Human-in-the-Loop Camera Control for a Mechatronic Broadcast Boom

Rares Stanciu and Paul Y. Oh

***Abstract*—Platforms like gantries, booms, aircrafts, and submersibles are often used in the broadcasting industry. To avoid collisions and occlusions, such mechatronic platforms often possess redundant degrees-of-freedom (DOFs). As a result, manual manipulating of such platforms demands much skill. This paper describes the implementation of several controllers that, by using computer vision, attempts to reduce the number of manually manipulated DOFs. Experiments were performed to assess the performance of each controller. A model for such a system was developed and validated. To determine how the visual servoing can improve the tracking, a novice operator and an expert were asked to manually track a moving target with the assistance of visual servoing. The results of these tests were analyzed and compared.**

***Index Terms*—Image processing, robotics, visual servoing.**

I. Introduction and System Description

THERE are many tools that carry cameras. Their working domain is usually surveillance, surface inspection, and broadcasting. Devices like rovers, gantries, and aircrafts often possess video cameras. The task is usually to maneuver the vehicle and position the camera to obtain the desired fields-of-view. A platform widely used in the broadcasting industry can be seen in Fig. 1. The specific parts are usually the tripod, the boom, and the motorized pan–tilt unit (PTU).

Manual operation of such a tool requires two skilled operators. Typically, one person will handle the boom while the second operator will coordinate the PTU camera to track the subjects using two joysticks. Tracking the moving objects is difficult because there are many degrees-of-freedom (DOFs) to be coordinated simultaneously. Increasing the target's speed increases the tracking difficulty. Using computer vision and control techniques ensures the automatic camera tracking and reduces the number of DOFs the operator has to coordinate. This way the platform can be operated by one person concentrating only on the booming. The use of such techniques enables the tracking of faster moving objects.

Searching through the literature on this subject reveals that there is a wealth of existing research in the visual servoing domain. An excellent starting point in the literature search is [1]. Extensive research is described in [2], [3], [4]–[8], [9]. It is to be noted that in these publications, researchers have dealt completely with the automated hardware (where no operator is involved). The system described in this paper is operated by humans. Some of the seminal man–machine interface work is represented by [10]–[12].

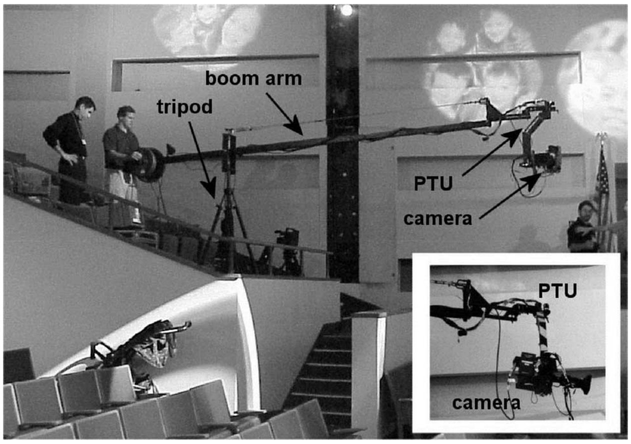


Fig. 1. Operator can move the boom horizontally and vertically to position the camera. The pan–tilt (*lower right inset*) head provides additional DOFs.

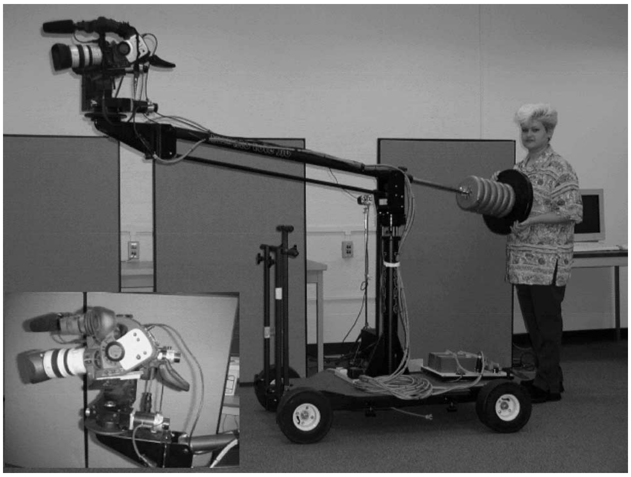


Fig. 2. Operator can boom the arm horizontally and vertically to position the camera. The pan–tilt head (*lower left inset*) provides additional DOFs.

The system utilized for experimentation is shown in Fig. 2. The platform is composed of a four-wheeled dolly, boom, motorized PTU, and camera. The dolly can be pushed and steered. The 1.2-m-long boom is linked to the dolly via a cylindrical pivot that allows the boom to sweep horizontally (pan) and vertically (tilt). Mounted on one end of the boom is a two-DOF motorized PTU and a video camera weighing 9.5 kg. The motors allow an operator to both pan and tilt the camera 360° at approximately 90°/s. The PTU and the camera are counterbalanced by a 29.5-kg dumbbell mounted on the boom's opposite end. Use of this boom-camera system normally entails one or more skilled personnel performing three different operations.

Manuscript received August 20, 2005; revised July 26, 2006. Recommended by Technical Editor H. Hashimoto.

The authors are with Drexel University, Philadelphia, PA 19104-2876 USA (e-mail: ris22@drexel.edu; paul@coe.drexel.edu).

Digital Object Identifier 10.1109/TMECH.2007.886252

1) With a joystick, the operator servos the PTU to point the camera. A PC-104 small board computer and an ISA bus motion control card allow for accurate and relatively fast camera rotations.
2) The operator physically pushes on the counterweighted end to boom the camera horizontally and vertically. This allows one to deliver a diverse range of camera views (e.g., shots looking down at the subject), overcomes PTU joint limitations, and captures occlusion-free views.
3) The operator can push and steer the dolly in case the boom and PTU are not enough to keep the target image in the camera's desired field-of-view.

Tracking a moving object using such a tool is a particularly challenging task. Tracking performance is thus limited to how quickly the operator manipulates and coordinates multiple DOFs. Our particular interest in computer vision involves improving the camera operator's ability to track fast-moving targets. By possessing a mechanical structure, actuators, encoders, and electronic driver, this boom is a mechatronic system. *Visual-servoing* is used to control some DOF so that the operator has fewer joints to manipulate.

This paper describes the implementation of several controllers in this human-in-the-loop system and discusses quantitatively the performance of each. The CONDENSATION algorithm is used for the image processing. As this algorithm is described in some publications [13], this paper will not focus on the image processing. Section II describes the experimental setups used. The controllers are described in Section III. Development and validation of a boom-camera model is also presented. Section IV describes a comparison between a well-skilled operator versus a novice, both with and without the visual servoing. Section V presents the conclusions.

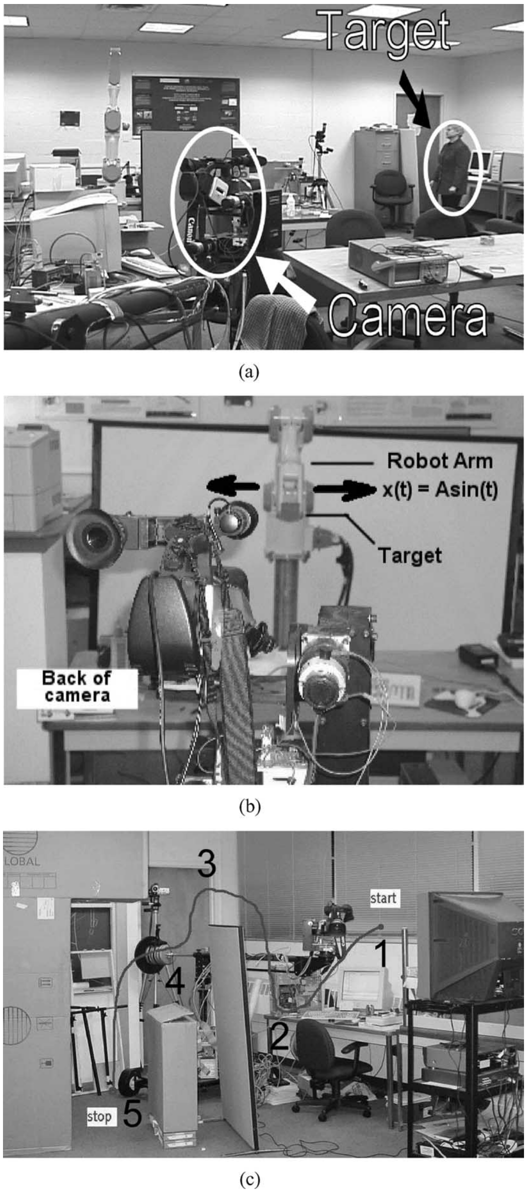
II. Experimental Setups

The "artistic" side of a film shooting scenario is often very important. Because they involve humans, these scenarios are (strictly speaking) not repeatable. Therefore, to compare the behavior of different controllers, an experimental framework is needed. As such, the experiments were designed to offer the best possible answers for both scientific and artistic community.

