

Student Mixture Model Based Visual Servoing

Mithun. P^a, Shaunak A. Mehta^b, Suril V. Shah^{b,*}, Gaurav Bhatnagar^b, K. Madhava Krishna^a

^a*International Institute of Information Technology, Hyderabad-500032, India*

^b*Indian Institute of Technology, Jodhpur-342037, India*

Abstract

Classical Image-Based Visual Servoing (IBVS) makes use of geometric image features like point, straight line and image moments to control a robotic system. Robust extraction and real-time tracking of these features are crucial to the performance of the IBVS. Moreover, such features can be unsuitable for real world applications where it might not be easy to distinguish a target from rest of the environment. Alternatively, an approach based on complete photometric data can avoid the requirement of feature extraction, tracking and object detection. In this work, we propose one such probabilistic model based approach which uses entire photometric data for the purpose of visual servoing. A novel image modelling method has been proposed using Student Mixture Model (SMM), which is based on Multivariate Student's t -Distribution. Consequently, a vision-based control law is formulated as a least squares minimisation problem. Efficacy of the proposed framework is demonstrated for 2D and 3D positioning tasks showing favourable error convergence and acceptable camera trajectories. Numerical experiments are also carried out to show robustness to distinct image scenes and partial occlusion.

Keywords: Visual servoing, mixture model, student's t -distribution

*Corresponding author

Email address: surilshah@iitj.ac.in (Suril V. Shah)

1. Introduction

In robotic applications where direct control by human intervention is inaccurate and error prone, an autonomous operation can be a more reliable solution. To achieve this autonomy the control system must rely on sensory feedback. Computer vision serves as a powerful component of such a system. The information obtained from such a vision sensor can be used for robot control tasks such as object inspection, grasping, tracking a seam in arc welding etc. Visual servoing [1, 2] is one such approach where a control law regulates an error in the image space to achieve desired positioning of the robot. To achieve this, a set of visual features has to be selected from the image of the object of interest as a primary step. Following this, the current and desired pose can be defined in terms of the selected visual features. Once the feature selection is done, an error function between desired and current image features is modeled. Subsequently, a control law [3, 4, 5] is designed to provide motion to the camera such that the error between current and desired image features is minimized to zero leading to the required configuration of the robot. The IBVS commonly uses points [6],[7],[8], lines, contours [9],[10] optical flow estimates, etc. as the features. For such features real-time feature tracking and correspondence are important but computationally involved steps of vision-based control. Various approaches and advancements have been proposed for feature tracking [9, 11, 12] to overcome this problem upto a limit but it is an unavoidable step. An extension of these approaches has been proposed in [13] where shape and moments are considered as visual features. These features provide more intuitive than the geometric features but a segmentation step is still required before extraction of features. Vision-based control technique applicable to real life environments with diverse objects and robust to noises are the main challenges of IBVS. Therefore, the present work proposes a solution to overcome these challenges.

Previous approaches rely mainly on geometric features. However, additional computations for feature extraction, matching and real-time tracking are some overheads for these visual servoing methods. In this regard, researchers have

concentrated towards feature less tracking to avoid the drawbacks of geometric approaches. A work in that direction has been initially presented in [14] and [15] where image intensity was used as indirect visual features to perform control action. Later image intensity was directly considered to perform 2D matching of images without any feature extraction in [16] and [17]. Here, a kernel projected value of intensities in image at each pixel and information theoretic approach of mutual information were illustrated. A similar method called Photometric visual servoing developed in [18] used image intensities as visual features. Use of the concept of Sum of Conditional Variance [19] results in a direct visual servoing task which is easy and fast to compute. Even though it is robust towards non-linear illumination variations, it is less robust towards local variations. A depth map obtained from a range sensor was used as a visual feature in [20]. This approach is both direct (without any 3D pose estimation) and dense (without feature extraction) but it requires an expensive range sensor which may have problems like noise in data and the absence of measurements. The above limitation is overcome in [21] where a histogram is considered as visual feature. Here, the applicability of the framework was shown to many histograms, like, intensity histograms, Histograms of Oriented Gradients (HOG) and colour histograms. However, this method suffers from low convergence and impact from uninformative image regions of an image under consideration. Although the above mentioned approaches vary in efficiency and precision, it also suffers from several drawbacks. For example, usage of direct image intensities makes it sensitive to illumination changes. Having a relatively small convergence domain of the cost function for these methods requires large overlapping regions between the current and desired images for the algorithm to converge. Many variations of direct visual servoing methods have been studied by considering different image and cost function representations (Photometric Moments [22], Principal Component Analysis [23], wavelet transforms [24], Discrete Cosine Transform [25]) to improve the convergence and robustness. A new approach using feature based mixture models in visual servoing has been initially studied in [26]. Here, feature points extracted from current and desired images were represented by

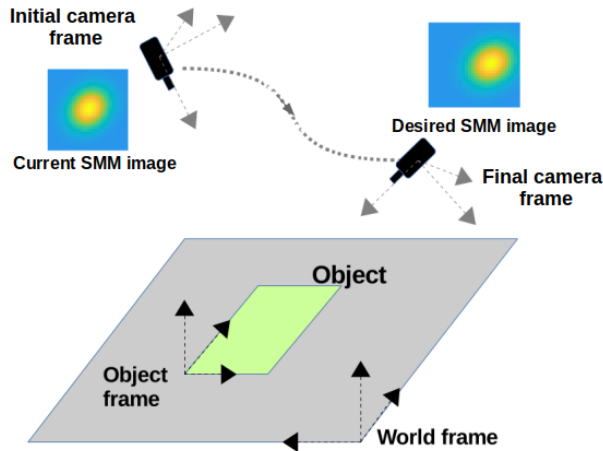


Figure 1: Performing SMM based visual servoing using a 6-DOF camera towards an object. The servoing task requires the camera to move to the desired view of target such that an error between proposed SMM images become zero.

two separate mixture models and the visual servoing minimized the difference between the two mixture models. It is advantageous since it considers feature position uncertainties by probabilistic models. However, the feature point detection is still a mandatory step and points extracted from the current images has to be the same as the points extracted from the desired image. Furthermore, it is implemented to control only three degrees-of-freedom (DOF) of the robot and also lacks an analytical formulation of interaction matrix. In [27], dense features were used to model mixture models alternatively to the above method. The main idea of this method is to model every pixel in the image as Gaussian function, and the algorithm tries to reduce the error between desired and current gaussian function of images. However, suffers from the drawback that it requires tuning of an extension parameter experimentally for providing overlapping of image regions.

In this paper, we propose a novel approach to visual servoing (see Fig. 1) using student t -distribution mixture model. In literature, it can be seen that t -distribution mixture models are successfully used in image processing applications like image registration [28]. Geometrically a t -distribution function has

heavily-tailed alternative to the normal distributions with same symmetric and bell-shaped density function. Such a characteristics of t -distribution guarantees inclusion of values which is far from its mean value. This property serves as one of the motivation for the proposed visual servoing framework. The general idea of proposed SMM based visual servoing techniques is to minimize error between current and desired SMMs. These SMMs are obtained from corresponding current and desired images, where each image is represented by mixture of t -distributions. This way to proceed not only has the advantages of the direct photometric method but also increases the convergence domain considerably. The proposed framework of visual servoing using Student's t -distribution is not reported in the literature to the best of authors knowledge and forms one of the fundamental contributions of this work. When the image is modeled using t -distributions, the control output results in better and smoother convergence than other classical servoing methods. Hence, the present work makes the following contributions.

- Image modeling using Student t -distribution Mixture Model (SMM) and an analytical formulation of interaction matrix.
- SMM based framework for IBVS without any requirement of feature tracking and correspondence.
- Systematic experimental validation of proposed framework.

The paper is organized as follows. Section 2 presents outline of basic visual servoing framework and the concept and formation of multivariate student t -distribution. This forms the foundation of the proposed research work. Section 3 describes outline of the proposed SMM based visual servoing method. The initial part of Section 4 explains modeling of an image using SMM and later part explains the development of the interaction matrix. Section 5 explains the modeling of visual servoing framework as an optimization problem. Section 6 provides extensive experiments which validate task completion, robustness and convergence. Finally, conclusions and future works are presented in Section 7.

2. Preliminaries

This section presents some preliminaries that surface behind the framework presented. Initially, an introduction to basic visual servoing control law is provided. Later, discussion on Student's t -distribution is carried out which forms a basis for the proposed framework.

2.1. Visual servoing

Visual servoing uses visual information for autonomous control of a robot. On a very basic level the vision based control proposes that a set of simple visual behaviours can be used to accomplish various application tasks. In such applications the visual information extracted from a camera is used as a feedback to close the control loop [4] so that the robot can reach the desired position accurately. In classical approach [5], the visual information extracted from the images are in the form of a set of features \mathbf{s} . Therefore, the basic working principle of any IBVS method is to minimise the error between the current features \mathbf{s} and the desired features \mathbf{s}^* , i.e.,

$$\mathbf{e} = (\mathbf{s} - \mathbf{s}^*). \quad (1)$$

Here $\mathbf{s} \in R^k$ is the feature vector of k feature points. The minimization of such an error function (1) is undertaken by providing appropriate velocity to the camera mounted on the end-effector of the robot. The relationship between the camera velocity and feature rate is given by

$$\dot{\mathbf{s}} = \mathbf{L}\mathbf{t}_c \quad (2)$$

where $\mathbf{t}_c \in R^6$ is camera velocity consisting of linear and angular velocities $(\mathbf{v}, \boldsymbol{\omega})$, and $\mathbf{L} \in R^{k \times 6}$ is the interaction matrix. If the desired feature in (1) is not changing with time, error rate can be expressed as $\dot{\mathbf{e}} = \dot{\mathbf{s}}$. As the main objective of visual servoing is to nullify the error function defined in (1), an exponential decay of error is selected, i.e.,

$$\dot{\mathbf{e}} = -\lambda\mathbf{e}, \quad (3)$$

where λ is a scalar gain which determines the speed of convergence. Upon substituting (3) in to (2), the visual servoing controller can be expressed as

$$\mathbf{t}_c = -\lambda \mathbf{L}^+ \mathbf{e}. \quad (4)$$

Here \mathbf{L}^+ is the pseudo inverse of the image Jacobian. The specific form of the interaction matrix depends on the features that are used to define \mathbf{s} . The above control law minimizes the error in feature space by guiding the camera with rate, \mathbf{t}_c and exponentially decreasing the errors to zero. Several approaches in visual servoing use this generic control law for different applications and scenarios.

