# DynaTracker: Target Tracking in Active Video Surveillance Systems

Prithwijit Guha, Dibyendu Palai, Dip Goswami
Dept. of Electrical Engineering
Indian Institute of Technology, Kanpur
Kanpur - 208016, UP, India
Email: {pguha,palai,dipg}@iitk.ac.in

Amitabha Mukerjee
Dept. of Computer Sc. & Engg.
Indian Institute of Technology, Kanpur
Kanpur - 208016, UP, India
Email: amit@cse.iitk.ac.in

***Abstract*— Active video surveillance systems provide challenging research issues in the interface of computer vision, pattern recognition and control system analysis. A significant part of such systems is devoted toward active camera control for efficient target tracking. DynaTracker is a pan-tilt device based active camera system for maintaining continuous track of the moving target, while keeping the same at a pre-specified region (typically, the center) of the image. The significant contributions in this work are the use of mean-shift algorithm for visual tracking and the derivation of the error dynamics for a proportional-integral control action. The stability analysis and optimal controller gain selections are performed from the simulation studies of the derived error dynamics. Simulation predictions are also validated from the results of practical experimentations. The present implementation of DynaTracker performs on a standard Pentium IV PC at an average speed of 10 frames per second while operating on color images of 320x240 resolution.**

## I. Introduction

Automated video surveillance deals with real time observation of people or vehicles in busy or restricted environments leading to tracking and activity analysis of the subjects in the field of view. Potential applications range from monitoring in car parking lots to military surveillance systems. A video surveillance system typically consists of the stages of (intrusion) detection, (intruder) recognition and trajectory tracking for generating high level (scene) activity descriptions; majority of such systems being based on a single fixed camera [1], [2], [3], [4].

In an active camera based system, the sensor is typically mounted on a dynamic platform, ranging from pan-tilt devices to robotic arms. Thus, an active monocular video surveillance system achieves wider coverage as compared to the ones with single fixed vision sensor. More so, in such cases the system can intelligently decide its field of view and can have a better look at the region of interest, where either an intrusion or some unusual scene activity is detected. Trivedi et al. [5] have used active camera networks for monitoring intelligent environments. A few researchers have proposed the use of optical flow based techniques combined with discrete Kalman filters for active camera control [6], [7]. Murray et al. [8], on the other hand, have used robust motion detection algorithms for active camera based target tracking.

This paper reports a significant part of our ongoing work on active video surveillance systems. In this work, we restrict ourselves to real-time vision based tracking while keeping the subject in a pre-specified region of the image. Generally, the task is to keep the target at the center of the grabbed image, the camera being attached to a pan-tilt positioning system. As the subject starts moving in the real world, its position in the grabbed image is reported in subsequent frames through a color feature based mean-shift tracking algorithm [9]. The image position error is processed by a proportional-integral controller [10] and the imaging device is re-positioned accordingly to place the target in the pre-specified image region.

This paper presents our work through the following sections. The system overview with the functional block diagram is explained in section II. In section III, we model the proposed system from a control systems viewpoint and derive its error dynamics. Section IV discusses the vision based tracking algorithm for estimating the current position of the target in the image. The experimental results and implementation issues are reported in section V. Finally, we conclude in section VI and discuss the future extensions of the present work.

## II. DynaTracker: An Overview

Consider an active monitoring system which is capable of detecting scene anomalies (an intrusion or a sudden rise in spatio-temporal variance in the image region etc.) and can re-orient itself toward the direction of interest for having a better look at the situation. Such an application combines two distinct challenges - firstly, the intelligent analysis of scene activity and secondly, the orientation control of the vision sensor based on image position errors. This paper focuses on the second problem, where the main objective is to track a target in its field of view, while keeping the same in a pre-specified image region.