The first experiment was people-tracking. A person was asked to walk in the laboratory. The camera attempted tracking while an operator boomed. Fig. 3(a) shows such an experiment.

Each new designed controller attempted to increase tracking performance. The second experiment was developed in an attempt to design a metric for performance. A Mitsubishi robotic arm was instructed to sinusoidally move the target back and forth [Fig. 3(b)]. While the operator boomed, the camera tracked the target. Target motion data, error, and booming data were recorded during the experiments and plotted for comparison with previous results.

At this point, it was interesting to determine whether the vision system was usable in sport broadcasting. An experiment in which the camera tried to track a ball moving between two people was set up. The experiment showed successful tracking but highlighted some challenges. This setup is described in Section III-H.



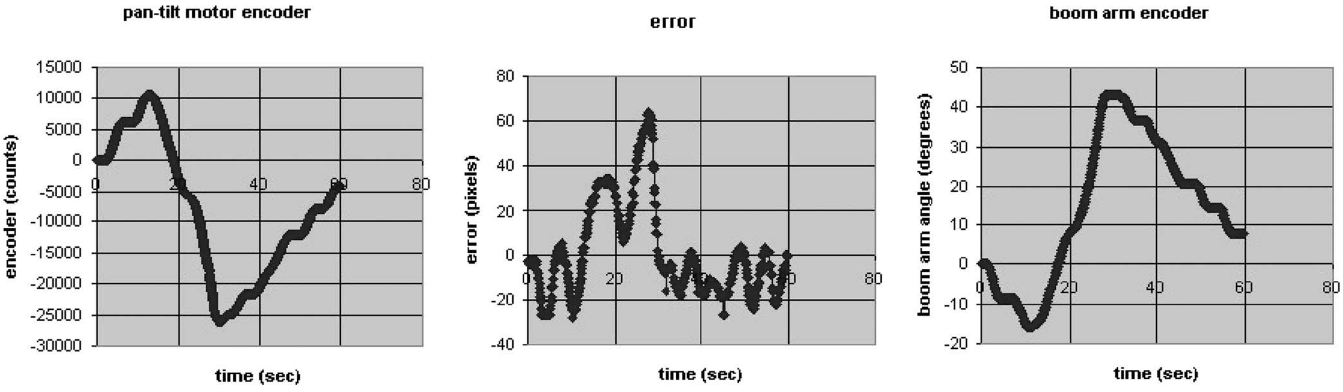
(a)

(b)

(c)

Fig. 3. (a) Typical people-tracking setup. A subject walks around and the camera attempts tracking while booming. (b) Wooden block target was mounted on the end-effector of a Mitsubishi robot arm (*background*). The boom-camera system (*foreground*) attempts to keep the target's image centered in the camera's field-of-view. (c) Novice and a well-skilled operator will manipulate the boom appropriately to move the camera along the shown path, with and without the help of visual servoing. In addition to booming, under manual control, the operator will also have to coordinate camera's two DOFs using a joystick. Visual servoing tracking error is recorded for comparison.

Once the camera was considered to ensure a satisfactory tracking performance, it was interesting to determine how it can help the operator. To answer this question, another experiment was designed. Again, the Mitsubishi robotic arm was used. This time, the robot moved the target on a trajectory corresponding to the number "8." A novice and an experienced operator boomed along a predefined path and attempted tracking the robot end-effector with and without vision. Fig. 3(c) shows the way the operator should boom. The visual servoing tracking error was recorded and plotted. This experiment is described in Section IV.



(a) (b) (c)

Fig. 4. $K_x = 100$. (a) PTU motor encoder. (b) Pixel error. (c) Boom-arm encoder.

III. Controllers Description

This section presents the hypotheses, describes the controllers in detail, and discusses the experiments and results during this research.

A. Proportional Controller

To establish a base level, the first of our hypotheses was launched. It states that by *using a very simple controller (proportional) and a very simple image processing technique (color tracking), the camera is able to track a moving target when booming*.

The proportional controller was implemented. The current target position in the image plane is compared with the desired position and an error signal is generated. This error signal will determine the speed of the camera in its attempt to bring the target in focus. The controller gain K_x was set to 100. People-tracking experiment was attempted using this controller [Fig. 3(a)]. A person wearing a red coat was asked to walk in the laboratory. The color-tracker board was trained for red. The task was to keep the red coat in the camera's field-of-view while an operator boomed. In this experiment, the camera–target distance was about 5 m.

To assess the controller performance quantitatively, a toy-truck was to be tracked. An artificial white background was used to help the vision system to detect the target. In this experiment, the camera–target distance was 3 m. The toy moved back and forth while the camera attempted tracking. Camera motion data, booming data, and tracking error were recorded. The plots can be seen in Fig. 4. Fig. 4(a) shows the pan motor encoder indication, (b) shows the error (in pixels), and (c) shows the booming angle (in degrees). It can be seen that as the operator is booming and the target is moving, the controller performs a visually servoed counterrotation. The system was able to track the moving target even when using a very simple controller. Still, as one expects, there were two challenges: system stability and tracking performance.

The experiments have demonstrated that the key design parameter, when visually servoing redundant DOF systems, is stability, especially when the target and the boom move 180° out of phase. If boom motion data is not included, camera pose cannot be determined explicitly because there are redundant DOFs. As a result, the system could track a slow-moving target rather well, but would be unstable when the target or boom moves quickly.

The second issue was the tracking performance. With the proportional controller, the operator boomed very slowly (less than 1°/s). The target also moved slowly (about 10 cm/s). Any attempt to increase the booming or target speed resulted in the tracking failure. Both the experiments proved the first hypothesis. It is important to underline that the vision had no information about booming. Introducing booming information could improve tracking performance as well as stability.

B. Feedforward Controller

The second hypothesis was that *by using a feedforward control technique, we can improve both the performance and the stability*. A feedforward controller was designed to validate the second hypothesis. This controller provides the target motion estimation [2]. Fig. 6 depicts a block diagram with a transfer function

$$\frac{{}^{i}X(z)}{X_t(z)} = \frac{V(z)(1 - G_p(z)D_F(z))}{1 + V(z)G_p(z)D(z)} \tag{1}$$

where ${}^{i}X(z)$ is the position of the target in the image, $X_t(z)$ is the target position, $V(z)$ and $G_p(z)$ are the transfer functions for the vision system and PTU, respectively. The previous and actual positions of the target in the image plane are used to predict its position and velocity one step ahead. Based on this, the feedforward controller will compute the camera velocity for the next step. $D_F(z) = G_F(z)G(z)$ represents the transfer function of the filter combined with the feedforward controller. $D(z)$ is the transfer function for the feedback controller.

