

# Probabilistic Integration of 2D and 3D Cues for Visual Servoing

A. H. Abdul Hafez<sup>1,2</sup>

<sup>2</sup> Dept. of Computer Science and Engineering  
University College of Engineering, Osmania University  
Hyderabad-500007, India  
Email: hafezsyra@ieee.org

C. V. Jawahar<sup>1</sup>

<sup>1</sup> Center for Visual Information Technology  
International Institute of Information Technology  
Gachibowli, Hyderabad-500032, India  
Email: jawahar@iiit.ac.in

**Abstract**—In this paper we present a new integration method for improving the performance of visual servoing. The method integrates image-based visual servoing (IBVS) and position-based visual servoing (PBVS) approaches to satisfy the widely varying requirements of the visual servoing process. We define an integration rule for IBVS and PBVS controllers. Density functions that determine the weighting factor of each controller are defined to satisfy the above constraints. We prove that this integration method provides global stability, and avoids local minima. The new integration method is validated on positioning tasks and compared with other switching methods.

## I. INTRODUCTION

Visual servoing exploits the visual feedback available in the control loop to increase the accuracy of the overall robotic system. This avoids the need to increase the accuracy of different electro-mechanical parts like end-effectors and sensors. In addition, visual feedback helps in controlling the robot pose with respect to a target even in the presence of calibration errors. As an optimization problem, visual servoing can be understood as the minimization of an error function of the object pose with respect to the camera. This error function could be defined in the Cartesian space [1], image space [2], or as a combination of both [3]. Visual servoing has become an attractive area of research, and has recently received considerable amount of attention [4], [5], [6].

Visual servoing techniques are divided into three categories: (i) image-based (IBVS), (ii) position-based (PBVS), (iii) and hybrid visual servoing. These divisions are based on the use of 2D information from the image space, 3D information from the Cartesian space, or a mixture of both kinds of information in defining the error function. While image-based visual servoing and position-based visual servoing have complementary advantages and disadvantages, hybrid methods attempt to incorporate the advantages of these both methods.

In addition to providing accurate control signal, a good visual servoing scheme has to address the following issues:

- 1) **Feature-visibility:** It is desirable for a visual servoing algorithm to keep the object features always visible in the camera field of view during the servoing process. This can help in obtaining reliable feedback signal during the servoing process.
- 2) **Local minima avoidance:** Getting trapped in a local minima during the optimization, results in the visual ser-

voing system converging to a final pose that is different from the desired one.

- 3) **Faster convergence:** Time of convergence is defined as the time needed to regulate the error function to the desired (or an acceptable) value. This time can be handled in the traditional methods like IBVS and PBVS while it is high in hybrid methods such as switching methods.
- 4) **Continuous control signal:** Discontinuity in the control signal can destabilize the process. This problem is of concern in switching methods. Other visual servoing methods provide a continuous control signal.
- 5) **Short camera path:** The camera path in the Cartesian space is a straight line if the error in the camera 3D position is minimized. However, non-straight camera path may cause the robot arm to get out of the Cartesian workspace.

In [7], we propose a unique algorithm, which satisfactorily addresses the above mentioned requirements of a visual servoing algorithm. A framework that integrates the IBVS and PBVS into one framework is proposed for this purpose. The integration process uses appropriate weights to determine the resultant control signal from the individual sub-controllers. These weights are computed through weighting functions which are designed to satisfy the above mentioned constraints. In this paper, we propose a method that computes the weights from single distribution by means of an energy function defined in the image and Cartesian spaces together. The method start by PBVS because all features are visible in each of the initial and desired views, and end by IBVS because IBVS is locally robust and stable. The proposed method solves the visibility and local minima problems efficiently. The camera trajectory is shown to be highly similar to the ideal straight path.

## II. RELATED WORK

Recent research in the visual servoing field has concentrated on development of algorithms which satisfy one or more of the requirements mentioned in Section I. The most comprehensive solutions are based on potential fields [8], [9], which was originally introduced to the robotic community by Khatib in [10] as a solution to the collision avoidance problem.

