To appear: ICCV 1998, Mumbai, India.

Achieving a Fitts Law Relationship for Visual Guided Reaching

N. J. Ferrier*
Department of Mechanical Engineering
University of Wisconsin-Madison, WI, 53706
Email: ferrier@engr.wisc.edu

Abstract

In order to take advantage of the top speed of manipulators, vision can not be tightly integrated into the motion control loop. Past visual servo control systems have performed satisfactorily with this constraint, however it can be shown that the task execution time can be reduced if the vision system is de-coupled from the low level motor control system.

For reaching, there is a trade-off between the accuracy of a motion and the time required to execute a motion. In studies of human motor control this trade-off is quantified by Fitts Law, a relationship between the motion time, the target distance, and the target width. These studies suggest that vision is not used tightly *within the control loop, i.e. as a sensor that is servo-ed on, but rather vision is used to determine where the reaching target is and whether target has been reached successfully. Through a simple robotic example we demonstrate that a similar trade off exists between motion accuracy and the motion execution time for visual guided robot reaching motions.*

1 Motivation

In visual servo controlled systems, visual information is used to derive actuator (motor) commands. Control inputs for an actuator (either a robot pan-tilt axis, robot head, mobile base, eye-in-hand configuration, or a separate robot manipulator) are derived from the image data. Typically the visual processor extracts contours, features, corners, or other visual primitives. Based on this visual data and the desired robot task, motion commands are given. The ability to control actuators based on visual servo control has been successfully demonstrated [1, 5, 6, 16, 18, 22]. Preliminary visual servo control efforts [6, 18] were hampered in that only the simplest of visual tasks could be performed fast enough to evaluate the control system, hence stable tracking was demonstrated only for simple visual tasks, such as tracking bright spots. Improved visual tracking methods [2, 20] perform computations in a region of interest (or within the region of uncertaintly) resulting in fast tracking of complex objects in some cases using only modest hardware [2, 9]. These tracking methods require a sufficiently accurate model of the dynamics and accurate initialization information (locating the object to track). However, even using enhanced visual tracking within the visual servo control systems will yield systems with the performance limited by the speed with which visual computations can be performed. The ability to separate the visual sensing from the motion control will facilitate the speed with which certain tasks can be performed.

Visual servo control systems must take into account the delays associated with the visual computation. Tracking stability can only be achieved if the sensing delays are sufficiently small, or, if the dynamical model is sufficiently accurate. Various approaches have been taken to compensate for the delay of the visual system in visual servo control. For Kalman-filter type tracking with a stochastic linear diffusion process, Olivier [17] mathematically describes the relationship between the processing time, the resolution of the sensing (effectively the search window size), and tracking accuracy. For certain system models and noise assumptions, it can be shown that the system *cannot* be real-time observable hence real-time tracking is not possible [17]. Successful visual tracking for servo control has been achieved by balancing this trade off between the accuracy of the sensing and the visual data processing time. Roberts *et al* [20] show that for systems for which the dynamical model is known with sufficiently accuracy, the search area (window size or resolution) can be kept small enough such that the processing time is sufficiently small for accurate tracking. Clark & Ferrier [6, 8] utilize feed-forward compensation for the processing delay. The dynamical model must be known with sufficient accuracy to produce stable tracking. Schnackertz & Grupen [21] also utilize a feed-forward term. Similar feedforward, or

*This research supported in part by NSF IRI-9703352

predictive, techniques are employed in visual tracking techniques which incorporate the computation delay (expressed as the number of frames 'dropped' during the compute cycle) within the dynamical model to predict the tracker motion [2]. (The tracker in [2] does not control a motor system, however the tracker could be readily incorporated into a servo-loop). Nelson *et. al.* [16] assume commensurate delays and model the delay d within the system model and a new control law is developed to account for this delay. For sufficiently small delays, tracking is successful, however based on Olivier's results, the fixed resolution sizes used in the visual processing may lead to non-observability of the tracker. All of the systems discussed above assume that the control loop incorporates the visual sensor, and it's inherent delays, *within* the motion control loop. The total speed of the motion control system is hence limited by the speed of the vision system. It should be noted that tracking enables temporal coherence that is necessary to facilitate the visual processing. The speed with which computer vision algorithms can identify an object within the visual field must be taken into account.

