Multiview System for Tracking a Fast Moving Object Against Complex Backgrounds

Conference or Workshop Item

Multiview System for Tracking a Fast Moving Object Against Complex Backgrounds

Patrick Wong
School of Computing and Communications
The Open University
Milton Keynes, United Kingdom
patrick.wong@open.ac.uk

Abstract— **Tracking the real world coordinate of a fast moving object against a complex background is very challenging. When designing a multi-view system for this purpose, one key consideration is the arrangement of the cameras such that the object can be constantly and accurately tracked. This paper discusses a novel cameras arrangement, which can provide redundancy for fault tolerance, yet do not require installing more cameras nor relying aerial views of the scene. Using a table tennis match as example, experiment results show that the multi-view system with this cameras arrangement has a promising potential for tracking a table tennis ball in a real match scene.**

Keywords—Object tracking, multiview

I. INTRODUCTION

For many computer vision applications, accurately detecting and tracking the real world coordinate of an object is crucially important. Conventionally, stereo-camera sets are employed for this purpose. A stereo camera set consists of two identical cameras separated by a small horizontal distance along the same X-axis. It captures two images of the object at a slightly different horizontal positions simultaneously. As a result, the object appears at a different spatial position on each view of the cameras. This difference, known as disparity, can be used to determine the three-dimensional (3D) real world position of the object using triangulation [1]. While this approach is well-established, there are drawbacks and limitations. First of all, it requires two cameras to cover one perspective and hence the number of cameras required to cover multi-perspective will be twice the number of perspectives. Secondly, the calculation of the object's real world coordinate is only possible when the object appears on both views of the stereo-camera set. However, if the object appears near the far left or right edges of the camera views, it may not be covered by both views. Furthermore, if one of the camera views of the object is blocked by an obstacle, calculation of the object's real world coordinate is also not possible. Finally, the accuracy of the calculated real world coordinate of the object often deteriorated as the object is situated further away from the stereo-camera set. This is due to the fact that the object appears smaller in the views when it is far away from the cameras. Hence the calculation of disparity is more error prone and the error can propagate to the calculation of the real world coordinate.

These drawbacks and limitations provide a motivation for developing a more robust multi-view tracking system, which requires fewer number of cameras to work and enables the calculations of the object's real world coordinate in multiple ways. It is also aimed to make this system inexpensive and portable. Using a table tennis rally captured in a real match scene as an example to demonstrate the ability of the new multi-view tracking system, this paper will discuss the arrangement of the multi-camera set up and compensation of the measuring errors. The reason for choosing a table tennis ball as the object to be tracked is that it travels fast in a table tennis match, its view can be blocked by the players and the background of a match scene is complex. Tracking a table tennis in a real match scene is very challenging.

The remainder of the paper is organized as follows: Section II reviews the literature on research works involving a multi-view tracking system for table tennis balls, while Section III describes the proposed multi-view tracking system. Section IV presents the experimental set up, results and discussion, while Section 5 makes some concluding comments.

II. LITERATURE REVIEW

As a table tennis ball was used as the object to be tracked in this study, the literature review was focused on recent research works based on tracking table tennis balls. In 2009, [2] discussed the design of a high speed tracking system using distributed parallel processing architecture. Their system employed two high speed cameras, which were mounted at two corners of the ceiling to obtain aerial views of the ball and the table. The ball was detected using a combination of background subtraction, pixel thresholding and the growth-of-sampled-points method, which aimed to recover incorrectly removed pixel lost during background subtraction. The image position of the ball from both camera views then sent to a powerful PC for determining its real world 3D coordinate using a standard camera model with intrinsic and extrinsic parameters. The location of the ball was tracked in the each frame with an aid of a landing point prediction model, which assumed the trajectory of the ball is a straight line in the X-Y plane and parabola in X-Z plane. The system could detect the ball and worked out its real world coordinate in about 8ms and the error of detection is less than 4cm. Despite the good result, the main drawback of this system is that it relied on successful detection of the ball in both views to determine its real world coordinate, i.e., it could not work out the coordinate of ball if it was not detected in one of the view.

Furthermore, the system worked only on aerial view of the scene, which had an advantage of viewing the ball against a simple uniform color background but made the system not portable. The setup was also in a controlled laboratory environment, which is not as complex as a real match scene.