Mesouar and Chaumette [8] have recently proposed a potential field based method for path planning in the image space. This method introduces the visibility and robot joint limits constraints in the design of the desired trajectories. Essentially, this is a local path planning method where the local minima is not ensured to be avoidable when repulsive and attractive fields are equal. In addition, the camera trajectory is not predictable when the repulsive forces are involved. In addition to features visibility and joint limits, constraints like image singularity, image local minima, and robot singularity are also considered in [9]. Most of the earlier works tried to address the visibility problem without any attention toward the local minima and Cartesian camera trajectory. The method presented in [9] solved the local minima problem, but the camera path is neither straight line nor smooth.

Gans and Hutchinson [11] have proposed a switching approach between IBVS and PBVS. The proposed controller consists of two sub-controllers for IBVS and PBVS. Whenever the features tend to get out of the field of view (FoV) of the camera, the control switches to IBVS, and whenever the camera starts retreat, the control switches back to PBVS. However, the global stability is not ensured in this scheme. This binary switching allows either IBVS or PBVS controller to work at a specific instance. They did not consider the potential local minima neither in the image space nor in the Cartesian space. The potential local minima was demonstrated in [12]. In addition, a discontinuity in the velocity control signal can be observed at the time of switching between the two sub-controllers. This algorithm also needs more time to converge. The reason behind this is that the image error may increase in PBVS. Switching from IBVS to PBVS may increase the image error, which is minimized using the IBVS controller.

In Chesi *et al.* [5], a switching approach between elementary camera movements is proposed. The method decomposes the homography between the current and desired images and computes a translational and rotational velocity commands. A sequence of high level if-then-else rules is used to switch between camera rotation, translation, or backward translation along the camera optical axis. This method, shows a discontinuity in the velocity control signal in addition to longer time of convergence owing to the switching time to keep features in the FoV. The Cartesian local minima due to errors in homography estimation is not considered. In addition, the camera trajectory is not straight line any more because of the backward translation.

The method proposed in this paper is based on an integration scheme. It outperforms the reported methods available in the literature. Experimental results validate our claim in Section-VI.

### III. INTEGRATION OF IBVS AND PBVS

Consider that we have  $N$  individual visual servoing control laws represented by the parameters  $\alpha_i$  where  $i = 1, \dots, N$ . For example, if an integration is done between IBVS and PBVS control laws, then  $i$  takes the values from  $\{im, po\}$ .

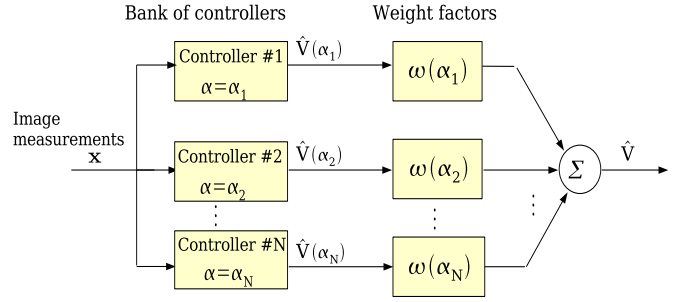


Fig. 1. Weighted sum of the velocity estimates

To integrate rotation and translation control laws,  $i$  takes the values from  $\{ro, tr\}$ . When visual servoing applications need to apply a pure backward translation to satisfy feature visibility in the image, then  $i$  may take the value  $\{ba\}$ . The optimal estimate of the velocity vector in the proposed integration framework is the weighted sum of the velocity values computed from each individual control law. This is shown in Fig (1). The problem is reduced to the computation of the velocity in each individual controller in addition to the computation of the importance factor of each of these controllers.

In this framework, image-based visual servoing and position-based visual servoing control laws can be integrated as

$$\hat{V} = \omega(\alpha_{im}) \hat{V}(\alpha_{im}) + \omega(\alpha_{po}) \hat{V}(\alpha_{po}). \quad (1)$$

Here,  $\hat{V}(\alpha_{im})$  is the velocity computed from the IBVS controller [2] that is given by

$$\dot{e}_i = J_i \hat{V}(\alpha_{im}) \quad (2)$$

$$\hat{V}(\alpha_{im}) = -\lambda_i J_i^+ e_i(s), \quad (3)$$

where  $J_i^+$  is the pseudo-inverse of the image Jacobian matrix  $J_i$  and  $e(s)$  is the image error vector. The velocity  $\hat{V}(\alpha_{po})$  is computed from the PBVS controller [2] that is given by

$$\dot{e}_p = J_p \hat{V}(\alpha_{po}) \quad (4)$$

$$\hat{V}(\alpha_{po}) = -\lambda_p J_p^{-1} e_p(s) \quad (5)$$

where  $J_p$  is the pose Jacobian matrix and  $e_p(s)$  is the pose error [1].

