122nd ASEE Annual
Conference & Exposition
June 14 - 17, 2015
Seattle, WA
Making Value for Society

Paper ID #13293

Development of Motion Analysis software for Dynamics Education

Dr. Hirohito Kobayashi, University of Wisconsin-Platteville

University of Wisconsin-Madison Ph.D. University of Wisconsin-Madison, M.S. Waseda University, Tokyo, JAPAN, B.S.

Development of Motion Analysis software for Dynamics Education

1 Abstract

As an attempt to introduce “virtual” hands-on experience in dynamics course, a motion analysis software, Virtual Mechanics Laboratory (VML), was developed based on Matlab® Graphic-User-Interface. VML was created as the kinematic information measurement tool to be used in a class project environment. In the project with VML, first, the student will capture a digital video image of an object subjected to the complex motion with a high frame rate digital camera that is widely available today. As the second step, the student will evaluate the kinematics, position and angle, of the object with digital motion tracking algorithm within VML. The kinematic information deduced from the motion tracking can be exported as the data in Microsoft® Excel format. The data can then be used to evaluate other kinematic information such as velocity, acceleration, angular velocity, and angular acceleration through proper manipulation. Finally, the student is tasked to evaluate key kinetic information such as the applied forces or moments by setting up and solving equations mastered in dynamics course. The software was completed in fall 2014 and tested through an end-of-semester mini-team project in one of the two dynamics classes. The efficacy of the “virtual” hands-on learning experience in dynamic via VML was assessed by in-class survey and the statistical analysis conducted on the final examination scores. The mean score of the final examination of the class conducted the mini project has improved slightly in comparison to the other class without mini-project. This score improvement was not significant based on statistical analysis. However, the in-class survey showed enthusiastic response from the students.

2 Introduction

Many studies indicate the importance of the hands-on experience in engineering or physics education[1,2]. However, in most colleges, dynamics, an import part of mechanics education, has been traditionally taught in lecture format without hands-on laboratory experiments due to the budget, space, and time limitation. Therefore, most students experience the difficulty of making the connection between theory and the real world examples, which then hampers student to achieve better comprehension.

To resolve the issue of the missing hands-on experience in the dynamics or physics lecture, many instructors have attempted to introduce a “virtual” hands-on learning experience by utilizing software. Dede, et al. presented the possibility of teaching Newtonian mechanics through a simulator named “NewtonianWorld”[3]. Jimoyiannis and Komis presented the efficacy of “virtual” hands-on learning with simulation software named “Interactive Physics” developed by Design Simulation Technologies Inc., Canton, MI[4]. However, the simulator-based tool only allows the user to experience limited number of problems that are defined in the relatively simplified physical setting. If a complex motion of an object can be captured easily with a digital camera and analyzed with a motion analysis software, we may be able to introduce a digital image technology based hands-on experience into dynamics lecture effectively. Due to the advancement of digital camera and the maturation of the motion tracking algorithms, we can easily achieve this goal with a low-cost digital camera and a personal computer today. The aim of this project is to develop Virtual Mechanics Laboratory (VML) that can be used as the tool to provide a “virtual” hands-on learning experience in the college level dynamics courses.

3 Object tracking in digital video image

The most important component of VML is the dynamics module that is established upon the object tracking algorithm. In last three decades, object tracking algorithm has matured through intensive research conducted in the field of computer or machine vision[5]. The proliferation of high-powered computers, the availability of high quality and inexpensive digital cameras, and the increasing need for automated video motion tracking lead to a generation of many object tracking algorithms. Despite that many different object-tracking algorithms have been developed, they can be regarded as derivatives of the following three major object-tracking algorithms.

3.1 Optical Flow Tracking

Optical Flow is a bundle of displacement vectors which defines the translation of each pixel in a region. It is computed using the intensity (brightness) constraint, which assumes intensity constancy of corresponding pixels in adjacent image frame[6,7]. Consider a single image pixel with intensity *I(x,y,t)* located within the two-dimensional image frame that was captured at time *t*. Later, the same pixel moved to a new location (*x+dx*, *y+dy*) in the adjacent image frame that was captured at time *t+dt* without changing the pixel intensity. By assuming the small spatial and temporal changes, Taylor series of the pixel intensity at *t+dt* is given by

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + H.O.T.\ . \qquad (1)$$

Since the pixel intensity *I* is assumed to be constant at all time in this method, above series will be simplified to

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = I_x\Delta x + I_y\Delta y + I_t\Delta t = 0. \qquad (2)$$