In 2012, [3] proposed a table tennis ball tracking system aimed to help robots to play table tennis. Their system made use of four high speed cameras, of which a pair were mounted on the ceiling above both side of the table. Their system employed a color based thresholding and features extraction method to detect the ball on each of the four views. Similar to [2], the real would coordinate of the ball was determined using the image positions of the ball in both views of each pair of cameras. As the two pairs of cameras are opposite facing, the view of the ball is likely to be captured by one of the pairs. This system also employed a trajectory prediction model. It used an aerodynamic model to estimate the ball trajectory when it was in mid-air and a bouncing model when the ball bounced on the table. As a result, the average Euclidean error reduced to less than 2cm, while the detection time is 8ms per frame despite high specification computer was used. The main weakness of this system was its reliance on aerial view of the scene to work, which made it not portable.

The weakness of reliance on successful detecting the ball in both views simultaneously was addressed by [4]. Their work was built on [2] but aimed to help robot to play table tennis. Their system still employed background subtraction to detect the ball, but it improved the establishment of the background by using 15 frames. It also used an improved Single Gaussian model to estimate the rough position of the ball based on its speed. They developed a trajectory model based on a n-order polynomial and estimated the image position of the undetected ball from a number of successful detection in previous frames. As a result, if a ball was not detected from a view, its image position could be estimated and used along with the detected image position of another view to determine the ball's real world coordinate.

In 2015, [5] proposed a multi-view tracking system that employed 4 high speed cameras, which were divided into two pairs and were mounted on tripods at a height of 1.4 meter. They had experimented with two cameras arrangements: 1) a pair of cameras were placed on each side (the long side only) of the table (opposite facing); 2) both pairs were placed on the same side of the table but each pair only monitored half of the table (side-by-side). The test videos were captured at a real match scene. The system detected the ball using a combination of background subtraction and adaptive color thresholding method. The image positions of the ball from two views of a pair of cameras were used to calculate the real world coordinate. The trajectory prediction was made using a second-order motion model, which estimates the current velocity and acceleration from ball coordinates of previous frames. Their results found that the opposite facing arrangement could handle the occlusion problem better as the ball was likely to be captured by one of the opposite facing pairs of cameras. However, the depth resolution of this arrangement was lower since each pair of cameras had to monitor the whole table and hence it achieved lower detection rate and higher detection error. In contrast, the side-by-side arrangement, which had higher depth resolution, achieved higher detection rate and lower detection error but could not detect the ball when it is occluded. It relied on the trajectory prediction model to estimate the ball location when it was not detected.

Based on the work of [5], this paper proposed a new multi-view cameras arrangement, which was aimed to reduce detection error and improved robustness without using more cameras. The details of the configuration will be discussed in Section III.

III. Multi-view Cameras Configuration

Tracking a table tennis ball in a real match scene is very challenging. It is because the ball is small, moving fast and its view can be occluded. The image of ball can become distorted or dark if the video is captured with inappropriate aperture size and shutter speed, of which the ranges of these parameters are limited by the specification of the cameras. Environmental factors such as uneven illumination, confusing background objects, spectators' movement and reflective table surface can also affect the ball being successfully detected. Fig. 1 shows some example images of the ball at these challenging detection situations.