This paper revisits robotic visual servoing from a new perspective: vision processing may take place in parallel with the motion control *but* the motion *execution* is independent of the visual system. This form of motor control is thought to exist in humans. Studies of visual attention and human motor control suggest that vision *is not* utilized during motion execution (motions are ballistic). Some evidence suggests that vision is used at the onset of the task (to determine initial conditions) and at the end of the task (to evaluate the motion and determine success, failure, etc.). Visual feedback could be used to assess the entire task rather than each individual movement of during the task. Vision processing is in parallel with motion control but motion is not 'servo-ed' on vision. While entire tasks (e.g. driving) cannot be performed without visual feedback, many tasks or portions of a task, are readily performed without tight integration of the vision system within the control loop. In this paper we explore the interaction of vision and motion control through analysis of robot reaching tasks for a simple manipulator.

In section 2 we present theories from studies of human motion control. In section 3 we present background on tracking and demonstrate that the resolution required to track a rapidly moving object will violate real-time observability constraints (hence tracking is impossible). In section 4 we present an elementary example from motor control which provides a basis to doubt the utility of closed visual servo control for some tasks. In section 4.2 explore the possibility of decoupling vision from motor control to achieve a Fitts Law relationship.

2 Motor Control, Vision & Attention

Separating vision and motor control tasks is inspired by studies of attention and vision in human motor control (see e.g.[12, 14]). Studies of human motor control have established a relationship between the accuracy of a movement (in terms of size of the target and distance to the target) and the time to perform a movement[12]. Accuracy depends on both the distance to the target and the size of a target. The time for a motion (MT) is given by Fitt's Law[10] which stated as:

$$MT = a + b \log_2 D/W$$

where a and b are constants, D is the distance from the starting position to the target, and W is the target width. Observe that the distance and width have inverse but proportional effects on the motion time. If the width is doubled while the distance is held constant the MT decreases by a constant, and if the distance is halved while the width is held contsant the MT decreases by the same amount (see figure 1).

Various explanations have been given in the literature for the observed pattern of motion time. Initially "learning" a motion involves more feedback, then as skill is acquired, movements (at least for short periods) may be autonomous of visual feedback [19]. Explanations of this phenonmena usually cite open-loop vs. closed-loop control: a coarse motion gets within reach of the target (open-loop), then fine motions correct the movement (closed loop) [13, 14]. Sensing, such as vision and touch, are presumed to play only in the corrective movements. Fine or accurate motions may require multiple corrective movements and hence the increase in MT. The kinesthetic sensors control the "ballistic" motion. Visual feedback is used for evaluation of motion and possible recalibration.

For human motion control, the decision of where to move and the movement itself are independent. Visual feedback is for periodic correction or modulation of the pattern of movements. In contrast, visual guided robot motions typically perform closed loop on the sensory data. The motion time limitation is thus the speed with which the visual sensing can be performed. Vision processing is typically the slowest sensor used in robotics (with data acquisition and visual data processing typically run much less than of 10 Hz).

Another property of motion control utilized in establishing our visual control framework is that motion is segmented. It is a common practice to break a complex motion into simpler motions. Trajectories generated for a robot are often composed of a series of simple motions. Visual feedback is not utilized during a motion segment but may be utilized at the start and/or end of a motion. A coordinated motion, requiring multiple subsystems will "down load" the parameters to each the subsystems (motor controllers and vision system). These parameters will apply while the motion segment is performed. The degree of interaction between the visual sensors and the motion control system depends on the length of the motion segment. Reaching is a simple motion segment (which can often be considered as part of a more complex task). We will concentrate on this motion segment in this work.