Here, (I_x, I_y, I_t) and Δt represent the measurable gradients of the intensity and the known time increment between image frames. eq.(2) is unsolvable due to the two unknown pixel displacements $(\Delta x = u, \Delta y = v)$. Lucas and Kanade [7] solved this issue by assuming of a uniform pixel displacement neighborhood (field) around the target pixel. In their method, a system of simultaneous optical flow equations (2) with two unknown displacements (u,v) are setup around target pixel and solved with Least Square Method. If the target pixel is located at the center of 3 by 3 image neighborhood, then the nine the optical flow equations can be constructed as,

$$\begin{bmatrix} I_{x1} & I_{y1} \\ \vdots & \vdots \\ I_{x9} & I_{y9} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{Bmatrix} I_{t1} \\ \vdots \\ I_{t9} \end{Bmatrix} \Delta t\ . \qquad (3)$$

Finally, the pixel displacements (*u*,*v*) are evaluated by applying Least Square Method to equations (3).

3.2 Block Matching Tracking

Block Matching algorithm is developed to locate matching block in a sequence of digital video frames for the purposes of object tracking. In this algorithm, a simple shaped region of interest (ROI), mostly rectangular "block", is defined around the target object or pixel on the current image frame (Fig.1). On the adjacent image frame, a "search region" that is larger than the ROI is defined at the same location. Finally, the ROI from the current frame is moved through the "search region" on the adjacent image frame until the matching block is located.
In the Block Matching algorithm, the object displacements (u,v) are evaluated as the optimal solution for minimizing a cost function. This minimization process is very similar to the process implemented in Digital Image Correlation algorithm described in the next section. Hence, the detailed description of the function minimization is left to the next section. There are various cost functions, of which the most popular is Mean Squared Error given by

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(C_{ij} - R_{ij}\right)^2. \tag{4}$$

Here, the block in current image frame and adjacent image frame are represented by C_{ij} and R_{ij} respectively. The degree of match is judged by checking the magnitude of norm difference between ROI and matching image[8,9].

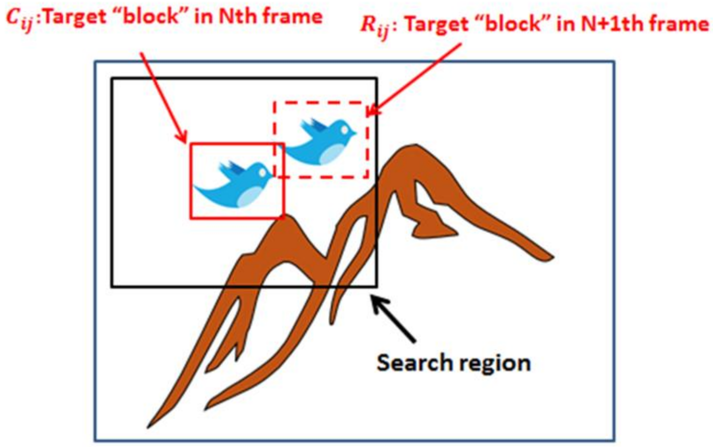


Figure 1 The target "block" and search region in Block Matching Tracking

3.3 Digital Image Correlation

Optical Flow and Block Matching algorithms are relatively easy to understand and implement. In addition, they are very effective to track the object 1) under rigid body translation and 2) without deformation. However, most real-world motion involves some degrees of rotation and deformation that cannot be tracked with above two methods. Hence Digital Image Correlation (DIC)[10,11] that is implemented in VML was developed to address the issue of object rotation and deformation. With the help of finite strain theory and nonlinear optimization theory, the concept of DIC can be described as follows.

The relation between the non-deformed object and the object subjected to the general rigid body motion and the deformation is presented in fig.2. Consider a point P in the non-deformed original body translated to point p through rigid-body translation $\vec{U}$. Due to the additional rigid body rotation and deformation, the target point Q displaced to point q. Subsequently, the small element vector $d\vec{X}$ changed to $d\vec{x}$.

The position vectors of point Q and q are given by

$$\vec{R} = \vec{X} + d\vec{X} = X\boldsymbol{i} + Y\boldsymbol{j} + dX\boldsymbol{i} + dY\boldsymbol{j} \qquad \textbf{(5)}$$

and

$$\vec{r} = \vec{x} + d\vec{x} = \vec{X} + d\vec{X} + \vec{x} + d\vec{x} = \vec{X} + d\vec{X} + \vec{U} + d\vec{U} = \vec{x} + \left(d\vec{X} + d\vec{U}\right)$$
$$= X\boldsymbol{i} + Y\boldsymbol{j} + dX\boldsymbol{i} + dY\boldsymbol{j} + u\boldsymbol{i} + v\boldsymbol{j} + du\boldsymbol{i} + dv\boldsymbol{j}. \qquad \textbf{(6)}$$

