An Advanced Visual Tracking Robot with Quality Estimation

Deepthi.N[1] Catherine Rakkini.D[2] Alamelu Mangai.M[3]
[1,2]PG Student [3]Assistant Professor
[1,2,3]Department of Electrical & Electronics Engineering
[1,2,3]Nehru Institute of Engineering and Technology, Coimbatore

Abstract— the handling of moving objects is a difficult problem in this regard because it requires that the position of the objects is determined with sufficient accuracy. Industrial paint lines are often built using an overhead conveyor with hanging trolleys that carry the objects to be painted. In the standard industrial solution, the trolleys are swinging freely to avoid mass forces on the trolley, and to make it easy to change trolleys. The swinging motion of the trolleys makes it difficult to move the robot to the correct delivery position. It is therefore difficult to load and unload objects on such hanging trolleys. Another problem with paint lines is that is it difficult to determine which loading positions that carry objects. This is important to avoid wasting paint on loading positions where there are no object. The first contribution is a system for accurate tracking of a hanging trolley so that a robot can place objects on the trolley while it is in motion. The second contribution is the detection of which loading positions on the hanging trolley that carry objects.

Key words: Visual Tracking of Moving Objects, Advanced Visual Tracking Robot

I. INTRODUCTION

Industrial paint lines are often built using an overhead conveyor with hanging trolleys that carry the objects to be painted .In the standard industrial solution the trolleys are swinging freely to avoid mass forces on the trolley, and to make it easy to change trolleys. The swinging motion of the trolleys makes it difficult to move the robot to the correct delivery position. It is therefore difficult to load and unload objects on such hanging trolleys. Another problem with paint lines is that is it difficult to determine which loading positions that carry objects. This is important to avoid wasting paint on loading positions where there are no objects. In this paper we present two contributions to the automation of this industrial task. The first contribution is a system for accurate tracking of a hanging trolley so that a robot can place objects on the trolley while it is in motion. The second contribution is the detection of which loading positions. This is useful for validating that the loading procedure is successful, and to improve tracking by not using occluded image features in the likelihood function.

II. VISUAL TRACKING OF MOVING OBJECTS

A. *Particle Filtering*

The more non-linear model, or the more non-Gaussian noise, the more potential particle filters have, especially in applications where computational power is rather cheap and the sampling rate is moderate. The particle filter approximates the optimal solution numerically based on a physical model, rather than applying an optimal filter to an approximate model. A well-known problem with the particle filter is that its performance degrades quickly when the dimension of the state dimension increases. Particle filter algorithm consists of four steps and they are,

1) Initialization
2) Measurement update
3) Re-sampling
4) Prediction

B. *Extended Kalman Filters*

The Extended Kalman filter (EKF) gives an approximation of the optimal estimate. The non-linearity of the system's dynamics are approximated by a linearized version of the non-linear system model around the last state estimate. This approximation to be valid, this linearization should be a good approximation of the non-linear model in all the uncertainty domain associated with the state estimate. Rather than propagating the non-Gaussian pdf, the Extended Kalman filter considers, at each cycle, a linearization of the non-linear dynamics around the last consecutive predicted and filtered estimates of the state, and for the linearized dynamics, it applies the Kalman Filter. It this important to state that the EKF is not an optimal filter, but rather it is implemented based on a set of approximations. This algorithm consists of two cycles and they are,

- Predict cycle
- Filtered cycle

C. *Rao-Blackwellization*

An estimator $\delta(X)$ is an observable random variable (i.e. a statistic) used for estimating some unobservable quantity. For example, one may be unable to observe the average height of all male students at the University of X, but one may observe the heights of a random sample of 40 of them. The average height of those 40—the "sample average"—may be used as an estimator of the unobservable "population average". A sufficient statistic $T(X)$ is a statistic calculated from data X to estimate some parameter θ for which it is true that no other statistic which can be calculated from data X provides any additional information about θ. It is defined as an observable random variable such that the conditional probability distribution of all observable data X given $T(X)$ does not depend on the unobservable parameter θ, such as the mean or standard deviation of the whole population from which the data X was taken. In the most frequently cited examples, the "unobservable" quantities are parameters that parametrize a known family of probability distributions according to which the data are distributed.