3 Visual Tracking

There has been considerable effort to produce real-time algorithms for tracking moving objects. The ability to track a target depends upon the dynamics of the target (how fast it is moving), the computation time (amount of data to process), and the desired tracking accuracy. Often the visual tracking problem is formulated using a recursive filter framework [2, 17]. A stochastic model of the dynamics of the target with Gaussian noise is used (with the notation based on [17]):

$$dx(t) = Ax(t)dt + Bdw(t) \tag{1}$$

where $w(t) \sim N[0, Q]$ and with measurement process $y(t) = Hx(t) + v$, $v \sim N[0, R]$, where $R(t)$ is the resolution of the measurement process. Under Gaussian noise assumptions, an estimate of the target, $\hat{x}$, and the associate covariance $P = E[(x-\hat{x})\cdot(x-\hat{x})^T]$ are obtained using Kalman filtering (see e.g. [4, 11]). Accuracy of tracking is expressed via the covariance $P(t)$. If $P(t)$ diverges then the process is not observable – tracking is not possible.

Olivier[17] gives a precise discussion on real-time observability of a one dimensional linear stochastic process with variable sample times. The sample time is the interval between measurements which effects the stability of the tracking. In visual tracking this time is typically used for processing visual data. Often this time is taken to be a constant (for example [16]). Mathematical formulations and analysis of tracking also assume is that the search window is sufficiently large[1] such that the measurement process "works" (the target is found). If the resolution within that window is too large ($R \to P$) then tracking will also fail. Figure 2 graphically represents these ideas. Olivier shows that for constant resolution measurement processes (constant R), real-time observability is *not* possible for a linear stochastic diffusion process. Hence the resolution should be adapted to ensure stable tracking. The main result of [17] which we will use here is that if the resolution is proportional to the covariance, $R = \alpha P$ ($0 < \alpha \ll 1$), and the sample time is proportional to the search window area (which is a function of the covariance P) then tracking is real-time observable (the covariance does not diverge) for **sufficiently slow dynamics**. For example, the one dimensional linear system:

$$\begin{aligned} dx &= axdt + dw \;;\; E[dw^2] = \Omega^2 dt & (2)\\ y(t_k) &= x(t_k) + v \;;\; v(t_k) \sim N[0, R_k] & (3) \end{aligned}$$

The Lyapanov equation for the covariance, P is

$$\frac{dP}{dt} = 2aP + \Omega^2$$

The Riccati equation with resolution $R_k = \alpha P_k$ is

$$P_{k+1} = \left(e^{2aT_k}\frac{\alpha}{1+\alpha}\right)P_k + \frac{\Omega^2}{2a}\left[e^{2aT_k} - 1\right] \tag{4}$$

where T_k is the sample time for the k^{th} measurement. The covariance evolves as

$$P_{k+1} = \gamma P_k + \beta.$$

For $\gamma < 1$ the covariance decreases and hence the target can be tracked. The expression

$$\gamma < 1 \qquad \text{or} \qquad e^{2aT_k}\frac{\alpha}{1+\alpha} < 1$$

makes explicit the trade off between the sample time T_k, the resolution policy α, and the system dynamics, a. If the system is sufficiently slow for the time scale used $a < a(T)$ then we can pick a resolution to ensure tracking. This is in agreement with the results of Blake *et al* [2] who also provides an algorithm for determining the spatio-temporal scale proportional to the estimated covariance.

The majority of Olivier's [17] work is for one dimensional tracking (as the mathematical analysis becomes difficult for higher dimensions). In addition, the system dynamics are assumed to be linear. Examples are presented for tracking Brownian motion, which can be considered a "slow" dynamical system. Blake *et al* [2] require approximate knowledge of the target dynamics and tracking is based on a constant acceleration dynamical model.

[1] A search window that is greater than 2σ on each axis where σ^2 is the variance in the estimated target position along that axis is typically used.

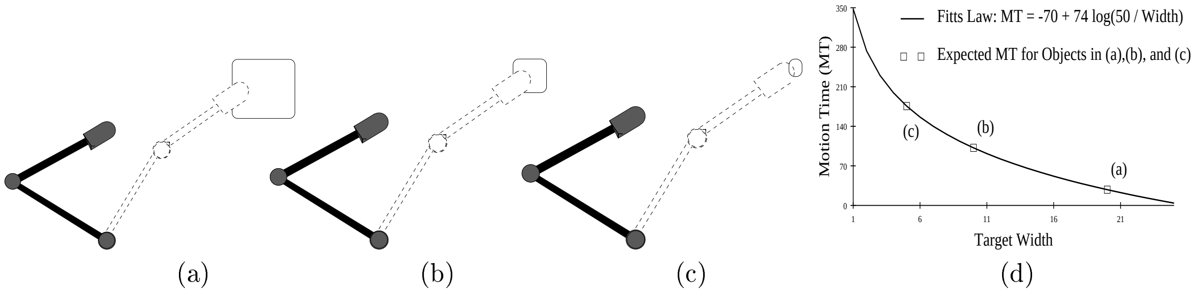

Figure 1: (a,b,c) Three motions with decreasing target size. (d) Fitt's Law using data from Fitts and Peterson with the expected motion times for (a), (b) and (c) indicated.

