A Machine Learning Based Long Jump Training Assistant

Jacob Sobota

Introduction

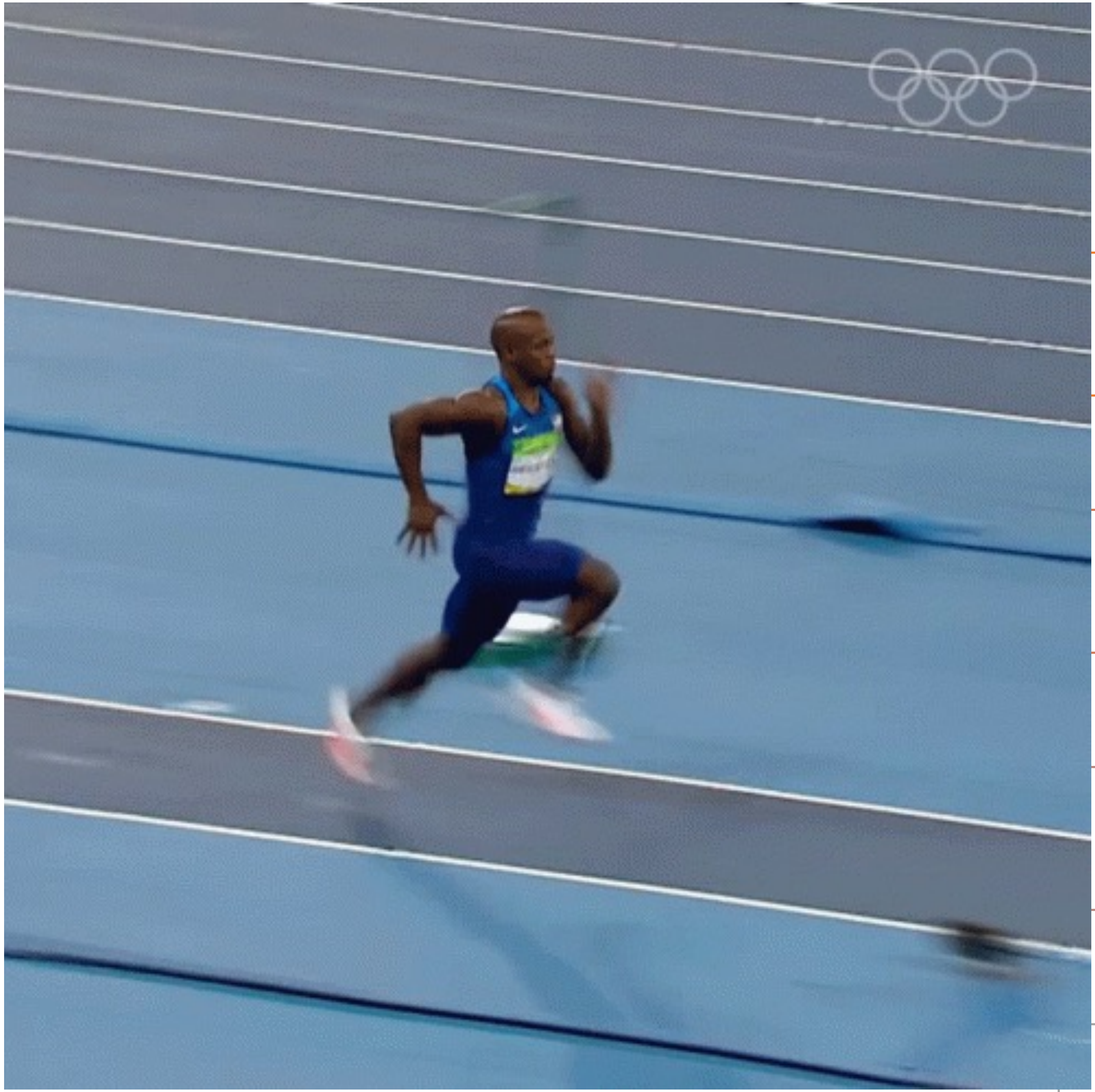
- The Long Jump is a technical event requiring speed, strength, and precise technique.
- Currently, long jump coaching is purely qualitative and up to the coach's interpretation
  - Any quantitative work is done at a later date with computers
- There are no available applications that offer real-time quantitative analysis of the jump

I propose an application that recognizes important frames and sequences from the long jump, and extracts relevant feature values to be compared to the objective bio-mechanical model

Research Goals

- Identify notable positions to measure in the long jump
- Identify, code, and extract features to aid in training machine learning models
- Build two machine learning models:
  - One for recognizing touchdown frame
  - One for recognizing takeoff sequence
- Combine ML models with easy-to-use framework to create application