2.2. Multivariate t -distribution

In probability theory, student t -distribution is a continuous distribution function describing the probability that a given value will occur. Suppose we have a simple random sample of size n drawn from a normal population with mean μ and standard deviation σ . Then the quantity

$$t = \frac{x - \mu}{\sigma^2 / \sqrt{n}} \quad (5)$$

where $x \in (-\infty, \infty)$ has a t -distribution function with $n - 1$ degrees of freedom for univariate case. The t -distribution density curves defined by (5) are symmetric and bell-shaped in nature like the normal distribution (Fig. 2(a)). However, the advantage lies in the spread of the distribution and is more for t -density function than that of standard normal distribution. In multivariate case the distribution consists of more than one random variable. For two random variables this is known as a bivariate distribution, but the concept generalizes to any number of random variables, giving a multivariate distribution. Since the input for visual servoing is a two-dimensional image feature, the representing distribution will also be multivariate in nature. With the concept of multivariate normal distribution a multivariate student t -distribution can be easily modeled as follows.

Consider a d -dimensional random variable \mathbf{x} following a multivariate normal distribution with mean $\boldsymbol{\mu}$ and positive definite, symmetric and real $d \times d$

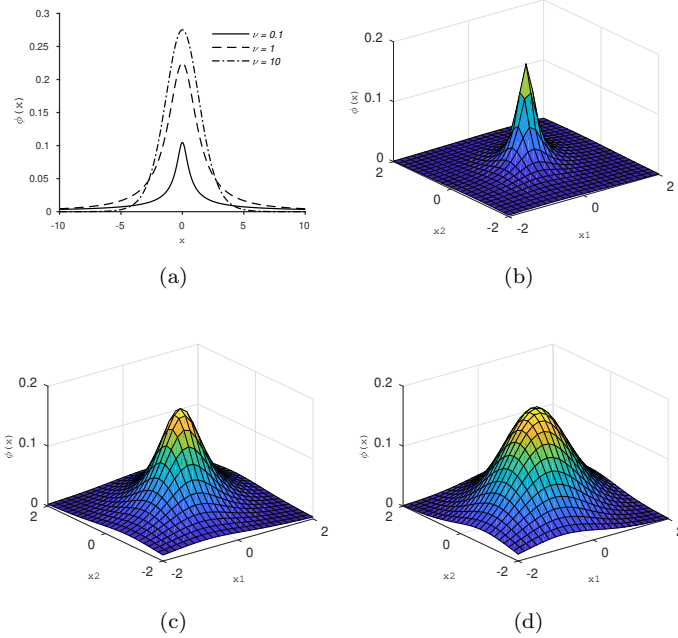


Figure 2: Students t -distribution for various degrees-of-freedom ν . (a) Univariate case, (b-d) Multivariate cases with $\nu = 0.1, 1, 10$ respectively.

covariance matrix Σ . The corresponding density function is represented by

$$N(\mathbf{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}} \Sigma^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}. \quad (6)$$

To produce a longer tail to normal distribution, two normal distributions are added, which is represented by

$$(1 - \epsilon)N(\mathbf{x}; \boldsymbol{\mu}, \Sigma) + \epsilon N(\mathbf{x}; \boldsymbol{\mu}, c\Sigma), \quad (7)$$

where c is large and ϵ is small. Equation (7) can be represented in integral form as

$$\int N(\mathbf{x}; \boldsymbol{\mu}, \Sigma/u) dH(u), \quad (8)$$

where H is a distribution function that places $(1 - \epsilon)$ at point $u = 1$ and ϵ at the point $u = 1/c$. The probability function H can be represented by a chi-squared

function having ν degree-of-freedom with random variable K such that

$$K \sim \text{gamma}(\nu/2, \nu/2). \quad (9)$$

The $\text{gamma}(a, b)$ has density function

$$f(u, a, b) = \frac{b^a u^{a-1}}{\Gamma(a)} \exp(-bu) \mathbf{I}_{0,\infty}(u); \quad (a, b > 0) \quad (10)$$

where $\mathbf{I}_{0,\infty}(u) = 1$ for $u > 0$ and zero elsewhere. From the above equations it can be summarized as, for the same random variable \mathbf{x} a multivariate t -distribution with ν degrees-of-freedom can be written as

$$\phi(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) \sim N(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}/u) \quad (11)$$

where

$$u \sim \text{gamma}(\nu/2, \nu/2). \quad (12)$$

Then a multivariate t -distribution density function [29] at a point \mathbf{x} can be represented by

$$\phi(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) = \frac{\Gamma(\frac{\nu+d}{2}) |\boldsymbol{\Sigma}|^{-\frac{1}{2}}}{(\pi\nu)^{\frac{d}{2}} \Gamma(\frac{\nu}{2}) [1 + \nu^{-1} \delta(\mathbf{x}, \boldsymbol{\mu}; \boldsymbol{\Sigma})]^{\frac{\nu+d}{2}}} \quad (13)$$

where the parameter Γ represents the Gamma function. Here the parameter ν provides a heavy-tailed alternative to the normal distribution family (Fig. 2) and as ν tends to infinity, the t -distribution approaches the normal distribution. Hence, this parameter ν may be viewed as a robustness tuning parameter. It can be fixed in advance or it can be inferred from the data for each component [30] based on different applications. The Mahalanobis distance $\delta(\mathbf{x}, \boldsymbol{\mu}; \boldsymbol{\Sigma}) = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma} (\mathbf{x} - \boldsymbol{\mu})$ denotes squared distance between \mathbf{x} and $\boldsymbol{\mu}$. In the proposed novel methodology for visual servoing, student t -distribution functions are used to model distributions of an image instead of directly taking image intensities. Finding an analytical representation of tuning parameter $\boldsymbol{\mu}$ from image attributes, which controls the spread of the distribution is a main challenge and is one of the major contribution of proposed approach. An overview of the proposed approach of servoing is given in the following section.

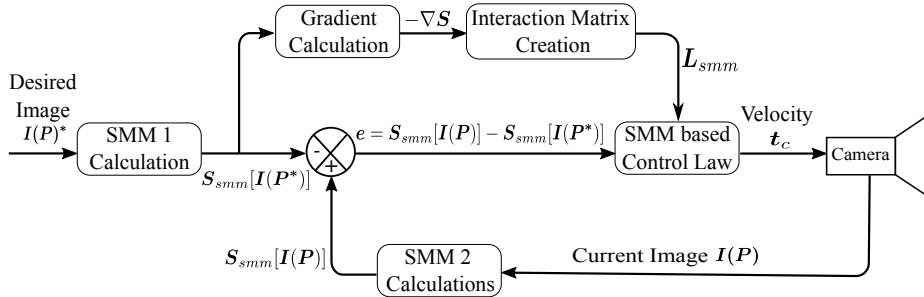


Figure 3: Block diagram for SMM based visual servoing in image space: The visual servoing control law which is derived under the framework of student t -distribution mixture models minimizes the error $S_{smm}[\mathbf{I}(\mathbf{p})] - S_{smm}[\mathbf{I}(\mathbf{p})^*]$ by providing appropriate velocity to the camera.

3. Outline of proposed Methodology

From the literatures on visual servoing it can be observed that there is a paradigm shift from using point features to entire image to generate control signals. Using entire photometric image information as visual feature gives advantage of avoiding feature detection, matching or tracking steps. On the other hand, the use of image intensity values to generate control signals are sensitive to illumination variations and suffers from smaller convergence domains. The idea of using the probability distributions, which consider the effect of association between random variables, can be used to overcome these disadvantages. The successful implementation of similar concepts can be seen in other applications like image registration [28]. This becomes motivation for the proposed SMM based visual servoing. The proposed idea is to replace each image intensity by a student t -distribution. The heavily tailed distribution function of t -distribution helps to consider the effect of neighbourhood values. By modelling each image pixels by t -distribution the image under consideration can be represented by a mixture of distributions. This mixture model is used to design the proposed control law.

The proposed framework has been developed to control the motion of camera to reach a desired pose as shown in Fig. 1 by considering whole image data

$(N \times M)$ as input. Fig. 3 provides an overview of the proposed methodology. The algorithm is initiated by feeding the desired image $\mathbf{I}(\mathbf{p})^* \in R^k$ and the initial image $\mathbf{I}(\mathbf{p}) \in R^k$ to SMM calculation modules 1 and 2, where $k = N \times M$. Here the current image is used as a feedback signal to the control law. The probabilistic models of desired image $\mathbf{S}_{smm}[\mathbf{I}(\mathbf{p})^*] \in R^k$ and current image $\mathbf{S}_{smm}[\mathbf{I}(\mathbf{p})] \in R^k$ are calculated using multivariate t -distributions and produced as an output from SMM calculation modules. It may be noted that here every pixel in the image is treated as feature point. In the proposed method the visual servoing operation is posed as a minimization problem in which the error to be minimized is $\mathbf{S}_{smm}[\mathbf{I}(\mathbf{p})] - \mathbf{S}_{smm}[\mathbf{I}(\mathbf{p})^*]$. Once the error is modeled, next the interaction matrix has to be calculated using image feature points. Besides conventional methods, in SMM based approach the interaction matrix is created using SMM of image. For this, the obtained SMM of desired image is passed through a gradient calculation module. The gradient of SMM image can be obtained by using a gradient filter over $\mathbf{S}_{smm}[\mathbf{I}(\mathbf{p})^*]$ or by taking gradient of SMM analytically. Since t -distribution is differentiable, analytical expressions can also be obtained easily. The obtained gradient is then used to calculate the interaction matrix $\mathbf{L}_{smm} \in R^{k \times 6}$.

