

Experimental Evaluation of Uncalibrated Visual Servoing for Precision Manipulation*

Martin Jägersand, Olac Fuentes, Randal Nelson

Department of Computer Science, University of Rochester, Rochester, NY 14627

{jag,fuentes,nelson}@cs.rochester.edu

http://www.cs.rochester.edu/u/{jag,fuentes,nelson}/

Abstract

In this paper we present an experimental evaluation of adaptive and non-adaptive visual servoing in 3, 6 and 12 degrees of freedom (DOF), comparing it to traditional joint feedback control. While the purpose of experiments in most other work has been to show that the particular algorithm presented indeed also works in practice, we do not focus on the algorithm, but rather on properties important to visual servoing in general. Our main results are: positioning of a 6 axis PUMA 762 arm is up to 5 times more precise under visual control, than under joint control. Positioning of a Utah/MIT dextrous hand is better under visual control than under joint control by a factor of 2. We also found that a trust-region-based adaptive visual feedback controller is very robust. For m tracked visual features the algorithm can successfully estimate online the $m \times 3$ ($m \geq 3$) image Jacobian (J) without any prior information, while carrying out a 3 DOF manipulation task. For 6 and higher DOF manipulation, a rough initial estimate of J is beneficial. We also verified that redundant visual information is valuable. Errors due to imprecise tracking and goal specification were reduced as the number of visual features, m , was increased. Furthermore highly redundant systems allow us to detect outliers in the feature vector, and deal with partial occlusion.

1 Introduction

Using visual input to control robot manipulators allows more flexible and robust robot behaviors than traditional position based control. Recently, “uncalibrated” visual servoing control has been demonstrated in a variety of settings (e.g. [1, 3, 2, 5, 6, 7, 11, 14, 13, 18, 19, 21]¹). Visual models suitable for specifying simple visual alignments have also been studied [22, 9, 8]. In [15] we show how to derive the robust visual servoing controllers used in this paper. In [13, 14, 11] we have shown how different visual servoing behaviors can be combined to solve complex real-world tasks in unstructured environments. Surprisingly, a thorough experimental evaluation of the accuracy-related properties of visual feedback control, to our knowledge, has never been performed. Experimental results in most papers have been limited to a few runs, only to validate that a particular algorithm indeed works in practice. The closest general experimental evaluations we have found are the work by Wijesoma *et al.* [6] and Chen *et al.* [17]. Wijesoma *et al.* showed for

a 2 DOF implementation, the advantage of visual feedback over open loop control when model errors are large. Chen *et al.* show that two heavy industrial robot arms can perform a peg-in-hole parts mating task with a clearance of 0.7mm between the peg and the hole using camera space control.

In this paper we present an evaluation of visual servoing, based on numerous real positioning experiments, giving the following four main results:

1. For two typical robot arms (PUMA 761 and 762) repeatability is up to five times better under visual servo control than under traditional joint control.
2. The adaptive visual servoing controller is very robust. The algorithm can successfully estimate the $m \times 3$, (m = number of visual measurements) image Jacobian without any prior information, while carrying out a 3 DOF manipulation task.
3. We were able to verify that redundant visual information is valuable. Both errors due to imprecise tracking and goal specification were reduced as m was increased. Furthermore highly redundant systems allow us to detect outliers in the visual feature measurement vector y , and deal with partial occlusion.
4. Uncalibrated visual servoing can be made to work on difficult to control manipulators. We show results on fine manipulation using the 16 DOF Utah/MIT hand.

Most of these results are not limited to only a specific algorithm, but apply to broader classes of algorithms (see section 4). The overall structure of our vision based manipulation system is shown in fig. 1. In the two following sections we review the online model acquisition. The control is described in section 2.2. Visual goal assignment, trajectory generation and what kind of visual features to use is not dealt with in this paper. We refer to [13] where we describe the high (task) level parts of the system and how to solve real world manipulation problems with this system, such as solving a shape sorting puzzle, handling flexible materials and exchanging a light bulb. The main contribution of this paper are the experiments in section 3 where we study low-level aspects of visual control, and use simple random trajectories to quantitatively evaluate positioning and model estimation.

2 Theory

This section reviews the active visual model acquisition and control used in these experiments. We have published a detailed description in [11, 15].

An active vision agent has control over its actions, and can watch the results of an action by observing the changes

*Support was provided by NSF grant CDA-9401142, ONR grant N00014-93-0671 and U of Maryland/ARPA subcontract Z840902.

¹For a review of this work we direct the reader to [14] or [18].