Figure 1 shows the functional block diagram of the proposed system. The user is expected to specify a set point to the system, typically the center of the image plane with a rectangular dead band window around. As the target in the real world starts moving, its position in the image also changes. The new position is however retrieved by the color feature based mean-shift tracking algorithm. A proportional-integral control action is then executed based on the difference between new position and the set point, which re-orients the pan-tilt device

0-7803-9177-2/05/$20.00/©2005 IEEE 

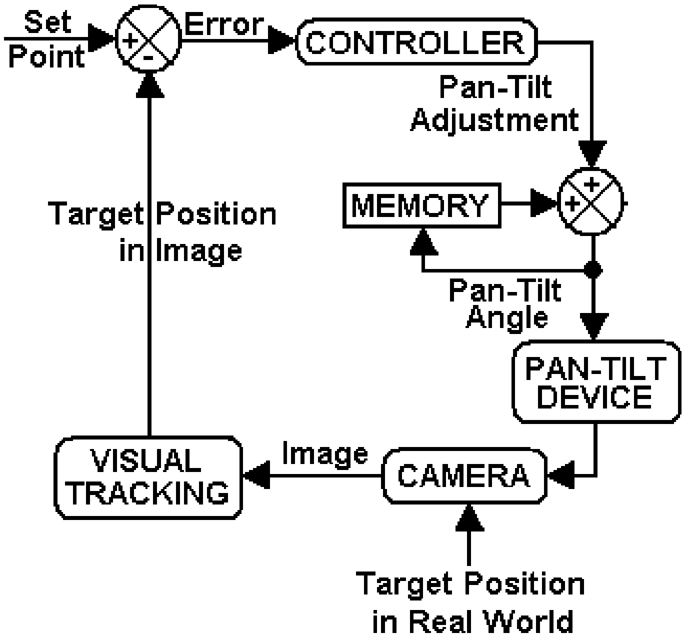


Fig. 1. The Functional Block Diagram of DynaTracker

so that the image position error is minimized. The system functions in a loop, executing the above mentioned procedures and keeps track of the target in the real world. However, for optimal performance of the system, the controller parameters need to be chosen properly ensuring both stability and fast convergence of the error sequence. In section III, we derive the error dynamics of the system and use the same for choosing the controller parameters from the simulation studies presented in section V.

## III. The Error Dynamics

In this section, we derive the error dynamics of the system through the modeling of image formation and control action. In figure 2, we redraw the functional block diagram (figure 1) from a typical control systems viewpoint, consisting of the controller, plant and a feedback loop. Here, we consider the case of panning, where only the image position in the horizontal direction ($X$-axis) undergoes a substantial change. Let the image position (abscissa) at the $t^{th}$ instant be $V_x(t)$, as obtained by tracking. The error $E_x(t)$ is computed by subtracting $V_x(t)$ from the command input $U_x(t)$ at the $t^{th}$ instant.

$$E_x(t) = U_x(t) - V_x(t) \tag{1}$$

The panning angle adjustment $\Delta\theta(t)$ at this instant is computed as a linear combination of the current error and the error derivative, which can be mathematically expressed as,

$$\frac{d\theta(t)}{dt} = K_I E_x(t) + K_P \frac{dE_x(t)}{dt} \tag{2}$$

This can be alternatively written as,

$$\theta(t) = K_I \int_{-\infty}^{t} E_x(t)dt + K_P E_x(t) \tag{3}$$

Equation 3 suggests a proportional-integral control action [10], where $K_P$ and $K_I$ are the parameters of the controller.

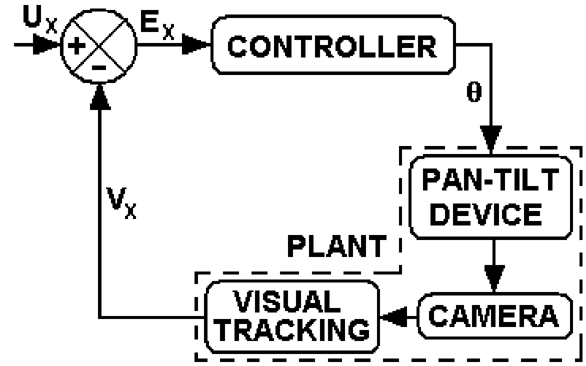


Fig. 2. Block Diagram of DynaTracker from the Control Systems approach

The procedure of image formation under a panning operation is emphasized in figure 3. Consider the case of panning (rotation about $Y$-axis) by an angle $\theta(t)$. Assuming that the object with global co-ordinate $\overline{W} \equiv (X_G, Y_G, Z_G)$ remains static, its new image co-ordinates $\overline{V(t)} \equiv (V_X(t), V_Y(t))$ can be computed as,

$$\begin{pmatrix} \overline{V(t)} \\ 1 \end{pmatrix} = K R_Y[\theta(t)][I_{3\times3}|0_{3\times1}] \begin{pmatrix} \overline{W} \\ 1 \end{pmatrix} \tag{4}$$

