Visual Servoing in Non-Rigid Environments: A Space-Time Approach

D Santosh and C V Jawahar

Abstract— **Most robotic vision algorithms are proposed by envisaging robots operating in structured environments where the world is assumed to be rigid. These algorithms fail to provide optimum behavior when the robot has to be controlled with respect to active non-rigid targets. This paper presents a new framework for visual servoing that accomplishes the robot positioning task even in non-rigid environments. We introduce a space-time representation scheme for modeling the deformations of a non-rigid object and propose a model-free hybrid approach that exploits the two-view geometry induced by the space-time features to perform the servoing task. Our formulation can address a variety of non-rigid motions and can tackle large camera displacements without being affected by the degeneracies in the task space. Experimental results validate our approach and demonstrate the robust and stable behavior.**

Fig. 1. Proposed scheme for visual servoing in non-rigid environments

I. Visual Servoing in Presence of Non-Rigid Motion

The problem of controlling the movement of robotic systems using visual feedback has been a topic of substantial research in the field of Visual Servoing [10]. Several influential approaches in this area have been envisaged to perform the servoing task [4], [13]; however, much of the research until now presumes structured and rigid environments. In this paper, we propose an approach to visual servoing that can control a dynamic system even in an unknown active non-rigid environment.

In robotic vision research, motion analysis has been largely restricted to rigid objects due to their simplicity, elegance and immediate industrial applicability. However, in real world situations, motion of physical objects is often non-rigid [2] in nature. Common examples include motion of human body, flying birds, ocean waves etc. It has been a persistent desire to employ robots in such natural and unconventional environments. For this to be successful, it is desirable to develop servoing strategies and algorithms that can perform optimally even in such unstructured scenarios. Our motivation towards non-rigid motion analysis has been driven by applications in the areas of surgical robotics, underwater robotics, active vision systems etc.

Dealing with non-rigid motion poses several challenges in the design of optimal servoing strategies. Non-rigid objects undergo a persistent change in their pose which forbids any single image to characterize their state. This is because motion instruction planned based on the features extracted at current time instant might not be relevant in the next instant as the object undergoes a change in its pose. Further, the
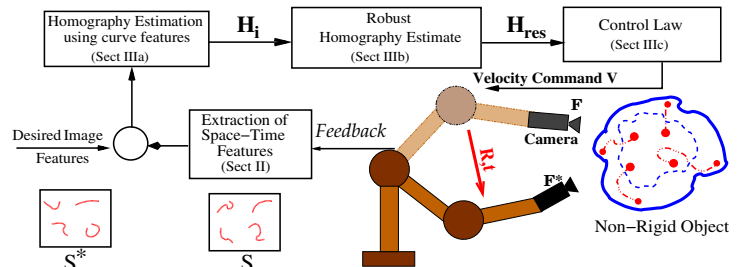
D Santosh and C V Jawahar are with the Center for Visual Information Technology, International Institute of Information Technology, Hyderabad 500032, India {santosh@research.,jawahar@}iiit.ac.in

desired configuration of the end-effector cannot be described by using only a single image or a single pose as it will lead to oscillations of the manipulator even after the goal position is reached. Note that the unavailability of static features (in case of whole body deformations) and background features (in case of moving targets) makes it imperative to engender new representation schemes for visual servoing using only the pose-varying features on the object surface. This necessitates a time-based representation, rather than a purely spatial one, due to the temporal nature of the object deformations. It must be emphasized that non-rigid motion encompasses wide range of possible motions ranging from simple translatory motion such as a waving hand to highly complex motion like that of a beating heart. A general representation for all kinds of motions is preferable, but appears to be inconceivable at this stage. Establishing correspondence between image features is usually the primary step in visual servoing. However, finding accurate correspondences is often difficult in practical situations; especially, while matching points in two views separated by large displacement. This is a highly formidable requirement in case of deformable objects as this demands frame-to-frame matching of the object deformations which is complicated even for simple motions.