If $D_F(z) = G_p^{-1}(z)$, the tracking error will be zero, but this requires knowledge of the target position that is not directly measurable. Consequently, the target position and velocity are estimated. For a horizontally translating target, its centroid in the image plane is given by the relative angle between the camera and the target

$$^{i}X(z) = K_{\text{lens}}(X_t(z) - X_r(z)) \tag{2}$$

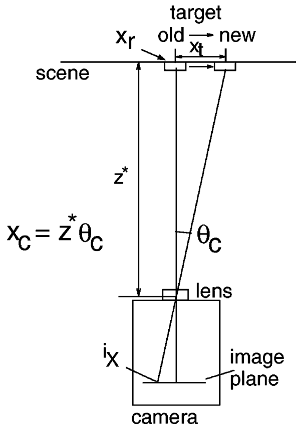


Fig. 5. Schematic of camera scene.

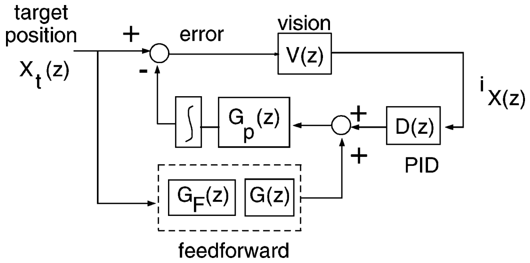


Fig. 6. Feedforward controller with a feedback compensation.

where $^iX(z)$ and $X_t(z)$ are the target positions in the image plane and world frame, respectively. $X_r(z)$ is the position of the point that is in the camera's focus (due to the booming and camera rotation) and K_{lens} is the lens zoom value. The target position prediction can be obtained from the boom and the PTU, as seen in Fig. 5. Rearranging this equation yields

$$\hat{X}_t(z) = \frac{^i\tilde{X}(z)}{K_{\text{lens}}} + X_r(z) \tag{3}$$

where $\hat{X}_t$ is the predicted target position.

C. The α–β–γ Filter

Predicting the target velocity requires a tracking filter. Oftentimes, a Kalman filter is used, but is computationally expensive. Since Kalman gains often converge to constants, a simpler α–β–γ tracking filter can be employed that tracks both position and velocity without steady-state errors [14], [15].

Tracking involves a two-step process. The first step is to predict the target position and velocity

$$x_p(k+1) = x_s(k) + Tv_s(k) + T^2 a_s(k)/2 \tag{4}$$

$$v_p(k+1) = v_s(k) + Ta_s(k) \tag{5}$$

where T is the sample time and $x_p(k+1)$ and $v_p(k+1)$ are the predictions for the position and velocity at iteration $k+1$, respectively. The variables $x_s(k)$, $v_s(k)$, and $a_s(s)$ are the corrected (smoothed) values of iteration k for position, velocity, and acceleration, respectively.

The second step is to make corrections

$$x_s(k) = x_p(k) + \alpha(x_o(k) - x_p(k)) \tag{6}$$

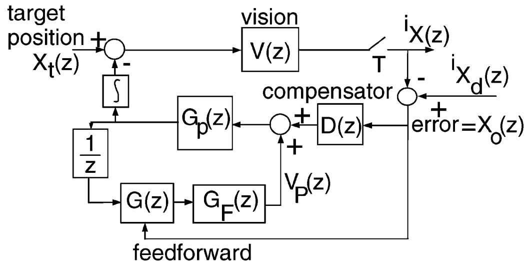


Fig. 7. Feedforward controller with a feedback compensation as it was implemented.

$$v_s(k) = v_p(k) + (\beta/T)(x_o(k) - x_p(k)) \tag{7}$$

$$a_s(k) = a_p(k-1) + (\gamma/2T^2)(x_o(k) - x_p(k)) \tag{8}$$

where $x_o(k)$ is the observed (sampled) position at iteration k. The appropriate selection of gains α, β, and γ will determine the performance and stability of the filter [15].

The α–β–γ filter was implemented to predict the target velocity in the image plane with gains set at $\alpha = 0.75$, $\beta = 0.8$, and $\gamma = 0.25$. This velocity was, then, used in the feedforward algorithm, as shown in Fig. 7.

Image processing in the camera system can be modeled as a $1/z$ unit delay that affects the camera position x_r and estimates of the target position. In Fig. 7, the block $G_F(z)$ represents the transfer function of the α–β–γ filter, with the observed position as the input and the predicted velocity as the output. $X_d(z)$ represents the target's desired position in the image plane and its value is 320 pixels. $X_o(z)$ represents the position error in the image plane (in pixels).

The constant K_{lens} converts pixels in the image plane to meters. K_{lens} was assigned a constant value, and it assumes a pinhole camera model that maps the image plane and world coordinates. This constant was experimentally determined by comparing the known lengths in world coordinates to their projections in the camera's image plane.

With the system equipped with the feedforward controller, a couple of experiments were performed. Again, the first was the people-tracking experiment. A subject was asked to walk back and forth in the laboratory environment. The operator boomed while the camera tracked the subject. Sequential images from the experiment can be seen in Fig. 8. The first row shows the boom camera view. It can be seen that the system is not in danger of losing the target. The second row shows the operator booming while the third row shows the program working. It can be seen that the target is well detected.

To quantitatively assess the performance, the Mitsubishi robotic arm was instructed to move the target sinusoidally. The camera was instructed to track this target using the proportional as well as the feedforward controller. An operator panned the boom at the same time. Data regarding Mitsubishi motion, booming motion, and tracking error were recorded. The performance is assessed by comparing the tracking error. The setup can be seen in Fig. 3(b).

The experiment was set up in the laboratory. The camera–target distance was 3.15 m. The target dimensions were 8.9 × 8.25 cm^2. The robotic arm moved the target sinusoidally with