The importance factors of each individual control law  $\omega(\alpha_i)$ , conditioned on the image measurement  $x$ , is determined by the image and Cartesian constraints to be satisfied during the process of visual servoing.

The weighting factors are normalized such that they sum up to one. Indeed, we replace the notation  $\frac{\omega(\alpha_{im})}{\omega(\alpha_{im}) + \omega(\alpha_{po})}$  by  $\omega$  and  $\frac{\omega(\alpha_{po})}{\omega(\alpha_{im}) + \omega(\alpha_{po})}$  by  $1 - \omega$ . Therefore,

$$\hat{V} = \omega \hat{V}(\alpha_{im}) + (1 - \omega) \hat{V}(\alpha_{po}), \quad (6)$$

where  $0 < \omega < 1$ .

As mentioned above,  $\omega$  is a function of the image and Cartesian constraints, and of the energy of the task function to

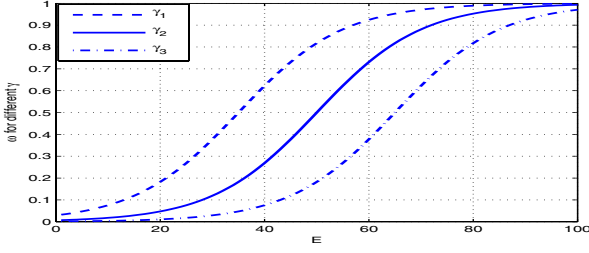


Fig. 2. The sigmoid function with different values of  $\gamma$ .  $\gamma_1 < \gamma_2 < \gamma_3$

be minimized. We define this energy function as the sum of two functions. The first one is defined as the energy function in the image space  $\mathcal{E}_i(t) = \frac{1}{2}e_i^T e_i$ , and the second one is defined as the energy function in the pose space  $\mathcal{E}_p(t) = \frac{1}{2}e_p^T e_p$ . Finally, the total energy function is written as

$$\mathcal{E}(t) = \mathcal{E}_i(t) + \mathcal{E}_p(t) = \frac{1}{2}(e_i^T e_i + e_p^T e_p). \quad (7)$$

Initially, the value of  $\omega$  is assumed to be small and near to zero. This allows us to start with PBVS where visibility and local minima constraints are already satisfied. With time the energy function decreases, and value of  $\omega$  should increase. This makes the system behave almost like IBVS near to the desired position. The task function converges to zero monotonically in the neighborhood of the desired position. A suitable function for these requirements is the sigmoid function

$$\omega = \frac{1}{1 + \exp[-(E - \gamma)]}, \quad (8)$$

where  $E = \frac{100(\mathcal{E}(0) - \mathcal{E}(t))}{\mathcal{E}(0)}$ . The behavior of this function with respect to  $E$  and  $\gamma$  is shown in Fig 2. In this figure one can see that the value of this function for a given  $E$  is small for larger  $\gamma$ .

#### A. Features-visibility and Local Minima

In IBVS, it is easy to show that the feature trajectory in the image space can be a straight line. If the initial and desired positions of the image points are well chosen in such a way that all features are in the camera field of view, the image points are ensured to be in the FoV during the whole process. This is subject to the availability of a good estimate of the depth and robust image measurements. The weighting factor of the image-based visual servoing controller  $\omega$ , conditioned on the image measurement  $x$ , increases when one of the image points approaches the image boundary.

Given  $N$  image points as feature, let us define the vector  $D$  where its element  $D_i$  is the distance vector of the  $i$ th point to the nearest image boundary. The weighting factor  $\omega$  should increase when  $D_{min}$  is decreasing, where  $D_{min}$  is

$$D_{min} = \min_{D_i}(D).$$

Therefore,  $\gamma$  should decrease in order to increase the weighting factor  $\omega$ . The decrement amount in  $\gamma$  may be proportional to  $e^{-D_{min}^2}$ .