Existing servoing schemes are not designed to tackle non-rigidity. Cartesian-based algorithms require complete $3D$ information of the object which is a strong assumption for deformable targets. Image-based servoing schemes cannot be directly used as these schemes use information only from a single image to guide the robot, which results in an oscillatory camera trajectory [15]. Also, these methods are not completely model-free, since depths of the observed features are needed in the control law [10]. Further, they demand the exact frame-to-frame correspondences between image features. Moreover, new representations conceived for modeling the non-rigid motion cannot be directly utilized in these schemes as the corresponding interaction matrix relating the feature motion in the image space to the camera

motion in the Cartesian space has to be derived.

In this paper, we propose a different approach to visual servoing in which the motion characteristics of active non-rigid objects are used to perform the servoing task, without the requirement of $3D$ structure information. Our approach is based on the bi-dimensional appearance of the objects in the environment and explicitly takes into account independent object motions. In most cases, where an object has a repetitive motion, the space-time trajectories of representative points on it will serve to uniquely represent the object (Sect. II). These trajectories are invariant to object deformations and can be utilized to obtain a stable estimate of the projective transformation relating the initial and desired views (Sect. III-A). The estimated transformation is then used in a feedback-based hybrid control to perform the servoing task (Sect. III-C). The overall algorithm is summarized in Fig. 1. In [15], a preliminary strategy to tackle non-rigidity was discussed. In that approach, gross features of the object deformations were extracted and used in the servoing algorithm. However, the method handles only simple non-rigid motions. Further, issues of optimal camera trajectory, degenerate configurations (such as local minima, singularity) etc. have not been analyzed. In the current formulation, we aim to not only generalize our approach to complex non-rigid motions but also achieve the desirable characteristics of the servoing algorithm (Sect. III).

II. GEOMETRY OF NON-RIGIDITY

Active non-rigid motion can essentially be classified into three primary types, namely articulated motion, elastic motion and fluid motion [2]. This classification is based on the constraints on the degree of the smoothness and continuity in the motion. Among the different forms of non-rigidity, elastic motion constitutes the most general form of non-rigid motion [2]. Elastic or cyclic motion is ubiquitous in the natural world. For instance, the motion of heart and other body organs; motion of flying birds, swaying trees and moving aquatic animals; ambulatory motion of humans and animals etc. All such motions involve a regularly repeating sequence of motion events and thus demonstrate a cyclic pattern in their deformations. In this paper, we concentrate on accomplishing the servoing task in presence of such stationary elastic objects. It may be noted that global motion from a moving non-rigid object can be separated by performing rigid and non-rigid motion segmentation [3].

Different modeling strategies have been proposed in the field of computer vision to characterize non-rigidity. Most approaches employ methods like the linear subspace model (appearance manifold), kinematic chains, dynamic Markov models etc. to model the deformations as variations to the model parameters. In these methods, assumptions regarding the objects and their motion are made, which restrict the class of objects that can be handled. A standard modeling scheme for all kinds of motion is very much desirable for the design of a general servoing strategy. Our pursuit is to engender a stable representation for a generic non-rigid object.
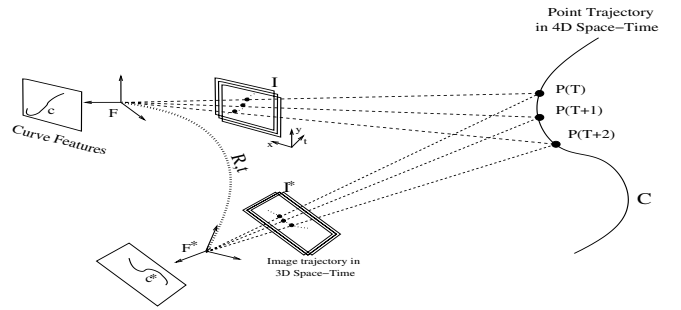

Fig. 2. Projective transformation in the Space-Time: Point trajectory in the $4D$ space-time projects onto the $3D$ space-time describing a curve in the image space

A. Non-Rigid Motion as Space-Time Curves

We represent a non-rigid object using a set of representative points moving with different velocities [12]. These interest points are locations where the object deforms in shape, and constitute to its surface appearance. A deformation of the object induces a change in the point locations. For a stationary target, its deformations can be described using the motion of these configuration of points. This is because repeated activity transforms the points such that they traverse a fixed trajectory in the three-dimensional space. For a static observer perceiving the object, these points always appear to traverse along the same $3D$ curves.