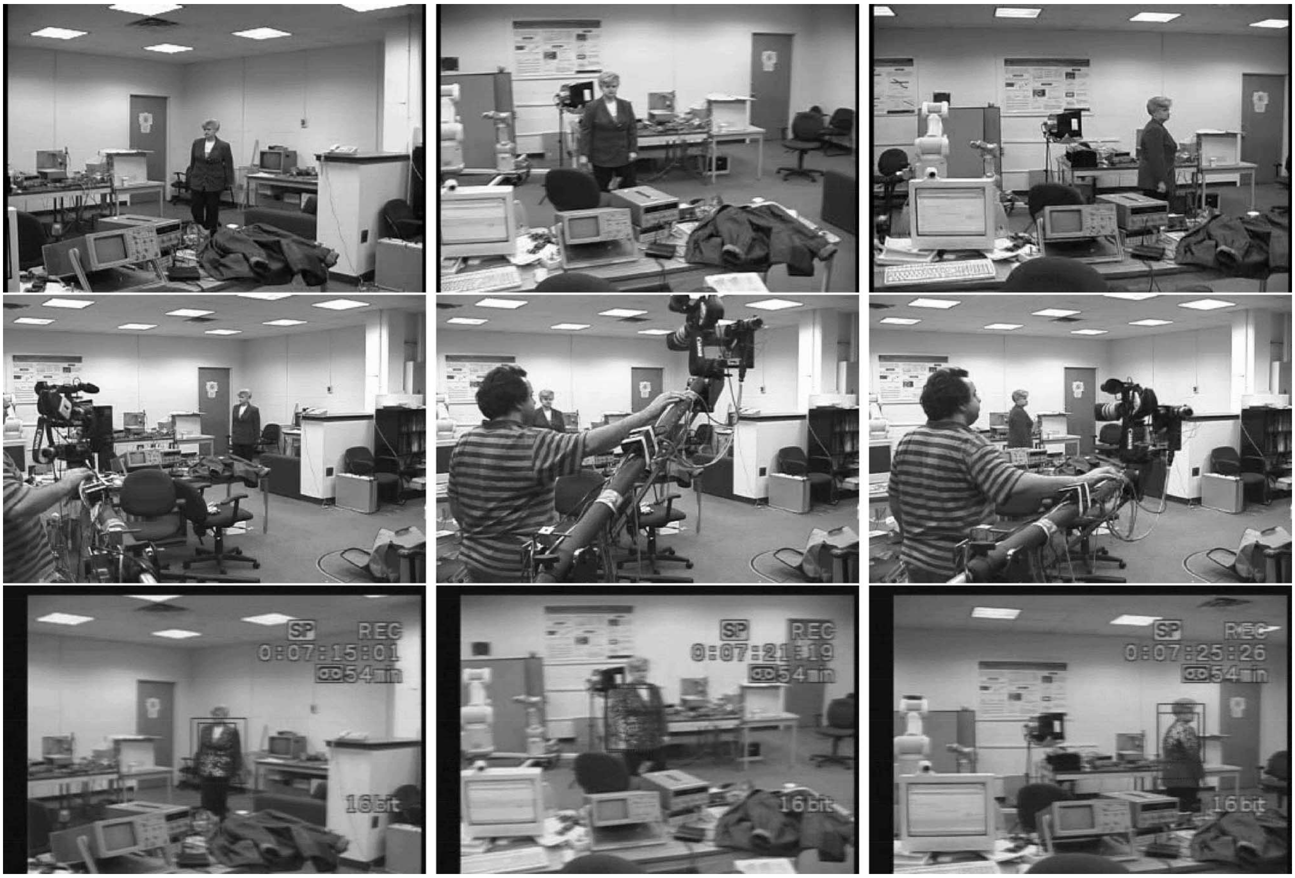


Fig. 8. Three sequential images from videotaping the feedforward controller experiment. Camera field-of-view shows target is tracked *top row*. Boom manually controlled *middle row*. Working program *bottom row*.

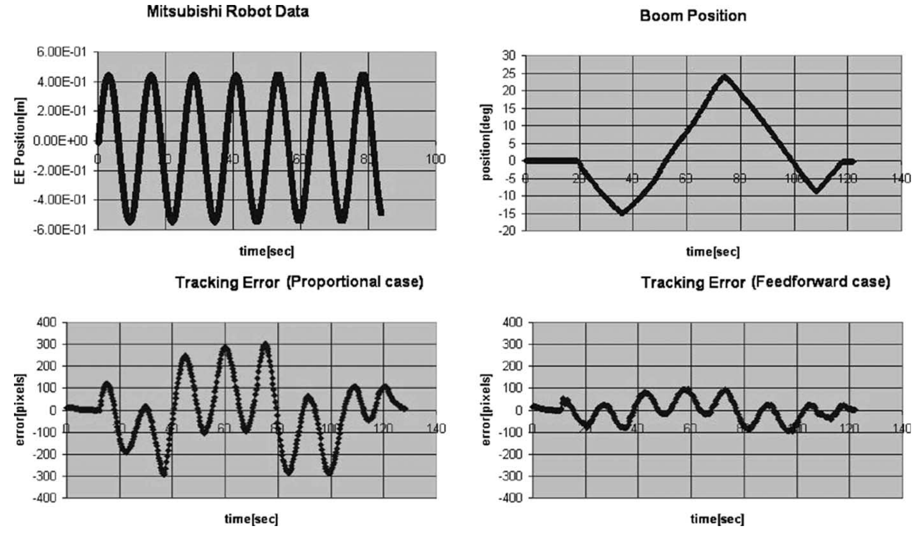


Fig. 9. Tracking errors comparing feedforward and proportional control in *human-in-the-loop visual-servoing*. (*top row*) Target sinusoidal motion and booming. It can be seen that the operator moved the boom real slow (about 1°/s). (*bottom row*) Tracking error using a proportional control (*left-hand side*) and a feedforward control (*right-hand side*). The image dimensions are 640×480 pixels.

a frequency of about 0.08 Hz and a magnitude of 0.5 m. CONDENSATION algorithm was employed for the target detection. As this algorithm is noisy, the target image should be kept small. The target dimensions in the image plane were 34×32 pixels. While both the controllers attempted to track, the boom was manually moved from $-15°$ to $+25°$. The plots can be seen in Fig. 9.

In the top row, the target motion and the booming plot (both versus time) can be seen. The operator moved the boom really slow (approximately 1°/s). This booming rate was used because

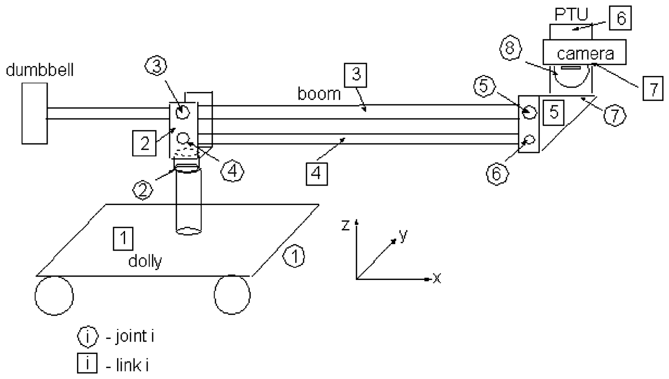


Fig. 10. Number assigned to every link and joint. *Circled numbers* represent joints while numbers in *rectangles* represent link.

of the proportional controller. The tracking errors are shown in the bottom row. The bottom left image shows the error when using the proportional controller for tracking. The bottom right image shows the error when using the feedforward controller. The peak-to-peak error was about 100 pixels with the feedforward controller, while the proportional controller yielded an error of more than 300 pixels. By comparing the error in the same conditions, the conclusion was that the feedforward controller is "much better" than the proportional controller. Still, considering that the focal length was about 1200 pixels and given the camera–target distance of 3.15 m, 100 pixels represented about 35 cm of error. This value was considered to be too big.

D. Symbolic Model Formulation and Validation

At this point, a model was desired for the boom-camera system. Simulation of new controllers would be much easier once the model was available. With satisfactory simulation results, a suitable controller can be implemented for experiments.

Both the nonlinear mathematical and simulation models of the boom were developed using *Mathematica* and *Tsi ProPac* [17], [18]. The former is in Poincaré equations enabling one to evaluate the properties of the boom and to design either a linear or a nonlinear controller. The latter is in the form of a C-code that can be compiled as an S-function in SIMULINK. Together, these models of the highly involved boom dynamics facilitate the design and testing of the controller before its actual implementation.

The boom, shown in Fig. 10, comprises seven bodies and eight joints. The bodies and joints are denoted by boxes and circles, respectively. The DOFs of various joints are detailed in Table I, while the physical data are given in Table II. They give the position or Euler angles of the *joint body* (JB) with respect to the *reference body* (RB). At the origin, which corresponds to a stable equilibrium, the boom and the camera are perfectly aligned. One characteristic of the boom is that it always keeps the camera's base parallel to the floor. This is because bodies 3 and 4 are part of a four-bar linkage. There are two constraints for the system that can be seen in (9)

$$\begin{aligned}\theta_{bb1} - \theta_{bt1} &= 0 \\ \theta_{bt1} + \theta_{bt2} &= 0. \end{aligned} \tag{9}$$

TABLE I
TYPES OF MOTION FOR LINKS

Joint #	RB	JB	x	y	R_x	R_y	R_z
1		1	x	y			
2	1	2					ψ_b
3	2	3				θ_{bt1}	
4	2	4				θ_{bb1}	
5	3	5				θ_{bt2}	
6	4	5				θ_{bb2}	
7	5	6					ψ_c
8	6	7				θ_c	

TABLE II
BOOM LINKS, MASSES, AND MOMENTS OF INERTIA

Object	Mass kg	Moment of Inertia $kg\ m^2$
Dolly (link1)	25	$I_{xx} = 2.48,$ $I_{yy} = 0.97,$ $I_{zz} = 3.465$
Link 2	0.6254	$I_{xx} = 0.000907$ $I_{yy} = 0.000907,$ $I_{zz} = 0.00181$
Boom (link 3)	29.5	$I_{xx} = 0$ $I_{yy} = 16.904,$ $I_{zz} = 16.904$
Link 4	0.879	$I_{xx} = 0$ $I_{yy} = 0.02379,$ $I_{zz} = 0.02379$
Link 5	3.624	$I_{xx} = 0.08204$ $I_{yy} = 0.00119,$ $I_{zz} = 0.00701$
PTU (link 6)	12.684	$I_{xx} = 0.276$ $I_{yy} = 0.234,$ $I_{zz} = 0.0690$
Camera (link 7)	0.185	$I_{xx} = 0$ $I_{yy} = 1.33\dot{\imath}0^{-5},$ $I_{zz} = 1.33\dot{\imath}0^{-5}$

The inputs acting on the system are the torques $Q1$ (about y) and $Q2$ (about z) exerted by the operator, and the torques $Q3$ and $Q4$ applied by the pan and tilt motors of the camera, that is, $\mathbf{u} = \{Q1, Q2, Q3, Q4\}$. The dumbbell at the end of body 3 is pushed to facilitate the target tracking with the camera. In this analysis, it is assumed that the operator does not move the cart, although it is straightforward to incorporate that as well. The pan and tilt motors correspond to the rotations ψ_c and θ_c, respectively.