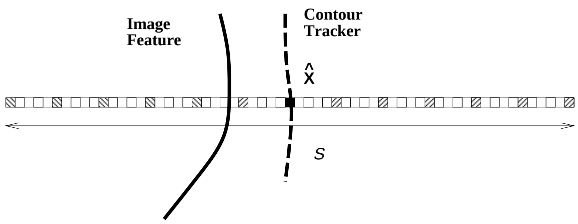

Figure 2: The width of the search region, S, should be large enough so that the target is found ($> 2\sigma$). Proportional resolution policies subsample that search region. The cartoon shows a of search window with the window size of 60 pixels and a resolution of 5 pixels. (For $P = \sigma^2 = 100$, we searched $\hat{x} \pm 3\sigma$ with a proportional resolution policy of $R = 0.05P = 5$ pixels. The processing (of only 10 pixels here), will produce an estimate $\hat{x}$ with accuracy limited by the resolution.

For tracking a robot manipulator, the system could utilize knowledge of the robot's motor commands (along with calibration information giving the change of coordinates from the camera to the end effector). This is the emphasis of the work on visual compliance [5, 21] and the works discussed earlier that used model reference tracking or feed forward. However, as we will show next, the sudden movement of the manipulator must be carefully orchestrated with the initialization of the tracker or the tracker will lose the target.

4 Tracking Robot Reaching tasks

In this section we demonstrate that a Fitts Law relationship for reaching tasks would be an improvement over a closed-loop visual servo control. The speed with which motors can move and the delays associated with visual tracking make task times longer when using visual feedback tightly within the control loop. We compare the speed with which a tracker, as described in the previous section, can follow a manipulator motion to the speed with which an open loop motion can be performed.

As a simple example, consider the motion of a "1 link" manipulator (a simple rotor and link). We use a simple dynamical model describing the angular motion $\theta(t)$ for a given input torque $\tau(t)$

$$\tau = J\ddot{\theta}.$$

To displace the rotor by $\Delta\theta$ radians, given a torque limited motor ($\tau \leq \tau_{max}$), and assuming bang-bang control, the minimum time for the motion is bounded from below and time is a function of the inertial properties of the link, J, the desired angular displacement, $\Delta\theta$, and the maximum torque, τ_{max}, ie. $T_{min} = T(\Delta\theta, J, \tau_{max})$. There is some overhead associated with computing the control parameters and "downloading" the motor commands to a motor controller. This time is typically independent of the size of the motion and can be assumed constant, a_o. With the execution time given by the time function the motion time for a displacement is:

$$MT \approx \underbrace{a_o}_{\text{overhead}} + \underbrace{b_o\sqrt{\frac{J\Delta\theta}{\tau_{max}}}}_{\text{task execution}} \qquad (5)$$

If the control is "perfect" and there is no friction or other unmodeled phenomena, we can achieve this "optimal" motion time given by (5). In practice there will be errors from inadequate models of inertia, friction, etc. In addition, bang-bang control is only a theoretical model, instantaneous changes in acceleration are not realized. Due to these errors, the final position may not be the exact desired position. Feedback control can correct for unmodeled phenomena but the resulting time for the motion may exceed the "optimum" time. Feedback for such simple tasks usually relies on kinesthetic sensors (shaft encoders). We consider visual feedback here to demonstrate the timing issues for tracking a manipulator.