Since entire image is being used, interaction matrix calculation for each iterations may take more time. Therefore in the proposed method the interaction matrix is calculated only once using desired image and used for the entire iterations. It has been shown in [31] that the closed loop system will remain stable even if the interaction matrix computed from the desired image is used. Next the SMM based control law module generate velocities to move the camera and generate new current image. The control law minimizes the error using an optimization framework. Eventually, the camera velocity become zero once the camera reaches the desired position which constitutes end of visual servoing operation. The SMM of an image and derivation of interaction matrix are detailed in the following sections.

4. Image modelling and Interaction matrix

Instead of directly considering pixels of an image as features the proposed method represent it as a distribution function. This takes into account the influence of neighbouring pixels. Hence, it is more suitable for the applications in real world environments in which features can not be extracted and tracked easily. The first part of this section explains statistical modelling of image using multivariate t -distribution. The next part presents an analytical formulation of interaction matrix for SMM based visual servoing.

4.1. Image representation using SMM

A student mixture model is a combination of t -distribution functions at several locations. The t -distribution density function (multivariate case) at a point $\mathbf{X} = (x, y)$ can be represented by (13) having parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu$. Using this formulation, an image can be modelled as a combination of multiple t -distribution functions given by

$$\mathbf{S}_{smm}(\mathbf{I}) = \sum_{i=1}^n \pi_i \phi(\mathbf{X}; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \nu_i), \quad (14)$$

where the mixing proportions π_i are non-negative and sum upto unity. The quantity n is the total number of pixel locations in the image. Here it should be noted that each pixel in the image contributes one component t -distribution to the SMM model. In general image processing application scenarios the mixture models are usually estimated from the image features in interest. Unlike in the general probabilistic mixture modelling paradigm, in the proposed method the mean of each multivariate t -distribution $\boldsymbol{\mu}_i \in R^{1 \times 2}$ is equal to its observed pixel position $\mathbf{X}_i = (x, y)$ and the covariance matrix $\boldsymbol{\Sigma}_i \in R^{2 \times 2}$ is proportional to the pixel intensity I_i at \mathbf{X}_i .

Finding the tuning parameter ν , which controls the spread of the distribution is main challenge and is one of the major part of our contribution. In order to obtain the value of each ν_i for each t -distribution, one of the basic property of t -distributions is considered. This property gives a relation between the variance

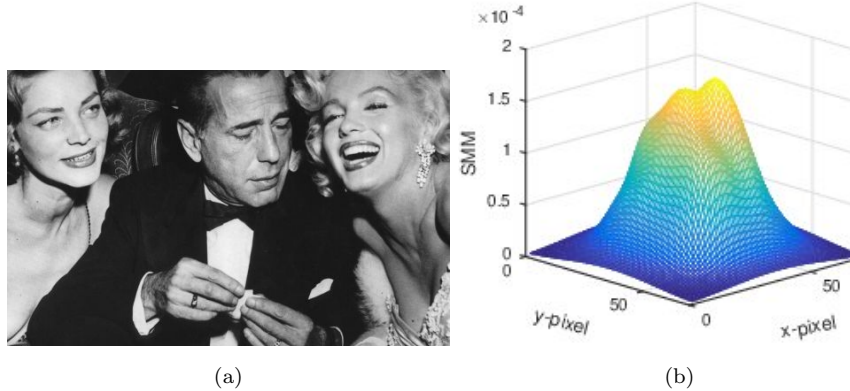


Figure 4: Image modeling by SMM: (a) Input image (b) SMM of the image. Each pixel location in the image contributes mean for t -distribution and intensity at that position for covariance matrix. The aggregation of all such t -distributions constitute student mixture model for the image

and degree-of-freedom as given below:

$$\sigma(t_{dist}) = \frac{\nu}{\nu - 2} \quad (15)$$

Since the image pixel intensity is used as variance ($\sigma = I_i$) for modeling SMM of an image, using (15) the values of ν_i can be obtained from the following relation

$$\nu_i = \frac{2\sigma_i}{(\sigma_i - 1)}. \quad (16)$$

This extra parameter ν , for modeling distribution (SMM) yields more smoothness to the density function with a probability density function having heavier tail. Using student t -distribution for modeling image is first of its kind to be used in visual servoing context. Fig. 4 shows an example of an image modeled by SMM using (13) and (14). Each pixel locations of the given image Fig. 4(a) serves as mean for its SMM (Fig. 4(b)) and intensity values as variance. The degree-of-freedom ν for each student t -distribution in SMM is calculated using the above formulation which relates degree-of-freedom of distribution to

image pixel intensity given in (16). From the SMM function modeled for the given image it can be observed that the surface is smooth and continuous, and inherently considers feature position uncertainties.

Once the SMM is modeled for current and desired images, an error between these two distributions are generated and applied to visual servo controller. The following section explains the detailed development of interaction matrix for SMM based visual servoing.

4.2. Development of interaction matrix

Once the feature models are obtained, next task is to compute the interaction matrix. The interaction matrix (\mathbf{L}) maps feature rate ($\dot{\mathbf{s}}$) into camera velocity (\mathbf{t}_c), i.e.,

$$\dot{\mathbf{s}} = \mathbf{L}\mathbf{t}_c. \quad (17)$$

Since all the pixels in the image are considered while development of SMM, the interaction matrix is obtained by arranging respective interaction matrices at locations \mathbf{x} as given below.

$$\mathbf{L}_{smm} = \begin{bmatrix} \vdots \\ \mathbf{L}_{smm}(\mathbf{x}) \\ \vdots \end{bmatrix}. \quad (18)$$

It can be deduce from (14) that $\mathbf{L}_{smm}(\mathbf{x})$ is the sum of every interaction matrices related to each student t -distribution expressed from all i locations to location \mathbf{x} as given below

$$\mathbf{L}_{smm}(\mathbf{x}) = \sum_{i=1}^n \mathbf{L}_{smm}(\mathbf{x}, i). \quad (19)$$

In order to obtain individual $\mathbf{L}_{smm}(\mathbf{x}, i)$, it is required to find the mapping between the time derivative of SMM ($\dot{\mathbf{S}}$) and 2D image pixel velocity ($\dot{\mathbf{x}}$). By using similar concept from temporal luminance constancy equation for optical flow [32], we can write

$$\mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t) = \mathbf{S}(\mathbf{x}, t). \quad (20)$$

Here assumption is made that SMM image also follow temporal luminance constancy hypothesis. First order Taylor series expansion on (20) provides

$$\begin{aligned} \mathbf{S}(\mathbf{x}, t) &= \mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t) \\ &+ \frac{\partial \mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t)}{\partial \mathbf{x}} \frac{d\mathbf{x}}{dt}, \\ &+ \frac{\partial \mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t)}{\partial t} \frac{dt}{dt} \end{aligned} \quad (21)$$

and we get

$$\frac{\partial \mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t)}{\partial \mathbf{x}} \frac{d\mathbf{x}}{dt} + \frac{\partial \mathbf{S}(\mathbf{x} + \delta\mathbf{x}, t + \delta t)}{\partial t} = 0. \quad (22)$$

In simple terms the above formula can be written into the following form

$$\nabla \mathbf{S}^T \dot{\mathbf{x}} + \dot{\mathbf{S}} = 0 \quad (23)$$

Upon rearranging (23) an equation similar to optical flow constraint equation is obtained which is given by

$$\dot{\mathbf{S}} = -\nabla \mathbf{S}^T \dot{\mathbf{x}}. \quad (24)$$

It is well known that the image pixel velocity $\dot{\mathbf{x}}$ is related to the camera velocity \mathbf{t}_c [4] in the following manner

$$\dot{\mathbf{x}} = \mathbf{L}_x \mathbf{t}_c, \quad (25)$$

where $\mathbf{L}_x \in R^{2 \times 6}$ is known as basic interaction matrix related to pixel locations x . Using (25) in (24) gives the relation between $\dot{\mathbf{S}}$ and \mathbf{t}_c , from which the interaction matrix

$$\mathbf{L}_{smm}(\mathbf{x}, i) = -\nabla \mathbf{S}^T \mathbf{L}_x \quad (26)$$

where

$$\nabla \mathbf{S} = \begin{bmatrix} \nabla S_u \\ \nabla S_v \end{bmatrix}. \quad (27)$$

is obtained. To compute the gradients $\nabla \mathbf{S} \in R^{1 \times 2}$, methods similar to horizontal and vertical image gradient calculations or analytical formulations can be used.

5. SMM based visual servoing

The aim of a vision-based control schemes is to minimize an error which is given in (1). In the proposed method the error function is modeled as difference between current and desired SMMs and, visual servoing is considered as an optimization problem. The error between current and desired SMMs act as a cost function to be minimized for optimization. This section presents the underlying optimization frame work for SMM based visual servoing.

