the expectation term. Using the observations collected (depth images) by the different cameras, the expectation can be replaced by its empirical mean [7]. For instance considering the observations $\mathcal{S}_1$ recorded by the first camera with extrinsic parameter Ψ_1, the expectation can be approximated by :

$$\mathbb{E}\left[p_{\boldsymbol{\epsilon}}(F(\mathbf{x}, \Theta, \Psi_1))\right] \simeq \frac{1}{N_1} \sum_{i=1}^{N_1} p_{\boldsymbol{\epsilon}}(F(\mathbf{x}_1^{(i)}, \Theta, \Psi_1)) \tag{7}$$

We note this expectation $\overline{\text{lik}}(\Theta, \Psi_1)$ because it can be understood as an average likelihood taking one observation at a time and each term $p_\epsilon(F(\mathbf{x}_1^{(i)}, \Theta, \Psi_1))$ represents a one-to-many mapping between the observation and the latent variables. Considering the C depth images collected, C average likelihood functions can be computed $\{\overline{\text{lik}}(\Theta, \Psi_c)\}_{c=1,\cdots,C}$ and merging all views for inference of the 3D shape when all extrinsic camera parameters are exactly known (noted $\hat{\Psi}_c, \forall c$) leads to

$$\overline{\text{lik}}(\Theta) = \sum_{c=1}^{C} \overline{\text{lik}}(\Theta, \hat{\Psi}_c) \tag{8}$$

Figure 1 shows this overall likelihood function $\overline{\text{lik}}(\Theta)$ using synthetic depth images generated using the Stanford bunny virtual object. All camera parameters are exactly known, and $\overline{\text{lik}}(\Theta)$ is a cost function modelling the surface of the object. The 3D shape can be inferred, for instance, by first computing this cost function on a fine grid covering the 3D space and then by thresholding to keep points on the surface. Alternatively gradient methods can also be used for inference [2].

3.3. Refining the nuisance parameters

In practice, in real environment, even with a careful calibration, the extrinsic camera parameters are not available and these are necessary to get a good cost function $\overline{\text{lik}}(\Theta)$ for inference of the shape. Choosing camera 1 as a reference camera ($\hat{\Psi}_1$ is available), we want to estimate the parameters $\{\Psi_2, \cdots, \Psi_C\}$ such that all average likelihoods overlap well in the 3D space. We formulate the problem as follow:

$$\forall c,\ \hat{\Psi}_c = \arg\max_{\Psi_c} \int \overline{\text{lik}}(\Theta, \hat{\Psi}_1)\ \overline{\text{lik}}(\Theta, \Psi_c)\ d\Theta \tag{9}$$

In practice, to compute this integral, we extract independent samples $\{\Theta^{(j)}\}_{j=1,\cdots,M}$ of the reference function $\overline{\text{lik}}(\Theta, \hat{\Psi}_1)$ and the integral becomes:

$$\int \overline{\text{lik}}(\Theta, \hat{\Psi}_1)\ \overline{\text{lik}}(\Theta, \Psi_c)\ d\Theta \simeq \sum_{j=1}^{M} \overline{\text{lik}}(\Theta^{(j)}, \Psi_c) \tag{10}$$

The estimate $\hat{\Psi}_c$ is then computed using an iterative gradient algorithm with an initial guess given by the initial calibration.

Note that we are only interested in recovering an accurate 3D surface of the object, not its exact position in the 3D world. Hence the parameters $\hat{\Psi}_1$ of the selected reference camera are not important and do not need to be accurate w.r.t. a known origin in the 3D world. What is important is that all other cameras are aligned perfectly with the reference camera.

To maximise overlap between average likelihood functions in this optimisation, while $\overline{\text{lik}}(\Theta, \Psi_2)$ is mapped on $\overline{\text{lik}}(\Theta, \hat{\Psi}_1)$ to estimate $\hat{\Psi}_2$, then $\overline{\text{lik}}(\Theta, \Psi_3)$ is then mapped on $\overline{\text{lik}}(\Theta, \hat{\Psi}_2)$ and so on. This process can lead to a propagation of errors on the estimated camera parameters. These errors can be reduced by having good initial guesses of the camera parameters by using standard camera calibration techniques. They can be further reduced by repeating the process with a different reference camera and by calibrating the cameras in reverse order.