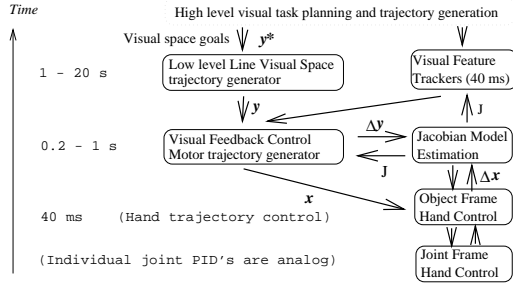


Figure 1: Structure of the system, and the time frames each part runs in. Arrows indicate information exchange.

in visual appearance. We study a robot agent, fig. 2, in an unstructured environment. The robot action reference frame is joint space, described as desired joint angles², $\mathbf{x} = (x_1 \dots x_n)^T$, and their time derivatives $\dot{\mathbf{x}}$, $\ddot{\mathbf{x}}$. The changes in visual appearance are recorded in a *perception* or *feature* vector $\mathbf{y} = (y_1 \dots y_m)^T$. Visual features can be drawn from a large class of visual measurements [1, 11], but we have found that the ones which can be represented as points or point vectors in camera space are suitable, since they yield smooth transfer functions f [14]. We track features such as boundary discontinuities (lines, corners) and surface markings. Redundant visual perceptions ($m \gg n$) are desirable as they are used to constrain the raw visual sensory information.

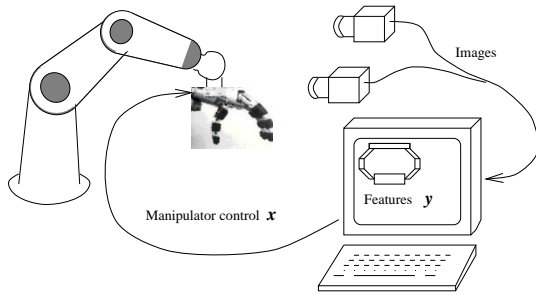


Figure 2: Visual control setup using two cameras.

The visual features and the agent's actions are related by an initially unknown, non-linear visual-motor function f , satisfying $\mathbf{y} = f(\mathbf{x})$. At any time k we want to estimate a first order model $f(\mathbf{x}) \approx f(\mathbf{x}_k) + J(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$, valid around the current system configuration \mathbf{x}_k , and described by the "image" [18] or visual-motor Jacobian defined as

$$(J_{j,i})(\mathbf{x}_k) = \frac{\partial f_j(\mathbf{x}_k)}{\partial x_i} \quad (1)$$

The image Jacobian not only relates visual changes to motor changes, as has been previously exploited in visual feedback control [2], but highly constrains the possible visual changes to the set of possible solutions \mathbf{y}_{k+1} of $\mathbf{y}_{k+1} = J\Delta\mathbf{x} + \mathbf{y}_k$. Thus the Jacobian J is also a visual model, parameterized in exactly the degrees of freedom our robot can move in.

²Vectors written bold, scalars plain and matrices capitalized.

(Remember we have $m \gg n$ so the solution set is only a small subspace of \mathfrak{R}^m .)

2.1 Model estimation

In most visual servoing work a Jacobian has been either (1) derived analytically, (2) derived partially analytically and partially estimated (eg. [2, 7]), or (3) determined experimentally by physically executing a set of orthogonal calibration movements \mathbf{e}_i (eg. [9, 22]) and approximating the Jacobian with finite differences:

$$\hat{J}(\mathbf{x}, \mathbf{d}) = (f(\mathbf{x} + d_1 \mathbf{e}_1) - f(\mathbf{x}), \dots, f(\mathbf{x} + d_n \mathbf{e}_n) - f(\mathbf{x})) D^{-1} \quad (2)$$

where $D = \text{diag}(\mathbf{d})$, $\mathbf{d} \in \mathfrak{R}^n$.

We instead use an on-line method, which estimates the Jacobian by just observing the process, without a-priori models or introducing any extra "calibration" movements. In observing the object movements we obtain the changes in visual appearance $\Delta\mathbf{y}_{measured}$ corresponding to a particular controller command $\Delta\mathbf{x}$. This is a secant approximation of the derivative of f along the direction $\Delta\mathbf{x}$. We want to update the Jacobian in such a way as to satisfy the most recent observation (secant condition): $\Delta\mathbf{y}_{measured} = \hat{J}_{k+1}\Delta\mathbf{x}$. The above condition is under-determined, and a family of *Broyden* updating formulas can be defined [26, 15]. In previous work we have had best results with this asymmetric correction formula [11, 15, 27]:

$$\hat{J}_{k+1} = \hat{J}_k + \frac{(\Delta\mathbf{y}_{measured} - \hat{J}_k\Delta\mathbf{x})\Delta\mathbf{x}^T}{\Delta\mathbf{x}^T\Delta\mathbf{x}} \quad (3)$$

This is a rank 1 updating formula in the *Broyden* hierarchy. For a set of orthogonal movements about a point $\{\Delta\mathbf{x}_k, k = 1 \dots n\}$ eq. 3 is identical to the finite difference update (eq. 2) in a rotated coordinate frame. Note however that this estimation accepts movements along arbitrary directions $\Delta\mathbf{x}$ and thus needs no additional data other than what is available as a part of the manipulation task we want to solve. During reaching (long transportation) movements this keeps the estimate \hat{J}_{k+1} accurately updated in the direction of movement, and reasonably approximated (due to process noise) along the other directions. During fine manipulation (many small independent movements in a small region) \hat{J}_{k+1} eventually approximates the true Jacobian J accurately in all directions. The initial Jacobian J_0 can be a random matrix, but for fine manipulation in high DOF systems (4 DOF and higher) a rough initial estimate is beneficial to avoid initially erroneous movements. A high DOF (e.g. 6 DOF) partial initial estimate can be obtained from a lower (e.g. 3 DOF) Jacobian. See our visual space task programming paper [13] on how to naturally integrate this into manipulation tasks.