Projections from $4D$ to $3D$ Space-Time More formally, let $\mathcal{O}$ be the observed object and $P_1, \ldots, P_n$ be the interest points on the object surface in the $3D$ space. Let the $3D$ coordinate of the point P_i, $i = 1, 2, \ldots, n$ be $[X_i\ Y_i\ Z_i]^T$. The motion of this point in the Euclidean space can be considered as a set of points $\mathcal{P}_k = [X_k\ Y_k\ Z_k\ T_k]^T$ defining a curve $\mathcal{C}_i$ in a $4D$ space-time where T_k denotes time. Assuming a pin-hole camera, the space-time projection of the point onto the $3D$ (image) space-time satisfies

$$\widetilde{\mathbf{p}_k} \approx \mathcal{M}\widetilde{\mathcal{P}_k}, \tag{1}$$

where matrix $\mathcal{M}$ denotes the 4×5 extended camera matrix, $\approx$ denotes equality up to scale and $\widetilde{\ }$ denotes corresponding homogeneous coordinates. Although the space-time projection from $\mathcal{P}_k$ to $\mathbf{p}_k$ cannot be described by projective cameras, (1) signifies that a point in the real space-time is projected to an image space-time point by an extended affine camera. Thus the motions in the $4D$ space are projected onto images and can be observed as a set of points $\mathbf{p} = [x\ y\ t]^T$ in a $3D$ space-time on image motions extracted from an image sequence (See Fig. 2). This $3D$ space-time can be perceived as a spatio-temporal entity with two spatial dimension x, y and a time dimension t. The corresponding image coordinates $\mathbf{m}$ can be obtained from the normalized coordinates $\mathbf{p}$ with an affine transformation

$$\mathbf{m} = K\mathbf{p}, \tag{2}$$

where K is the camera intrinsic matrix [8]. These points define the image trajectory $\boldsymbol{c}_i$ of the $4D$ space-time curve $\mathcal{C}_i$.

Curves have been employed in computer vision for a very long time. However, most of the works until now refer to them in the context of shape descriptors. In this paper, the concept of 'space-time curves' is being introduced for visual servoing purpose. Such a representation offers multiple benefits. First, such features allow large class of motions to be accommodated as few constraints are enforced on the kinds of motion. As they are invariant to the changes in the object pose, they provide a stable and unique set of features for visual servoing. Moreover, the desired configuration of the manipulator can be stably defined using these invariant features. Further, such a representation avoids the complex problem of establishing frame-to-frame feature correspondences during the servoing task. The space-time curves provide a more geometric and intuitive representation of the object than other past features and thus are more interesting. Compared to finding corresponding points, corresponding curves can be easily and robustly identified using multiple cues e.g., periodicity, curvature etc.

B. Navigation Formulation

Let $\mathcal{F}_0$ be the coordinate frame attached to the target, and $\mathcal{F}^*$ and $\mathcal{F}$ be the coordinate frames attached to the calibrated cameras at the desired and current positions respectively. Let $\mathcal{F}^*$ be displaced from $\mathcal{F}$ in the Euclidean space by $\mathbf{R} \in SO(3)$ and $\mathbf{t} = [t_x, t_y, t_z]^T \in \Re^3$, where $\mathbf{R}, \mathbf{t}$ denote the rotation matrix and the translation vector respectively. Considering the angle-axis representation for rotation matrix $\mathbf{R}$, we have $\mathbf{R} = exp([\mathbf{r}]_\times)$, where $\mathbf{r} = u\theta$, is the vector containing the angle of rotation θ and the axis of rotation u, exp is the matrix exponential function and $[\mathbf{r}]_\times$ is the skew-symmetric representation of the vector $\mathbf{r}$. The relative camera pose with respect to frame $\mathcal{F}^*$ is defined by a 6×1 vector $\mathcal{E} = [\mathbf{t}^T \ \mathbf{r}^T]^T$. The points on the space-time curve $\mathcal{C}_i$ in the current frame $\mathcal{F}$ get transformed to desired frame $\mathcal{F}^*$ as $P^* = \mathbf{R}P + \mathbf{t}$, defining the curve $\mathcal{C}_i^*$. These points project onto the normalized plane $\mathcal{I}^*$ as $\mathbf{p}^*$ and their corresponding image coordinates are obtained using the relation $\mathbf{m}^* = K\mathbf{p}^*$ (similar to (2)). These image points define the trajectory $\mathbf{c}_i^*$ in the desired frame $\mathcal{F}^*$ (See Fig. 2).

