

CONCISE TRACK CHARACTERIZATION OF MANEUVERING TARGETS^{*}

Stephen Paul Linder[†]
spl@alum.mit.edu

Matthew D. Ryan

Richard J. Quintin

Abstract The identification and characterization of targets is difficult using conventional Kalman filter-based trackers, because these trackers are not able to provide an accurate and succinct list of maneuvers for use in track and target identification. We propose a new approach, representing target tracks as a sequence of circular and linear splines which directly correspond to the sequence of coordinated turns and constant velocity motion that a target may perform. The splines provide a locally optimal estimate of the track by (1) minimizing the total least-squares error of the data fit, and (2) minimizing the discontinuity in position and heading between successive splines. In addition, a parsimonious representation is obtained by dynamically merging redundant splines using a recursive algorithm. Our simulations studies show a large improvement in estimation accuracy, especially with respect to turn rate and heading. Our new approach shows great promise as a technique for ascertaining concise track characterization of maneuvering targets.

Keywords: target tracking, target characterization, track characterization, feature analysis.

INTRODUCTION

Track characterization of maneuvering targets is an area of active research^{8, 14}. Efficient characterization of a target's track promotes rapid target identification, identification of the target's operating mode and situational assessment⁵. A useful track characterization algorithm

effectively compresses the target track into a small set of parameters that provide a concise spatial description of the target's track². This paper proposes a new technique for parameterizing a target track that is very similar to those used in the pattern recognition community to ascertain a parametric representation of scanned mechanical drawings, and even handwriting, but has not been applied to the target tracking problem. Our technique, the **Segmenting Track Identifier** (STI), develops, in real-time, a parametric spline representation of the target track consisting of a series of linked circular arcs and straight lines. This decomposition of the target track corresponds to the coordinated turn model, turns executed at a constant turn rate, and the constant velocity model often used to describe the motion of targets. These same models are used by the Kalman filters in the Integrated Multiple Model (IMM) tracker². However, the IMM tracker does not provide any convenient method for extracting a parametric model of the target track, while our STI tracker inherently extracts a parsimonious description of the target track.

A parametric representation of the target track as a series of circular arcs and lines allows for track characterization that transcends the common kinematic metrics used today. The target track can thus be easily described as a sequence of linked turns with each turn described by familiar parameters: start of turn, initial turn heading, turn rate, and final turn heading. Using these descriptions, operating modes of targets can be inferred and the current metrics used for situational assessment can be augmented².

The STI algorithm performs a three step analysis of the target track. Initially, as data is acquired, the target

^{*} This material is based on work supported by Dr. Teresa McMullen through the Office of Naval Research under Contract No. N00039-D-0042, Delivery Order No. D.O. 278.

[†] Currently at SUNY Plattsburgh, Computer Science, 101 Broad Street., Plattsburgh, NY 12901

track is segmented into a sequence of circular arcs and lines. In a second phase, the segments are locally optimized to minimize the total least-squares error of the curve fit and to maximize the continuity of position and velocity at the knots between the curve segments. In a final phase, segments are fused to minimize the number of knots. Minimizing the number of segments compresses the parametric representation of the track and facilitates reasoning about the track, reasoning that can identify and ascertain the operating mode of the target.

This paper continues with background from the spline and pattern recognition literature. Then a concise description of our STI algorithm is presented. Finally, we present simulation results that compare the performance of the STI algorithm to a Kalman filter-based tracker.

BACKGROUND

Obtaining a globally optimal track estimate of a target with constrained kinematics is computationally intensive and currently impractical to perform in real-time. Instead, our approach segments the track and computes locally optimal estimates of the target track, using circular arcs and straight lines to describe the major kinematic modes of the target. This partitioning of the track into splines insures computational tractability.

We continue our background discussion with a brief description of splines, then a discussion of pattern recognition using segmentation and fusion of arcs.