Where, $K$ is the intrinsic camera matrix and $R_Y(\theta(t))$ is the rotational matrix for a rotation of $\theta(t)$ around the $Y$-axis [11]. Here, we assume the intrinsic camera matrix in its canonical form, which reduces $K$ to a $3\times3$ identity matrix($I_{3\times3}$). From equation 4, it can be shown that,

$$V_x(t) = \frac{X_G cos(\theta(t)) + Z_G sin(\theta(t))}{Z_G cos(\theta(t)) - X_G sin(\theta(t))} \tag{5}$$

Moreover, from equation 5, it can be shown that the differential changes in $V_x(t)$ and $\theta(t)$ are related as,

$$\frac{dV_x(t)}{dt} = (1 + V$$

error dynamics from both real system and simulation are presented in section V.

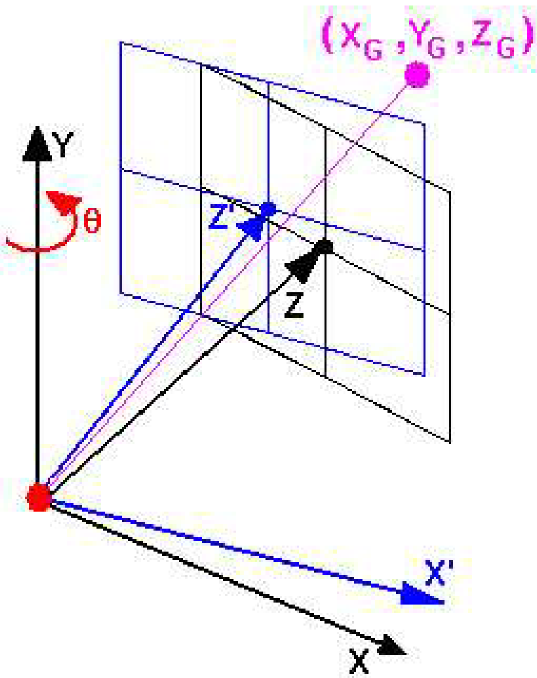


Fig. 3. The image formation under panning operation

## IV. Vision Based Tracking

In this work, we presume the availability of computer vision modules for detection of the region of interest, which would serve in automatic initialization of the target features. The initial target regions may be extracted by template based object recognition modules [12], [13] or background subtraction techniques [14], [15], [16]. The computer vision community has used several algorithms for tracking objects through image sequences, the most noticeable ones being based on Kalman filters [3] and the CONDENSATION algorithm [17]. The Kalman filter based methods fail in many practical applications due to its assumptions of unimodality and linearity in motion and measurement. The CONDENSATION algorithm based approach (also known as particle filtering) overcomes these difficulties, but at the cost of high computational burden.

Recently, Comaniciu et al. [9] have proposed a novel method for tracking non-rigid objects based on the Mean-shift algorithm. They have proposed a color histogram based representation of the target model and a similarity measure based on the Bhattacharya coefficient for comparison of subject and target regions. The proposed algorithm typically assumes a monotonically radially decreasing kernel profile, going from unity magnitude at the center to zero value at the periphery and outside the target region. This provides us with a weighting function that gives maximum importance to the central pixels and assigns minimum belief to the peripheral ones. Such a weight assignment is also useful as the peripheral pixels are less trustworthy due to higher chances of occlusions and belongingness to background. The proposed algorithm has typically assumed the Epanechnikov kernel with an elliptical support over the target region. Thus, if the target region is centered at $C$, then the weight $W_i$ of the pixel $X_i$ will be given by,

$$W_i = \begin{cases} 1 - \|X_i - C\|^2; & \|X_i - C\| \leq 1 \\ 0; & \text{otherwise} \end{cases} \tag{8}$$