By comparing the position vector $\vec{R}$ and $\vec{r}$, the small element vector $d\vec{x}$ at deformed state is derived as

$$d\vec{x} = d\vec{X} + d\vec{U} = d\vec{X} + \overrightarrow{\nabla_X}\vec{U} \cdot d\vec{X} = \left(I + \frac{d\vec{u}}{d\vec{X}}\right) d\vec{X} \qquad (7)$$

or

$$\begin{Bmatrix} dx \\ dy \end{Bmatrix} = \begin{bmatrix} 1 + \frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} \\ \frac{\partial v}{\partial X} & 1 + \frac{\partial v}{\partial Y} \end{bmatrix} \begin{Bmatrix} dX \\ dY \end{Bmatrix} \qquad (8)$$

in matrix form.

The distance between two pixels in the non-deformed configuration is fixed and known. Hence dX and dY in eq.(8) can be replaced by ΔX and ΔY respectively. Therefore the coordinate of a point in the deformed configuration with respect to non-deformed configuration is given by

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{Bmatrix} X \\ Y \end{Bmatrix} + [\beta] \begin{Bmatrix} \Delta X \\ \Delta Y \\ 1 \end{Bmatrix}. \qquad (9)$$

Inversely, the point coordinate in the non-deformed configuration with respect to deformed configuration is given by

$$\begin{Bmatrix} X \\ Y \end{Bmatrix} = \begin{Bmatrix} x \\ y \end{Bmatrix} - [\beta] \begin{Bmatrix} \Delta X \\ \Delta Y \\ 1 \end{Bmatrix}. \qquad (10)$$

Here Affine transformation, matrix β, is given by

$$[\beta] = \begin{bmatrix} \beta_1 & \beta_3 & \beta_5 \\ \beta_2 & \beta_4 & \beta_6 \end{bmatrix} = \begin{bmatrix} 1+\frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} & u \\ \frac{\partial v}{\partial X} & 1+\frac{\partial v}{\partial Y} & v \end{bmatrix}.$$

In DIC, the parameters in eq.(9) or (10) be can be interpreted as follows.
X,Y: The known coordinate of the origin of the local coordinate in the current image frame.
ΔX, ΔY: The X and Y component of the known distance between two neighboring pixels in the current image frame.
β : The Affine transform that represents the rigid body translation, rotation and deformation of the tracked object.

The Block Matching algorithm described in the previous section assumes that tracked object is subjected to pure translation. Hence, the coordinate of a point in adjacent image with respect to the current image is given by

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{Bmatrix} X \\ Y \end{Bmatrix} + \begin{Bmatrix} u \\ v \end{Bmatrix} \tag{11}$$

.

In DIC, the displacements (u,v) of the tracked ROI (Number of image pixels =N) are evaluated as the solutions for minimizing the cost function such as

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{N} \varphi(X_i, Y_i, \boldsymbol{\beta})^2 = \sum_{i=1}^{N} (C(X_i, Y_i) - R(x_i, y_i, \boldsymbol{\beta}))^2. \tag{12}$$

Here, the ROI in current image frame and adjacent image frame are represented by C_{ij} and R_{ij} respectively. Along with the displacement (u,v) of the ROI, the four strains $(\frac{\partial u}{\partial X}, \cdots, \frac{\partial v}{\partial Y})$ of the ROI are also deduced through the minimization of the cost function (12).
The optimal solutions for cost function $S(\boldsymbol{\beta})$ are evaluated by

$$\frac{\partial S(\boldsymbol{\beta})}{\partial \beta_k} = 2 \sum_{i=1}^{N} \varphi_i \frac{\partial \varphi_i}{\partial \beta_k} = 0 \quad (k = 1, \cdots, 6) \tag{13}$$

In a nonlinear system, above gradient equations do not have a closed solution. Instead, initial values must be chosen for the parameters. Then, the parameters are refined iteratively by successive approximation,

$$\beta_k \sim \beta_k^{l+1} = \beta_k^l + \Delta\beta_k \tag{14}$$

Here, l and $\Delta\beta_k$ represent the number of iteration and the parameter increments respectively. At each iteration the model, the final configuration of the tracked ROI in DIC, is approximated to a first-order Taylor series expansion about$\boldsymbol{\beta}^l$by

$$R(x_i, y_i, \boldsymbol{\beta}) = R(x_i, y_i, \boldsymbol{\beta}^l) + \sum_{k=1}^{6} \frac{\partial R(x_i, y_i, \boldsymbol{\beta}^l)}{\partial \beta_k} (\beta_k - \beta_k^l) = R(x_i, y_i, \boldsymbol{\beta}^l) + \sum_{k=1}^{6} J_{ik} (\beta_k - \beta_k^l)\,. \tag{15}$$