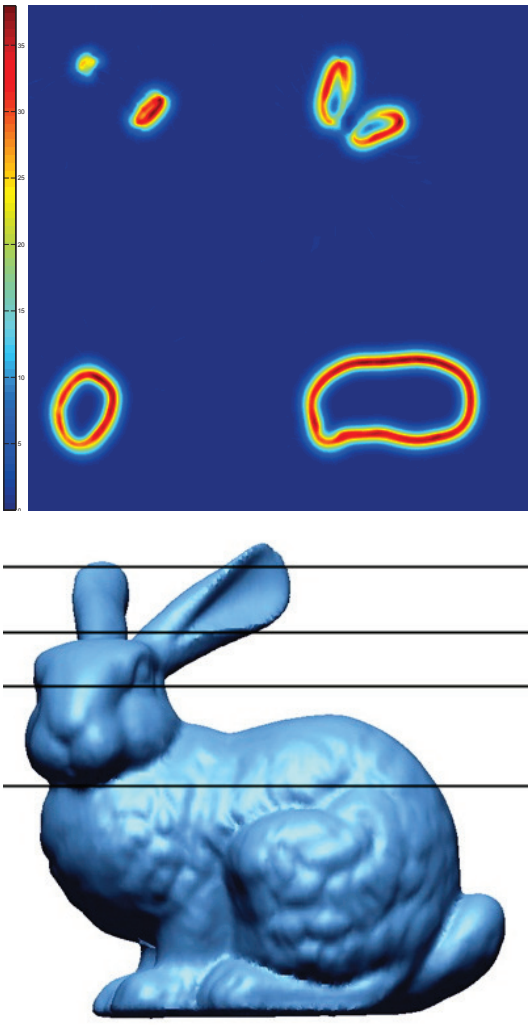
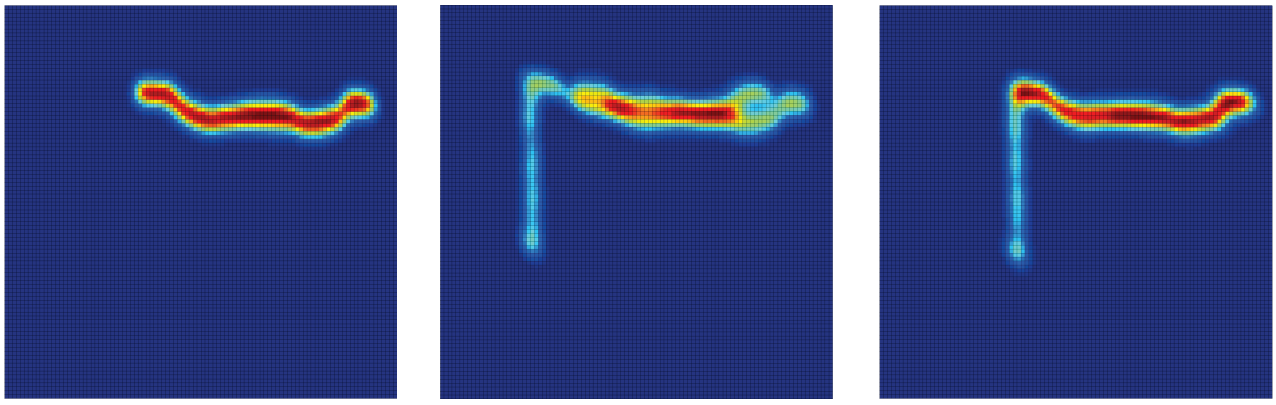


Fig. 1. Slices of the cost function $\overline{\text{lik}}(\Theta)$ (Top: top left corresponding to the ears, top right the middle of the ears, bottom left the head and bottom right the middle of the body) computed for depth images of the Stanford Bunny object (bottom) using 36 depth images generated around the object.

