THESIS

APE-V: ATHLETE PERFORMANCE EVALUATION USING VIDEO

Submitted by

Chaitanya Roygaga

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall 2021

Master's Committee:

Advisor: Nathaniel Blanchard

Ross Beveridge
Raoul Reiser

ABSTRACT

APE-V: ATHLETE PERFORMANCE EVALUATION USING VIDEO

Athletes typically undergo regular evaluations by trainers and coaches to assess performance and injury risk. One of the most popular movements to examine is the vertical jump — a sport-independent means of assessing both lower extremity risk and power. Specifically, maximal effort countermovement and drop jumps performed on bilateral force plates provide a wealth of metrics; however, detailed evaluation of this movement requires specialized equipment (force plates) and trained experts to interpret results, limiting its use. Computer vision techniques applied to videos of such movements are a less expensive alternative for extracting such metrics. Blanchard *et al.* [1] collected a dataset of 89 athletes performing these movements and showcased how OpenPose could be applied to the data. However, athlete error calls into question 46.2% of movements — in these cases, an expert assessor would have the athlete redo the movement to eliminate the error. Here, I augmented [1] with expert labels of error and established benchmark performance on automatic error identification. In total, 14 different types of errors were identified by trained annotators. My benchmark models identified errors with an F1 score of 0.710 and a Kappa of 0.457 (Kappa measures accuracy over chance).

ACKNOWLEDGEMENTS

I am grateful to the contributions of Prof. Raoul Reiser in providing the necessary expert guidance required for identifying the jump motion errors in the videos, and the eventual development of rules to define the same. I am also grateful for the work of Michael Boyle during the error annotations and posing as the second annotator for the subset of videos, to get the required Kappa confidence scores for the annotations. I am grateful to Dhruva Patil for putting in the time for actively guiding me during the research process. Lastly, I would like to thank Prof. Aparna Bharati and my advisor Prof. Nathaniel Blanchard for making sure I performed the research tasks thoroughly and correctly, and provided a complete analysis of the works that were performed.

DEDICATION

I would like to dedicate this thesis to my parents Rohit and Pradnya.

TABLE OF CONTENTS

ABSTRACT ii
ACKNOWLEDGEMENTS iii
DEDICATION iv
LIST OF TABLES vi
LIST OF FIGURES vii

Chapter 1 Introduction 1

Chapter 2 Related Work 4

Chapter 3 Dataset 8
3.1 Errors in jump motion 11

Chapter 4 Experiments & Method 18
4.1 Training procedure 19
4.2 Network Architecture and Multi-view Fusion 20

Chapter 5 Results and Analysis: Train models on OpenPose skeleton outputs 22
5.1 OpenPose experiments with Center View Videos 22
5.2 OpenPose experiments with Left View Videos 23
5.3 OpenPose experiments with Right View Videos 24
5.4 OpenPose experiments with Combined View Videos 26
5.5 OpenPose: Comparison between best models trained on data from Confident Frames and Confident Keyframes 28
5.6 Validating usage of Confident Frames from OpenPose Skeleton Outputs . . 30

Chapter 6 Conclusion, Limitations and Future Work 31

Bibliography 33

LIST OF TABLES

3.1 Dataset sub-categories used for APE-V experiments. 12
3.2 Video Dataset summary. 12
3.3 Errors in jump motion. 14 annotated errors (sub-categories) in the evaluative jumps, along with the number of samples in the dataset. 10 errors are annotated for both jump types, while 4 errors are specific to the drop jump. Errors in bold are the 6 primary errors used to train the classification models. 16

4.1 Hyperparameter Search Space for LSTM network architectures. 20

5.1 Best LSTM model architecture for "OpenPose Center: Confident Frames". 23
5.2 Best LSTM model architecture for "OpenPose Center: Confident Keyframes". 24
5.3 Best LSTM model architecture for "OpenPose Left: Confident Frames". 25
5.4 Best LSTM model architecture for "OpenPose Left: Confident Keyframes". 25
5.5 Best LSTM model architecture for "OpenPose Right: Confident Frames". 26
5.6 Best LSTM model architecture for "OpenPose Right: Confident Keyframes". 27
5.7 Best LSTM model architecture for "OpenPose Combined: Confident Frames". 27
5.8 Best LSTM model architecture for "OpenPose Combined: Confident Keyframes". . . . 28
5.9 Comparison between best models trained on data from Confident Frames (CF) — which are selected based on confidence threshold of 0.3 — and Confident Keyframes (CKF). 29
5.10 I verify the use of Threshold 0.3 across all experiments for extracting the OpenPose skeleton outputs. Comparison is made based on the Cohen Kappa score. The values in bold signify the best results in that experiment, and the corresponding column gives the threshold used for pose data. [CF: Confident Frames, CKF: Confident Keyframes] . 30

LIST OF FIGURES

1.1 Athletes can be evaluated for performance and injury risk via countermovement and drop jumps. However, athlete error makes some jumps non-interpretable. In the pursuit of a system to automatically provide feedback to athletes, I annotate a dataset of correct and incorrect jumps and train machine learning models to automatically identify when jumps have been performed incorrectly. In the 'Incorrect' example, the participant lands on one foot, rather than both; a proper landing is exemplified with the 'Correct' example. 2

3.1 Countermovement Jump – 'original jump' and 'jump with pose detection'. 8
3.2 Drop Jump – 'original jump' and 'jump with pose detection'. 9
3.3 The dataset contains videos from multiple views. Three viewpoints of a jump help in providing models with more information on the movement of the participant. 10
3.4 Important Events of an Evaluative Jump. The 'Start of Jump' is different for both jump types, while other events of the jump are similar. 13
3.5 Understanding Cohen's Kappa. It is used as a measure for the agreement between two raters, and is useful in this case for evaluating if the rules defined for annotating the errors in the jumps can be used universally as a standard without changes in annotation outcomes. 15

4.1 Steps to extract pose data from video. 19
4.2 Steps to extract keyframes from video. 19

Chapter 1

Introduction

Athletes of any caliber, from professionals to first time amateurs, try to limit injury while maximizing training/exercise. At the professional level, this balance is facilitated by trainers and coaches who perform expert-level assessments of athletes [2, 3] and customize training plans [4, 5]. Amateur athletes do not have access to the same equipment [1, 6] or personnel employed by professional athletes [7]. However, RGB cameras are ubiquitous — a computer vision system that performs personalized assessments using only RGB cameras would scale to athletes of all means. As an initial step toward such a system, I provide expert annotations to assist athlete performance evaluation and establish a baseline system which automatically assesses if athletes perform movements correctly.

RGB video is a promising modality for athlete evaluation — videos of movements are commonly used by clinicians to assess injury risk [8] and to perform evaluations [9–13]. Estimating pose and joints from video is now ubiquitous in computer vision [9–12], and such information is even extractable from in-motion athletes (for example, joints can be estimated while an athlete swims [10, 11]).

Blanchard *et al*. [1] published a video dataset of 89 athletes performing countermovement and drop jumps. Such movements are regularly used by professional trainers to evaluate athlete performance. However, performance evaluation was not the original intent of the dataset — Blanchard *et al*. [1] were explicitly interested in assessing jumps for ACL injury risk. Since performance evaluation was not the intent, 46.2% of the jumps featured errors that make them unsuitable for evaluating performance [14].

I present APE-V — Athlete Performance Evaluation using Video — a performance-centric augmentation of [1] containing fine-grained annotations of errors found in the dataset. During evaluations [15, 16], athletes are instructed to do some athletic motion [17], and an equipment or a set of equipments then capture variables from the action performed [18, 19] – these variables are in-

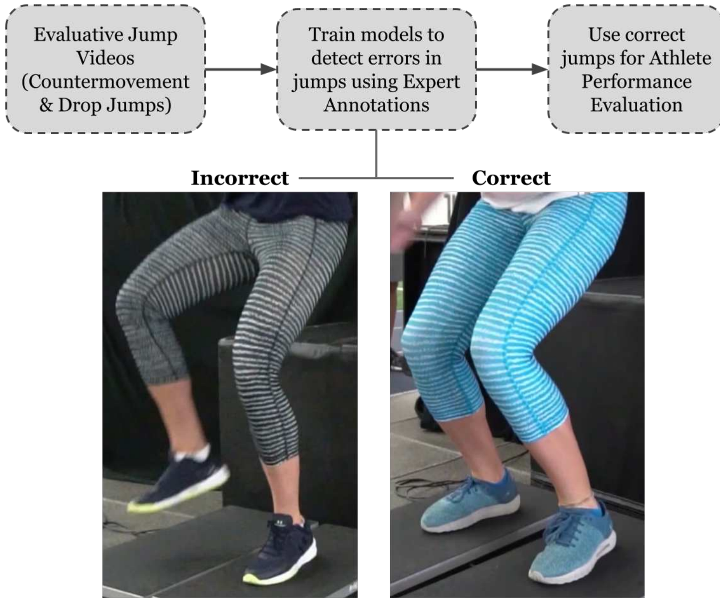

Figure 1.1: Athletes can be evaluated for performance and injury risk via countermovement and drop jumps. However, athlete error makes some jumps non-interpretable. In the pursuit of a system to automatically provide feedback to athletes, I annotate a dataset of correct and incorrect jumps and train machine learning models to automatically identify when jumps have been performed incorrectly. In the 'Incorrect' example, the participant lands on one foot, rather than both; a proper landing is exemplified with the 'Correct' example.

terpreted by experts to determine the performance progress of the athletes. The collected variables depend on the athletes correctly performing their evaluative actions [20] — errors would invalidate evaluative measurements [14]. For the dataset, these actions include the countermovement and the drop jumps. The APE-V pipeline is designed to detect the improper techniques used by athletes while performing the jumps, so that only properly performed jumps are included in an athlete's assessment. Specifically, I provide expert annotations of 14 error types for the countermovement and drop jump videos.