2.2 Control

The active robot agent specifies its actions in terms of desired perceptions \mathbf{y}^* . We need a control system capable of turning these goal perceptions into motor actions \mathbf{x} . While previous work in visual servoing have mostly used proportional control, typically with slight modifications to account for some dynamics, we have found that a more sophisticated approach is beneficial. We use two ideas from optimization: (1) A trust region method [27] estimates the current model validity online, and controller response is restricted to be inside this "region of trust". (2) A homotopy or path following method [25] is used to divide a potentially non-convex

problem into several smaller convex problems by creating subgoals along trajectories planned in visual space. For details on the latter see [13], in which we describe a method for high level visual space task specification, planning, and trajectory generation. The trust region method adjusts a parameter α so that the controller never moves out of the validity region of the current Jacobian estimate. To do that we solve a constrained problem for $\|\delta_k\| < \alpha$ instead of taking the whole Newton step of a proportional controller.

$$\min_{\|\delta_k\| < \alpha_k} \|\mathbf{y}_k - \mathbf{y}^* + \hat{J}_k \delta_k\|^2 \quad (4)$$

Define a model agreement as $d_k = \frac{\|\hat{J}\delta\|}{\|\Delta\mathbf{y}_{measured}\|}$ and adjust α according to (d_{lower} and d_{upper} are predefined bounds):

$$\alpha_{k+1} = \begin{cases} \frac{1}{2}\alpha_k & \text{if } d_k < d_{lower} \\ \alpha_k & \text{if } d_{lower} < d_k \leq d_{upper} \\ \max(2\|\delta\|, \alpha) & \text{if } d_k > d_{upper} \end{cases} \quad (5)$$

Video frame rate and image processing overhead limit the maximum visual feedback bandwidth in a visually controlled robot. Feddema points out that standard frame rate (30 Hz) is too low for direct joint motor feedback, and suggests that this can be overcome by also using joint feedback [2]. We use a two level approach. The motor commands (δ_k in eq. 4 above) are used to generate a desired joint space trajectory, where the low level joint control is implemented using RCCL. We achieve smooth *visual servoing* movement, rather than a *look and move* (in Weiss terminology [1]), by queuing the next trajectory segment before the current is finished, see fig. 3. Note that \hat{J}_k in eq. 4 is updated by eq. 3 in each step. This serves to synchronize the model acquisition with the control, and causes the model to be estimated on an adaptive size mesh; dense when the visual-motor mapping is difficult, sparse when it is near linear. Similarly the visual cycle frequency varies, lightening the computational load whenever possible.

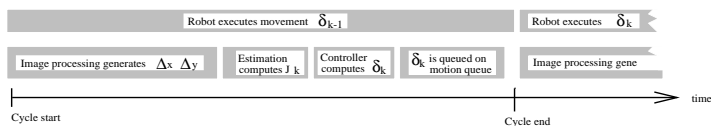


Figure 3: Time frames in a visual cycle.

3 Experiments with visual servoing

In this section we present the main contribution of this paper: an experimental evaluation of visual servoing on several real manipulators (PUMA 761 and 762 6 DOF robot arms, and Utah/MIT 16 DOF dextrous hand). In choosing the experiments we had the following objectives: We wanted to show how visual servoing performs by collecting statistics over numerous repetitions of positioning tasks. In these experiments trajectories have been selected uniformly randomly sampled in the manipulator workspace. Most other published results are from just one or a few repetitions. In [13] we have shown how visual servoing can be integrated into a system solving several real world visual manipulation tasks (6 DOF insertions, solving a child's shape sorter puz-

Robot	Measured (mm)			Spec. (mm)
	Visual feedback	Perf. vis. alignment	Joint feedb.	Joint feedback
Bill	0.13	0.079	0.64	0.2
Hillary	0.059	0.039	0.080	0.2

Table 1: Measured and specified repeatability of our Unimation PUMA 762 and 761 robots under visual and joint feedback control.

zle, exchanging a light bulb). During those experiments we found that the limits of positioning accuracy were caused by the difficulty of accurately generating and tracking visual goals, not the visual servoing sub-system's ability to achieve the goals. In this paper we study the performance of the visual servoing alone. Specification errors are minimized by measuring repeatability rather than accuracy³, and tracking errors are reduced by tracking special markers or LED's mounted to the test objects.

The positioning repeatability of visual feedback is evaluated, and compared to traditional joint feedback positioning in a set of repeatability experiments for robot arms in section 3.1 Section 3.2 presents results on how positioning accuracy is dependent on the number and quality of visual feature measurements. Controller convergence is related to the accuracy of the controller parameters. In section 3.3 convergence of adaptive and non-adaptive controllers are experimentally compared. Repeatability of the dextrous hand is evaluated in section 3.5. For an in depth treatment of the experimental conditions and results we refer to our technical report [11].

3.1 Repeatability

We tested repeatability under closed loop visual control and compared the results to traditional joint control, both with our own experiments and with published figures.