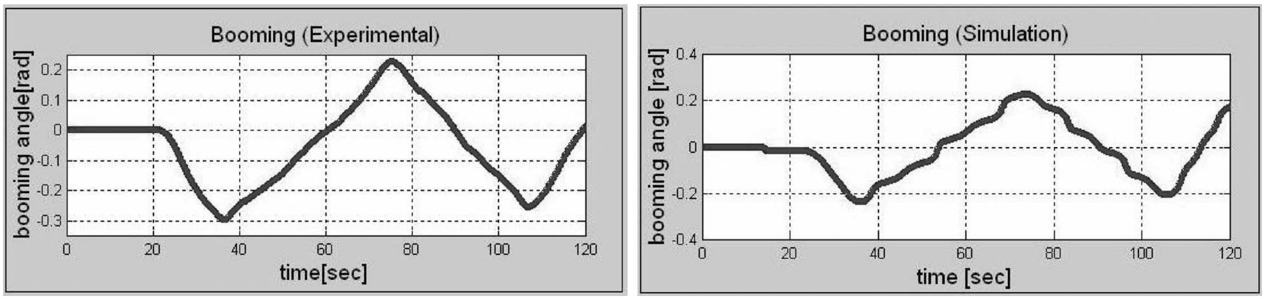


Fig. 11. Booming. Experiments (*left*) and simulation (*right*).

The model can be obtained in the form of Poincaré equations (see [17] and [18] for details)

$$\begin{aligned}&\dot{\mathbf{q}} = V(\mathbf{q})\mathbf{p}\\&M(\mathbf{q})\dot{\mathbf{p}} + C(\mathbf{q})\mathbf{p} + \mathbf{Q}(\mathbf{p},\mathbf{q},\mathbf{u}) = \mathbf{0}.\end{aligned} \tag{10}$$

The generalized coordinate vector $\mathbf{q}$ (see Table I for notation) is given by

$$\mathbf{q} = [x, y, \psi_b, \theta_{bt1}, \theta_{bt2}, \theta_{bb1}, \theta_{bb2}, \psi_c, \theta_c]^T.$$

Vector $\mathbf{p}$ is the 7×1 vector of quasi-velocities given by

$$\mathbf{p} = [\Omega_{yc}, \Omega_{zc}, \Omega_{bb2}, \Omega_{bt2}, \Omega_{zb}, v_y, v_x]^T.$$

They are quasi-velocities associated with joints 8, 7, 6, 5, 2, and a double-joint 1, respectively. The first set of equations are the kinematics and the second are the dynamics of the system.

E. Model Validation

The simulation model is generated as a C-file that can be compiled using any standard C-compiler. The MATLAB function *mex* is used to compile it as a dll file, which defines an S-function in SIMULINK. To ascertain the fidelity of the model, the experimental results in [19] were simulated in SIMULINK. The experimental setup is depicted in Fig. 3(b). The booming angles, the target motion, and the errors are shown in Figs. 11 and 12, respectively. In spite of the fact that the dynamics of the wheels and the friction in the joints are neglected, the experimental and simulated results show fairly good agreement.

F. Output Tracking Regulation Controller (OTR)

The target position in the image plane is a time-dependent function. By applying the Fourier theory, such a function can be expressed as a sum of sinusoids with decaying magnitudes and increasing frequencies. If the controller can be fine-tuned to ensure lower frequency sinusoids tracking, then the tracking error will be acceptable. The last of our hypotheses was that *adding such a controller to our system will improve the performance by reducing the error to* ±50 *pixels (*50%*) in case of the Mitsubishi robot experiment*.

This paper investigated the effectiveness and advantages of the controller implemented as a regulator with disturbance-rejection properties. This approach guarantees the regulation of the desired variables, while simultaneously stabilizing the system and rejecting the exogenous disturbances. As a first step, a linear controller was designed to regulate only the pan motion. Its structure can be seen in Fig. 13.

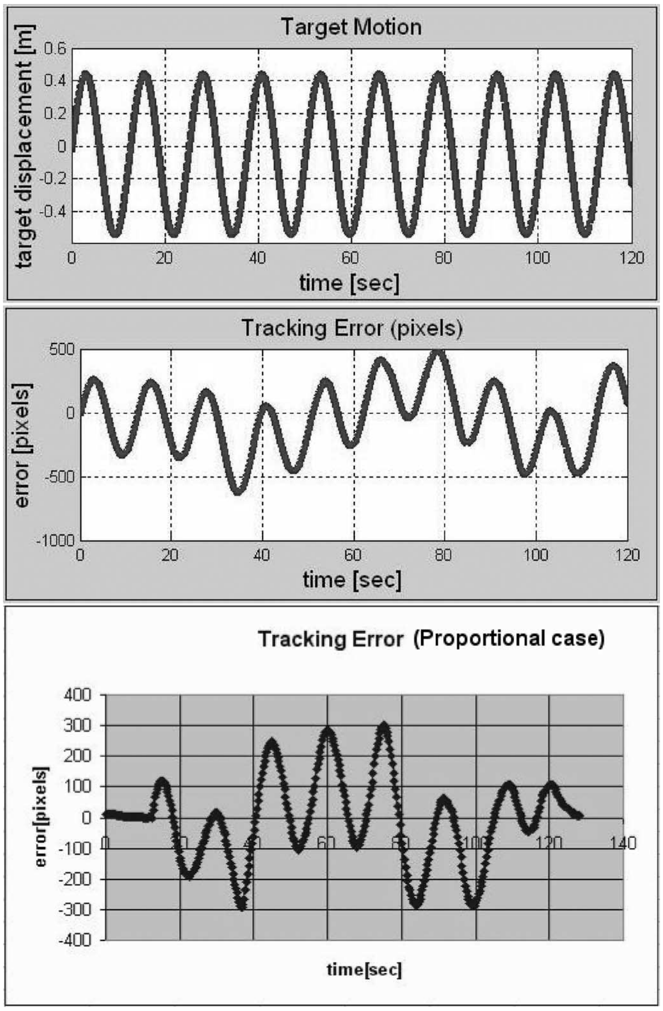


Fig. 12. Target motion (*top*). Simulation errors and experimental error in pixels.

The linearized equations are recast as

$$\dot{x} = Ax + Pw + Bu, \quad \dot{w} = Sx, \quad e = Cx + Qw. \tag{11}$$

The regulator problem is solvable if and only if Π and Γ satisfy the linear matrix equations (12) [16], [20]:

$$\begin{aligned}\Pi S &= A\Pi + P + B\Gamma\\0 &= C\Pi + Q\end{aligned} \tag{12}$$

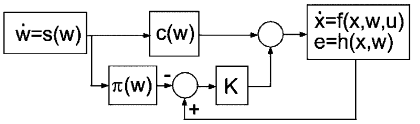


Fig. 13. Output tracking regulation controller as it was implemented.