These weights are used while constructing the normalized target histogram $H^T$ with $m$ color bins $b_1, \ldots, b_m$. Such a representation suggests that the probability of occurrence of $b_j$ in the target region is $H^T(j)$. Thus, the probability $p_i$ that a particular pixel $X_i$ with color value $Q_i$ belongs to the target region is given by,

$$p_i = H^T(j), Q_i \in b_j \tag{9}$$

The mean-shift iterations start from an initial region centered at $C_0$ and gradually converges to the desired target region centered at $C^*$. Let, $H_k$ be the normalized weighted color histogram computed in the $k^{th}$ iteration from the elliptic region centered at $C_k$. Consider the pixel $X_i(k)$ of color $Q_i$ belonging to this region and define the quantity $G_i(k)$ corresponding to this pixel as,

$$G_i(k) = \sqrt{\frac{H^T(j)}{H_k(j)}}, Q_i \in b_j \tag{10}$$

Thus, the center update rule is given by,

$$C_{k+1} = \frac{\sum_{i=1}^{N_k} X_i(k) G_i(k)}{\sum_{i=1}^{N_k} G_i(k)} \tag{11}$$

where, the current elliptical region centered at $C_k$ consists of $N_k$ pixels and $G_i(k)$ are computed from equation 10. Comaniciu et al. [9] have used the Bhattacharya coefficient $B(H^T, H_k)$ as a measure of comparison of the (normalized) weighted color distribution $H_k$ (at the $k^{th}$ step) and the target histogram $H^T$ and is given by,

$$B(H^T, H_k) = \sum_{j=1}^{m} \sqrt{H^T(j) H_k(j)} \tag{12}$$

This algorithm is proved to converge [9] subject to proper choice of kernel function and is shown to maximize the Bhattacharya coefficient at each step of the iterations. The current value of the Bhattacharya coefficient is used as a termination criteria and algorithm is seen to converge within 2 to 3 iterations in most of the cases. In our case, the present image position of the target is obtained after the convergence of the mean-shift iterations and the final Bhattacharya coefficient reflects the quality of tracking.

## V. Implementation and Results

The present prototype of DynaTracker is implemented with a pan-tilt device consisting of a pair of stepper motors (1.8 degrees/step, interfaced through PC parallel port) and a IEEE1394 web cam. Experimental results are presented for keeping continuous track of a ball held by a person moving around in a room, the camera being continuously re-oriented to keep the ball at the center of the image. DynaTracker was found to perform satisfactorily, in spite of the problems caused by step misses in stepper motors and tracking inaccuracies due to poor image quality of web cams.

Simulation studies are performed for analyzing the stability of DynaTracker by varying the values of $K_P$ and $K_I$ from $0.0$ to $5.0$ in steps of $0.05$ for a step command input of magnitude $10.0 Units$. It was found that the system becomes unstable, if the ratio $\frac{K_I}{K_P}$ exceeds 2.0. The control chart in the $K_P - K_I$ plane is shown in figure 4(a). The system responses are studied for a step input of magnitude $10.0 Units$ for varying values of $K_P$ and $K_I$ in the stable region. The simulation results are shown in figure 4(b)-(e). Considering the peak overshoot, settling time and steady state errors from these graphs, we opt for $K_P = 1.0$ and $K_I = 1.0$. Figure 4(f) shows the simulation results for the image position error in response to a sinusoidal command input with chosen values of $K_P$ and $K_I$. Figure 5 shows the real system responses to a step input of magnitude $50 Pixels$ for varying values of $K_P$ and $K_I$. Figure 5 also witnesses the occurrences of spikes leading to short term errors. This is caused by the phenomenon of occasional step-misses of the stepper motors and can be eliminated by using superior quality motors and/or driver circuitry. However, it is worth observing that the proposed system is well capable of driving back to stability in short time durations in spite of the occasional spikes. In figure 6, we present the results of tracking a ball held by a person moving around in the lab environment. The controller gains are set as $K_P = 1.0$ and $K_I = 1.0$ with the objective of keeping the target at the center of the image with a dead band of $5 Pixels$. The system is found to operate on a Pentium-IV PC at an average speed of 10 frames per second.

