DETERMINING SUCCESS IN SNATCH WEIGHTLIFTING USING POSE LANDMARKS AND BARBELL DETECTION

BY

MING QI

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF ENGINEERING
IN ELECTRICAL AND COMPUTER ENGINEERING
COLLEGE OF ENGINEERING**

**GRADUATE SCHOOL, RANGSIT UNIVERSITY
ACADEMIC YEAR 2024**

Thesis entitled

DETERMINING SUCCESS IN SNATCH WEIGHTLIFTING USING POSE LANDMARKS AND BARBELL

by

MING QI

was submitted in partial fulfillment of the requirements
for the degree of Master of Engineering in Electrical and Computer Engineering

Rangsit University
Academic Year 2024

Assoc.Prof. Opas Chutatape, Ph.D.
Examination Committee Chairperson

Asst.Prof. Supattana Nirukkanaporn, D.Eng.
Member

Assoc.Prof. Rong Phoophuangpairoj, Ph.D.
Member and Advisor

Approved by Graduate School

(Prof. Suejit Pechprasarn, Ph.D.)
Dean of Graduate School
October 31, 2024

Acknowledgements

I would like to extend my sincere gratitude to my advisor, Assoc. Prof. Rong Phoophuangpairoj, Ph.D., for his invaluable guidance and profound insights throughout my research journey, particularly in selecting and exploring the research topic. His mentorship has been instrumental in shaping my academic pursuits and enhancing my understanding of this field.

Ming Qi

Researcher

6509121	:	Ming Qi
Thesis Title	:	Determining Success in Snatch Weightlifting Using Pose Landmarks and Barbell Detection
Program	:	Master of Engineering in Electrical and Computer Engineering
Thesis Advisor	:	Assoc.Prof. Rong Phoophuangpairoj, Ph.D.

Abstract

This research integrated computer vision and machine learning techniques to objectively evaluate snatch weightlifting success. By leveraging MediaPipe for skeletal detection and You Only Look Once (YOLO) object detection for barbell detection, the study classified snatch into six phases. An artificial neural network (ANN) and support vector machine (SVM) were applied to classify weightlifting phases from features extracted using MediaPipe. The distances between an athlete's hands and a barbell were computed using the MediaPipe features, which represented the points on an athlete's right and left hands as well as the points on a barbell. This study employed different methods to evaluate weightlifting success. For example, the method that used the holding period of the sixth phase could obtain a 95% accuracy rate, whereas the method that evaluated the presence of all six phases in sequence could derive a lower accuracy of 70%. A method that evaluated the ordered six phases, the holding time of the sixth phase, and the barbell slipping achieved the highest accuracy rate of 100%. The proposed method, which did not require specialized equipment, could achieve notable weightlift phase classification and efficiently determine the success or failure of snatch weightlifting.

)Total 111 pages(

Keywords: machine learning, MediaPipe, object detection, posture landmarks, snatch weightlifting

Student's Signature..Thesis Advisor's Signature...

Table of Contents

Table of Contents (continued)

List of Tables

List of Tables (continued)

List of Figures

List of Figures (continued)

List of Figures (continued)

Chapter 1

Introduction

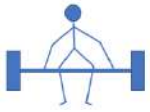
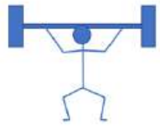
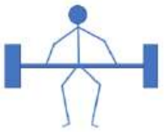
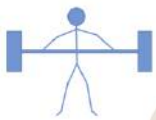
1.1 Background of the Research

Weightlifting is considered one of the greatest tests of strength and power (Ulareanu, Potop, Timnea & Cheran, 2014). Weightlifting is a global sport that is part of the Olympics. People from all over the world follow the weightlifting tournament and cheer on their athletes in the fight for gold. Weightlifting athletes attempt to successfully lift the heaviest weights by lifting a barbell with weight plates up from the floor to overhead. The snatch is one of the two main lifts used in weightlifting. The athlete lifts the barbell up from the floor in a single movement (Géron, 2019).

1.2 Significance of the Research

This research was underscored by its comprehensive exploration of the intersection between weightlifting, computer vision, and machine learning. The integration of MediaPipe and YOLO in addressing pose and a barbell added a layer of complexity and practicality to the study. The following points were investigated in this study: 1) Snatch phase classification: By accurately identifying the six phases of snatch lifting as shown in Table 1.1, the research provided a valuable tool for determining the success of snatch weightlifting. 2) Success/Failure Detection: Training features based on successful and failed lifts empowered the system to method classify attempts. The method removed subjectivity from judging and provided immediate feedback to athletes, enabling real-time adjustments and performance optimization. 3) Accessibility and Generalizability: Because MediaPipe did not rely on specialized equipment. Additionally, this research paved the way for adapting the system to other weightlifting disciplines or even other sports altogether.

Table 1.1 Snatch weightlifting phases

Phase	Posture	Phase	Posture
(1) the first pull		(4) turnover under the barbell	
(2) the transition from the first to the second pull		(5) the catch phase	
(3) the second pull		(6) rising from the squat position (and fully stand)	

Source: Qi & Phoophuangpairoj, 2024

To integrate MediaPipe and YOLO, the study proposed a two-pronged approach to tackle the challenges of snatch weightlifting analysis. 1) MediaPipe for Skeletal Recognition: This open-source framework efficiently extracts skeletal keypoints from video data, captures the athlete's movement patterns during the snatch lifting. Analyzing these keypoints in relation to the six phases allows for phase identification and technique evaluation. 2) YOLO for Barbell Detection: YOLO's object detection capabilities accurately locate the barbell in each frame, enabling its trajectory and interaction with the athlete to be tracked. This information is crucial for understanding barbell path, speed, and overall lift mechanics.

By extracting relevant features from successful and failed lift attempts, the research aims to train a model that can method classify future lifts. This involves: 1) Feature Identification: Key features such as the distance between hands and barbell were extracted from both successful and failed lifts. 2) Model Training: These features were used to train a machine learning model to distinguish snatch weightlifting phases which were used for determining the weightlifting success or not.

Overall, this research presented a promising approach for applying computer vision and deep learning to develop efficient methods to classify weightlifting phase and determine the success or not by using MediaPipe and YOLO, this research has the potential to help referees to judge whether the lifting is successful.

1.3 Significance of the Problem

The challenges associated with weightlifting analysis presented significant barriers to effective training and performance improvement, impacting both professional and amateur athletes. Addressing these issues holds immense potential for athletes, coaches, and the sport as a whole. Here 's why: 1) Subjectivity in performance evaluation: The current reliance on subjective judgments from coaches and judges leads to inconsistencies and biases in technique assessment. 2) Addressing this challenge is promising: Classification of successful and failed lifts eliminates subjectivity from judging, ensuring fair and consistent evaluations, particularly in competitive settings, benefiting both athletes and judges.

Therefore, the investigation of novel methods for objective, accessible analysis of weightlifting, as proposed in this research, assumes significant importance in advancing the sport of weightlifting as a whole.

1.4 Research Objectives

The research objectives include: 1) Develop a method for classifying the phases of snatch weightlifting. 2) Develop a method for determining weightlifting success or failure.

1.5 Scope of the Study

The scope includes: 1) Classify snatch weightlifting phases from images. 2) Classify the success of snatch weightlifting based on short videos. 3) Use front-view snatch weightlifting images and videos. 4) Weightlifting videos consisted of only the weightlifting scenes.

1.6 Research Framework

The research framework includes: 1) Extract images from a snatch weightlifting video. 2) Classify the extracted images into phases. 3) Detect a barbell in a image. 4) Compute the distances between weightlifter's hands and a barbell. 5) Based on the list of recognized phase for each frame in the video sequence, determine the order of the weightlifting video sequence, phase duration information, and assess the success of the lift. 6) Determine the success in weightlifting based on the list of recognized phase for each frame in the video sequence and the distance between weightlifter's hands and barbells.

1.7 Definition of Terms

Term 1 Skeleton behavioral recognition

Skeleton behavioral recognition utilizes principles from computer vision and deep learning to detect and analyze human behavior (Patil, Rao, Utturwar, Shelke & Sard, 2022). This process focuses on interpreting the positions and movements of skeletal joints. In the context of weightlifting, it plays a crucial role in understanding and assessing the biomechanics of athletes' movements during lifts. This study primarily relied on information obtained from 33 key points, including the head, limbs, and torso, using MediaPipe.

Term 2 Object recognition

Object recognition is a computer vision technology that identifies, detects, and locates specific objects, items, or patterns within visual data, typically derived from images or videos (Howard et al., 2017). Due to the unique characteristics of the barbell in this study, the standard YOLO model is unable to detect and provide the position. To overcome this limitation, a custom barbell model was trained using sampled images from weightlifting videos. This ensures the effectiveness of recognizing weightlifting actions and a barbell as well as provides reference data to determine the success or failure of a lift.

Term 3 MediaPipe and YOLO integration

MediaPipe and YOLO were applied for skeletal feature extraction and object detection, respectively. This study used MediaPipe to extract skeletal features of an athlete and YOLO to detect a barbell.

Term 4 Image training model and feature training model

This study provided two image classification solutions for the detection weightlifting phases. The image training model involves classifying images based on different behavioral phases, trained using algorithms such as Convolutional Neural Networks (CNNs). The feature-based models for phase classification were obtained from extracted features using MediaPipe and YOLO which were trained using algorithms such as ANN and SVM.

Term 5 Landmark

Landmarks referred to the 33 skeletal coordinates and two barbell coordinates extracted via MediaPipe. These landmarks played a crucial role in the research as they form the features for classifying phases and determining the success or not.

Term 6 Snatch lifting phases

The phases of a snatch lift contain 1) the first pull, 2) the transition from the first to the second pull, 3) the second pull, 4) the turnover under the barbell, 5) the catch phase, and 6) rising from the squat position (and fully standing).

Term 7 Basic and extended features of weightlifting

The features are categorized into 3 groups: skeleton landmark coordinates (66 features, with 33 coordinates each along both the x and y axes); barbell landmark coordinates (4 features, with 2 coordinates each along both the x and y axes). These 70 basic features were used for classifying weightlifting phases; additionally, there are 2 extended features representing the distances between each hand and the barbell, which were used for determining success. The total number of features is 72.

Term 8 Distance between barbell and hand

The distance involves treating the two barbell points as a spatial vector. The distances from the wrist points (landmarks 15 and 16 in MediaPipe) to the barbell vector line were used to compute the closest distance between the hand and the barbell. Then the distance was divided by the length of a barbell. A proportion threshold, such as 0.3, indicated a dropped barbell. The diastace was applied to judge whether the barbell slipped from hands.

Term 10 Sequence of a video

A video sequence consists of continuous frames from a weightlifting video. This sequence was used in the snatch weightlifting classification and the determination of weightlifting success.

Chapter 2

Literature Review

The recognition of weightlifting activities in this study encompasses image recognition, behavioral recognition, skeleton detection, and object recognition. Image recognition primarily relies on classification methods within CNN models. Behavioral recognition involves the temporal analysis of results obtained from 2D CNN image recognition, representing a process of temporal image sequence processing and analysis. This temporal analysis encompasses object recognition and feature extraction. This part covered weightlifting posture, image recognition, feature extraction and feature classification, action recognition, skeletal recognition, and object recognition.

2.1 Weightlifting Posture

Several studies focused on classifying snatch lifting phases to improve technique analysis. For example, (Korkmaz & Harbili, 2015; Korayem et al., 2010) developed methods to accurately segment and evaluate each phase, enhancing feedback for athletes and increasing the precision of performance assessments. They divided the snatch weightlifting into six phases, as shown in figure 2.1.

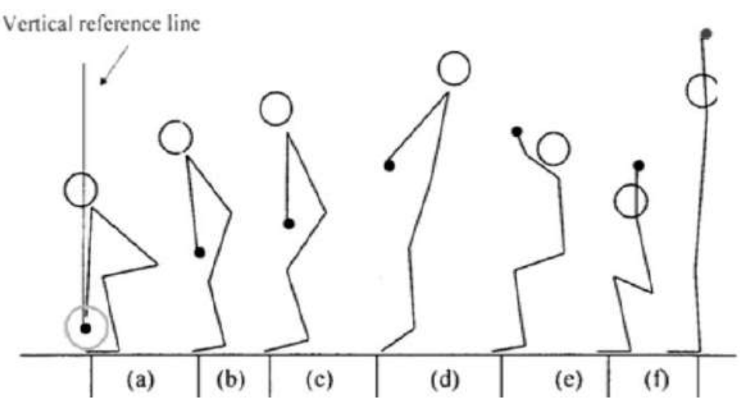


Figure 2.1 Snatch weightlifting phases

Source: Korkmaz & Harbili, 2015; Korayem, Mustafa, Korayem & Amanati, 2010

2.2 Image Recognition

Image recognition, as a pivotal domain in computer vision, has undergone a remarkable evolution over the years, shaping the landscape of visual perception and paving the way for advancements in various applications. The history of image recognition can be traced back to the early days of computer vision research when basic pattern recognition techniques were employed to discern primitive features within images.

The initial phase of image recognition primarily revolved around handcrafted feature extraction methods, where researchers meticulously designed algorithms to identify specific patterns or edges in images. However, these early approaches were limited by their reliance on predefined features, making them susceptible to variations in lighting conditions, scale, and orientation.

A transformative breakthrough occurred with the advent of machine learning and the introduction of more sophisticated techniques in the late 20th century. The emergence of CNNs marked a paradigm shift in image recognition, allowing systems to automatically learn hierarchical representations of features directly from raw pixel data. LeCun et al. (1998) examined the utilization of gradient-based learning in document recognition, demonstrating its utility in advancing recognition technology.

The early 21^{st} century witnessed the ascent of deep learning, further propelling the capabilities of image recognition systems. Notably, the ImageNet large-scale visual recognition challenge played a pivotal role, in the development of increasingly sophisticated CNN architectures. Krizhevsky, Sutskever & Hinton (2017) investigated the application of deep convolutional neural networks in image recognition and achieved breakthrough results in the ImageNet challenge for large-scale visual recognition.

Géron (2019) examined the developments in CNN architectures such as AlexNet, VGGNet, and ResNet. These models significantly improved image recognition and reduced top-5 error rates in competitions. AlexNet achieved a top-5 error rate of 17%, VGGNet introduced a simple yet effective architecture, and ResNet pioneered residual learning for training extremely deep CNNs.

Despite these strides, challenges persist in image recognition. Robustness to adversarial attacks, interpretability of complex models, and the need for large labeled datasets are among the ongoing research fronts. Addressing these challenges is crucial to enhancing the reliability and practical applicability of image recognition systems.

2.2.1 Background of Image Recognition

Image recognition technology is an important branch of computer science, which studies how to extract information from images. Image recognition technology has applications in many fields, including medical, transportation, security, etc. In the field of weightlifting, image recognition technology has also been widely researched and applied.

Early weightlifting image recognition technology was mainly used for technical analysis of weightlifters. Researchers use image recognition technology to analyze weightlifters' movements, postures, timing of force exertion, etc. to help athletes improve their technical level. For example, Korayem (2010) examined a weightlifter's technical analysis method based on image recognition. This method could identify the weightlifter's starting posture, push-up, bench press and other actions, and provide technical improvement suggestions.

With the development of image recognition technology, its application in the field of weightlifting is becoming more and more extensive. In recent years, researchers have made some progress in the application of image recognition technology to the field of weightlifting. For example, Ulareanu et al. (2014) studied a weightlifting penalty judgment method based on image recognition. This method can automatically identify whether a weightlifter is overweight, whether he has landed, etc., and improves the accuracy of penalty decisions. Olaya-Mira, Soto-Cardona, Palacio-Peña and Acevedo-

Tangarife (2020) studied a weightlifter training method based on image recognition. This method can track the weightlifter's movement trajectory, analyze the weightlifter's power output, etc., and help athletes train better.

The application of image recognition technology in the field of weightlifting is still in its infancy. With the continuous development of image recognition technology, its application in the field of weightlifting will become more extensive. Image recognition technology will be more real-time and can analyze the movements and postures of weightlifters in real time to improve the efficiency of judgment and training. The technology is becoming more diverse and can be combined with other technologies such as artificial intelligence, machine learning, etc. to improve the performance of an application.

2.2.2 Characteristics of Image Recognition

Image recognition technology has a number of features that make it suitable for application in weightlifting. These features include: 1) Accuracy: Image recognition technology can automatically identify athlete movements with high accuracy. 2) Efficiency: Image recognition technology can automatically perform judging and training, improving work efficiency. 3) Objectivity: Image recognition technology can avoid the influence of human factors, improving the objectivity of judging and training. 4) Diversification: Image recognition technology will become more diversified and can be combined with other technologies, such as artificial intelligence, machine learning, etc., to improve application effects.

2.2.3 Types of Image Recognition

Various image recognition architectures find applications in weightlifting, each bringing distinct advantages to the field. These architectures include Géron (2019) suggested: 1) Standard CNN models, with their established structures, serve as common choices for weightlifting image recognition tasks. These models provide a benchmark for comparison and excel at capturing essential features from weightlifting-related images. 2) ResNet, a sophisticated CNN architecture, stands out for its effectiveness in diverse image recognition tasks. Despite its complexity, ResNet proves capable of discerning intricate patterns within images, making it a robust choice for in-depth weightlifting analysis. 3) MobileNet reduces the traditional convolution process used in standard convolutional neural networks to a depth-wise separable convolution, which comprises of depth-wise and point-wise convolution.

This diverse array of architectures allows for a comparative analysis of their practical model accuracies, paving the way for the selection of optimal models for subsequent research endeavors. The consideration of complexity, efficiency, and adaptability ensures a well-rounded approach to weightlifting image recognition.

2.3 Feature Extraction and Feature Classification

Feature extraction and feature classification play pivotal roles in the realm of machine learning and pattern recognition, apply to the accurate and efficient analysis of complex data.

2.3.1 Historical Overview of Feature Extraction and Classification

Traditional Methods: Lowe (2004) introduced a method for extracting distinctive image features. During this period, feature extraction heavily relied on traditional techniques such as SIFT, SURF, and HOG, often coupled with classical machine learning algorithms like SVM and KNN (K-Nearest Neighbors) for classification. While these methods found extensive use in tasks such as image processing and object recognition, they required manual feature engineering and struggled with complex datasets.

Emergence of Deep Learning: Krizhevsky et al. (2017) achieved significant results in the ImageNet classification task using deep convolutional neural networks. The advent of deep learning, notably with the introduction of CNNs like AlexNet in 2012, marked a paradigm shift in feature extraction and classification. Deep learning models enabled the extraction of abstract and hierarchical features directly from raw data, eliminating the need for handcrafted features.

End-to-End Learning and Pre-trained Models: Howard et al. (2017) proposed MobileNets, an efficient convolutional neural network architecture for mobile and embedded vision applications. The focus shifted towards end-to-end learning and the utilization of pre-trained deep learning models. Models such as VGG, ResNet, and BERT, pre-trained on large datasets, learned universal feature representations and were directly applied to classification tasks, yielding significant performance improvements.

Self-Supervised Learning and Reinforcement Learning: Chen, Kornblith, Norouzi and Hinton (2020) presented a simple framework for contrastive learning of visual representations. Recent advancements have seen increased interest in self-supervised learning and reinforcement learning for feature extraction and classification. These methods leverage inherent data structures or agent-environment interactions to learn effective feature representations, promising further breakthroughs in the field.

2.3.2 Background of Feature Extraction and Classification in This Study

In the context of this study, feature extraction and classification are integral to recognizing weightlifting activities. By transforming raw video and image data into meaningful features, it is possible to accurately classify different stages and techniques of weightlifting. This involves using both handcrafted features and features learned through deep learning models.

For this study, focus on extracting spatial and temporal features from weightlifting videos. Spatial features capture the static aspects of the scene, such as the positions and orientations of the weightlifter and the barbell. Temporal features, on the other hand, capture the dynamic aspects, such as the movement trajectories over time.

To classify these features, employ models such as SVMs and ANNs. SVMs are chosen for their robustness in handling high-dimensional data and their effectiveness in binary and multiclass classification tasks. ANNs, particularly deep neural networks, are utilized for their ability to learn complex, hierarchical representations of data, which are essential for capturing the intricate patterns in weightlifting movements..

2.3.3 Characteristics of Feature Extraction and Classification

Characteristics include: 1) Automated Feature Learning: Modern deep learning models can automatically learn relevant features from raw data, significantly reducing the need for manual feature engineering. 2) High Accuracy: Methods like SVMs and ANNs have demonstrated high accuracy in classification tasks, making them suitable for complex applications like weightlifting recognition. 3) Scalability: These methods can handle large datasets and high-dimensional feature spaces, making them scalable for extensive weightlifting analysis. 4) Computational Complexity: Training deep learning models and SVMs can be computationally intensive, requiring significant computational resources. 5) Data Dependency: The performance of these models heavily depends on the quality and quantity of the training data. Poor data quality can lead to suboptimal model performance. 6) Interpretability: Deep learning models, particularly deep neural networks, often operate as "black boxes," making it challenging to interpret the learned features and understand the decision-making process.

2.3.4 Types of Feature Extraction and Classification

2.3.4.1 Feature Extraction Methods

Feature extraction methods such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and CNNs play crucial roles in various computer vision frameworks like MediaPipe and YOLO.

MediaPipe and YOLO are widely used frameworks for tasks like object detection, pose estimation, and hand tracking. They leverage advanced feature extraction techniques to analyze and process visual data effectively.