Splines

Splines are used to plan the smooth motion of objects in computer graphics and robotics^{1, 4, 7} as well as to describe the boundaries of objects. Given a finite set of data points, splines provide a concise mathematical description of smooth parametric curve, $Q(t) = [x(t), y(t)]$, that passes through all the points. The representation is generated by fitting polynomials to local patches of the data and then insuring continuity between the patches. In practice, splines are limited to cubic polynomials, because higher order polynomials have more than two points of inflection and thus are harder to manipulate. The point at which each local spline meets is referred to as a *knot*⁴.

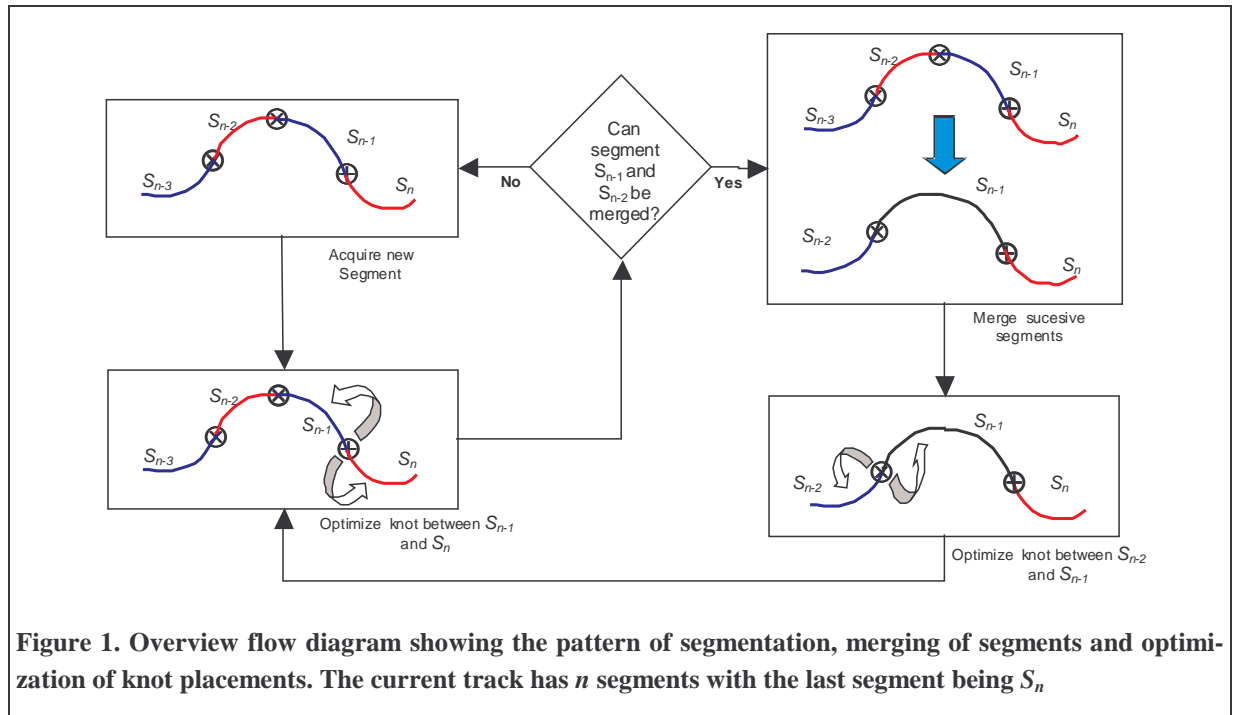
The STI approach corresponds most closely to curve construction using Hermite splines with first and second order, $dQ(t)/dt$, continuity constraints at the knots. The individual Hermite spline patch is uniquely defined by the location and tangent vector at each knot. The first and second order continuity constraints corresponds to target position and velocity continuity in the STI algorithm; the target model assumes bounded velocity and acceleration.

The general Hermite spline approach uses cubic polynomials for the individual spline patches, making the construction of accurate target tracks very susceptible to degradation from noisy measurements. The STI approach reduces the sensitivity to noise by limiting the splines to circular arcs, one form of a quadratic polynomial, and linear polynomials.