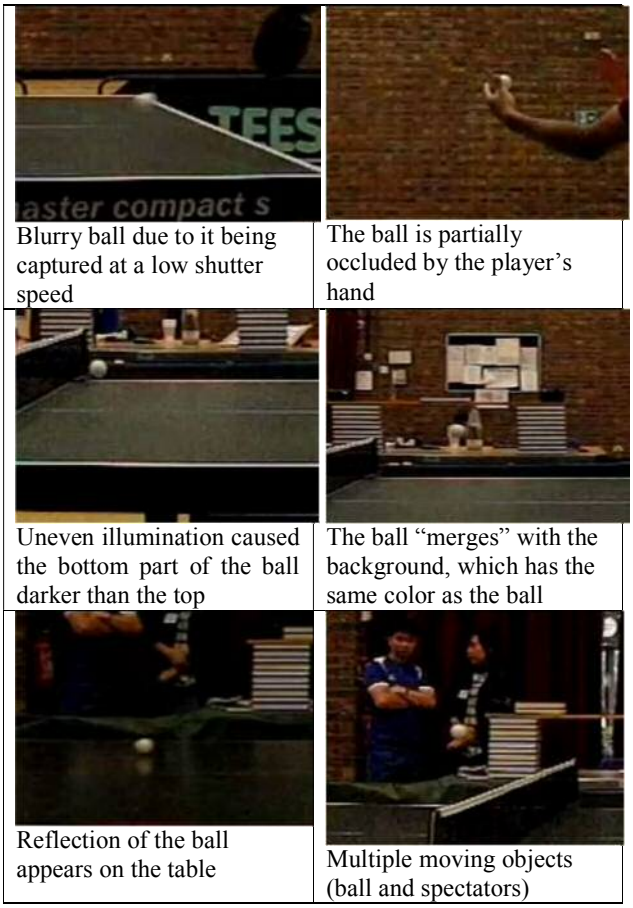


Fig. 1. Example images of the ball at challenging detection situations

To cope with these challenging detection situations, the multi-view system should provide redundancy so that when one

camera fails to detect the ball, another one that has a different perspective can detect the ball. Furthermore, table tennis tournaments usually take place at multi-purpose hall, where installing aerial-view cameras is often disallowed. Without using more cameras and obtaining aerial view, a novel multi-view cameras arrangement is proposed, as illustrated in Fig. 2. The system consists of four high speed cameras distributed evenly along both sides (long-side) of the table. Each camera only needs to monitor two third of the table, so that they can be placed closer to the table to get a better view. Each camera has an opposite facing partner and a side partner, e.g. Cam 2 and Cam 3 are opposite and side partners of Cam 1 respectively. The opposite facing cameras can work together to tackle the occlusion problem. The side pair can monitor the whole length of the table together, with the views overlapped at the middle, where the net is.

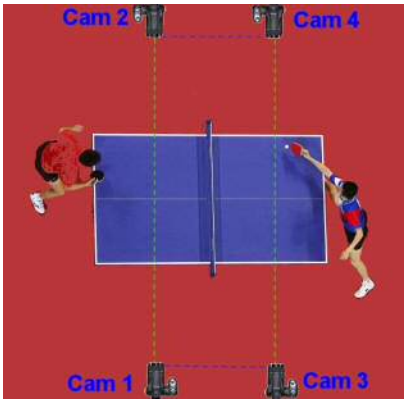
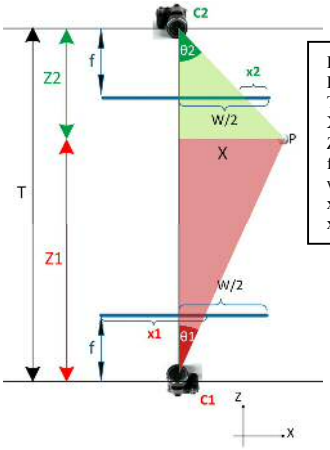


Fig. 2. Multi-view cameras configuration

To derive the 3D real world coordinate of the ball from 2D image positions of it, this arrangement enables two options. When the ball appears at a location where one pair of opposite facing cameras can see it, the real world coordinate of the ball can be calculated using the image positions of the ball detected by this pair of cameras. Figure 3 shows the aerial view of an opposite facing cameras pair, which allows the X- and Z-coordinates to be calculated.



Legends:
P: ball's position in real world
T: distance between two cameras
X: X coordinate of the ball
Z1: Z coordinate of the ball
f: Focal length of the cameras
w: width of the screen
x1: screen position of the ball on c1
x2: screen position of the ball on c2

Fig. 3. Aerial view of the opposite facing camera pair

Let the principal point of camera 1 (C1) be the origin. The X- and Z- coordinates can be calculated using Equ. (1) – (4).

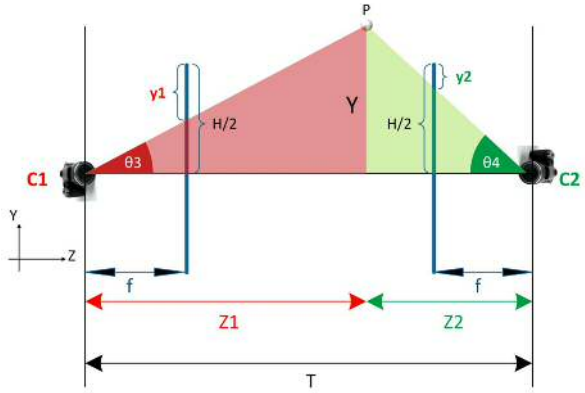
$$tan(\theta_1) = \frac{(x_1 - \frac{W}{2})}{f} = \frac{X}{Z_1} \quad (1)$$