In MediaPipe, which is a popular framework for real-time perception tasks, CNNs are often employed to extract features from input images or video frames. These features are then used for tasks such as hand tracking, pose estimation, and facial recognition. The hierarchical features learned by CNNs through convolutional layers enable accurate and efficient detection and tracking of objects and body parts.

YOLO, which is an efficient object detection framework, utilizes CNNs to extract features from images or video frames. These features are then processed to detect and classify objects in real-time. YOLO's architecture is optimized for speed and accuracy, making it suitable for applications requiring fast and precise object detection.

In both MediaPipe and YOLO, the choice of feature extraction method depends on the specific task requirements, computational resources, and desired level of accuracy. While CNNs offer powerful feature extraction capabilities, they may require significant computational resources and may overfit with limited data. On the other hand, techniques like PCA and LDA provide dimensionality reduction and classification capabilities but may not capture complex spatial hierarchies present in images as effectively as CNNs.

2.3.4.2 Feature Classification Methods

SVM: SVMs are find the optimal hyperplane for classifying data into different categories. They are effective in high-dimensional spaces and are used for both classification and regression tasks.

ANNs: An ANN consists of interconnected layers of neurons that can learn complex patterns in data. They are highly flexible and can be adapted for various classification tasks.

KNN: KNN is a simple, instance-based learning algorithm that classifies data points based on the majority class of its k-nearest neighbors. It is easy to implement and effective for small datasets.

SVM offers robust classification in high-dimensional spaces but can be sensitive to the choice of kernel and parameters. ANN, which includes deep neural networks, provides flexibility and can capture complex patterns; however, it requires large amounts of labeled data and substantial computational resources for training. KNN is simple and effective for small datasets but can be computationally intensive during inference and is sensitive to noise. Each method has its advantages and limitations in feature classification.

2.4 Action Recognition

2.4.1 Action Recognition

Action recognition, a pivotal domain within computer vision, has witnessed substantial evolution over distinct phases, each characterized by diverse methodologies and approaches. In its nascent stages, researchers predominantly embraced conventional techniques such as Improved Dense Trajectories (IDT) Xu, Zhou, Yuan and Huang (2021) examined relying on manually engineered features and traditional machine learning for classification. While intuitive, these methods faced limitations in adapting to intricate scenarios and diverse movements.

The advent of deep learning has ushered in a new era for action recognition, with 2D CNNs standing out as a pivotal technology. Researchers have effectively utilized CNNs to extract spatial features from video frames, thereby significantly improving the precision in recognizing intricate movements. However, the intrinsic focus of 2D CNNs on static images imposes limitations on their ability to model temporal information, Wang, Lu, Jin and Hu (2022) examined, particularly in the context of actions characterized by dynamic variations. To address this, the integration of 2D CNNs with temporal image sequence processing and analysis proves instrumental, synergistically enhancing the method's capability to capture and interpret temporal dynamics. This combined approach, leveraging both spatial and temporal information, stands as a robust strategy for achieving more nuanced and accurate action recognition, especially in scenarios involving dynamic variations.

In action recognition, 3D CNNs excel in capturing temporal dynamics for weightlifting, while Two-Stream Networks concurrently process RGB and optical flow streams, adapting to complex scenarios. Their practical adoption varies, with ongoing refinement for optimal performance in weightlifting studies.

Feichtenhofer, Pinz and Wildes (2016) examined the practical adoption of these methodologies using multiple methods, noting that the choice of methodology varies based on the specific requirements of the recognition task and the intricacies of weightlifting movements. Researchers and practitioners leverage these approaches in diverse applications, each offering a unique set of advantages and challenges. In this research, 2D CNNs with temporal sequence analysis will be predominantly leveraged, showcasing a dynamic interplay that reflects ongoing exploration and refinement to tailor these methodologies for optimal performance in weightlifting studies.

2.4.1.1 Time Series Analysis

Time Series Analysis is a statistical technique that deals with time-ordered data. Its primary goal is to understand the underlying structure and patterns within the data to make forecasts, detect anomalies, or extract meaningful insights. Time series data are typically collected at successive points in time, spaced at uniform intervals, and can be represented as a sequence of data points indexed in time order. Action recognition integrates spatial features and is geared towards understanding dynamic activities in video, whereas time series analysis is more focused on uncovering temporal patterns and making predictions based on numerical time-ordered data.

2.4.1.2 Processing of Action Recognition and Time Series

Action recognition involves collecting video data containing various actions, preprocessing it to reduce noise and normalize frames, and then extracting spatial and temporal features using algorithms like CNNs or Optical Flow. These features are classified using machine learning or deep learning models such as 2D CNNs, 3D CNNs, LSTMs (Donahue et al., 2015; Feichtenhofer et al.,2016), or Two-Stream Networks (Simonyan & Zisserman, 2014). The results are refined in the post-processing step to enhance accuracy.

Time series analysis involves collecting sequential data points over time, preprocessing to clean and prepare the data, and extracting meaningful features using statistical methods like Fourier Transform and Wavelet Transform. Base above, Aralimarad, Meena and Mallapur (2020) examined, Models such as ARIMA, LSTM, RNN, or TCNs are then applied for forecasting or pattern detection, including programmatic logic for decision-making, followed by post-processing to analyze, validate, and refine the model outputs.

2.4.2 Types of Action Recognition

2.4.2.1 Direct Video-based for Action Recognition

3D CNNs: 3D CNNs directly process video data by convolving over spatial and temporal dimensions, allowing them to capture both spatial and temporal features. This method analyzes motion patterns and recognizes actions directly from video sequences. The output of 3D CNNs includes action labels for classification or features describing specific actions in video segments, providing insights into the recognized actions within the video.

Two-Stream Networks: Two-Stream Networks simultaneously process RGB video frames and optical flow data. By utilizing parallel streams, they capture both spatial and temporal information, enabling a comprehensive analysis of motion patterns in videos. The output of Two-Stream Networks consists of action labels for classification or features describing specific actions in video segments, offering a detailed understanding of the actions present in the video, incorporating both spatial and temporal information.

Improved Dense Trajectories (IDT): IDT extracts dense trajectories from video sequences and analyzes motion patterns using handcrafted features such as optical flow, HOG, and HOF. This method focuses on capturing detailed motion trajectories to characterize different actions present in the video. The output of IDT is motion trajectory features used as inputs to subsequent classifiers for action recognition. These features provide detailed information about the motion patterns within the video, aiding in the accurate identification of actions.

2.4.2.2 Feature Extraction of Sequence Analysis for Action Recognition

Optical Flow: Optical flow detects motion between consecutive frames in a video, extracting motion patterns. It directly analyzes the pixel-level changes between frames to identify movement, providing valuable insights into object motion within the video. The output comprises optical flow features capturing the motion of objects, which serve as a basis for further sequence analysis.

Improved Dense Trajectories (IDT): IDT extracts dense trajectories from video sequences and integrates them with handcrafted features such as optical flow, HOG, and HOF. By combining multiple motion descriptors, IDT provides a comprehensive representation of motion trajectories in the video. The output includes trajectory features containing information about motion trajectory, direction, and speed, facilitating detailed analysis of motion patterns.

CNN: CNNs extract spatial features from single-frame images or image sequences by applying convolutional and pooling operations. They analyze the visual content of video frames to capture essential features such as edges, textures, and shapes. The output consists of image features representing various visual elements present in the video frames, enabling subsequent sequence analysis based on these extracted features.

Two-Stream Networks: Two-Stream Networks combine RGB images with optical flow features to process both spatial and motion information in videos. By incorporating information from both streams, these networks capture a comprehensive understanding of video content. The output comprises features that integrate spatial and motion information, providing a holistic representation of the video content for further sequence analysis.

Handcrafted Features: Handcrafted features such as HOG and HOF describe motion and texture information in video frames using predefined algorithms. These features capture specific characteristics of the video content, such as object motion and visual patterns. The output includes handcrafted features utilized for subsequent classification or recognition tasks, offering valuable insights into the visual attributes of the video content.

2.4.2.3 Further Sequence Analysis for Action Recognition

LSTM Networks: LSTM networks process feature data extracted from images or videos, specializing in modeling long-term dependencies within time sequences. Their architecture, comprising memory cells and gating mechanisms, allows them to retain and utilize information over extended periods, enabling the capture of complex temporal patterns in video data. The outputs of LSTM networks, such as time sequence predictions, classification labels, or descriptions of specific action occurrence times within the video data, offer valuable insights into the temporal dynamics and patterns present in the video sequences.

Temporal Convolutional Networks (TCNs): TCNs focus on processing feature data extracted from videos or images, leveraging one-dimensional convolutions to capture patterns within time sequences effectively. By utilizing a series of convolutional layers, TCNs extract hierarchical features from the input data, enabling robust modeling of temporal relationships. The outputs of TCNs, including time sequence predictions, classification labels for different actions, or descriptions of action occurrence times within the video data, facilitate the understanding and interpretation of temporal patterns and dynamics encoded in the video sequences.

Recurrent Neural Networks (RNNs): RNNs, including variants like GRUs, excel at processing feature data extracted from videos or images and are tailored to model temporal dynamics within time sequences. By maintaining internal state representations that evolve over time, RNNs capture sequential patterns and dependencies efficiently, enabling the extraction of context and temporal relationships. The outputs of RNNs, such as time sequence predictions, classification labels for different actions, or descriptions of action occurrence times within the video data, provide valuable insights into the temporal evolution of events and actions captured in the video sequences.

2.4.2.4 Integrated Approach for Action Recognition

An integrated approach leveraging open-source components such as MediaPipe and YOLO for both direct video-based action recognition and feature extraction for further sequence analysis. This approach combines the functionalities and benefits of Direct Video-based Action Recognition and Feature Extraction for Further Sequence Analysis, providing a comprehensive solution for processing video data.

Utilizing MediaPipe and YOLO, extract rich spatial and temporal features from video data. MediaPipe offers efficient solutions for pose detection, hand tracking, and facial recognition, providing detailed information about human actions and interactions within the video frames. YOLO, on the other hand, facilitates object detection and tracking, enabling the identification and localization of relevant objects or subjects in the video scenes. By integrating these components, achieve robust feature extraction, capturing both high-level semantics and fine-grained details from the video content.

The extracted features are then passed through a Features Dataset Logical Analysis Layer (FDLAL), which functions similarly to LSTM and RNNs but involves code logic for analysis. The FDLAL processes the sequential feature data, capturing temporal dependencies and patterns present in the video sequences. By leveraging code logic, enhance the sequence analysis process, enabling dynamic decision-making based on predicted action sequences and contextual insights derived from the video data.

2.4.3 Characters of Types of Action Recognition

2.4.3.1 Direct Video-based for Action Recognition

Characters include: 1) Processes video data directly, which simplifies the workflow by eliminating the need for additional feature extraction steps. 2) Methods like 3D CNNs or Two-Stream Networks are capable of capturing both spatial and temporal features in videos, leading to improved accuracy in action recognition. 3) Can directly output classification labels or features of video segments, enabling rapid identification and analysis of specific actions. 4) High computational costs are incurred when dealing with large-scale video data, requiring significant computing resources and time. 5) For complex behaviors like weightlifting, deeper video analysis and model optimization may be necessary to achieve satisfactory results. 6) Susceptibility to the influence of video quality and environmental factors, necessitating high-quality data for accurate recognition.

2.4.3.2 Feature Extraction of Sequence Analysis

Characters include: 1) Extracts video features using methods like optical flow or Improved Dense Trajectories (IDT), capturing richer spatial and temporal information for a more comprehensive understanding of video content. 2) Allows for the extraction of different features tailored to different types of behaviors, enhancing the flexibility and applicability of action recognition methods. 3) Suitable for scenarios requiring further analysis of time sequence data, such as determining action frequency, speed, duration, etc. 4) Requires additional feature extraction steps, which increases the complexity of the processing workflow and computational costs. 5) High requirements for algorithm and parameter selection during feature extraction, potentially requiring tuning for different datasets and scenarios. 6) The quality and effectiveness of feature extraction are influenced by factors such as algorithm selection and parameter settings, necessitating careful design and debugging to ensure optimal performance.

2.4.3.3 Integrated Approach for Action Recognition

Characters include: 1) Comprehensive Solution: By leveraging open-source components such as MediaPipe and YOLO, the approach combines the functionalities of direct video-based action recognition and feature extraction for sequence analysis. This integration provides a holistic solution for processing video data, addressing both spatial and temporal aspects of action recognition. 2) Rich Feature Extraction: Utilizing MediaPipe and YOLO enables the extraction of rich spatial and temporal features from video data. MediaPipe offers efficient solutions for pose detection, hand tracking, and facial recognition, while YOLO facilitates object detection and tracking. This comprehensive feature extraction captures both high-level semantics and fine-grained details from the video content, enhancing the analysis capabilities. 3) Robust Sequence Analysis: The extracted features are passed through a Features Dataset Logical Analysis Layer (FDLAL), which involves code logic for analysis. This layer captures temporal dependencies and patterns present in the video sequences, enabling robust sequence analysis. By leveraging code logic, the approach enhances the sequence analysis process, facilitating dynamic decision-making based on predicted action sequences and contextual insights derived from the video data.

The integrated approach offers a versatile and effective solution for video-based action recognition and sequence analysis, providing enhanced feature extraction and robust temporal analysis capabilities.

2.5 Skeletal Recognition

Skeletal recognition has evolved as a critical component in the trajectory of action recognition, providing a profound understanding of human movements. In the early stages, Improved Dense Trajectories (IDT) and traditional machine learning techniques were primary tools for action recognition, yet they faced challenges in capturing the intricate dynamics of skeletal movements.

The paradigm shifted with the advent of deep learning, ushering in innovative approaches to skeletal recognition. A notable contribution came from Wei, Ramakrishna, Kanade and Sheikh (2016), who introduced a model leveraging Convolutional Pose Machines (CPM) for accurate human pose estimation, signifying a significant leap in sophisticated skeletal representation.

Advancements continued with the exploration of recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) in skeletal modeling. Ren, Liu, Ding and Liu (2024) and Saoudi, Jaafari and Andaloussi (2023) demonstrated the effectiveness of an LSTM-based approach in capturing temporal dependencies, underscoring the importance of sequential skeletal information in action recognition.

The inclusion of 3D pose estimation techniques, exemplified by Berretti et al. (2018), added a layer of depth to skeletal recognition. This advancement enabled models to comprehend not only spatial configurations but also the three-dimensional aspects of human poses.

Recent strides in skeletal recognition involve the fusion of skeletal data with RGB information. Kong, Deng and Jiang (2021) showcased a two-stream network that effectively integrated skeletal and RGB features, contributing to enhanced action recognition.

The advent of frameworks such as MediaPipe and YOLO has introduced new dimensions to skeletal recognition. MediaPipe provides a comprehensive solution for face, hand, and pose detection, while YOLO offers real-time object detection capabilities, influencing the fusion of skeletal and visual information in action recognition systems.

In essence, the evolution of skeletal recognition has transitioned from traditional methods to sophisticated deep learning approaches, progressively refining the understanding of human actions through nuanced skeletal representations.

2.5.1 Characteristics of Skeletal Recognition

The characteristics of skeletal behavior recognition encompass several key features that contribute to the effectiveness of this approach: 1) Temporal Dynamics: Skeletal behavior recognition excels in capturing the temporal dynamics of human movements, allowing for a detailed analysis of actions unfolding over time. 2) Spatial Configuration: The method provides a comprehensive understanding of the spatial configuration of human poses, enabling precise recognition and interpretation of intricate movements. 3) Depth Information: With the integration of 3D pose estimation techniques, skeletal behavior recognition incorporates depth information, enhancing the model's ability to perceive the three-dimensional aspects of human poses. 4) Sequential Dependency: Models leveraging recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) demonstrate a capacity to understand sequential dependencies in skeletal data, crucial for accurate action recognition. 5) Integration with RGB Data: Recent advancements involve the fusion of skeletal data with RGB information, offering a holistic approach to action recognition by combining the strengths of both modalities.

These characteristics collectively make skeletal behavior recognition a powerful and nuanced approach, particularly in the context of weightlifting studies where understanding both spatial and temporal aspects of movements is essential.

2.5.2 Benefits of MediaPipe Skeletal Recognition

MediaPipe brings several benefits to the field of skeletal behavior recognition, making it a valuable tool for comprehensive action analysis: 1) Multi-Modal Capabilities: MediaPipe supports multi-modal skeletal tracking, allowing for the simultaneous analysis of various body parts and movements. This capability enhances the richness of skeletal data for a more detailed understanding of actions. 2) Integration with Diverse Applications: The versatility of MediaPipe allows for seamless integration with diverse applications. Whether applied to fitness tracking, gesture recognition, or interactive experiences, MediaPipe's capabilities extend beyond skeletal behavior recognition, adding value to a range of domains. 3) Community Support and Development: MediaPipe benefits from an active community and ongoing development efforts. This ensures the continuous improvement of the framework, with updates, new features, and optimizations that contribute to its effectiveness in skeletal behavior recognition. 4) Cross-Platform Compatibility: MediaPipe offers cross-platform compatibility, supporting applications across different devices and operating systems. This flexibility enhances the accessibility and usability of the framework in various settings.

2.6 Object Recognition

Object recognition has evolved as a critical facet in the landscape of action recognition, playing a pivotal role in understanding weightlifting movements and enhancing the overall comprehension of complex scenarios. In the early stages of research, authors like LeCun et al. (1998) and Viola and Jones (2001) laid the groundwork for object recognition with landmark works on CNNs and cascaded classifiers.

Over time, the domain has witnessed a paradigm shift towards more sophisticated deep learning techniques, particularly in the context of weightlifting studies. Renowned authors such as Krizhevsky et al. (2017) introduced the groundbreaking AlexNet, significantly advancing the capabilities of CNNs in object recognition tasks. This marked a turning point, as the enhanced depth and complexity of deep neural networks proved instrumental in discerning and classifying objects within weightlifting scenes.

The advent of region-based CNNs (R-CNNs) Girshick (2015) and their subsequent improvements Faster R-CNNs Ren et al. (2016), brought about substantial improvements in both accuracy and efficiency. These approaches revolutionized object recognition by introducing region proposal networks, allowing for selective and focused analysis of specific regions in images, a crucial aspect in weightlifting scenarios where the emphasis is on key objects like barbells and body postures.

Noteworthy contributions by authors like Redmon, Divvala, Girshick and Farhadi (2016) with the introduction of YOLO models and the subsequent evolution to YOLOv3 Redmon and Farhadi (2018) further streamlined object detection tasks. YOLO models, with their real-time processing capabilities, proved valuable in dynamically recognizing and tracking relevant objects during weightlifting activities.

The field of object recognition in weightlifting studies has recently witnessed the integration of advanced frameworks like MediaPipe, as demonstrated by authors such as Lugaresi et al. (2019). MediaPipe, with its multi-modal skeletal tracking and object recognition capabilities, adds an additional layer of sophistication to the analysis, enabling a more holistic understanding of weightlifting scenes.

2.6.1 Benefits of Yolo Object Recognition

YOLO object recognition within the context of weightlifting studies present notable features that contribute to its effectiveness in identifying and analyzing objects such as barbells and body postures. These characteristics include: 1) Bounding Box Predictions: YOLO provides accurate bounding box predictions around detected objects, enabling precise localization. This characteristic is particularly valuable in weightlifting scenarios where identifying the exact location of objects like barbells is crucial for a detailed understanding of the lifting process. 2) Multi-Class Recognition: YOLO supports multi-class object recognition, allowing the model to simultaneously identify and classify various objects within the scene. In weightlifting studies, this feature enables the recognition of different components such as barbells, body positions, and other relevant entities. 3) Robustness to Object Size Variations: YOLO exhibits robustness in handling variations in object sizes. In the context of weightlifting, where the size and position of barbells and body postures can vary, this characteristic ensures that the model can adapt to diverse scenarios commonly encountered in training or competition settings. 4) Accuracy Estimation: YOLO provides mechanisms for accuracy estimation, allowing practitioners to assess the reliability of object recognition results. This characteristic is essential in weightlifting studies, where precise identification of barbells, body postures, and other elements contributes to the accuracy of performance analysis. 5) Customization Capabilities: YOLO offers customization options, enabling researchers and practitioners to tailor the model to specific requirements of weightlifting scenarios. This includes the ability to fine-tune the model on a dataset that reflects the nuances and variations present in weightlifting activities, enhancing the model's adaptability to the unique characteristics of this domain.

Chapter 3

Research Methodology

3.1 Introduction

In the realm of weightlifting behavior analysis, the determination of snatch weightlifting success requires a multifaceted approach that integrates advanced machine learning techniques and pose landmark features. To provide insight about the steps used throughout this process, the steps to determine the success of snatch weightlifting are illustrated in Figure 3.1.