Curve Segmentation

Curve segmentation techniques determine the optimal partition of curves into a sequence of splines. Segmentation is a nonlinear optimization problem that (1) optimizes the fit to the original data, (2) optimizes the knot positions and (3) minimizes the number of spline segments¹³. Whereas the first condition is required by any target tracker, the second and third condition are required for generating a parsimonious track characterization.

The segmentation of curves is the topic of multiple pattern recognition papers^{6, 9-11}. Segmentation is important for the meaningful and compact description of objects. Hsin-Teng Sheu and Wu-Chih Hu¹¹ describe a methodology for partitioning contours using only circular arcs and line segments. Knots are identified that have both continuity and discontinuity in the slope of the two splines meeting at the knot, corresponding to corners and smooth breakpoints. Once the initial segmentation of the curve is complete, adjacent spline segments are recombined. Adjacent arcs, with similar radii and center points, are merged when the significance of the new arc, the ratio of the Root Mean Squared (RMS) error to the arc length, is decreased. This biases the merging process towards longer arcs that do not proportionally increase the error of the curve



fit. However, these algorithms depend on calculating the curvatures of the curve and require relatively noise free data.

Curve segmentation has also been applied to the automatic translation of paper-based engineering drawings into an algorithmic representation that can be used as input into CAD/CAM systems^{3, 12}. Again these algorithms rely on noise free data from which to construct the curve segments.

STI ALGORITHM

The **Segmenting Track Identifier** (STI) algorithm is a real-time algorithm that recursively develops an optimal, in the least squares sense, and parsimonious spline representation of a continuously acquired track. The algorithm fits circular arcs to the coordinated turns and linear segments to constant velocity phases of the track. In order to perform optimally, the STI algorithm must therefore accurately collocate the start and the end of all maneuvers with the location of the knots used in the splines representation of the track.

As data is acquired, the STI algorithm generates new spline segments. Data points are appended to the last spline, S_n , until the spline's *significance*¹¹ begins to

decrease, and then a new knot is placed. As shown in Figure 1, the STI algorithm next optimizes the last two splines, S_{n-1} and S_n , by optimizing the parameters of both splines while searching for the optimal location of the connecting knot. Next, segments S_{n-1} and S_{n-2} are tested to see if they can be merged.

However, unlike the technique of Wenyin and Dori¹² where an exhaustive search is made for the common center for consecutive arcs, the STI algorithm (1) tests for a plausible merging, (2) finds the new optimal spline for the merged data and then (3) tests the significance of the new arc spline compared to the significance of the original two splines. If the new spline has greater significance than the shorter arcs, the new longer arc is used.

The algorithm continues to recursively merge the two terminating spline segments until merging no longer improves the significance of the splines. When this recursion terminates, new data is again acquired and a new spline segment is appended to the target track.

Splines are optimized by (1) performing a least squares fit of the spline to the data, and (2) matching the boundary condition between the spline and its two neighboring splines. Whereas the Hermite splines and other pat-

