Non-Rigid Motion Canceling

D Santosh, C V Jawahar

Center for Visual Information Technology
International Institute of Information Technology
Hyderabad 500032, India
{santosh@students.,jawahar@}iiit.ac.in

Abstract. The objective of our work is to develop vision strategies and algorithms for robotic manipulation tasks in domestic environments where there is ubiquitous presence of non-rigid objects. Non-rigid objects undergo a persistent change in their structure which prevents them from being easily manipulated. However, if the relative motion between the object and the camera can be nullified, then this would simplify the manipulation task. In this paper, we analyze this problem and propose a novel motion canceling algorithm to cancel the non-rigid motion between the object and the camera.

1 Introduction

Conventional robotic vision algorithms [1, 2] have been proposed by envisaging robots operating in industrial environments, where the world is assumed to be static and rigid. These algorithms cannot be used in domestic environments as the assumption of a rigid and static world no longer holds. Our aim is to design servoing algorithms that are robust to non-rigidity and thus can perform optimally even in unconventional environments.

The task of servoing in presence of non-rigid motion can be classified into two primary sub-tasks. The first task involves moving the camera (attached to the robot) from an initial position to a desired position with respect to a non-rigid object. Next, the camera needs to be predictively positioned with respect to the deforming object so that the robot end-effector can easily interact with it. We refer to the former task as *positioning* while the latter as *manipulation*. Recently, Kumar et al. analyzed the positioning task in [3]. By defining a set of invariant features for the non-rigid object, they accomplished the positioning task using conventional servoing algorithms. In this paper, the task of manipulation is studied.

Non-rigid objects [4] undergoes a persistent change in their structure which prevents them from being easily manipulated. To employ robots to interact with such objects, we need to design algorithms that can anticipate the future deformations of the object and position the robot appropriately. Such scenarios are encountered in non-rigid object grasping [5], manipulation [6], motion following etc. However, if the relative motion between the object and the camera can be nullified, then the manipulation task could be easily performed. In fact, in most of the above scenarios, performing the manipulation task essentially boils down to canceling the relative motion between the non-rigid object and the robot which facilitates better interaction between them.

The task of manipulating non-rigid objects has not been analyzed well in the literature. Though some work has been pursued in the area of non-rigid tracking [7, 8], manipulation of non-rigid objects has not been explored. In [6], Smith proposed the use of spatial Jacobian in a three-phased formulation for modeling the non-rigid motion. However in that work, only 'linear' non-rigid objects have been considered. Further, it presumes a affine camera projection model and does not analyze the case of perspective projection. These two assumptions enforce severe restrictions on the domain of scenarios that can be considered by the proposed algorithm. The most related work in this regard was recently attempted by [9]. Cavusoglu et al. proposed a method to handle non-rigidness by canceling the relative motion between a non-rigid object and the camera in a control theory framework (by using a model predictive controller). However, the manipulation task can be efficiently solved using computer vision algorithms as vision is the most reliable sensor. To avoid the intrusiveness of mechanical and magnetic sensors it is desirable to use 'touch free' computer vision-based systems. As a first attempt in this direction, we propose a novel vision-based algorithm to perform the above task.

Our algorithm proceeds by constructing a model of the non-rigid object over time and uses this model to predict the future appearances of the object. Given the current and the future appearance, an image-based visual servoing algorithm [10] is employed to move the camera from its current pose to the desired pose with respect to the object. Any resultant error during this step is corrected by the visual feedback. By repeatedly performing the above steps, the camera is predictively positioned such that it observes the same appearance of the object over time. This effectively cancels the relative motion between the robot and the non-rigid object.

The rest of the paper is organized as follows. In section 2, a brief introduction to non-rigid objects is presented. Section 3 explains the subspace model used to represent the non-rigid object deformations. In section 4, the proposed motion canceling algorithm is described. Experimental results are reported in section 5 and section 6 concludes the paper.

2 Geometry of Non-rigidity

Active non-rigid motion can essentially be classified into three primary types namely articulated motion, elastic motion and fluid motion [4]. 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 [4]. Elastic or cyclic motion is ubiquitous in the natural world. For instance the motion of heart and other body organs, motion of swaying trees, moving aquatic animals etc. In this paper, we concentrate on accomplishing the motion canceling task in presence of such stationary elastic objects. It must be emphasized that global motion from a moving non-rigid object can be separated by performing rigid and non-rigid motion segmentation [11]. In our current work, we consider elastic objects undergoing deformations described by rotation, translation and uniform scaling.