In other words, a sufficient statistic $T(X)$ for a parameter θ is a statistic such that the conditional distribution of the data X, given $T(X)$, does not depend on the parameter θ.A Rao–Blackwell estimator $\delta_1(X)$ of an unobservable quantity θ is the conditional expected value $E(\delta(X) \mid T(X))$ of some estimator $\delta(X)$ given a sufficient statistic $T(X)$. Call $\delta(X)$ the "original estimator" and $\delta_1(X)$ the "improved estimator". It is important that the improved estimator be observable, i.e. that it not depend on θ. Generally, the conditional expected value of one function of these data given another function of these data does depend on θ, but the very definition of sufficiency given above entails that this one does not.

D. *Fisher Information Matrix*

In mathematical statistics, the Fisher information (sometimes simply called information) is a way of measuring the amount of information that an observable random variable *X* carries about an unknown parameter *θ* of a distribution that models *X*. Formally, it is the variance of the score, or the expected value of the observed information. In Bayesian statistics, the asymptotic distribution of the posterior mode depends on the Fisher information and not on the prior. The role of the Fisher information in the asymptotic theory of maximum-likelihood estimation was emphasized by the statistician Ronald Fisher (following some initial results by Francis Ysidro Edgeworth). The Fisher information is also used in the calculation of the Jeffreys prior, which is used in Bayesian statistics.

III. NEED FOR THE PROJECT

The project serves as an aid for improved performance in automatic loading of trolleys hanging from a moving overhead conveyor. It also provides a robotic solution that describe a method for the interaction of an industrial robot and a free swinging object. This is based on visual tracking using particle filtering where the equations motion of the object are included in the filtering algorithm. Thus the project focuses on handling moving objects with robot manipulators and tracking of objects with high accuracy.

IV. MODELLING

A. *Particle Filtering*

Visual tracking using particle filters involves the estimation of the probability density function of a hidden Markov model

$$p(xk|I1{:}k) \qquad (1)$$

Where xk is the state vector and I 1:k is the sequence of images from time 1 to timek.

At each time-step k the particle filter has a set of N particles

$$X_k = (w_k^i, x_k^i),\ \sum_{i=1}^{N} w_k^i = 1 \qquad (2)$$

as a numerical approximation to the probability density function in(1)

$$p(xk|I1{:}k) \approx \sum_{i=1}^{N} \delta(x_k - x_k^i) \qquad (3)$$

In this paper the well-known Sequential Importance sampling with Resampling (SIR) procedure.

B. *Dynamic Model*

The hanging trolley can be considered as a rigid body rotating around a fixed point. The kinematic configuration is therefore described by a point in SO(3), here represented by a unit quaternion

$$q=\begin{bmatrix}\mu\\ \varepsilon\end{bmatrix}, \mu \in R,\ \varepsilon \in R^3 \qquad (4)$$

The dynamics of the trolley is described by the Euler equations of motion in the body coordinate system *b*

$$M\dot{\omega}\ b + \omega b \times M\omega b = pb \times gb + vb \qquad (5)$$

Where M ∈ R3×3 is the inertia matrix, pb ∈ R3 is the center of mass position, gb ∈ R3 is the vector of gravity and vb is the noise vector.

C. *Image Kinematics*

The image is a two dimensional array of pixel intensities I(p) where p = (u, v, 1)T and u and v are the integer pixel coordinates. The image intensity I(x) at coordinates of x that are not integer values is computed with linear interpolation between the neighboring pixels. It is assumed that the camera calibration matrix is known and given by

$$K = \begin{bmatrix}\alpha_x & 0 & \mu_o\\ 0 & \alpha_y & v_o\\ 0 & 0 & 1\end{bmatrix} \qquad (6)$$

As the trolley consists of line segments, and the configuration of the trolley is given by the angle θ. Given the angle θ and the position of two fixed points Pb and Qb on each line segment, it is possible to calculate the position of the two points in camera frame c coordinates from Pc = T cb Pb and Qc = T cbQb where the points are given in homogeneous coordinates. The corresponding homogeneous pixel coordinates p and q are found from,

p = K(R t) Pb

q = K(R t) Qb

Where R is the rotation matrix and t is the translation vector of the homogeneous transformation matrix from the body frame b to the camera frame c. The line segment defined by Pb and Qb is found in pixel coordinates as the line segment through p and q, and can be described as the homogeneous vector l = (a, b, c)T, which is found from the cross product,

$$l = \gamma S(q)p$$

Where S(q) is the skew symmetric form of q. A scaling factor γ is used to ensure that the two dimensional vector n = (a, b)T is a unit vector. It is noted that in this description the vector n is the normal vector to the line segment in image coordinates.