The main goal of the dataset and its annotations is to assist in checking the progress of an athlete by analyzing the body's physical state, and to decide the possible changes in that athlete's training schedule based on their body evaluation results. This can be achieved by training appropriate machine learning models with the evaluative jump videos in the dataset and their corresponding curated error annotations (See Chapter 3). I conduct hyperparameter optimization for architectures and report results for the best models. Note, all evaluations are done using cross-validation, with no overlap between athletes in the training and test set, thereby establishing a person-independent assessment pipeline.

This is the first work to provide expert annotations of 14 kinds of athlete errors found in countermovement and drop jumps. I establish benchmarks showcasing the feasibility of automatic, person-independent assessment of athlete movements.

Chapter 2

Related Work

Athlete performance evaluations are conducted by trainers and coaches. However, traditionally, these evaluations require specialized equipment (e.g., force plates, leap measurement software, and agility ladders [21–23]). Force plates are one of the most common methods for such evaluations, as they can measure power and force outputs during the evaluative jumps such as the countermovement jump [24–27]. Studies have also been conducted using vertical jumps, to assess the Change of Direction Speed (CODS) along with interlimb asymmetries [28–30].

The knee valgus angle has been studied during the drop jump evaluations [31–35] – knee valgus creates extra pressure on the lateral knee compartment, which might cause tears in the cartilage or lateral meniscus [36]. It creates a greater risk in athletes for suffering the anterior cruciate ligament (ACL) injury [37, 38]. Limb asymmetries tend to persist even after recovery from the ACL injury [39], and appropriate measures need to be taken during training to protect the athletes from re-injury.

Various screening methods have been used to analyze these jumps, like the Landing Error Scoring System (LESS) [40] — which looks at the mechanics of the bilateral jump landing [8, 40–42], and the Functional Movement Screen (FMS) — which evaluates fundamental movement patterns in individuals with no pain complaints or musculoskeletal injury [43]. LESS was developed to identify athletes with higher risk of injury, which can be utilized during a team screening session to save time and resources, and is shown to be accurate and reliable. Videos [8] for multiple angles of athlete jumps were recorded, and LESS scores were assigned in conjunction with the erroneous movement patterns across the multiple planes of motion. LESS evaluations showed that ACL injury risk could be visually identified from movement alone. This gives us confidence that injury risk and performance assessments can be done using only a visual medium.

Recent automated techniques used to analyze athlete motion include Einfalt *et al.*'s [9] technique of using 2D human pose sequences as a representation of the actual motion. They demonstrated two approaches for event detection in pose sequences — using multiple fixed cameras and domain information of sport, or using sequence recordings of athlete's motion from a single camera. Yagi *et al*. [44] provided a method to estimate movement of runners from video – by generating a panoramic image of the 100-m track using image stitching, and estimating runner positions in that image using leg joint information – to get additional information like per stride length and speed transition of the runner. Sha *et al*. [45] proposed a method to locate a swimmer in video throughout the frame sequence, and also to detect the stroke rate of that swimmer in motion. This was done by knowing the root position of the swimmer – using pool boundaries and lane ropes – and combining individual body parts detections, like elbows, hands, and head, so that noisy underwater detections were avoided. The stroke cycle could similarly be approximated by using the trajectory of elbows. Li *et al*. [10] presented their work in automatically detecting and examining player actions in sports video sequences for indexing these videos according to actions, and getting kinematic measurements of these actions for improving athlete performance. Colyer *et al*. [46] have discussed the different methods involving extraction of kinematic information from images. The limitations of current systems are highlighted – like requiring markers, controlled conditions and longer processing times – and state-of-the-art automatic markerless systems from computer vision and machine learning are introduced.

El-Sallam *et al*. [13] provided a markerless system for athlete performance optimization in the sports of pole vault, javelin throw, and jumping. They used multiple calibrated cameras for multiple view captures. This method segmented the subject's body from video, and a 3D representation of the body was then reconstructed using silhouettes, which was tracked over the frames in video. This method is extensible to other sports which do not explicitly need body joint detection, but can benefit from detection of the athlete body as a whole. Another markerless method of human motion tracking was developed by Saini *et al*. [47]. Their method was based on "Hierarchical Multi-Swarm Cooperative Particle Swarm Optimization (H-MCPSO)" – the primary purpose was to detect the pose and position of the subject from video, by comparing a rendered body model with the image in a video frame. Particle Swarm Optimization (PSO) continuously tried to improve a solution with respect to a quality measure, but would undergo premature convergence to a local optima when applied to problems like pose tracking, due to the high dimensionality of the search space. H-MCPSO solved this issue by dividing the original population into a master swarm and sub-swarms – to maintain population diversity – and the master swarm evolved independently using viable candidates supplied by itself and the sub-swarms.

Elhayek *et al*. [48] suggested a method for capturing multiple 3D human skeletal movements, even with cluttered and moving backgrounds in videos captured from regular-use camera setups like mobile phones. This method required fewer and possibly asynchronous cameras. For single camera scenarios, Mehta *et al*. [49] combined a CNN-based pose regressor and kinematic skeleton fitting to propose a real-time 3D skeletal pose estimation method. This method created a 3D representation of the real-time motion of the subject in a video by reconstructing a 3D skeleton based on joint predictions. In addition to 3D skeletal information, Cao *et al*. [50, 51] developed an efficient tool to detect 2D poses of multiple subjects in an image. They used Part Affinity Fields (PAFs) to establish pairwise relationships between body parts using their location and orientation. The technique of Elhayek *et al*. [52] combined a stable skeleton motion capture method and 2D joint detection using ConvNet for a kinematic skeleton model. Lienhart *et al*. [11] designed a method to efficiently mine and get explanatory and performance relevant information from noisy pose data of individual sport video recordings, for performance evaluation of athletes. Einfalt *et al*. [12] have discussed automatic event detection during athlete motion for athlete performance analysis, using stride, jump and landing related events from athlete recordings of long and triple jump. They used 2D poses of athletes to get abstract information from videos, for inferring important events in athlete motion.

For identifying lower-body injury risk in athletes, Blanchard *et al*. [1] released a multi-angle video dataset of female athletes performing two specific athletic movements: countermovement and drop jumps. These evaluative jumps are used for research in sports medicine for identifying athleticism and factors in an athlete's jump motion indicating ACL injury risk. The dataset is targeted towards Computer Vision researchers, who could build accurate models and track key movements in these jumps, for evaluating injury risk in the participants. One of the main features of this dataset is that the collection mechanism can be easily replicated, as it is inexpensive when compared with the high-end approaches that require non-portable setups.

My work specifically looks at RGB video recordings of athletes performing two evaluative jumps: the countermovement and the drop jumps. The dataset consists of videos in which participants perform these evaluative jumps [1]. Unlike previous work, I observed that athletes sometimes err when performing these movements; I provide expert-level annotation of such errors (See Chapter 3) and train computer vision models to identify them. Long term, such models will be essential for ensuring that performance or risk assessments are accurate, and no measures are estimated from faulty data.

Chapter 3

Dataset

Here, I augment the dataset originally collected by Blanchard *et al*. [1] with novel labels to adapt it for athlete evaluation. The original dataset can be found in [1], although notably it was expanded post-publication (from 55 athletes to 89). The dataset includes 89 participants performing evaluative jumps regularly used by professional trainers to detect risks of injury in athletes. The two types of jumps used are countermovement and drop jumps, as described below and shown in Figures 3.1 and 3.2.

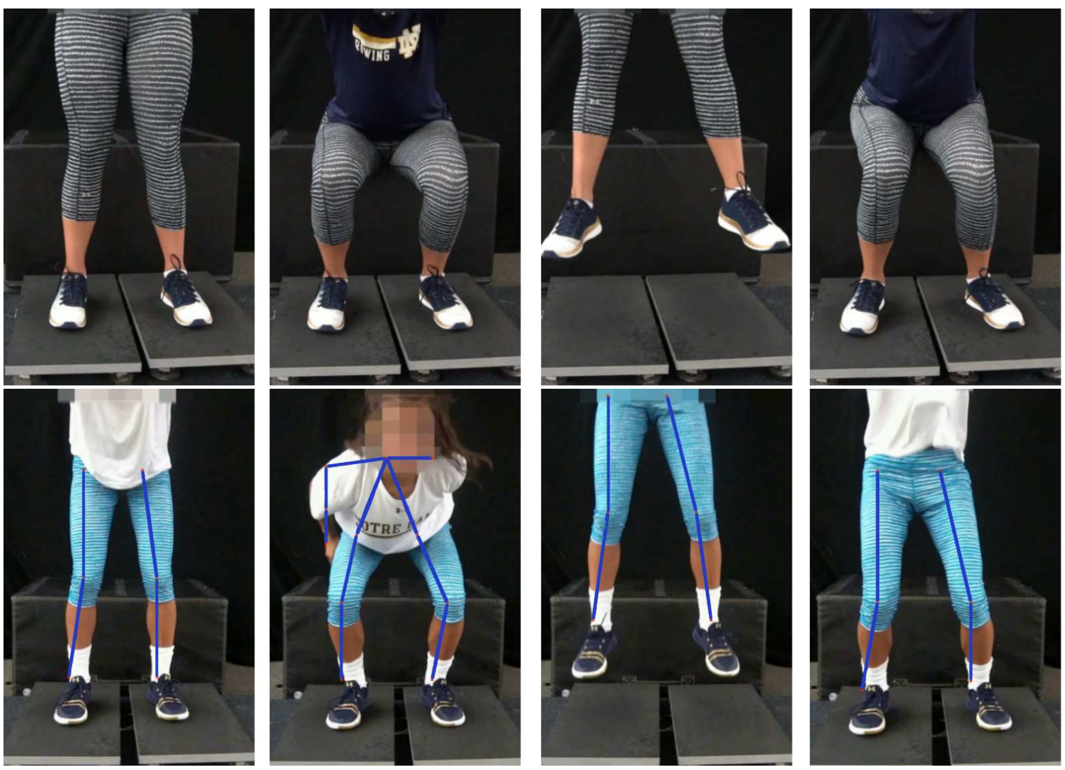

Figure 3.1: Countermovement Jump – 'original jump' and 'jump with pose detection'.