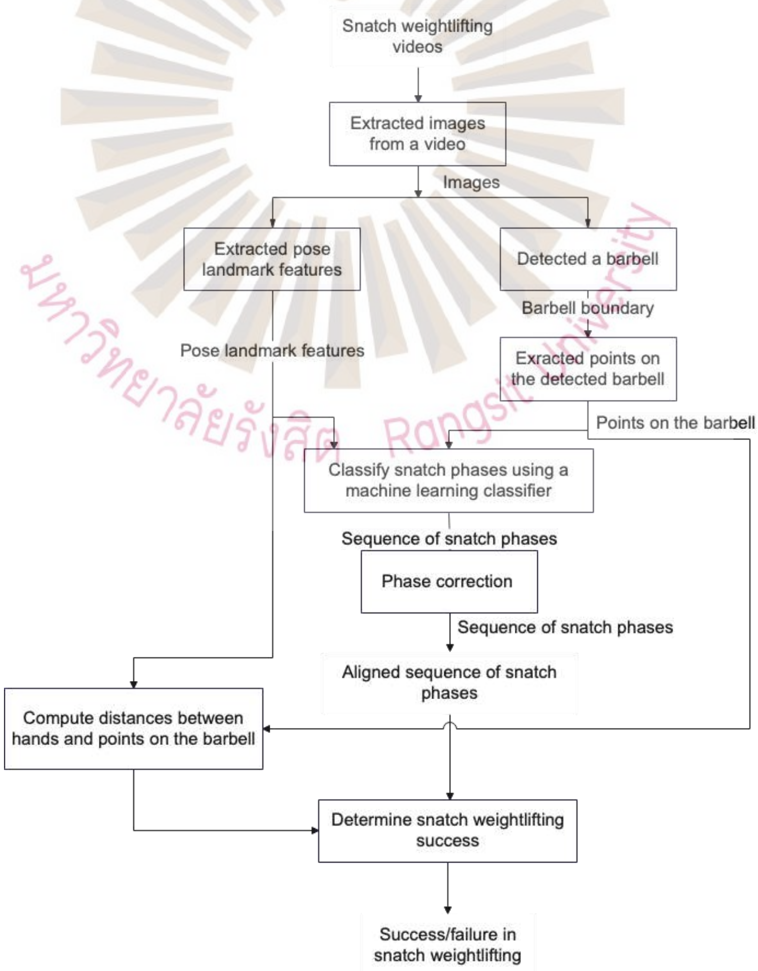


Figure 3.1 Steps to determine the success of snatch weightlifting

Figure 3.1 showed the method for determining success in snatch weightlifting. First, extract images from a video. The pose landmark features were extracted. A barbell was detected and the barbell features consisting of 2 points on it were extracted. The pose landmark features and the barbell features were combined and used to classify the snatch weight lifting phases. After a sequence of weight lifting phases was determined, the phases were corrected or aligned using rules. On the other hand, the distance between hands and a barbell was calculated from the features extracted from each image. Finally, the distances of the hands from the barbell and the aligned phase sequence were used to determine success in snatch weightlifting.

To create more understanding, the steps to detect a barbell, classify snatch phases using machine learning classifiers, and determine snatch weightlifting success were shown in Figures 3.2–3.4.

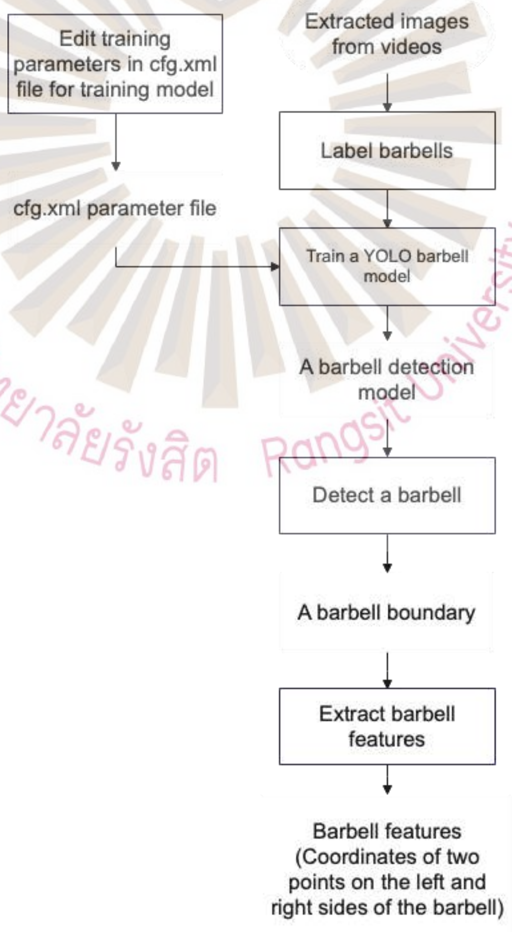


Figure 3.2 Steps to detect a barbell

Figure 3.2 showed the steps for detecting a barbell using the YOLO object detection model. First, frames were extracted from weightlifting videos and converted into still images. Then these images were annotated with bounding boxes to label the barbell. A YOLO model's configuration file was edited to set training parameters, and the model was trained using the annotated images. Once trained, the model could detect barbells in the videos. Fanally, the features, which were the coordinates of two points on the left and right sides of the barbell, were computed from the bounding box.

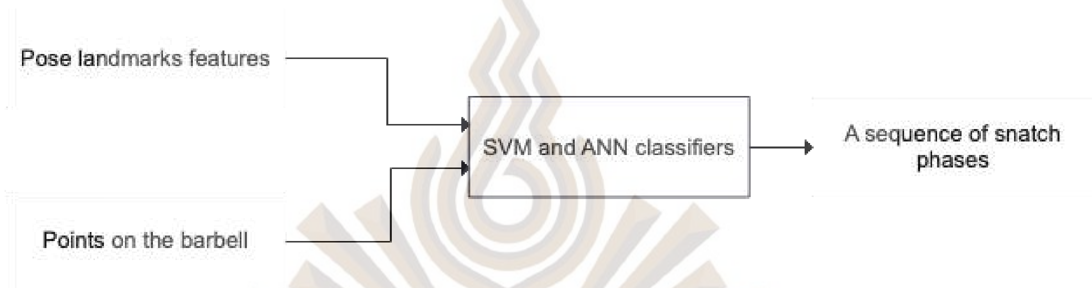


Figure 3.3 Steps to classify snatch phases using machine learning classifiers

Figure 3.3 illustrated the steps to classify snatch phases using SVM and ANN classifiers. Initially, images were extracted from weightlifting videos, and features such as pose landmarks and points on the barbell were detected and computed. These features were fed into the SVM and ANN classifiers. The outputs from these classifiers were used to determine the sequence of snatch phases, which enabled a detailed analysis of the weightlifting process.

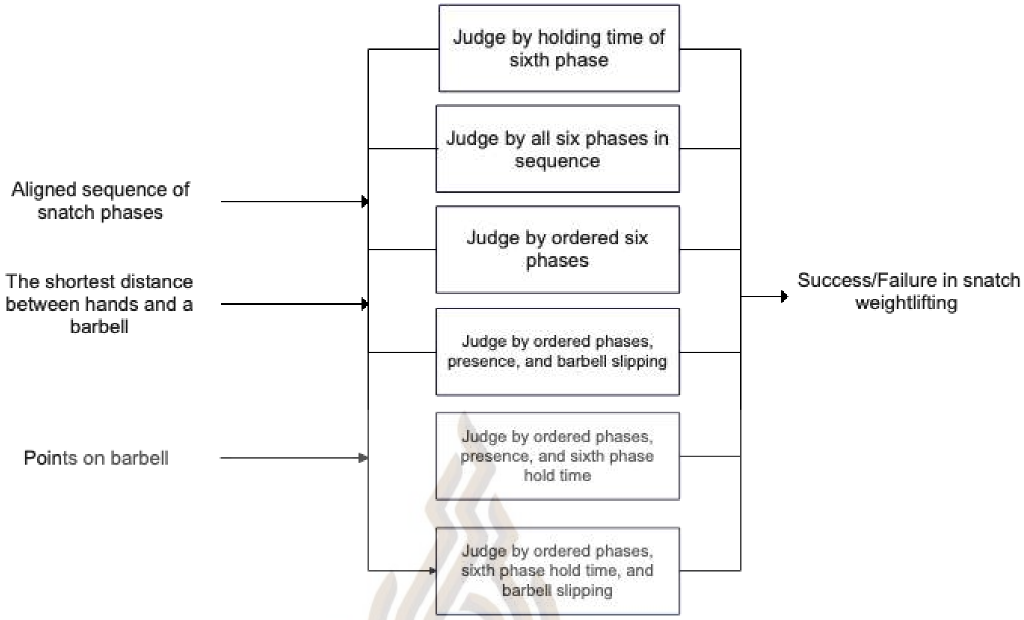


Figure 3.4 Steps to determine snatch weightlifting success

Figure 3.4 showed how to use video analysis to determine the success of a snatch weightlifting attempt. The aligned sequence of snatch phases, the shortest distance between hands and a barbell were used as features to determine snatch weightlifting success. The process involved verifying the holding of the sixth phase for at least one second, ensuring the completion of all six phases in order, and ensuring proper barbell handling.Additional checks included confirming the sixth phase's holding time and checking a barbell's slip. Each criterion helped assess whether the lift was successful.

3.2 Study Design

The research adopted a comprehensive methodology, incorporating advanced tools and techniques for an in-depth analysis of weightlifting performances. The central focus involved the intricate process of classifying weightlifting behaviors, estimating datasets for image and feature data models, customizing the YOLO model training process, and designing the expansion of feature values. Additionally, the study encompassed the design of weightlifting success/failure determination functionalities.

Due to the complexity of the research process, the study was divided into two main parts: 1) Images and features classification snatch weightlifting: This part of the research focuses on classifying weightlifting images and features using various methods. The aim was to explore the feasibility of feature classification and provide reliable results for distinguishing different phases of the snatch weightlifting. 2) Determining the success in snatch weightlifting: Building on the feature classification, this part involves researching the factors that determine the success or failure of snatch weightlifting attempts. The objective is to develop a robust model that can accurately classify and predict successful and failed lifts based on extracted features.

3.2.1 Classification of Images and Features in Snatch Weightlifting

This part of the research focuses on collecting videos from various snatch weightlifting and extracting images from these videos. The images were then classified into six phases of the snatch lift using two different approaches. The first approach involved training CNN models, such as MobileNet and VGG16, for image classification. The second approach utilized skeletal recognition and object detection to extract feature data, which was then used to train SVM and ANN models for feature classification. This will be base for research of determining the success in snatch weightlifting.

3.2.2 Determination of the Success in Snatch Weightlifting

Extract images from video: This step involved capturing individual frames from a video recording of snatch weightlifting performances. These frames served as the basis for further analysis, allowing researchers to extract key features and information from each frame to understand the dynamics of the weightlifting movement.

Extract pose landmark features: Pose landmark features refered to specific points or landmarks on the athlete's body that were indicative of their posture and movement during the snatch weightlifting competition. These features were extracted using tools MediaPipe, which could detect and track key points like joints and limbs throughout the video frames.

Detect a barbell: In this step, the presence of a barbell in each frame was detected using object detection techniques. YOLO was employed for this purpose, allowing for accurate identification and localization of the barbell within the frame.

Extract points on the detected barbell: Once the barbell was detected, specific points on the barbell were extracted. These points included the positions of the barbell 2 ending sides relevant landmarks that provided additional information about the weightlifting movement.

Classify snatch phases using a machine learning classifier: The extracted pose landmark features and barbell features were combined and used as input to a machine learning classifier. This classifier was trained to recognize and classify different phases of the snatch weightlifting movement based on the features extracted from each frame. The classifier assigned a phase class to each frame, indicating the phase of the lift that the athlete is currently in.

Phase correction: After the initial classification of snatch weightlift phases, a phase alignment process was employed to correct or refine the sequence of classified phases. This part involved applying rules or algorithms to ensure that the sequence of phases was consistent and accurately reflected the progression of the weightlifting movement.

Compute distances between hands and points on the barbell: The distances between the athlete's hands and specific points on the barbell were computed using the extracted pose landmark features and barbell features. These distances provided valuable information about the athlete's grip and positioning relative to the barbell throughout the lift.

Determine snatch weightlifting success: Determine snatch weightlifting success: The computed distances between hands and points on the barbell, along with the aligned phase sequence, were used to determine the success of the snatch weightlifting performance. This determination considered the correctness and completeness of the phase sequence, ensuring that all phases are present and in the correct order. Additionally, the analysis evaluated whether the duration of the weightlifting conforms to regulatory standards, ensuring that the weightlifting was performed within the designated time frame.

3.3 Instruments

Adobe Premiere: This video editing tool was essential for clipping and preparing complete weightlifting sequences from competition scenes, forming the basis of the research dataset.

LabelImg: Used as a marking tool for object recognition, it was employed to annotate barbell positions in training images. It was use for training the barbell detection model and ensuring accurate object recognition.

MediaPipe: This framework was utilized to extract skeletal features from videos, providing a detailed analysis of athletes' movements during the weightlifting process.

YOLO: As an object detection tool, YOLO identified and located the barbell in the images. This data, combined with skeletal recognition, allowed for accurate calculation of distances between the athlete and the barbell, which was essential for performance assessment.

Python: Python was chosen as a programming language for integrating various components of the research, including object detection and skeletal recognition frameworks. Python was also used to write machine learning code necessary for model training and testing.

scikit-learn: This additional Python module was used in building the machine learning models required for classifying the different phases of snatch weightlifting.

3.4 Data Collection

The research involved two main aspects: images and features classification and determining the success in snatch weightlifting. To support these studies, three distinct datasets were used: 1) Image Classification Dataset: This dataset was based on 50 successful snatch weightlifting videos and the images extracted from these videos. It was used to train image classification models, employing CNNs such as MobileNet and VGG16, as well as SVM and ANN for feature classification. 2) Barbell Detection Dataset: Comprising 20 successful weightlifting videos and annotated barbell images, this dataset was utilized to train the YOLO model for accurate barbell detection in various weightlifting scenarios. 3) Success Determination Dataset: The dataset included 20 weightlifting videos (10 successful and 10 failed videos), as well as the images used for training models to classify weightlifting phases and detect barbells. It was used for the research and analysis focused on determining the success or failure of snatch weightlifting attempts, providing a balanced representation of both outcomes.

These datasets collectively enabled the detailed analysis and model training necessary for classifying weightlifting phases and assessing performance outcomes. The details were shown in Table 3.1.

Table 3.1 Dataset

No.	Dataset Name	Description	Number of Data
1	Image Training	Original data	50 videos
		For training	3644 images
		For testing	1331 images
2	Feature Extraction	Original data	50 videos
		For training	3644 images
		For testing	1331 images
3	Feature Training	Original data	50 videos
		For training	3644 images
		For testing	1331 images
4	Barbell Training	Original data	20 videos
		For training	2078 images
		For testing	4975 images
5	Success Determination	Successful videos	10 videos, 1228 images
		Failed videos	10 videos, 1265 images

3.4.1 Image Training and Feature Extraction

The initial phase of data collection involved extracting relevant information from front-view videos of various weightlifting competitions. A total of 50 successful snatch weightlifting videos were selected for analysis. The videos were subjected to preprocessing steps, including resizing all frames to a standardized resolution of 224x224 pixels. This standardization ensured consistency in the dataset, a crucial factor for subsequent model training and extraction.

3.4.2 Feature Training

The dataset comprised 3,644 images extracted from the training videos for training model, with a balanced distribution across the six predefined phases. Additionally, a separate dataset for testing purposes was created, consisting of 1,331 images from 16 successful snatch weightlifting videos.

3.4.3 Barbell Training

The YOLO model for barbell detection was trained and optimized using a specially created dataset. The 2,078 images in the dataset were taken from various weightlifting frames extracted from videos. Each image was carefully annotated to highlight the precise position of the barbell.

This specialized dataset played a pivotal role in enhancing the YOLO model's proficiency in recognizing barbells within the weightlifting context. By training on this dedicated dataset, the model gained the capability to accurately detect and locate barbells in real-world weightlifting scenarios.

3.4.4 Success Determination for Weightlifting Sequence Videos

In preparation for weightlifting sequence analysis, two subsets of videos were meticulously selected: 1) Successful videos subset: Comprising 10 videos, this subset contributed a total of 1228 images for analysis. These videos captured successful executions of weightlifting movements, providing insights into well-performed actions. 2) Failed videos subset: Consisting of 10 videos capturing unsuccessful attempts, this subset contributed a total of 1265 images. The inclusion of failed attempts offered a comprehensive understanding of the challenges and difference in weightlifting motions.

The combination of these subsets formed a robust dataset for weightlifting sequence analysis, ensuring a balanced representation of both successful and unsuccessful scenarios. This diversity was essential for a nuanced exploration of temporal dynamics and accurate classification during subsequent analysis.

3.5 Data Analysis

3.5.1 Algorithm Parameters for Weightlifting Phase Classification

This section first introduced the parameter configuration of the algorithm model for snatch phase classification, then it expanded on the study of determining whether weightlifting is successful or unsuccessful. The classification of the snatch phase is a crucial basis for determining the success or failure of weightlifting.

Including: 1) SVM parameters: Used a linear kernel function for its classification. 2) ANN architecture: The ANN consisted of an input layer, a hidden layer with 128 neurons and ReLU activation, and an output layer with 6 neurons and softmax activation. 3) CNN Architecture: The CNNs comprised of 3 convolution layers. Each layer used 32 filters to extract image features. The flatten layer transformed the features into a one-dimensional vector passed through the ANN. The ANN, which was a part of the CNN, consisted of 64 hidden nodes and 6 output nodes. 4) ResNet50 Architecture: ResNet-50 was utilized. The top section of the ResNet-50 featured a global average pooling layer, a dense layer with 64 hidden nodes, a 0.2 dropout layer, a dense layer with 64 hidden nodes and a ReLU activation function, and a 0.2 dropout layer. 5) MobileNet Architecture: MobileNetV3 was implemented. The top section of the MobileNetV3 featured a global average pooling layer, a dense layer with 64 hidden nodes, a 0.2 dropout layer, a dense layer with 64 hidden nodes and a ReLU activation function, and a 0.2 dropout layer.

Based on the above models, weightlifting behavior classification was achieved. The next part of the research focused on classifying weightlifting video sequences to identify successful and failed lifts.

3.5.2 Extract Images from a Video

Images were extracted using 15 Olympic snatch weightlifting videos. Then they were resized to 224 x 244 pixels. The resized images were used as a source to extract MediaPipe features.

3.5.3 Extract Pose Landmark Features

In the feature extraction, MediaPipe was used to extract pose landmark features from images (Kukil, 2021). The features and their descrptions were shown in Table 3.2.

Table 3.2 Features obtained from MediaPipe and their descriptions

No.	MediaPipe	Descriptions
1	NOP_X	Nose Positions Value in X Axis
2	NOP_Y	Nose Positions Value in Y Axis
3	LEIP_X	Left Eye Inner Positions Value in X Axis
4	LEIP_Y	Left Eye Inner Positions Value in Y Axis
5	LEP_X	Left Eye Positions Value in X Axis
6	LEP_Y	Left Eye Positions Value in Y Axis
7	LEOP_X	Left Eye Outer Positions Value in X Axis
8	LEOP_Y	Left Eye Outer Positions Value in Y Axis
9	REIP_X	Right Eye Inner Positions Value in X Axis
10	REIP_Y	Right Eye Inner Positions Value in Y Axis
11	REP_X	Right Eye Positions Value in X Axis
12	REP_Y	Right Eye Positions Value in Y Axis
13	REOP_X	Right Eye Outer Positions Value in X Axis
14	REOP_Y	Right Eye Outer Positions Value in Y Axis
15	LEARP_X	Left Ear Positions Value in X Axis
16	LEARP_Y	Left Ear Positions Value in Y Axis
17	REARP_X	Right Ear Positions Value in X Axis
18	REARP_Y	Right Ear Positions Value in Y Axis
19	MLP_X	Mouth Left Positions Value in X Axis
20	MLP_Y	Mouth Left Positions Value in Y Axis
21	MRP_X	Mouth Right Positions Value in X Axis
22	MRP_Y	Mouth Right Positions Value in Y Axis
23	LSP_X	Left Shoulder Positions Value in X Axis
24	LSP_Y	Left Shoulder Positions Value in Y Axis
25	RSP_X	Right Shoulder Positions Value in X Axis
26	RSP_Y	Right Shoulder Positions Value in Y Axis
27	LEP_X	Left Elbow Positions Value in X Axis
28	LEP_Y	Left Elbow Positions Value in Y Axis
29	REP_X	Right Elbow Positions Value in X Axis
30	REP_Y	Right Elbow Positions Value in Y Axis
31	LWP_X	Left Wrist Positions Value in X Axis
32	LWP_Y	Left Wrist Positions Value in Y Axis
33	RWP_X	Right Wrist Positions Value in X Axis
34	RWP_Y	Right Wrist Positions Value in Y Axis
35	LPP_X	Left Pinky Positions Value in X Axis
36	LPP_Y	Left Pinky Positions Value in Y Axis
37	RPP_X	Right Pinky Positions Value in X Axis
38	RPP_Y	Right Pinky Positions Value in Y Axis
39	LIP_X	Left Index Positions Value in X Axis
40	LIP_Y	Left Index Positions Value in Y Axis
41	RIP_X	Right Index Positions Value in X Axis

Table 3.2 Features obtained from MediaPipe and their descriptions (cont.)