The evaluation of image local minima is done in terms of the energy function in the image space

$$\mathcal{E}_i = \frac{1}{2}(s_i - s_d)^T (s_i - s_d), \quad (9)$$

and its gradient vector  $G_i(t)$  with respect to time  $t$ . The image feature  $s$  represents, in case of image point, the image coordinates. At every time instant  $t$ , the gradient vector  $G_i(t)$  of the energy function  $\mathcal{E}_i$  is given as

$$G_i(t, s) = \frac{\partial \mathcal{E}}{\partial t} \Big|_{t,s} = (s_t - s_d)^T \frac{\partial s_t}{\partial t} = (s_t - s_d)^T J_i \hat{V}, \quad (10)$$

where  $J_i$  is the image Jacobian. Image local minima occurs when the norm of the gradient vector  $\hat{G}_i(t)$  is near the zero. Therefore, the weighting factor  $\omega$  should decrease at or near to the image local minima. This means increasing  $\gamma$ . The increment amount in  $\gamma$  may be proportional to the amount  $e^{-\hat{G}^2}$ .

Finally, the parameter  $\gamma$  is given as

$$\gamma = -e^{-D_{min}^2} + e^{-\hat{G}^2}. \quad (11)$$

From (11), (8), and Fig 2, one can observe that the weighting factor  $\omega$  increases when one of the features point approaches the image border. However, it decreases near to the image local minima.

#### IV. PROBABILISTIC INTEGRATION FRAMEWORK

The integration rule given in (6) can be arrived from a probabilistic formulation. The weights  $\omega(\alpha_i)$  represents the probability of using a certain control law.

Let  $\mathcal{V} = \{V_i\}$ , where  $\mathcal{V} \subseteq R^6$  be a set of possible states of the velocity screw vector commanded to the robot arm controller. The probability that the velocity vector  $V$  uses the value  $V_i$  conditioned to the image measurement vector  $x$  is

$$p(V = V_i|x) = p(V_i|x). \quad (12)$$

By convention, these probabilities sum to unity  $\int_{\mathcal{V}} p(V_i|x) = 1$ . The conditioned expected value of the velocity vector over the universe  $\mathcal{V}$  is

$$\hat{V} = E(V_i|x) = \int_{\mathcal{V}} V_i p(V_i|x) dV_i. \quad (13)$$

It is possible to rewrite the probability function  $p(V_i|x)$  as

$$p(V_i|x) = \int_{\alpha} p(V_i, \alpha|x) d\alpha = \int_{\alpha} p(V_i|\alpha, x) p(\alpha|x) d\alpha. \quad (14)$$

By substituting (14) in (13), the conditional mean of the velocity can be rewritten as

$$\hat{V} = \int_{\alpha} p(\alpha|x) \hat{V}(\alpha) d\alpha, \quad (15)$$

where,  $\hat{V}(\alpha) = \int_{\mathcal{V}} V_i p(V_i|\alpha, x) dV_i$ , and in the discrete case

$$\hat{V} = \sum_{\alpha_i} \hat{V}(\alpha_i) p(\alpha_i|x), \quad \text{where } \sum_{\alpha_i} p(\alpha_i|x) = 1. \quad (16)$$

Here  $p(\alpha_i|x)$  is the discrete probability for  $\alpha_i$ , conditioned on the image measurement  $x$ . Assuming that the value of  $i$  is in the range  $\{im, po\}$ , and replacing  $p(\alpha_i|x)$  by  $\omega(\alpha_i)$ , equation (16) reduced to (1) and consequently (6).

## V. STABILITY ANALYSIS

### A. Local Stability

Local stability is considered only in a neighborhood of the convergence (equilibrium) point. An equilibrium point is locally asymptotically stable if all solutions starting in a small neighborhood converge to the equilibrium point. stability of equilibrium point is characterized by Lyapunov function. Considering linear systems, the local and global stability are same. The necessary and sufficient condition for local/global asymptotic stability of a linear system  $\dot{e} = \mathcal{A}e$  is that the matrix  $\mathcal{A}$  describing the system has negative real parts for all its eigenvalues. Eigenvalues with zeros real part do not void the stability. However, they need not give robustness because it may shift to the positive part when the system is affected by noise or disturbance. In fact, our system is a convex combination of two linear systems; the matrices  $J_i$  and  $J_p$  are fixed and non-singular in a neighborhood of the equilibrium point. Stability for such a system needs to ensure that there exist a Lyapunov function for this combination.