Here, the Jacobian, J_{ik}, is given as the function of known pixel coordinate in current image frame and changes from one iteration to the next. Thus, Jacobian with respect to current image frame is derived as

$$J_{iK} = -\frac{\partial\varphi(X_i,Y_i,\beta_K)}{\partial\beta_K} = -(\frac{\partial C(X_i,Y_i,\beta_K)}{\partial\beta_K})=-\frac{\partial C(X_i,Y_i,\beta_K)}{\partial X_i}\frac{\partial X_i}{\partial\beta_K} - \frac{\partial C(X_i,Y_i,\beta_K)}{\partial Y_i}\frac{\partial Y_i}{\partial\beta_K}$$
$$= \frac{\partial C(X_i,Y_i,\beta_K)}{\partial X_i}\Delta X_K + \frac{\partial C(X_i,Y_i,\beta_K)}{\partial Y_i}\Delta X_K \qquad (16)$$

where $\Delta X_K = [\Delta X, \Delta Y, 1]$.

Finally, the increment of parameter $\Delta\beta_s$ for next step is evaluated with updated Jacobian and the image intensity differences of between current and adjacent image frame by

$$\sum_{i=1}^{N}\sum_{s=1}^{6} J_{ik}J_{is}\Delta\beta_s = \sum_{i=1}^{N} J_{ik}(C(X_i,Y_i) - R(x_i,y_i,\boldsymbol{\beta})) \qquad (17)$$

or

$$\boldsymbol{J}^T\boldsymbol{J}\Delta\boldsymbol{\beta} = \boldsymbol{J}^T\Delta\mathbf{I} \qquad (18)$$

in matrix form. This iteration is repeated until the increment vectors cross predefined threshold.

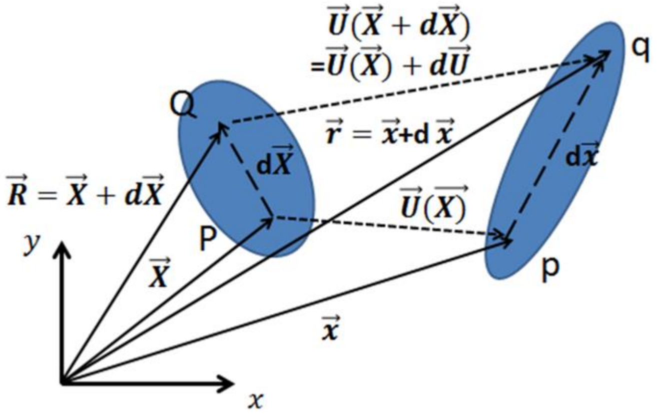


Figure 2 The relation between non-deformed object and object at to post general rigid body motion and deformation.

4 Overview of the Virtual Mechanics Laboratory (VML)

VML (Fig.3) is a Graphics-User-Interface (GUI) based software created in MATLAB®. MATLAB® is a multi-paradigm numerical computing environment and fourth-generation programming language developed by MathWorks, Natick, Massachusetts.
VML consists of following three major modules that act independently.

- **Video Edit Module:** In this module, users can edit the captured digital video image to a proper video length so that the optimal motion and mechanics analysis can be achieved in other two modules.

- **Dynamics Module:** In this module, users can conduct 1) motion analysis (point and vector tracking), 2) data export in Microsoft Excel ® format, and 3) videotaping the composite image frames of the tracked points and vectors overlay over original image.
- **Mechanical of Material Module:** In this module, users can conduct material strain evaluation in a deforming material using the same digital image tracking algorithm implemented in Dynamics Module. Currently under construction.