For a robot, the end effector pose \mathbf{P} can be represented by rigid transformation $[\mathbf{R}, \mathbf{T}]$ with respect to world coordinate system. Here, \mathbf{R} is a 3×3 rotation matrix and $\mathbf{T} = [t_x \ t_y \ t_z]^T$ is the 3×1 translation vector. The rotation matrix \mathbf{R} can be parametrized in terms of 3 Euler angles (α, β and γ) such that $\mathbf{R} = \mathbf{R}_x(\alpha)\mathbf{R}_y(\beta)\mathbf{R}_z(\gamma)$. In visual servoing it is required to move the robot end-effector from an initial pose $\mathbf{P} \in \mathbf{R}^3 \times SO(3)$ to reach a desired pose \mathbf{P}^* . To achieve this goal, a cost function $f()$ has to be defined. Most of the time this cost function is an error which needs to be minimized. Thus, a visual-servoing problem can be written as an optimization problem given by

$$\hat{\mathbf{P}} = \underset{\mathbf{P}}{\operatorname{argmin}} f(\mathbf{P}, \mathbf{P}^*) \quad (28)$$

where $\hat{\mathbf{P}}$ is the pose closest possible to \mathbf{P}^* , reached after the optimization. In classical visual-servoing, the cost function is defined as the distance between geometrical features extracted from the image, and the corresponding minimization is

$$\hat{\mathbf{P}} = \underset{\mathbf{P}}{\operatorname{argmin}} \|(\mathbf{s}(\mathbf{P}) - \mathbf{s}(\mathbf{P}^*))\|. \quad (29)$$

For direct approaches like Kernel and Photometric Visual Servoing $f()$ is the sum of squared differences of the image intensities as shown below.

$$\hat{\mathbf{P}} = \underset{\mathbf{P}}{\operatorname{argmin}} \|(\mathbf{I}(\mathbf{P}) - \mathbf{I}(\mathbf{P}^*))\| \quad (30)$$

It must be noted that the proposed method also uses the entire photometric information (14) provided by the images as in the other dense visual servoing

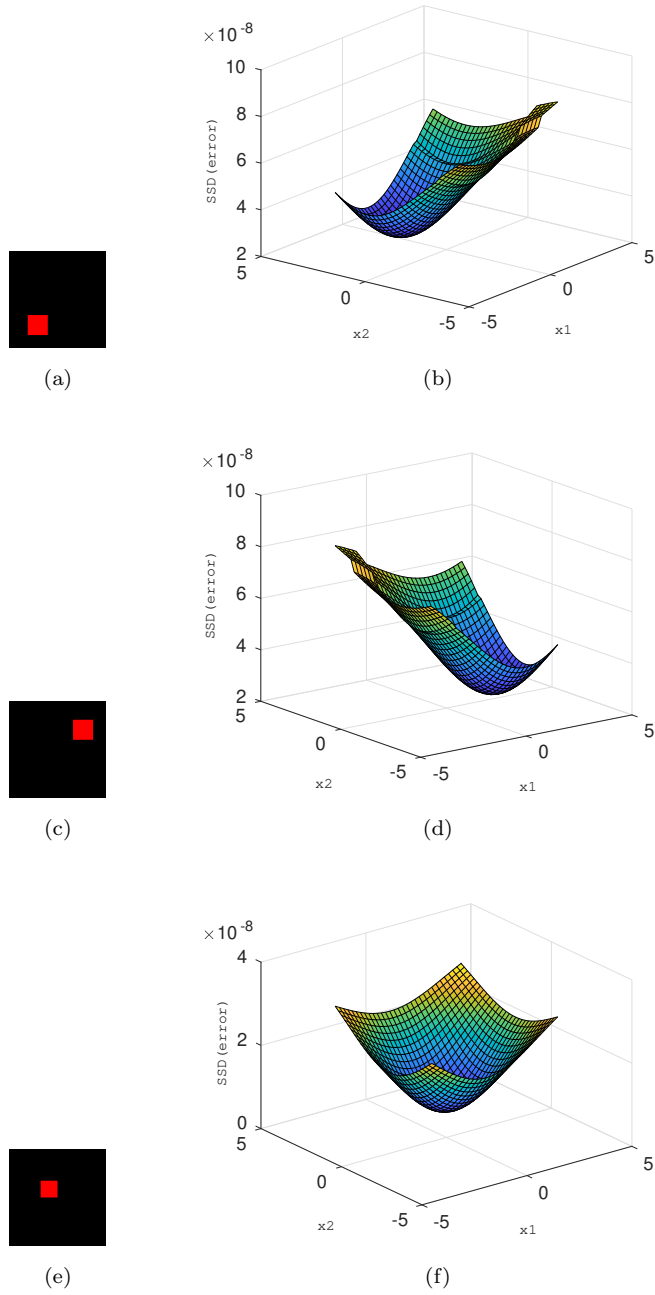


Figure 5: Cost functions for different test images having different point locations using SMM approach. It can be observed that the SMM based cost functions are smooth and differentiable. This makes the optimization framework to work easily and guarantees convergence to servoing operation.

approaches. Therefore in the proposed framework the objective is to regulate the following error function to zero:

$$\mathbf{e}(\mathbf{P}) = \mathbf{S}_{smm}[\mathbf{I}(\mathbf{P})] - \mathbf{S}_{smm}[\mathbf{I}(\mathbf{P}^*)], \quad (31)$$

where term $\mathbf{S}_{smm}[\]$ is calculated using (14) for current and desired images. In other words, the problem is to minimize an error vector $\mathbf{e}(\mathbf{P})$ of visual features $\mathbf{S}_{smm}(\mathbf{P})$ extracted from the initial/current image by finding a vector \mathbf{t}_c that incrementally minimizes a cost function $E(\mathbf{S}_{smm}(\mathbf{P}))$. Viewing the problem as a non-linear least squares minimization, the cost function is reformulated as

$$E(\tilde{\mathbf{S}}(\mathbf{P})) = \frac{1}{2}[\tilde{\mathbf{S}}(\mathbf{P}) - \tilde{\mathbf{S}}(\mathbf{P}^*)]^T[\tilde{\mathbf{S}}(\mathbf{P}) - \tilde{\mathbf{S}}(\mathbf{P}^*)] \quad (32)$$

where $\tilde{\mathbf{S}}$ represents $\mathbf{S}_{smm}(\mathbf{I}(\cdot))$.

Fig. 5 depicts various cost functions for 2D cases with different point images. It was observed that the SMM based cost functions are smooth and also makes the surface differentiable without using any extension parameters as required for GMM [27]. This makes the modeling easier and guarantees convergence to servoing operation.

Since first order optimization methods are known to be slow, methods that are based on the second order Taylor series approximation of the cost function like Gauss-Newton method are considered here. With the proposed novel error function (31) in-terms of SMM and interaction matrix in (18) the required camera velocity is obtained as

$$\mathbf{t}_c = -\lambda \mathbf{L}_{smm}^+ \mathbf{S}_{smm}[\mathbf{I}(\mathbf{P})] - \mathbf{S}_{smm}[\mathbf{I}(\mathbf{P}^*)] \quad (33)$$

where λ is a positive scalar and \mathbf{L}_{smm}^+ is the pseudo inverse of SMM based interaction matrix given in (18).

It will be shown in the following section that the above controller results in (a) exponential decrease of error function and (b) convergence of camera and joint velocities are regulated to zero at the desired pose.

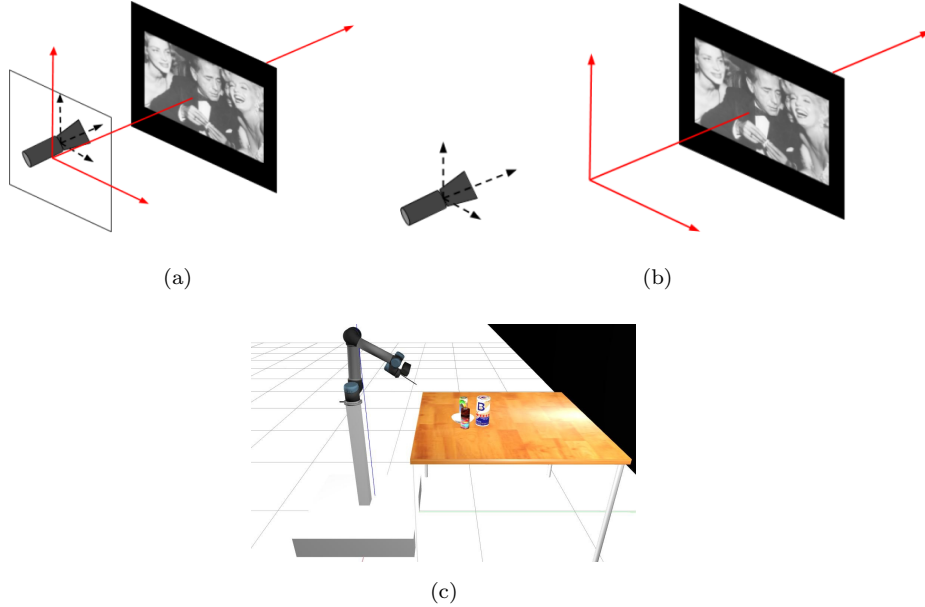


Figure 6: SMM based visual servoing simulation environment:(a) 2D positioning task with planar camera motion (b) 3D positioning task using 6-DOF camera motion(c) Realistic environment in Gazebo with UR5 manipulator and 3D objects.

6. Results and Discussion

This section discusses simulation results of visual servoing using the proposed framework. An eye in hand system is considered and the camera reaching desired pose is considered as convergence of an experiment. Number of simulated experiments are performed to decipher performance of the proposed methodology. Initial part of this section shows the behaviour of the proposed method by simple 2D (Fig. 6a) and 3D (Fig. 6b) camera movements which is servoing towards a planar image object. For verifying the effectiveness of the proposed algorithm in real scenario, visual servoing experiments are carried out in Gazebo environment with using UR5 manipulator and 3D objects as shown in Fig. 6(c). Finally to show the robustness of the framework, its performance under different image sizes, image contents and occlusions are investigated.