For this simple system, there are two points of note. First of all, when using bang-bang control the motion commands produce abrupt changes in the angular acceleration. The manipulator is initially at rest. When the motor commands are executed the manipulator moves with high accelerations. Observability constraints for visual tracking all require sufficiently slow dynamics. The second point is that the accelerations are not constant, as the tracker models assume. In joint space the accelerations are constant but, as shown in figure 3, the resulting motions in workspace are not linear. The dynamics governing motion of a rotor are linear in the angular position variable and hence PID control is highly successful. However, controlling the motion with visual feedback requires expressing the system dynamics in workspace (or camera space) variables resulting in non linear dynamics.

Visual feedback for reaching tasks must track the position of the end effector in workspace. This motion is fast, abrupt, and may follow erratic trajectories (depending on the control used). For a multi-linked arm the dynamics are non-linear (even with just two links!), and we cannot find closed forms solutions for the equations of motion (or the time for motion). Motion can be described along a specified path, then determining the optimum time requires involved computations of switching curves[3, 7, 23], and in general the motion time can not be expressed in closed form for more than one degree of freedom. For point to point motion, the time will depend on the control strategy chosen. For a simple control strategy, such as a strategy which moves one link at a time, the resulting motion in workspace may not be a smooth curve – discontinuities in the derivative are seen as abrubt changes in direction (see figure 4). The dynamics of these trajectories is not well modeled by the linear, constant acceleration trackers described in the previous section. Even if the vision system is provided with a reference model for the manipulator motion, the dynamics are non linear and the tracking error (from visual trackers based on linear models) can only be kept small if the motion is slow. In addition the speed with which manipulators can move requires the tracker and the manipulators to be *synchronized*! A slight offset in starting to track will lose the target.

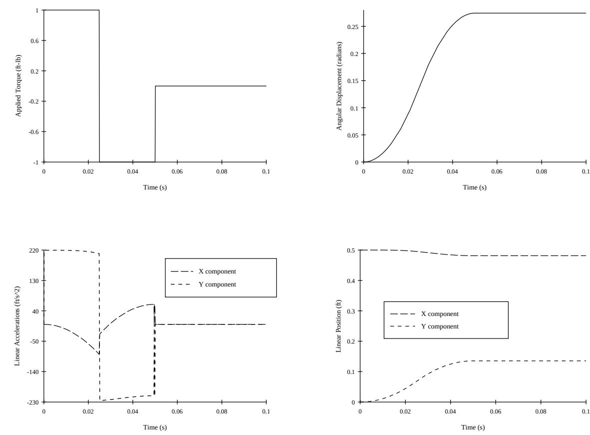

Figure 3: For a single link manipulator which weights 8 oz, of length 6 in., and max torque of 200 oz-in., we obtain the profiles for a motion of $\pi/8$ radians (total time is $\approx$ 0.05 s). The top row shows the motion in joint space (joint acceleration and joint position). Here the accelerations are constant. The lower row shows graphs of the acceleration and position in workspace coordinates.

4.1 Visual guided reaching speed bounds

We have given arguments as to why tracking a manipulator could be problematic. We return now to the simple 1 link manipulator to demonstrate numerically what the motion/tracking implications are. The dynamics of the manipulator motion on the image plane will depend upon the kinematic relationship between the camera and the manipulator. In [5, 21] this relationship is given by the visual compliance. For our simple analysis we will assume an overhead camera in a plane parallel to the plane of the manipulator.This configuration gives a simple scaling between the manipulator dynamics and the image plane dynamics. Assuming classical pinhole optics and a fixed distance, z_o from the camera the image coordinates (in pixels) for the manipulator end-point (x, y) is

$$u = k_x \frac{f}{z_o} x; \quad v = k_y \frac{f}{z_o} y \tag{6}$$

where k_x, k_y are the scale factors to convert to length measured in pixels. Figure 3 plots typical values for accelerations of the manipulator in workspace coordinates. The image motion corresponding to the motion of the manipulator is scaled using (6). It is straight forward to see that image motions can be extremely large. Using a 4mm lens at a distance of 30 cm from the manipulator, we expect image motions of the order of 1000 pixels/frame (for imaging at 30 frames/s). At