In visual servoing, the goal is to reduce the error in the desired and the current features so as to drive the disparity in pose between $\mathcal{F}$ and $\mathcal{F}^*$ to zero. The objective function $\mathbf{e}$ can be defined as a function of time t as

$$\mathbf{e}(t) = \mathbf{c_i}(t) - \mathbf{c_i^*}. \tag{3}$$

Since the image features are a function of the camera pose *i.e.,* $\mathbf{c_i}(t) = f(\mathcal{E}(t))$, (3) can be redefined as

$$\mathbf{e}^{'}(t) = \mathcal{E}(t) - \mathcal{E}^*. \tag{4}$$

Thus the servoing task reduces to the problem of estimation of the partial displacement of the camera followed by a minimization of error in the relative pose parameters. We assume that the frame rate of the camera is sufficiently high for capturing one cycle of the points trajectory.
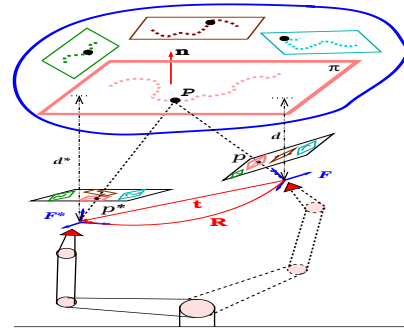

Fig. 3. Homography-based visual servoing using planar curves: The curves indicate the space-time trajectory of the points on the non-rigid object

III. PROPOSED SERVOING APPROACH

The goal of our solution is to design a servoing algorithm to drive the disparity between the current and the desired camera configurations to zero. The desirable characteristics of the algorithm are –

- Immunity to non-rigid deformations and continuity in the velocity screw.
- Ability to tackle large camera displacements without being affected by degeneracies in the task space.
- Robustness to image measurement errors.
- Independence from prior knowledge of the object model and parameter initialization.
- Decoupling of camera motion and ensuring feature visibility.

We employ a hybrid homography-based formulation to achieve the above desirable characteristics. In literature, model-free hybrid approaches have been developed to deal with unknown environments [5], [13]. These methods exploit the information provided by the projective reconstruction of the scene computed only from the visual features extracted from the images.

We begin with a simple, not so strict assumption that the motion of a point on a non-rigid object can be approximated to a planar motion [7]. The most general motion of a sufficiently small element of a deformable object can be represented in three mutually orthogonal directions (*i.e.,* as a sum of translation, rotation and an extension). However, in presence of opaque objects, the visible deformations are those occurring at the object surface. These deformations can be assumed to occur locally on the plane. Hence the dominant motion of the points can be assumed to be planar in nature. Note that each curve can be planar in any orientation while the object is non-planar (See Fig. 3).

A. Homography-based Visual Servoing

Given the planarity assumption of the point trajectories, the projective transformation between the two views of the scene can be defined using a homography [8]. Several methods have been proposed in literature to estimate homography from planar curves [1]. However, most of the approaches deal with parametric curves. Since non-rigid motion can be complex, parametric methods might not be capable of estimating homography in all situations as they
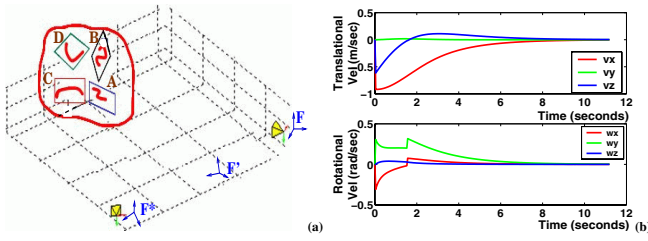
Fig. 4. Effect of degenerate configurations: (a) Servoing begun using reference curve A reaches a degeneracy at F' whose origin lies on the reference plane containing curve A (b) Discontinuity in the velocity screw is due to the switching of curves (A to C) at F'

are dependent on the chosen parametrization and cannot handle changes in curve topology. To accommodate arbitrary large class of motions, we extend the notion of higher order primitives further to include an ordered collection of points on a contour. We employ the homography estimation from contours technique proposed in [11]. In this method, the homography is estimated using only an ordered set of pixels of the contour, without the requirement of explicit point to point correspondences. The algorithm converges to the actual homography in few iterations and is robust to outliers and errors in image measurements.