Table 1: Test cases for 2D positioning task with fixed depth and desired position at $(t_x, t_y, \theta_z) = (0, 0, 0)$. Camera positioning error are shown for the start and end of SMM visual servoing

Exp.	Initial error			Final error		
	$t_x(m)$	$t_y(m)$	θ_z	$t_x(m)$	$t_y(m)$	θ_z
1	0.3	0.3	0°	0.0021	0.0083	0°
2	0.25	0.25	-15°	-0.0002	0.0050	-0.0892°
3	0.25	0.25	10°	0.0007	0.0058	-0.01°
4	-0.3	0.25	-18°	-0.0012	0.0077	-0.0081°
5	0.4	0.35	5°	0.0015	0.0076	0.054°

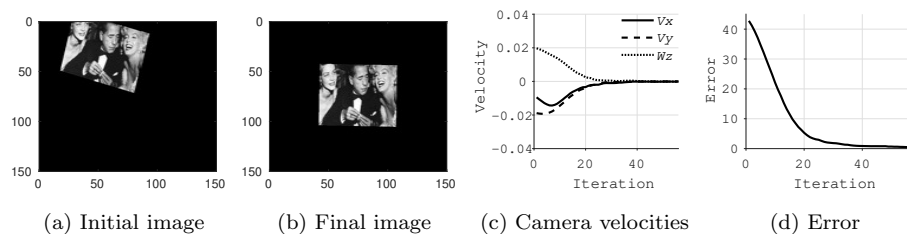


Figure 7: One of the test case for 2D SMM visual servoing.

6.1. 2D and 3D positioning task

First a 2D case was considered where the objective is to control linear and rotational motion (v_x, v_y, ω_z) along a plane (see Fig. 6(a)) exhibited by a 3-DOF camera to reach the desired pose. In this experiment a planar image is considered in the workspace and desired camera position is taken as $(t_x, t_y, \theta_z) = (0, 0, 0)$. To evaluate the validity of the proposed framework different initial poses have been chosen as shown in Table 1. The end of the simulation is considered when the norm of error, pertaining to (32), approaches to zero. From the error in final pose shown in Table 1, it can be concluded that the motion exhibited by proposed framework made the camera to successfully reach the desired pose with negligible errors. The initial image, final image, camera velocities and error plots are shown in Fig. 7 for Experiment 3 in Table 1 as an illustration. It can be seen that the velocities decay to zero and the norm of error shows the convergence of cost function to minimum.

Next, a spatial case with a 6-DOF camera and a planar object placed at a fixed distance as depicted in Fig. 6(b) is considered. For experimental purposes,

Table 2: Test cases for 3D positioning task with 6-DOF camera for reaching desired pose $(p_x, p_y, p_z, \theta_x, \theta_y, \theta_z) = (0, 0, 0, 0, 0, 0)$. Camera positioning error are shown for the start and end of SMM visual servoing

Exp.	Initial error						Final error					
	$p_x(m)$	$p_y(m)$	$p_z(m)$	θ_x	θ_y	θ_z	$p_x(m)$	$p_y(m)$	$p_z(m)$	θ_x	θ_y	θ_z
1	-0.4	-0.4	-0.50	0.5°	1°	-10°	-0.0071	-0.1220	-0.0500	0.0156°	0.0023°	-1.8307°
2	-0.4	-0.4	-0.53	1°	0.3°	-20°	-0.1193	-0.0035	-0.0013	0.0089°	0.0171°	-1.1296°
3	0.36	0.38	-0.48	0°	0.3°	-12°	0.0286	0.0249	-0.0028	0.0269°	0.0311°	-0.6179°
4	0.35	0.37	-0.55	0°	0°	8°	0.0210	0.0322	-0.0012	-0.0080°	0.0046°	0.1668°
5	0.22	-0.32	-0.51	0.1°	1.2°	20°	0.0274	0.0231	-0.0164	0.0885°	0.7329°	0.4003°
6	0.37	0.42	-0.46	0.8°	0.5°	8°	0.0204	0.0315	-0.0050	0.1583°	0.0519°	0.9347°
7	0.30	-0.45	-0.48	1°	0.5°	-20°	0.0274	-0.0125	-0.0081	0.4828°	0.1682°	-0.8595°
8	-0.37	0.28	-0.52	-0.6°	-0.1°	-20°	-0.0123	0.0352	-0.0021	-0.1498°	-0.0718°	0.0590°
9	0.35	0.38	-0.53	0.4°	0.3°	-20°	0.0181	0.0136	-0.0210	-0.0602°	0.0348°	-1.3683°
10	0.27	-0.35	-0.49	0.4°	1°	14°	0.0112	0.0833	-0.0089	0.0951°	0.0743°	0.9961°

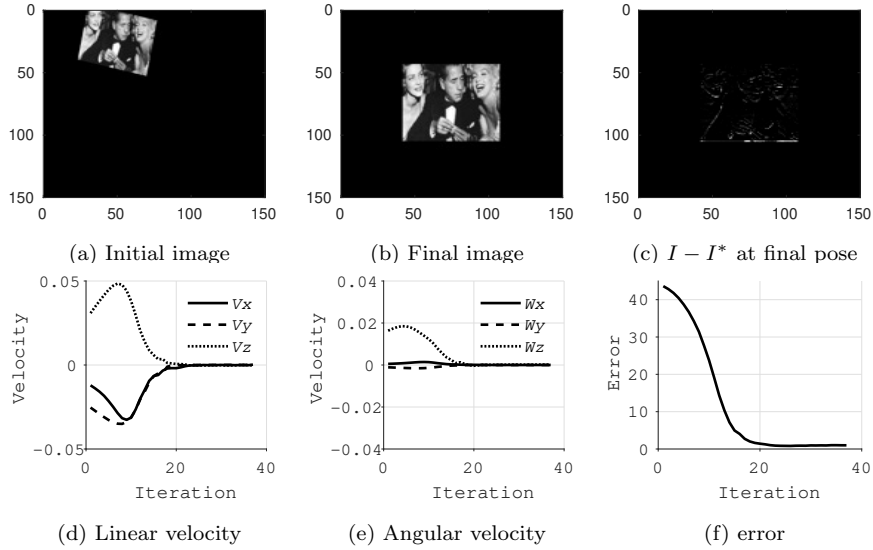


Figure 8: Test case for 3D SMM visual servoing.

initial depth is considered as $-0.5m$ and initial camera plane is assumed parallel to the desired one. Figure 8 shows the results for SMM based visual servoing for one of the 3D test case from Table 2 (i.e., Experiment 3). It can be observed that the camera converges to the desired pose with final errors close to zero. From the final error in camera pose, in Table 2, and the pixel level error between final and desired images ($I - I^*$), in Fig. 8(c), it is evident that desired pose is

reached. This validates the convergence of proposed framework in 3D case. Plots of linear and angular velocity, and norm of error in Fig.8(d-f) depict that the proposed algorithm converges successfully to desired values. In order to validate robustness of proposed approach, servoing using different camera configurations were performed as presented in Table 2. It can be seen from the error in position that the algorithm is able to perform successful visual servoing for all the cases. This validates the robustness to different camera poses.

6.2. SMM visual servoing using a robot manipulator

In this subsection, effectiveness of the proposed framework is demonstrated for 3D objects. For this purpose, a 3D spatial environment was created in ROS Gazebo as shown in Fig.6(c). The environment consists of an UR5 manipulator (6-DOF), a depth camera (Microsoft Kinect) mounted at the end-effector and objects such as a cereal bowls and coke cans placed randomly on a table. A detailed description and results of the experiments are discussed in following sections.

6.2.1. Visual servoing towards 3D objects in gazebo

The UR5 manipulator mounted on a stand is positioned in front of a wooden table as seen in Fig. 6(c). In this setup, experiments were performed to test the algorithm for reaching a fixed desired position, row 3 in Fig. 9, from different initial positions, row 1 in Fig. 9. The actual images obtained are shown in the row 2 in Fig. 9 which match with the desired images in row 3. From the plots of joint velocity in row 4 and norm error in row 5, it can be concluded that the algorithm converges satisfactorily with exponential decay in norm of error and joint velocities tending to zero. The camera positions in row 6 shows the path traced by the end-effector reaching the desired position. A number of experiments were performed as depicted in Table 3 which shows that final error in pose is reduced close to zero for all cases similar to Fig. 9. This validates the convergence and robustness of the proposed framework in a realistic environment using a robotic manipulator.