Positions were measured through a 0.001" accuracy dial meter mounted on the robot end effector. Accurate visual measurements were provided by LED's mounted on the robot end effector. Two cameras were positioned approximately 60 degrees apart and 1 pixel \approx 0.25mm robot movement near the goal. Repeatability was measured by moving the robot on a random trajectory towards a rigid reference surface until the dial meter was in contact, reading the dial value and visual or joint values, then retreating on a second random trajectory, and then trying to reach the visual or joint goal. The algorithm is specially designed to execute the final approach from a different, random direction each time.

We tested the two robot arms in our lab. Bill is our old PUMA 762, and has during the years developed a significant backlash in the gears, as well as some sticktion. Hillary is a PUMA 761 and a more recent acquisition, in better shape. Table 1 shows our results for visual feedback repeatability. The results indicate convergence to subpixel accuracy. We think "vernier accuracy" effects from the high dimensional ($m = 16$) input space can account for this. Compared to joint control repeatability measured under the same condi-

³Repeatability is the ability of the robot to reach a previously attained pose. Accuracy measures the ability to achieve any prespecified pose [23].

tions, visual space repeatability is about 5 times better for Bill. Results for Hillary show that for a robot in good mechanical shape, visual control is at least as good as joint control.

The perfect visual alignment shows the repeatability at 0 visual error, i.e. when in camera space the tracked points are perfectly aligned over the goal points. This is not achieved at all trials. For Bill, for instance, 22 of 50 trials resulted in 0 visual error. We believe the main reason why not all trials achieve 0 visual error is the difficulty in positioning the manipulator, near or below its joint encoder resolution limit (about 0.12 mm here). Very close to the goal, noise in the controller causes it to drive the motors randomly back and forth. This samples a lot of possible positions below the resolution limit, and by stopping if we hit 0 pixel visual error, we can sometimes achieve positioning below the resolution limit.

The distribution of positioning errors for Bill can be seen in fig. 4. The two modes, around 0.25mm and 1mm, in the open loop distribution are most likely due to the different backlash errors introduced by driving the first joint (j1) in one of two directions during a measurement. Note that both the bimodality and the variance are substantially reduced under visual control.

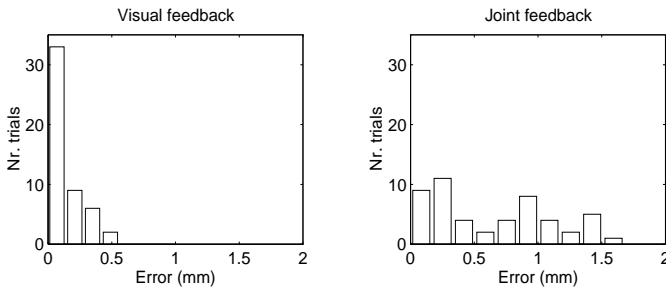


Figure 4: Distribution of positioning errors. **Left:** closed loop visual feedback. **Right:** joint feedback.

3.2 Effects of over-determined input spaces

In this experiment we studied how accuracy is affected by adding redundant visual information, making the system more or less over determined. In 3D computer vision the value of redundant information has long been recognized. Viewing geometry models, and full or partial camera calibration are used to relate redundant feature measurements to improve accuracy in pose estimation, e.g. see [10]. However, our vision system does not make any viewing geometry assumptions, and our cameras are uncalibrated. But as explained in the theory section, the visual-motor manipulation model gives a new way of constraining the DOF's of the visual feature vector to the smaller robot action space.

We found two main advantages of a redundant feature vector. The first is that errors due to inaccuracies in either visual feature tracking or goal generation can be lessened. The second main advantage is that in the event of one or more of the features becoming occluded, or giving a badly conditioned measure, we can rely on the others to solve the task. We can even add more features dynamically, first letting the system identify their visual-motor model, without using them for control, then putting them into the control loop Jacobian. In the best case, if we add m signals with zero mean and σ^2 variance noise (admittedly not typically a

valid assumption in a vision system), the variance, and thus the noise of the final signal, decreases as σ^2/m . However, even if this model of the signal does not hold, as long as at least some of the visual errors are uncorrelated we will get an improvement from using redundant visual features.

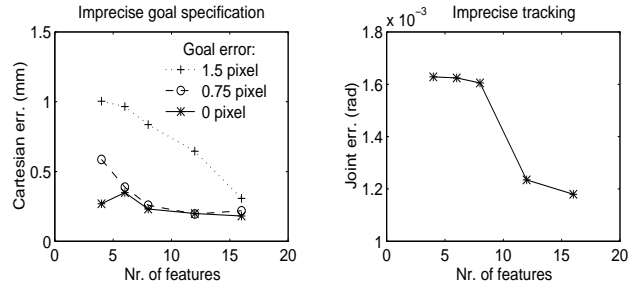


Figure 5: Left: 250 runs varying error in the visual goal specification and the number of visual features used show that, particularly for imprecise visual specifications, having redundant features helps. Right: A similar experiment, but with errors from imprecise tracking, caused placing the cameras very far away, shows a similar result.