No.	MediaPipe	Descriptions
42	RIP_Y	Right Index Positions Value in Y Axis
43	LTP_X	Left Thumb Positions Value in X Axis
44	LTP_Y	Left Thumb Positions Value in Y Axis
45	RTP_X	Right Thumb Positions Value in X Axis
46	RTP_Y	Right Thumb Positions Value in Y Axis
47	LHP_X	Left Hip Positions Value in X Axis
48	LHP_Y	Left Hip Positions Value in Y Axis
49	RHP_X	Right Hip Positions Value in X Axis
50	RHP_Y	Right Hip Positions Value in Y Axis
51	LKP_X	Left Knee Positions Value in X Axis
52	LKP_Y	Left Knee Positions Value in Y Axis
53	RKP_X	Right Knee Positions Value in X Axis
54	RKP_Y	Right Knee Positions Value in Y Axis
55	LAP_X	Left Ankle Positions Value in X Axis
56	LAP_Y	Left Ankle Positions Value in Y Axis
57	RAP_X	Right Ankle Positions Value in X Axis
58	RAP_Y	Right Ankle Positions Value in Y Axis
59	LHP_X	Left Heel Positions Value in X Axis
60	LHP_Y	Left Heel Positions Value in Y Axis
61	RHP_X	Right Heel Positions Value in X Axis
62	RHP_Y	Right Heel Positions Value in Y Axis
63	LFIP_X	Left Foot Index Positions Value in X Axis
64	LFIP_Y	Left Foot Index Positions Value in Y Axis
65	RFIP_X	Right Foot Index Positions Value in X Axis
66	RFIP_Y	Right Foot Index Positions Value in Y Axis

3.5.4 Detect a Barbell

The Yolo 7.0 algorithm was utilized to identify and locate a barbell. The outcome consisted of four (x,y) coordinates representing the corners of a rectangle that encloses the identified barbell (Kathuria, 2022; Chernytska, 2022). Figure 3.5 demonstrates an instance of barbell detection using Yolo.

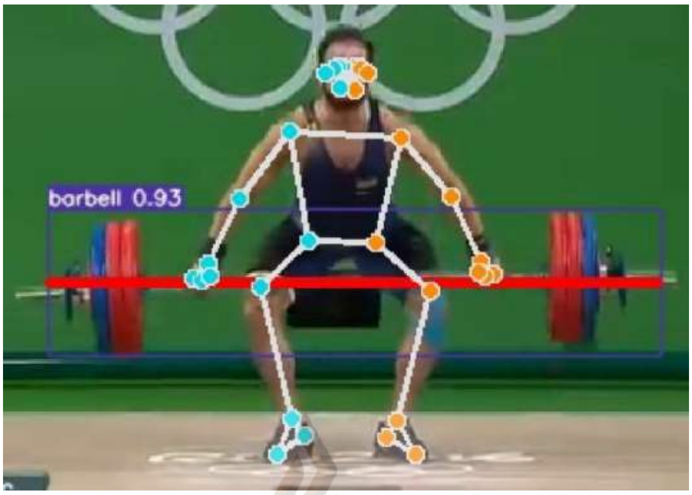


Figure 3.5 Detected barbell

3.5.5 Extract Points on the Detected Barbell

After acquiring the container that outlines the perimeter of a barbell. Two points are derived from the four coordinates (x,y) of the box.

The YOLO detector outputed a bounding box that consisted of four absolute position coordinates. These coordinates reflected the positions of the top-left and bottom-right corners of the bounding box. Two points were derived from the four coordinates (x,y) of the box. In order to determine the midpoints of the left and right sides of a "barbell" shaped bounding box, it began by identifying the corners of the bounding box. Next, calculated the midpoint on the left side by taking the average of the coordinates of the two left corners. Similarly, calculated the midpoint on the right side by averaging the coordinates of the two right corners by using the following algorithm.

The steps include: 1) Extract the identified items from the YOLO detection data and locate the specific object labeled as "barbell". 2) Retrieve the precise positional coordinates of the four corners of the bounding box that encompasses the "barbell" object. 3) Determine the coordinates that lie exactly in the middle between the left and right coordinates of the bounding box of the "barbell".

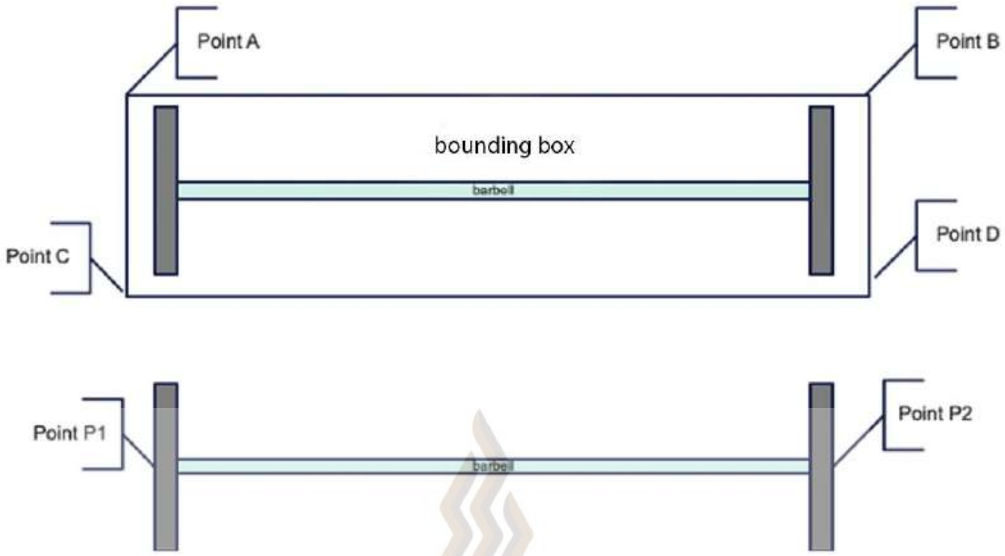


Figure 3.6 Compute P1 and P2 points from a barbell box

The midpoints between the left and right coordinates of the "barbell" using the following method. Given: The points A(Xa,Ya), B(Xb,Yb), C(Xc,Yc) and D(Xd,Yd) were absolute position coordinates of the four corners of the bounding box surrounding the "barbell" object. The midpoints P1 and P2 between the left and right coordinates of the "barbell" were P1(|Xc-Xa|/2,|Yc-Ya|/2) and P2(|Xd-Xb|/2,|Yd-Yb|/2).

The barbell features consisting of the left and right coordinates of the midpoints P1 and P2 (LBX, LBY, RBX and RBY) were described in Table 3.3.

Table 3.3 Barbell features

No.	Ballbell features	Descriptions
1.	LBX	Left Barbell x axis value
2.	LBY	Left Barbell y axis value
3.	RBX	Right Barbell x axis value
4.	RBY	Right Barbell y axis value

3.5.6 Classify Snatch Phases Using a Machine Learning Classifier

The 70 features (33*2+2*2 features), which consisted of x axis and y axis of 33 skeleton landmarks and 2 barbell position points, were classified using SVM and ANN into six phases(Makdoun, 2022).

3.5.7 Correct Phases

The system classified a sequence of images extracted from a video into six phases. However, in fact, some errors were found in the phase classification results. Therefore, the rules shown in Table 3.4 were applied to correct a sequence of classified phases. According to the rules, the five consecutive classified phases were used as information to determine the probable phases.

Table 3.4 Rules applied to correct sequences of phases

	Same
	Different
	Fixed

No.	Classified phases					Corrected phases				
	Frm t	Frm t+1	Frm t+2	Frm t+3	Frm t+4	Frm t	Frm t+1	Frm t+2	Frm t+3	Frm t+4
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										

Table 3.4 Rules applied to correct sequences of phases (cont.)

No.	Classified phases					Corrected phases				
	Frm t	Frm t+1	Frm t+2	Frm t+3	Frm t+4	Frm t	Frm t+1	Frm t+2	Frm t+3	Frm t+4
27										
28										
29										
30										
31										
32										

Remark: Frm t was the phase classification result of the frame or image at time t.

The algorithm to correct phases was explained as follows:

Given

P : A sequence of classified weightlifting phases (p_0, p_1, p_2, … p_T)

T: Last time that the phase was classified

i: Classified weightlifting phase number

for i=0 to T-4

Apply the rules to correct the classified weightlifting phases p_i to p_{i+4}

The phase correction process aimed to resolve classification errors within the weightlifting video sequences. After correction, the corrected phase sequence was used to determine success in weightlifting.

3.5.8 Compute distances between hands and points on a barbell

The system also used the distances between hands and points on a barbell to determine whether a lifting was successful or not. The method for calculate the distances between hands and points on a barbell is shown in Figure 3.7.

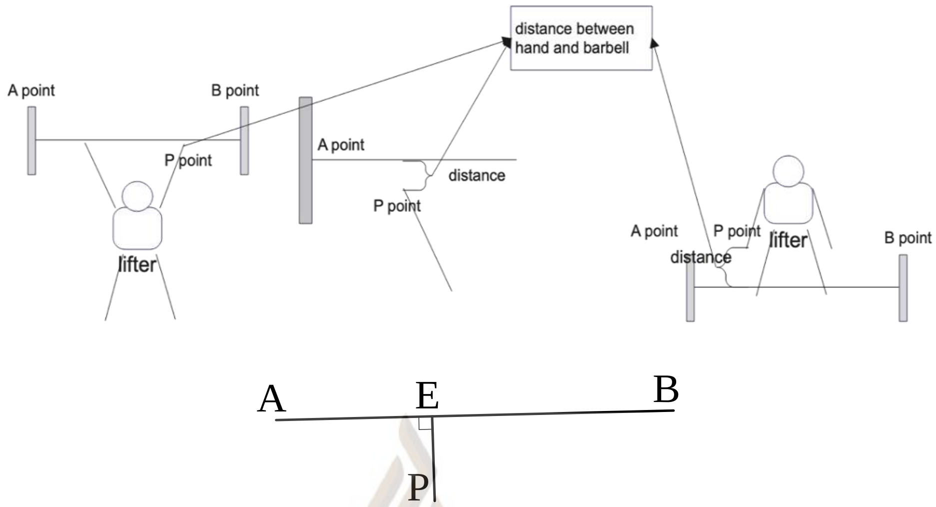


Figure 3.7 Compute the distance between a hand and a barbell

The distances between hands and points on a barbell were computed using the following steps:

Find a linear equation: ax+by+c = 0 given two points, A(x1, y1) and B(x2, y2) by using the following equation.

$$(y_1 - y_2)x + (x_2 - x_1)y + (x_1y_2 - x_2y_1) = 0 \qquad (3\text{-}1)$$

Find a distance PE: The distance between the line ax+by+c = 0 and the point PE (x_3, y_3), where a, b and c are real numbers and both a and b cannot be zero, can be calculated using the following equation.

$$PE = \frac{|ax_3+by_3+c|}{\sqrt{a^2+b^2}} \qquad (3\text{-}2)$$

Calculate a distance AB using the following equation.

$$AB = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (3\text{-}3)$$

Calculate a relative distance $PE_{relative}$

$$PE_{relative} = \frac{PE}{AB} \qquad (3\text{-}4)$$

The $PE_{relative}$ values for both left and right hands were calculated and used as features (DIS_LEFT, DIS_RIGHT) to represent the distances between barbell and hands, as described in Table 3.5.

Table 3.5 Extended features

No.	Barbell features	Descriptions
1.	DIS_LEFT	Distance between barbell and left hand
2.	DIS_RIGHT	Distance between barbell and right hand

The distance between the hands and a barbell (DHB) was computed using the following equation.

$$DHB = \max(DIS\_LEFT, DIS\_RIGHT) \quad (3\text{-}5)$$

The value of DHB used to judge whether the barbell slip from hands.

3.5.9 Determination of Snatch Weightlifting Success

Several comprehensive approaches to determining weightlifting success are presented here. The evaluation process included several criteria, including: the duration of the hold in the final phase, the completeness and order of the six phases, and the presence of any barbell slippage. Each criterion is systematically analyzed to determine weightlifting success. Figure 3.8 showed the classification results that contained all six weightlifting phases while in Figure 3.9, the classification results contained only three weightlifting phases. Phase classification was applied because successful lifting must include all six phases.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	2	2	2	2	2	2	2	3	3	3	3	4	4
4	4	4	4	4	4	4	4	4	4	4	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5								

Figure 3.8 Show a result contained all six weightlifting phases

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2									

Figure 3.9 Show a result contained only three weightlifting phases

3.5.9.1 Judge based on the holding period of the sixth phase

The algorithm for calculating the holding time of the sixth phase and checking whether the holding time was more than **threshold_hold_time** was explained as follows:

Given the phase sequence shown in Figure 3.10 which contained 45 frames of the sixth phase and the parameters as shown in Table 3.6.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	2	2	2	2	2	2	2	3	3	3	3	4	4
4	4	4	4	4	4	4	4	4	4	4	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5								

Figure 3.10 A phase sequence

Table 3.6 Parameters for calculating the holding time of the sixth phase

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.10)
2	num_6th_phase	The number of 6th phase frames (e.g. 45 in Figure 3.10)
3	threshold_hold_time (in seconds)	Threshold of holding time for the sixth phase. (e.g. 1 second)

Steps:

Count the number of frames in the sixth phase (phase number 5) in phase_seq_array, then divide it by a frame rate-- to get the holding time, T_6th_phase.

$$\text{T\_6th\_phase} = \frac{num_{\_6th\_phase}}{rame\_rate} \tag{3-6}$$

Given frame_rate equaled 30 frames/second, based the phase sequence in Figure3.7, T_6th_phase can be calculates as:

$$\text{T\_6th\_phase} = \frac{45}{30} = 1.5 \text{ seconds} \tag{3-7}$$

Check if T_6th_phase is less than threshold_hold_time seconds; if so, return False, otherwise return True.

```
if(T_6th_phase<threshold_hold_time)
        return False
else:
        return True                                        (3-8)
```

Because T_6th_phase was greater than 1, the method returns True and passed checking. The T_6th_phase was determined exclusively from the number of sixth phases, as this method did not take into account the distance between the hands and a barbell.

3.5.9.2 Judge Based on the Presence of all six phases in a Phase Sequence

This part explained the algorithm for counting the number of frames for each phase and judging whether each phase contains more than two frames. A sequence in which a phase spanned fewer than two frames was judged to be a missing phase sequence.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2									

Figure 3.11 Missing phase in the phases sequence (marked in red)

Given the phases in a phase sequence and the parameters for judging whether each phase contains more than two frames, as shown in Figure 3.11 and Table 3.7.

Table 3.7 Parameters for counting the number of frames for each phase

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.11)
2	threshold_times_phase	Threshold of the minimum appearance time for each phase in a video sequence (e.g. 2)

Steps:

Count the number of each phase in phase_seq_array

Check if any phase is shorter than threshold_times_phase

If there is every phase is shorter than threshold_times_phase, consider it as False (unsuccessful).

If every phase is longer than threshold_times_phase, consider it as True (successful).

So Figure 3.11 shows that the classification leaded to the absence of the fourth, fifth and sixth phases.

3.5.9.3 Judge From Six Phases in Order

The part explained the algorithm for judging whether a video sequence contains all phases (0~5) in ascending order. For a phase sequence as shown in Figure 3.12, the algorithm was described as follows:

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	2	2	2	3	3	5	5	5	5	3	3	1	1	1
2	3	5	3	3	3	4	3	4	3	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figure 3.12 Phases in a phase sequence (Disordered parts were marked red)

As shown un Table 9, Parameters for judging whether a video sequence includes all phases (0~5) in the ascending order

Table 3.8 Parameters for checking the order of frames for each phase

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.12)

The steps to check whether a video sequence contains all phases (0~5) in the ascending order were described below. The method checked whether a lower value phase value was observed later in the array.

Steps:

Check whether the sequence in phase_seq_array included all phases from 0 to 5 in ascending order by scanning every element in the array using index (i) from 0 to the number of elements in the array – 1 (len(phase_seq_array) – 1), the check should complete once the fifth phase has been verified.

If any element does not follow the ascending order, return False.

If every element follows the ascending order, return True.

```
for i = 0 to len(phase_seq_array) – 1
  if (phase_seq_array[i] > phase_seq_array[i + 1]
        or (phase_seq_array[i+1]- phase_seq_array[i])>1)
              and phase_seq_array[i]<5:
        return False
return True
```

Thus, Figure 3.12 shows that the classify feature result does not follow correct order (not pass checking) and the sequence was determined as unsuccessful weightlifting.

3.5.9.4 Judge Based on the Ordered six Phases, Presence of all six phases and Barbell slipping

This method checked the presence of all six phases and the distance between hands and a barbell until the last maximum phases obtained from the phase classification. Figures 3.13–3.14 showed the last maximum phase was found at the 82nd frame. The distances, as shown in Figure 3.15, were checked from the first frame to the 82nd frame, and it was found that the distances were above a threshold, indicating that the barbell was out of control and fell.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	2	2	2	3	3	5	5	5	5	3	3	1	1	1
2	3	5	3	3	3	4	3	4	3	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figure 3.13 Phase fluctuation shown in a red area

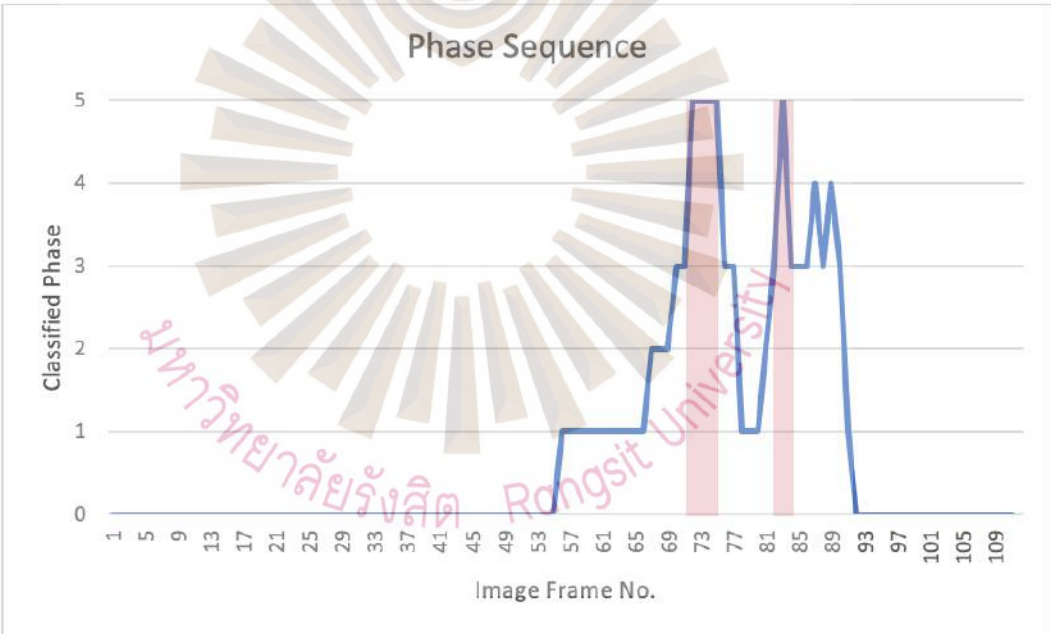


Figure 3.14 A curve in the red area of the graph shown the classified phase fluctuation

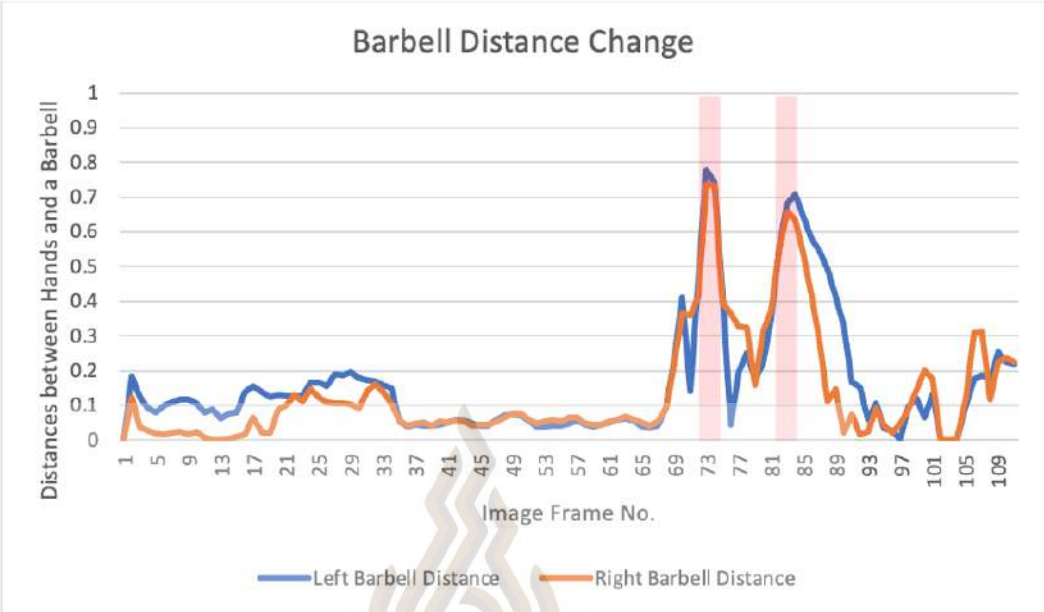


Figure 3.15 The fluctuation in the distances between hands and a barbell

Table 3.9 Parameters for judging the correct order and presence of all phases and determining whether a video sequence contains barbell slipping

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.13)
2	dis_seq_array_left	A sequence of distances between the left hand and a barbell (e.g. one side hand distance sequence in Figure 3.15)
3	dis_seq_array_right	A sequence of distances between the right hand and a barbell (e.g. one side hand distance sequence in Figure 3.15)
4	threshold_times_phase	A minimum appearance time threshold for every phase in a video sequence (e.g. 2)
5	num_cont_frms	The number of consecutive frames that a hand is left off a barbell (e.g. 3)
6	threshold_dis_val	The threshold for determining whether a hand is left off a barbell (e.g. 0.3)

The steps for checking that all phases are present and in the correct order (see subsections 3.5.9.2 and 3.5.9.3), and whether a hand is left off a barbell.

Steps:

Judge whether each phase contains more than 2 frames from an array, phase_seq_array (see subsection 3.5.9.2).