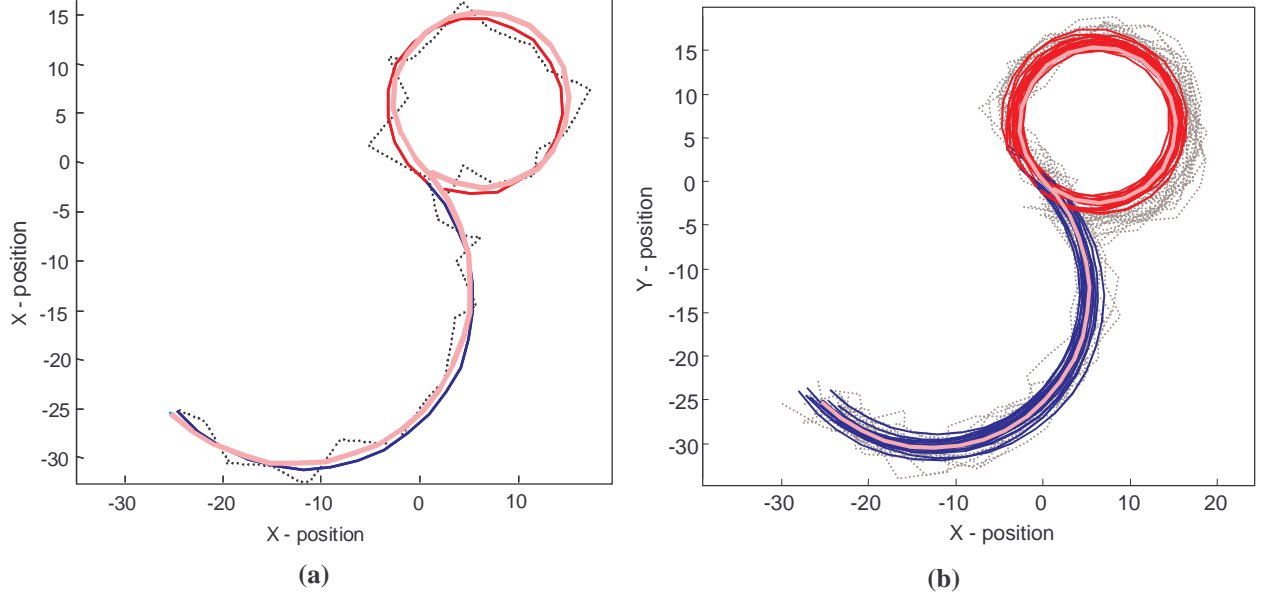


Figure 2. When tracking a constant speed target executing two consecutive coordinated turns the STI tracker (black line) shows superior performance over a Kalman filter-based tracker (dotted line). In (a) one trial is compared while in (b) the tracks from 20 trials are superimposed. The gray line shows the true track. The measurement noise has a STD = 2.0.

tern recognition techniques use linear techniques to determine the optimal spline, our technique requires the use of nonlinear least squares optimization. Currently, STI uses the *lsqnonlin* function found in the Matlab® Optimization Toolbox.

The total least squares term for a line segment Q_L is

$$Q_L = \sum_{i=1}^p \left[\frac{y_i - mx_i - b}{\sqrt{m^2 + 1}} \right]^2 \quad (1)$$

where p is the number of data points fitted to the spline, m the estimated slope of the line and b the estimated y-intercept.

The total least squares term for circular arc segment Q_A is

$$Q_A = \sum_{i=1}^p \left(\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} - r \right)^2 \quad (2)$$

where (x_c, y_c) is the estimated center of the circular arc and r is the estimated radius.

The C^0 and C^1 continuity condition is given by

$$Q_C(n) = k_p (\Delta\phi_{n-1,n} + \Delta\phi_{n,n+1} + \Delta\mathbf{R}_{n-1,n} + \Delta\mathbf{R}_{n,n+1}) \quad (3)$$

where k_p is a proportionality constant based on the length of the diagonal of the spline's bounding box, $\Delta\phi_{n,n+1}$ is the difference in heading at the knot between the n and $n+1$ segment, and $\Delta\mathbf{R}_{n,n+1}$ is the difference in position at the knot between the n and $n+1$ segment. The optimal spline is one that minimizes the total cost Q where Q is either $Q_A + Q_C$ or $Q_L + Q_C$.

The current implementation of the STI algorithm limits the placement of knots to the location of the data points in order to limit computational complexity of the algorithm. This limitation is not severe when data is spaced evenly, but is problematic when no data measurements are made near the onset of a target maneuver. Future versions of the algorithm will interpolate knot location between measurements when necessary to insure optimal knot placement and continuity.

SIMULATION RESULTS

The performance of the STI tracker is compared to a baseline tracker consisting of a properly tuned Kalman Filter using a Singer maneuver model². The target track is a sequence of either two or five linked coordinated turns. Tracker parameters for both the STI and Kalman