$$tan(\theta_2) = \frac{(\frac{W}{2} - x_2)}{f} = \frac{X}{Z_2} \quad (2)$$

$$T = Z_1 + Z_2 = \frac{X}{tan(\theta_1)} + \frac{X}{tan(\theta_2)} \quad (3)$$

$$X = T(\frac{tan(\theta_1)tan(\theta_2)}{tan(\theta_1)+tan(\theta_2)}) \quad (4)$$

The Y- coordinate cannot be seen in the aerial view, so a side view of the camera configuration (Y against Z axes) was drawn and is shown in Fig. 4.



Legends:
Y: Y coordinate of the ball
H: height of the screen
y1: screen position of the ball on c1
y2: screen position of the ball on c2

Fig. 4. Side view of the opposite facing camera pair

The Y- coordinate can be calculated using Equ. (5) – (8).

$$tan(\theta_3) = \frac{(\frac{H}{2} - y_1)}{f} = \frac{Y}{Z_1} \quad (5)$$

$$tan(\theta_4) = \frac{(\frac{H}{2} - y_2)}{f} = \frac{Y}{Z_2} \quad (6)$$

$$T = Z_1 + Z_2 = \frac{Y}{tan(\theta_3)} + \frac{Y}{tan(\theta_4)} \quad (7)$$

$$Y = T(\frac{tan(\theta_3)tan(\theta_4)}{tan(\theta_3)+tan(\theta_4)}) \quad (8)$$

However, if the ball cannot be detected by one of the cameras in the pair, the other camera in the pair can attempt to work with its side partner to derive the real world coordinate using the geometry calculation, as shown in Fig 5 (aerial view). Based on similar triangles, the Z- coordinates can be calculated using Equ. (9) – (10).

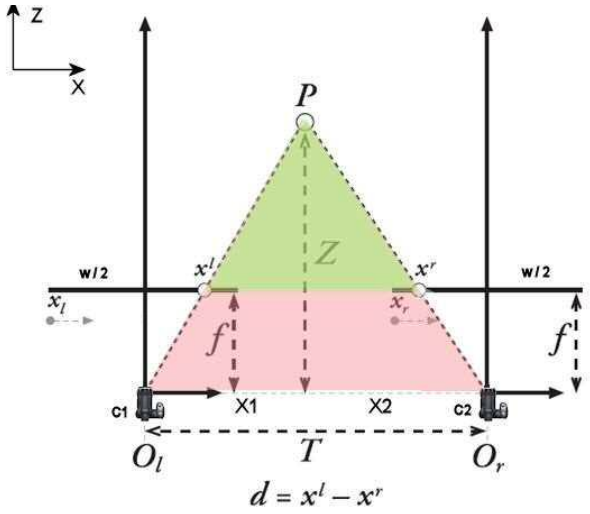


Fig. 5. Aerial view of the side-by-side camera pair

$$\frac{T}{Z} = \frac{T-(x^l-\frac{w}{2})-(\frac{w}{2}-x^r)}{Z-f} \quad (9)$$

$$Z = \frac{Tf}{x^l-x^r} = \frac{Tf}{d} \quad (10)$$

The X- coordinate can be calculated using Equ. (11) – (12).

$$\frac{(x^l-\frac{w}{2})}{f} = \frac{X_1}{Z} \quad (11)$$

$$X_1 = \frac{z(x^l-\frac{w}{2})}{f} \quad (12)$$

The Y- coordinate can be calculated by looking at side view (Y against Z axis) of the side-by-side pair of cameras, as shown in Fig. 6, and using Equ. (13) and (14).