Figure 5 shows the effects of increasing the number of tracked points. In the first experiment we placed the cameras 0.5m from the robot, allowing accurate feature tracking. We introduced varying amounts of error in the visual goal specification by adding a vector in a random direction, with a magnitude uniformly distributed in the range 0.0 to 1.5, and 0.0 to 3.0 pixels (expected value 0.75 and 1.5 pixels) to each of the visual goals. Final positioning accuracy was measured with the dial meter as in the repeatability experiment. The results indicate that for inaccurate visual goals, positioning improves significantly when using redundant visual information.

In the second experiment we placed the cameras 6m from the robot, thus decreasing the accuracy in tracking. Here it is more practical to measure errors as differences in joint encoder values. We observe improvement in joint space end point accuracy when increasing the number of features.

3.3 3 and 6 DOF Adaptive v.s. Non-adaptive visual control

In this experiment we compare performance of adaptive and non-adaptive visual controllers in 3 DOF and 6 DOF. The adaptive controller adjusts its internal visual-motor model $\{\hat{J}_k, \alpha_k\}$ to environment and configuration changes over time (k) and other non-linearities in the visual motor function f using the method in section 2.1. The non-adaptive one has a fixed model $\{\hat{J}, \alpha\}$. The accuracy of the internal model affects the controller performance.

At first it may seem that the most relevant evaluation measure for an adaptive controller is how accurately it can estimate the visual-motor model. However, accurate model estimation has to be traded off for efficiency, and ideally one would do this so the internal model (\hat{J}) gets estimated only as precisely as necessary to do the task at hand. Tuning of this accuracy is accomplished by varying d_{lower} and d_{upper} in eq. 5. For a 3 DOF task we have found that estimating the derivatives (elements $j_{i,j}$ in the Jacobian) with 50% – 80% accuracy is sufficient. This finding is in line with Hosoda and Asada's observations for their visual servoing controller

in [19].

Instead of studying internal aspects of the adaptive and non-adaptive controllers, we wanted to compare how robustly the two controllers react to a suddenly introduced discrepancy between the internal model and the real world. In real world applications this discrepancy can stem from a variety of sources such as errors in initial model estimate, errors in visual data used to adjust the model, and for the non-adaptive case, poor correspondence between model (linearized) and real world (typically. non-linear).

The experimental procedure is as follows: The goal for the robot is to move along a preselected random trajectory to a visual goal \mathbf{y}^* . In the middle of the movement the Jacobian is disturbed in one updating cycle by adding a random matrix: $\hat{J}_{dist} = a * \hat{J} + (1 - a) * J_{random}$ where $a \in [0, 1]$ (model accuracy) changes the magnitude of the disturbance and J_{random} is a random $m \times n$ matrix with elements drawn from a uniform distribution on the interval $[-|2\hat{J}_{i,j}|, |2\hat{J}_{i,j}|]$.

During the experiment convergence, trajectory error, and final endpoint positioning error were studied for the adaptive and non-adaptive cases. (Trajectory error is the average deviation in visual space from the planned trajectory between the initial feature values \mathbf{y}_0 and the goal \mathbf{y}^* .) Each test consisted of 50 runs with varying trajectories and values of a .

Fig. 6 plots convergence percentage⁴ (left) and positioning error (right) while varying the disturbance magnitude a . For the non-adaptive algorithm, the percentage of the trials which converged quickly falls to zero for model accuracies a below 0.5. The adaptive algorithm had no problem reaching the goal, even when $a = 1$. This means it can recover from a completely random internal model. The trajectory error for the adaptive algorithm increased slightly when the disturbance is large. This increase comes from a few (order of $n = \text{motor DOF}$) erroneous movements just after the disturbance, before the model is accurately re-estimated.

For the non-adaptive algorithm, we found a huge increase in trajectory error for the disturbed trials. This is caused by divergent trials, where the robot goes off in a completely incorrect direction and long oscillatory behavior at the transition between convergence and divergence, which occurs near $a = 0.25$. An illustration of a typical oscillatory move is shown in fig. 7.

When controlling the full 6 DOF of the manipulator we do not get quite the same robustness against model errors as in the 3 DOF case. Disturbances worse than $a = 0.6$ yield a significant and increasing proportion of divergent trials. To get near 100% convergence we needed to restrict model accuracy a to be .6 or above. In a test of 50 trials in this range, two did not converge. One showed oscillatory behavior, and the other diverged immediately. In general, average visual endpoint errors as well as trajectory errors remain low even at low model accuracies, as can be seen in fig. 6.