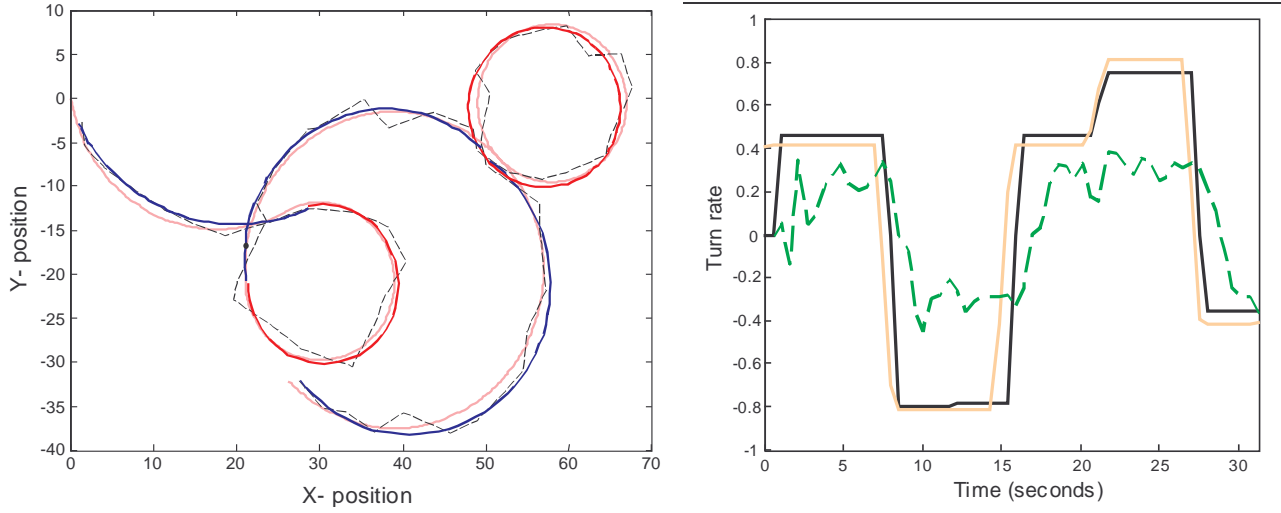


Figure 3. A comparison in (a) of the tracking performance of the STI algorithm(back line), against the Kalman filter-based tracker (dashed line) using 60 data points with a measurement noise STD = 1.0 (dotted line). The gray line shows the true track. (b) shows the accuracy of the two trackers in estimating the target turn rate.

filter-based trackers were held constant over all the various trial configurations. Simulations were performed using Matlab®, Version 5.3 with the Optimization Toolbox, Version 2.0.

Initial trials were run using a segment consisting of two circular arcs. Figure 2 (a) compares the performance of the STI tracker for one trial. The target track consists of two consecutive coordinated turns. Forty uniformly distributed position measurements are made using Gaussian measurement noise with a Standard Deviation (STD) = 2.0. Not only does the STI tracker provide better position estimates, it also provides better velocity and acceleration estimates. A comparison of twenty Monte Carlo trials, in Figure 2 (b), shows that the STI provides consistently superior estimates. As shown in Table 1, the RMS error of the position, velocity and acceleration estimates for the STI tracker are at least half of that for our baseline tracker.

Table 1. Comparison of RMS errors between the baseline tracker and STI tracker for 20 trials of the two-maneuver track.

RMS Error	Kalman filter-based tracker	STI Tracker
position	2.1273	0.6554
velocity	3.5165	0.84407
acceleration	3.7928	1.4518

Further studies were made using a track composed of five circular arcs where the target makes two closed loops. Figure 3 (a) shows that the position estimates of the STI tracker does not degrade when the target follows a track with five consecutive coordinated turns. The STI tracker gives a superior estimate of the target turn rate as shown in Figure 3 (b). The estimate from the STI tracker clearly differentiates between the five different maneuvers, while the Kalman filter-based tracker is unable to distinguish between the third and fourth maneuver.