4. EXPERIMENTAL RESULTS

This framework has been used for infering 3D shapes from depth images recorded with a kinect camera with a turning table. Figure 2 shows a slice of the average likelihood functions before and after the extrinsic camera parameters have



$\overline{\text{lik}}(\Theta, \hat{\Psi}_1)$ $\overline{\text{lik}}(\Theta, \Psi_2^{(0)})$ overlapped with $\overline{\text{lik}}(\Theta, \hat{\Psi}_1)$ $\overline{\text{lik}}(\Theta, \hat{\Psi}_2)$ overlapped with $\overline{\text{lik}}(\Theta, \hat{\Psi}_1)$

Fig. 2. Reference average likelihood (left), reference average likelihood overlapped with the average likelihood from another camera using the initial guess $\Psi_2^{(0)}$ obtained by calibration (middle), and the two likelihoods overlapped after refinement of the camera parameters $\hat{\Psi}_2$ using our method (right).

been refined. Note the original mismatch between the two densities that is compensated for with the refinement of the camera parameters.

Figure 3 shows the camera positions and orientations in the horizontal plane in our setting using the kinect before and after the camera parameters have been refined. Indeed our setting uses a homemade turning table and the deviation of the true camera parameters from the original values obtained by calibration is quite important. The axis reports dimension in meters.

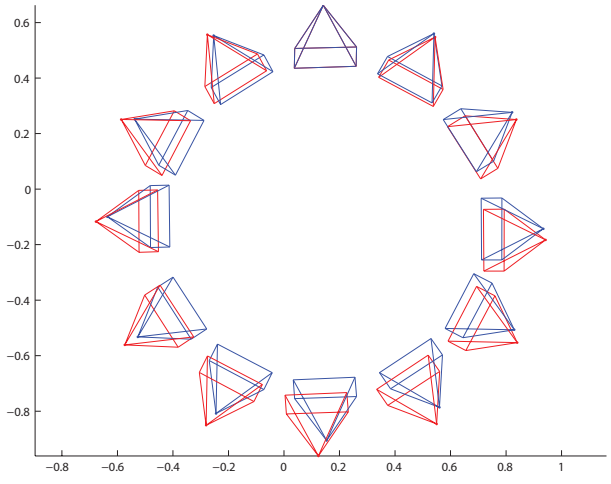


Fig. 3. Camera parameter refinement using a Kinect and a turning table. The red cameras show the original position and orientation of the cameras. The blue cameras show the position and orientation after refinement. Note the reference camera at the top.

Figure 4 illustrates the 3D meshes obtained for several objects that have been captured with the turning table and a Kinect camera. Reconstructions using sihouettes [2] and depth images are shown for comparison. While silhouette images do not provide information about the concavities of the object, depth images allows these concavities to be well recovered. These reconstructions are only accurate if all camera parameters have been accurately estimated. The real world dimensions of the objects are in centimeters: $18 \times 15 \times 32$ (Gnome) and $18 \times 16 \times 26$ (Lighthouse). Some very small details are not recovered (eg. roof tiles of the lighthouse), but the inferred meshes are far less noisy and more detailled than the original depth scans.

5. CONCLUSION

We have extended the Shape from Silhouettes (SfS) modelling [2] to Shape from depth images (SfD). This modelling like any other modelling for 3D reconstruction requires to estimate accuratetly the extrinsic camera parameters, and we have proposed to estimate robustly these parameter using the correlation between probability density functions. Experimental result shows that these nuisance parameters can be refined from these obtained in the calibration stage with greater accuracy.

6. REFERENCES

[1] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2004.