It was shown in [7] that our integration system is a convex combination system  $\dot{e} = \mathcal{A}e$ . Here,

$$\mathcal{A} = \omega A_1 + (1 - \omega)A_2, \quad (17)$$

$$\text{where } A_1 = \begin{bmatrix} -I_6 & 0 \\ -J_p J_i^+ & 0 \end{bmatrix} \text{ and } A_2 = \begin{bmatrix} 0 & -J_i J_p^{-1} \\ 0 & -I_6 \end{bmatrix}.$$

As it is proved in [7],  $\forall \omega \in [0, 1]$ , the convex combination of the systems given in (17) is stable and there exist a quadratic common Lyapunov function. A common Lyapunov function means that there exist symmetric positive definite matrices  $P$  and  $Q$  such that  $A^T P + P A = -Q$ . In other words,

$$\omega (A_1^T P + P A_1) + (1 - \omega) (A_2^T P + P A_2) = -Q, \quad (18)$$

$$\begin{aligned} \omega e^T (A_1^T P + P A_1) e + (1 - \omega) e^T (A_2^T P + P A_2) e \\ = -e^T Q e < 0, \end{aligned} \quad (19)$$

$\forall e \in R^n \setminus \{0\}$ . In other words,  $R^n \setminus \{0\}$  is covered by the union of the two open conic regions: (i)  $\mathfrak{R}_1 = \{e, e^T (A_1^T P + P A_1) e < 0\}$ , and (ii)  $\mathfrak{R}_2 = \{e, e^T (A_2^T P + P A_2) e < 0\}$ . Thus, the function  $V(e) = e^T P e$  decreases along solutions from  $A_1$  in region  $\mathfrak{R}_1$  and along solutions from  $A_2$  in region  $\mathfrak{R}_2$ . While all matrices in the matrix pencil  $\gamma_\omega(A_1, A_2)$  are stable, the Lyapunov function  $V(e)$  decreases along solutions for all  $\omega$ .

### B. Global Stability

It was proved in [13] that for both IBVS and PBVS, there exist a range of camera poses where under a visual servoing algorithm both image error  $e_i$  and pose error  $e_p$  decrease to zero. In addition, if the pose error consists of a translation and sufficiently small rotation, the system is within this region. Consequently, pose error with a translation and sufficiently small rotation can be regulated to zero either using IBVS or PBVS.

To analyze the behaviour of the proposed integrated system in the neighborhood of an image local minima, a 2D projection

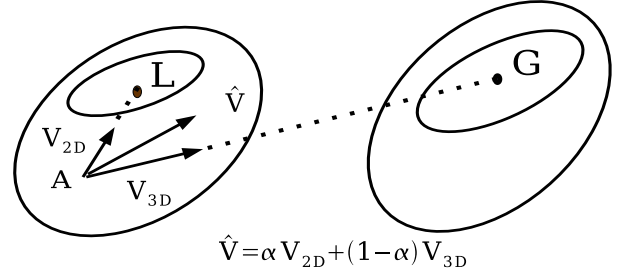


Fig. 3. Representation of the velocity vectors in 2D space.

of the velocity vector is presented in Fig 3. The point A represents the current work point in the neighborhood of an image local minima represented by the point L. The point G represents the global minima. In the neighborhood of L, the direction of  $\hat{V}(\alpha_{im})$  is toward L. Where PBVS is globally stable or at least local minima may occur in the neighborhood of the global one G, the vector  $\hat{V}(\alpha_{po})$  is independent on L and always points toward the global minima G. It is clear that the weighted sum of these two vectors  $\hat{V}(\alpha_{im})$  and  $\hat{V}(\alpha_{po})$  will drive the system away from the local minima L toward the global minima G. After a few iterations, the system escape from the local minima by getting out of its neighborhood. In case of L is on the straight line from A to G, the weight  $\omega$  is designed to be much smaller than  $1 - \omega$  in the neighborhood of the local minima L. Regardless the rotation error has been increased or decreased in the neighborhood of the local minima, it will start decreasing again after escaping it up to the reaching the stability region. In the stability region *i.e.* the neighborhood of the global minima, both image error and pose error decrease monotonically to the global minima.