Segments of the Long Jump

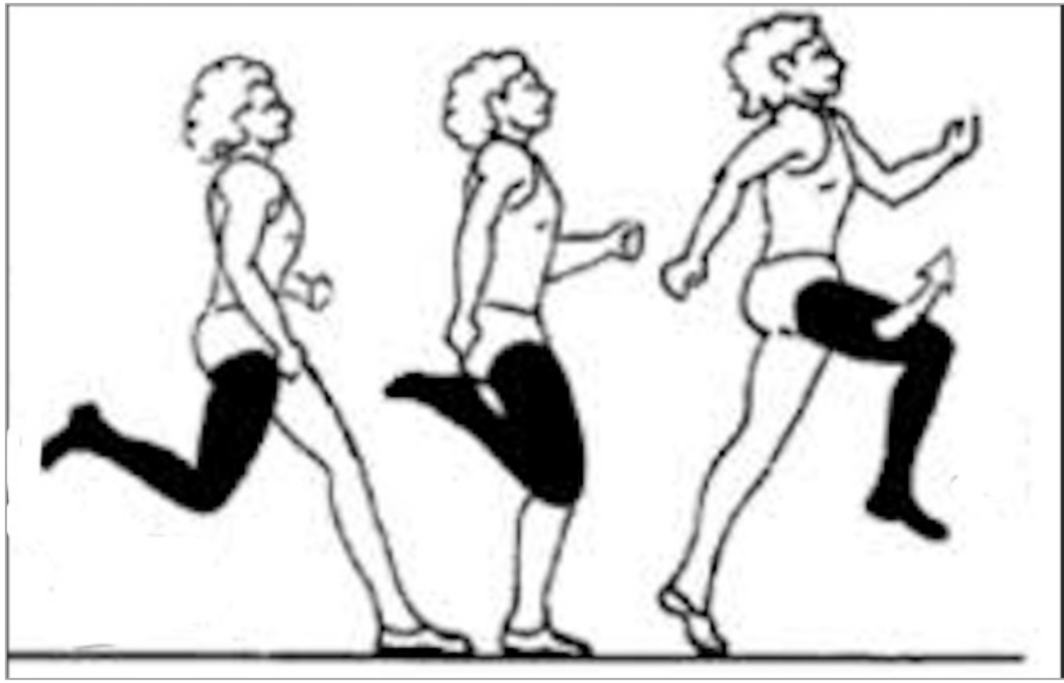


Jeff Henderson (USA): 2016 Olympic Champion

- Approach run
- Takeoff Sequence
- Flight
- Landing

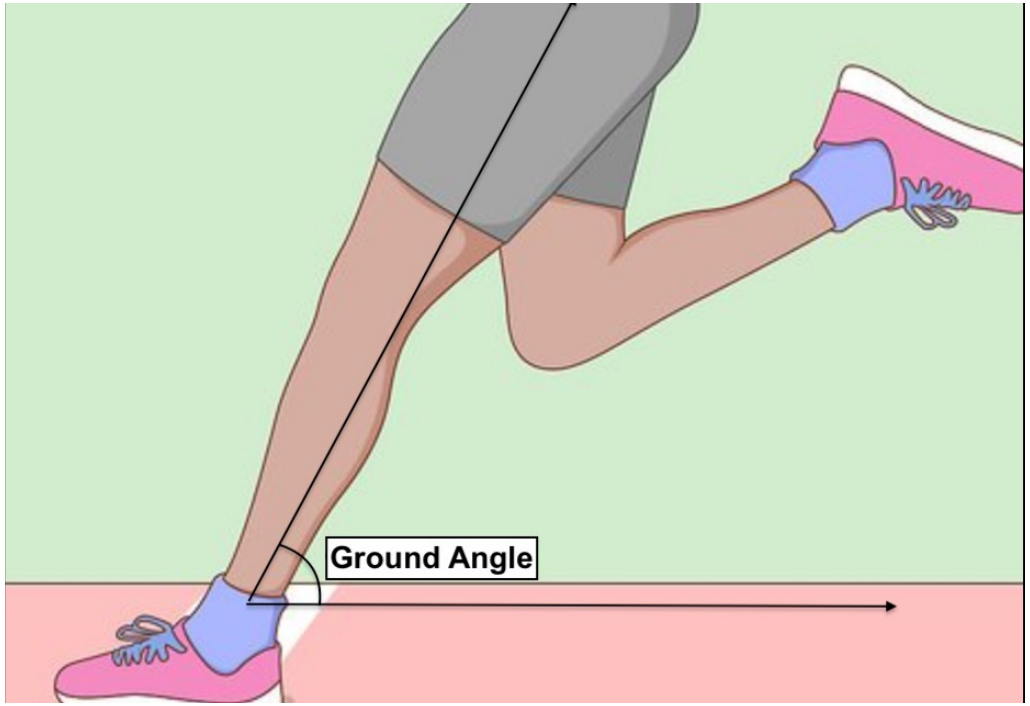
The Takeoff Sequence

- Touchdown: The instant the jump foot makes contact with the ground
- Takeoff: The instant the jump foot leaves the ground
- Takeoff Sequence: The athlete's movements between touchdown and takeoff.
- The athlete's velocity at takeoff is the greatest predictor of jump distance.
- So, what takeoff positions provide the most efficient transfer of speed from touchdown to takeoff?