Check whether a video sequence contains all phases in the correct order using an array, phase_seq_array (see subsection 3.5.9.3).

From dis_seq_array_left, find the number of consecutive frames that a hand stays away from a barbell using the distances between the left hand and a barbell. The frame in which the distance between the right hand and a barbell is greater than threshold_dis_val is counted as a frames in which a hand is left off a barbell.

Use the same method to check the slipping of a barbell from the right hand.

Find the last maximum phase position or frame.

Check the distances between hands and a barbell from the first frame to the last maximum phase position or frame. If a distance was detected as greater than a threshold, a barbell is considered to have fallen or slipped.

3.5.9.5 Judge Based on the Ordered six Phases, Presence of all Phases and the Holding Time for the sixth Phase

In this part, the algorithm not only checks for the correct order and presence of all phases (see subsections 3.5.9.2 and 3.5.9.3), but also calculate the holding time of the sixth phase and check whether the holding time was more than threshold_hold_time second (see subsections 3.5.9.1).

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	2	2	2	2	2	2	2	3	3
3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5												

Figure 3.16 Data shown all phases in the correct order and sufficient holding time for the sixth phase

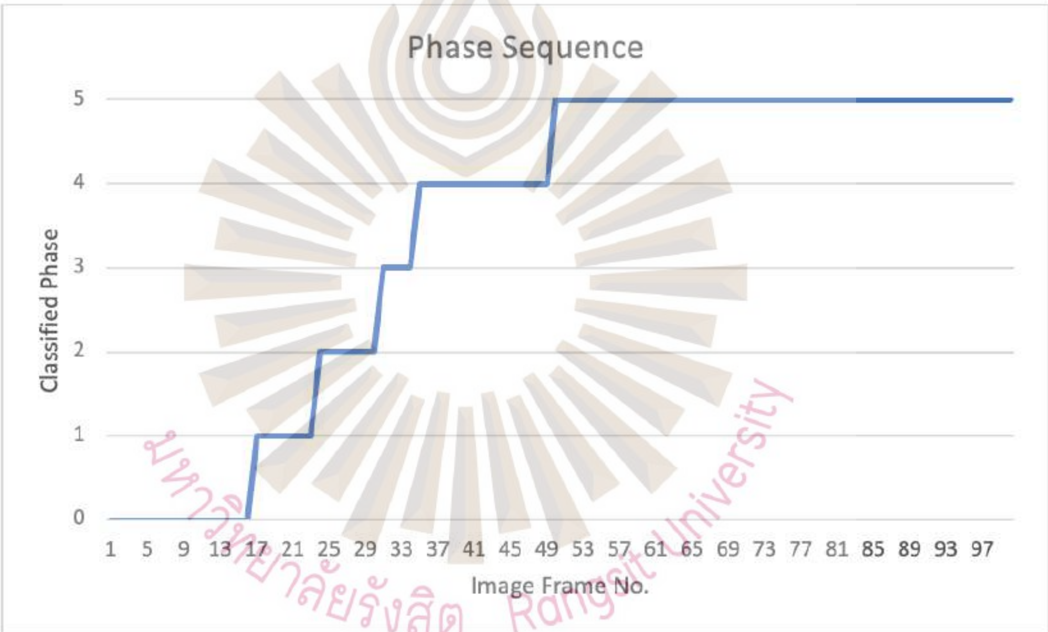


Figure 3.17 All phases in the correct order with sufficient holding time for the sixth phase in a sequence

Table 3.10 Parameters for judging the correct order and presence of all phases and for checking whether the holding time of the sixth phase was at least 1 second

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.16)
2	threshold_times_phase	A minimum appearance time threshold for every phase in a video sequence (e.g. 2)

Table 3.10 Parameters for judging the correct order and presence of all phases and for checking whether the holding time of the sixth phase was at least 1 second (cont.)

No.	Name	Description
3	num_6th_phases	The number of frames classified as the 6th phase (e.g. 45 in Figure 3.10)
4	threshold_hold_time	Threshold of holding time for the sixth phase. (e.g. 1 second)

The steps to verify that all classified phases were and appeared in the correct order, as well as the holding time of the sixth phase, were at least "threshold_hold_time" seconds (see subsection 3.5.9.1).

Steps:

Judge whether each phase contains more than 2 frames from an array, phase_seq_array (see subsection 3.5.9.2).

Check whether a video sequence contains all phases in the correct order using an array, phase_seq_array (see subsection 3.5.9.3).

Use the phase_seq_array array to assess whether the calculated sixth phase lasts longer than a threshold_hold_time (see subsection 3.5.9.1).

Judge whether the computed sixth phase holding time more than a time threshold in seconds (threshold_hold_time) using an array, phase_seq_array (see subsection 3.5.9.1).

If all the above conditions were met, it is considered as a successful lifting. Otherwise, it is considered as an unsuccessful lifting.

Thus, Figures 3.17-3.18 show that the results of the classified features follow the correct order and contain all phases. The holding time of the sixth phase of more than one second indicated that the sequence was a successful weightlifting attempt.

3.5.9.6 Judge Based on the Ordered six Phases, the Holding Time for the sixth Phase and the Barbell Slipping

In this part, the algorithm not only checks for the correct order and presence of all phases (subsections 3.5.9.2, 3.5.9.3, and 3.5.9.1), and verifies whether the barbell was held for longer than threshold_hold_time seconds. It also makes sure the weight does not slip during the sixth phase of the phase sequence. This method included the distance between the hands and a barbell. The sixth phase holding time in this method was different from the first method, in which the holding time was calculated from only a phase sequence. This method calculated the time from the first sixth-phase frame to the last sixth-phase frame in which the distance between hands and a barbell was not greater than a threshold.

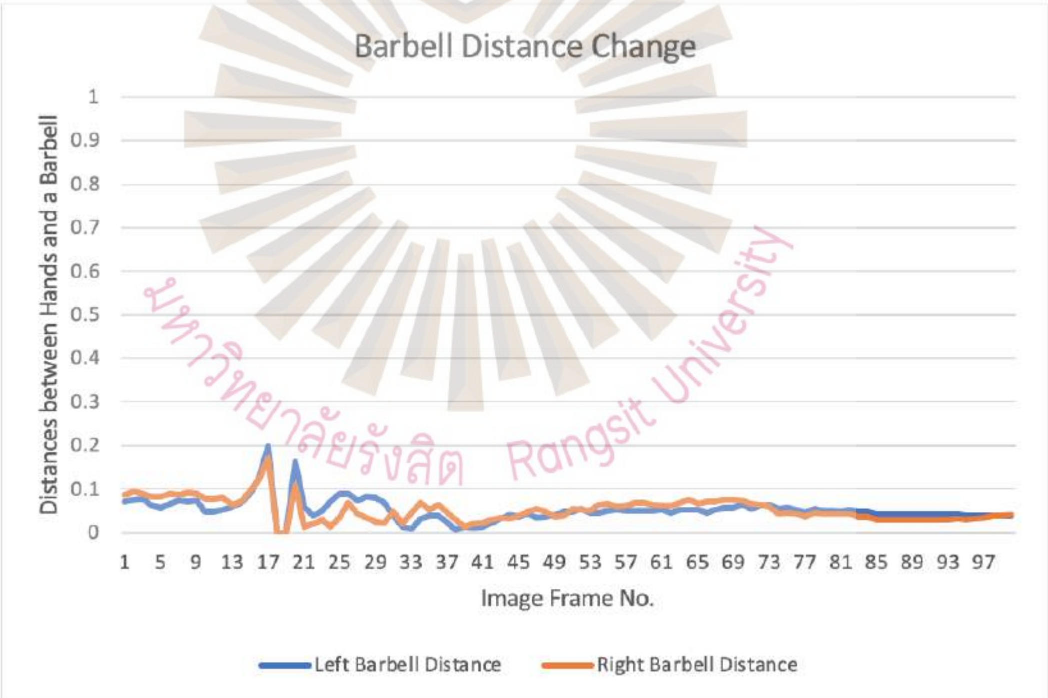


Figure 3.18 The fluctuation in the distances between hands and a barbell

Table 3.11 Parameters for judging the correct order and presence of all phases, verifying whether the barbell was held for longer than threshold_hold_time seconds, and ensuring that the barbell does not slip during the sixth phase of the phase sequence

No.	Name	Description
1	phase_seq_array	An array representing a classified phase sequence of a video (e.g. a phase sequence shown in Figure 3.16)
2	dis_seq_array_left	A sequence of distances between the left hand and a barbell (e.g. one side hand distance sequence in Figure 3.18)
3	dis_seq_array_right	A sequence of distances between the right hand and a barbell (e.g. one side hand distance sequence in Figure 3.18)
4	threshold_times_phase	A minimum appearance time threshold for every phase in a video sequence (e.g. 2)
5	num_cont_frms	The number of consecutive frames that a hand is left off a barbell (e.g. 3)
6	threshold_dis_val	The threshold for determining whether a hand is left off a barbell (e.g. 0.3)
7	num_6th_phases	The number of frames classified as the sixth phase (e.g. 45 in Figure 3.11)
8	threshold_hold_time	Threshold of holding time for the sixth phase. (e.g. 1 second)

The steps to check the correct order and presence of all phases, and verify whether the barbell was held for longer than threshold_hold_time seconds (subsections 3.5.9.2, 3.5.9.3, and 3.5.9.1). Step 4 ensures that the barbell does not slip during the sixth phase of the phase sequence.

Steps:

Judge whether each phase contains more than 2 frames from an array, phase_seq_array (see subsection 3.5.9.2).

Check whether a video sequence contains all phases in the correct order using an array, phase_seq_array (see subsection 3.5.9.3).

Use phase_seq_array array and the distance between the hands and a barbell to calculate the time from the first sixth-phase frame to the last sixth-phase frame in which the distance between hands and a barbell was not greater than a threshold.

Checks whether the calculated sixth phase lasted longer than a threshold_hold_time.

If all the above conditions were met, it is considered as a successful lifting. Otherwise, it is considered as an unsuccessful lifting.

Thus, Figures 3.17, 3.14 and 3.19 showed that the classified feature results follow the correct order and presence of all phases, the holding time of the sixth phase was more than 1 second, and the barbell does not slip during the sixth phase in the phase sequence (passing parts of the checking items), determining the sequence as a successful weightlifting attempt.

3.5.10 Validation Sample Strategies Design for Determination Methods of Weightlifting

This section designed a test strategy for detecting the success or failure in snatch weightlifting from videos. It included 20 weightlifting attempt videos, comprising 10 successful and 10 failure scenarios. Specifically, the failure videos exhibit various error cases. The goal was to catalog all error scenarios occurring in real contests and rank them based on their likelihood in real-life situations. Weightlifting failure videos were divided into five categories, as shown in Table 3.12.

Table 3.12 Examples of weightlifting failure videos and checking methods used

Example No.	Description
Example 1	Six distinct checking methods were used to test it. Six methods successfully identified the failure.
Example 2	Six distinct checking methods were used to test it. Five methods successfully identified the failure, whereas only M_AllPhases failed to detect the failure.

Table 3.12 Examples of weightlifting failure videos and checking methods used (cont.)

Example No.	Description
Example 3	Six distinct checking methods were used to test it. Three methods successfully identified the failure, whereas M_AllPhases, M_OrderPh, and M_AllPhSlip failed to detect the failure.
Example 4	Six distinct checking methods were used to test it. Three methods successfully identified the failure, whereas M_AllPhases, M_OrderPh, and M_AllPhSlip failed to detect the failure.
Example 5	Six distinct checking methods were used to test it. Two methods successfully identified the failure, whereas M_SixHold, M_AllPhases, M_OrderPh and M_OrderHold failed to detect the failure.

Table 3.13 The samples video of weightlifting attempts

Video Sample	Description
FWS1~FWS4	This sample video was an Example 1 type video, showcasing a common weightlifting mistake where failure occurred due to the omission of important phases. There were four test samples within this category, representing the most common error scenario.
FWS5~FWS7	This sample video was the Example 2 type video, representing the second most common error scenario. After completing all essential phases, the lifter experienced a failure or chose to abandon the last phase. This category included three test samples that allow for the validation of all methods, except for the M_AllPhases method, which identifies the absence of some phases.
FWS8	This sample video was the Example 3 type video, representing a rare occurrence of a mistake with only one test sample. The sample was utilized to assess the efficacy of method M_AllPhSlip in detecting slippage of a barbell.
FWS9	This sample video was an Example 4 video type, which was a rare error situation used to evaluate the performance of the last phase holding time method-based techniques M_SixHold and M_OrderHold.
FWS10	This sample video was an Example 5 video type, demonstrating another rare error situation. This video evaluates the performance of two techniques, M_SixHold and M_OrderHold, which are based on the last phase holding time method.
SWS1~SWS3	These videos showed the athlete at a greater size, which made it easier to see their facial expressions and movements.
SWS4~SWS10	These videos showed the athlete at a smaller size and focused on the overall continuity of the athlete's movements.
SWS1~SWS4	Weightlifting competition at the Rio Olympics, with a green background color in the weightlifting venue
SWS5~SWS7	Weightlifting competition at the Tokyo Olympics, with a red background color in the weightlifting venue

Table 3.13 The samples video of weightlifting attempts (cont.)

Video Sample	Description
SWS8~SWS10	Weightlifting competition at the Beijing Olympics, with a blue background color in the weightlifting venue
SWS1~SWS7	Men's heavyweight weightlifting competition, with many weights loaded and a barbell length of 2.2 metres
SWS8~SWS10	Women's lightweight weightlifting competition, with fewer weights loaded and a barbell length of 2.15 metres

Remark: FWS means "failure weightlifting sample," and SWS means "successful weightlifting sample."

Chapter 4

Results and Discussion

The results were divided into 5 parts: 1) Weightlifting phase classification. 2) Weightlifting phase correction. 3) Calculating distances between hands and a barbell. 4) Determining success or failure in weightlifting sequences. 5) Determine snatch weightlifting success

4.1 Weightlifting Phase Classification

Weightlifting phase classification is an important part of the research. The image classification methods consisting of CNN, MobileNet, and ResNet50 were applied to directly distinguish weightlifting phases from images. On the other hand, the features extracted by using MediaPipe were used with ANN and SVM classifiers to identify the weightlifting phases. The classification using images and the MidiaPipe features were then evaluated and compared. Table 4.1 showed that the classification using MidiaPipe features and SVM outperformed that using ANN. In addition, it provided higher accuracy than the classification from images using CNN, MobileNet and ResNet50.

Table 4.1 Weightlifting phase classification accuracy

Classifier	Accuracy using images	Accuracy using posture landmarks and barbell features
ANN	-	89.86%
SVM		91.96%
CNN	88.13%	-
MobileNet	88.58%	-
ResNet50	78.14%	-

Table 4.1 shows the accuracy of various classifiers in detecting weightlifting phases. When classifying weightlifting phases from images, MobileNet obtained an accuracy of 88.58%. CNN achieved 88.13%, and ResNet50 achieved 78.14%.

With the posture landmarks and barbell features extracted by MediaPipe, the SVM classification achieved 91.96% accuracy, which was higher than the ANN classifier's accuracy. The results suggested that utilising MediaPipe features in conjunction with SVM yielded higher accuracy compared to the other approaches examined.

Based on the findings, the subsequent part of the study focused on using phase classification results from MidiaPipe features and SVM to determine success in snatch weightlifting. Nonetheless, some errors in the phase classification were corrected after the phase classification. Therefore, we applied the procedure to correct any likely incorrect phases and presented the findings in the following section.

4.2 Weightlifting Phase Correction

The phase correction, as explained in section 3.5, is based on five neighboring classified phases. Figures 4.1-4.2 provided examples of weightlifting phases both before and after the correction. The correction of the weightlifting phases of an unsuccessful and successful weightlifting, respectively, was illustrated in Figures 4.1 and 4.2.

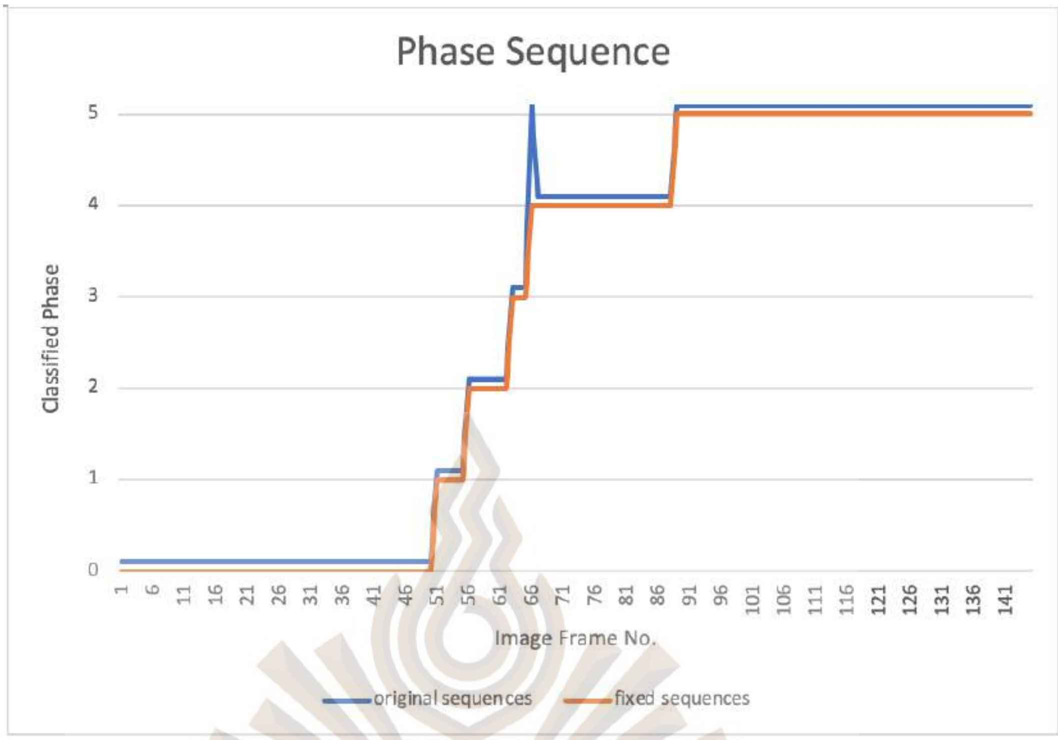


Figure 4.1 Phases derived from a successful weightlifting, both prior to and following fixing

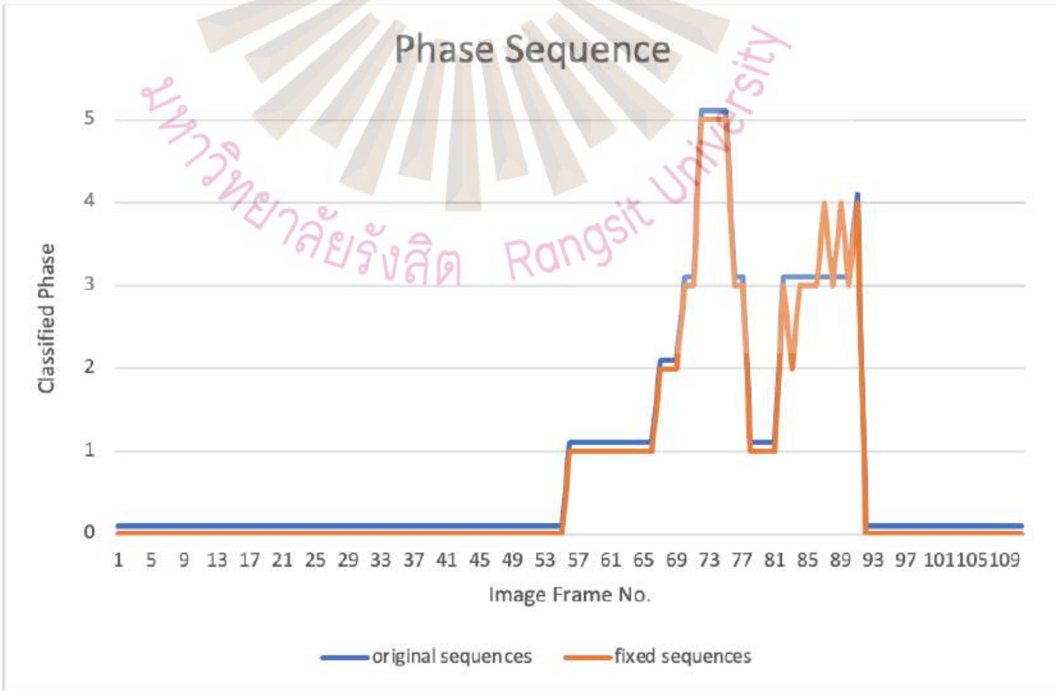


Figure 4.2 Phases derived from an unsuccessful weightlifting, both prior to and following fixing

4.3 Calculating Distances Between Hands and a Barbell

This section illustrated the distances between the hands and the barbell in both successful and unsuccessful weightlifting video sequences. Two curves represented the distances between left and right hands to a barbell, with the values expressed relative to the length of the barbell.

Figure 4.3 shows the distances between hands and a barbell for a video sequence considered successful, whereas Figure 4.4 illustrates the same measurement for a video sequence considered failed. Unsuccessful attempts led to substantial variations in the distance between hands and a barbell, occasionally surpassing 0.8. In contrast, the fluctuation in distance during successful attempts consistently stayed below 0.2 (In the real program, the threshold was fixed at 0.3 to avoid some fluctuation), indicating a more consistent grip on the barbell throughout the whole lifting operation.