We argue that the rotation part in the pose error decreases to small value in spite of the existence of an image local minima. This allows the translation part to start decreasing when the rotation error becomes sufficiently small. Because  $0 < \omega < 1$ , the system avoids the local minima in the image and Cartesian spaces. In fact, the local minima in IBVS and PBVS are not correlated and do not tend to happen together. This justifies the claim that our method is globally asymptotically stable.

## VI. SIMULATION AND RESULTS

We present the simulation experiments where our proposed method is compared to the previous hybrid methods, namely switching methods proposed by Gans and Hutchinson [11], [13] and the one proposed by Chesi *et al.* [5]. In the remaining of this paper and for comparison purpose, we call the first method as Gans' method and second one as Chesi's method. These two methods, in addition to the proposed, are implemented.

Comparison is carried out for two positioning tasks. First one is with a 180 degrees rotation error around the camera optical axis. Using this task we evaluate performance of the algorithm for keeping the visibility of features. The second task is a general positioning task that contains rotational

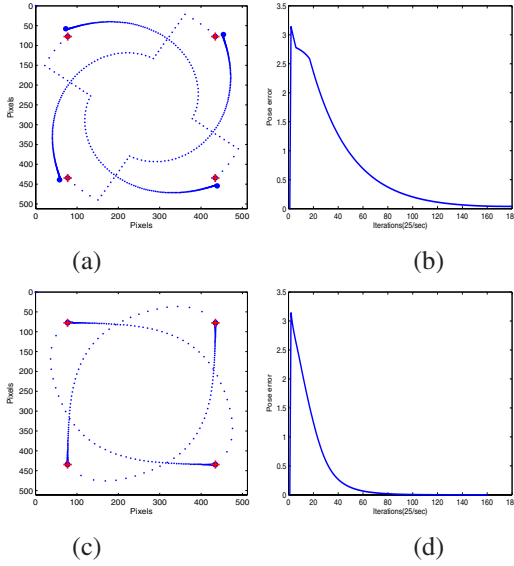


Fig. 4. The image features trajectories and pose error in presence of Cartesian local minima. Gans method in the first row and our integration method in the second row.

and translational errors. The task is useful to evaluate the time of convergence and the continuity of the control signal. The servoing target object consists of four planar points. We assume a perspective projection camera model with unit aspect ratio. The processing rate is considered to be fixed at 25 Frames/Sec.

#### A. Feature-visibility and Local Minima

The positioning task with 180 degrees rotation error is attempted here. Our proposed method is compared to Gans' method. Ours provided satisfactory results for the features visibility as shown in Fig 4(a). Gans' method started with PBVS, producing a pure rotation about the camera optical axis. When the image features approach the image border, the control switches to IBVS, in order to keep the image features visible. IBVS produces a straight line image trajectory but it is accompanied with a camera retreat in the Cartesian space. At the retreat threshold, the control switches back to PBVS to produce the pure rotation about the camera optical axis again. The image features stay far from the image border the control will continue with PBVS till the desired pose. Owing to errors in the pose estimation process, the system converges to a local minima in the neighborhood of the desired pose. In Fig 4(a), the final position of the image features different from the desired one while the pose error in Fig 4(b) converges to a small value near to its zero desired value. Figures 5(a) and 5(b) show the local minima where the velocity converges to zero and the image error does not.