3.4 12 DOF control of a non-rigid foam beam

High DOF control problems involving manipulation of non rigid objects are very hard to solve with traditional model based robot control paradigms. The modeling of the problem can be messy or completely impractical, which makes design of a control system difficult. Our adaptive controller, on the other hand, does not need an exact a-priori

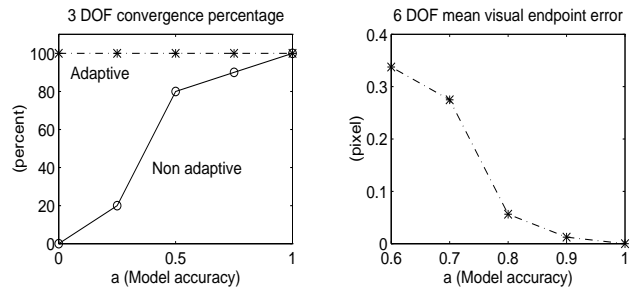


Figure 6: Results from 50 runs of an experiment with a disturbed internal model in the controller. **Left:** Convergence for an adaptive and a non adaptive 3 DOF controller. **Right:** Average final endpoint error per feature in \mathbb{R}^{16} feature space (right) for a 16 tracked features and 6 controlled DOF problem (right)

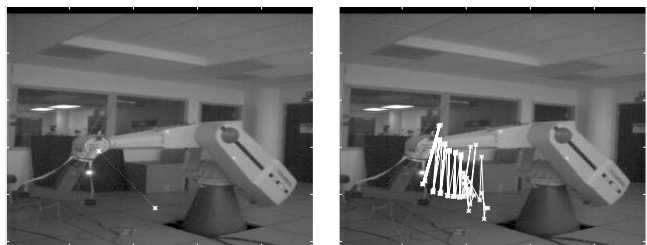


Figure 7: **Left:** Visually planned path is a straight line between the white stars plotted on the image. **Right:** Typical oscillatory path taken by the non-adaptive controller when model accuracy is near the convergence limit.

model. Instead the robot learns and refines successive linear models of the visual-motor function while performing its task in the real environment.

This experiment used two 6 DOF PUMA manipulators, one attached to each end of a piece of flexible packing material foam. The attachment is rigid, so the arms can exert torques as well as forces to the beam. The object of the system was to bend or fold the beam into a specified shape. The manipulation is specified by showing the system a sequence of training images of the desired manipulation. From each image in the teaching sequence a feature vector \mathbf{y}_k^* ; $k = 1..q$ is extracted, representing the shape of the non-rigid foam beam. \mathbf{y}_q^* is the final goal shape, and \mathbf{y}_k^* ; $k = 1..q - 1$ are intermediate goals, or *way points*, defining the desired trajectory in visual space.

The Oxford snake trackers [4] we usually use for tracking rely on an affine constraint. We could not get them to track the foam edges through non-rigid deformations. Instead a point representation of the foam outline is tracked using special purpose trackers on the markers seen in fig. 8. From a control point of view this is a hard problem. To start with, the visual-motor function is highly nonlinear. We also start from a nearly singular state (nearly straight beam). It turned out that we needed to show a sequence of at least 3 images in order to generate a convergent (sub)goal way point sequence \mathbf{y}_k . As seen in the m-peg video⁵, the motion of the two

⁴For convergence we require ($\|\mathbf{y}_{final} - \mathbf{y}^*\| \leq 2\text{pixel}$).

⁵Electronic m-peg videos of the experiments in this section

robots is quite wiggly. This is a result of the problem being difficult (ill conditioned and highly non-linear). Despite this accurate final point positioning was achieved. The robots were able to position the foam outline within 1 pixel of the specified shape.

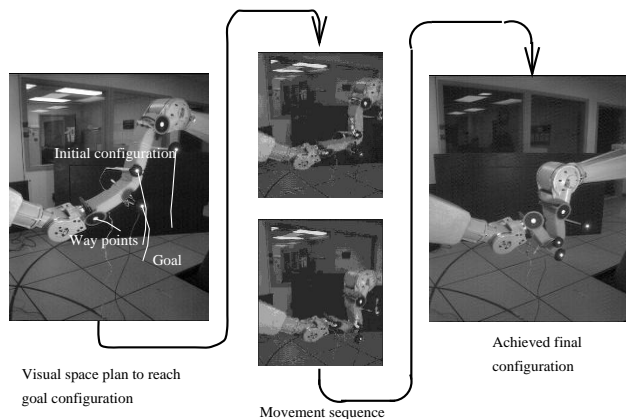


Figure 8: The carrying out of the foam folding task

3.5 Review of Utah/MIT dextrous hand experiments

Dextrous high DOF robotic hands provide versatile motions for fine manipulation of potentially very different objects (fig. 9). However, fine manipulation of an object grasped by a multifinger hand is much more complex than if the object was rigidly attached to a robot arm. Creating an accurate manipulation model is difficult if not impossible due to the many possible objects that could be grasped, and the inherent manipulator inaccuracies.

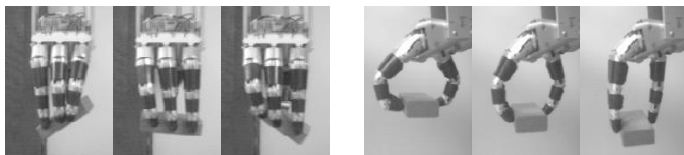


Figure 9: Examples of manipulation with Utah/MIT hand.

In [16] we augment the visual model estimation and control with fine manipulation capabilities for the Utah/MIT hand. This is a review of the experiments in that paper. We tested repeatability under closed loop visual control and compared the results to traditional joint control. Visual feedback consist of $m = 16$ feature values from 8 markers, and 2 cameras, see fig. 10. The markers could be placed arbitrarily on the object, since the servoing algorithm is self calibrating. Placement near the corners gives better conditioned visual measurements with respect to rotations. Two cameras were positioned approximately 90 degrees apart. 1 pixel corresponds approximately to a 0.3mm object movement. In fig. 10 let x be the optic ray, y the image row and z the column. Around the x , y and z axis respectively, a grasped object can be translated about 40mm, 60mm and

30mm, and rotate 70, 45 and 90 degrees (see fig. 9). We tested but did not find any correlation between trajectory length and positioning error.