The recovered homography can then be decomposed to obtain the rotation matrix R, the scaled translation vector $\frac{t}{d}$ and the plane normal n using the procedure described in [6]. It must be emphasized that information from multiple homography estimates, computed for a set of point trajectories, can be utilized to unambiguously decompose H without the requirement of *a priori* knowledge or parameter initialization. The motion parameters are then used in the control law to generate the optimal velocity instruction. In [13], a similar method is discussed by Malis *et al.* wherein the homography between two views of a planar contour is estimated and the parameters obtained from its decomposition are used in a $21/2D$ control to servo the end-effector.

B. Reliable Homography Computation

A single homography estimate is not sufficient when the camera has to undergo large displacements in visual servoing as it can be affected by occlusion of visual features, tracking (drift) errors, the camera center approaching the feature plane or due to singular homographies [8]. These degenerate configurations render the homography invalid. In either of the cases, when a degeneracy is reached, a new planar curve has to be chosen as the current curve can no longer be used for estimating the two-view projective transformation. This switching causes a discontinuity in the velocity screw affecting the stability of the robotic system. In Fig. 4, the effect of switching is demonstrated, where a positioning task with respect to a non-rigid object is simulated. The above limitation is caused by the fact that only information from a single homography is being utilized at a time. To circumvent this problem, we compute a robust and reliable estimate of the homography using information from multiple homography estimates as described in [14]. In this method,

the resultant homography is efficiently computed as a linear combination of four independent homographies by exploiting the rank-4 subspace constraint on homographies.

C. Control Law

Given the stable estimate of the motion and structure parameters (*i.e.*, $R, \frac{t}{d}, n$), we describe a control law to obtain a decoupled camera trajectory without loosing the visibility of features during servoing. Our approach is motivated by the controls presented in [5] and [13].

The translational velocity to move directly to the goal can be determined as $-\lambda_v(\frac{t}{d}) d$, where λ_v is a gain factor and d is the distance to the plane (See Fig. 3). The rotational velocity is computed as $-\lambda_\omega \mathbf{u}\theta$, where λ_ω is again a gain factor and u, θ denote the rotation axis and angle that are obtained using the Rodriguez formula as $\theta = \arccos(\frac{1}{2}(tr(R) - 1))$, $[\mathbf{u}]_\times = \frac{R - R^T}{2 \sin c(\theta)}$, where $tr(R)$ indicates the trace of matrix R. A direct control in the Cartesian space might result in the features leaving the camera field of view. However, information from image features can be incorporated into the decoupled control to enforce the visibility constraint. We know from the image-based visual servoing control [10]

$$\left[\begin{smallmatrix} u - u^* \\ v - v^* \end{smallmatrix} \right] = \left[\underbrace{\begin{smallmatrix} -\frac{1}{Z} & 0 & \frac{u}{Z} \\ 0 & -\frac{1}{Z} & \frac{v}{Z} \end{smallmatrix}}_{L_\nu} \underbrace{\begin{smallmatrix} uv & -(1 + u^2) \\ 1 + v^2 & -uv \end{smallmatrix}}_{L_{\omega_{xy}}} \underbrace{\begin{smallmatrix} v \\ -u \end{smallmatrix}}_{L_{\omega_z}} \right] \left[\begin{smallmatrix} \nu \\ \omega \end{smallmatrix} \right],$$
(5)

where $p = [u \; v \; 1]^T = [x \; 1]^T$, $p^* = [u^* \; v^* \; 1]^T = [x^* \; 1]^T$ and $Z = \mathbf{Z}(P)$ (See Fig. 3), while $[\nu \;\; \omega]^T$ denotes the velocity screw. Equation (5) can be rewritten as $x - x^* = [L_\nu \; L_{\omega_{xy}} \; L_{\omega_z}][\nu \; \omega_{xy} \; \omega_z]^T$, which yields

$$\omega_{xy} = L_{\omega_{xy}}^{-1}[(x - x^*) - L_\nu \nu - L_{\omega_z}\omega_z],$$
(6)