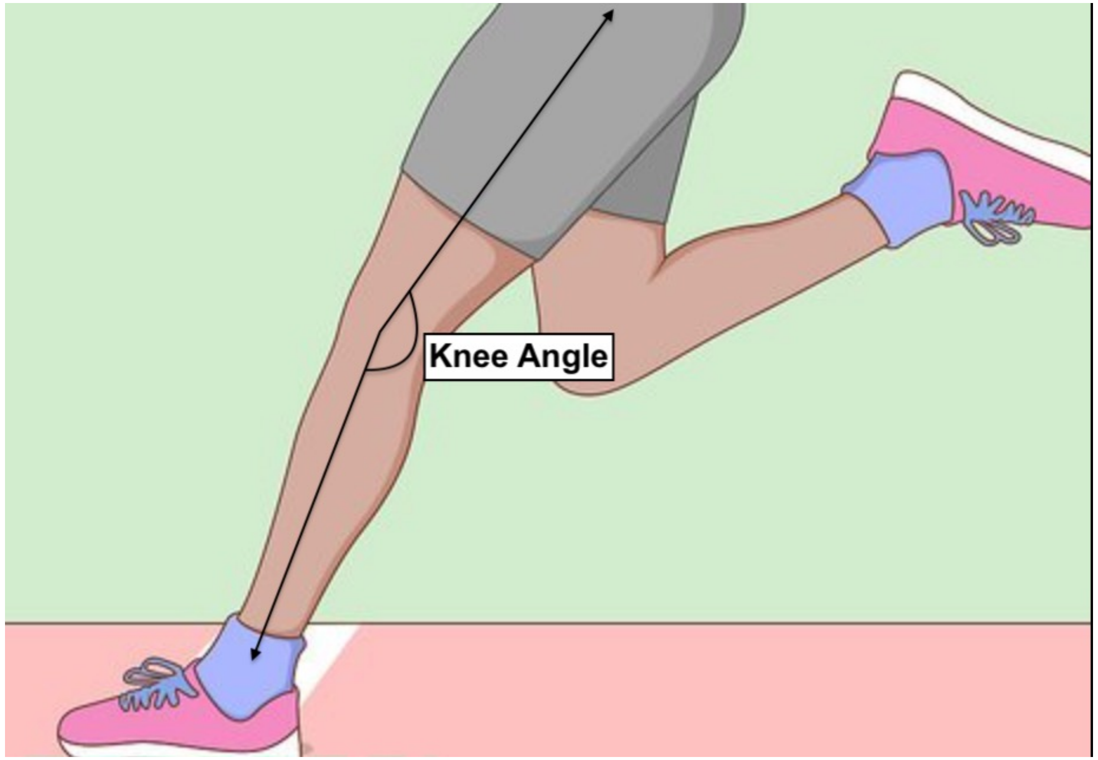
Touchdown Ground Angle

- Lees (1994) found the feature most strongly correlated with takeoff speed was the touchdown ground angle
- This angle is measured from the hip joint, to the ankle, to the horizontal of the jump leg
- Regardless of the athlete’s speed, the optimal touchdown ground angle is between 60 and 70 degrees (Bridgett 2006)



Touchdown Knee Angle

- The jump leg receives an enormous amount of force throughout the takeoff
- A straighter and stiffer knee handles load better than a bent/collapsed knee
- This angle is measured from the jump leg ankle, to knee, to hip
- The optimal angle range is between 165-170 degrees



Minimum Knee Angle

- No athlete is capable of maintaining their knee angle through the takeoff sequence.
- The best athletes will allow their knee angle to decrease as little as possible
- Takes place in the middle of takeoff sequence
- Alexander (1990) recommends the minimum knee angle should be no less than 135 degrees

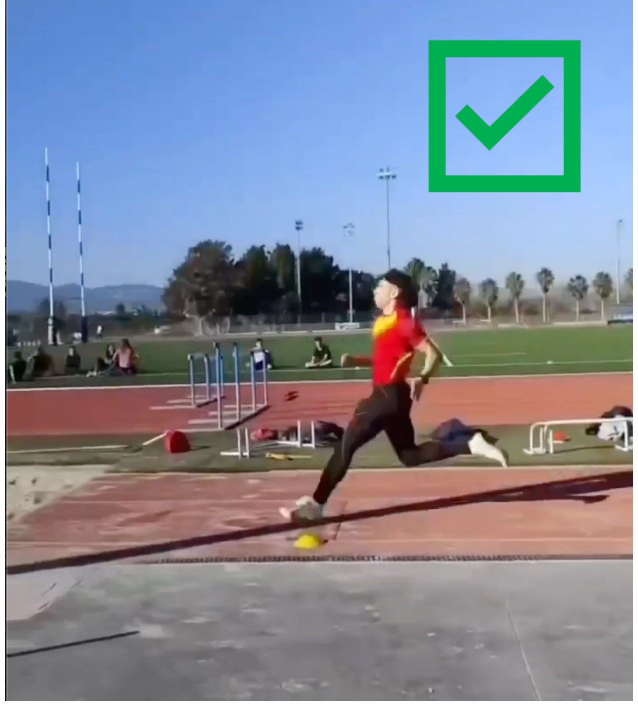
Mediapipe Pose Estimation

- Open-source software, but developed by Google
- Takes a video or image as input
- Predicts the location of the most prominent individual's joints and face features
- Returns these predictions as pixel coordinates
- These coordinates can be overlayed on top of the image, as seen here.
- This is the foundation of the application