Countermovement Jump - The countermovement jump [53] is a simple and reliable measure of lower-body power. The jump helps coaches in determining performance changes [54] and fatigue levels. Performances in the countermovement jump are linked with maximal speed, maximal strength, and explosive strength. The countermovement jump consists of three phases [53], which are performed without pausing. It begins with the participant standing straight on the force plate, with eyes in the front. The participant then undergoes an unweighting phase, after which there is a braking phase. At this point, the participant's knee angle is at about a 90 degrees, which is subject to each individual. A successive propulsive phase puts the participant in the flight phase, while fully extending their legs and using the momentum to jump higher. The athlete then lands as close to the jumping off point as possible, thus completing one complete motion of the jump.

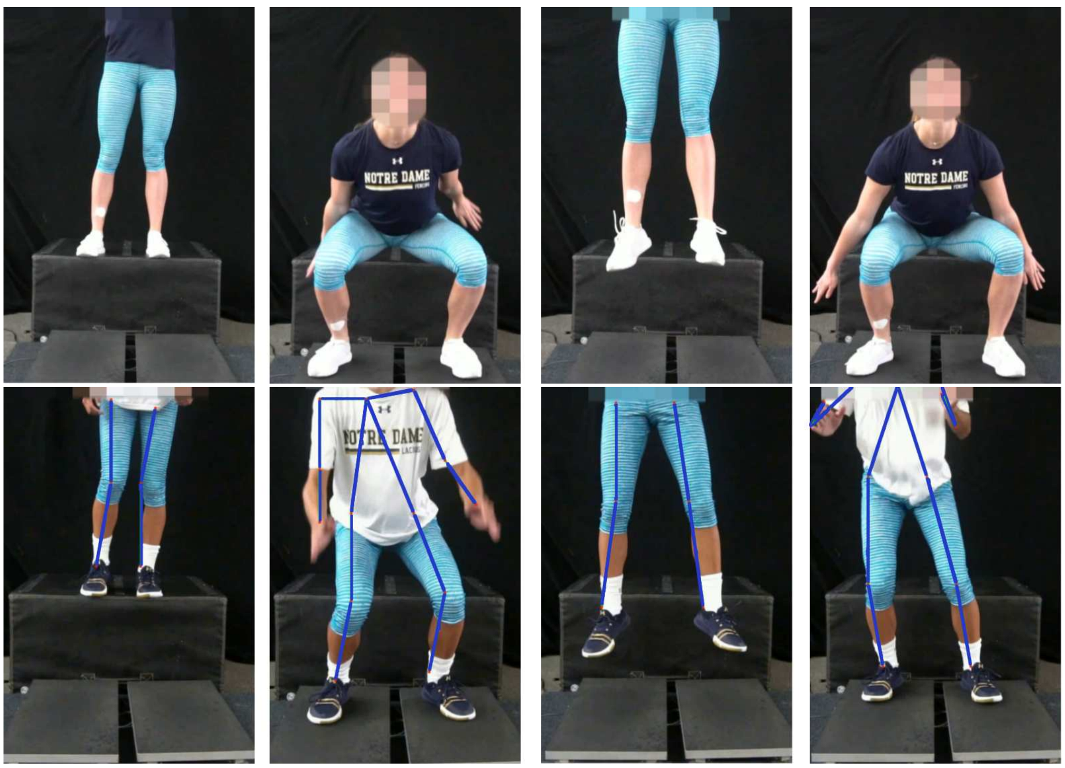

Figure 3.2: Drop Jump – 'original jump' and 'jump with pose detection'.

Drop Jump - The drop jump [55] is designed to examine athlete reactivity. It is considered a fast stretch-shortening movement [56]. One of the main measures of the test is how quickly the athlete can move from absorption to propulsion. It also provides a qualitative indication of an athlete's lower limb alignment [57] in the frontal plane, with a straightforward drop-jump and subsequent and immediate vertical jump. The drop jump has phases similar to the countermovement jump. It begins from an elevated platform. The first phase is the drop phase, in which the participant jumps onto the force plate. As the athlete drops to the ground, they should land with their knees slightly bent. This is the braking phase, also called as the deceleration phase. The propulsive, flight and landing phases are similar to the countermovement jump.

The participants were provided with basic instructions for performing the jumps by showing them examples for the same. All jumps performed by these athletes have been included in the dataset. The first 47 participants performed 10 jumps each on an average — 5 each for the countermovement and drop jumps. The participants after that performed only one of these jumps, with a jump count of 3 on an average. This dataset includes videos from three different angles — center, left and right, as shown in Figure 3.3 — providing the possibility of training models on multiple sources of information.

Figure 3.3: The dataset contains videos from multiple views. Three viewpoints of a jump help in providing models with more information on the movement of the participant.

The videos have been manually annotated with 14 errors found in these evaluative jumps, and I will go through those errors later in Chapter 3.1. Two subsets have been created from the original dataset for training the baseline models. The first subset includes body skeletons of athletes generated using OpenPose [51], with video frames in which the hips, knees and ankles are more confidently detected – above a confidence threshold of 0.3. A confidence value is provided as output by the OpenPose model for every joint detected in a body. A higher confidence value for a detected joint could be attributed to OpenPose being able to more accurately detect that joint. Hence, using a threshold to select the frames from a video would give as output only those frames in which the joints are detected well, providing a better training data for the models. After training models on the data extracted with the confidence threshold of 0.3, I re-trained architectures of the best models of each experiment with data extracted using confidence thresholds of 0.1, 0.2, 0.4 and 0.5, and the results are presented in Chapter 5. This was done to validate if data extracted using threshold 0.3 gave better performing models than those trained with the other thresholds.

LSTM models are trained using x and y coordinates of the detected hips, knees and ankles in these videos. Training on this subset would provide insight into using specific but important points in video frames – whether these are enough information to train a model to confidently detect errors in the jump motion. The second subset is generated on similar lines, but includes only keyframes from the videos present in the first subset. This was done to see if models would perform better even if lesser but targeted temporal information was provided. The dataset categories are illustrated in Table 3.1. There are a total of 582 jumps in the dataset - 346 samples of the countermovement jump, and 236 samples of the drop jump. The dataset summary is as shown in Table 3.2. There are four important events that take place in every jump, as shown in Figure 3.4, three videos being associated with every jump – one each for the center, left and right views.

3.1 Errors in jump motion

I found 14 types of errors in the videos — errors are broken down in Table 3.3. Note that each jump may have multiple errors. For selecting these specific errors, I first went through the videos and found anomalies in them. Then, I went over the jump evaluation literature and had a discussion with an expert from the Health and Exercise Department to focus only on the errors which could directly be identified using the visual 'change of movement'. These errors are spread across the jump. A correct evaluative jump performed by a participant has a few characteristics;

Table 3.1: Dataset sub-categories used for APE-V experiments.

Dataset category	Features	Training
OpenPose [51] Skeleton Outputs: confidently detected frames	Skeleton outputs generated using OpenPose [51]. Frames stored in which hips, knees and ankles detected with confidence above 0.3.	Train LSTM models using x and y coordinates of the detected hips, knees and ankles, across video temporal information.
OpenPose [51] Skeleton Outputs: keyframes	Skeleton outputs generated using OpenPose [51]. Frames stored in which hips, knees and ankles detected with confidence above 0.3, which are then filtered to keep only key-frames.	

Table 3.2: Video Dataset summary.

Total No. of Jumps	**582**
No. of Countermovement Jumps	346
No. of Drop Jumps	236
Total No. of Participants	**89**
No. of Participants Performing Countermovement and Drop Jumps	47
No. of Participants Performing only Countermovement Jumps	41
No. of Participants Performing only Drop Jumps	1
No. of Camera Views for each Video	**3**
Total No. of Videos	1746
Camera settings: No fixed camera angle and height.	