where $\nu = (\frac{t}{d})\hat{d}$ and $\omega_z = \mathbf{u_z}\theta$. In (6), the rotational motion ω_{xy} is controlled not only to minimize the differences between the current and the goal image features but also to compensate the effects caused by translation on the image. This ensures a straight-line trajectory of the features in the image. Estimates of Z and d are required in (5) and (6). This can be obtained as $\hat{Z} = \frac{\hat{d}}{n^T p}$ and $\hat{d} = \frac{\hat{d}^*}{det(H)}$, where $\hat{d}^*$ is an estimate of the constant distance to the plane π in $\mathcal{F}^*$ [13]. In general, this quantity is considered as a gain ratio and a coarse estimate obtained from a simple stereo technique is adequate [13]. Consequently, all the parameters required for the control are now available directly from the homography decomposition. In summary, the resultant expression for the velocity $\mathbf{v}$ is given as

$$\mathbf{v} = B \left[\begin{matrix} \nu \\ L_{\omega_{xy}}^{-1}[(x - x^*) - L_\nu \nu - L_{\omega_z}\omega_z] \\ \omega_z \end{matrix} \right]$$

$$where \; B = \left[\begin{matrix} -\lambda_v I_{3 \times 3} & 0_{3 \times 2} & 0_{3 \times 1} \\ 0_{2 \times 3} & -\lambda_{\omega_{xy}} I_{2 \times 2} & 0_{2 \times 1} \\ 0_{1 \times 3} & 0_{1 \times 2} & -\lambda_{\omega_z} I_{1 \times 1} \end{matrix} \right].$$
(7)

Equation (7) has only one singularity that occurs at $Z = 0$ (See expression for L_ν). However, the robust homography computation ensures that this degeneracy is always avoided.
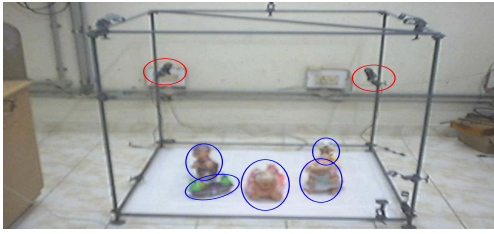
Fig. 5. Experimental Setup with the cameras and the objects used

Fig. 6. Non-Rigid Objects used in the experiments: Three sampled frames depicting the extreme positions during their motion

Thus using the space-time curves described by the points belonging to the object, the relative camera displacement can be reliably computed and used in the above control to achieve stable servoing behavior uninfluenced by the object deformations.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

We present a series of real and simulation results to validate the performance of the proposed technique. Real experiments were conducted to validate the proposed modeling scheme and also to verify the underlying concept behind the proposed servoing strategy *i.e.,* the estimation of projective transformation using space-time curve features and obtaining the relative end-effector displacement; while the performance of the servoing algorithm was studied in simulation. The basic implementation of our proposed algorithm can be summarized into the following steps.

1) In an off-line step, acquire images from the final goal position and extract the curves c_i^* describing the $3D$ point trajectories from these images
2) Acquire a sequence of images from the current camera pose and obtain c_i
3) Estimate homography H_i induced by the space-time curves (Sect. III-A) and compute the reliable homography estimate H_{res} (Sect. III-B)
4) Decompose H_{res} to obtain the motion and structure parameters
5) Using (7), obtain the velocity instruction **v** and move the end-effector to the new pose
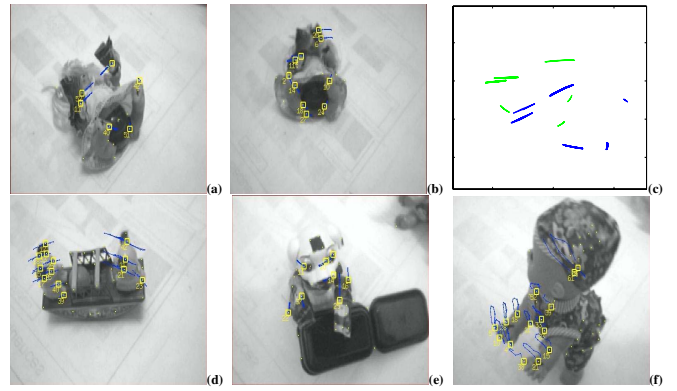6) Repeat steps 2 to 5 until convergence