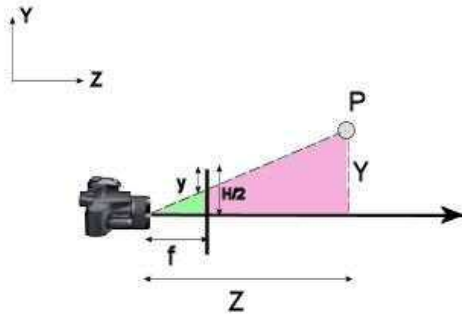


Fig. 6. Side view of the side-by-side camera pair

$$\frac{Y}{Z} = \frac{\frac{H}{2}-y}{f} \quad (13)$$

$$Y = \frac{Z(\frac{H}{2}-y)}{f} \quad (14)$$

IV. Experimental Setup and Results

To evaluate the accuracy of the system, a set of reference points, where their real world coordinates are known, were used as ground-truth for comparison. This ground truth were constructed by carefully placing an upright double sided checkerboard at various marked locations on the table during filming, as shown in Fig. 7. The checkerboard had 4 rows and 5 columns of identical sized black squares distributed evenly upon a white board, hence the position of each corner was known. The distance between the checkerboard and the principal point of Camera 1(used as the origin) was carefully measured at each marked location. Therefore the real world coordinates of all the corners of the checkerboard can be calculated. Apart from the checkerboard corners, the table corners and the tips of the net poles at both sides were also used to composite the ground truth. In total, there were 288 reference points in the ground truth.

Fig. 7. A upright double-sided checkerboard was used for creating reference points.

An initial test was conducted by manually identifying the image positions of several reference points in the images captured by the opposite facing cameras pair and calculated their real world coordinate using Equ. (1) to (8). The preliminary result found that the average Euclidean distance between the coordinates of the measured reference points and those calculated by the system is 4.8cm. The Euclidean distances varied non-linearly with respect to the distance between the reference points and the origin. This discrepancy was mainly due to the misalignment between the opposite facing and side-by-side cameras, measuring error and inaccurate image positions of the reference points. As each camera can has six degrees of freedom (X-, Y-, Z- translation, pitch, roll and yaw), it is very difficult to align them perfectly. The resolution of the image produced by these high speed camera is low (512 x 384 pixels), obtained accurate image position of the reference points are also difficult. To reduce this discrepancy, an error model was built, which takes the calculated coordinate as an input and produces an estimated error vector (E) for that particular coordinate, as shown in Equ. (15). By subtracting the error vector from the calculated coordinate, it will bring it closer to its true coordinate.

$$\boldsymbol{E}(x, y, z) = F(x,y,z)\boldsymbol{i}, G(x,y,z)\boldsymbol{j}, H(x,y,z)\boldsymbol{k} \quad (15)$$

where $\boldsymbol{E}(x, y, z)$ is the 3-D error vector, $F(x,y,z)$, $G(x,y,z)$, $H(x,y,z)$ are functions determining the magnitudes of the $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ components respectively, and (x, y, z) is the calculated coordinate of a reference point.

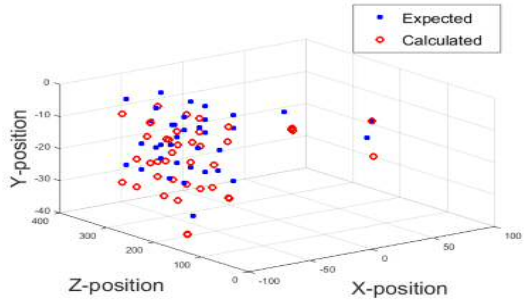
As the error appeared to be non-linear, $F(x,y,z)$, $G(x,y,z)$, $H(x,y,z)$ were defined as quadratic surfaces, as shown in Equ. (16) – (18) where a_n, b_n, c_n, d_n, e_n, f_n, g_n, h_n, i_n and j_n are coefficients of the surfaces, for $n = 1, 2$ and 3.

$$F(x,y,z)=a_1x^2+b_1y^2+c_1z^2+d_1xy+e_1xz+f_1yz+g_1x+h_1y+i_1z+j_1 \quad (16)$$

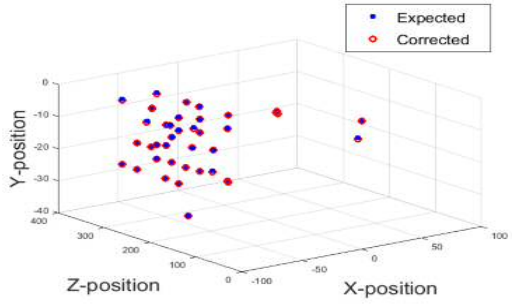
$$G(x,y,z)=a_2x^2+b_2y^2+c_2z^2+d_2xy+e_2xz+f_2yz+g_2x+h_2y+i_2z+j_2 \quad (17)$$

$$H(x,y,z)=a_3x^2+b_3y^2+c_3z^2+d_3xy+e_3xz+f_3yz+g_3x+h_3y+i_3z+j_3 \quad (18)$$