3 Linear Subspace Model

The first stage of the algorithm is to build a model of the non-rigid object. This is a crucial step as a model is needed not only to represent the object deformations but also to infer its internal motion. In literature, several models have been proposed to represent non-rigidness *viz.,* linear subspace models [12], kinematic models [13, 7], finite element methods etc. These models allow the prediction of future states of the object, given the current state.

Interestingly, for the most general class of non-rigid objects, the linear subspace model (or the *manifold* [14]) perfectly describes their object motion. The popularity of the subspace model comes from its simplicity and computational efficiency. Normally, the relationship between the input image and the manifold is nonlinear, but useful results have been obtained using linear mappings between them. Principal Component Analysis (PCA) and Factor Analysis (FA) are two examples of this. Principal component analysis can be obtained as the eigenvectors of the sample covariance matrix associated with the largest eigenvalues. This has proven to be an excellent tool for dimensionality reduction of multivariate data. Here we propose to use PCA for modeling the subspace of appearances of the object. Appearance, in the current context, refers to a configuration of point features on the target extracted from the image (See Fig. 1).

3.1 Building a Manifold

The manifold model was first proposed by Nayar et al. in [14]. Here, we consider a modified formulation of their method. The first step of our method involves computation of the Eigen features from the given image features. Given a set of features extracted from images taken from some camera pose r (where each image represent a deformations of the object), we rearrange them in a vector form $x_1^{'}, x_2^{'}, ..., x_M^{'}$ to get $2N$ dimensional vectors (where N denotes the number of point features and M denotes the number of images). The vectors are normalized and then stacked as columns of a $2N$ x M data matrix $A = [x_1, x_2, ..., x_M]$. The Eigen values of the covariance matrix AA^T are computed and the Eigen vectors (e_i) corresponding to the top k $(\frac{\Sigma_{i=1}^{k}\lambda_i}{\Sigma_{i=1}^{M}\lambda_i} > \Theta)$ Eigen values are chosen as the basis of the linear subspace.

Each feature vector x_i is then projected on to the Eigen-space by first normalizing it and then finding its dot product with each of the eigenvectors (basis) of the subspace. By projecting all the feature vectors, we obtain a set of discrete points in the Eigen-space. Since consecutive images are correlated, their projections in the Eigen space are close to one another. The discrete points $g(p)$ describe a smoothly varying manifold as shown in Fig. 1. Note that the above procedure generates a discrete manifold. To obtain a continuous one, we interpolate between the points using a standard cubic-spline algorithm.

3.2 Projection, Prediction & Reconstruction of Appearance using the Manifold

Given a test image, the first task is to identify its resemblance in the manifold. This is done by extracting the features from the image and then projecting it onto the manifold. Due to image noise, z (the corresponding Eigen feature of the input image) may not lie exactly on the manifold. Therefore, we find a point p' that gives the minimum distance d between the manifold $g(p)$ and z i.e., $d(p') = min||z - g(p')||$. If $d(p')$ is within some threshold, we conclude that the image is of the point p' (See Fig. 1).

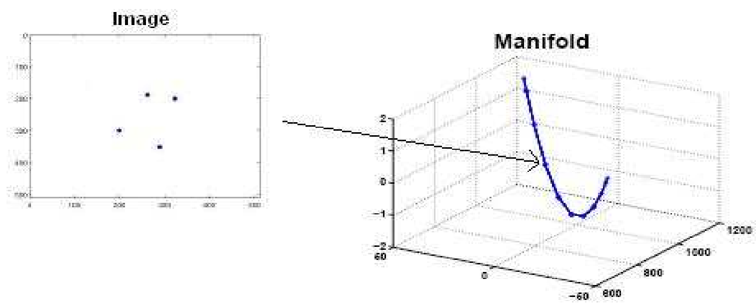

Fig. 1. Projection of image onto the manifold (a) Four points describe the image appearance (b) Manifold constructed for the non-rigid object

Once the point p' corresponding to the test image is determined, we can predict the next appearance of the object using the knowledge of the manifold. As the appearance of the object varies smoothly, consecutive images will be projected close to one another on the manifold. Hence the point adjacent to the projected point will correspond to the next appearance of the object (p_{new}). By using the basis vectors of the Eigen-space, the image corresponding to the predicted point can be reconstructed as

$$I = \sum_{i=1}^{k} p_{new}(k) * e_k. \quad (1)$$

4 Motion Canceling Algorithm

Our aim is to cancel the relative motion between the non-rigid object and the robot end-effector so as to facilitate better manipulation of the object by the robot. We assume that the non-rigid object exhibits some pattern in its deformations and does not change its structure arbitrarily. This is a mild assumption as most of the real world non-rigid objects exhibit some periodicity in their motion. Hence, by observing an object for a finite amount of time, all its deformations can be captured and can be modeled using a manifold.