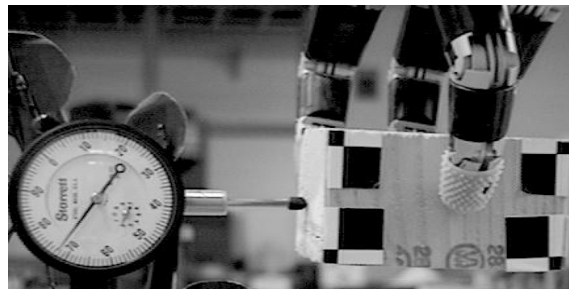


Figure 10: Setup for hand fine manipulation repeatability experiments

In fig. 11 we compare repeatability for three positioning methods, based on 50 trials with each method. The 3 DOF visual feedback controls only translations of the object. The 4 DOF visual feedback controls the translations and rotation around the optical axis of fig. 10. In 6 DOF we had some problems with singularities and used a slightly modified tracking discussed below. The two joint feedback methods differ in that one only uses joint feedback, while in the Cartesian one, the error vector is projected onto the 6DOF space defined by the grasp tetrahedron[20] (The tetrahedron formed by the grasp points of the four fingers). Visual feedback gets more difficult in high DOF systems. One reason is that the number of parameters estimated in the Jacobian increases. Another is that high DOF tasks are often more ill-conditioned. Particularly, there is a difference between pure translations (3 DOF) and combinations of translations and rotations (4-6 DOF). The high condition numbers in the 5 and 6 DOF cases stem from two distinct problems. With the two camera setup we use, the rotation around the z -axis is (visually) ill conditioned (differentially similar to the translation along the optic axis.) This problem occurs with a two camera setup, when the points tracked in each camera are co-planar. The other problem is that two of the manipulations are differentially very similar in Cartesian motor space. By taking out the two problematic DOF's we get the low condition numbers of the 3 and 4 DOF manipulations.

In order to evaluate visual servoing in the full 6 DOF pose space we improved the visual measurements by tracking LED's mounted on adjustable screws attached to the test objects. The LED's facilitate more precise tracking than the surface markings, and using the screws we adjusted them into a non-coplanar configuration, avoiding the visual singularity mentioned above. With this setup we were able to do 6 DOF visual servoing.

In the visual servoing behavior we noted that trials fell into three categories. About half the trials converged very fast. Most of the rest converged, initially fast, but much slower towards the end. Last, a small percentage (about 2%) did not converge at all. We have not observed this difference in convergence speed when using the same visual feedback controller on robot arms. We think a partial explanation may be that mechanical phenomena in the hand and remotizer linkage cause some positions to be harder to reach than others. We have also observed that when the visual servo controller is turned off, the hand often does not remain in the same position, but drifts significantly, despite the joint

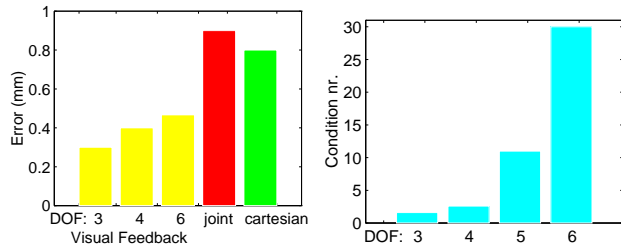


Figure 11: **Right:** Repeatability using visual or joint sensor feedback. **Left:** Condition numbers κ of different DOF Jacobians.

setpoints remaining the same.

4 Discussion

The estimation and control algorithms we developed have strong theoretical properties (see [14, 15]). It is still very important to evaluate experimentally how they perform in real environments, with real process disturbances/noise and manipulators with real mechanically caused errors and finite precision.

Our results show that visual feedback control yields significant repeatability improvement in an imprecise PUMA robot arm, while only marginal improvement in the Utah/MIT hand. This is explained by the different characteristics of the manipulators. The inaccuracies in positioning the PUMA arm stem mainly from backlash, which the visual feedback can correct. In the hand a combination of sticktion and flexibility makes it much harder to control accurately. When trying to make small movements, the hand will initially not move at all, and the controller ramps up the signal. When the hand finally unsticks it grossly overshoots the intended goal. This makes it very hard to control with a feedback controller, and even harder to estimate the visual motor Jacobian. To estimate the Jacobian, large scale movements, where the relative effects of this sticktion is small need to be added to fine manipulation tasks.

While non-adaptive visual servoing methods have been shown to converge in simple settings (eg. stationary cameras, world coordinate robot control [3, 9]), we have found that in more complex cases the adaptiveness is crucial. Eye-in-hand type manipulation for instance does not work well without on-line Jacobian estimation [13]. The Utah/MIT hand control we presented, as well as control of large, complex object rotations we have shown earlier [12], needed the trust region controller, and we could not make the 12 DOF nonrigid control in section 3.4 work without the way points, and the homotopy method. In addition the visual space way point generation is the natural way of doing trajectory planning in a hand-eye system [13].

