Video Modeling: Combining Dynamic Model Simulations with Traditional Video Analysis

Douglas Brown
Cabrillo College, Aptos, CA   dobrown@cabrillo.edu

Introduction
The free Tracker video analysis program allows users to overlay simple dynamic particle models on a video clip. In a typical video modeling experiment students capture and/or open a digital videofile, calibrate the scale, and define appropriate coordinate axes just as for traditional video analysis. But instead of tracking objects with a mouse, students define theoretical force expressions and initial conditions for a dynamic model simulation that synchronizes with and draws itself on the video. The behavior of the model is thus compared directly with that of the real-world motion. Tracker uses the Open Source Physics code library so relatively sophisticated models are possible.

This poster showcases video modeling experiments conducted by students in my Fall 2007 Introductory Mechanics course. The assignment was very open-ended:

1. Students study real objects but focus on identifying forces and initial conditions in a "Model Builder" as shown in Figure 1. Function expressions are interpreted and evaluated using a parser which recognizes parameter and variable names, mathematical and logical operations, common math functions and if statements. The motion of the models is computed with an ODE solver using a Runge-Kutta algorithm with an adaptive step size.

2. Visual comparisons are rapid, intuitive and capable of discerning fine differences.

3. Visual agreement may validate models more convincingly than numerical or graphical comparisons for beginning students.

4. Visual disagreement leads naturally to explorations of limitations and refinements of models.

5. Quantitative analysis of model and/or video data can support and extend the visual comparison to any degree desired.

Building models
Students define dynamic models by entering force functions and initial conditions in a "Model Builder" as shown in Figure 1. Force expressions are interpreted and evaluated using a parser which recognizes parameters and variable names, mathematical and logical operations, common math functions and if statements. The motion of the models is computed with an ODE solver using a Runge-Kutta algorithm with an adaptive step size.

Figure 1. Model Builder for a falling monkey

Figure 2. Shooting a monkey falling with a parachute

Monkey with a parachute
One student group updated the traditional monkey/hunter demo by substituting a coffee filter for the falling monkey. Before trying to shoot this "monkey with a parachute," they modeled it as a particle experiencing gravitational and drag forces as shown in Figure 1. Using a similar model "as is," they then predicted the initial velocity angle needed to hit the monkey. After making their prediction, they hit the real coffee filter on the first try (Figure 2)!

Figure 3. Use of an if statement

Spring wars
Here students modeled a cart being pulled by opposing springs and slowed by friction. The Model Builder displays the use of a separately defined frictional force "f" and an if statement to specify its direction (Figure 3). The model displays a linear amplitude decay that agrees well with the motion of a real cart captured on video and marked with a mouse (Figure 4).

Figure 4. Frictional damping

Bouncing cart
This group modeled a cart bouncing from a spring at the lower end of a tilted air track. The "fx" force expression uses an if statement to apply the spring force only when the cart has reached the spring position "x" (Figure 5). The model cart was placed just above the real cart in the video for easy comparison as seen in Figure 6. The decreasing bounce heights were assumed to result from friction between the cart and track rather than from an imperfect spring.

Figure 5. Model Builder

Figure 6. Comparing the model (purple) with a real cart

Accelerating cart pendulum
What is the motion of a ball suspended from an accelerating cart? My students first discovered "experimentally" that the model ball’s acceleration was unmeasurable—they were able to relate the motion theoretically to a pendulum in a skewed gravitational field!

Figure 7. Ball suspended from an accelerating cart

Discussion
Computer modeling that engages students directly in the development of physical models is becoming an increasingly important part of physics education. Dynamic particle models based on Newton’s laws provide a natural first exposure to the modeling process for students in introductory physics.

Other modeling tools such as Easy Java Simulations also give students a relatively simple way to build particles (and many other) models and to control virtually every aspect of the modeling process. While Tracker’s video models are more limited, it’s ability to compare the models with real-world videos combines features of both traditional video analysis and animation-only modeling and offers these additional advantages:

1. Students study real objects but focus on identifying forces rather than tracking and analyzing motion.

2. Visual comparisons are rapid, intuitive and capable of discerning fine differences.

3. Visual agreement may validate models more convincingly than numerical or graphical comparisons for beginning students.

4. Visual disagreement leads naturally to explorations of limitations and refinements of models.

5. Quantitative analysis of model and/or video data can support and extend the visual comparison to any degree desired.

Acknowledgements
Many thanks to Wolfgang Christian and the Open Source Physics team Mario Belloni, Francisco Esquembre, Anne Cox and Bill Junkin for their invaluable expertise and guidance. The Open Source Physics project is supported in part by the National Science Foundation grant DUE-0442461.

References
1. Tracker: http://www.cabrillo.edu/~dbrown/tracker/

2. Open Source Physics: http://www.opensourcephysics.org/