## VI. Conclusions

This paper reports a significant part of our ongoing work on active camera based surveillance systems. Here, we have reported a dynamic camera (mounted on a pan-tilt device) based tracking system. A color feature based mean-shift tracking algorithm is used for estimating the target position in a new frame. The pan-tilt parameters are adjusted by a proportional-integral control action, obtained by processing the error between the current target position and the user specified set point. Simulation studies are performed for optimal selection of controller gains to achieve fast and stable camera re-orientation. More so, the simulation predictions are verified by evaluating the system response to a step change of the command input. The results of experimentation are shown, where the system continuously re-orients itself to place a target in the central region of the image.

The extensions to this work should incorporate improvements in both computer vision and analysis of system dynamics. The vision system of DynaTracker should be enhanced further by equipping the same with modules for automatic detection and initialization of the target region. More so, the system should also be able to handle the problems caused by occlusions. In this paper, we have only focused on the error dynamics of the system. A plant dynamics will also require attention, if a servo-based pan-tilt device is introduced. However, a more challenging study will be the dynamics of visual tracking. Thus, a combined study of a servo-based plant behavior along with the stochastic dynamics of visual tracking and the analysis of subsequent control action will be a significant contribution to the scientific research community.

## References


[1] J. Batista, P. Peixoto, and H. Araujo, "Real-time active visual surveillance by integrating peripheral motion detection with foveated tracking," in *Visual Surveillance*, January 1998, pp. 18–25.

[2] F. Bremond and M. Thonnat, "Issues of representing context illustrated by video-surveillance applications," *International Journal of Human-Computer Studies Special Issue on Context*, vol. 48, pp. 375–391, 1998.

[3] J. M. Ferryman, S. J. Maybank, and A. D. Worrall, "Visual surveillance for moving vehicles," *International Journal of Computer Vision*, vol. 37, pp. 187–197, June 2000.

[4] I. Haritaoglu, D. Harwood, and L. Davis, "W4 : Real time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 809–830, August 2000.

[5] M. Trivedi and I. H. K. Mikic, "Intelligent environments and active camera networks," in *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, 2000, pp. 804–809.

[6] N. Papanikolopoulos, P. Khosla, and T. Kanade, "Vision and control techniques for robotic visual tracking," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '91)*, vol. 1, April 1991, pp. 857–864.

[7] K. Daniilidis, C. Krauss, M. Hansen, and G. Sommer, "Real-time tracking of moving objects with an active camera," *Real-Time Imaging*, vol. 4, no. 1, pp. 3–20, 1998.

[8] D. Murray and A. Basu, "Motion tracking with an active camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 449–459, 1994.

[9] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Computer Vision and Pattern Recognition*, vol. 2, 2000, pp. 142–149.

[10] J. Slotine and W. Li, *Applied Non-linear Control*. Prentice Hall, October 1990.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[12] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian detection using wavelet templates," in *Computer Vision and Pattern Recognition*, June 1997, pp. 193–199.

[13] B. Heisele and C. Woehler, "Motion based recognition of pedestrians," in *International Conference on Pattern Recognition*, 1998.

[14] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 90, pp. 1151–1163, July 2002.

[15] R. Mester, T. Aach, and L. Dumbgen, "Illumination-invariant change-detection using a statistical co linearity criterion," in *DAGM*, 2001.

[16] O. Javed, K. Shafique, and M. Shah, "A hierarchical approach to robust background subtraction using color and gradient information," in *Motion and Video Computing*, December 2002, pp. 22–27.

[17] M. Isard and A. Blake, "Condensation : Conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, pp. 5–28, 1998.