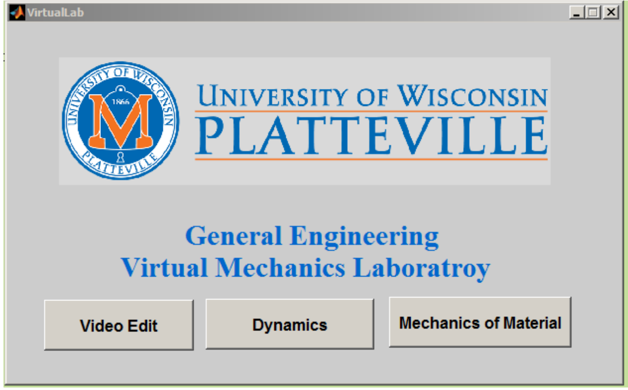
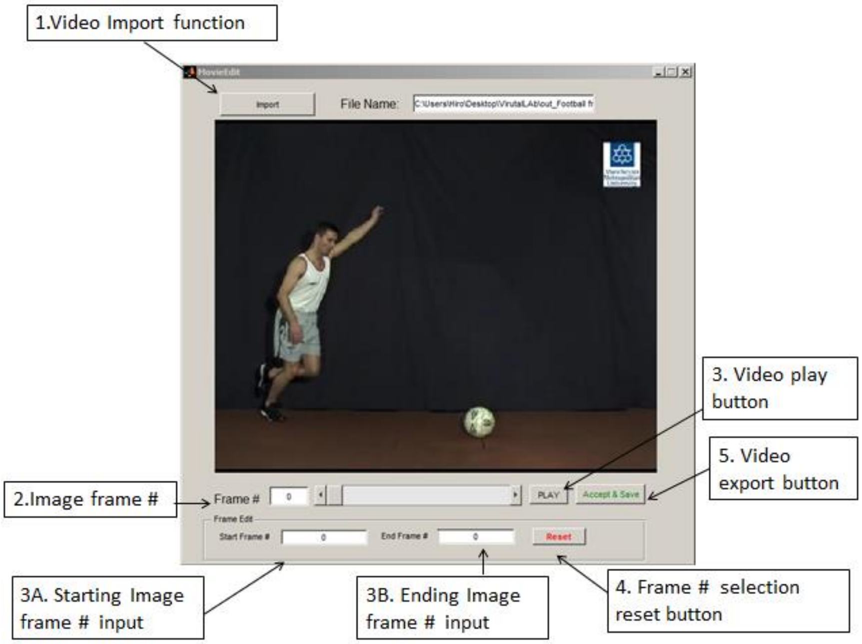


Figure 3 Stating menu of Virtual Mechanics Laboratory



File Name: C:\Users\Hiro\Desktop\VirutalLAb\out_Football fr

Frame # 0

Start Frame # 0

End Frame # 0

Figure 4 The key functions and workflow in Video Edit Module

4.1 Video Edit Module (Fig.4)

In this module, users can clip the captured digital video image to the optimal length so that the meaningful motion analysis and mechanics analysis can be achieved in the Dynamics modules. For example, the user captures the video image of a soccer player kicking a ball to conduct motion analysis of human-ball interaction by treating the problem as the eccentric impact. Clearly, the most important image view range is the view range that is set around the ball. However, it is much easier for the user to videotape the entire process of soccer player 1) start running from resting position, 2) reaching maximum speed, and 3) kicking the ball. Hence, the captured video image may contain numerous irrelevant frames, in which only the still ball is visible, therefore, should be removed by utilizing the Video Edit module. The typical workflow of video editing consists of following steps.

- **Video file import:** Users can import video saved in most types of video format.
- **Identifying the essential segment of the video:** The essential segment of the video can be identified by "Play" video feature.
- **Video clipping**: The video is clipped down to the essential segment.
- **Video saving and export**: The video is saved and exported as AVI format video.