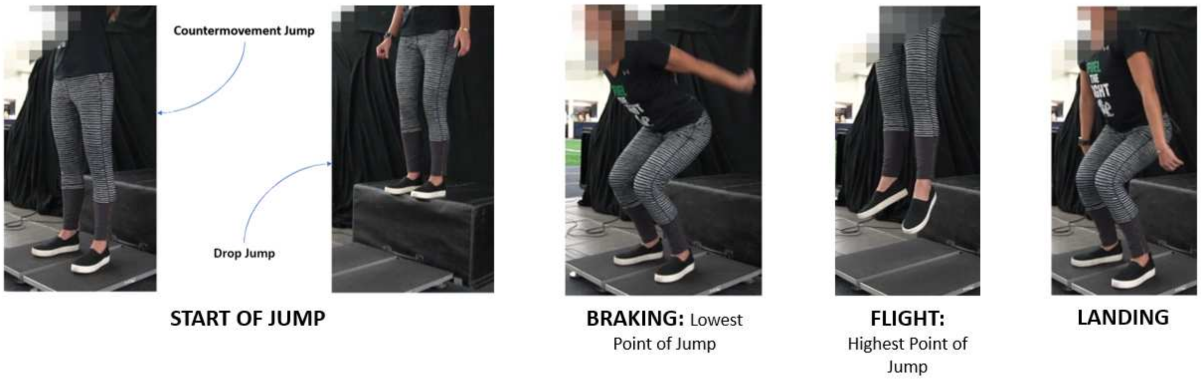

Figure 3.4: Important Events of an Evaluative Jump. The 'Start of Jump' is different for both jump types, while other events of the jump are similar.

I used these characteristics to identify pivotal errors in jumps. First, the participant assumes a straight posture while looking forward [58], jumping high enough after an initial squat, and then landing back in a similar position from which they started [59]. During these jump motions the participant might not start with the correct position, or, they might perform an irregular landing. Additionally, for the drop jump, the participant drops from a box onto the force plate [55]. A well executed drop culminates with both feet touching simultaneously [60], followed by a quick reflex jump. Associated errors include jumping instead of dropping or lingering on the force plate for too long, rather than immediately jumping.

The most impactful errors are emphasized in bold in Table 3.3. The other eight errors tend to be more subtle deviations from the correct body movement, which provide supporting information regarding motion flaws in the jumps. In the long term, these annotations are essential for proper assessment, but for now, I do not consider jumps with only subtle deviations to be erroneous.

The following rules have been used to annotate the errors:

- Category: Initial incorrect position common to both jumps [58, 61].
  - **Feet less than shoulder width apart**: If feet of participant are closer to each other than half of the hip-width of the participant.
- Category: Initial incorrect position, during drop jump [60, 62].

- **Jumped upward from box, rather than forward**: If jump is visually more upwards.
- Asymmetric landing after jump: If one foot hits the force plate before the other.

- Category: Intermediate incorrect motion, during drop jump [63].
  - **Squat too low**: If participant squats below their knee.
  - Heels touch force plate: If either heel touches the force plate for more than a couple seconds.
- Category: First or final landing on force plate, common to both jumps.
  - Knee collapse: If knees move visually more inwards when participant lands on force plate.
  - Both feet not on respective platforms: If either foot is not on respective force plate on landing.
  - Land off-balance: If participant lands off-balance or partially on the feet.
- Category: During jump, common to both jumps [59].
  - Off-balance: If participant has an excess sideways body motion during jump.
  - Body twists, landing at different angle: If participant rotates about a vertical axis during jump and lands at different angle.
- Category: Final landing error, common to both jumps [64].
  - **Landing at different position from initial landing**: If either feet land a slight distance away from their respective starting position.
  - **Excessive hip and knee flexion before returning to upright standing position**: If squat is lower than knee position.
  - **Take additional steps to maintain balance**: If balance lost on landing and participant takes additional steps to maintain balance.

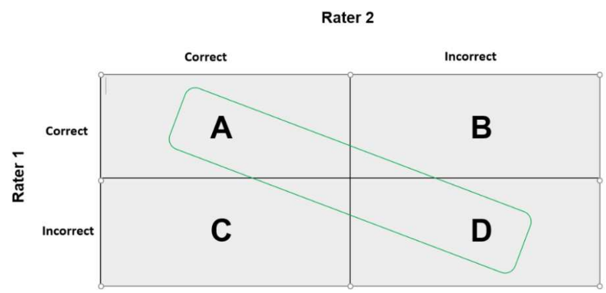

Figure 3.5: Understanding Cohen's Kappa. It is used as a measure for the agreement between two raters, and is useful in this case for evaluating if the rules defined for annotating the errors in the jumps can be used universally as a standard without changes in annotation outcomes.

– Feet less than shoulder width apart: Similar to starting position.

The labels for the videos are modified to train binary classifiers — videos which have any of the 6 primary errors [Errors in bold, Table 3.3] are labeled with a '1', and the rest are labeled with a '0'.

Inter-observer agreement - To be certain that the annotations were correctly labeled after following the defined set of rules, I chose to use the Cohen's Kappa scores for checking the inter-observer agreement. Cohen's kappa [65–67] is used as a measure for the agreement between two raters, and is useful in this case for evaluating if the rules defined for annotating the errors in the jumps can be used universally as a standard without changes in annotation outcomes; to ensure there was agreement within and across error types. One observer, other than myself, annotated an example set of 100 videos (17% of dataset) using the above set of rules. These annotations were then scored based on Figure 3.5, considering the chance agreement. Equations 3.1 refer to Cohen's Kappa calculations. Chance agreement accounts for the fact that raters might sometimes guess on some variables based on uncertainty – being unsure. Any score above 0.6 shows substantial agreement between the raters, although a score of 1 is always preferred. Across all errors, the average Kappa was 0.89 (Max Kappa is 1.00), indicating very high agreement.

Table 3.3: Errors in jump motion. 14 annotated errors (sub-categories) in the evaluative jumps, along with the number of samples in the dataset. 10 errors are annotated for both jump types, while 4 errors are specific to the drop jump. Errors in bold are the 6 primary errors used to train the classification models.

Errors (Overall categories)	Jump Type	Sr. No.	Errors (Sub-categories)	Inter-observer agreement (Subset of 100 video samples)		No. of samples in dataset
				Kappa (κ)	Samples in subset	
Start position	Both	1	Feet less than shoulder width apart	1.00	7	30
Initial position, after start	Drop jump	2	Jumped upward from box, rather than forward	1.00	2	22
		3	Asymmetric landing after jump	1.00	3	15
First landing on force plate	Drop jump	4	Squat too low	1.00	14	37
		5	Heels touch force plate	0.90	14	83
First or Final landing on force plate	Both	6	Knee collapse	0.72	19	64
		7	Both feet not on respective platforms	1.00	1	5
		8	Land off-balance	0.91	12	49
During jump	Both	9	Off-balance	0.75	8	79
		10	Body twists, landing at different angle	0.75	16	74
Final landing on force plate	Both	11	Landed at different position from initial landing	0.81	42	133
		12	Excessive hip and knee flexion before returning to upright standing position	0.88	16	78
		13	Take additional steps to maintain balance	0.73	7	94
		14	Feet less than shoulder width apart	1.00	1	3

$$\begin{cases} P_0 = Number\,in\,Agreement\,/\,Total \\ P_{correct} = (A+B/A+B+C+D)*(A+C/A+B+C+D) \\ P_{incorrect} = (C+D/A+B+C+D)*(B+D/A+B+C+D) \\ P_e = P_{correct} + P_{incorrect} \\ k(Kappa) = P_0 - P_e\,/\,1 - P_e \end{cases} \tag{3.1}$$

Chapter 4

Experiments & Method

I conducted baseline experiments to investigate the usability of the dataset and its corresponding annotations. I focused on two major questions: is video information enough to facilitate detection of errors during motion of evaluative jumps? And, if video information is good enough, what kinds of features (raw video or pose) provide optimal performance?

Two categories of subsets for the data have been processed, as shown in Table 3.1. A set of experiments is performed on the participant lower-body joint data extracted from the videos using OpenPose [51]. The joint data is extracted using the following steps:

1. Video is read using the OpenCV [68] function 'VideoCapture'.

2. Output video file is created, using required 'fps' and dimensions of frames in the video.

3. Pose is detected in a frame using the OpenPose [51] trained model 'pose_iter_440000.caffemodel'. A threshold of 0.3 is used for confidence in the pose detection (See Chapter 3). This threshold ensures that most of the less confident pose estimations are eliminated, to provide a more stable temporal data on pose estimation for the participant jumps.

4. Six joints are considered when looking at the 0.3 threshold: Right Hip, Right Knee, Right Ankle, Left Hip, Left Knee, and Left Ankle. If atleast 3 of these joints are detected with a 0.3 confidence or more by OpenPose [51], then the frame is kept, else it is discarded.

5. This process repeats for all frames in a video, and the kept frames are stored in the output video file. The steps are summarized in Figure 4.1.

6. An extra step is performed for the second set of data, in which keyframes are selected as a subset from the confidently detected frames.

   (a) For getting keyframes, frames are first converted to blurred gray frames.

(b) If the frame is the first of the video, it is saved in a variable.

(c) Difference between the previous and the current blurred gray frames is calculated.

(d) Number of non-zero pixels in the image difference are calculated.

(e) Using these non-zero pixels, the baseline of the data is defined.

(f) Indexes of the peaks are then extracted: These are indexes of frames with the non-zero pixel count above a specified threshold.

(g) These indexes are then used to identify and store the keyframes. The steps are summarized in Figure 4.2.