Nine sets of 100 trials, combinations of three measurement noise levels and three sampling rates, were performed to test the robustness of the STI algorithm. The three noise levels are 0.5, 1.0, and 2.0 STD. The 2.0 STD corresponds to 2/9th of the radius of the closed loops. Trials were run with 60, 120 and 180 data points. 60 data points correspond to only 12 data points per maneuver. The bubble chart in Figure 4. A bubble chart is used to compares the mean RMS error in the velocity and acceleration estimates for nine set of experimental parameters: a combination of three RMS measurement errors, 0.5, 1.0 and 2.0, and three sample sizes, 60, 120 and 180 samples. The y-axis represents the velocity RMS error, while the diameter of the circle represents the RMS acceleration estimate. The smaller, nine circles, lower on the graph are the results from the STI tracker, while the larger circles, higher in the graph, are

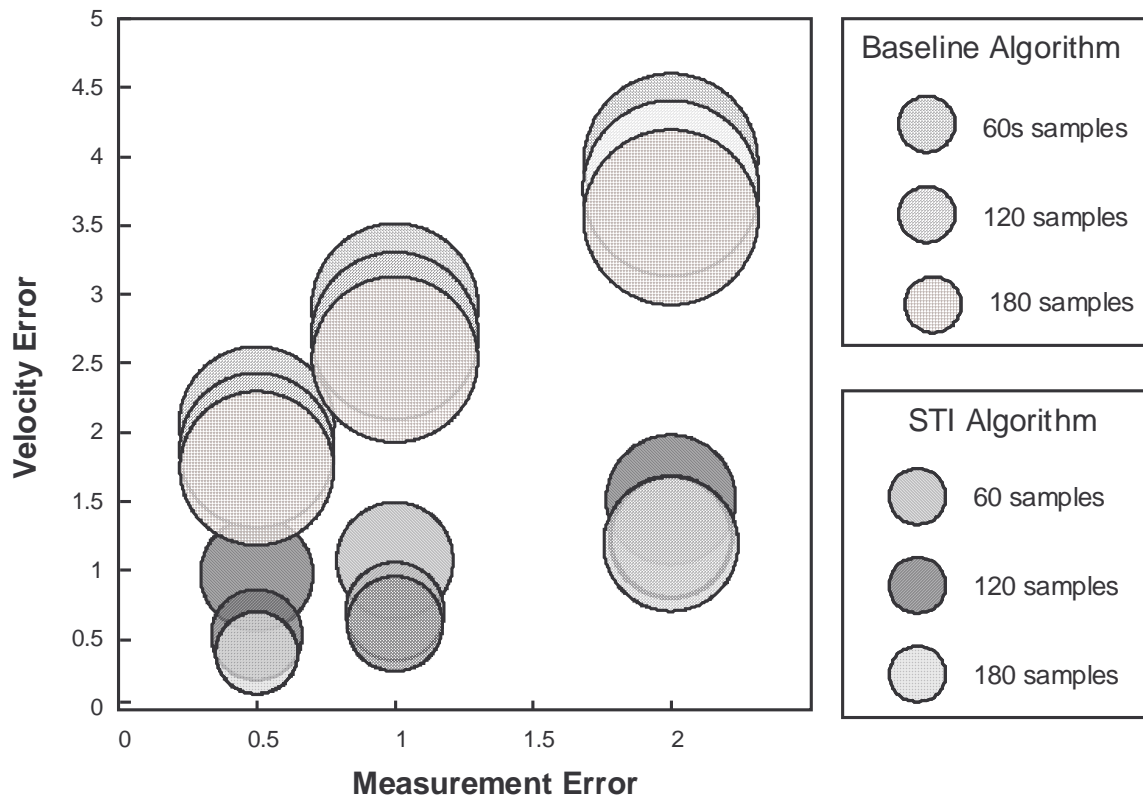


Figure 4. A bubble chart is used to compares the mean RMS error in the velocity and acceleration estimates for nine set of experimental parameters: a combination of three RMS measurement errors, 0.5, 1.0 and 2.0, and three sample sizes, 60, 120 and 180 samples. The y-axis represents the velocity RMS error, while the diameter of the circle represents the RMS acceleration estimate. The smaller, nine circles, lower on the graph are the results from the STI tracker, while the larger circles, higher in the graph, are the results from the conventional tracker.

the results from the conventional tracker. shows that for all nine sets of trials the STI tracker outperforms the conventional tracker. The bubble chart represents the RMS velocity estimation error as the y-location of the bubble’s center, while the RMS acceleration error is represented by the diameter of the bubble. Both trackers show a correlation between increased sample size and a reduction in velocity estimation error. However, only the STI tracker reduces acceleration estimation error when the sample size increases. Table 2 lists the RMS error for all 900 Monte Carlo trials and confirms the superior accuracy of the STI tracker.