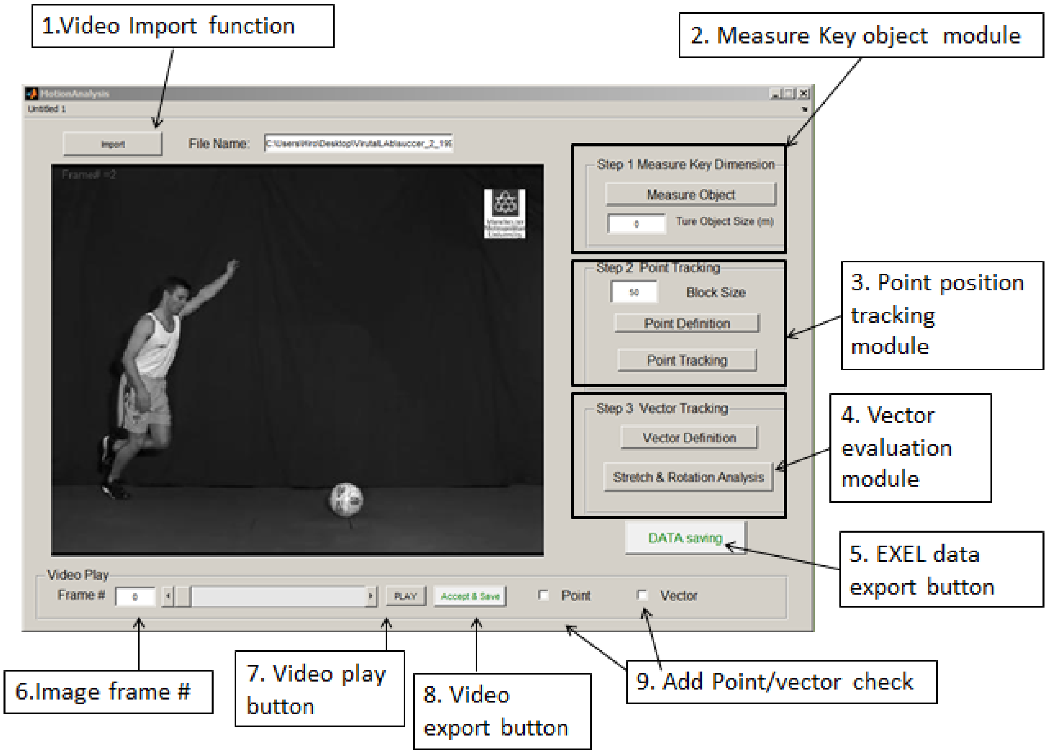


Figure 5. Overview of Dynamics Module functions and workflow.

4.2 Dynamics Module (Fig.5)

In Dynamics module, users can conduct motion analysis on imported video images to evaluate the basic kinematic information such as <u>position</u> and <u>angle</u> of the target object. Digital Image Correlation[10,11] algorithm is used for object tracking in this module. The deduced kinematic information can be exported as Microsoft Excel® format data for further analysis outside VML. In the class project, users are expected to evaluate important kinematic information such as velocity, acceleration, angular velocity, and angular acceleration by processing exported data. The typical workflow of motion analysis in Dynamics Module is as follows.

- **Video import and frame rate input:** Video file that is clipped and saved with Video Edit Module is imported into Dynamics module. Subsequently, users are required to key in the video frame rate that will be used to calculate the timestamp of each frame.
- **Space dimension calibration:** Users are required to 1) measure the pixel size of the object of known size by mouse clicking and 2) input true space dimension(*m*, *ft*) of the object, so that pixel-to-space dimension ratio can be evaluated and exported as the part of Excel® format data.
- **Point selection and tracking:** Users are required to input 1) the block size (default setting 60 by 60 pixel block) for object tracking and 2) select points of interest by mouse clicking action. Once the points have been selected, VML will estimate and present the total time required for tracking all points through all image frames. If the estimated total processing time (sec) is not acceptable, users can repeat 1) and 2) to achieve shorter processing time. Otherwise, VML will initiate automatic object tracking. Generally speaking, a larger block size achieves more accurate tracking yet requires more tracking time.
- **Vector definition and the evaluation of vector rotation and length change**: The evaluation of the kinematic information such as angle, angular velocity and angular acceleration of a rotating object is an import part of dynamics. The object angle is evaluated by calculating the vector inner product of user-defined vectors and the unit vector pointing in x axis (Fig.6).

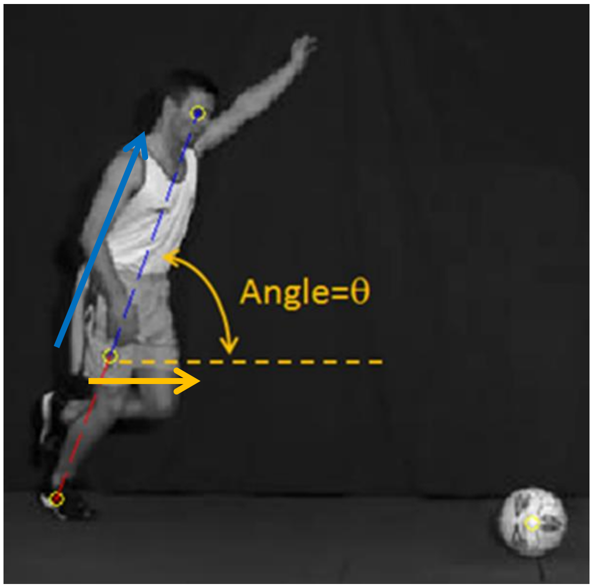


Figure 6 The vector angle in Virtual Mechanics Laboratory
Object angle is evaluated by the vector inner product of user defined vector (blue arrow) and the unit vector in x axis (yellow arrow).

- **Data export:** The deduced kinematic information can be exported as both Excel® format and MATLAB® specific MAT format data that can be later manipulated outside the program.

- **Data review by video playback:** Once the point and vector tracking are complete, users can check the tracking result by playing back the video with tracked points and vectors overlay on each frame.
- **Videotaping and export:** Users can export video image with tracked pixels and vectors as the AVI format video.

Figure 7 Data review with video play-back function.