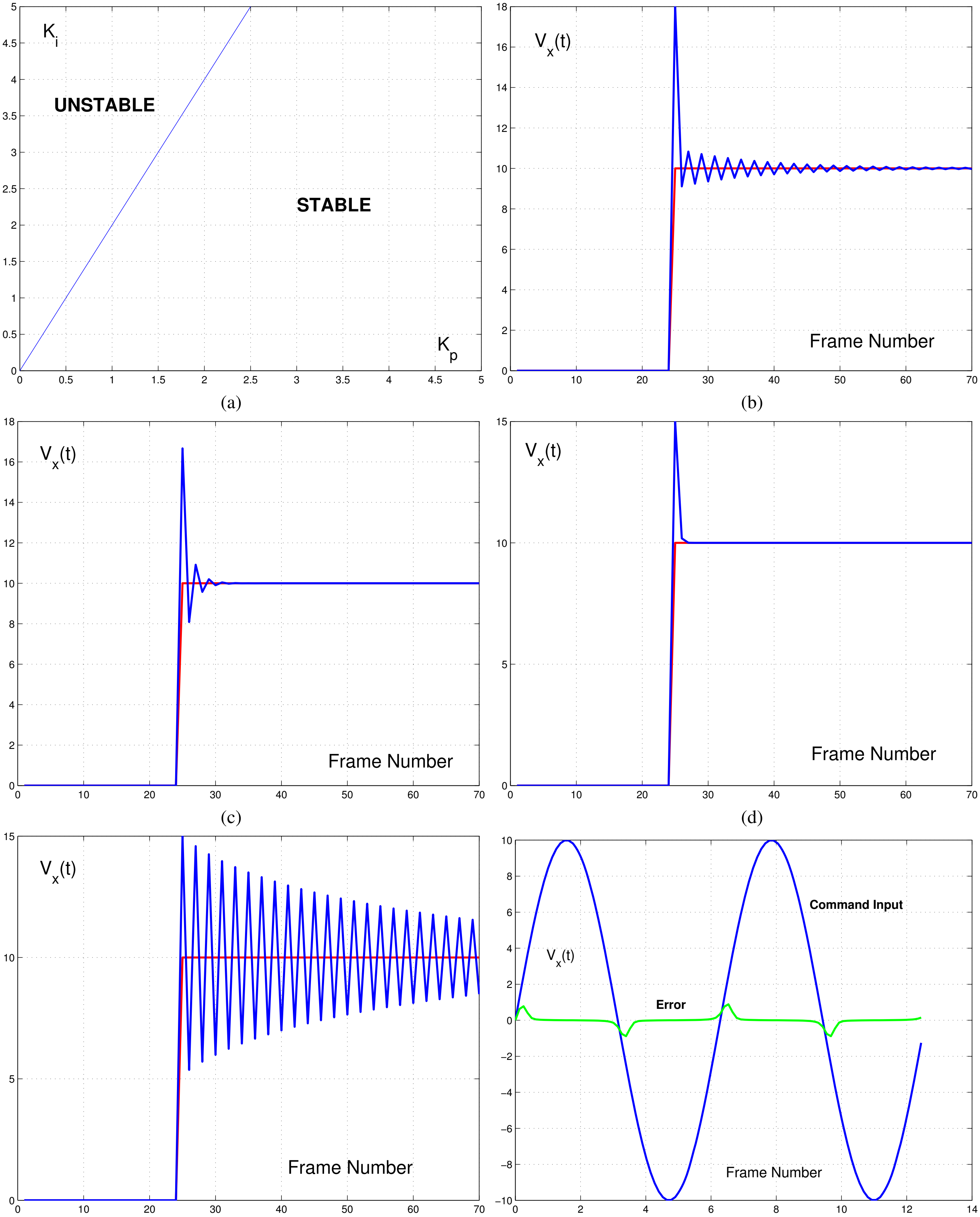


Fig. 4. The simulation results for the error dynamics. (a) Control chart in $K_P - K_I$ plane; DynaTracker Responses for a Step Input: (b) $K_P = 0.25, K_I = 0.5$, (c) $K_P = 0.5, K_I = 0.75$, (d) $K_P = 1, K_I = 1$, (e) $K_P = 1, K_I = 2$; (f) Image position error for sinusoidal command input with $K_P = 1, K_I = 1$.