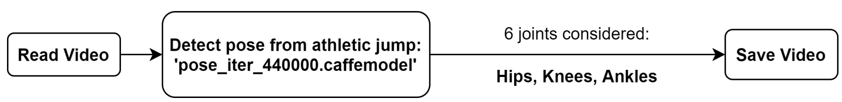

Figure 4.1: Steps to extract pose data from video.

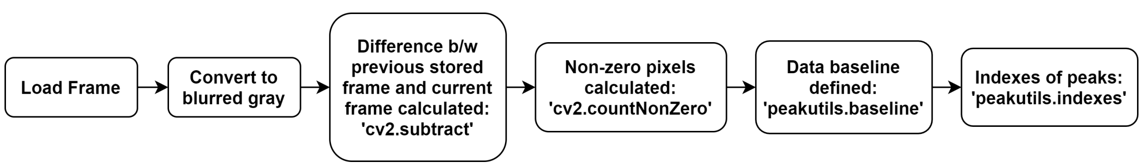

Figure 4.2: Steps to extract keyframes from video.

Multiple experiments on Long Short-Term Memory (LSTM) networks were employed on specific types of input data to obtain view-specific detection results; model training on individual view data – center, left or right.

4.1 Training procedure

A standard training procedure was followed across all experiments. The experiments were designed to run hyperparameter search from the given set of hyperparameters [Table 4.1]. Hy-

Table 4.1: Hyperparameter Search Space for LSTM network architectures.

LEARNING RATE	TRAINABLE LAYERS	HIDDEN NODES	BATCH SIZE	EPOCHS
0.0001 - 0.1	1 - 4	10-200	[8, 16, 24, 32, 40, 48, 56, 64, 128, 256]	10-200

peropt [69] with the Tree Parzen Estimator (TPE) algorithm was used for this purpose. For each hyperparameter combination, the models were trained and evaluated using 5-fold cross validation. K-fold cross validation helps to evaluate a given model on the entire dataset, providing more robust measures of performance for small datasets. As features in the video frames might be similar for a particular participant, the data is distributed into folds based on participants and not jumps. Due to this, there is a difference in the number of jumps in the training and validation set for each of these folds, as all participants have not performed the same number of jumps. Each model trained on (K-1) partitions, during cross validation, is then evaluated for validation loss at every epoch. The model corresponding to the lowest validation loss is saved. Note that not all participants performed the same number of jumps.

Models saved for each of these data folds at the end of the training cycle were then evaluated using Cohen's Kappa score and F1 score (positive error class). These values were used for selecting the best models during hyperparameter search. The data is unevenly distributed — 46% positive class, and 54% negative class. Cohen's Kappa represents how well a model performs when compared to a model that randomly predicts an output (i.e., accuracy above chance). A positive score for Kappa indicates that the model performs better than chance. After individual models are trained for each of the five data folds, metric scores were averaged across folds.

4.2 Network Architecture and Multi-view Fusion

The experiments used Long Short-Term Memory (LSTM) networks, which were trained from scratch on pose estimation joint data. This information was extracted from the evaluative jump videos using OpenPose [51], as discussed in Chapter 3. The LSTM-based architecture was chosen to learn the order dependence between items in a long data sequence, and is suitable for the task of detecting changes in athlete motion through the video frames. The models were trained on data subsets presented in Table 3.2. Each model goes through a 5-fold cross validation. The training data undergoes scaling using the 'MinMaxScaler', and then the training dataloader is defined. The scaler which is fit on the training data is then used to scale the validation data, followed by defining the validation dataloader.

The LSTM model architecture takes in joint coordinates of the six important joints — hips, knees, and ankles — for every selected frame across the video, with a label 1 (Error) or 0 (No error) annotated for every such sequence. Each model trained on individual data folds is then evaluated on Cohen's Kappa score and F1 score (positive error class). After individual models are trained for each of the five data folds, the metric scores are averaged, and these are presented as the final scores.

To understand if a model trained on multiple sources of information for the same task could perform better than the individual models, I perform experiments by combining the best models trained on individual view data. Specifically, the classification layer of the best models trained on each of these views was replaced with a separate classifier layer at the end of the combined architecture. This new layer was trained for a few epochs, and the trained model was then evaluated in the same manner as the individual models described in Chapter 4.1.

Chapter 5

Results and Analysis: Train models on OpenPose skeleton outputs

For the experiments described above using different data modalities and architectures, I obtain evaluation results on the novel annotated dataset for the task of erroneous jump evaluation. I specifically focus on answering questions such as which type of data input and architecture performs better and how does view information affect detection results.

The experiments were designed on the assumption that lower body joint coordinates detected on athletes performing the evaluation jumps were enough information to train a machine learning model to distinguish between erroneous jumps and those useful for athlete evaluation. OpenPose [51] detected joint coordinates for hips, knees, and ankles, for the full length of the videos or selected frames, were then used to train an LSTM model.

Three individual models were trained on lower body joint data from the center, left, and right view videos. There were two types of input sequences for these models. The first category of data were frames for which OpenPose was highly confident in detecting lower body joints, when run on the entire video. The second category had keyframes extracted from the frames where pose was confidently detected. About 100 models were trained during each of these experiments. For the view fusion model, I combined the individual models trained on the different views for each type of data input.

5.1 OpenPose experiments with Center View Videos

This set of experiments is performed with the lower body joint data extracted from the center view videos.

OpenPose Center: Confident Frames - The architecture of the best model trained on the confident frames has just 1 hidden layer with 50 LSTM units in that layer. It gives a F1 positive class score of 0.6291, and a Cohen Kappa score of 0.3737. Table 5.1 shows the best model obtained from this experiment.

Table 5.1: Best LSTM model architecture for "OpenPose Center: Confident Frames".

No. of Hidden Layers	1
No. of Hidden Units in each layer	50
Batch Size	40
Learning Rate	0.0038
Testing Loss	0.0165
Testing Accuracy	**69.2%**
True Positive	**30**
True Negative	**50**
False Positive	**12**
False Negative	**23**
F1 Score (+ve class: Error in Jump)	**0.6291**
F1 Score (-ve class: No Error in Jump)	**0.7371**
Cohen Kappa score	**0.3737**

OpenPose Center: Confident Keyframes - The architecture of the best model trained on the confident keyframes has 2 hidden layers with 60 LSTM units in each layer. It gives a F1 positive class score of 0.5142, and a Cohen Kappa score of 0.2677. Table 5.2 shows the best model obtained from this experiment.

5.2 OpenPose experiments with Left View Videos

This set of experiments is performed with the lower body joint data extracted from the left view videos.

OpenPose Left: Confident Frames - The architecture of the best model trained on the confident frames has 4 hidden layers with 80 LSTM units in each layer. It gives a F1 positive class score of

Table 5.2: Best LSTM model architecture for "OpenPose Center: Confident Keyframes".

NO. OF HIDDEN LAYERS	2
NO. OF HIDDEN UNITS IN EACH LAYER	60
BATCH SIZE	40
LEARNING RATE	0.0019
TESTING LOSS	0.0178
TESTING ACCURACY	**64.6%**
TRUE POSITIVE	**27**
TRUE NEGATIVE	**49**
FALSE POSITIVE	**13**
FALSE NEGATIVE	**26**
F1 SCORE (+VE CLASS: ERROR IN JUMP))	**0.5142**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP))	**0.6975**
COHEN KAPPA SCORE	**0.2677**

0.5083, and a Cohen Kappa score of 0.2241. Table 5.3 shows the best model obtained from this experiment.

OpenPose Left: Confident Keyframes - The architecture of the best model trained on the confident keyframes has 2 hidden layers with 95 LSTM units in each layer. It gives a F1 positive class score of 0.5090, and a Cohen Kappa score of 0.2497. Table 5.4 shows the best model obtained from this experiment.

5.3 OpenPose experiments with Right View Videos

This set of experiments is performed with the lower body joint data extracted from the right view videos.

OpenPose Right: Confident Frames - The architecture of the best model trained on the confident frames has 2 hidden layers with 165 LSTM units in each layer. It gives a F1 positive class score of 0.6349, and a Cohen Kappa score of 0.3560. Table 5.5 shows the best model obtained from this experiment.

Table 5.3: Best LSTM model architecture for "OpenPose Left: Confident Frames".

NO. OF HIDDEN LAYERS	4
NO. OF HIDDEN UNITS IN EACH LAYER	80
BATCH SIZE	56
LEARNING RATE	0.0099
TESTING LOSS	0.0142
TESTING ACCURACY	**62.2%**
TRUE POSITIVE	**26**
TRUE NEGATIVE	**46**
FALSE POSITIVE	**15**
FALSE NEGATIVE	**27**
F1 SCORE (+VE CLASS: ERROR IN JUMP)	**0.5083**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP)	**0.6818**
COHEN KAPPA SCORE	**0.2241**

Table 5.4: Best LSTM model architecture for "OpenPose Left: Confident Keyframes".

NO. OF HIDDEN LAYERS	2
NO. OF HIDDEN UNITS IN EACH LAYER	95
BATCH SIZE	8
LEARNING RATE	0.0069
TESTING LOSS	0.0842
TESTING ACCURACY	**63.4%**
TRUE POSITIVE	**26**
TRUE NEGATIVE	**47**
FALSE POSITIVE	**15**
FALSE NEGATIVE	**27**
F1 SCORE (+VE CLASS: ERROR IN JUMP))	**0.5090**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP))	**0.6920**
COHEN KAPPA SCORE	**0.2497**

Table 5.5: Best LSTM model architecture for "OpenPose Right: Confident Frames".