These findings emphasized the significance of monitoring the distances between hands and a barbell as a potential indicator of barbell slippage during weightlifting. The graphs visually depicted the fluctuation in hand-barbell distances and aided in identifying ballbell slipping situations. The results demonstrated significant disparities in the fluctuation of hand-barbell distance between successful and unsuccessful weightlifting attempts, highlighting the crucial role of grip stability in the lifting procedure.

These findings were significant for determining success or failure in weightlifting sequences.

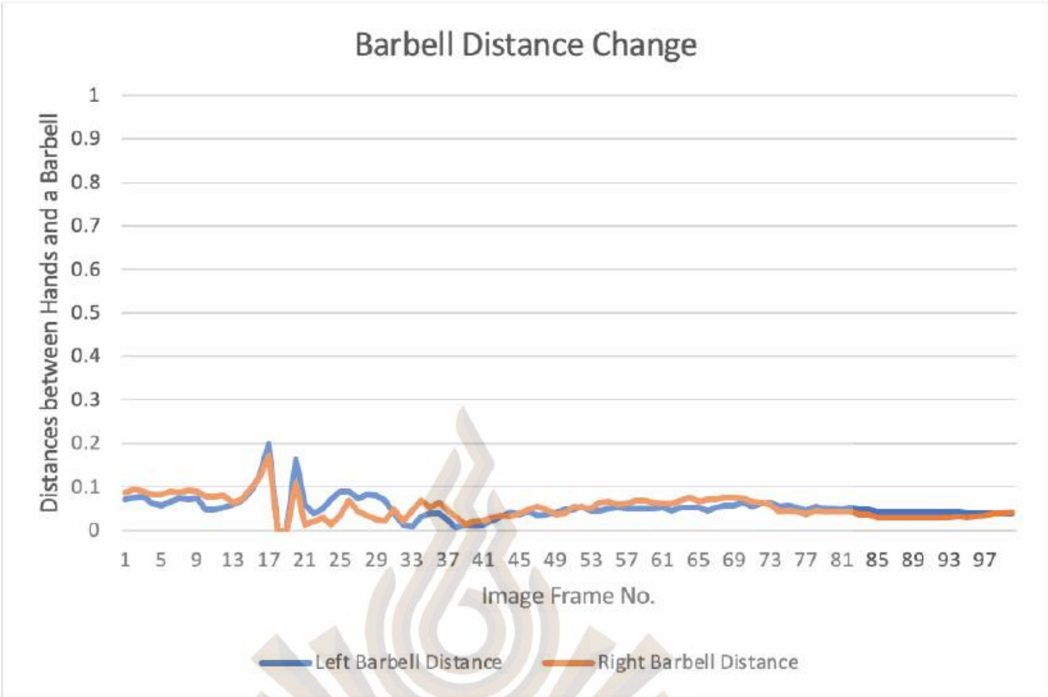


Figure 4.3 The fluctuation in the distances between hands and a barbell of a successful video

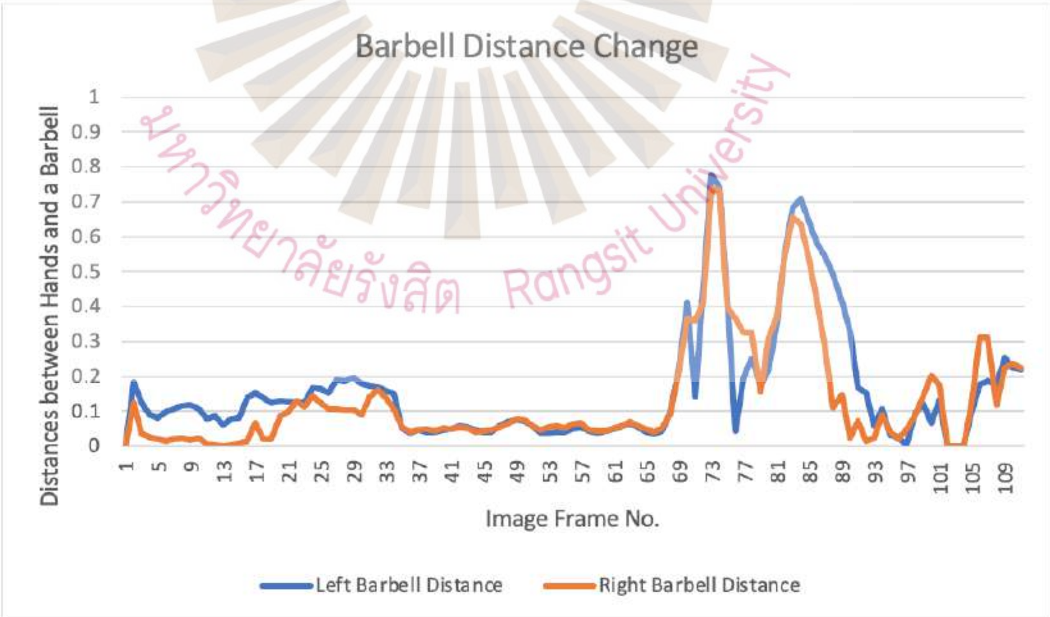


Figure 4.4 The fluctuation in the distances between hands and a barbell of an unsuccessful video

4.4 Determination of Success or Failure in Weightlifting Sequences

This section presented the results of detecting the success or failure of weightlifting from videos. The analysis used video image recordings, estimated phase sequences from the video sequences, and the distance between the hands and the barbell. Tables 4.2 to 4.7 demonstrate how six different testing methods perform in detecting errors and incidents in five types of weightlifting attempts. Six video-based approaches for assessing snatch weightlifting success are shown in Table 4.2.

Table 4.2 Six approaches used to determine the success of snatch weightlifting

Method Name	Description
M_SixHold	Judge based on the holding period of the sixth phase (subsection 3.5.9.1)
M_AllPhases	Judge based on the presence of all six phases in a phase sequence (subsection 3.5.9.2)
M_OrderPh	Judge from six phases in order (subsection 3.5.9.3)
M_AllPhSlip	Judge based on the ordered six phases, presence of all six phases and barbell slipping (subsection 3.5.9.4)
M_OrderHold	Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase (subsection 3.5.9.5)
M_OrderSlip	Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping (subsection 3.5.9.6)

The results of six different testing methods are displayed in Table 4.3 for five different categories (examples) of videos. The word "Unidentified" indicated that weightlifting failure could not be detected, whereas the word "Identified" indicated that it could.

Table 4.3 Results from six different testing methods on five different examples

Example / Method	Example 1	Example 2	Example 3	Example 4	Example 5
M_SixHold	Identified	Identified	Identified	Identified	Unidentified
M_AllPhases	Identified	Unidentified	Unidentified	Unidentified	Unidentified
M_OrderPh	Identified	Identified	Unidentified	Unidentified	Unidentified

Table 4.3 Results from six different testing methods on five different examples (cont.)

Method \ Example	Example 1	Example 2	Example 3	Example 4	Example 5
M_AllPhSlip	Identified	Identified	Unidentified	Unidentified	Identified
M_OrderHold	Identified	Identified	Identified	Identified	Unidentified
M_OrderSlip	Identified	Identified	Identified	Identified	Identified

4.4.1 Example 1: Initial Phases Barbell Slip Video

The video in this section shows an athlete performing the weightlifting sequence successfully through the first phases through the fifth phase, but failing to complete the sixth phase, which causes the barbell to drop. This part elucidated the process of recognizing an incomplete phase sequence during an unsuccessful effort. The figures 4.7, 4.8 and 4.9 depicted the outcome of an unsuccessful snatch weightlifting endeavour. The athlete in Figure 4.5 completes the fifth phase, but an injury causes them to give up on the lift and lose control of the barbell. Figure 4.6 depicts the phase sequence in which the athlete returns to the first and second phases' movements. The phase changes were shown in Figure 4.7, where the movements are shown in a continuous sequence with variation throughout the lift attempt.

According to the analysis, the athlete successfully finished the fifth phase in the 83th frame but stopped the lift prematurely owing to injury or other factors. In the 93rd frame, there is an evidence of an abortion. The frame was identified as the fourth phase. Then, in the 118th frame, the athlete performed the action that the second phase was recognized. Simultaneously, based on the distances between hands and a barbell, Figure 4.8 indicated that starting with the 118th frame, the barbell was no longer within the athlete's grasp. A significant increase in distances between hands and a barbell indicated the athlete's loss of control and grip on the barbell. The action was identified as the lift's first phase in the 145rd frame because the bar was still uncontrollably lifting. The lift attempt failed because the sixth phase's holding period was less than a predetermined threshold since there were no sixth phase frames.

a) In the 83rd frame of a weightlift

b) In the 93rd frame of a weightlift

c) In the 118th frame of a weightlift

d) In the 145th frame of a weightlift

Figure 4.5 Unsuccessful weightlifting screenshots

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	3	3	3	3	4	4	4	4
4	4	4	4	4	4	4	4	4	3	3	3	3	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0									

Figure 4.6 Results of unsuccessful weightlifting's classified phases (After the Phase Correction)

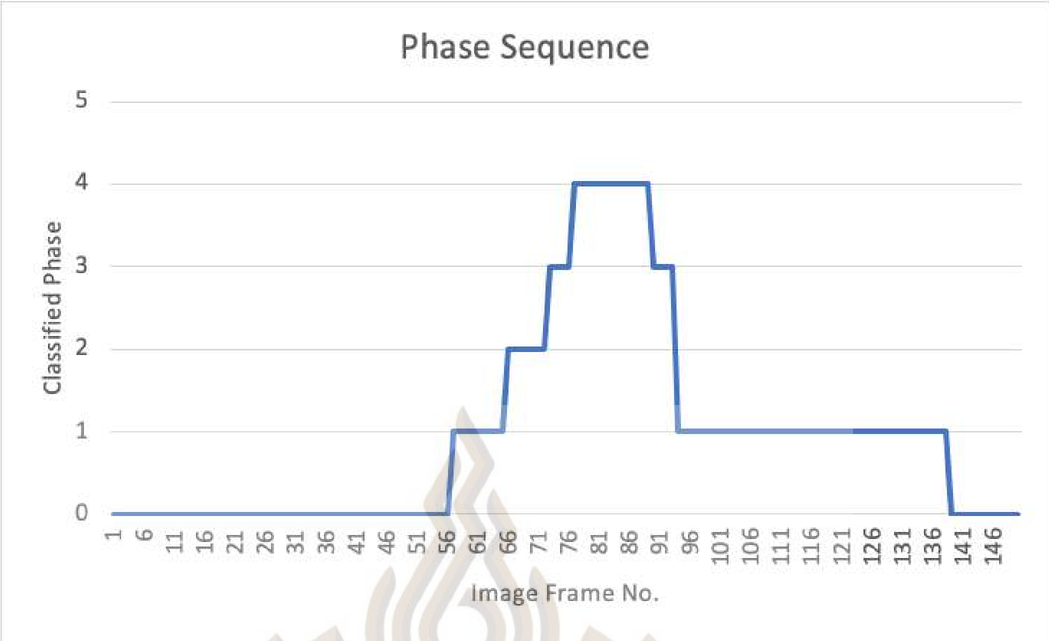


Figure 4.7 Classified phases of unsuccessful weightlifting

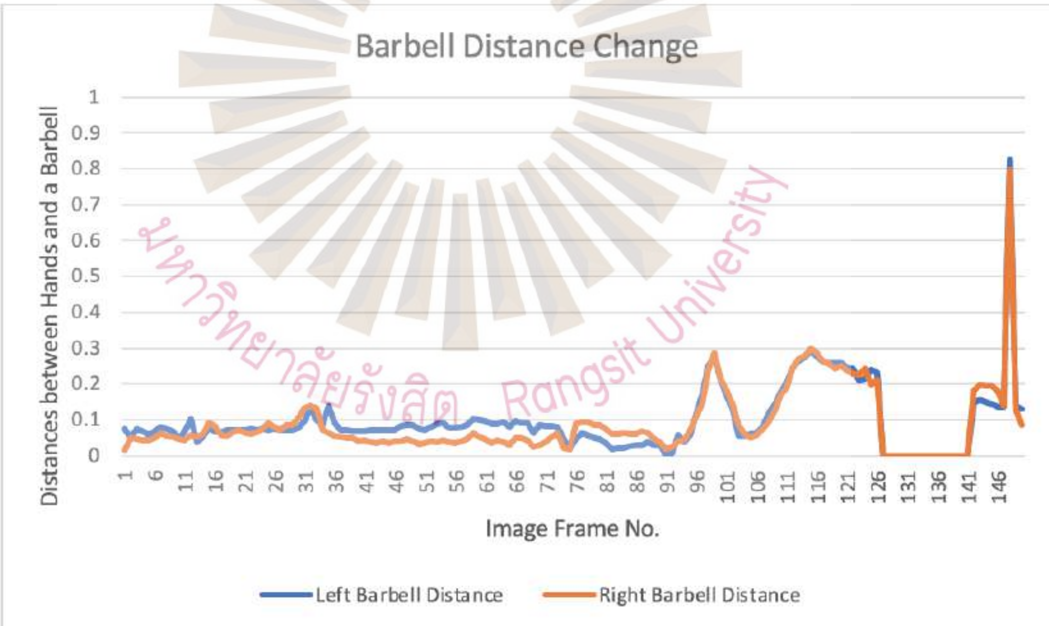


Figure 4.8 The fluctuation in the distances between hands and a barbell

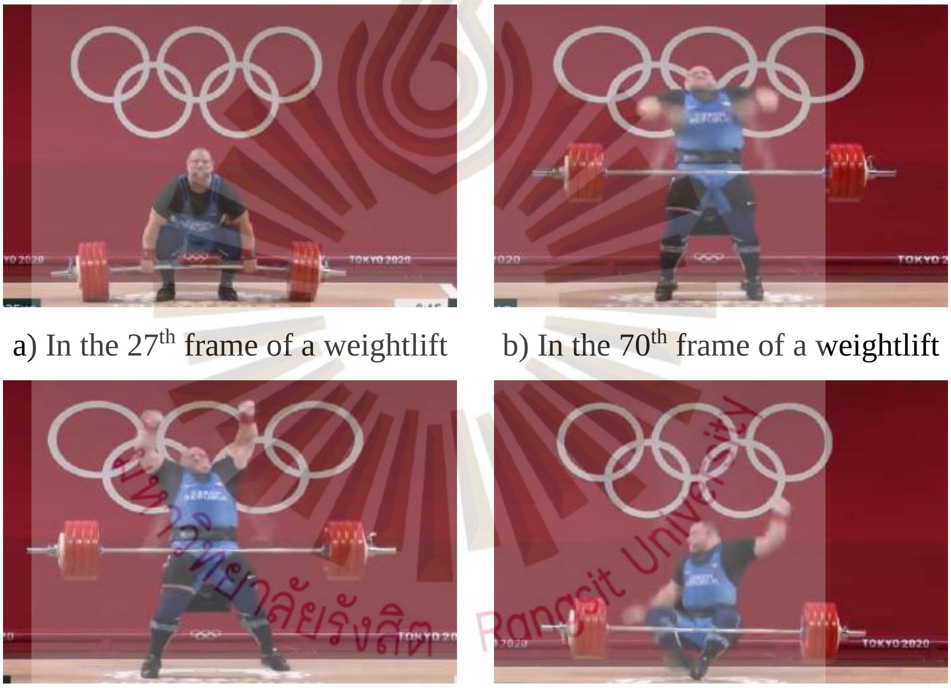
This example underwent several checks based on phase sequences and barbell handling. 1) The "Judge based on the holding period of the sixth phase" check was not passed, successfully identifying the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second. 2) The "Judge based on the presence of all six phases in a phase sequence" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not finish all six phases of weightlifting. 3) The "Judge from six phases in order" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not complete all six phases that followed correct order. 4) The "Judge based on the ordered six phases, presence of all six phases and barbell slipping" check was not passed. The sixth phase was not found. 5) The "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" check was not passed. No the sixth-phase found. The athlete did not finish all six phases of weightlifting in correct order. 6) The "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not sustain holding the barbell during the sixth phase of weightlifting for more than 1 second, or did not finish all six phases of weightlifting in correct order.

Therefore, the weightlifting failure video underwent testing utilizing six distinct methods, all of which accurately classified the first example as a weightlifting failure video.

4.4.2 Example 2: Later Phases Barbell Slip Video

The video in this section showed an athlete trying to complete the sixth phase of snatch weightlifting, but it was unsuccessful. In this example, all phases were presented. However, there was an abrupt halt or termination of the motions abruptly halted or terminated. Figures 4.11 and 4.12 display the outcomes of an unsuccessful snatch weightlifting attempt. Figure 4.9 shows a snapshot of the hand slip action during a failed snatch lift. Figure 4.10 shows the phase sequence, highlighting that all phases are present but in a disordered order. This was because the athlete’s movements lacked a smooth progression through the phases, particularly with no consistent trend towards the sixth phase. Figure 4.11 presents a curve graph illustrating the phase changes during the failed lift. The barbell distance changes in Figure 4.12 showed that the hands release the barbell in the latter half of the video, indicating a loss of control.

To elaborate, according to the analysis, these figures demonstrated that the athlete maintained the appropriate postures and phases from the 1st phase at the 27th frame to the 4th phase at the 69th frame. Between the 70th and 74th frames, anomalous motions were seen, specifically during the fourth and sixth phases, when the barbell had already slipped out of the athlete's grasp (Figure 4.12). During the 100th frame, the athlete lost balance and experienced more unconventional movements, finally leading to the failure of the lift attempt.



a) In the 27th frame of a weightlift   b) In the 70th frame of a weightlift

c) In the 74th frame of a weightlift   d) In the 100th frame of a weightlift

Figure 4.9 Unsuccessful weightlifting screenshots

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	2	2	2	3	3	5	5	5	5	3	3	4	4	4
1	3	3	3	3	3	3	3	3	3	4	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figure 4.10 Results of unsuccessful weightlifting's classified phases (After the phase correction)

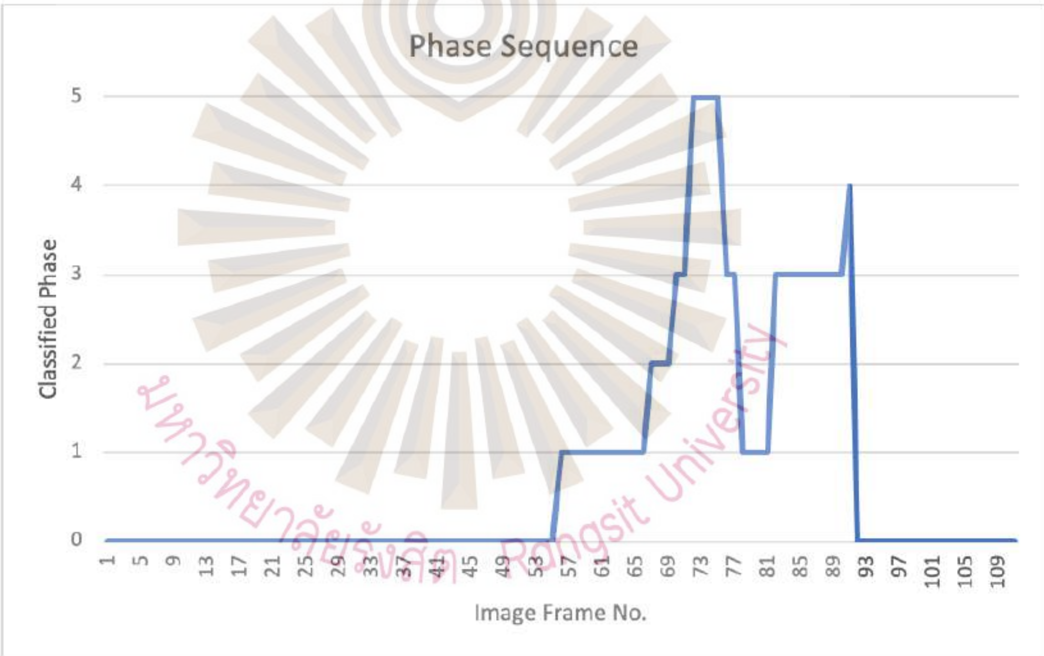


Figure 4.11 Classified phases of unsuccessful weightlifting

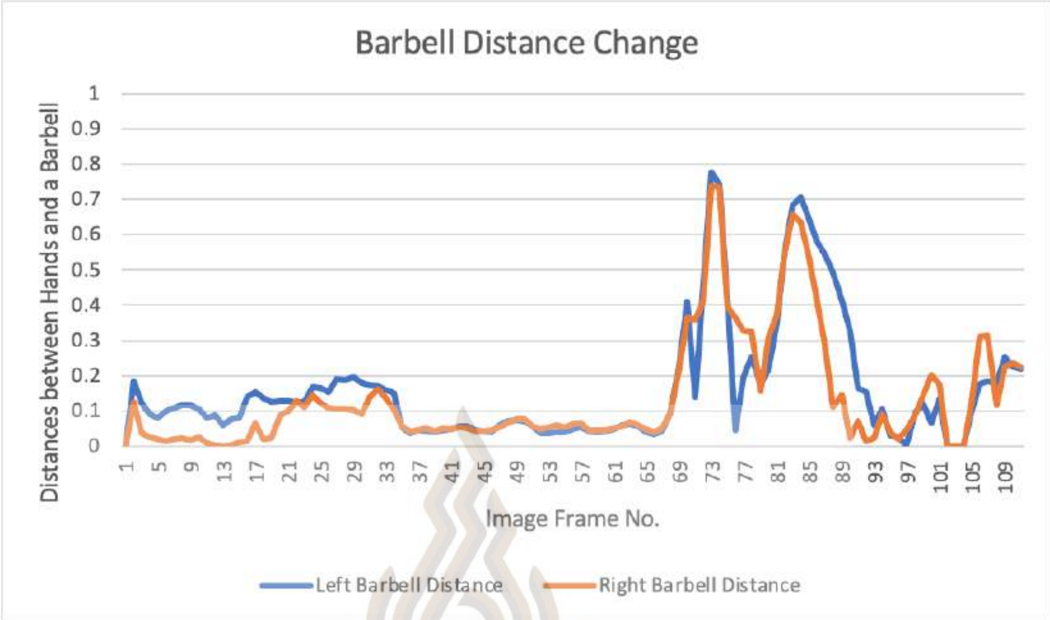


Figure 4.12 The fluctuation in the distances between hands and a barbell

This sample was subjected to multiple checks based on phase sequences and barbell handling. 1) The "Judge based on the holding period of the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second. 2) The "Judge based on the presence of all six phases in a phase sequence" check was passed, failing to identify the abort caused by injury or unexpected events in weightlifting. In fact, the athlete had finished all six phases of weightlifting. 3) The "Judge from six phases in order" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not complete all six phases that followed correct order. 4) The "Judge based on the ordered six phases, presence of all six phases and barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not hold the barbell in last two frames of sixth phase of weightlifting. 5) The "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second, or did not finish all six phases of weightlifting in correct order. 6) The "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not sustain holding the barbell during the sixth phase of weightlifting for more than 1 second, or did not finish all six phases of weightlifting in correct order.