In order to cancel the relative motion, we need to predictively position the camera such that it observes the same appearance of the object in every frame. At the end of the positioning task [3], the camera reaches the desired pose corresponding to the object appearance that has to be maintained during the manipulation task. The image observed at this camera pose is captured and projected onto the manifold. The next appearance of the object can be known by predicting and reconstructing the image using the knowledge of the manifold. However, in section 3.2, it was assumed that the test image was taken from the same camera pose r from where the sample images were captured to construct the manifold. If the image is taken from a different camera pose, then the Eigen feature z (corresponding to this test image) would not get projected near to its corresponding point on the manifold as the manifold was constructed using images taken from a different camera view. One possible solution to circumvent this problem is to maintain a manifold for every possible view of the camera. However, this turns out to be an impractical solution. Rather an efficient solution is to construct the manifold from a single reference camera pose and then map images from any arbitrary camera pose to this reference pose. Once such a mapping is defined, the transformed image can be projected onto the manifold and the next appearance of the object could be obtained. One possible mapping function is discussed below.

Projection of Image features Given two camera matrices M_1 and M_2 and an image I_1 taken from the pose corresponding to M_1, the task is to obtain the image I_2 as seen from the pose corresponding to M_2. Let $M_1 = K_1[R_1\ t_1]$ and $M_2 = K_2[R_2\ t_2]$ where K_i denotes the intrinsic parameter matrix and $[R_i\ t_i]$ denotes the pose (rotation and translation). Let p_1 represent a point in I_1 and Z_1 be an estimate of the depth of the corresponding $3D$ point P_1 in the camera frame given by a partial $3D$ model. The coordinates of point P_1 are given by $P_1 = K_1^{-1} Z_1 p_1$ [15].

Now to obtain the 3D coordinate of this point in the second camera frame, we have

$$P_2 = [R_2\ t_2][R_1\ t_1]^{-1} P_1. \tag{2}$$

Finally, the image coordinates of the point with respect to the second camera are given by $p_2 = \frac{1}{Z_2} K_2 P_2$.

Hence using this function, we can transform the test image taken from the current camera pose to the reference pose. This transformed image can now be projected onto the manifold and its next image can be predicted and reconstructed. The reconstructed image is then transformed back to the current camera pose using the similar procedure as described above. Thus, we obtain the future appearance of the object given the current image captured from an arbitrary camera pose without explicitly maintaining the manifolds for every view.

Given the current image and the predicted image, we employ a visual servoing algorithm to move the camera from its current position to the desired position from where the camera would observe the same appearance of the object at the next instant. Visual servoing describes a broad class of navigation problems in which a robot is positioned with respect to a target using computer vision as the primary feedback sensor. It is achieved by processing the visual feedback and minimizing an appropriate error function. For a brief review on visual servoing, the reader may refer to [10].

We employ the image based visual servoing [1] algorithm for positioning the camera. We define the error signal e in terms of image feature parameters as $e(S) = S - S^*$, where S and S^* are the current and the predicted image feature parameters respectively. By differentiating this error function with respect to time, we get:

$$\frac{de}{dt} = \frac{dS}{dt} = (\frac{\partial S}{\partial r})\frac{dr}{dt} = L_S V, \tag{3}$$

where $V = (v^T, \omega^T)^T$ is the camera screw, r is the camera pose and L_S is the interaction matrix. It relates the motion of the features in the image space to the camera motion in the Cartesian space. The main objective of the visual servoing process is to minimize the error function $e(S)$. For exponential convergence, we use a proportional control law *i.e.*, $\frac{de(S)}{dt} = -\lambda e(S)$. By substituting this in (3), the required velocity of the camera can be computed as

$$V = -\lambda L_S^+ e(S), \tag{4}$$

where λ is a gain factor that depends on the kinematic constraints of the robot end-effector and L_S^+ is the pseudo-inverse of the interaction matrix. Assuming a perspective projection model with unit focal length, the interaction matrix L_{S_i} for each point feature (u_i, v_i) in the image is given by [10]

$$\begin{bmatrix} -\frac{1}{Z_i} & 0 & \frac{u_i}{Z} & u_i v_i & -(1+{u_i}^2) & v_i \\ 0 & -\frac{1}{Z_i} & \frac{v_i}{Z_i} & 1+{v_i}^2 & -u_i v_i & -u_i \end{bmatrix}, \tag{5}$$

where $i = 1, ..., N$, and Z_i is the depth of the point in the camera coordinate frame. The interaction matrix L_S for the complete set of N points is $L_S = \begin{bmatrix} L_{S_1} \\ \vdots \\ L_{S_N} \end{bmatrix}$, where L_{S_i} is the interaction matrix given by (5).