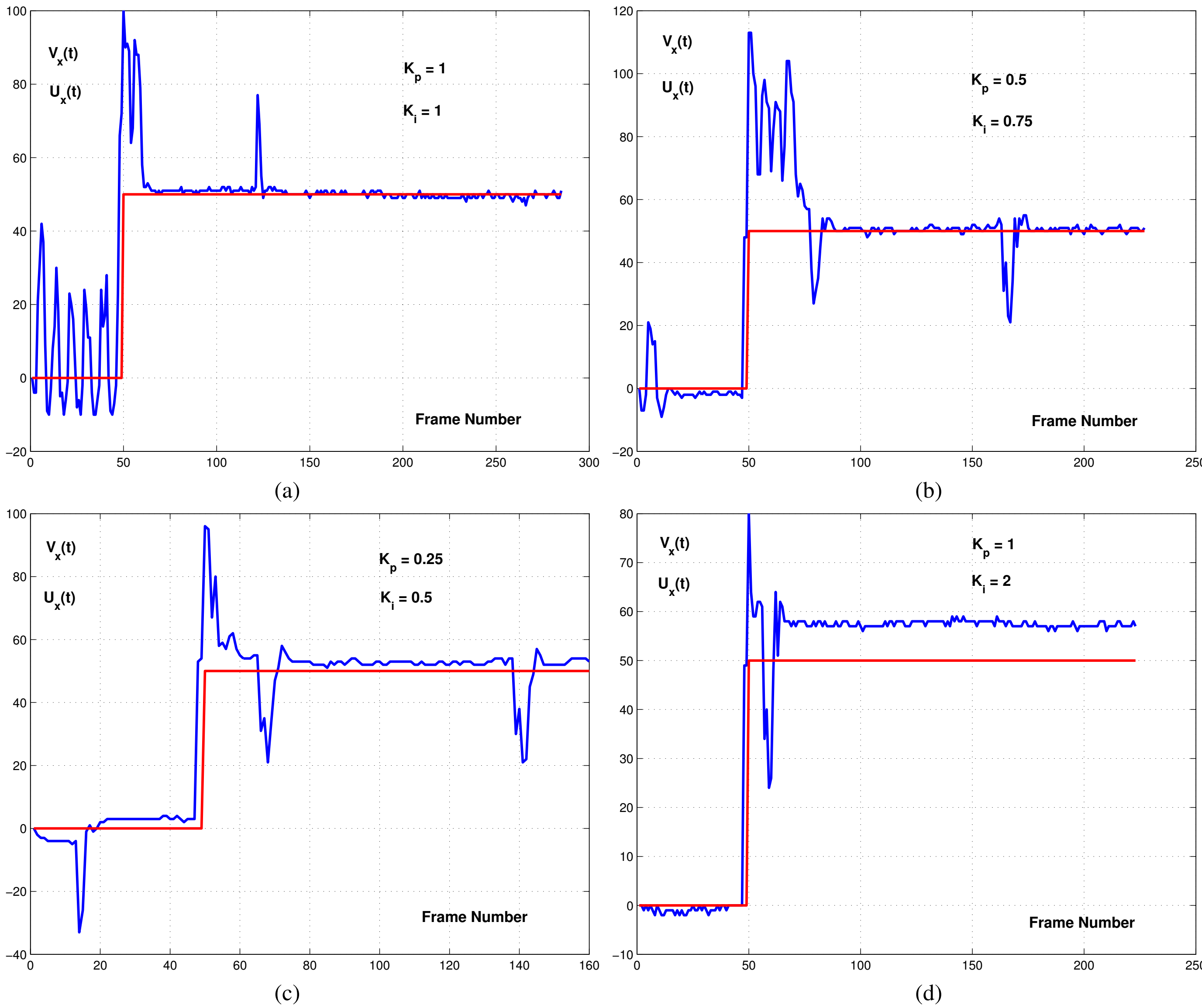


Fig. 5. DynaTracker response to a step input of $50 Pixels$. (a) $K_P = 1.0, K_I = 1.0$, (b) $K_P = 0.5, K_I = 0.75$, (c) $K_P = 0.25, K_I = 0.5$, (d) $K_P = 1.0, K_I = 2.0$.

(a) (b) (c) (d) (e) (f)

Fig. 6. The tracking results of a ball held by a person moving around in the lab environment with the objective of keeping the target at the center of the image. (a) Frame Number 7, (b) Frame Number 291, (c) Frame Number 449, (d) Frame Number 1064, (e) Frame Number 1210, (f) Frame Number 1349.