Fig. 7. Interest points tracked on the objects shown in Fig. 6. Fig.(a) and (b) show the two corresponding views of Object1. Fig.(c) plots the point trajectories obtained in case of (a) and (b) that were used to estimate the homography. Fig.(d),(e) and (f) show one of the views of the Objects 2,3 and 4 respectively, along with the interest point trajectories

Experiments were conducted using an imaging set-up consisting of a $1m$ cubical structure with holders to support CCD cameras (See Fig. 5). The object motion was tracked using the real-time GPU-based tracking system developed in [16]. The different objects used in the experiments are shown in Fig. 6. The objects possess multiple interest points moving with different velocities resembling non-rigid motion. Fig. 7 shows the tracked motion of the points on the object surface. Using these space-time curve features, the projective transformation between the two views was estimated and the relative camera displacement was computed. The computed result was compared to the ground truth obtained from a simple calibration technique. The estimated parameters were very close to the actual values in almost all the cases except in a few, when the considered point trajectory was out of plane. From this experiment, we ascertained that the relative displacement of the camera can be reliably estimated using the space-time curve features of the non-rigid object.

Simulations were conducted in MATLAB environment using a camera with a 512×512 pixel array and a sampling time of $T = 40ms$. Visual servoing was halted when the pixel error reduced below 1 pixel. An arbitrary configuration of points emulating a non-rigid object was considered. The non-rigid motion of the points was simulated using arbitrary planar curves. The image acquired at the initial and the desired camera position is displayed in Fig. 8(a). The servoing task was performed using the algorithm summarized above. Fig. 8(b) shows the smooth convergence of error norm and Fig. 8(d) displays the camera trajectory. At convergence, the camera arrives at the reference pose and the visual features coincide with the desired features. We observe that the control is stable and the translational and rotational velocities (Fig. 8(c)) converge to zero within few seconds. The proposed approach was also tested successfully on multiple initializations of curves. Further, several experiments were performed, using different initializations of the camera configurations, obtaining similar results.

Experiments were also conducted to analyze the performance of the algorithm in presence of noise in calibration
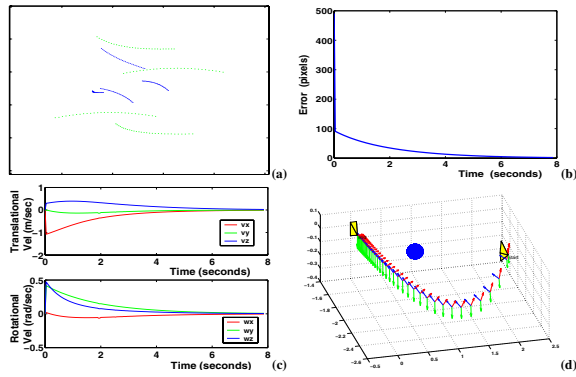
Fig. 8. Simulation Result: (a) Initial (blue) and Desired (green) images (b) Smooth convergence of Error Norm (c) Stable Velocity Screw (d) Camera Trajectory
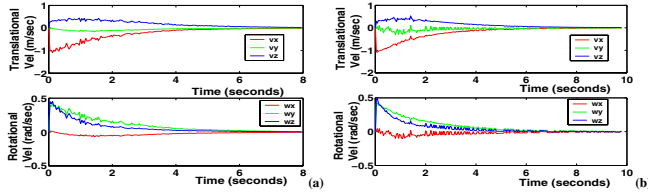

Fig. 9. Results in presence of (a) Calibration and (b) Image Measurement errors for the same case as in Fig. 8

(pose) and image measurements. An additive, zero-mean Gaussian noise with variance $\sigma = 0.1$ was considered. The parameters were varied in turn and average of the error measures was analyzed. From Fig. 9, we observe that the convergence is achieved even in presence of errors demonstrating the robust behavior of the approach.

Finally, an experiment was conducted to analyze the behavior of the robust homography computation algorithm. The degenerate case as shown in Fig. 4 was considered. The displacement that the camera had to realize was approximatively composed of a rotation of $5, -40$ and 0 degrees around camera x, y and z axis respectively and a translation of $5, 0$ and $1cm$ along those axis. Fig. 10(a) shows the variation in weights corresponding to the homographies. Observe that the weight corresponding to degenerate H tends towards the minimum value as the camera approaches the degeneracy. The smooth velocity screw in Fig. 10(b) demonstrates the stable behavior of the algorithm unlike Fig. 4(b).