We believe that the results we show are of general value in that the repeatability for a visual feedback method is insensitive to the exact controller parameters. As long as they converge, different controllers can be expected to converge with about the same repeatability, although the time, and trajectory taken may vary, and whether they converge or not depends much on the controller. The convergence results in section 3.3 should be similar among non-adaptive methods, and also valuable for adaptive methods using other Jacobian estimation schemes (eg. [19]), or (partially) model based schemas (eg. [2]), as long as the estimate accuracy is similar.

References

- [1] Weiss L. E. Sanderson A. C. Neumann C. P. "Dynamic Sensor-Based Control of Robots with Visual Feedback" *J. of Robotics and Aut.* v. RA-3 1987
- [2] Feddema J. T. Lee G. C. S. "Adaptive Image Feature Prediction and Control for Visual Tracking with a Hand-Eye Coordinated Camera" *IEEE Tr. on Systems, Man and Cyber.*, v 20, no 5 1990
- [3] Conkie A. Chongstitvatana P. "An Uncalibrated Stereo Visual Servo System" DAITR#475, Edinburgh 1990
- [4] Curwen R. Blake A. "Dynamic Contours: Real time active splines" *Active Vision* Blake, Yuille MIT Press 1992.
- [5] Espiau B. Chaumette F. Rives P. "A New Approach to Visual Servoing in Robotics" *IEEE Tr. on Robotics and Automation* p 313-326 v 8 no 3, 1992.
- [6] Wijesoma S. W. Wolfe D. F. H. Richards R. J. "Eye-to-Hand Coordination for vision guided Robot Control Applications" *Int. J. of Robotics Research*, v 12 No 1 1993
- [7] Papanikolopoulos N. P. Khosla P. K. "Adaptive Robotic Visual Tracking: Theory and Experiments" *IEEE Tr. on Aut. Control* Vol 38 no 3 1993
- [8] Harris M. "Vision Guided Part Alignment with Degraded Data" DAI TR #615, Edinburgh 1993
- [9] Hollinghurst N. Cipolla R. "Uncalibrated Stereo Hand-Eye Coordination" *Brit. Machine Vision Conf* 1993
- [10] Shapiro L. Zisserman A. Brady M. "Motion from Point Matches Using Affine Epipolar Geometry" *Proc. ECCV* 1994, p 73-84.
- [11] Jägersand M. Nelson R. *Adaptive Differential Visual Feedback for uncalibrated hand-eye coordination and motor control* TR# 579, U. of Rochester 1994.
- [12] Kutulakos K. Jägersand M. "Exploring objects by purposive viewpoint control and invariant-based hand-eye coordination" *Workshop on vision for robots* In conjunction with IROS 1995.
- [13] Jägersand M. Nelson R. "Visual Space Task Specification, Planning and Control" In *Proc. IEEE Int. Symp. on Computer Vision*, 1995.
- [14] Jägersand M. "Perception level control for uncalibrated hand-eye coordination and motor actions" Thesis proposal, University of Rochester, May 1995. <ftp://ftp.cs.rochester.edu/pub/u/jag/lic.ps.gz>.
- [15] Jägersand M. "Visual Servoing using Trust Region Methods and Estimation of the Full Coupled Visual-Motor Jacobian" In *Proc. IASTED Applications of Robotics and Control*, 1996.
- [16] Jägersand M. Fuentes O. Nelson R. "Acquiring Visual-Motor Models for Precision Manipulation with Robot Hands" In *Proc. of ECCV*, 1996.
- [17] W. Z. Chen U. A. Korde S. B. Skaar "Position Control Experiments Using Vision" *Int. Journal of Robotics Research*, v13 No 3 p199-208, 1994
- [18] Corke P. I. *High-Performance Visual Closed-Loop Robot Control* PhD thesis U of Melbourne 1994.
- [19] Hosoda K. Asada M. "Versatile Visual Servoing without Knowledge of True Jacobian" *Proc. IROS* 1994.
- [20] Fuentes O. Nelson R. "Experiments on Dextrous Manipulation without Prior Object Models" *Proc. ISIC* 1996.
- [21] B. H. Yoshimi P. K. Allen "Active, Uncalibrated Visual Servoing" *Proc. of ICRA* 1995.
- [22] Hager G. "Calibration-Free Visual Control Using Projective Invariance" In *Proc. of 5th ICCV* 1995.
- [23] Andersen J. N. "Specifications" In *Int. Enc. of Robotics*, Ed: Dorf R. C., Wiley 1988.
- [24] Broyden C. G. *Mathematics of Computation*, v 19 p 577-593, 1965.
- [25] Garcia, Zangwill *Pathways to solutions, fixed points, and equilibria*, Prentice-Hall, 1981.
- [26] Fletcher R. *Practical Methods of Optimization* Chichester, second ed. 1987
- [27] Dahlquist G. Björck Å. *Numerical Methods* Second Ed, Prentice Hall, 199x, preprint.