Therefore, the weightlifting failure video underwent testing utilizing six distinct methods. Except for method M_AllPhases, all other methods correctly identified the second example as a weightlifting failure video.

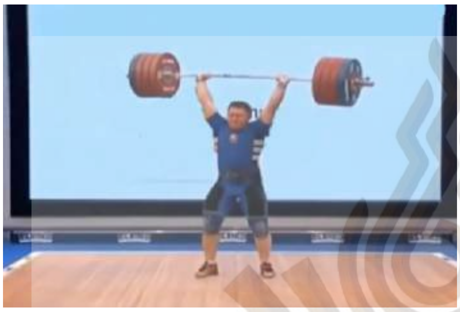
4.4.3 Example 3: 6th Phase Barbell Slip Video

The athlete in this section's video successfully completed all the necessary movements, but either an injury interrupted the final action of the sixth phase or it abruptly ended before the time deemed a successful lift. The barbell was also dropped. Figures 4.15 and 4.16 depict the outcomes of an unsuccessful snatch weightlifting endeavor. Figure 4.13 shows the athlete dropping the barbell while completing the sixth phase. Figure 4.14 displays the phase sequence, showing all six phases. However, the duration of the sixth phase was too short, followed by the barbell release. Figure 4.15 details continuous movement phase changes during the lift attempt. Significant distance changes in the latter part of Figure 4.16 indicated that the barbell was released.

Based on the analysis, the athlete successfully finished the first, second, and sixth phases in the 23rd, 61st, and 117th frames, respectively. The athlete successfully executed a lift of the barbell. According to Figure 4.16, the athlete experienced distance changes over the threshold after the 117th frame, indicating a loss of grip due to a malfunction. At the 143rd frame, the athlete relinquished control of the barbell and ceased the lifting attempt. The athlete was unable to sustain the sixth phase for the necessary duration, leading to a failure in the lift.

a) In the 23rd frame of a weightlift

b) In the 61st frame of a weightlift



c) In the 117th frame of a weightlift

d) In the 143rd frame of a weightlift

Figure 4.13 Unsuccessful weightlifting screenshots

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	3	3	3	3	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4
4	4	3	3	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2						

Figure 4.14 Results of unsuccessful weightlifting's classified phases (After the phase correction)

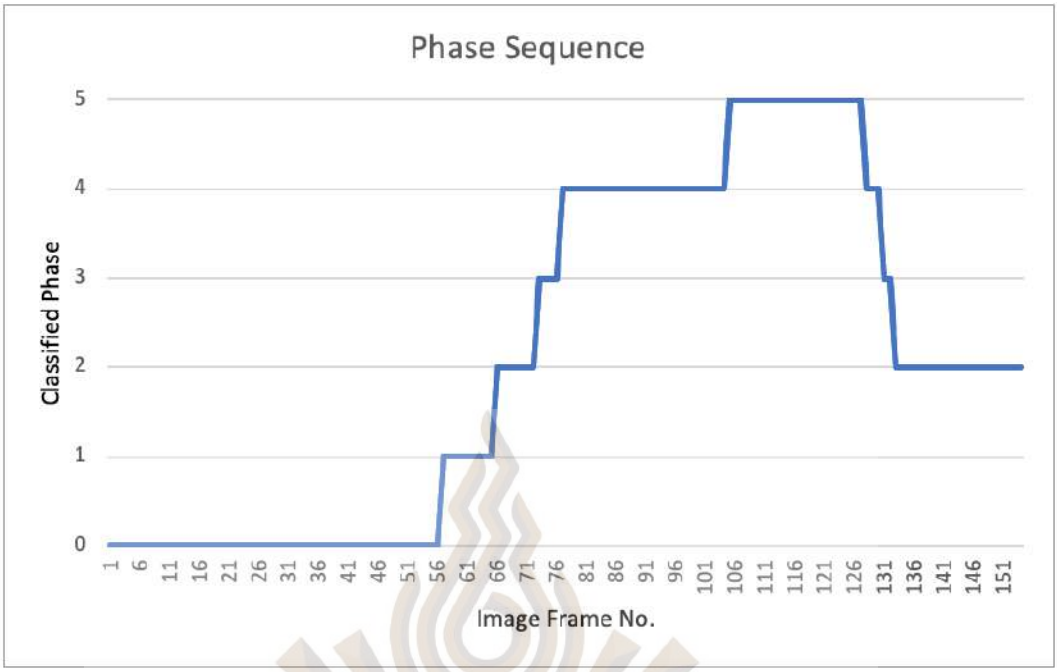


Figure 4.15 Classified phases of unsuccessful weightlifting

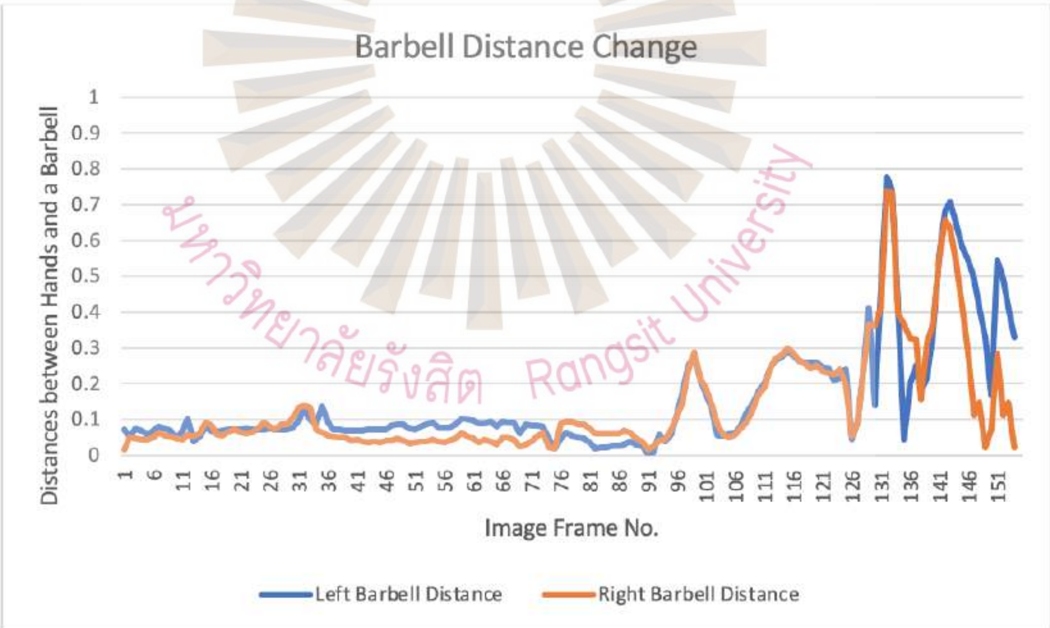


Figure 4.16 The fluctuation in the distances between hands and a barbell

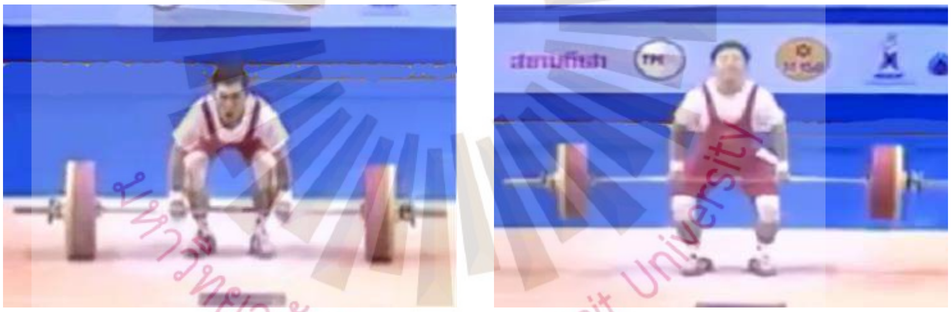
This sample was subjected to multiple checks based on phase sequences and barbell handling. 1) The "Judge based on the holding period of the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second. 2) The "Judge based on the presence of all six phases in a phase sequence" check was passed. The athlete had finished all six phases of weightlifting. 3) The "Judge from six phases in order" check was passed. The athlete had completed all six phases that followed correct order. 4) The "Judge based on the ordered six phases, presence of all six phases and barbell slipping" check was passed. After checking the distances between hands and a barbell from the first frame to the last maximum phase position or frame, the distances was lower than a threshold. 5) The "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second, or did not finish all six phases of weightlifting in correct order. 6) The "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not sustain holding the barbell during the sixth phase of weightlifting for more than 1 second, or did not finish all six phases of weightlifting in correct order.

Therefore, the weightlifting failure video underwent testing utilizing six distinct methods. Four methods correctly identified the third example as a weightlifting failure video; however, methods M_AllPhases and M_OrderPh did not identify the errors.

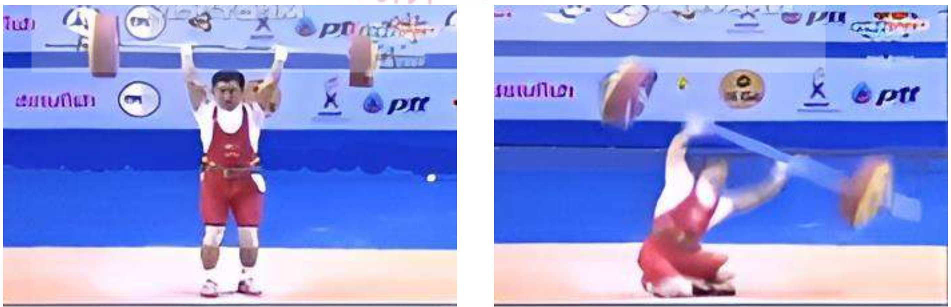
4.4.4 Example 4: No-Slip 6th Phase Fall Video

The video in this section showed an athlete who successfully completed all the preceding phases but was unable to maintain the sixth phase for the necessary duration. Despite completing all motions, the athlete disrupted his hold on the barbell during the sixth phase, making it challenging to determine if the barbell has released. Figures 4.19 and 4.20 depict the outcomes of an unsuccessful snatch weightlifting endeavor. Figure 4.17 shows the athlete fails to sustain the sixth phase, resulting in a fall, with the barbell remaining close to the hands. Figure 4.18 displays the phase sequence and shows all six phases completed, but the sixth phase did not hold for enough time, and the fall was categorized as an unrecognized phase. Figure 4.19 depicts the phase change sequence and illustrates continuous changes, detailing the actions taken during the lift attempt. Figure 4.20 shows the barbell distance change sequence, which shows significant fluctuations without loss of grip and with a shorter sixth phase duration than required.

According to the analysis, the athlete successfully completed the first, second, and sixth phases in the 12th frame, 28th frame, and 80th frame, respectively, and effectively lifted the barbell. The barbell was held by the athlete until the 82nd frame, as depicted in Figure 4.20. Upon reaching the 82nd frame, the athlete experienced a predicament that led to variations in distances; however, they remained within the acceptable limit, indicating the absence of any loss of grip. During the 109th frame, the athlete experiences a fall but manages to maintain their grip on the bar. The athlete failed to meet the necessary sixth-phase time requirement, leading to a failure to complete the lift.



a) In the 12nd frame of a weightlift b) In the 28th frame of a weightlift



c) In the 82nd frame of a weightlift d) In the 109th frame of a weightlift

Figure 4.17 Unsuccessful weightlifting screenshots

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	4	4	3	3	3	2	2	1	1	0	0	

Figure 4.18 Results of unsuccessful weightlifting's classified phases (After the phase correction)

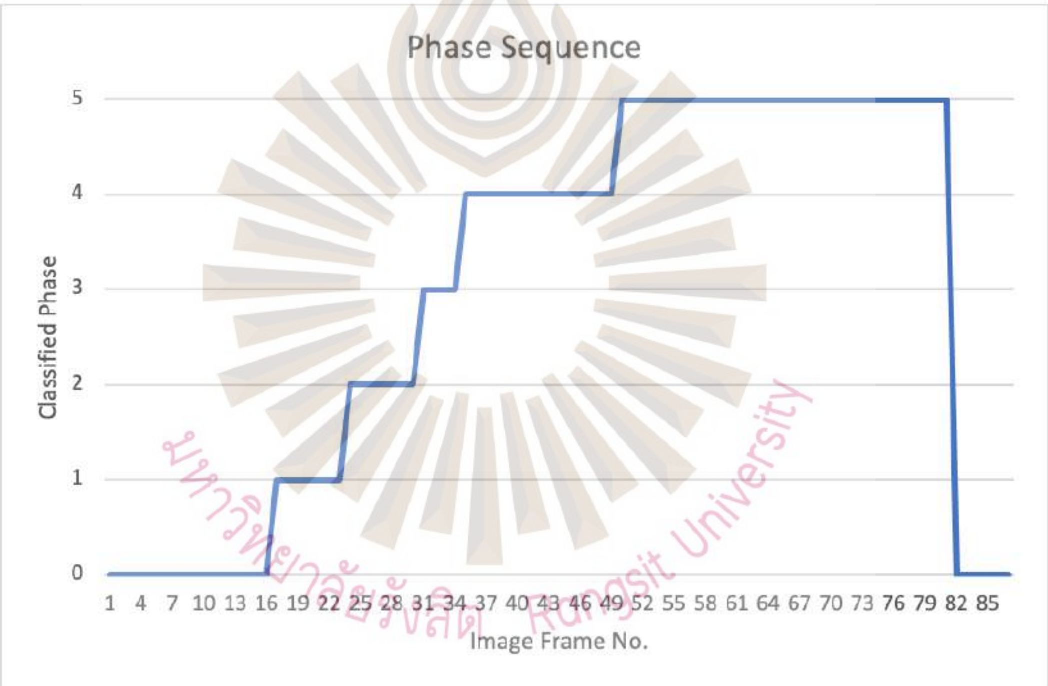


Figure 4.19 Classified phases of unsuccessful weightlifting

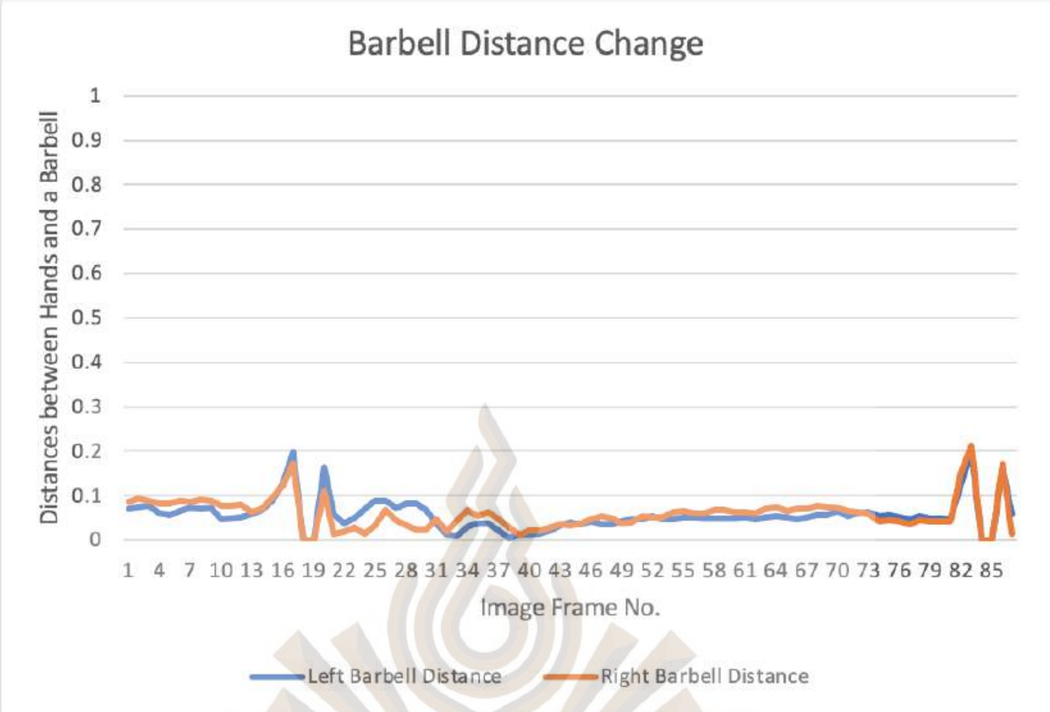


Figure 4.20 The fluctuation in the distances between hands and a barbell

This sample underwent several checks based on phase sequences and barbell handling: 1) The "Judge based on the holding period of the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second. 2) The "Judge based on the presence of all six phases in a phase sequence" check was passed. The athlete had finished all six phases of weightlifting. 3) The "Judge from six phases in order" check was passed. The athlete had completed all six phases that followed correct order. 4) The "Judge based on the ordered six phases, presence of all six phases and barbell slipping" check was passed, failing identify the issue caused by injury or unexpected events in weightlifting. In fact, the athlete had held the barbell during all phases of weightlifting (the distance threshold was fixed at 0.3). 5) The "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" check was not passed. The system identified the abort caused by injury or unexpected events in weightlifting. The athlete did not sustain the lifting posture during the sixth phase for more than 1 second. 6) The "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not sustain holding the barbell during the sixth phase of weightlifting for more than 1 second.

Therefore, the weightlifting failure video underwent testing utilizing six distinct methods. 3 methods correctly identified the fourth example as a weightlifting failure video, however methods M_AllPhases, M_OrderPh, and M_AllPhSlip did not identify errors

4.4.5 Example 5: Unchanged Posture After Barbell Slip

The video in this section shows an athlete seems to execute all the motions successfully. Nevertheless, as a result of an unforeseen occurrence, the athlete hurls the barbell aside. As the athlete is uninjured, he remains motionless for a brief period and raises his hands before accepting the error. Several frames that occur when the barbell is mistakenly thrown were identified as the sixth phase. The figures 4.23 and 4.24 depict the outcomes of an unsuccessful snatch attempt. In Figure 4.21, observe the athlete failing during the sixth phase, although the athlete completed all phases. Figure 4.22 shows the phase sequence chart. The athlete looked like completes all six phases with sufficient duration for the sixth phase. Figure 4.23 illustrates continuous changes in movement phases, providing a detailed visualization of the athlete's actions during the lift attempt. Figure 4.24 depicts the barbell distance change sequence. The latter part of the sequence shows significant fluctuations, indicating that the barbell was effectively released.