V. CONCLUSIONS

A new framework for visual servoing has been proposed in this paper that accomplished the positioning task in unknown non-rigid environments. The algorithm utilized multiple homography estimates relating the current and desired camera views in conjunction with the novel non-rigid modeling scheme to accomplish the servoing task. The algorithm can handle most types of non-rigid motions and can tackle large camera displacements without being affected by degenerate configurations. A drawback with our present approach is the requirement of a high fps camera as the sensor has to perceive the target for a minimum of one cycle at every time step so as to acquire the complete space-time trajectory of the
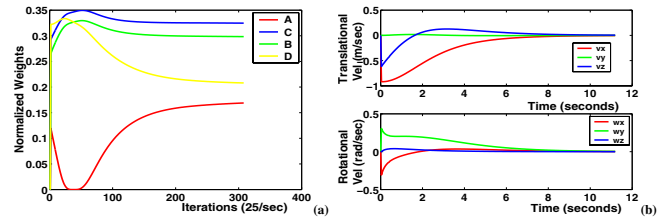

Fig. 10. Robust Homography Computation: (a) Normalized weight values (b) Velocity Screw. The weight corresponding to the degenerate curve reduces to a minimum when the degeneracy is approached, thus leading to a smooth velocity screw (unlike Fig. 4(b))

points. Note that although the points motion is cyclic with respect to a stationary camera; with a moving camera, the motion will not project onto periodic image paths due to the constantly changing camera pose relative to the point motion. However, this limitation can be overcome by using the recent developments in the field of multiple-view geometry of the space-time [9] that attempt to predict the point trajectory at the current pose using information acquired from current image and the past views. Our future work will be devoted to the development of this predictive control.

REFERENCES

[1] A. Agarwal, C. V. Jawahar, and P. J. Narayanan. A survey of planar homography estimation techniques. Technical Report IIIT/TR/2005/12, International Institute of Information Technology, Hyderabad, 2005.
[2] J. K. Aggarwal, Q. Cai, W. Liao, and B. Sabata. Nonrigid motion analysis: articulated and elastic motion. *Computer Vision and Image Understanding*, 70(2):142–156, May 1998.
[3] A. D. Bue, X. Llad, and L. Agapito. Non-rigid metric shape and motion recovery from uncalibrated images using priors. *Computer Vision and Pattern Recognition (CVPR)*, 1:1191–1198, June 2006.
[4] F. Chaumette. Image moments: a general and useful set of features for visual servoing. *IEEE Transactions on Robotics*, 20(4):713–723, August 2004.
[5] K. Deguchi. Optimal motion control for image-based visual servoing by decoupling translation and rotation. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2:705–711, October 1998.
[6] O. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(3):485–508, June 1988.
[7] A. Giachetti and V. Torre. The use of optical flow for the analysis of non-rigid motions. *IJCV*, 18(3):255–279, June 1996.
[8] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.
[9] K. Hayakawa and J. Sato. Multiple view geometry in the space-time. *Asian Conference on Computer Vision*, 2:437–446, January 2006.
[10] S. A. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, October 1996.
[11] P. K. Jain and C. V. Jawahar. Homography estimation from planar contours. *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, June 2006.
[12] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, September 2005.
[13] E. Malis, G. Chesi, and R. Cipolla. 2 1/2 D visual servoing with respect to planar contours having complex and unknown shapes. *International Journal of Robotics Research*, 22(10-11):841–854, October 2003.
[14] D. Santosh Kumar and C. V. Jawahar. Robust homography-based control for camera positioning in piecewise planar environments. *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, 906–918, December 2006.
[15] D. Santosh Kumar and C. V. Jawahar. Visual servoing in presence of non-rigid motion. *International Conference on Pattern Recognition*, 4:655–658, August 2006.
[16] S. N. Sinha, J. M. Frahm, M. Pollefeys, and Y. Genc. GPU-based video feature tracking and matching. Technical Report 06-012, Dept. of Computer Science, University of North Carolina, Chapel Hill, 2006.