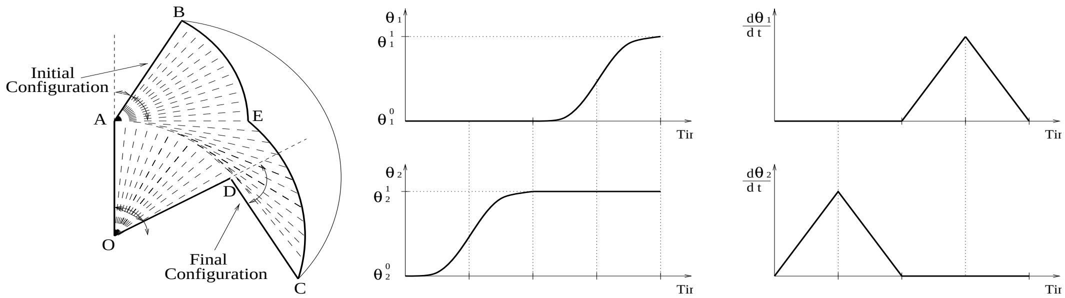

Figure 4: A decoupled control law that moves one link at a time produces abrupt changes in direction.

such distances the target leaves the field of view of the camera rather quickly! Visual tracking as described in section 3 will not be successful at these speeds.

4.2 Motion Segments

Because visual tracking may not be feasible for rapid manipulator motions, the control strategy must be modified. For many tasks it is acceptable to slow down the robot motion to facilitate visual tracking. However, there are tasks for which the actual motion does not *need* to be monitored. In reaching tasks, the motors systems can be commanded to move (independent of the visual system). The vision system can "saccade" to the expected the final manipulator position and re-locate the manipulator. *To do so the fingers have to be readily located in the image.* After the reach motion is completed, the vision system evaluates success or failure and may re-initiate a reach task. New motor commands are issued and the cycle repeats. Segmented motor control can use visual feedback for periodic correction or modulation of the pattern of movements, as is thought to occur in humans. As described earlier, this is the form of visual-motor interaction that is thought in human motor control. The control is broken into a sequence of steps which may be iterated.

Step 1 Estimate (if not known) the current manipulator position. Estimate the target position with respect to the camera. Without prior knowledge, the entire image needs to be searched. The initial covariance P_o is commensurate with the image size. The resolution, however, can be coarse to determine a rough estimate of the target location. Kinematic information from calibration or the image jacobian [21] converts the image location to a desired manipulator position.

Step 2 Determine the required motor commands to perform the motion. It is beyond the scope of this paper to discuss how the motor commands can be determined. The motion is executed and we assume that the final position is sufficiently close to the commanded position. (This is not an unrealistic assumption as most well-tuned control strategies can achieve high accuracy. Examples in the literature include [3, 7, 23, 15]).

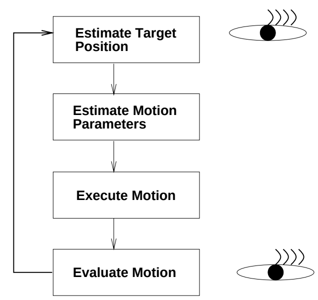

Figure 5: Graphical depiction of the vision/motion control strategy. Vision is used to monitor the motion, not as servo feedback.

Step 3 The resulting position for the manipulator will depend on many factors, such as the accuracy of the image jacobian calibration, the accuracy of the motion control strategy used, the models used for the manipulator dynamics, friction effects, etc. With respect to this work the relevant parameter is the covariance of the target estimate obtained from the image analysis. The initial covariance of the target estimate and the resolution at which the processing was performed give an approximation to the accuracy of the reaching task. If the covariance is small enough, and the target width is wide enough, one motion segment will suffice to "reach" the target. If the target width is less than the covariance, the process must be repeated (from step 1).

4.2.1 Adding up the motion segment times

In the following section we make the (unrealistic) assumption that the only uncertainty in the position is due to uncertainty from the visual information processing. 'Perfect' calibration has been performed to determine the image jacobian exactly. This assumption will allow us to get an approximation to the task motion time based on motions which are bounded by the target position uncertainty. We now try to determine the task time for a reach motion.