D. *Likelihood Function*

The following likelihood function is proposed for the particle filter:

$$L = \sum_i L_i \qquad (7)$$

Here Li = wiDi/(j wj) is the likelihood function for line segment i, wi is a weighting constant, and Di is the edge detection function for line segment i given by

$$Di = [I(pi) - I(pi + \lambda ni)], \qquad (8)$$

Where pi is the two-dimensional version of the homogeneous vector pi, ni is the normal vector to line segment i, and λ is a parameter which was set to λ = 5, which gives a distance between the points of approximately 5 pixels. The main contribution of the approach is to compute the weights wi of the likelihood function from the associated Fisher information matrix. The advantage of this approach is that the features which provides the most information is given a higher weighting, which leads to improved accuracy. The following analysis explains how the weights are selected. The Fisher information matrix is found from the second order Taylor approximation of the log likelihood function at θ0:

$$\ln Li(\theta) \approx \ln Li(\theta_0) + Si(\theta-\theta_0) - 1/2(\theta-\theta_0)TFi(\theta_0)(\theta-\theta_0) \qquad (9)$$

Here Si is called the score and

$$Fi(\theta_0) = \frac{\partial^2 \ln(L_i(\theta))}{\partial \theta^2} \qquad (10)$$

is the Fisher information matrix.

E. *Detection of Objects on Trolley*

In this section we propose an extension to the particle filter to make it possible to detect the presence of an object on a particular trolley. To achieve this, we augment the dynamic model (8) with one state $\xi(1)$ and $\xi(2)$ for each of the two loading positions. If $\xi(i)$ is greater than a threshold value, then there is no object on the loading position. If $\xi(i)$ is smaller than this threshold value, then there is an object on the loading position. The state vector is then

$$x_k = [q^T \ \vartheta^T \ \varepsilon^1 \ \varepsilon^2]_K^T \quad (11)$$

To achieve this an occlusion detection function Oj(li) for each panel is introduced.. The occlusion detection function is defined as

Oj(li) = 1 if panel j occludes line segment li

Oj(li) = 0 if line segment li is visible where the actual occlusion detection is performed using a ray-in-triangle test .The sum term adds up the number of visible line segments that would have been occluded by a pane at j, which means that the sum term is large if the line segments that would have been occluded by a panel j are detected as visible, which will cause $\xi(j)$ *k* to become large. If the line segments that would have been occluded by a panel *j* are not detected as visible, then the sum term is small, then this is an indication that the panel is there, and $\xi(j)$ *k* will tend to zero.

V. Methodology

A. General

Handling moving objects with robot manipulators is a challenging task as it involves tracking of objects with high accuracy. An industrial application of this type is the loading and unloading of objects on an overhead conveyor. A robotic solution to this problem is presented in this paper, where we describe a method for the interaction of an industrial robot and a free swinging object. Our approach is based on visual tracking using particle filtering where the equations of motion of the object are included in the filtering algorithm. The first contribution of this paper is that the Fisher information matrix is used to quantify the information content from each image feature. In particular, the Fisher information matrix is used to construct a weighted likelihood function. This improves the robustness of tracking algorithm significantly compared to the standard approach based on an unweighted likelihood function. The second contribution of this paper is that we detect occluded image features, and avoid the use of these features in the calculation of the likelihood function. This further improves the quality of the likelihood function. The project aims in concatenating different algorithms for troubleshooting their problem. The algorithm is done using the MATLAB R2013 a software which is capable of processing with real time images. The software allows debugging and run for each line of code. This feature allows easier debugging and troubleshooting. The methodology can be represented can be represented in a flow chart as given below:

1) Read the Image

By specifying the correct location read the image and it will be in matrix form.

2) Pre Processing

This is the steps which are done before interpolation to get a noise free enhanced image.

- Noise removal: It is done by using corresponding filters.
- Image enhancement: It is done to increase the contrast by equalizing the histogram.