Note that in case of a moving object, the expression for V will get modified as $V = -\lambda L_S^+ e(S) - L_S^+ \widehat{\frac{\partial e}{\partial t}}$, where $\widehat{\frac{\partial e}{\partial t}}$ represents the object motion model (see [10]). Using this velocity estimate, the camera is moved from its current position to the desired position so as to maintain the same object appearance and in-turn cancel the relative motion between the object and the robot end-effector.

Fig. 2. Oscillatory motion considered in the experiment (a) Three sampled frames depicting the extreme positions during the object motion (b) Features tracked on the object

Thus, by repeatedly performing the above steps, the desired appearance of the object can be maintained over time. Any error that results, either during the prediction of the next appearance of the object or during the positioning of the camera, can be corrected by using the visual feedback.

5 Results and Discussion

A series of experiments were conducted in simulation to test the performance of the algorithm. In the experiments, a set of four $3D$ points on the surface of a non-rigid object was considered. A perspective camera projection model was assumed. The projection of the points onto the image were considered as features. Two kinds of motion were studied, namely oscillatory motion and throbbing motion. It must be emphasized that these motions are very common in real-life scenarios. Examples of oscillatory motion include waving of the hand, swaying of trees etc while the motion of heart and other living organs pertain to throbbing motion.

The basic implementation of the proposed algorithm can be summarized into the following steps.

1. In an offline step, acquire images of the non-rigid object from the reference camera position. Extract features from these images and compute the Eigen-space as described in section 3.1
2. Project each sample image onto the Eigen space and compute the object manifold
3. Take an image from the current camera pose and project it onto the manifold
4. Predict the future image using the knowledge of the manifold
5. Given the current and the predicted images, move the camera using the image-based visual servoing algorithm
6. Repeat steps $3, 4, 5$ until the manipulation task is completed

In case of oscillatory motion, we considered the motion exhibited by point features on the object shown in Fig. 2(a). The feature motion is shown in Fig. 2(b).

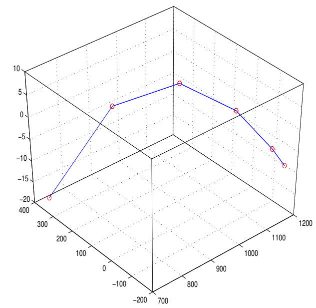

Fig. 3. Manifold for Oscillatory Motion

Fig. 3 shows the manifold obtained in this case. The reference position was chosen such that all the deformations of the object were visible from that camera pose. The first three Eigen vectors were selected as the basis vectors.

Next an image from the reference pose was taken and projected onto the Eigen-space. Its future image was predicted and reconstructed using the manifold. The error between the reconstructed image and the actual next image was computed. This error should be small which signifies that the Eigen-basis capture the important image information that is sufficient to construct the predicted image. In the current case, the error was negligible as the first three Eigen values contained 99% of the information.

5.1 Robustness Issues

We next analyzed the robustness of the proposed approach to camera calibration and depth estimation errors. Recall that these parameters are crucial inputs to the mapping function. A similar experiment as described above was conducted in this case but the camera was located at a different pose $r^{'}$. A test image was taken from this pose and projected onto the manifold. A perfect $3D$ model of the world was assumed to be available. A 10% noise in the camera matrix was introduced. The next image was predicted and the error between the reconstructed image and the actual image was computed. As the error in the camera parameters was increased, the reconstructed image was very poor. This is not only because the current image was projected very far from the manifold (and hence its correct resemblance could not be identified) but also due to the noisy transformations from current camera frame to the reference camera frame.

The above experiment was again repeated to test the robustness of the algorithm to depth estimation errors by introducing noise into the depth values (perfect camera calibration was assumed). Similar statistics were obtained even in this case. For small errors $(10 - 15\%)$ in the depth, the error difference between the reconstructed and the actual next image was small. But as the noise in depth values was increased, the reconstruction was very poor.

Finally, the above experiments were repeated for the throbbing motion. The motion was simulated by using a sinusoidal signal perturbed by a white Gaussian noise of SNR $10dB$. The manifold obtained in this case is shown in Fig. 4. Similar analysis was conducted and the results were studied. The results were very similar in this case also.