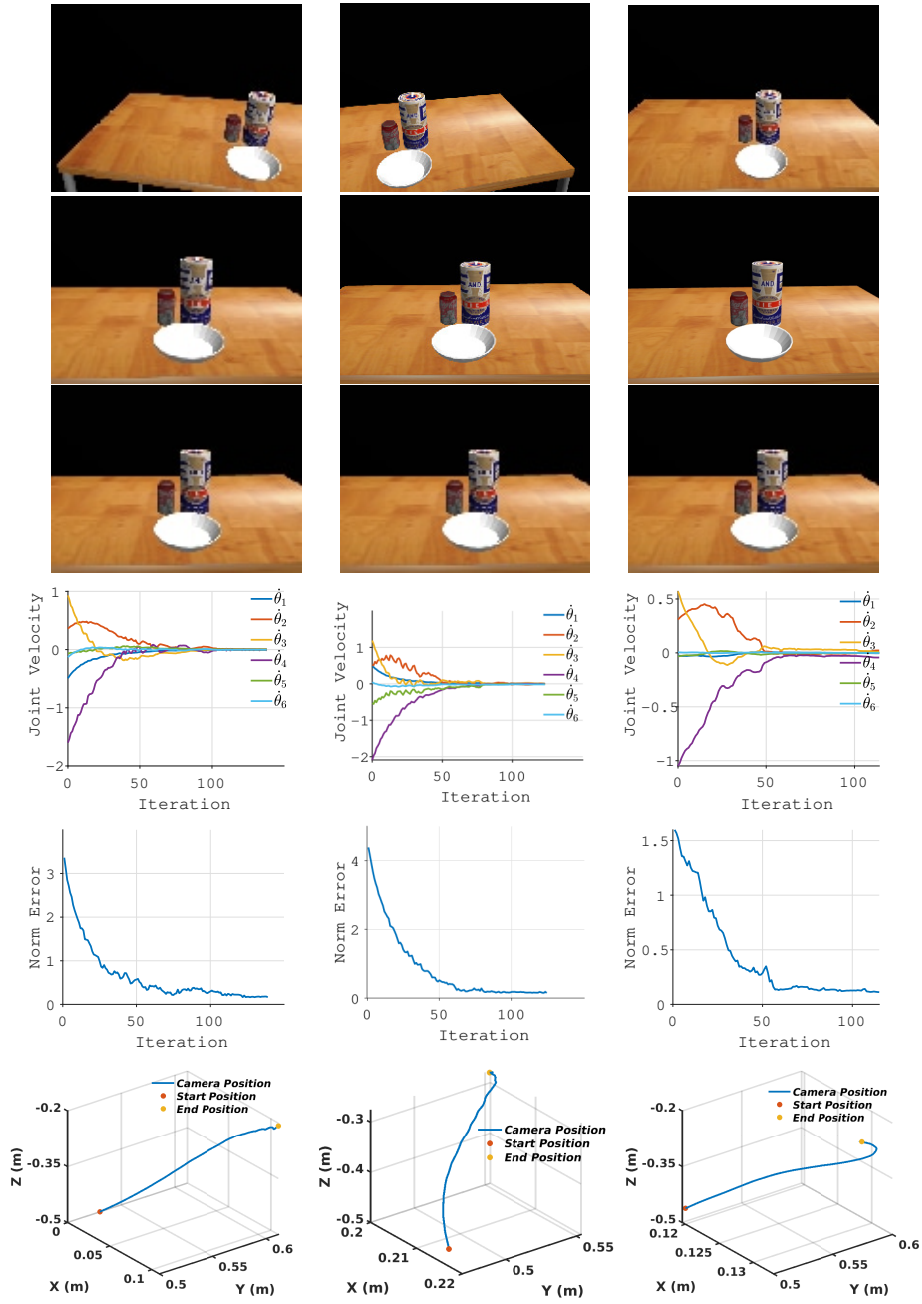


Figure 9: Results of SMM visual servoing for three different start positions in gazebo using UR5 manipulator. Rows 1-6 shows initial image, final obtained image, desired image, joint velocities (rad/s), norm error (pixels) and camera trajectory plots respectively.

Table 3: Positioning error : 3D positioning task in gazebo

Exp.	Initial error						Final error					
	$p_x(m)$	$p_y(m)$	$p_z(m)$	θ_x	θ_y	θ_z	$p_x(m)$	$p_y(m)$	$p_z(m)$	θ_x	θ_y	θ_z
1	0.1	0.07	0.23	-5.73°	0.46°	-11.46°	0.0122	-0.0257	-0.0242	1.19°	-0.51°	0.54°
2	-0.1	0.12	0.23	-5.73°	0.46°	11.46°	-0.078	0.033	0.046	-1.66°	1.03°	-2.29°
3	0.0018	0.082	0.23	-5.73°	0.46°	0°	-0.0105	0.0048	0.011	-0.04°	0.28°	0.49°
4	-0.052	0.36	0.21	-6.87°	0°	8.59°	-0.0037	-0.0094	0.015	-1.54°	0.47°	3.78°
5	0.17	-0.4	0.24	-22.9°	0.02°	-20.05°	-0.0014	-0.0011	0.0016	-0.12°	-0.034°	-0.057°
6	0.16	0.28	0.15	-5.73°	0°	-14.32°	0.0012	-0.0012	-0.0011	-0.085°	0.47°	0.25°

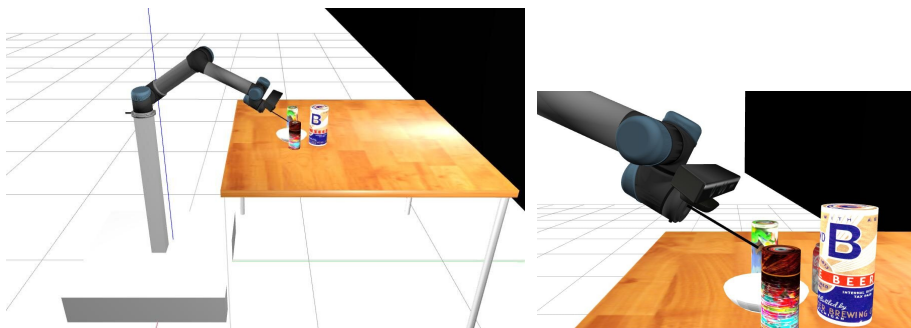


Figure 10: UR5 manipulator positioning at the end of SMM Visual Servoing

6.2.2. Close object pursuit using SMM

In the previous section visual servoing was performed towards a group of 3D objects requiring small camera motion. In this subsection, visual servoing is performed towards a particular object among a cluster of objects, i.e., close object pursuit, requiring large camera motion. The pose of the manipulator reached at the end of visual servoing is shown in Fig. 10. Here, an additional link depicting gripper is attached to the end-effector of the UR5 manipulator that touches the object once the manipulator reaches its final position at the end of servoing. A number of experiments were performed to show efficacy of the proposed framework in close object manipulation requiring large camera motion. It can be seen from row 1 in Fig. 11 that the manipulator starts from a position where the entire cluster of objects is visible. The final image corresponds to closer view of the object under consideration as seen in row 3

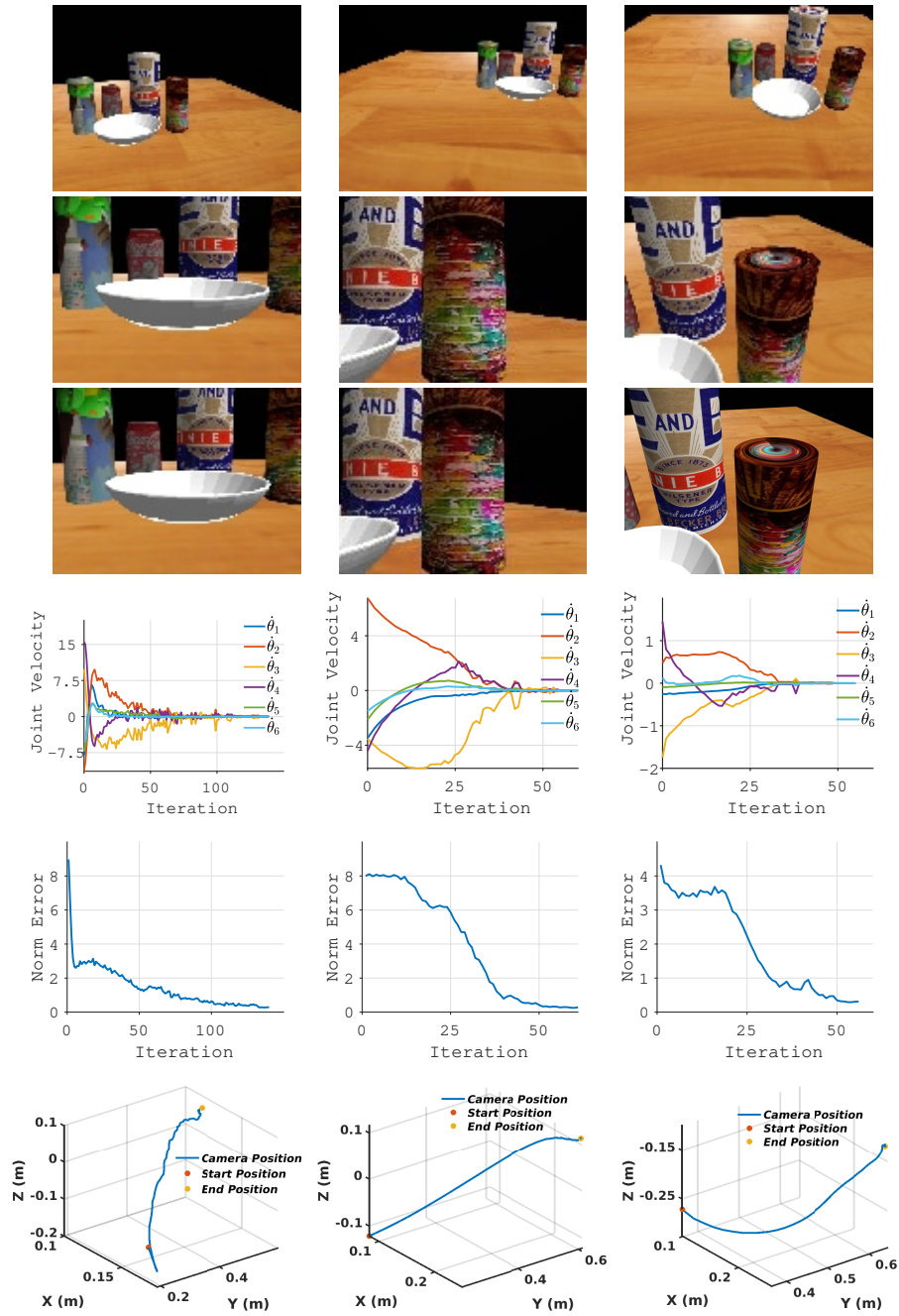


Figure 11: Results of SMM visual servoing for three different start and desired positions in gazebo using UR5 manipulator. Rows 1-6 shows initial image, desired image, final obtained image, joint velocities (rad/s), norm error (pixels) and camera trajectory plots respectively.

in Fig. 11, which clearly depicts that the manipulator has moved close enough to see the desired image in row 2. This shows the effectiveness of the proposed method for object manipulations requiring large camera motion. Plots in row 4 and row 5 in Fig.11 show that the joint velocities and norm error values are reduced exponentially from the start to end as the algorithm converges. The path traced by the end-effector during the pursuit can also be seen in row 6.

6.3. Robustness Evaluation

This section discusses the robustness of the proposed visual servoing framework through a larger set of simulated experiments. This includes experiments with different image scenes, image resolutions and occlusions. Results are verified for both 2D image in Fig. 6(b) and 3D object in Fig. 6(c) in gazebo environment.