CONCLUSION

The STI approach to track characterization has been shown through simulation to achieve superior estimation of target position, velocity, and acceleration for a maneuvering target performing linked coordinated turns even when there is significant measurement noise and only sparse data. Beyond the improved estimation of the target track, the STI approach allows us to succinctly describe the target track as series of linked turns. Each turn is described by the turn rate, beginning and ending heading, and length of turn. Such a parameterization of the target track allows the development of inferring systems that both identify a target and char-

Table 2. Comparison of RMS errors between the baseline tracker and STI tracker for 900 trials of the five-maneuver track for a combination of three error standard deviations, 0.5, 1.0 and 2.0, and three sample rates, 60, 120 and 180.

RMS Error	Kalman filter-based tracker	STI Tracker
position	1.56	0.64
velocity	2.79	0.91
acceleration	3.41	1.52

acterize the target's current operating mode based on the track description.

Our current research is extending the STI algorithm to work directly with both passive and active radar measurements, improve instantaneous estimates, and automatically generate target behavior descriptions.

BIBLIOGRAPHY

- [1] J. Angeles, A. Alivizatos, and P. J. Zsombor-Murray, *The Synthesis of Smooth Trajectories for Pick-and-Place Operations*, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, pp. 173-178, 1988.
- [2] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.
- [3] D. Dori, *Vector-Based Arc Segmentation in the Machine Drawing Understanding System Environment*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 1057 - 1068, 1995.
- [4] James D. Foley, Andries van Dam, Steven K. Feiner, and J. F. Hughes, *Computer Graphics : Principles and Practice, Second Edition in C*. Reading, MA: Addison-Wesley Pub Co, 1996.
- [5] W. T. Johnson and I. W. Dall, *From Kinematics to Symbolics for Situation and Threat Assessment*, presented at Information, Decision and Control, 1999.
- [6] K. B. Lim, K. Xin, and G. S. Hong, *Detection and Estimation of Circular Arc Segments*, *Pattern Recognition Letters*, vol. 16, pp. 627-636, 1995.
- [7] H. Ozaki and C.-j. Lin, *Optimal B-Spline Joint Trajectory Generation for Collision-Free Move-*
- ments of a Manipulator under Dynamic Constraints*, presented at IEEE International Conference on Robotics and Automation, Minneapolis, MN, USA, 1996.
- [8] B. S. Rao and H. Durrant-Whyte, *A Decentralized Bayesian Algorithm for Identification of Tracked Targets*, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, pp. 1683 - 1698, 1993.
- [9] P. L. Rosin and G. A. W. West, *Segmenting Curves into Elliptic Arcs and Straight Lines*, presented at Computer Vision, 1990.
- [10] S. Saga, *A Freehand Interface for Computer Aided Drawing Systems Based on the Fuzzy Spline Curve Identifier*, presented at IEEE International Conference on Systems, Man and Cybernetics, 1995.
- [11] H.-T. Sheu and W.-C. Hu, *Multiprimitive Segmentation of Planar Curves - a Two-Level Breakpoint Classification and Tuning Approach*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 791-797, 1999.
- [12] L. D. Wenyin, D., *Incremental Arc Segmentation Algorithm and Its Evaluation*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 424 - 431, 1998.
- [13] F. M. Yoshimoto, M.; Harada, T., *Automatic Knot Placement by a Genetic Algorithm for Data Fitting with a Spline*, presented at International Conference on Shape Modeling International '99, 1999.
- [14] X. Yu and M. R. Azimi-Sadjadi, *Neural Network Directed Bayes Decision Rule for Moving Target Classification*, *IEEE Transactions on Pattern Analysis and Machine*, vol. 36, pp. 176 - 188, 2000.