3) Image Interpolation Technique

Two methods are done for a comparison between them. An image f(x,y) tells us the intensity values at the integral lattice locations, i.e., when x and y are both integers. Image interpolation refers to the “guess” of intensity values at missing locations, i.e., x and y can be arbitrary. Image interpolation is about D-A conversion. Limitation of bicubic and bilinear: Edge blurring and Jagged artifacts.

a) Nearest Neighbor

Probably the most basic form of interpolation. As the actual pixels are proportionally copied to their new locations, their position in relation to one another remains the same. Since the image is enlarged, filler pixels must be placed in between the actual pixels. With the most basic nearest neighbor interpolation, just copy the exact same pixel values over to the filler pixel closest to the pixel. Since my images have been initialized with the pixel at 0,0 being the same pixel in the original and enlarged image, I choose the pixel to the right or the pixel below, dependant on where the filler pixel is placed.

b) Linear Interpolation

A better algorithm than nearest neighbor that takes into account the gradual transition of pixel color values. By finding the means between two pixel values, the filler pixel is better suited for overall image enhancement. In other words, it just looks plain better. Once again we return to the trusty diagram.

A texture mapping technique that produces a reasonably realistic image, also known as "bilinear filtering" and "bilinear texture mapping." An algorithm is used to map a screen pixel location to a corresponding point on the texture map. A weighted average of the attributes (color, alpha,etc.) of the four surrounding texels is computed and applied to the screen pixel. This process is repeated for each pixel forming the object being textured. The term bilinear refers to the performing of interpolations in two dimensions (horizontal and vertical). The top and bottom pairs of each texel quadrant are averaged (horizontal) and then their results are averaged (vertical). This method is often used in conjunction with MIP mapping.

4) Visual tracking and servoing

Visual tracking of moving objects and visual servoing with robots have received much attention in recent years as the computational power has increased. One approach to tracking is to use extended Kalman filters. Another widely used approach is particle filtering, which can handle nonlinearities and probability density functions that are not Gaussian.

5) Result

Result is obtained through existing methods like particle filtering, extended Kalman filters and rao-blackwellisation. This is compared with proposed method fisher information matrix.

VI. Project Phase Overview

A. Description

The project has developed a system for accurate tracking of a hanging trolley so that a robot can place objects on the trolley while it is in motion. The second contribution is the detection of which loading positions on the hanging trolley that carry objects. This is useful for validating that the loading procedure is successful, and to improve tracking by not using occluded image features in the likelihood function in this project, we used two GC1020 cameras from AVT. The cameras were connected via Ethernet to a computer running the tracking algorithm. The computer used Ubuntu 14.04 with an i7-3820 CPU and 16 GB of RAM. Most of the particle filter computations were implemented in CUDA and were run on a Nvidia GeForce GTX Titan card.

B. *Block Diagram*

Here, the block diagram consists of embedded part and robotic part. Embedded part consists of LCD Display, Power supply, level sensor, IR sensor RS232, cameras and PC. Robotic part consists of PIC, pick and place robot and obstacle sensor. To interface with robotic arm, we use the Matlab2013a**.**