NO. OF HIDDEN LAYERS	2
NO. OF HIDDEN UNITS IN EACH LAYER	165
BATCH SIZE	32
LEARNING RATE	0.0081
TESTING LOSS	0.0217
TESTING ACCURACY	**67.8%**
TRUE POSITIVE	**32**
TRUE NEGATIVE	**47**
FALSE POSITIVE	**15**
FALSE NEGATIVE	**21**
F1 SCORE (+VE CLASS: ERROR IN JUMP)	**0.6349**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP)	**0.7163**
COHEN KAPPA SCORE	**0.3560**

OpenPose Right: Confident Keyframes - The architecture of the best model trained on the confident keyframes has only 1 hidden layer with 25 LSTM units in that layer. It gives a F1 positive class score of 0.6253, and a Cohen Kappa score of 0.3372. Table 5.6 shows the best model obtained from this experiment.

5.4 OpenPose experiments with Combined View Videos

This set of experiments is performed with the lower body joint data extracted from all the view videos – center, left and right.

OpenPose Combined: Confident Frames - The best models trained on the confident frames of each of the views – center, left and right – have been used to create an ensemble architecture for this experiment. It gives a F1 positive class score of 0.7104, and a Cohen Kappa score of 0.4567. Table 5.7 shows the best model obtained from this experiment.

OpenPose Combined: Confident Keyframes - The best models trained on the confident keyframes of each of the views – center, left and right – have been used to create an ensemble architecture for

Table 5.6: Best LSTM model architecture for "OpenPose Right: Confident Keyframes".

NO. OF HIDDEN LAYERS	1
NO. OF HIDDEN UNITS IN EACH LAYER	25
BATCH SIZE	24
LEARNING RATE	0.0056
TESTING LOSS	0.0282
TESTING ACCURACY	**67.0%**
TRUE POSITIVE	**32**
TRUE NEGATIVE	**46**
FALSE POSITIVE	**16**
FALSE NEGATIVE	**21**
F1 SCORE (+VE CLASS: ERROR IN JUMP))	**0.6253**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP))	**0.7099**
COHEN KAPPA SCORE	**0.3372**

Table 5.7: Best LSTM model architecture for "OpenPose Combined: Confident Frames".

BATCH SIZE	16
LEARNING RATE	0.0054
TESTING LOSS	0.0446
TESTING ACCURACY	**72.4%**
TRUE POSITIVE	**39**
TRUE NEGATIVE	**45**
FALSE POSITIVE	**17**
FALSE NEGATIVE	**14**
F1 SCORE (+VE CLASS: ERROR IN JUMP)	**0.7104**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP)	**0.7425**
COHEN KAPPA SCORE	**0.4567**

this experiment. It gives a F1 positive class score of 0.6581, and a Cohen Kappa score of 0.4065. Table 5.8 shows the best model obtained from this experiment.

Table 5.8: Best LSTM model architecture for "OpenPose Combined: Confident Keyframes".

BATCH SIZE	16
LEARNING RATE	0.0043
TESTING LOSS	0.0413
TESTING ACCURACY	**70.8%**
TRUE POSITIVE	**33**
TRUE NEGATIVE	**49**
FALSE POSITIVE	**12**
FALSE NEGATIVE	**20**
F1 SCORE (+VE CLASS: ERROR IN JUMP))	**0.6581**
F1 SCORE (-VE CLASS: NO ERROR IN JUMP))	**0.7444**
COHEN KAPPA SCORE	**0.4065**

5.5 OpenPose: Comparison between best models trained on data from Confident Frames and Confident Keyframes

From Table 5.9, it is observed that the models trained on Confident Frames performed better than the models trained on Keyframes extracted from that data, with Kappa scores considered as the baseline. These results show that the frames in which skeleton data was confidently detected provided more meaningful information for improved model performance. I also observe that the ensemble model using all three view data as input gives better performance than the individual models. I get the best overall model when training with the combined view data, each with Confident Frames extracted at threshold 0.3 – Accuracy: 72.4 %, F1 positive class score: 0.7104, Kappa: 0.4567. This could be because the individual best models trained on Confident Frames did better (or only slightly worse) in all the experiments. Also, the best model trained on these individual views came with the right view data of the Confident Frames (with a Kappa of 0.3560).

When combining the best models trained on each of the individual view data, and using this ensemble architecture to predict an output, I feel that the concatenated features given as output from each of these individual models provide a better contextual information to the classifier. This result shows two things:

1. It is possible to train a simple LSTM model with pose estimates detected from the evaluative jumps using any individual view data.

2. If more views of the same jump are available, the individual models trained on these other views could be combined to provide better overall performance.

Table 5.9: Comparison between best models trained on data from Confident Frames (CF) — which are selected based on confidence threshold of 0.3 — and Confident Keyframes (CKF).

Experiment		Test Acc. %	Confusion Matrix				F1 Score: Error in Jump	F1 Score: No Error in Jump	Cohen Kappa
			TP	TN	FP	FN			
CF	Center	69.2 ±8.01	30	50	12	23	**0.629 ±0.10**	0.737 ±0.07	**0.374 ±0.16**
CF	Left	62.2 ±4.60	26	46	15	27	**0.508 ±0.21**	0.682 ±0.04	**0.224 ±0.13**
CF	Right	67.8 ±3.63	32	47	15	21	**0.635 ±0.06**	0.716 ±0.03	**0.356 ±0.08**
CF	Combined View	72.4 ±4.72	39	45	17	14	**0.710 ±0.05**	0.743 ±0.05	**0.457 ±0.09**
CKF	Center	64.6 ±6.80	27	49	13	26	**0.514 ±0.26**	0.698 ±0.10	**0.268 ±0.18**
CKF	Left	63.4 ±5.73	26	47	15	27	**0.509 ±0.24**	0.692 ±0.06	**0.250 ±0.16**
CKF	Right	67.0 ±3.61	32	46	16	21	**0.625 ±0.08**	0.710 ±0.02	**0.337 ±0.09**
CKF	Combined View	70.8 ±6.83	33	49	12	20	**0.658 ±0.10**	0.744 ±0.06	**0.407 ±0.15**

5.6 Validating usage of Confident Frames from OpenPose Skeleton Outputs

I selected frames where the pose detection model had a confidence of c or higher in its predictions, as confident frames for the experiments described in Chapter 3. To obtain the confidence threshold, I performed additional experiments. I trained the same architecture with the subset of data extracted using different thresholds. The architecture and hyperparameters remained constant, and only the data changed. This helped me to evaluate the effect of different thresholds used for obtaining a good set of video pose features from raw video frames.

From the comparison in Table 5.10, it is observed that models performed best with joint data extracted with a confidence threshold of 0.3. This threshold eliminates many noisy frames with fluctuating pose estimations for the lower body joints, while retaining ample information to train good models for any of the three views.

Table 5.10: I verify the use of Threshold 0.3 across all experiments for extracting the OpenPose skeleton outputs. Comparison is made based on the Cohen Kappa score. The values in bold signify the best results in that experiment, and the corresponding column gives the threshold used for pose data. [CF: Confident Frames, CKF: Confident Keyframes]

Experiment		Cohen Kappa				
		Threshold 0.1	**Threshold 0.2**	**Threshold 0.3**	**Threshold 0.4**	**Threshold 0.5**
CF	Center	0.117	0.248	**0.374**	0.285	0.239
	Left	0.153	0.155	**0.224**	0.153	0.175
	Right	0.130	0.314	**0.356**	0.224	0.325
CKF	Center	0.138	0.231	**0.268**	0.191	0.203
	Left	0.246	0.191	**0.250**	0.194	0.142
	Right	0.195	0.234	**0.337**	0.296	0.178

Chapter 6

Conclusion, Limitations and Future Work

In this work, I presented expert-level error annotations for a jump video dataset [1] to facilitate fitness assessment from RGB video. Further, I provided baselines showcasing that, while these annotations include relatively fine-grained phenomena, it is feasible to identify them with computer vision techniques. I present automated approaches to detect and screen out improper techniques present in jumps of three views — center, left, and right — performed for athlete evaluation, so that time and expertise can be allocated for assessing only the correctly performed jumps, and feedback can be provided for improving jump motion of those which are discarded.

Overall, I have shown two things:

1. The curated expert-level annotations of errors in the evaluative jumps performed by athletes, and the rules defined for these annotations, could be considered as a standard for labeling videos collected for athlete performance evaluation.

2. Classification models could be easily trained with a video of an athlete's motion and its corresponding error annotations.

While the set of experiments described here solely rely on pose estimations extracted using OpenPose [51], I plan to perform further experiments on the complete video frames. This would help to discern if more features from a video frame help improve the overall model performance, or if lesser data points from the frames – like in the pose experiments – are the course forward for training models with the presented type of data and annotations. One of the experiments in the pipeline is to use a pretrained CNN model like ResNet18 [70] to extract features from the video frames, and then train an LSTM architecture on the extracted features. Another set of experiments would make use of fine-tuning the original version of the TSM [71] action recognition model, to see if a widely used action recognition model is capable of learning and building on the presented dataset.