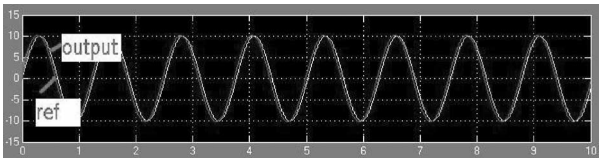


Fig. 14. Reference (5 rad/s) as well as the output of the PTU using the new controller.

A regulating control can, then, be constructed as

$$u = \Gamma w + K(x - \Pi w) \tag{13}$$

where K is chosen so that the matrix $(A + BK)$ has the desired eigenvalues. These eigenvalues determine the quality of the response.

The PTU motor model has the transfer function

$$\frac{\theta(s)}{V_a(s)} = \frac{0.01175}{1.3s^2 + 32s} \tag{14}$$

where the output is the camera angle. In this case, the state space description of the system is given by matrices A, B, and C

$$A = \begin{vmatrix} -24.61 & 0 \\ 1 & 0 \end{vmatrix}$$

$$B = \begin{vmatrix} 0.0088 \\ 0 \end{vmatrix}$$

and

$$C = |0 \quad 1|.$$

From (12)

$$\Pi = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}$$

while

$$\Gamma = |-113.6 \quad 2796.6|.$$

The matrix K was

$$K = |-10000 \quad -380|.$$

G. Simulation and Experiments Using Output Tracking Regulation Controller

Prior to the implementation experiment, a new controller was simulated using Matlab Simulink. Sinusoidal reference signals corresponding to 1, 5, and 10 rad/s were applied to the controller (in simulation). Both the reference and the output of the system were plotted on the same axes frame. The plots corresponding to the 5 rad/s input can be seen in Fig. 14.

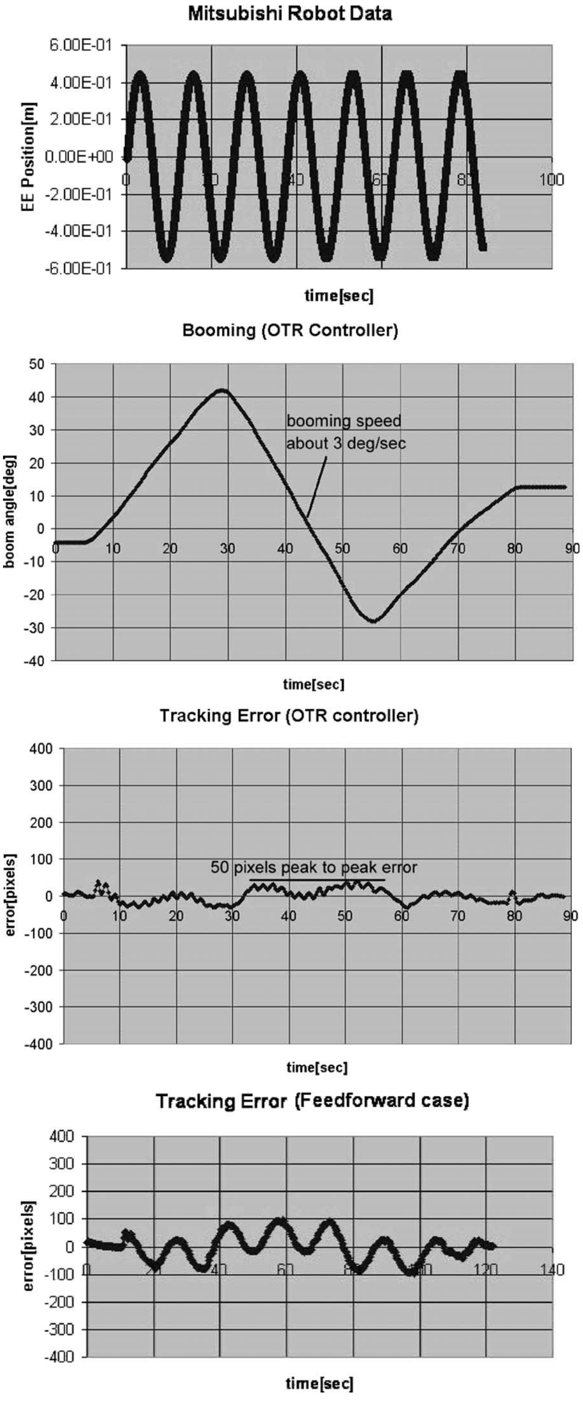


Fig. 15. Mitsubishi experiment using the OTR controller. The first figure shows the moving target. The second figure shows the boom motion. The third figure shows the tracking error in case of the output tracking controller. The fourth figure shows the error using the feeedforward controller. It can be seen that by using the OTR controller, the error is less then ±50 pixels. This value reflects a gain in performance of 50%.

After the implementation experiment, several experiments were performed using this controller. First, the controller was tested with the Mitsubishi robotic arm for a comparison of the performance of the feedforward and proportional controllers. In the second experiment, the system attempted to track a ball kicked by two players.

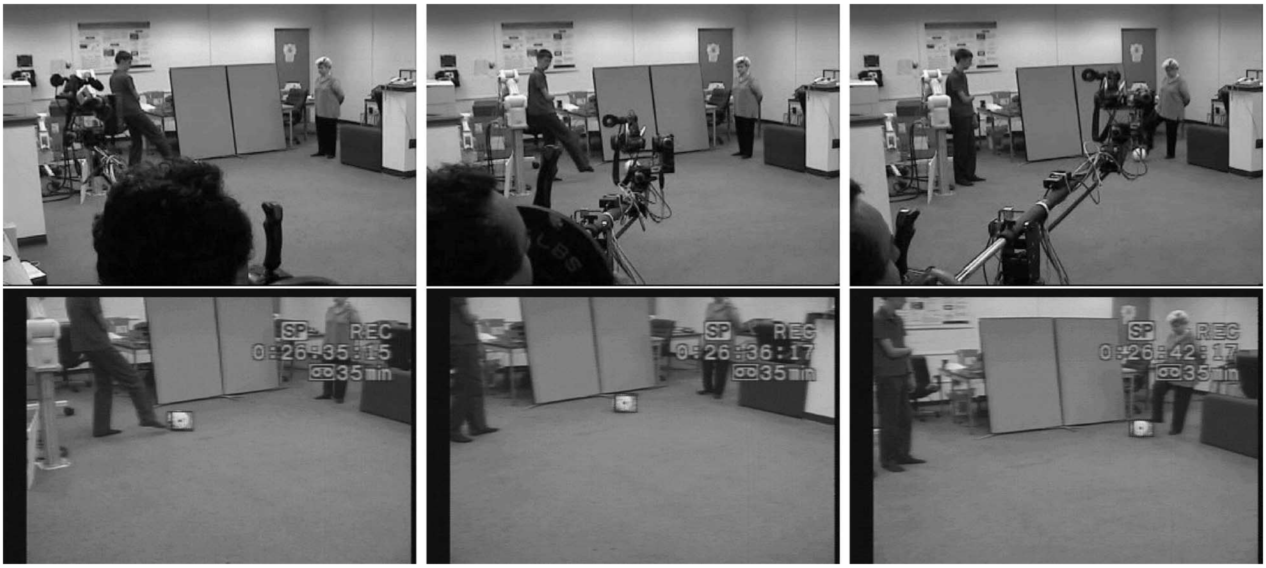


Fig. 16. Ball-tracking experiment. Operator booming and camera point of view *top row*. Program working *bottom row*.