The initial reach depends on the distance of the initial manipulator position to the target. The time to perform the motion depends on the distance to the target (as well as the peak torques and the inertial properties which are fixed). If we assume that the manipulator is within the field of view, and that the initial image contains the target, then the uncertainty for the initial target estimate is related to the size of the image (P_o is the image size). Recall that under real-time observable conditions, the covariance decreases as a geometric sequence with $P_k \leq \gamma P_{k-1} + \beta$ where γ depends on the resolution, the sample time, and the system dynamics. The resolution policy $\alpha \ll 1$ will determine the resolution of the initial target estimate, R_o.

Using a bang-bang control strategy, the time to perform a motion depends on the distance moved. However, each iteration decreases the size of the motion as subsequent motions are always bounded by the size of the uncertaintly of the target estimate from the previous motion. The process terminates when the target covariance is small enough (i.e. $P_k < W^2$). The total time is the sum of each of the motion segments. Each segment time is related to the displacement, a decreasing series.

Once again, we return to the one dof manipulator to give precise values to justify this arguement. For a 1d linear system (equation 3) the Riccati equation was given (4). For a stationary target (the manipulator resting at the current target estimate)

$$P_k \approx \left[\frac{\alpha}{1+\alpha}\right] P_{k-1} + \beta_{k-1}.$$

or

$$P_k \approx \left(\frac{\alpha}{1+\alpha}\right)^k P_0 + \sum_{i=0}^{k-1} \left(\frac{\alpha}{1+\alpha}\right)^i \beta_{k-1-i}$$

In one dimension the covariance is a scalar (the variance) and the search scale (if using a 2σ scale) will be on the order of $2\sqrt{P_k}$. The motion to the target will be within this bound. For each motion segment, the time to move is proportional to the square root of the distance moved (equation 5):

$$\text{time}_k \;\sim\; \lambda\sqrt{\Delta\theta} \;\sim\; \lambda\sqrt[4]{P_k}.$$

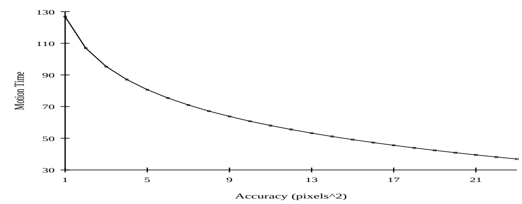

Figure 6: The motion time vs. positional accuracy for a visual guided reach. This closely fits the Fitts law relationship with constants $a = 127$ and $b = 19.9$.

and the total time will be:

$$T_o + \sum_{k=1}^{M} \lambda\sqrt[4]{P_k}$$

where M is the iteration when the covariance is smaller than the target width ($R_M \approx W^2$ so $\alpha P_M \approx W^2$). This computation of motion time closes approximates the Fitts Law relationship (figure 6).

5 Discussion/Future Work

We have presented a framework in which one may de-couple the vision system from the motor control system. Control loops that are tightly coupled to *slow* sensors, such as vision, are inherently limited. We explore whether tasks, such as reaching, can be achieved in an open-loop manner, much like the human system. As the task difficulty increases (narrower target), then the system must use intermediate feedback to reach the target. Thus motion control for a robot can also behave according to a modified Fitts Law. For smaller target width, W, a finer resolution must be used and hence the reaching task time increases.

The framework presented here merely scratches the surface. Most of the analysis is in one dimension because analytic results (such as an estimate on the time to execute a path) are generally not solved in closed form. In addition in order to isolate the errors due to uncertainty in the motor control *vs.* errors in the image analysis, the system must be carefully calibrated. Our initial experiments have shown that for our particular mechanism this may not be possible and further analysis will have to formulate the problem with explicit uncertainty parameters from the motor control and from the visual system.

This framework however provides some unique opportunities. The position of the end effector after a 'coarse motion' can be used to modify the commanded joint torques. To truly perform open-loop ballistic motion segments, the motor controller must specify the joint torques to be applied for a particular time instance. In general, PID control is used to ensure that the final manipulator position is the desired position. We are exploring the possibility of using the "adjusted" torque commands (the commanded setpoint and the PID controller modifications to this command) to determine the equivalent open-loop command for a reaching motion. When coupled with a learning algorithm, such control strategies will enable improved performance for motion control systems which utilize vision. As vision will always be the "slow sensor", such de-coupling could facilitate a move from closed to open loop control.