Along with the possibilities of easier athlete assessment, there are a few drawbacks to the implementations in the current pipeline. The dataset has 582 jumps from 89 participants, and these jumps only include female athletes – as they are at a higher risk of injury [72–76]. A larger dataset could be collected in its place which would include athletes across genders, ages and races. This could provide the models with more variety of data with respect to the athletes, eventually improving the generalizability of the models, and could help in determining if this pipeline actually helps in identifying and eliminating the bad jumps used for performance evaluations.

Lastly, the data preprocessing step uses OpenPose [51], which provides the skeleton outputs of participants performing the evaluative jumps on which the models are trained. While OpenPose skeleton outputs provide decent approximations of the actual participant joints, there is still a lot of scope for improvement. The models are trained on these skeleton outputs – the more accurate the skeleton outputs, higher the possibility of improved models. Hence, improved skeleton data approximations is the next step for getting better overall model performance. Alternatives like AlphaPose [77] could be used in place of OpenPose.

Accurately evaluating athletes based on movements recorded with ubiquitous RGB cameras has a multitude of implications for fitness recommendations. Ideally, accurate evaluations can enable widespread access to state-of-the-art fitness recommendations. Although this work does not focus on injury prevention, an implicit side effect of appropriate fitness recommendations is the limitation of overextension, which can lead to injury. I anticipate that the resources presented in this work are essential for future investigations for performance assessments with the dataset.

Bibliography

[1] Nathaniel Blanchard, Kyle Skinner, Aden Kemp, Walter Scheirer, and Patrick Flynn. “keep me in, coach!": A computer vision perspective on assessing acl injury risk in female athletes. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1366–1374. IEEE, 2019.

[2] Pitre C Bourdon, Marco Cardinale, Andrew Murray, Paul Gastin, Michael Kellmann, Matthew C Varley, Tim J Gabbett, Aaron J Coutts, Darren J Burgess, Warren Gregson, et al. Monitoring athlete training loads: consensus statement. *International journal of sports physiology and performance*, 12(s2):S2–161, 2017.

[3] Carl Foster, Jose A Rodriguez-Marroyo, and Jos J De Koning. Monitoring training loads: the past, the present, and the future. *International journal of sports physiology and performance*, 12(s2):S2–2, 2017.

[4] Marcel Lopes Dos Santos, Melissa Uftring, Cody A Stahl, Robert G Lockie, Brent Alvar, J Bryan Mann, and J Jay Dawes. Stress in academic and athletic performance in collegiate athletes: A narrative review of sources and monitoring strategies. *Frontiers in Sports and Active Living*, 2:42, 2020.

[5] Sean Scantlebury, Kevin Till, Thomas Sawczuk, Padraic Phibbs, and Ben Jones. Navigating the complex pathway of youth athletic development: Challenges and solutions to managing the training load of youth team sport athletes. *Strength & Conditioning Journal*, 42(6):100–108, 2020.

[6] David H Fukuda. *Assessments for sport and athletic performance*. Human Kinetics, 2018.

[7] Stephanie H Clines, Cailee E Welch Bacon, Christianne M Eason, Kelly D Pagnotta, Robert A Huggins, and Bonnie L Lunen. Athletic directors’ perceptions regarding the value of employ-

ing athletic trainers in the secondary school setting. *Journal of Physical Education and Sports Management*, 2019.

[8] Darin A Padua, Stephen W Marshall, Michelle C Boling, Charles A Thigpen, William E Garrett Jr, and Anthony I Beutler. The landing error scoring system (less) is a valid and reliable clinical assessment tool of jump-landing biomechanics: the jump-acl study. *The American journal of sports medicine*, 37(10):1996–2002, 2009.

[9] Moritz Einfalt and Rainer Lienhart. Decoupling video and human motion: towards practical event detection in athlete recordings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 892–893, 2020.

[10] Haojie Li, Jinhui Tang, Si Wu, Yongdong Zhang, and Shouxun Lin. Automatic detection and analysis of player action in moving background sports video sequences. *IEEE transactions on circuits and systems for video technology*, 20(3):351–364, 2009.

[11] Rainer Lienhart, Moritz Einfalt, and Dan Zecha. Mining automatically estimated poses from video recordings of top athletes. *International Journal of Computer Science in Sport*, 17(2):94–112, 2018.

[12] Moritz Einfalt, Charles Dampeyrou, Dan Zecha, and Rainer Lienhart. Frame-level event detection in athletics videos with pose-based convolutional sequence networks. In *Proceedings Proceedings of the 2nd International Workshop on Multimedia Content Analysis in Sports*, pages 42–50, 2019.

[13] Amar A. El-Sallam, Mohammed Bennamoun, Ferdous Sohel, Jacqueline A. Alderson, Andrew Lyttle, and Marcel Mourao Rossi. A low cost 3d markerless system for the reconstruction of athletic techniques. In *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 222–229. IEEE, 2013.

[14] Carlo Castagna, Marco Ganzetti, Massimiliano Ditroilo, Marco Giovannelli, Alessandro Rocchetti, and Vincenzo Manzi. Concurrent validity of vertical jump performance assessment systems. *The Journal of Strength & Conditioning Research*, 27(3):761–768, 2013.

[15] Robert Manske and Michael Reiman. Functional performance testing for power and return to sports. *Sports Health*, 5(3):244–250, 2013.

[16] Luis F Aragón. Evaluation of four vertical jump tests: Methodology, reliability, validity, and accuracy. *Measurement in physical education and exercise science*, 4(4):215–228, 2000.

[17] Steven Khuu, Lindsay L Musalem, and Tyson AC Beach. Verbal instructions acutely affect drop vertical jump biomechanics—implications for athletic performance and injury risk assessments. *The Journal of Strength & Conditioning Research*, 29(10):2816–2826, 2015.

[18] Robert Wood. "countermovement jump.", 2008. https://www.topendsports.com/testing/tests/bosco-counter-movement-jump.htm.

[19] Luca Petrigna, Bettina Karsten, Giuseppe Marcolin, Antonio Paoli, Giuseppe D'Antona, Antonio Palma, and Antonino Bianco. A review of countermovement and squat jump testing methods in the context of public health examination in adolescence: reliability and feasibility of current testing procedures. *Frontiers in Physiology*, 10:1384, 2019.

[20] W Young, CH MacDonald, T Heggen, and J Fitzpatrick. An evaluation of the specificity, validity and reliability of jumping tests. *The Journal of sports medicine and physical fitness*, 37(4):240–245, 1997.

[21] George Beckham, Tim Suchomel, and Satoshi Mizuguchi. Force plate use in performance monitoring and sport science testing. *New Studies in Athletics*, 29(3):25–37, 2014.

[22] José Afonso, Israel Teoldo da Costa, Miguel Camões, Ana Silva, Ricardo Franco Lima, André Milheiro, Alexandre Martins, Lorenzo Laporta, Fábio Yuzo Nakamura, and Filipe Manuel Clemente. The effects of agility ladders on performance: A systematic review. *International journal of sports medicine*, 2020.

[23] Chris Richter, Enda King, Siobhan Strike, and Andrew Franklyn-Miller. Objective classification and scoring of movement deficiencies in patients with anterior cruciate ligament reconstruction. *PloS one*, 14(7):e0206024, 2019.

[24] Kevin R Ford, Gregory D Myer, Rose L Smith, Robyn N Byrnes, Sara E Dopirak, and Timothy E Hewett. Use of an overhead goal alters vertical jump performance and biomechanics. *The Journal of Strength & Conditioning Research*, 19(2):394–399, 2005.

[25] Naruhiro Hori, Robert U Newton, Naoki Kawamori, Michael R McGuigan, William J Kraemer, and Kazunori Nosaka. Reliability of performance measurements derived from ground reaction force data during countermovement jump and the influence of sampling frequency. *The Journal of Strength & Conditioning Research*, 23(3):874–882, 2009.

[26] Franco M Impellizzeri, Ermanno Rampinini, Nicola Maffiuletti, and Samuele M Marcora. A vertical jump force test for assessing bilateral strength asymmetry in athletes. *Medicine & Science in Sports & Exercise*, 39(11):2044–2050, 2007.

[27] Ricardo Peterson Silveira, Pro Stergiou, Felipe P Carpes, Flávio A de S Castro, Larry Katz, and Darren J Stefanyshyn. Validity of a portable force platform for assessing biomechanical parameters in three different tasks. *Sports biomechanics*, 16(2):177–186, 2017.

[28] Matthew J Jordan, Per Aagaard, and W Herzog. Lower limb asymmetry in mechanical muscle function: a comparison between ski racers with and without acl reconstruction. *Scandinavian journal of medicine & science in sports*, 25(3):e301–e309, 2015.

[29] Marc Madruga-Parera, Chris Bishop, Azahara Fort-Vanmeerhaeghe, Maria R Beltran-Valls, Oliver G Skok, and Daniel Romero-Rodríguez. Interlimb asymmetries in youth tennis players: Relationships with performance. *The Journal of Strength & Conditioning Research*, 34(10):2815–2823, 2020.

[30] Georgios Papaiakovou. Kinematic and kinetic differences in the execution of vertical jumps between people with good and poor ankle joint dorsiflexion. *Journal of sports sciences*, 31(16):1789–1796, 2013.

[31] Lee Herrington and Allan Munro. Drop jump landing knee valgus angle; normative data in a physically active population. *Physical Therapy in Sport*, 11(2):56–59, 2010.

[32] Allan Munro, Lee Herrington, and Paul Comfort. The relationship between 2-dimensional knee-valgus angles during single-leg squat, single-leg-land, and drop-jump screening tests. *Journal of sport rehabilitation*, 26(1):72–77, 2017.