6 Mini team project conducted with Virtual Mechanics Laboratory

The first version of VML was completed in September 2014 and tested through a mini project in December 2014. In order to evaluate the efficacy of the "virtual" hands-on experience with VML, this project was implemented only in one of the two sections of dynamics instructed in fall 2014. The goal of the project is to analyze the soccer player kicking action based on the eccentric impact model by utilizing the kinematic information deduced from VML. The video captured from https://www.youtube.com/watch?v=lBMA2wWuqh8 was used for analysis.

First, students are tasked to figure out which points and vectors to be defined so that the proper kinematic information can be deduced for solving the eccentric impact problem. Later, students are expected to evaluate 1) the **coefficient of restitution (*e*)** at the interface of soccer ball and shoe and 2) the **mass moment of inertia (I)** of the kicking leg by setting up necessary relations with the kinematic information that was deduced from VML. In order to solve the problem, students were provided with following parameters.

1) Mass of a soccer ball=0.5kg
2) Length of leg (knee-foot)=0.56m
3) Diameter of the ball =0.22m.

In addition, the kicking leg is assumed to be under pure rotation around fixed knee during the short period roughly about the impact time.

In order to solve this eccentric impact problem, students are required to measure the position of ball and foot or position of ball and angle of knee-to-foot vector from pre- to post- impact time points. All students were able to select and track the position of the ball. Most students encountered the difficulty in tracking the knee due to the lack of image texture. Hence, some students tracked the logo on the player's pants instead and used logo-to-foot vector as the substitute for knee-to-foot vector. Other students simply tracked the foot and utilized the deduced foot kinematic information.

By observing the recorded video and plot data (fig.8), the frame number 490 at 0.1875sec and frame number 520 at 0.25sec are selected as the starting and ending time points of the impact process. During this period, the foot movement was mostly horizontal that allowed the calculation of velocity to be simplified. The position of the ball and the foot through impact period are presented in Fig.8.

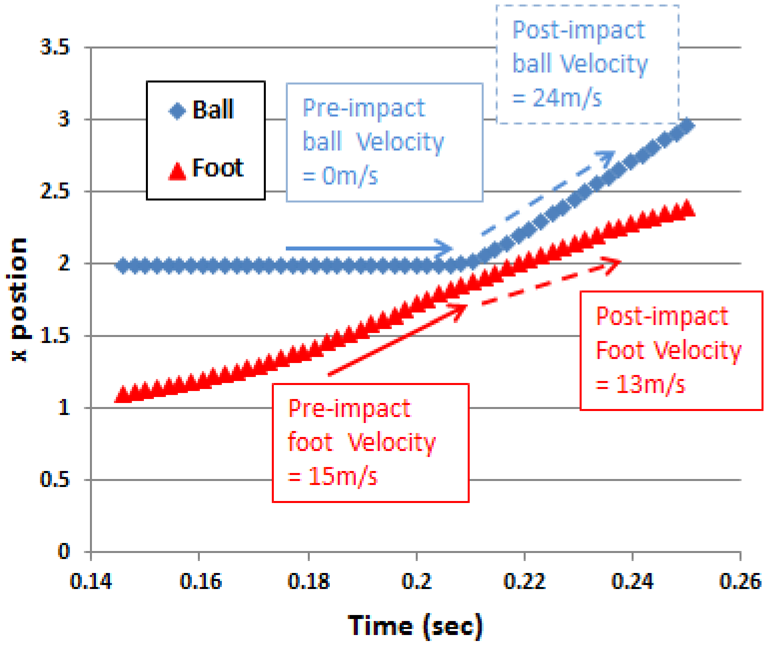


Figure 8 Measured positions of ball and foot through impact period

Finally, students were tasked to evaluate the coefficient of restitution and the mass moment of inertia of the leg by setting up two equations by substituting given parameters and velocities deduced from position information. In following relations, number 1 and 2 in the subscripts of parameter represents the pre-and post- impact time points respectively.

The coefficient of restitution:

$$Coeffieicnt\ of\ Restitution: e = \frac{V_{ball\_2} - V_{foot\_2}}{V_{foot\_1} - V_{ball\_1}} = \frac{24-13}{15-0} = 0.73$$

Conservation of the angular momentum:

$$H_{knee\_1} = I_{leg}\omega_{leg_1} + I_{ball}\omega_{ball_1} = I_{leg}\omega_{leg_2} + I_{ball}\omega_{ball_2}$$

$$= I_{leg}\frac{V_{foot\_1}}{L} + m_{ball}L^2\frac{V_{ball\_1}}{L} = I_{leg}\frac{V_{foot\_2}}{L} + m_{ball}L^2\frac{V_{ball\_2}}{L} = H_{knee\_2}$$

The mass moment of inertia of the leg is estimated to be $I_{leg} = 1.88 kg\ m^2$by solving conservation angler moment.