Video Dataset

- 49 videos total
- Retrieved from:
  - University of Kentucky Track and Field
  - Professional Track and Field Athlete Instagram Accounts
  - @jumpersworld (Instagram Account)
- 23 Different Long Jumpers are Represented
- All videos meet following criteria

Robbie Springfield: University of Kentucky Long Jumper



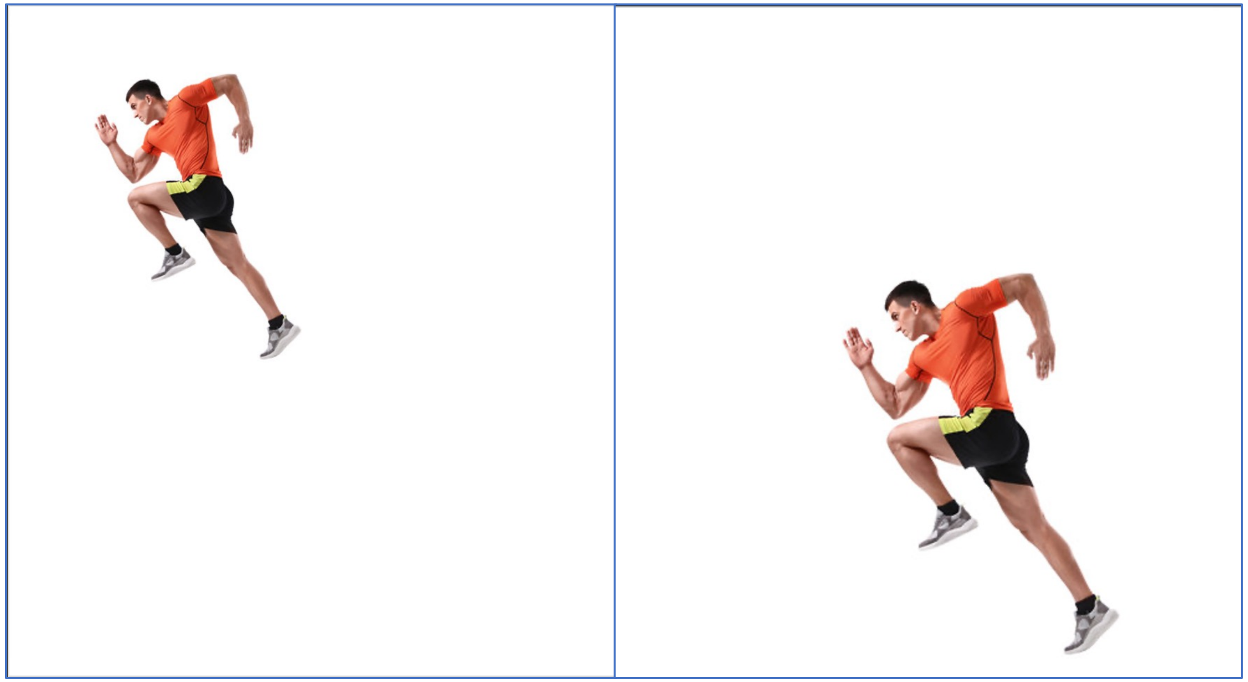
Video Criteria

- <u>Cameraman Position</u>: In line with the athlete during the takeoff sequence
- <u>Unobstructed View of Athlete</u>: Nothing blocking the athlete during the takeoff sequence
- <u>Running to the Left</u>: Athlete is running to the left on film. Videos that did not meet this criteria were horizontally flipped.
- <u>Functional with Mediapipe</u>: Mediapipe could accurately predict joints. Only a few videos failed. Otherwise, no restrictions on video quality

Feature Selection

- Mediapipe’s raw coordinate data would return drastically different results for each image.
- Must use features that would return the same result from each image
- Settled on a combination of limb angles and joint distance ratios
- These features are mostly inspired by existing long jump research

These two images each contain the same figure of a runner that has been moved and scaled