Our method successfully keeps the image features visible (see Fig 4(c)) during the servoing process. In the same figure one can see that the image features converge perfectly to the desired position depicting a global minima. The method also behaves perfectly with respect to the Cartesian local minima where the IBVS controller is still working and avoids the

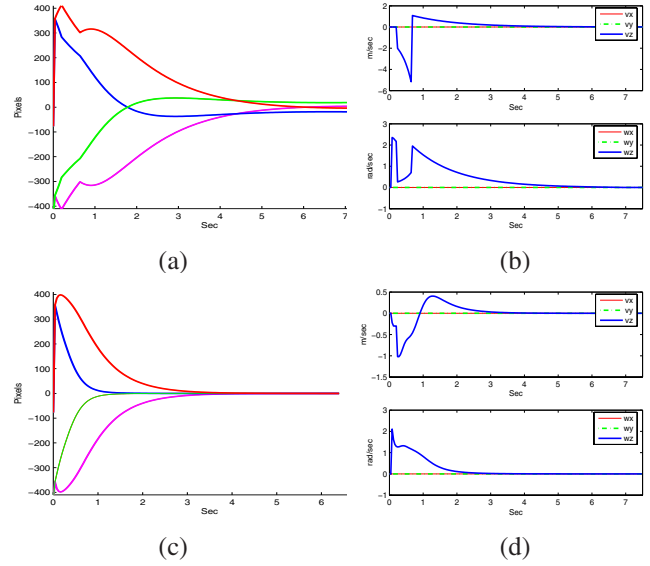


Fig. 5. The image error and velocity command. Gans method in the first row and our integration method in the second row.

local minima. This is clear from the pose error in Fig 4(d). The image error and velocity command in Figs 5(c) and 5(d) converge both to zero that depicts a global minima.

#### B. Discontinuity and Convergence Time

To evaluate the time of convergence and the smoothness of the velocity control signal, we use a general positioning task with both rotational and translational error. We compare it with both Chesi's and Gans' methods. Gans' method (Figs 6(a) and 6(b)) shows a discontinuity in the control signal due to the switching between IBVS and PBVS. In Chesi's method (Figs 6(c) and 6(d)), there are a frequent switching. This means a discontinuity in the control signal. In contrast to these two methods, our method (Figs 6(e) and 6(f)) provides a continuous and smooth control signal *i. e.* the velocity screw command.

The switching design of Gans' and Chesi's methods, in order to keep the visibility of the image features, came out with a delay in the convergence. This is clear from the image error graphs in Fig 6(a), 6(c), and 6(e). From the figures one can note that our integration method converges within 3-4 Sec, while Gans' method converges within 6-7 Sec and Chesi *et al.* method needs 9-10 Sec to converge for the same task.

A comparison of the the camera trajectory is shown in Figure 7. Our method and Gans' method show similar camera trajectory while Chesi's method shows the worst path that similar to the one given using IBVS methods.

## VII. CONCLUSION AND FUTURE WORK

The method presented in this paper describes a unique solution that satisfies a set of widely varying requirements in visual servoing literature. These requirements are such as feature visibility, local minima, and straight camera trajectory. The method gives similar performance to previous methods like Gans' one with respect to image features visibility. However, it gives faster convergence and continuous velocity control