1) *Embedded Part*

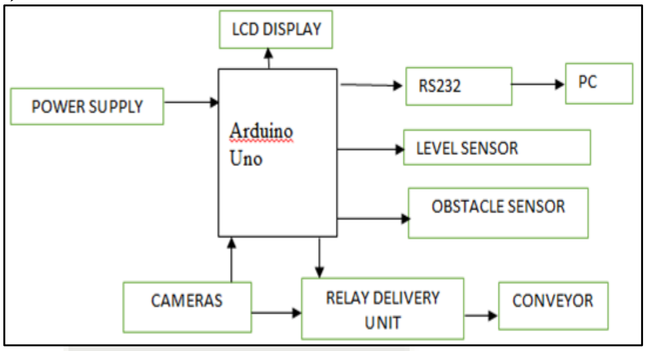


Fig. 1: Embedded Part

2) *Robotic Part*

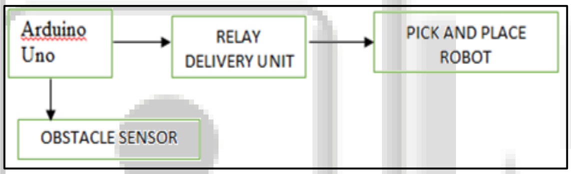


Fig. 2: Robotic Part

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. A relay is an electrically operated switch. Many relays use an electromagnet to operate a switching mechanism mechanically, but other operating principles are also used. Relays are used where it is necessary to control a circuit by a low-power signal (with complete electrical isolation between control and controlled circuits), or where several circuits must be controlled by one signal. This device emits and/or detects infrared radiation to sense a particular phase in the environment. Generally, thermal radiation is emitted by all the objects in the infrared spectrum. The infrared sensor detects this type of radiation which is not visible to human eye. Fuel level sensor DUT-E is designed for precision fuel level measurement in all kinds of vehicle tanks, also in tanks of fixed installations. DUT-E can be used as a part of a fuel monitoring system or to substitute the standard fuel meter of a vehicle.

VII. Result and Discussion

A. *Result*

The fisher information matrix is used for visual tracking and visual servoing. The original product compared with defected products using fisher information matrix based on the change in kinematic parameter values. In addition the output also validate the superiority of the proposed scheme which have provided high performance efficiency**.**

B. *Output*

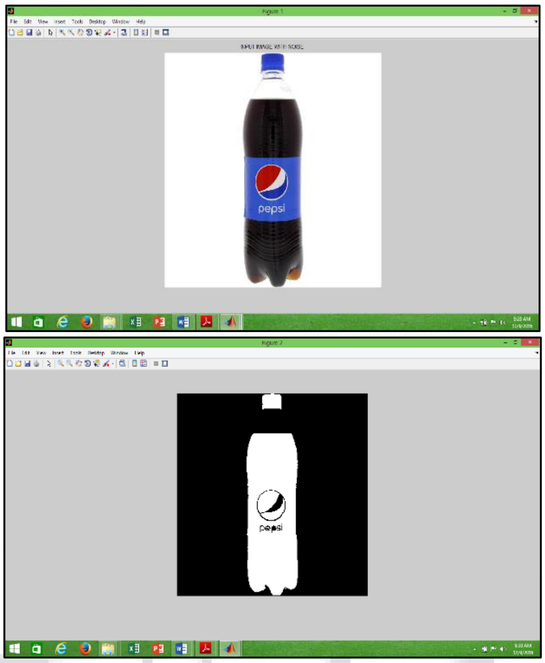


Fig. 3: Input product compared with defected product.

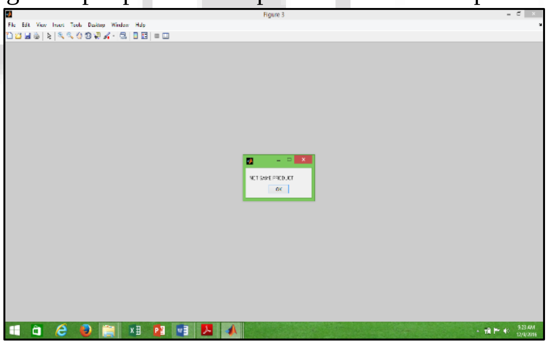
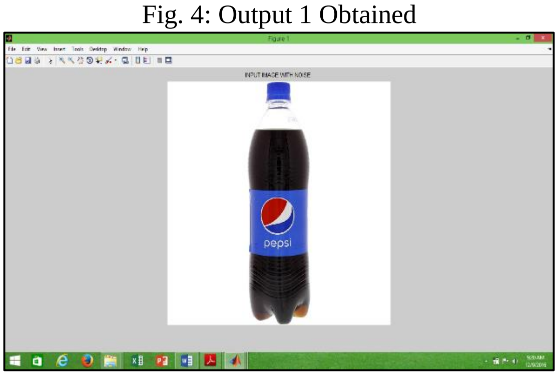