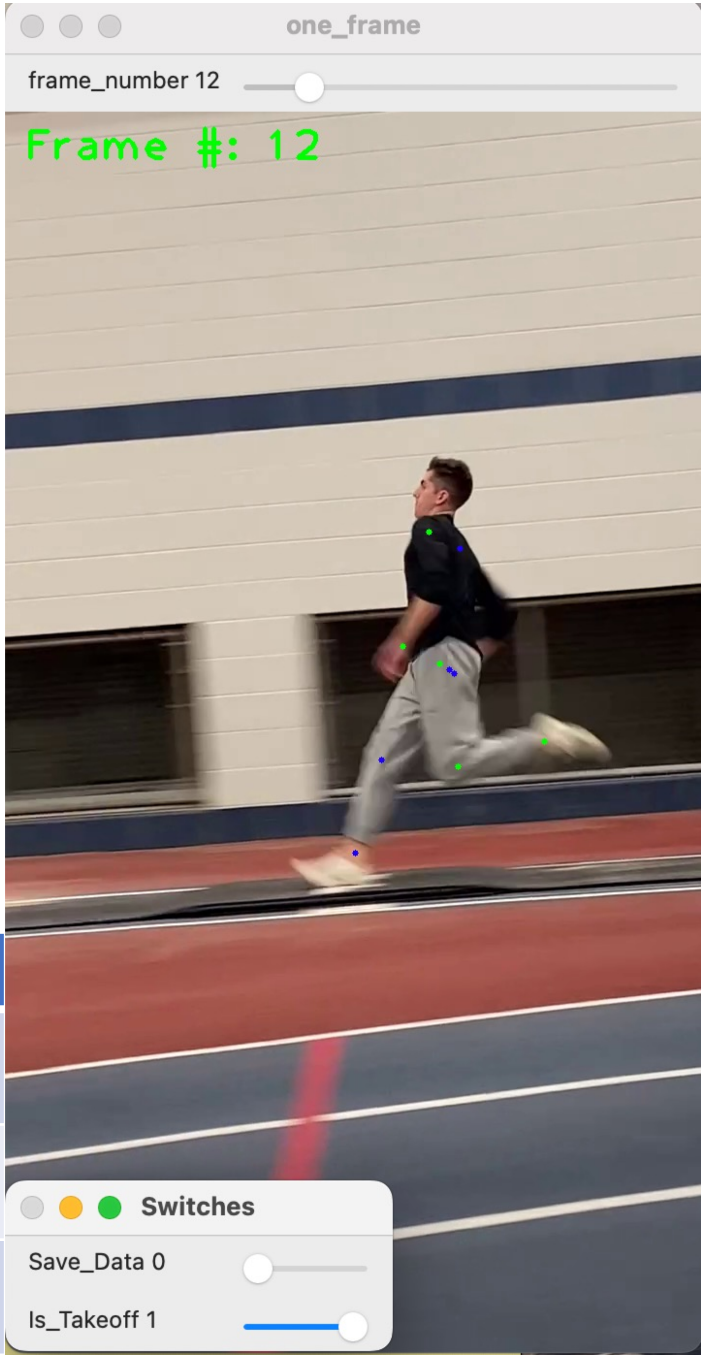


Feature	Description	Justification
Right leg knee angle	Angle formed by right ankle, right knee, right hip	This is an angle of interest that informs the quality of takeoff. See Related Work Section. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Right hip angle	Angle formed by right knee, right hip, right shoulder	This is an angle of interest that informs the quality of takeoff. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Left leg knee angle	Angle formed by left ankle, left knee, left hip	This is an angle of interest that informs the quality of takeoff. See Related Work Section. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Left hip angle	Angle formed by left knee, left hip, left shoulder	This is an angle of interest that informs the quality of takeoff. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Right foot ground angle	Angle formed by right hip, right ankle, and ground (horizontal)	This is an angle of interest that informs the quality of takeoff. Created by adding an invisible point horizontal to the ankle joint. See Related Work section. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Left foot ground angle	Angle formed by left hip, left ankle, and ground (horizontal)	This is an angle of interest that informs the quality of takeoff. Created by adding an invisible point horizontal to the ankle joint. See Related Work section. Used by Lees 1994, Seyfarth 2000, Panoutsakopoulos 2010.
Ankle Distance / Leg Length	Distance between each ankle divided by left leg length	Measures distance between each ankle joint. The greater this distance, the more likely the athlete is in a touchdown or takeoff position. Not directly utilized in research, but many papers use ankle distance to the hip, and x-coordinate distance between the two ankles. This combines the two into one measure
X distance between right hip and right ankle / right leg length	The x-coordinate distance from right hip to right ankle divided by right leg length	Measures distance of the ankle in front of the hip. When the value is larger, it indicates that the athlete is reaching the foot far out in front of them, which indicates a takeoff. Measured by Lees 1994
X distance between left hip and left ankle / left leg length	The x-coordinate distance from left hip to left ankle divided by left leg length	Measures distance of the ankle in front of the hip. When the value is larger, it indicates that the athlete is reaching the foot far out in front of them, which indicates a takeoff. Measured by Lees 1994
Right and left hip distance / left leg length	Coordinate distance between right and left hip divided by left leg length	Provides information on the relationship between camera and athlete. When hip coordinates are closer to each other, the athlete is likely in line with camera. If they are farther apart, the athlete is likely facing the camera. This was a personal inclusion that helped model performance
Right hip to right wrist distance / right leg length	Distance from right hip to right wrist divided by right leg length	Athletes often reach positions similar to touchdown while cycling their legs in the air post-takeoff. The main difference between these two positions are arm location. At touchdown, wrists are generally near the hip. In flight, at least one wrist is in the air above the head. Helps the model distinguish between the two. This was a personal inclusion that helped model performance
Left hip to left wrist distance / left leg length	Distance from left hip to left wrist divided by left leg length	Athletes often reach positions similar to touchdown while cycling their legs in the air post-takeoff. The main difference between these two positions are arm location. At touchdown, wrists are generally near the hip. In flight, at least one wrist is in the air above the head. Helps the model distinguish between the two. This was a personal inclusion that helped model performance
Forward foot	Indicates which leg is in front based purely on the x-coordinates	This feature helps the model distinguish left and right foot jumps and their common features. This was a personal inclusion that helped model performance
Takeoff foot	Indicates which leg is on bottom based solely on y-coordinates	This feature is more useful in identifying sequences. Provides information to the sequence model on which leg is the takeoff leg as the takeoff ankle will always be the bottom of the