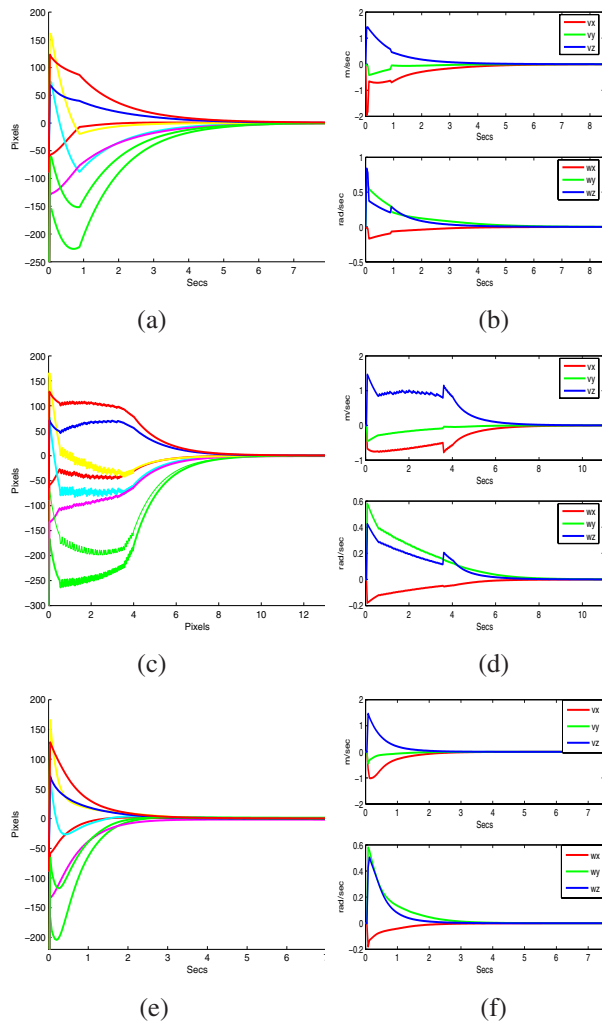


Fig. 6. The image error (left) and velocity command (right) show the discontinuity in Gans' method, (a) and (b), and Chesi's, (c) and (d). Our integration method shows faster convergence and smooth velocity control in (e) and (f).

signal in contrast to the discontinuity present in other switching methods like Gans' and Chesi's ones.

This method is a part of larger activity that aims to design a unified frame work to estimate the 3D information online. Then, use these information to improve the performance of visual servoing using this integration method.

#### ACKNOWLEDGMENT

The authors would like to acknowledge Dr. F. Chaumette and Dr. E. Cervera for their valuable feedback on the early draft of this paper.

#### REFERENCES

[1] W. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative endeffector control using cartesian position-based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, October 1996.

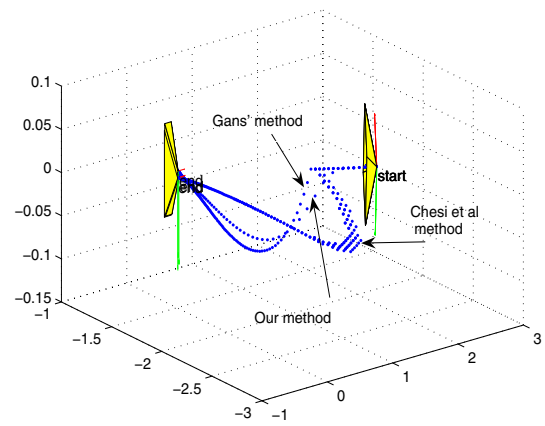


Fig. 7. The camera trajectory of the three methods. Ours and Gans' methods produce similar trajectories while Chesi's one show the worst path in the 3D space.

[2] S. Hutchinson, G. Hager, and Cork, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 18–27, 1996.

[3] E. Malis, F. Chaumette, and S. Boudet, "2 1/2 d visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, April 1999.

[4] V. Kyrki, D. Kragic, and H. I. Christensen, "New shortest-path approaches to visual servoing," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'04*, Sendai, Japan, September–October 2004, pp. 349–354.

[5] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: A switching approach," *IEEE Trans. on Robotics*, vol. 20, no. 5, pp. 534–549, Oct 2004.

[6] J. Pags, C. Collewet, F. Chaumette, and J. Salvi, "An approach to visual servoing based on coded light," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2006*, Orlando, Florida, May 2006.

[7] A. H. Abdul Hafez and C. V. Jawahar, "Probabilistic integration framework for improved visual servoing in image and cartesian spaces," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'06*, Beijing, China, October 2006.

[8] Y. Mezouar and F. Chaumette, "Path planning for robust image-based control," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 4, pp. 534–549, Aug 2002.

[9] L. Deng, F. Janabi-Sharifi, and W. J. Wilson, "Hybrid strategies for image constraints avoidance in visual servoing," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'02*, Switzerland, October 2002, pp. 348–353.

[10] O. Khatib, "Real time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.

[11] N. R. Gans and S. A. Hutchinson, "An experimental study of hybrid switched system approach to visual servoing," in *IEEE Int. Conf. on Robotics and Automation, ICRA'03*, Taiwan, September 2003, pp. 3061–3068.

[12] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The Confluence of Vision and Control*, D. Kriegman, G. Hager, and A. Morse, Eds. LNCS Series, No 237, Springer-Verlag, 1998, pp. 66–78.

[13] N. R. Gans and S. A. Hutchinson, "An asymptotically stable switched system visual controller for eye in hand robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'03*, Las Vegas, October 2003, pp. 3061–3068.