6.3.1. Effect to image resolution

Here, two test cases are considered in order to compare the execution of proposed approach. Case A uses small image size of 50×50 pixels and Case B uses larger image size of 100×100 pixel. First 3D positioning task was considered where for both cases, the initial configurations of camera were taken same as given by experiment 4 in Table 2. The row 1 and row 2 of Fig. 12 show results for Case A and Case B, respectively. Similarly, results were obtained using UR5 manipulator for 3D object case in Fig. 13 where rows 1 and 2 show the results for Case A and Case B, respectively.

From these plots it can be observed that even though both cases converge and follow exponential decay, error in Fig.12 (Column 4) and Fig. 13 (Column 3), Case B have slightly better convergence than Case A. It can also be observed that velocity profiles in Fig.12 (Column 2-3) and Fig.13 (Column 2) are smoother when (100×100) image is used. This infers that the increase in image size results in smoothness and better convergence. But as image size increases the number of calculations for computation of SMM also increases. Hence, there is

a trade off between performance and cost of operation for a particular servoing application.

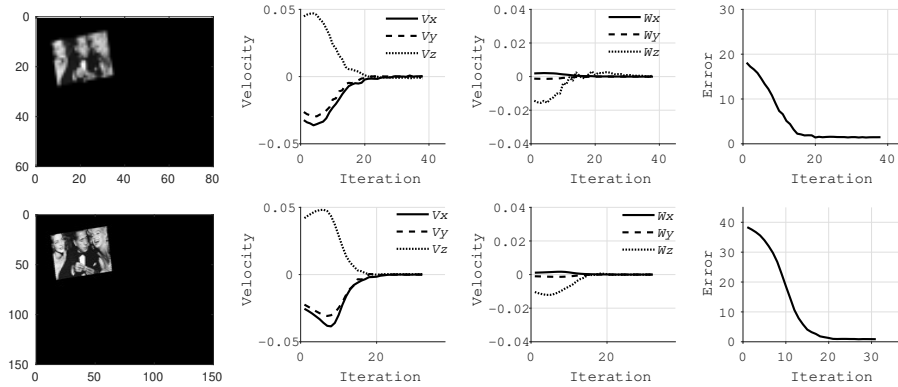


Figure 12: SMM visual servoing 6-DOF case using different image resolutions. First and second row uses (50×50) and (100×100) pixels respectively. The columns are initial image, linear velocities, angular velocity and norm error.

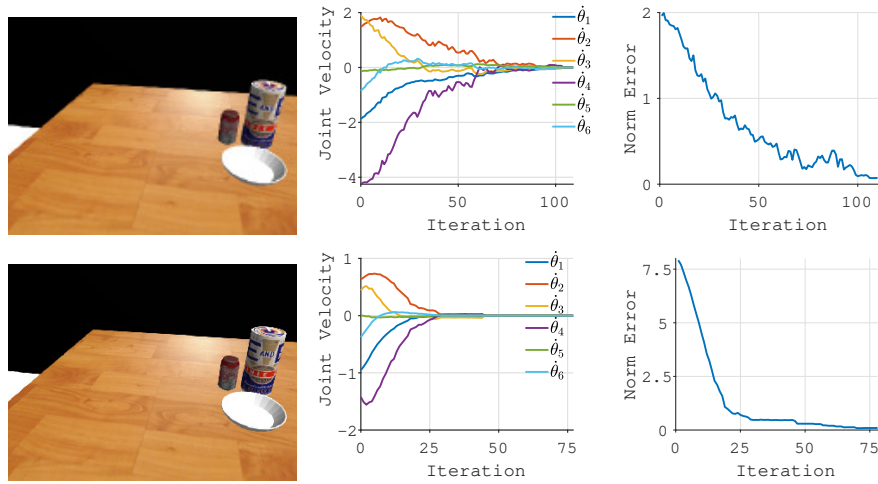


Figure 13: SMM visual servoing with UR5 manipulator using different image resolutions. First and second row uses (50×50) and (100×100) pixels respectively. The columns are initial image, joint velocities and norm error

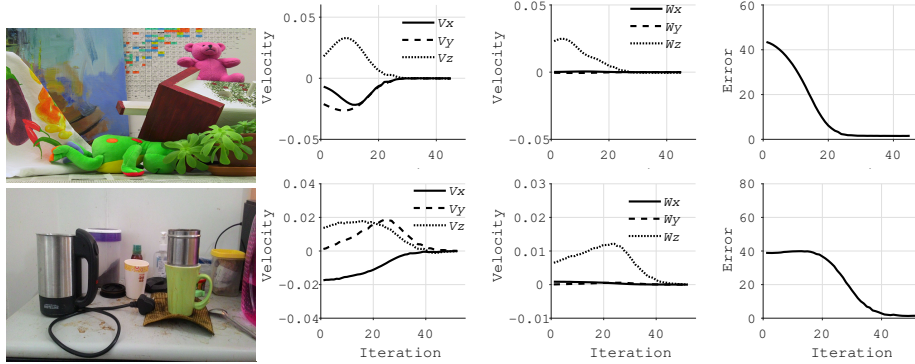


Figure 14: SMM visual servoing 6-DOF case using using different image content. Column 1-4 shows the image, camera linear and angular velocities, and error plot.

6.3.2. Robustness to image content

In this subsection, further investigations are carried out to demonstrate robustness in terms of different images and image contents. The experiments were conducted to depicts performance of the proposed framework for different textured images from different poses. In Fig. 14 two different images have been considered. The first include standard test image (Fig. 14(Row 1)) and the second image presents real objects ((Fig. 14(Row 2)) using a web cam. Another set of experiments was conducted in gazebo using a UR5 manipulator wherein different objects were used in the environment along with different poses as shown in Fig.15.

The results of the experiments in Fig. 14 and Fig.15 depict that norm of error and joint velocities for different objects converges to the desired values. It is worth noting that for the latter case, the control law converges even in the case of a low-textured scenes and luminance variations. This shows efficacy of the proposed framework when using different shape and structure of the objects in the scene.

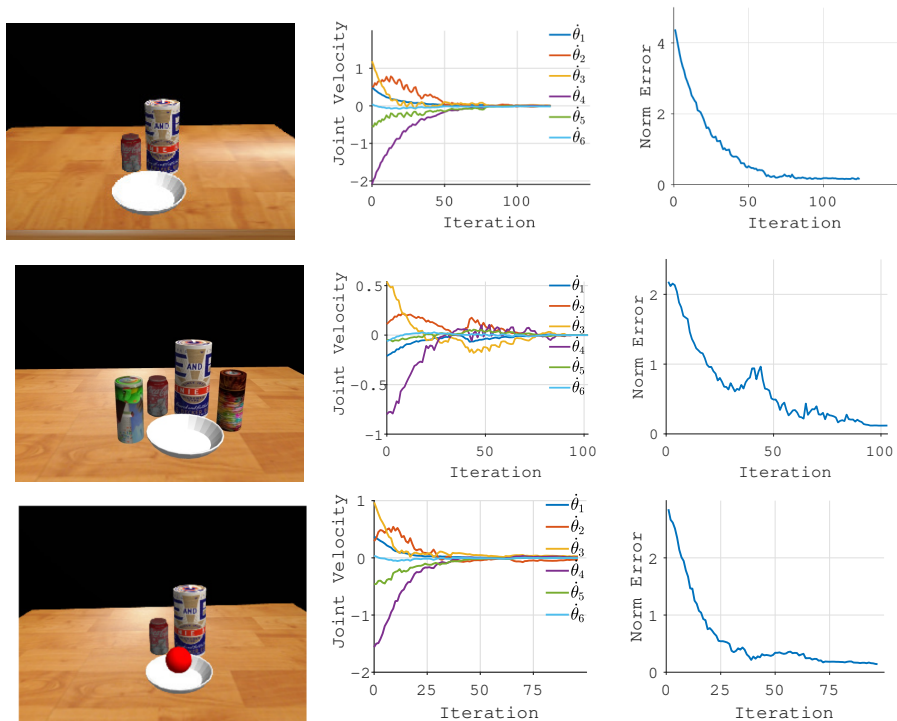


Figure 15: SMM visual servoing of UR5 manipulator using different image content. Column 1-4 shows the image, manipulator joint velocities (rad/s) and norm error (pixels) plot.

6.3.3. Robustness to partial occlusions

As the proposed method uses mixture models of photometric information, its sensitivity to partial occlusions is crucial. Hence, robustness to partial occlusion is tested in this subsection. For introducing occlusion, one of the face in image under consideration is partially covered by a synthetic image patch as shown in Fig. 16(b). Here, it must be noted that the desired image is still the one without the patch. For the experiment shown in Fig. 16 the starting pose is same as given in Experiment 5 in Table 2. Despite the occluded image considered, the control law still converges as evident from velocity and error plot in Fig. 16(d-f). A more realistic scenario is considered in Fig. 17 where a small cube was introduced in the scene while performing visual servoing using UR5. For both