[2] Jonathan Ruttle, Michael Manzke, and Rozenn Dahyot, "Smooth kernel density estimate for multiple view recon-

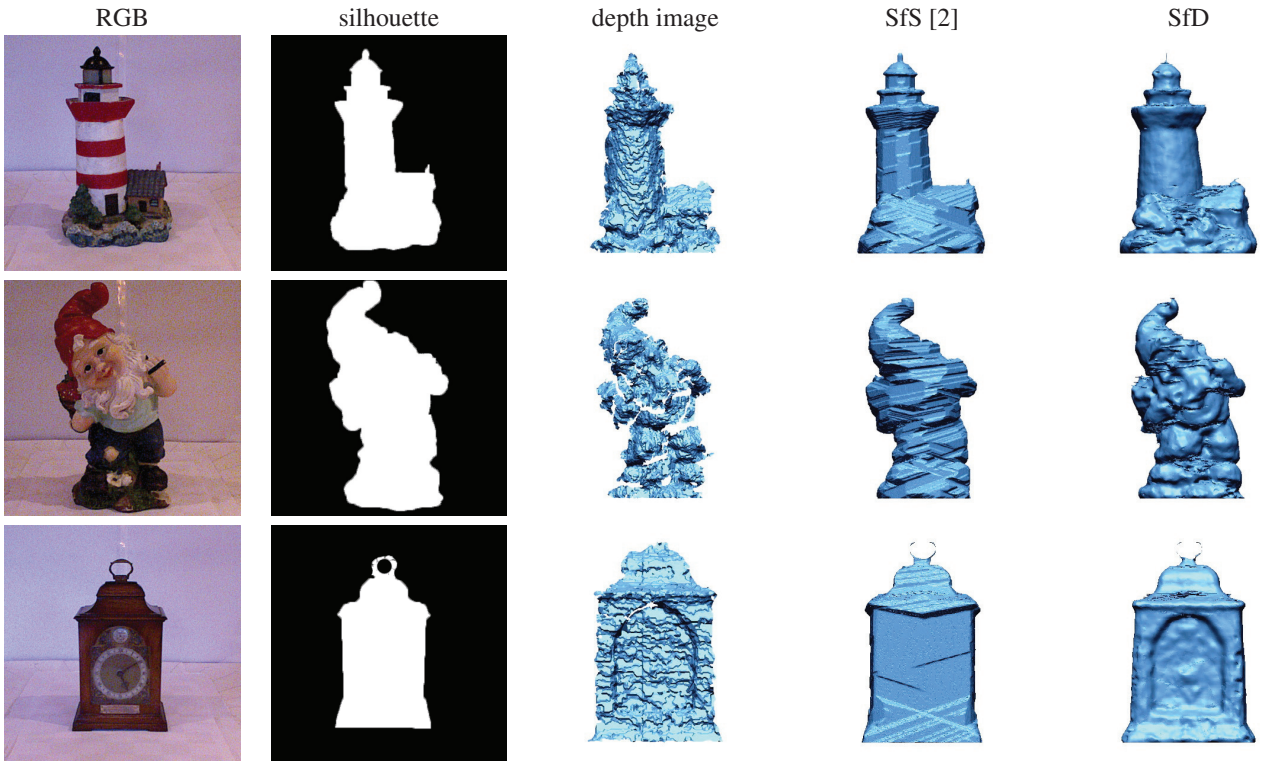


Fig. 4. 3D surface reconstruction using 12 camera views evenly distributed in the horizontal plane. From left to right for one view: RGB image, silhouette, depth image, reconstruction using the 12 silhouettes using Ruttle et al modelling [2] and reconstruction with depth images using $\overline{\text{lik}}(\Theta)$ (eq. (8)).

struction," in *proceedings of The 7th European Conference for Visual Media Production, CVMP 2010*, 17 - 18 November 2010, pp. 74 –81.

[3] Yan Cui, Sebastian Schuon, Chan Derek, Sebastian Thrun, and Christian Theobalt, "3d shape scanning with a time-of-flight camera," in *IEEE conference on Computer Vision and Patern Recognition, CVPR*, 2010.

[4] Y. Tsin and T. Kanade, "A Correlation-Based Approach to Robust Point Set Registration," *European Conference in Computer Vision ECCV*, pp. 558–569,2004.

[5] B. Jian and B. Vemuri, "Robust point set registration using gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[6] David W. Scott, "Parametric statistical modeling by minimum integrated square error," *Technometrics*, vol. 43, no. 3, pp. pp. 274–285, 2001.

[7] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer Verlag, 1999.