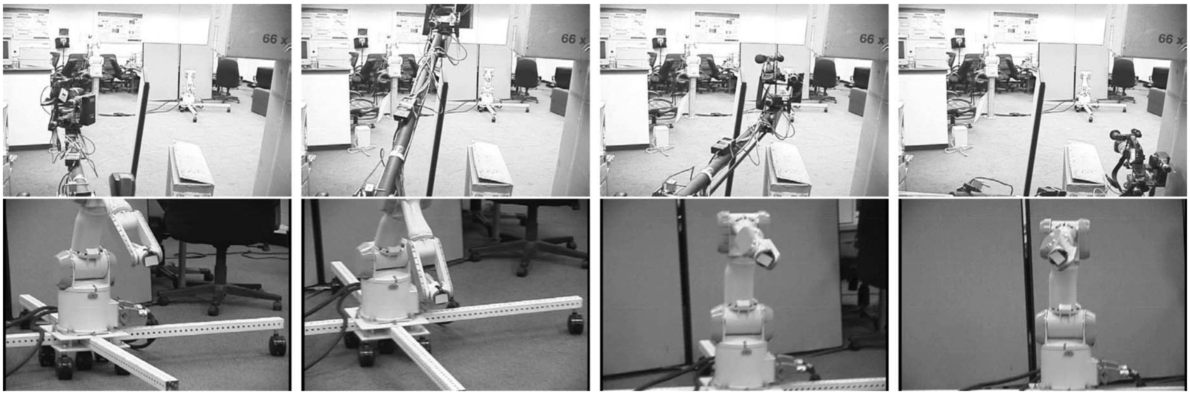


Fig. 17. Unexperienced operator with the vision system.

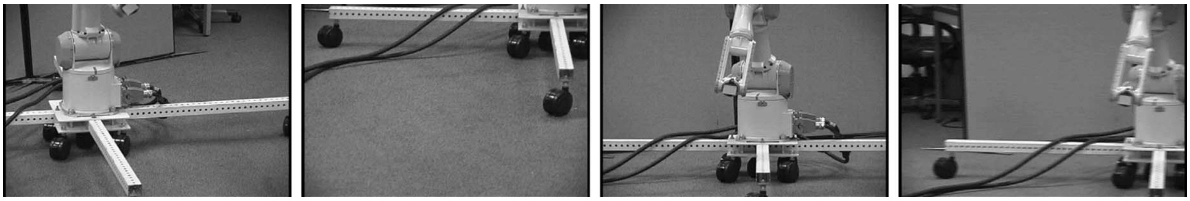


Fig. 18. Unexperienced operator without the vision system. The target was lost eight times. The pictures were taken when the camera was in positions of interest shown in Fig. 3(c) and the top row of Fig. 17. Because the target was lost, the tracking error curve has no relevance.

In the first experiment, the robotic arm was instructed to sinusoidally move the target with the same frequency and magnitude as in the case of the feedforward controller. The camera tracked the target while the operator boomed. The booming data and the tracking error were recorded. The plots can be seen in Fig. 15. In this figure, the top plot represents the target motion while the second plot shows the operator booming. It can be seen that the booming takes place with a frequency of about 3°/s (when comparing the proportional and the feedforward controllers, the booming speed was about 1°/s). The third plot is the horizontal error when using the OTR controller (provided for comparison). It can be seen that when the OTR controller is used, the error becomes ± 50 pixels (half of the value obtained using only the feedforward controller).

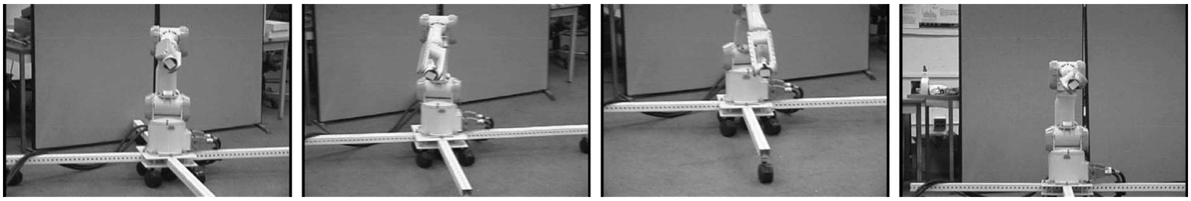


Fig. 19. Experienced operator with the vision system. Again, the target is never lost. The pictures were taken when the camera was in positions of interest shown in Fig. 3(c) and the top row of Fig. 17.

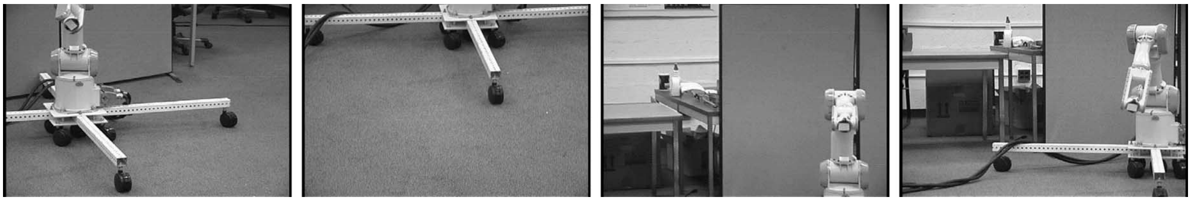


Fig. 20. Experienced operator without the vision system. The target was lost five times. Because the target was lost, the tracking error curve has no relevance. The pictures were taken when the camera was in positions of interest shown in Fig. 3(c) and the top row of Fig. 17.

H. Ball-Tracking Experiment

Since the tracking error reduced when the robotic arm was used, it was interesting to see its behavior in a more natural environment. This time the task was to track a ball moving between two players. The experiment was set up in the laboratory and videotaped using three cameras. Sequential pictures can be seen in Fig. 16. The top row shows the operator booming as the camera tracks the ball. The bottom row shows the boom camera point-of-view. It can be seen that the target is precisely detected and tracked. Despite its "not so scientific nature" (no data was recorded), this experiment highlighted one challenge. If the ball is kicked softly, the image processing algorithm will successfully detect it and the camera is able to track it. If the ball is kicked harder, the camera fails to track it. This means that at a frequency of 3–4 Hz (the total time to process a frame and compute the controller outputs was around 340 ms), the target acceleration is limited to small values. This particular challenge was not revealed by experiments involving the robotic arm.

IV. Human Versus Human–Vision Control: A Comparison

It was interesting to determine if and how this system is able to help the operator. To assess the increase in performance due to the vision system, an experiment was set up. Again, the Mitsubishi robot was used. Its end-effector moved the target on a trajectory corresponding to a figure "8" for 60 s. An experienced operator and a beginner were asked to handle the boom with and without the help of vision. When vision was not used, the operator manually controlled the camera using a joystick.

A booming path was set up in an attempt to increase the experiment repeatability [shown in Fig. 3(c)]. Each operator boomed two times: first, when using the vision system, and second, when manually manipulating the camera using a joystick. The objective was to keep the target in the camera's field-of-view while both the target and boom move. Several positions of interest were marked along the booming path using numbers [see Fig. 3(c)]. Tracking error was recorded while using vision. Under the manual manipulation experiment, both the operators lost the target. When the target was outside the image plane, the image processing algorithm focused on other objects in the image. Because of this, the tracking error had no relevance during manual manipulation.

Sequential images from the experiment can be seen in Figs. 17–20. The images are taken when the camera was in one of the positions marked in Fig. 3(c).

In the case of using the vision system, the target was never lost (Figs. 17 and 19). Moreover, the output regulation controller (which is implemented for the pan motion) maintains the target very close to the image center.

In the case of manual tracking (Figs. 18 and 20), the operator has to manipulate the boom as well as the camera. It can be seen that both the operators have moments when the target is lost. In case of an unexperienced operator without vision, the booming took longer than the motion of the robotic arm simply because there are more DOFs to be controlled simultaneously. The unexperienced operator lost the target eight times. The experienced operator was able to finish booming within 60 s, but he lost the target five times. Because the program focuses on something else in the absence of the target, the data regarding the tracking error is not relevant when the target was lost. The target was never lost when using vision.

The absolute value of the error in both the cases is shown in Fig. 21. One can see that the values are in the same range. This means that visual servoing helps the novice operator to obtain performance similar to that of the expert.