Clearly some tasks will not lend themselves to this form of control strategy, however for various assembly tasks, the ability to separate the visual control from the motor will facilitate the incorporation of visual sensors in robotic devices. This paper is one step closer towards building an understanding how vision can be used to facilitate motion control without servoing on the visual sensor.

References

[1] P. Allen, B. Yoshimi, & A. Timcenko. Real–time visual servoing. In *Proc. IEEE Int. Conf. Robotics & Automation*, pp. 851–856, 1991.

[2] A. Blake, R. Curwen, & A. Zisserman. A framework for spatio-temporal control in the tracking of visual contours. *Int. J. Computer Vision*, 1993.

[3] J. Bobrow, S. Dubowsky, & J. S. Gibson. Time-optimal control of robotic manipulators along specified paths. *Int. J. Robotics Research*, 4(3):3–17, 1976.

[4] A. Bryson & Y. Ho. *Applied Optimal Control.* Hemisphere Publishing Corp., New York, 1975.

[5] A. Castano & S. Hutchinson. Visual compliance: Task-directed visual servo control. *IEEE Trans. on Robotics & Automation*, 10(3):334–342, 1994.

[6] J. Clark & N. Ferrier. Modal control of visual attention. In *Proc. Int. Conf. on Computer Vision*, pp 514–531, Tarpon Springs, Florida, 1988.

[7] E. Croft, B. Benhabib, & R. Fenton. Near-time optimal robot motion planning for on-line applications. *J. robotic systems*, 12(8):553–567, 1995.

[8] N. Ferrier & J. Clark. The Harvard binocular head. *Int. J. Pattern Recognition & AI*, March 1993.

[9] N. Ferrier, S. Rowe, & A. Blake. Real-time traffic monitoring. In *IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, 1994.

[10] P. Fitts & J. R. Peterson. *J. Experimental Psychology*, 65:423–432, 1964.

[11] A. Gelb *et. al.* *Applied optimal estimation.* MIT Press, Cambridge, MA, 1974.

[12] S. Keele. *Attention & Human Performanace.* Goodyear Publishing Co., 1973.

[13] S. Keele. Learning & control of coordinated motor patterns: the programming perspective. In J. Kelso, editor, *Human Motor Behavior.* Lawrence Erlbaum Associates, 1982.

[14] R. Klein. Attention and movement. In G. Stelmach, editor, *Motor Control: Issues & Trends.* Academic Press, 1976.

[15] A. Kumagai, D. Kohli, & R. Perez. Near-minimum time feedback controller for manipulators using on-line time scaling of trajectories. *Trans. of the ASME*, June 1996.

[16] B. Nelson, N. Papanikolopoulos, & P. Khosla. Visual servoing for robotic assembly. In *Visual Servoing: Real-Time Control of Robot Manipulators Based on Visual Sensory Feedback*, pp 139–164. World Scientific Press, 1993.

[17] C. Olivier. Real-time observatiblity of targets with constrained processing power. *IEEE Trans. on Automatic Control*, 41(5):689–701, 1996.

[18] K. Pahlavan & J.-O. Eklundh. Heads, eyes and head-eye systems. In H. Christensen, K. W. Bowyer, & H. Bunke, editors, *Active Robot Vision.* World Scientific Press, 1993.

[19] R. Pew & S. Baron. The components of an information processing theory of skilled performance based on an optimal control perspective. In G. Stelmach, ed., *Information processing in motor control & learning.* Academic Press, 1978.

[20] J. Roberts & D. Charnley. Parallel attentive visual tracking. *Engineering Applications of Artificial Intelligence*, 7(2):205–215, 1994.

[21] T. Schnackertz & R. Grupen. A control basis for visual servoing tasks. In *Proc. 1995 IEEE Conf. on Robotics & Automation*, Nagoya, Japan, 1995.

[22] J. Son, R. Howe, J. Wang, & G. Hager. Preliminary results on grasping with vision and touch. In *Proc. IEEE Int. Conf. on Intelligent Robots & Systems*, 1996.

[23] H. Yang & Slotine. Fast algorithms for near-minimum-time control of robot manipulators. *Int. J. Robotics Research*, 13(6):521–532, 1994.