Fig. 4: Output 1 Obtained



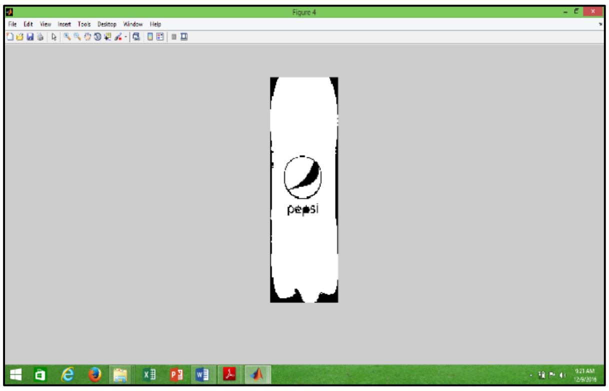


Fig. 5: Input product is compared with perfect product

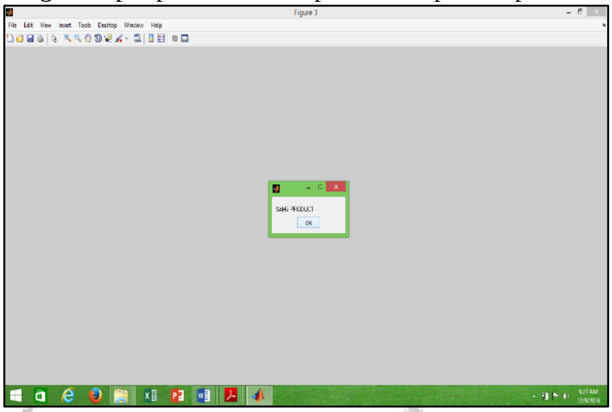


Fig. 6: Output 2 Obtained

VIII. CONCLUSION AND FUTURE WORK

Two techniques were introduced to improve the tracking performance. The first technique is based on weighting the likelihood function using the Fisher information matrix of the image features. Intuitively this means that the particle filter is paying more attention to image features with high information content about the kinematic parameter of interest. This was shown to improve tracking performance in simulations and experiments in situations where the proposal distribution were badly conditioned for the physical system. The second technique is a method to detect whether objects were hanging on the loading position on the trolleys. This information was used to further improve tracking performance.

In the future we would like to test our proposed method with a weighted likelihood function on other types of tracking problems. We would also like to investigate how to use this method when there are several kinematic parameters of interest.

REFERENCES

[1] A.Blake, M. Isard et al., Active contours. Springer London, 2000, vol. 1.
[2] A.Doucet, N. De Freitas, K. Murphy, and S. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 2000, pp. 176–183.
[3] A.J. Davison, "Active search for real-time vision," in Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, vol. 1.IEEE, 2005, pp. 66–73.
[4] C. Choi and H. I. Christensen, "Robust 3D visual tracking using particle filtering on the SE (3) group," in Proc. IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2011, pp. 4384–4390.
[5] E. Hairer, S. P. Nørsett, and G. Wanner, Solving ordinary differential equations I: Nonstiff problems. Springer Science & Business, 2008, vol. 1.
[6] F. Pukelsheim, Optimal design of experiments. siam, 2006, vol. 50.
[7] F. Gustafsson, Statistical sensor fusion. Studentlitteratur,, 2010.
[8] J. Kwon and F. C. Park, "Visual tracking via particle filtering on the affine group," The International Journal of Robotics Research, vol. 29, no. 2-3, pp. 198–217, 2010
[9] J. G. Proakis, Digital signal processing: principles algorithms and applications. Pearson Education India, 2001.
[10] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review,"Statistical Science, pp. 273–304, 1995
[11] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density," in Computer VisionECCV'96. Springer, 1996, pp.343–356.
[12] M. Chli and A. J. Davison, "Active matching," in Computer Vision–ECCV 2008. Springer, 2008, pp. 72–85.
[13] O. Egeland and J. T. Gravdahl, Modeling and simulation for automatic control. Marine Cybernetics, Trondheim, Norway, 2002
[14] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge., 2003.
[15] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," IEEE Transactions on Robotics and Automation, vol. 12, no. 5, pp. 651–670, 1996.
[16] T. M¨oller and B. Trumbore, "Fast, minimum storage ray-triangle intersection,"Journal of graphics tools, vol. 2, no. 1, pp. 2128,1997
[17] V. Lippiello, B. Siciliano, and L. Villani, "Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration," IEEE Transactions on Robotics, vol. 23, no. 1, pp. 73–86, 2007.
[18] V. I. Arnol'd, Mathematical methods of classical mechanics. Springer,1989, vol. 60
[19] Y. Pawitan, in all likelihood: statistical modelling and inference using likelihood. Oxford University Press, 2001.

