Motion Capture Analysis of Diabolo Juggling

Jeb Cui

TJHSST 2021-22: Computer Systems Research Lab

Introduction

The diabolo is an under-researched area of modeling the dynamic systems of juggling. Diabolo, or a Chinese yoyo, is similar to what a hand-size yoyo would be. However, there are several differences: the string is not attached, a pair of sticks control the diabolo, and the diabolo is hourglass shaped.

Von Drigalski et al. developed an analytical diabolo predictor that estimates the next state based on the current state and the stick positions using a constrained Euler's method. Murooka et al. created their algorithm, Diabolo-Manipulation-Net, and it's trained using the diabolo's pitch and yaw, the robot's arm height difference, and the robot's spin speed.

My research develops von Drigalski et al.'s research further by adding an algorithm that predicts the pitch of the diabolo. Using research by Rodriguez Ladron de Guevara et al. and mechanics, I also determined that the change in pitch is directly related to the ratio of offset and rotational speed. The algorithm uses Euler's method to predict pitch angle. I created a residual-physics learning neural network (R-PLNN) to improve the analytical model trained on its difference with real-world data.

Methods

I used TJ's Vicon Motion Capture system to gather diabolo performance data. The system has 9 Cameras, consisting of MX T20's and MX T40's, and is operated through the Vicon Tracker Software. My diabolo performances included tilting moves for changing pitch data. Each recording ranges from a minute to two minutes long.

The diabolo prediction model has two steps: the analytical model and the machine learning model. I transpiled von Drigalski et al.'s analytical model from C++ to Python. I did this to remove the C++ specific dependencies, make it integrat-able to a Flask web server, and connect it with Python machine learning libraries.

Figure 3 illustrates the pseudocode for my prediction algorithm. This rela-tionship is based on Rodriguez Ladron de Guevara et al.'s work and physics mechan-ics.

I used TensorFlow 2's Keras to develop the R-PLNN and SciKit Learn for creating the training dataset. I created the training dataset by running von Drigalski et al.'s and my pitch predicting algorithms. The attributes were current diabolo posi-tion, velocity, and rotational velocity, and the current and next stick positions. The labels were the offset between the diabolo position and pitch (orientation) compared to the real-life values. The neural network was a sequential model with three hidden layers of 100 nodes using the ReLu activation function. I trained the model over 300 epochs using a batch size of 100.

My site has a Flask backend. Additionally, there's also a Gazebo server for operating the graphical display of the model's output. The Flask backend includes my model and Vicon Data streaming using SocketIO.

Figure 1. A blue diabolo.

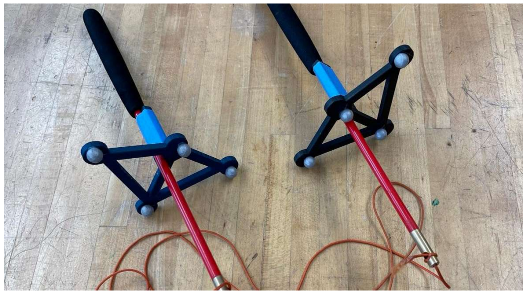

Figure 2. Sticks with retroreflective markers.

Algorithm 1 Pitch predicting algorithm

Require: Q = Quaternion angles, ω = rotational speeds, R, L = right and left stick positions

function PITCH-PREDICTION(Q, ω, R, L)
  $|\text{Pitches}| \leftarrow n$ ▷ n is total number of time steps
  $|\text{Offsets}| \leftarrow n$
  for $t \leftarrow 1 \dots t_n$ **do**
    $d \leftarrow Q_t \langle 0, 0, 1 \rangle Q'_t$ ▷ Quaternion rotation
    $h \leftarrow \frac{\langle d_1, d_2, 0 \rangle}{\| \langle d_1, d_2, 0 \rangle \|}$
    $\Delta s \leftarrow R_t - L_t$
    $\Delta s \leftarrow \frac{\langle \Delta s_1, \Delta s_2, 0 \rangle}{\| \langle \Delta s_1, \Delta s_2, 0 \rangle \|}$
    $\text{Offsets}_t \leftarrow cos^{-1}(\Delta s \cdot h) - \pi/2$
    if $t = 1$ **then**
      $\text{Pitches}_1 \leftarrow cos^{-1}(d \cdot h)$
      if $d_3 \leq 0$ **then**
        $\text{Pitches}_1 \leftarrow -\text{Pitches}_1$
      end if
    end if
  end for
  SAVITZKY-GOLAY FILTER(Offsets, 101, 5)
  for $t \leftarrow 1 \dots t_{n-1}$ **do**
    $\frac{dP}{dt} \leftarrow k \cdot \frac{\text{Offsets}_t}{\omega_t}$ ▷ k is the empirical constant of proportionality
    $\text{Pitches}_{t+1} \leftarrow \text{Pitches}_t + \frac{dP}{dt} \Delta t$
  end for
  return Pitches
end function

Figure 3. Pitch predicting using the forward Euler method.

Results

The website contains these main functions: displaying data, running predic-tions using my model, and streaming Vicon motion capture data. Users can upload their data to simply display the movements or to run a prediction on them. My model has two substantial additions to von Drigalski et al's diabolo predictor: diab-olo pitch prediction and the use of an R-PLNN. The pitch predictor could inde-pendently closely predict the actual pitch of the diabolo. The R-PLNN's predictions has a mean squared error of 0.132 m² for the main section of the recordings I tested. However, the model is slow on the scale of minutes.

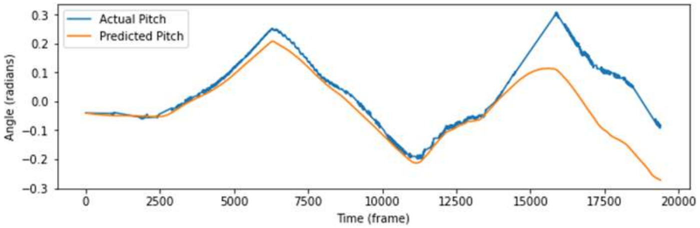

Figure 4. A graph of actual pitch values and predicted pitch values. The prediction diverges at the end, starting at the sharp jump.

Discussion and Conclusion

I was able to extend the prediction ability of von Drigalski et al.'s algorithm to include pitch, and the graphs align relatively close together. The analytical model by itself had a MSE of 0.771 m², and the R-PLNN addition reduced the MSE to 0.188 m². However, the analytical algorithm's shortcoming is its inefficiency. An-other limitation was that the Motion Capture system skip frames when the diabolo spun too fast, but I couldn't sacrifice too much resolution for FPS.

For the future, the research I did can be improved in these potential ways. Improving the efficiency of the analytical model, such as ways to optimize the size of timesteps automatically. Developing prediction for yaw, since a limitation to my testing was that even with my best efforts, I wasn't able to completely balance the markers and it introduced some yaw not inherently caused by my stick movements. Experimenting on fixed axle diabolos, since I used a triple bearing axle diabolo for my research.

Acknowledgements

I would like to thank Dr. Gabor, my lab director, for guiding me on the path to finishing my project.

I would like to also thank Mr. Kosek for his help and support with the Vicon Motion Capture System in the Web Application Development Lab.