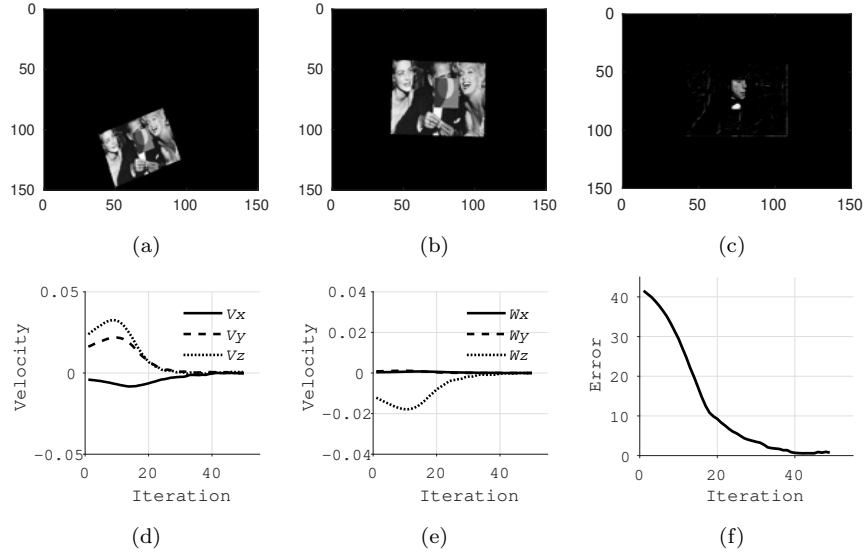


Figure 16: SMM visual servoing 6-DOF case with occlusion. (a) Initial image. (b) Final image. (c) $I - I^*$ at desired position. (d) Linear velocity. (e) Angular velocity. (f) norm error

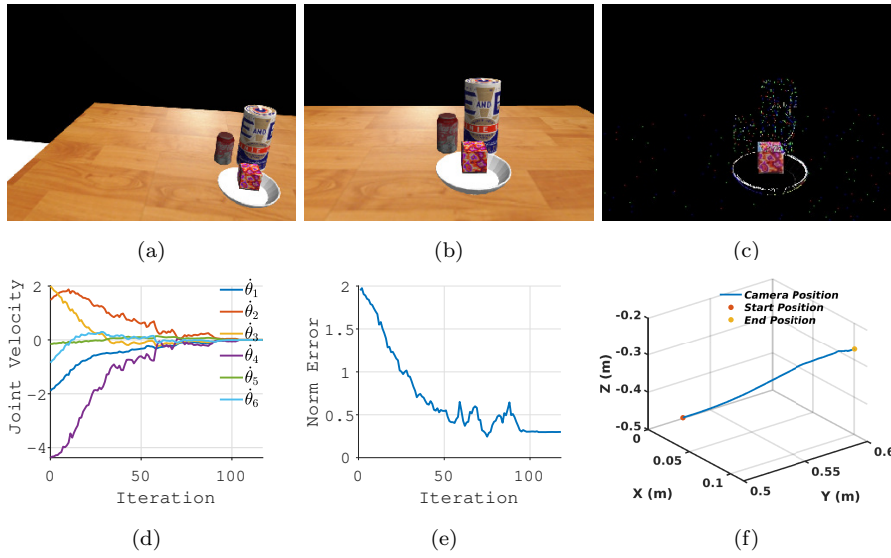


Figure 17: SMM visual servoing in 3D with occlusion. (a) Initial image. (b) Final image. (c) $I - I^*$ at desired position (d) manipulator joint velocities (rad/s). (e) Norm error (pixels). (f) Camera position

cases, there exist error in final and desired images ($I - I^*$) in terms of pixel intensity values as evident from Fig. 16(c) and Fig. 17(c). As the final image is not the desired one, the error cannot vanish completely at the end of visual servoing. It is worth noting that regardless of the non-zero value of error, the convergence is not affected by partial occlusions and final pose is achieved. This shows the robustness of the proposed approach to partial occlusions.

7. CONCLUSIONS AND FUTURE WORK

In this paper, a student t -distribution mixture model based framework is proposed for visual servoing. Instead of using image features, the entire image is modeled as a function using SMM. The SMM model enables a method to deal with non-textured/textured objects and also it serves as a replacement of methods where extracting visual features is too complex or too costly. An analytical formulation of interaction matrix using the proposed novel SMM feature model is introduced for visual servoing. The proposed method relies on the minimization of distances between probability density functions defined by SMM of current and desired images. Thus the control law is modeled as an optimization problem for minimizing the cost function.

This control law is able to perform servoing operations in successful manner even with the influence of image lighting changes, texture changes and occlusions. To prove its efficacy, the proposed algorithm was initially implemented on a numerical simulation model for 2D, 3D cases and later a realistic simulation is performed in gazebo using UR5 manipulator. The experimental results showed the effectiveness and accuracy of the proposed method for different images, different poses, different image size and under occlusions. While the proposed system works well on the numerical model, execution on the real-world setup and detailed stability analysis will be taken up as future work. The method will also be extended for visual servoing towards an object which is under motion.

References

References

- [1] B. Espiau, F. Chaumette, P. Rives, A new approach to visual servoing in robotics, *IEEE Transactions on Robotics and Automation* 8 (3) (1992) 313–326.
- [2] P. I. Corke, Visual control of robot manipulators—a review, *Visual servoing* 7 (1993) 1–31.
- [3] S. Hutchinson, G. D. Hager, P. I. Corke, A tutorial on visual servo control, *IEEE transactions on robotics and automation* 12 (5) (1996) 651–670.
- [4] F. Chaumette, S. Hutchinson, Visual servo control. i. basic approaches, *IEEE Robotics & Automation Magazine* 13 (4) (2006) 82–90.
- [5] F. Chaumette, S. Hutchinson, Visual servo control, part ii: Advanced approaches, *IEEE Robotics and Automation Magazine* 14 (1) (2007) 109–118.
- [6] A. I. Comport, D. Kragic, E. Marchand, F. Chaumette, Robust real-time visual tracking: Comparison, theoretical analysis and performance evaluation, in: *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, IEEE, 2005, pp. 2841–2846.
- [7] C. Lazar, A. Burlacu, Performance evaluation of point feature detectors for eye-in-hand visual servoing, in: *Industrial Informatics, 2007 5th IEEE International Conference on*, Vol. 1, IEEE, 2007, pp. 497–502.
- [8] J. Xin, B.-J. Ran, X.-M. Ma, Robot visual sliding mode servoing using sift features, in: *Control Conference (CCC), 2016 35th Chinese*, IEEE, 2016, pp. 4723–4729.
- [9] É. Marchand, F. Chaumette, Feature tracking for visual servoing purposes, *Robotics and Autonomous Systems* 52 (1) (2005) 53–70.

- [10] É. Marchand, F. Spindler, F. Chaumette, Visp for visual servoing: a generic software platform with a wide class of robot control skills, *IEEE Robotics & Automation Magazine* 12 (4) (2005) 40–52.
- [11] M. Pressigout, E. Marchand, Real time planar structure tracking for visual servoing: a contour and texture approach, in: *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, IEEE, 2005, pp. 251–256.
- [12] A. A. Hafez, V. Chari, C. Jawahar, Combining texture and edge planar trackers based on a local quality metric, in: *Robotics and Automation, 2007 IEEE International Conference on*, IEEE, 2007, pp. 4620–4625.
- [13] F. Chaumette, Image moments: a general and useful set of features for visual servoing, *IEEE Transactions on Robotics* 20 (4) (2004) 713–723.
- [14] S. K. Nayar, S. A. Nene, H. Murase, Subspace methods for robot vision, *IEEE Transactions on Robotics and Automation* 12 (5) (1996) 750–758.
- [15] K. Deguchi, A direct interpretation of dynamic images with camera and object motions for vision guided robot control, *International Journal of Computer Vision* 37 (1) (2000) 7–20.
- [16] V. Kallem, M. Dewan, J. P. Swensen, G. D. Hager, N. J. Cowan, Kernel-based visual servoing, in: *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, IEEE, 2007, pp. 1975–1980.
- [17] A. Dame, E. Marchand, Mutual information-based visual servoing, *IEEE Transactions on Robotics* 27 (5) (2011) 958–969.
- [18] C. Collewet, E. Marchand, Photometric visual servoing, *IEEE Transactions on Robotics* 27 (4) (2011) 828–834.
- [19] B. Delabarre, E. Marchand, Visual servoing using the sum of conditional variance, in: *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, IEEE, 2012, pp. 1689–1694.

- [20] C. Teuliere, E. Marchand, Direct 3d servoing using dense depth maps, in: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, IEEE, 2012, pp. 1741–1746.
- [21] Q. Bateux, E. Marchand, Histograms-based visual servoing, IEEE Robotics and Automation Letters 2 (1) (2017) 80–87.
- [22] M. Bakthavatchalam, O. Tahri, F. Chaumette, A direct dense visual servoing approach using photometric moments, IEEE Transactions on Robotics 34 (5) (2018) 1226–1239.
- [23] E. Marchand, Subspace-based direct visual servoing, IEEE Robotics and Automation Letters 4 (3) (2019) 2699–2706.
- [24] M. Ourak, B. Tamadazte, O. Lehmann, N. Andreff, Direct visual servoing using wavelet coefficients, IEEE/ASME Transactions on Mechatronics 24 (3) (2019) 1129–1140.
- [25] E. Marchand, Direct visual servoing in the frequency domain, IEEE Robotics and Automation Letters 5 (2) (2020) 620–627.
- [26] A. A. Hafez, S. Achar, C. Jawahar, Visual servoing based on gaussian mixture models, in: Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, IEEE, 2008, pp. 3225–3230.
- [27] N. Crombez, G. Caron, E. M. Mouaddib, Photometric gaussian mixtures based visual servoing, in: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, IEEE, 2015, pp. 5486–5491.
- [28] D. Gerogiannis, C. Nikou, A. Likas, Robust image registration using mixtures of t-distributions, in: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, IEEE, 2007, pp. 1–8.
- [29] D. Peel, G. J. McLachlan, Robust mixture modelling using the t distribution, Statistics and computing 10 (4) (2000) 339–348.

- [30] K. L. Lange, R. J. Little, J. M. Taylor, Robust statistical modeling using the t distribution, *Journal of the American Statistical Association* 84 (408) (1989) 881–896.
- [31] F. Chaumette, Potential problems of stability and convergence in image-based and position-based visual servoing, in: *The confluence of vision and control*, Springer, 1998, pp. 66–78.
- [32] B. K. Horn, B. G. Schunck, Determining optical flow, *Artificial intelligence* 17 (1-3) (1981) 185–203.