Feature Extraction

- Performed manually with the pictured framework in OpenCV using Mediapipe
- Left-side joint estimates are green, right side joint estimates are blue
- Hidden Python code calculates feature values for each frame
- Then, sliders in the bottom left allow for frame labeling and saving feature values to CSV
- Each frame had to be saved individually
- Each dataset used roughly the same framework

Dataset	Rows	Columns
Touchdown Recognition Dataset	105	15
Takeoff Sequence Recognition Dataset (Frames Only)	327	3
Takeoff Sequence Recognition Dataset (Full Sequences)	61	2



Touchdown Recognition Data

right_knee_angle	right_hip_angle	left_knee_angle	left_hip_angle	right_ground_angle	left_ground_angle	ankle_distance	right_hip_distance	left_hip_distance	hip_distance	right_hand_hip_dist	left_hand_hip_dist	forward_foot	is_touchdown
104	112	166	174	86	106	0.9108	0.5132	-0.1653	0.0815	0.4216	0.7782	1	0
136	204	172	152	159	47	1.5487	-0.6910	0.7218	0.0082	0.6215	0.5551	0	0
98	199	162	147	-168	61	1.3764	-0.6576	0.5988	0.0525	0.6448	0.1002	0	1
153	205	143	126	152	61	1.4731	-0.7282	0.6925	0.0217	0.7859	0.4551	0	0
135	145	55	132	106	171	0.5363	0.0552	-0.1349	0.0476	0.5040	0.0731	1	0
102	201	164	148	-169	61	1.4304	-0.6982	0.5848	0.0150	0.4168	0.2931	0	1
142	208	153	127	164	54	1.5478	-0.7870	0.7321	0.0257	0.4923	0.4547	0	0

Touchdown Recognition Model

- Performed Z-score Normalization on the angle values of this dataset
- Uses Gaussian Naïve Bayes machine learning Model (Scikit-Learn)
  - Classifier
  - Probabilistic
  - Effective for small datasets
- Using a 70/30 train/test split, achieved 80%-91% test accuracy with different random seeds
- The model used in the application was trained on all 105 datapoints

Takeoff Sequence Recognition Raw Data

Frame Features	Video Number	Class Label
[143, 133, 54, 154, 84, -161, 0.9137625897991168, 0.37861068360312666, -0.29711697999255643, 0.07618384102373242, 0.22969143798832742, 0.3836520204249679, 1, 1]	3	Running
[135, 145, 55, 132, 106, 171, 0.5363333904215749, 0.05522870898473681, -0.13487563949833123, 0.04760316688176396, 0.5040343959435514, 0.07314658556239256, 1, 1]	3	Running
[125, 152, 76, 110, 131, 129, 0.45784197506534824, -0.24957010215164926, 0.17214589115901327, 0.03657396909185952, 0.6871624923856804, 0.19246495733941688, 0, 1]	3	Running
[132, 174, 109, 110, 147, 90, 0.9792095347888339, -0.5106181882516534, 0.4833115378189281, 0.04813420645316375, 0.7845727999536309, 0.3504223124123475, 0, 1]	3	Running
[98, 199, 162, 147, -168, 61, 1.3764305018531278, -0.657594467096523, 0.5988448940217928, 0.0525259835555807, 0.644844925638734, 0.10023396564537061, 0, 0]	3	Takeoff
[65, 169, 142, 148, -166, 89, 0.9538851837998036, -0.4044149547959074, 0.3123971725361309, 0.052066195422688484, 0.38656283366932565, 0.33955228814599425, 0, 0]	3	Takeoff
[46, 137, 138, 156, 176, 109, 0.6700000489660961, -0.15017621388797545, 0.0, 0.08409543714999052, 0.07613643141146119, 0.6622119038703979, 0, 0]	3	Takeoff
[62, 110, 168, 181, 131, 111, 0.7409835615738265, 0.15067980605460743, -0.2743563722782319, 0.05647124003834986, 0.24925997628704605, 0.8185139667003865, 1, 0]	3	Takeoff