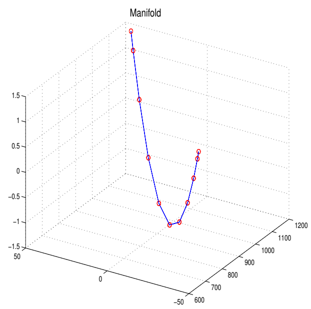

Fig. 4. Manifold for Throbbing Motion

5.2 Manifold Update

If there is a large error between the reconstructed image and the actual next image consistently, then it indicates that an update to the manifold is required. Several approaches can be adopted to perform this step. If the error between the reconstructed image and the actual image is very large, then it is possibly due to a change in the object deformations. In such a case, the manifold needs to be recomputed again using a new set of images. However, to avoid recomputing the whole Eigen-space again, an incremental algorithm can be formulated [16].

6 Conclusion

In this paper, we have proposed a novel non-rigid motion canceling algorithm to cancel the relative motion between a non-rigid object and the robot end-effector. By building a linear subspace model, we maintain a lower dimensional representation of the object deformations and use this model to predict the future appearance of the object. A visual servoing algorithm is employed to move the camera from its current position to the desired position. Results have been analyzed for two kinds of non-rigid motion and the performance of the algorithm in presence of noise in various parameters was studied. We think that this approach can open new research directions in the field of vision-based robot control of non-rigid objects. Many improvements of the proposed methods can be studied. In future, we plan to study the issues involved in updating the manifold. We also plan to analyze the use of the whole appearance of the object rather than only using point features as *appearance* is a better representation of the object.

References

1. Chaumette, F., Espiau, B.: A new approach to visual servoing in robotics. IEEE Transactions on Robotics and Automation **8**(3) (June 1992) 313–327
2. Wang, H., Liu, Y.H., Lam, K.K.: A new approach to visual servoing in uncalibrated environments. IEEE International Conference on Robotics and Biomimetics (August 2004) 578–583
3. Kumar, D.S., Jawahar, C.V.: Visual servoing in presence of non-rigid motion. International Conference on Pattern Recognition **4** (August 2006) 655–658
4. Aggarwal, J.K., Cai, Q., Liao, W., Sabata, B.: Nonrigid motion analysis: articulated and elastic motion. Computer Vision and Image Understanding **70**(2) (May 1998) 142–156
5. Kragic, D., Miller, A., Allen, P.: Real time tracking meets on–line grasp planning. International Conference on Robotics and Automation **3** (May 2001) 2460–2466
6. Smith, P.W.: Image-based manipulation planning for non-rigid objects. International Conference on Robotics and Automation **4** (May 1998) 3540–3545
7. Kanade, T., Morris, D.D., Rehg, J.: Ambiguities in visual tracking of articulated objects using two- and three-dimensional models. International Journal of Robotics Research **22**(6) (June 2003) 393 – 418
8. Laskov, P., Kambhamettu, C.: Tracking non-rigid objects using functional distance metric. Indian Conference on Computer Vision, Graphics and Image Processing (December 2000)
9. Cavusoglu, M.C., Rotella, J., Newman, W.S., Choi, S., Ustin, J., Sastry, S.S.: Control algorithms for active relative motion cancelling for robotic assisted off-pump coronary artery bypass graft surgery. International Conference on Advanced Robotics (July 2005) 431–436
10. Hutchinson, S.A., Hager, G.D., Corke, P.I.: A tutorial on visual servo control. IEEE Transactions on Robotics and Automation **12**(5) (October 1996) 651–670
11. Bue, A.D., Llad, X., Agapito, L.: Non-rigid metric shape and motion recovery from uncalibrated images using priors. IEEE Conference on Computer Vision and Pattern Recognition **1** (June 2006) 1191–1198
12. Shashua, A., Levin, A., Avidan, S.: Manifold pursuit: A new approach to appearance based recognition. International Conference of Pattern Recognition **3** (August 2002) 590–594
13. Chaumette, F., Marchand, E., Comport, A.: Object-based visual 3d tracking of articulated objects via kinematic sets. IEEE Workshop on Articulated and Non-Rigid Motion, CVPRW **1** (June 2004) 2–9
14. Nayar, S., Murase, H.: Visual learning and recognition of 3d objects from appearance. International Journal of Computer Vision **14**(1) (January 1995) 5–24
15. Zisserman, A., Hartley, R.: Multiple view geometry in computer vision. Cambridge University Press (2003)
16. Artac, M., Jogan, M., Leonardis, A.: Incremental pca or on-line visual learning and recognition. International Conference on Pattern Recognition **3** (August 2002) 781–784