To find the coefficients for the quadratic surfaces, the Multivariate Polynomial Regression [6] was employed. To prevent overfitting, a small subset of 32 reference points were randomly selected from ground truth as training data, another 45 "unseen" points were chosen for validating. Fig. 8(a) shows the 45 uncompensated calculated (red) and expected (blue) coordinates of the reference points, while Fig. 8(b) shows the corrected coordinates of the reference points after error compensation. It is evident the calculated and expected positions are much closer. When the model was tested on the full date set, the average Euclidean distance is only 0.1cm, comparing to 4.8cm obtained at the initial test.



(a) Uncompensated expected and calculated reference points



(b) Corrected expected and calculated reference points

Fig. 8. Expected and calculated 3D coordinates of the reference points

V. Conclusion and Future Work

This paper presented a novel multi-view system that can be used to track a table tennis ball in a real match scene, which has complicated background and challenging ball detection condition. The system does not rely on an aerial view of the scene to work. This means fixing cameras to the ceiling is not needed, hence the system are more portable. With this novel cameras arrangement, fewer cameras are required to cover a large area, yet detection redundancy is provided.

Despite the difficulty in perfectly align the cameras, the developed error model effectively compensated the calculation errors, of which the original average Euclidean distance was 4.8cm but was reduced to 0.1cm after compensation.

However, the proposed system also has weaknesses. For the system to work as an automatic ball tracking system, the detected image position of the ball from each camera needs to be collated to derive the real world coordinate of the ball. This means the cameras needs to be able to communicate with each other or with a central server. Furthermore, as each camera only monitors two third of the table, it needs to work with its side partner effectively to monitor the whole table. While this cameras arrangement has many benefit, one main weakness is that when the ball is at or near the vertical plane that joins the principal points of the two opposite facing cameras, they will not be able to derive the real world coordinate of the ball. It is because when θ_1 in Equ (1) is zero, Z will be infinitive. When this occurs, the real world coordinate of the ball can be derived using the side-by-side camera pair.

To address the abovementioned weakness, a multi-agent system (MAS) is being developed to control and manage the data flow. A MAS consists of a number of inter-connected intelligent agents, which can jointly achieve a goal. This characteristic makes it very suitable for this application. For example, each camera can be controlled by an agent, which can detect the ball and send the detected ball location to another agent which can decide how best to derive the ball's real world coordinate and check whether it follows the predicted trajectory. If the detected ball location does not make sense, this agent can also feedback where the expected ball location to the camera agent and ask it to detect again at or near the expected ball location. The workload of the MAS can also be distributed to a network of computers such that the overall performance and reliability can be improved.

References

[1] R. Szeliski, "*Computer vision: algorithms and applications"*. Springer Science & Business Media, 2010.

[2] Z. Zhang, D. Xu, "*Design of High-Speed Vision System and Algorithms Based on Distributed Parallel Processing Architecture for Target Tracking*" in Seventh Asian Control Conference, 27-29 August, 2009, Hong Kong, China.

[3] H. Bao, X. Chen, Z. Wang, M. Pan, and F. Meng, "*Bounc-ing model for the table tennis trajectory prediction and the strategy of hitting the ball*" in 2012 International Confer-ence on Mechatronics and Automation (ICMA), 2012, pp. 2002 –2006.

[4] J. Liu, Z. Fang, K. Zhang, and M. Tan, "*Improved high-speed vision system for table tennis robot*," in 2014 IEEE International Conference on Mechatronics and Au-tomation (ICMA), 2014, pp. 652–657.

[5] H. Myint, P. Wong, L. Dooley and A. Hopgood (2015). "*Tracking a table tennis ball for umpiring purposes*" In: Fourteenth IAPR International Conference on Machine Vision Applications (MVA2015), 18-22 May 2015, Tokyo, Japan.

[6] J.O Rawlings., S.G. Pantula, and D.A. Dickey, "*Applied Regression Analysis: A Research Tool*" Second Edition, Springer-Verlag, 1998