7. Assessment tool

In order to assess the efficacy of the "virtual" hands-on experience with VML, an in-class survey was conducted. During the survey, students were asked to answer following three questions. All questions were scored with a five-point scale (very positive (5) to very negative (1)). The results of the assessment are presented below with survey questions:

1) Did you find the mini-project conducted with Virtual Mechanics Laboratory to be interesting?
   Avg. score = 4.31
2) Did you find the mini-project conducted with Virtual Mechanics Laboratory enhanced the understanding of the impact problem? Avg. score =4.12
3) Did you find Virtual Mechanics Laboratory to be useful and can be implemented in more topics in dynamics? Avg. score = 4.57

The score of the final examination from the section administered with mini project and section without project were also compared to examine the efficacy. The average and standard deviation of the final exam score from two sections were evaluated as follows.

Average final exam score of the implemented section $= 83.08 \pm 8.68$
Average final exam score of the non-implemented section $= 81.69 \pm 11.56$

No statistical differences in final examination scores were observed between the implemented and non-implemented section.

Finally, two students out of 30 students in the implemented section presented strong interest in the Virtual Mechanics Laboratory and initiated student projects in spring of 2015.

8. Summary

To enhance the understanding of dynamics course, a Graphics-User-Interface based motion analysis software, Virtual Mechanics Laboratory, was developed based on MATLAB®. Virtual Mechanics Laboratory aims to provide "virtual" hands-on experience in dynamics course. Students are expected to conduct motion analysis using the digital video image of a real-world object under complex motion. The efficacy of the concept is tested by implementing the software in one of the two dynamics classes instructed in fall 2014. As the assessment tool, 1) an in-class survey and 2) the score of the final examination from the section administered with mini project and section without project were also compared. The result of the in-class survey presented a strong positive response while the final examination score between the project implemented and non-implemented sections did not show significant differences. A longer trial period and more data collection are expected in the near future to investigate the efficacy of the "virtual" hands-on experience in dynamics via Virtual mechanical Laboratory. Also as a final point, the mechanics of material module is under construction and to be tested in fall 2015.

9. References

1. Coller, B. D., “An Experiment in Hands-On Learning in Engineering Mechanics: Statics” , International Journal of Engineering Education, Volume 24, Number 3, 2008, pp. 545-557.
2. Carlson, L,E. and Sullivan J.F., “Hands-on Engineering : Learning by Doing in the Integrated and Learning Program*”International Journal of Engineering Education, Volume 15, Number 1, 1999, pp. 20-31
3. Dede, C.J., Salzman M. and Loftin, R.B.,“The Development of a Virtual World for Learning Newtonian Mechanics”, Multimedia, Hypermedia, and Virtual Reality: Models, Systems, and Applications, First International Conference, MHVR '94, Moscow, Russia, September 14-16, 1994, Selected Papers
4. Jimoyiannis A. and Komis V., “Computer simulation in physics teaching and learning: a case study on student’ understanding of trajectory motion”., Computer & Education Vol. 36 pp.183-204, 2001
5. Sonka M, Hlavac V and Boyle R, *Image Processing, Analysis, and Machine Vision*, Chp.16, Cengage Learning, 2014
6. B.K.P. Horn and B.G. Schunck, "Determining optical flow." Artificial Intelligence, vol 17, pp 185–203, 1981.
7. B. D. Lucas and T. Kanade, “ An iterative image registration technique with an application to stereo vision.” Proceedings of Imaging Understanding Workshop, pp. 121--130
8. Renxiang Li, Bing Zeng, and Ming L. Liou, “A New Three-Step Search Algorithm for Block Motion Estimation”, IEEE Trans. Circuits And Systems For Video Technology, vol 4., no. 4, pp. 438-442, August 1994
9. Jianhua Lu, and Ming L. Liou, “A Simple and Efficient Search Algorithm for Block-Matching Motion Estimation”, IEEE Trans. Circuits And Systems For Video Technology, vol 7, no. 2, pp. 429-433, April 1997
10. T.C. Chu, W.F. Ranson, M.A. Sutton, W.H. Peters,” Applications of digital-image-correlation techniques to experimental mechanics ”, Exp Mech 25 (1985) 232-244.
11. H.A. Bruck, S.R. McNeill, M.A. Sutton, W.H. Peters,” Digital image correlation using Newton-Raphson method of partial differential correction” Exp Mech 29 (1989) 261-267