Takeoff Sequence Recognition Training Data

Numeric Class Label	Features
0	[[130, 132, 89, 193, 72, -161, 1.300377131790072, 0.6121206839237116, -0.6075173213135778, 0.07942929939034747, 1.1063221224934636, 0.6935865390311152, 1, 1], [153, 141, 64, 190, 66, -145, 1.2750963789644199, 0.5760350525115393, -0.47088317797067086, 0.09716637005744001, 0.33367700266340883, 0.44970416680792, 1, 1], [173, 162, 48, 181, 67, -145, 1.0312991919031917, 0.4404622187773885, -0.31052839281672173, 0.07573863239432237, 0.0610624375866661, 0.13074515096992181, 1, 1], [169, 172, 41, 170, 81, -157, 0.8676327784902595, 0.22679851419248523, -0.21912347603086565, 0.05959372916099731, 0.04297364923721788, 0.25673764240296637, 1, 1]]
0	[[143, 133, 54, 154, 84, -161, 0.9137625897991168, 0.37861068360312666, -0.29711697999255643, 0.07618384102373242, 0.22969143798832742, 0.3836520204249679, 1, 1], [135, 145, 55, 132, 106, 171, 0.5363333904215749, 0.05522870898473681, -0.13487563949833123, 0.04760316688176396, 0.5040343959435514, 0.07314658556239256, 1, 1], [125, 152, 76, 110, 131, 129, 0.45784197506534824, -0.24957010215164926, 0.17214589115901327, 0.03657396909185952, 0.6871624923856804, 0.19246495733941688, 0, 1], [132, 174, 109, 110, 147, 90, 0.9792095347888339, -0.5106181882516534, 0.4833115378189281, 0.04813420645316375, 0.7845727999536309, 0.3504223124123475, 0, 1]]