V. Conclusion and Future Work

This paper integrates visual-servoing for augmenting the tracking performance of camera teleoperators. By reducing the

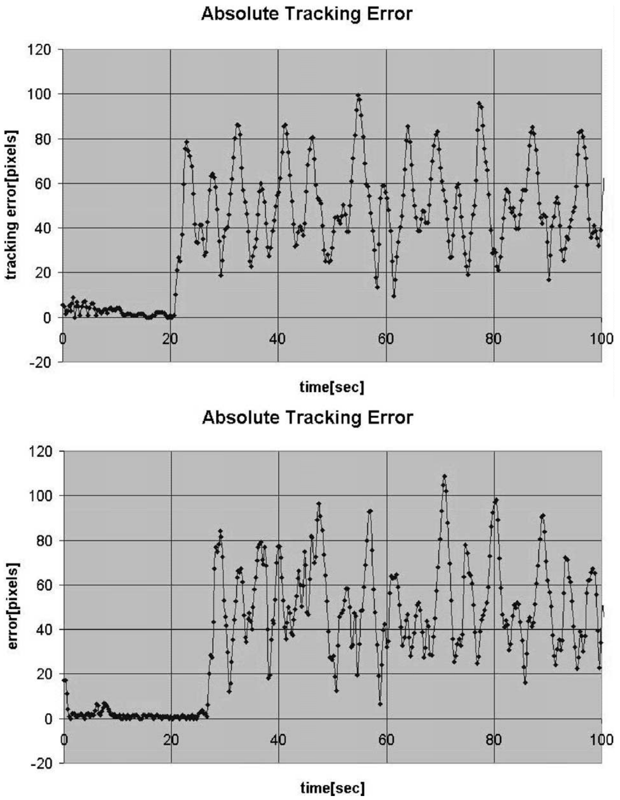


Fig. 21. Tracking error. Experienced operator with vision (*top*). Unexperienced operator using vision (*bottom*). Booming path was restricted. It can be seen that there are no significant differences between these two plots.

number of DOFs that need to be manually manipulated, the operator can concentrate on coarse camera motion. Using a broadcast boom system as an experimental platform, the dynamics of the boom PTU were derived and validated experimentally. A new controller was added to the feedforward scheme and tested experimentally. The performance of the new control law was assessed by comparing the use of the vision system versus manual tracking for both an experienced and an unexperienced operator. The addition of the OTR controller to the feedforward scheme yielded lower errors. The use of the vision system helps the operator (the target was precisely detected and tracked). This suggests that by using the vision system, even an unexperienced operator can achieve a performance similar to that of a skilled operator. Also, there are situations when vision is helpful for a skilled operator.

Still, there are situations when the target detection and tracking fail. A mechanism to detect such situations and alert the operator is desirable. When such situations occur, the camera can be programmed to automatically move to a particular position. The ball-tracking experiment proves to be successful if the ball is hit softly. When the ball is hit harder, the image processing fails to detect it, and tracking fails. However, there is no proof that controllers would be able to track a harder-hit ball if image processing did not fail.

Another case that is not investigated in this paper is occlusion. Such experiments were not performed. They should be studied in future work. Because the focus of this research was the control part, the case of appearance of similar targets in the image plane was not studied. The effect of the image noise, when the camera moves quickly, was also not studied.

Future work will also have to focus on increasing tracking performance. If this tracking system is to be used in sports broadcasting, it will have to be able to track objects moving with higher acceleration. The sampling time (which now corresponds to 3–4 Hz) will have to decrease (perhaps one way to achieve this is to use a faster computer). When tracking sports events (football, soccer, etc.), when the target moves with high accelerations and its dimensions vary in the image, a target estimation mechanism will be desirable. Such a mechanism would record ball positions and estimate its trajectory. Once the estimation is done, this mechanism would command the camera to move to the estimated "landing" position and try to re-acquire the ball. Combining this mechanism with zooming in and out would allow tracking of faster objects.

References

[1] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. Robot. Autom.*, vol. 12, no. 5, pp. 651–670, Oct. 1996.

[2] P. I. Corke and M. C. Good, "Dynamic effects in visual closed-loop systems," *IEEE Trans. Robot. Autom.*, vol. 12, no. 5, pp. 671–683, Oct. 1996.

[3] J. Hill and W. T. Park, "Real time control of a robot by visual feedback in assembling tasks," *Pattern Recognit.*, vol. 5, pp. 99–108, 1973.

[4] P. Y. Oh and P. K. Allen, "Visual servoing by partitioning degrees of freedom," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 1–17, Feb. 2001.

[5] P. Y. Oh, "Biologically inspired visual-servoing using a macro/micro actuator approach," presented at the Int. Conf. Imaging Sci., Syst. Technol. (CISST), Las Vegas, CA, Jun. 2002.

[6] A. C. Sanderson and L. E. Weiss, "Image-based visual servo control using relational graph error signals," *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 1074–1077, 1980.

[7] R. Stanciu and P. Y. Oh, "Designing visually servoed tracking to augment camera teleoperators," in *Proc. IEEE Int. Conf. Intell. Robots Syst. (IROS)*, vol. 1, Lausanne, Switzerland, 2002, pp. 342–347.

[8] ——, "Human-in-the-loop visually servoed tracking," in *Proc. Int. Conf. Comput., Commun. Control Technol. (CCCT)*, vol. 5, Orlando, FL, Jul. 2003, pp. 318–323.

[9] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Visual tracking of a moving target by a camera mounted on a robot: A combination of vision and control," *IEEE Trans. Robot. Autom.*, vol. 9, no. 1, pp. 14–35, Feb. 1993.

[10] T. B. Sheridan and W. R. Ferell. (1963). Remote manipulative control with transmission delay. *IEEE Trans. Human Factors in Electronics*. [Online]. HFE-4, pp. 25–29. Available: http://citeseer.ist.psu.edu/cs?cs=1&q=W.+R.+Ferrell&submit=Citations&co=Citations&cm=50&cf=Any&ao=Citations&am=20&af=Any

[11] N. Ferrier, "Achieving a Fitts law relationship for visual guided reaching," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Bombay, India, Jan. 1998, pp. 903–910.

[12] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *J. Exp. Psychol.*, vol. 47, no. 6, pp. 381–391, 1954.

[13] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.

[14] P. R. Kalata and K. M. Murphy, "α–β target tracking with track rate variations," in *Proc. 29th Southeastern Symp. Syst. Theory*, Mar. 1997, pp. 70–74.

[15] D. Tenne and T. Singh, "Optimal design of α–β–(γ) filters," in *Proc. Am. Control Conf.*, 2000, vol. 6, Jun. 2000, pp. 4348–4352.

[16] A. Isidori, *Nonlinear Control Systems*, 3rd ed. New York: Springer-Verlag, 1995.

[17] H. G. Kwatny and G. L. Blankenship, *Nonlinear Control and Analytical Mechanics: A Computational Approach*. Boston, MA, Birkhauser, 2000.

[18] ——, "Symbolic construction of models for multibody dynamics," *IEEE Trans. Robot. Autom.*, vol. 11, no. 2, pp. 271–281, Apr. 1995.

[19] R. Stanciu and P. Y. Oh, "Feedforward control for human-in-the-loop camera systems," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, vol. 1, New Orleans, LA, Apr. 2004, pp. 1–6.

[20] H. G. Kwatny and K. C. Kalnitsky, "On alternative methodologies for the design of robust linear multivariable regulators," *IEEE Trans. Autom. Control*, vol. AC-23, no. 5, pp. 930–933, Oct. 1978.

Rares Stanciu, was born in Alba Iulia, Romania. He received the B.S. and M.S. degrees in electrical engineering from "Politehnica" University of Timisoara in 1994 and 1995, respectively, the second B.S. degree in electronics and telecommunication from the same university and the Ph.D. degree from Drexel University in 2004.

Paul Y. Oh, photograph and biography not available at the time of publication.