[33] Odd-Egil Olsen, Grethe Myklebust, Lars Engebretsen, and Roald Bahr. Injury mechanisms for anterior cruciate ligament injuries in team handball: a systematic video analysis. *The American journal of sports medicine*, 32(4):1002–1012, 2004.

[34] PHILIP K Schot, BARRY T Bates, and JANET S Dufek. Bilateral performance symmetry during drop landing: a kinetic analysis. *Medicine and science in sports and exercise*, 26:1153–1153, 1994.

[35] Annemie Smeets, Bart Malfait, Bart Dingenen, Mark A Robinson, Jos Vanrenterghem, Koen Peers, Stefaan Nijs, Styn Vereecken, Filip Staes, and Sabine Verschueren. Is knee neuromuscular activity related to anterior cruciate ligament injury risk? a pilot study. *The Knee*, 26(1):40–51, 2019.

[36] Chris Centeno. Can you treat a valgus knee with severe arthirtis without surgery?, June 2020. https://regenexx.com/blog/valgus-knee/\#gref.

[37] Timothy E Hewett, Gregory D Myer, Kevin R Ford, Robert S Heidt Jr, Angelo J Colosimo, Scott G McLean, Antonie J Van den Bogert, Mark V Paterno, and Paul Succop. Biomechanical measures of neuromuscular control and valgus loading of the knee predict anterior cruciate ligament injury risk in female athletes: a prospective study. *The American journal of sports medicine*, 33(4):492–501, 2005.

[38] Eirik Kristianslund and Tron Krosshaug. Comparison of drop jumps and sport-specific sidestep cutting: implications for anterior cruciate ligament injury risk screening. *The American journal of sports medicine*, 41(3):684–688, 2013.

[39] Mark V Paterno, Kevin R Ford, Gregory D Myer, Rachel Heyl, and Timothy E Hewett. Limb asymmetries in landing and jumping 2 years following anterior cruciate ligament reconstruction. *Clinical Journal of Sport Medicine*, 17(4):258–262, 2007.

[40] Helen C Smith, Robert J Johnson, Sandra J Shultz, Timothy Tourville, Leigh Ann Holterman, James Slauterbeck, Pamela M Vacek, and Bruce D Beynnon. A prospective evaluation of the landing error scoring system (less) as a screening tool for anterior cruciate ligament injury risk. *The American journal of sports medicine*, 40(3):521–526, 2012.

[41] Darin A Padua, Michelle C Boling, Lindsay J DiStefano, James A Onate, Anthony I Beutler, and Stephen W Marshall. Reliability of the landing error scoring system-real time, a clinical assessment tool of jump-landing biomechanics. *Journal of sport rehabilitation*, 20(2):145–156, 2011.

[42] Darin A Padua, Lindsay J DiStefano, Anthony I Beutler, Sarah J De La Motte, Michael J DiStefano, and Steven W Marshall. The landing error scoring system as a screening tool for an anterior cruciate ligament injury–prevention program in elite-youth soccer athletes. *Journal of athletic training*, 50(6):589–595, 2015.

[43] Kornelius Kraus, Elisabeth Schütz, and Ralf Doyscher. The relationship between a jump-landing task and functional movement screen items: a validation study. *The Journal of Strength & Conditioning Research*, 33(7):1855–1863, 2019.

[44] Kentaro Yagi, Kunihiro Hasegawa, Yuta Sugiura, and Hideo Saito. Estimation of runners' number of steps, stride length and speed transition from video of a 100-meter race. In *Proceedings of the 1st International Workshop on Multimedia Content Analysis in Sports*, pages 87–95, 2018.

[45] Long Sha, Patrick Lucey, Sridha Sridharan, Stuart Morgan, and Dave Pease. Understanding and analyzing a large collection of archived swimming videos. In *IEEE Winter Conference on Applications of Computer Vision*, pages 674–681. IEEE, 2014.

[46] Steffi L Colyer, Murray Evans, Darren P Cosker, and Aki IT Salo. A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system. *Sports medicine-open*, 4(1):24, 2018.

[47] Sanjay Saini, Nordin Zakaria, Dayang Rohaya Awang Rambli, and Suziah Sulaiman. Markerless human motion tracking using hierarchical multi-swarm cooperative particle swarm optimization. *PLoS One*, 10(5):e0127833, 2015.

[48] Ahmed Elhayek, Carsten Stoll, Kwang In Kim, and Christian Theobalt. Outdoor human motion capture by simultaneous optimization of pose and camera parameters. In *Computer Graphics Forum*, volume 34, pages 86–98. Wiley Online Library, 2015.

[49] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):1–14, 2017.

[50] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[51] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[52] Ahmed Elhayek, Edilson de Aguiar, Arjun Jain, Jonathan Tompson, Leonid Pishchulin, Micha Andriluka, Chris Bregler, Bernt Schiele, and Christian Theobalt. Efficient convnet-based marker-less motion capture in general scenes with a low number of cameras. In *Pro-*

ceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3810–3818, 2015.

[53] John J McMahon, Timothy J Suchomel, Jason P Lake, and Paul Comfort. Understanding the key phases of the countermovement jump force-time curve. *Strength & Conditioning Journal*, 40(4):96–106, 2018.

[54] Christopher J Sole. *Analysis of countermovement vertical jump force-time curve phase characteristics in athletes*. PhD thesis, East Tennessee State University, 2015.

[55] Eamonn P Flanagan and Thomas M Comyns. The use of contact time and the reactive strength index to optimize fast stretch-shortening cycle training. *Strength & Conditioning Journal*, 30(5):32–38, 2008.

[56] Jason S Pedley, Rhodri S Lloyd, Paul Read, Isabel S Moore, and Jon L Oliver. Drop jump: A technical model for scientific application. *Strength & conditioning journal*, 39(5):36–44, 2017.

[57] Frank R Noyes, Sue D Barber-Westin, Cassie Fleckenstein, Cathy Walsh, and John West. The drop-jump screening test: difference in lower limb control by gender and effect of neuromuscular training in female athletes. *The American journal of sports medicine*, 33(2):197–207, 2005.

[58] Nicholas P Linthorne. Analysis of standing vertical jumps using a force platform. *American Journal of Physics*, 69(11):1198–1204, 2001.

[59] Guillaume Laffaye, Phillip P Wagner, and Tom IL Tombleson. Countermovement jump height: Gender and sport-specific differences in the force-time variables. *The Journal of Strength & Conditioning Research*, 28(4):1096–1105, 2014.

[60] Randy J Schmitz, John C Cone, Amanda J Tritsch, Michele L Pye, Melissa M Montgomery, Robert A Henson, and Sandra J Shultz. Changes in drop-jump landing biomechanics during prolonged intermittent exercise. *Sports Health*, 6(2):128–135, 2014.

[61] Standing vertical jump. http://people.brunel.ac.uk/~spstnpl/BiomechanicsAthletics/VerticalJumping.htm.

[62] Robert Wood. "drop jump test.", September 2019. https://www.topendsports.com/testing/tests/drop-jump.htm.

[63] Mitch Hauschildt. Landing mechanics: What, why, and when. http://myweb.facstaff.wwu.edu/chalmers/PDFs/Landing\%20mechanics.pdf.

[64] Carl Valle. The complete guide to vertical jump testing for coaches. https://simplifaster.com/articles/vertical-jump-testing/.

[65] Alan Agresti. *Categorical data analysis*, volume 482. John Wiley & Sons, 2003.

[66] Joseph Cavanaugh et al. Encyclopedia of statistical sciences . samuel kotz, campbell b. read, n. balakrishnan, and brani vidakovic. *Journal of the American Statistical Association*, 102:1074–1075, 2007.

[67] W Paul Vogt and Burke Johnson. *Dictionary of statistics & methodology: A nontechnical guide for the social sciences*. Sage, 2011.

[68] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[69] James Bergstra, Daniel Yamins, and David Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International conference on machine learning*, pages 115–123. PMLR, 2013.

[70] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arxiv 2015. *arXiv preprint arXiv:1512.03385*, 2015.

[71] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7083–7093, 2019.

[72] Alicia M Montalvo, Daniel K Schneider, Laura Yut, Kate E Webster, Bruce Beynnon, Minin-der S Kocher, and Gregory D Myer. "what's my risk of sustaining an acl injury while play-ing sports?" a systematic review with meta-analysis. *British journal of sports medicine*, 53(16):1003–1012, 2019.

[73] Jesper Bencke, Per Aagaard, and Mette K Zebis. Muscle activation during acl injury risk movements in young female athletes: a narrative review. *Frontiers in physiology*, 9:445, 2018.

[74] Hossein Mokhtarzadeh, Katie Ewing, Ina Janssen, Chen-Hua Yeow, Nicholas Brown, and Peter Vee Sin Lee. The effect of leg dominance and landing height on acl loading among female athletes. *Journal of biomechanics*, 60:181–187, 2017.

[75] Miho J Tanaka, Lynne C Jones, and Jared M Forman. Awareness of anterior cruciate ligament injury-preventive training programs among female collegiate athletes. *Journal of athletic training*, 55(4):359–364, 2020.

[76] Natalie Voskanian. Acl injury prevention in female athletes: review of the literature and practical considerations in implementing an acl prevention program. *Current reviews in musculoskeletal medicine*, 6(2):158–163, 2013.

[77] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2334–2343, 2017.