This table shows two full sequences and their class labels

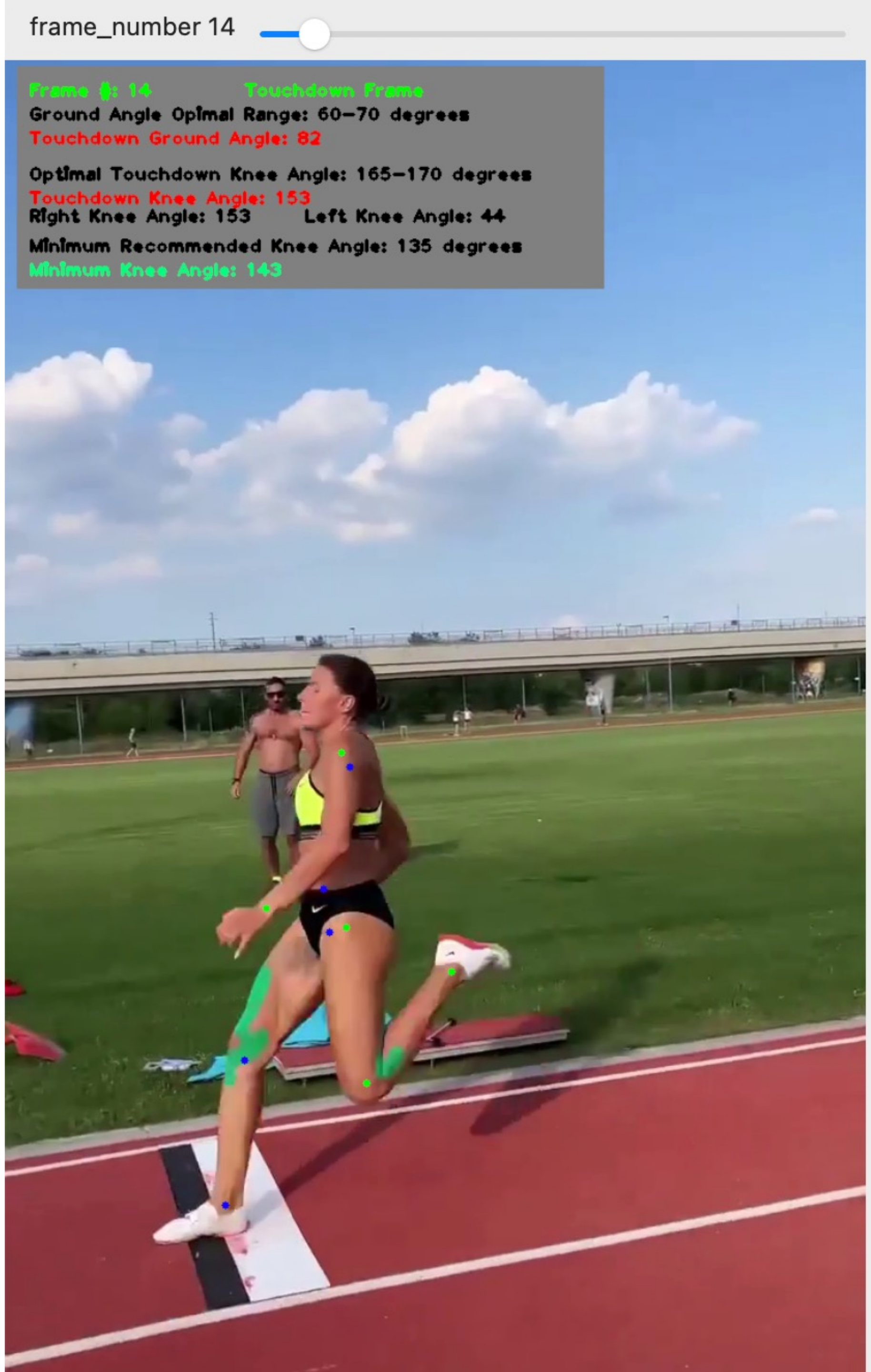
Takeoff Sequence Recognition Model

- Max sequence length was set to 5 frames
    - All sequences were either padded or cut to length of 5
- Using a 70/30 train/test split, achieved 84%-95% accuracy on the test data with different random seeds
- The application model was trained on all 61 datapoints

- Uses a Keras Sequential model with 3 layers.
- Layer 1: LSTM with 64 nodes
- Layer 2: LSTM with 16 nodes
- Layer 3: Dense layer. Softmax function that helps assign the class
- Utilizes “Adam” optimizer

Application Build

- Allows user to cycle through frames
- Displays observed touchdown ground angle, touchdown knee angle, and takeoff sequence minimum knee angle
    - Color coded based on whether they are in range of optimal degree values.
- Displays indicator for whether user is on a touchdown frame
- Displays the knee angles of the current frame



Application Demo

Significance

- There are no current applications like this in track and field
- First system to offer real-time quantitative feedback
- Can inform both technical training and strength training through tracking feature values over time
- Bridge the gap in technology access between amateur and professional coaches
- Provides a technological stepping-stone for use in other T&F events
    - Sprinting
    - Pole Vaulting

Future Work

- Implement pose estimation software that models joint positions in 3D space
  - Eliminates need for cameraman to be in-line with takeoff
  - More accurate angle measurements
  - Would allow for training each model on more data
- Experiment with different features/feature extractors
  - Try deep learning
  - Feature selection techniques
  - Look into existing pose estimation feature extractors
- Measure additional data points
  - Takeoff efficiency: Measure speed at touchdown and at takeoff just from video
  - Angle velocities

References

1. Lees, A., Fowler, N., & Derby, D. (1993). A biomechanical analysis of the last stride, touch-down and take-off characteristics of the women's long jump. *Journal of sports sciences*, *11*(4), 303–314. https://doi.org/10.1080/02640419308730000
2. Alexander, R.M. (1990). Optimum take-off techniques for high and long jumps. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences, 329 1252*, 3-10 .
3. Bridgett, Lisa & Linthorne, Nicholas. (2006). Changes in long jump take-off technique with increasing run-up speed. Journal of sports sciences. 24. 889-97. 10.1080/02640410500298040.
4. Panoutsakopoulos, Vassilios & Papaiakovou, Georgios & Katsikas, Fotios & Kollias, Iraklis. (2010). 3D Biomechanical Analysis of the Preparation of the Long Jump Take-Off. New studies in athletics. 25. 55-68.
5. Linthorne, Nick. " Optimum Take-Off Angle in the Long Jump." *Long Jump Optimum Take-off Angle*, https://www.brunel.ac.uk/~spstnpl/BiomechanicsAthletics/LJOptimumAngle.htm#:~:text=The%20optimum%20take%2Doff%20angle%20for%20the%20athlete%20is%20obtained,%C2%B0%2C%20but%20about%2022%C2%B0.
6. Google. (2021). Mediapipe: A framework for building multimodal (cross, face, hand, iris) applied machine learning pipelines. https://google.github.io/mediapipe/solutions/pose.html
7. OpenCV. (2021). OpenCV: Open Source Computer Vision Library. https://opencv.org/
8. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830. https://jmlr.csail.mit.edu/papers/volume12/pedregosa11a/pedregosa11a.pdf
9. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., ... & Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16) (pp. 265-283). USENIX Association. https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf
10. Chollet, F., & Keras Team. (n.d.). Video classification with pre-trained models. Keras. Retrieved April 13, 2023, from https://keras.io/examples/vision/video_classification/
11. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Naive Bayes. scikit-learn. Retrieved April 13, 2023, from https://scikit-learn.org/stable/modules/naive_bayes.html

Questions