According to the analysis, the athlete finishes the first phase at the 10th frame, the fifth phase at the 54th frame, and the sixth phase at the 72nd frame. Efficiently executes a barbell lift during the sixth phase in the 89th frame, but then loses control and throws the barbell. According to Figure 4.24, the distance between the hand and the barbell surpasses the threshold after the 85th frame, suggesting a loss of grip. The alteration in distance resulted in the lift malfunctioning.

a) In the 10^{th} frame of a weightlift

b) In the 54^{th} frame of a weightlift

c) In the 72^{nd} frame of a weightlift

d) In the 89^{th} frame of a weightlift

Figure 4.21 Unsuccessful weightlifting screenshots

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2
2	2	2	2	2	3	3	3	3	3	3	3	3	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	

Figure 4.22 The results of unsuccessful weightlifting's classified Phases (After the phase correction)

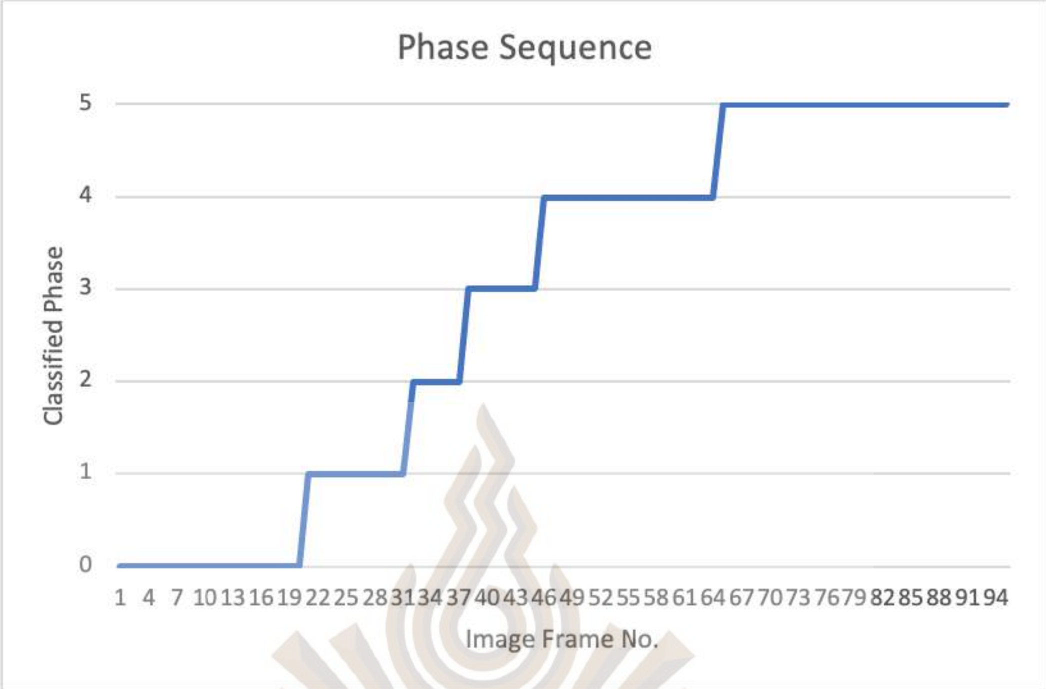


Figure 4.23 Classified phases of unsuccessful weightlifting

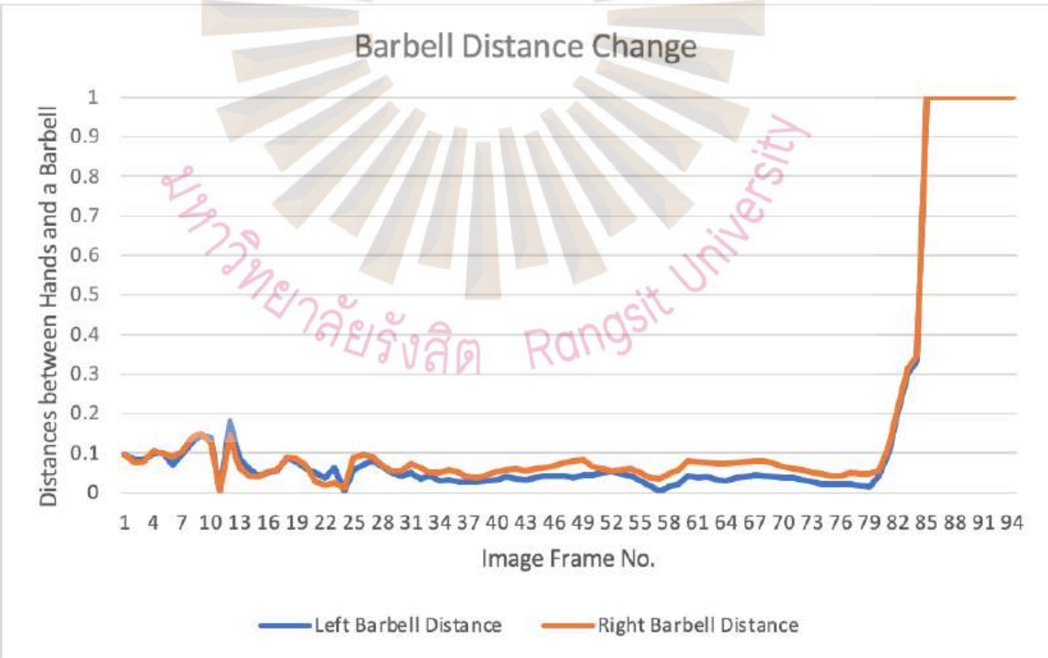


Figure 4.24 The fluctuation in the distances between hands and a barbell

This sample underwent several checks based on phase sequences and barbell handling: 1) The "Judge based on the holding period of the sixth phase" check was passed, failing to identify the abort caused by injury or unexpected events in weightlifting. In fact the athlete had sustained the lifting posture during the sixth phase for more than 1 second. 2) The "Judge based on the presence of all six phases in a phase sequence" check was passed. The athlete had finished all six phases of weightlifting. 3) The "Judge from six phases in order" check was passed. The athlete had completed all six phases that followed correct order. 4) The "Judge based on the ordered six phases, presence of all six phases and barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. The athlete did not hold the barbell in last frames of sixth phase of weightlifting. 5) The "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" check was passed, failing to identify the abort caused by injury or unexpected events in weightlifting. In fact the athlete had sustained the lifting posture during the sixth phase for more than 1 second. 6) The "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" check was not passed. The system identified the issue caused by injury or unexpected events in weightlifting. During the sixth phase of weightlifting, the athlete did not sustain holding the barbell for more than 1 second, and the distances between hands and a barbell were beyond the allowance.

Therefore, the weightlifting failure video underwent testing utilizing six distinct methods. 2 methods correctly identified the fifth example as a weightlifting failure video, however methods M_SixHold, M_AllPhases, M_OrderPh, and M_OrderHold did not identify errors

4.5 Determination of Snatch Weightlifting Success

This section provided the detection results of 20 video samples, including of 10 successful weightlifting attempts and 10 failure weightlifting attempts. The efficacy of each approach was assessed by its capacity to precisely distinguish between successful and unsuccessful snatch weightlifting attempts.

4.5.1 Accuracy of M_SixHold Method

The method described in Section 3.5.9.1, titled "Judging based on the holding period of the sixth phase" was applied in this experiment. The ability to determine success or failure in weightlifting was 90% accurate. As shown in Table 4.4, the system incorrectly identified 10% of unsuccessful lifts as successful lifts. Example 5's video provided the reason for the misidentification.

Table 4.4 Accuracy of the "Judge based on the holding period of the sixth phases" method

Prediction Result / Actual Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	10%	90%

4.5.2 Accuracy of M_AllPhases Method

The method described in Section 3.5.9.2, titled "Judging based on the presence of all six phases in a phase sequence" was applied in this experiment. The accuracy of determining success or failure in weightlifting was 40%. As shown in Table 4.5, the system incorrectly identified 60% of unsuccessful lifts as successful lifts, indicating that the determination based on the presence of all six phases in a weightlifting sequence was not enough and resulted in a substantial decline in the ability to identify unsuccessful lifts. The reasons for the misidentification were explained in Examples 2–5.

Table 4.5 Accuracy of the "Judge based on the presence of all six phases in a phase sequence" method

Prediction Result / Actual Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	60%	40%

4.5.3 Accuracy of M_OrderPh Method

The method described in Section 3.5.9.3, titled "Judge from six phases in order" was applied in this experiment. The accuracy of determining success or failure in weightlifting was 70%. As shown in Table 4.6, the system incorrectly identified 30% of unsuccessful lifts as successful lifts, indicating that the determination based on the six phases in order found in a weightlifting sequence was not enough. The reasons for the misidentification were explained in Examples 3–4.

Table 4.6 Accuracy of the "Judge from six phases in order" method

Prediction Result / Actual Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	30%	70%

4.5.4 Accuracy of M_AllPhSlip Method

The method described in Section 3.5.9.4, titled "Judging based on the presence of all six phases and barbell slipping" was applied in this experiment. The ability to determine success or failure in weightlifting was 80% accurate. As shown in Table 4.7, the system incorrectly identified 20% of unsuccessful lifts as successful lifts. The reasons for the misidentification were explained in Example 3.

Table 4.7 Accuracy of the "Judge based on the ordered six phases, presence of all six phases and barbell slipping" method

Prediction Result / Actual Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	20%	80%

4.5.5 Accuracy of M_OrderHold Method

The method described in Section 3.5.9.5, titled "Judging based on the ordered six phases, presence of all phases, and the holding time for the sixth phase" was applied in this experiment. The ability to determine success or failure in weightlifting was 90% accurate. As shown in Table 4.7, the system incorrectly identified 10% of unsuccessful lifts as successful lifts. The reasons for the misidentification were explained in Example 5.

Table 4.8 Accuracy of the "Judge based on the ordered six phases, presence of all phases and the holding time for the sixth phase" method

Prediction Result / Actual Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	10%	90%

4.5.6 Accuracy of M_OrderSlip Method

The method described in Section 3.5.9.5, titled "Judging based on the sequential six phases, the duration of the sixth phase, and the occurrence of barbell slipping" was applied in this experiment. The ability to determine success or failure in weightlifting was 100% accurate, suggesting the highest level of precision in identifying unsuccessful lifts compared to other approaches.

Table 4.9 Accuracy of the "Judge based on the ordered six phases, the holding time for the sixth phase and the barbell slipping" method

Actual Result \ Prediction Result	Successful Lifting	Unsuccessful Lifting
Successful Lifting	100%	0%
Unsuccessful Lifting	0%	100%

To sum up, Figures 4.25 and 4.26 revealed the accuracy of each method for judging successful and unsuccessful lifting videos.

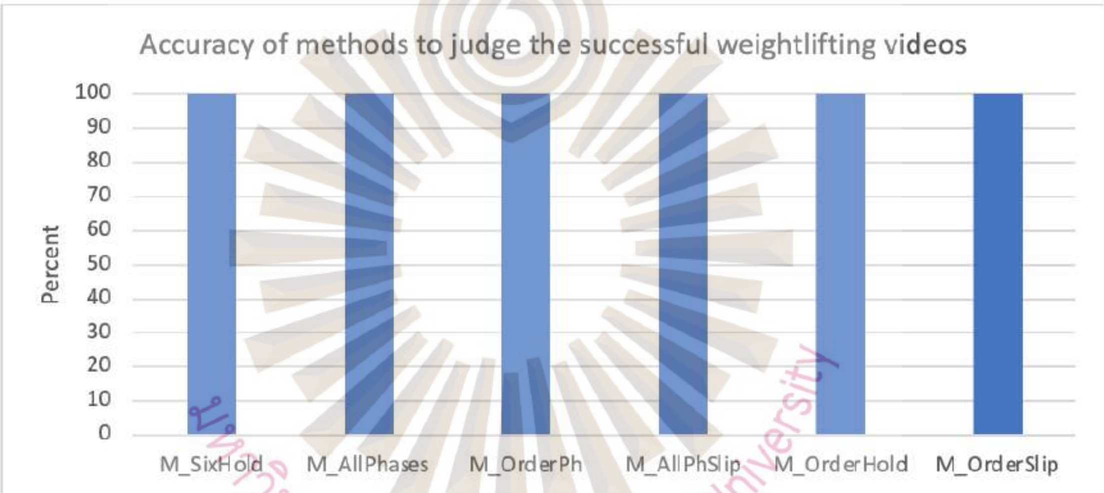


Figure 4.25 Accuracy of successful videos

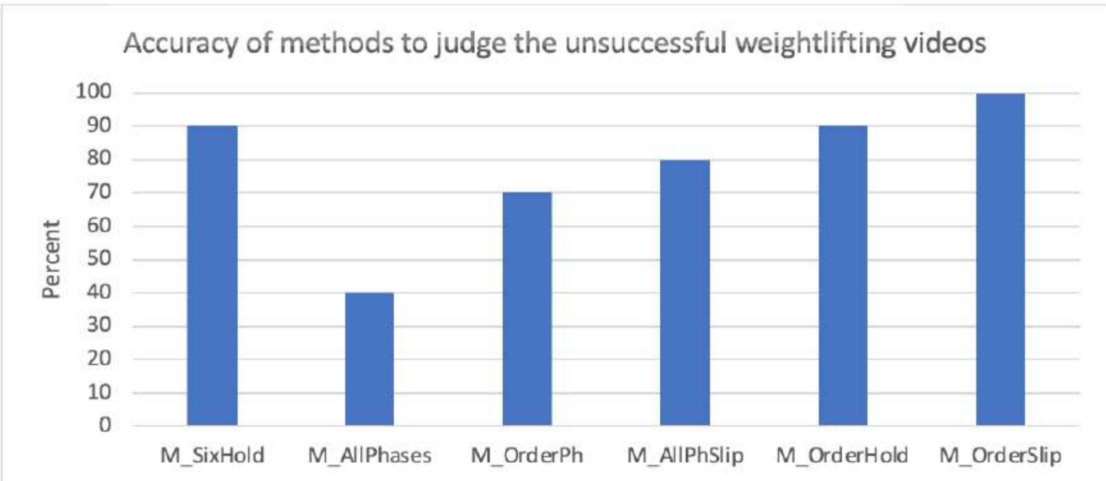


Figure 4.26 Accuracy of unsuccessful videos

Table 4.10 presents six distinct methodologies employed to assess the effectiveness of snatch weightlifting endeavors. The evaluation of these strategies was based on their capacity to effectively differentiate between successful and unsuccessful attempts. The method M_OrderSlip exhibited exceptional precision, attaining a flawless detection rate of 100%. The M_SixHold and M_OrderHold methods demonstrated impressive performance, each with an accuracy of 95%. The M_OrderPh method demonstrated a respectable accuracy of 85%, while the M_AllPhases method exhibited the lowest accuracy of 70%.

Table 4.10 Accuracy of algorithms to determine snatch weightlifting success

No.	Algorithms	Accuracy
1	M_SixHold	95%
2	M_AllPhases	70%
3	M_OrderPh	85%
4	M_AllPhSlip	90%
5	M_OrderHold	95%
6	M_OrderSlip	100%

Table 4.10 shows that the M_OrderSlip algorithm achieved a 100% accuracy rate, while M_AllPhases only achieved a 70% accuracy rate. This discrepancy can be attributed to the nature of the test samples and the methods used by each algorithm. In the test samples, the majority of failed snatch weightlifting attempts still completed all six phases of the lift. This means that the M_AllPhases method, which evaluates the presence of all six phases in sequence, struggled to differentiate between successful and failed attempts accurately. The presence of all phases does not necessarily indicate a successful lift, leading to a lower accuracy rate for M_AllPhases.

Chapter 5

Conclusion and Recommendations

5.1 Conclusion

This study encompassed a thorough research procedure aimed at identifying the success or failure of weightlifting in videos. The research method includes feature extraction, image classification, video sequence analysis, and the assessment of weightlifting success or failure.

Through a comparative study of image classification techniques and the application of various logical approaches to the analysis of video sequences, we have achieved the following significant results: 1) Initially, we conducted a thorough comparison of various image classification algorithms to choose the most appropriate one for weightlifting image classification. This step served as a solid foundation for our later study. 2) By employing logical reasoning, we were able to correct several errors in identifying the weightlifting phases. 3) Furthermore, we examined the utilization of several logical methods in the analysis of weightlifting videos. Through the integration of image classification, phase correction, video sequence analysis, the proposed method could effectively determine the success in weightlifting attempts.

This study provided a comprehensive method for the process of determining weightlifting videos, which produced remarkable results through analysis and experiments. The implications of our research are important for improving the accuracy of referees in weightlifting competitions.

5.2 Recommendations

Extend the scope of the research to encompass a wider variety of scenes and lifting scenarios. Examine whether it is possible to obtain a more complete view of weightlifting movements by using multi-angle video recognition.

The implementation of sequence analysis and prediction algorithms should be strengthened in order to obtain a deeper comprehension of weightlifting performances. Provide algorithms able to predict movement patterns and identify patterns indicating successful or unsuccessful lifts.

5.2.1 Limitations

Although our research has made notable progress in weightlifting analysis, it is important to highlight several limitations: 1) The use of truncated video in the analysis may disregard environmental influences. Future research should take into account the influence of environmental factors on judging weightlifting success. 2) The lack of sound recognition analysis impedes our understanding of the auditory cues present during weightlifting performances. Subsequent investigations should look into the use of sound recognition algorithms to get further contextual information.

5.2.2 Future Outlook

Examining the feasibility of implementing recognition systems on multiple platforms, including computers and smartphones running distinct operating systems.

Conduct an in-depth analysis of the environmental aspects, such as crowd noise, lighting conditions, and competition atmosphere, to determine their influence on weightlifting performances. Incorporate environmental data into analysis frameworks to enhance the comprehension of performance dynamics.

By prioritizing these specific areas, future studies can potentially enhance the method to determine the success or failure of both snatch and clean and jerk weightlifting. Furthermore, expanding the understanding of weightlifting biomechanics can lead to more efficient training methods and strategies for improving performance.

References

Aralimarad, M., Meena, S. M., & Mallapur, J. D. (2020). A Comprehensive Survey on Human Action Recognition. *International Journal of Recent Technology and Engineering (IJRTE), 9*(2), 902-908. https://doi.org/10.35940/ijrte.B3933.079220

Berretti, S., Daoudi, M., Turaga, P., & Basu, A. (2018). Representation, Analysis, and Recognition of 3D Humans: A Survey. *ACM Trans. Multimedia Comput. Commun. Appl.*, *14*(1), 1-36. https://doi.org/10.1145/3182179

Chatzitofis, A., Vretos, N., Zarpalas, D., & Daras, P. (2013). Three-dimensional monitoring of weightlifting for computer assisted training. *Proceedings of the Virtual Reality International Conference: Laval Virtual*.(pp. 1–7). ACM. https://doi.org/10.1145/2466816.2466820

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A Simple Framework for Contrastive Learning of Visual Representations. *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 1597–1607

Chernytska, O. (2022, August 18). *Training YOLO: Selecting Anchor Boxes Like Thi s. Towards Data Science*. Retrieved from https://towardsdatascience.com/trainin g-yolo-select-anchor-boxes-like-this-3226cb8d7f0b

Donahue, J., Hendricks, L. A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., & Darrell, T. (2017). Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(4), 677-691. https://doi.org/10.1109/TPAMI.2016.2599174

Feichtenhofer, C., Pinz, A., & Wildes, R. (2016). Spatiotemporal Residual Networks for Video Action Recognition. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems (Vol. 29)*. Curran Associates, Inc.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). Sebastopol, CA: O’Reilly.

References (cont.)

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 580–587. https://doi.org/10.1109/CVPR.2014.81

Girshick, R. (2015). Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., … Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv*. https://arxiv.org/abs/1704.04861

Karunaratne, A., Jayasooriya, C., Deegalla, S., & Navarathna, R. (2021). Objectively Measure Player Performance on Olympic Weightlifting. *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, 410–415. https://doi.org/10.1109/ICIAfS52090.2021.9605963

Kathuria, A. (2022, October 20). *Train YOLOv5 on Custom Data*. Retrieved from https://blog.paperspace.com/train-yolov5-custom-data/

Korayem, M. H., Mustafa, S. A., Korayem, A. H., & Amanati, A. (2010, July). Modeling of the Snatch Weightlifting. *Proceedings of the 10th Australasian Conference in Mathematics and Computers in Sport.*

Korkmaz, S., & Harbili, E. (2015). Biomechanical Analysis of the Snatch Technique in junior elite female weightlifters. *Journal of Sports Sciences, 34*(11), 1088–1093. https://doi.org/10.1080/02640414.2015.1088661

Kong, J., Deng, H., & Jiang, M. (2021). Symmetrical Enhanced Fusion Network for Skeleton-Based Action Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(11), 4394–4408. https://doi.org/10.1109/TCSVT.2021.3050807

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386

References (cont.)

Kukil. (2021, March 8). *Building a Poor Body Posture Detection & Alert System Using MediaPipe Body Tracking. Learn OpenCV*. Retrieved from https://learnopencv.com/building-a-body-posture-analysis-system-using-mediapipe/

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. https://doi.org/10.1109/5.726791

Liu, G., Fekete, G., Yang, H., Ma, J., Sun, D., Mei, Q., & Gu, Y. (2018). Comparative 3-Dimensional Kinematic Analysis of Snatch Technique Between Top-elite and Sub-elite Male Weightlifters in 69-kg Category. *Heliyon*, *4*(7), e00658. https://doi.org/10.1016/j.heliyon.2018.e00658

Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, *60*(2), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., … Grundmann, M. (2019, June 19). MediaPipe: A Framework for Building Perception Pipelines. *arXiv*. https://doi.org/10.48550/arXiv.1906.08172

Makdoun, I. (2022, March 17). *Simple Classification using ANN. Medium.* Retrieved from https://medium.com/@ibtissam.makdoun/simple-classification-using-ann-436c896e15c

Olaya-Mira, N., Soto-Cardona, I., Palacio-Peña, R., & Acevedo-Tangarife, N. (2020). Kinematic analysis of the snatch technique in high performance weightlifters. *Apunts Educación Física y Deportes*, *140*, 70–76. https://doi.org/10.5672/apunts.2014-0983.es.(2020/2).140.10

Parmar, P., & Morris, B.T. (2017). Learning to Score Olympic Events. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 76–84. https://doi.org/10.1109/CVPRW.2017.16

References (cont.)

Parmar, P., & Morris, B. (2019). What and How Well You Performed? A Multitask Learning Approach to Action Quality Assessment. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 304–313. https://doi.org/10.1109/CVPR.2019.00039

Patil, A., Rao, D., Utturwar, K., Shelke, T., & Sard, E. (2022). Body posture detection and motion tracking using AI for medical exercises and recommendation system. *ITM Web of Conferences*, *44*, 03043. https://doi.org/10.1051/itmconf/20224403043

Pirsiavash, H., Vondrick, C., & Torralba, A. (2014). Assessing the quality of actions. *In Computer Vision – ECCV 2014* (Vol. 8694, pp. 556–571). Springer.

Qi, M. & Phoophuangpairoj, R. (2024). Classification of Snatch Weightlifting Phases. *2024 12th International Electrical Engineering Congress (iEECON)*, 1–5. https://doi.org/10.1109/iEECON60677.2024.10537843

Ren, B., Liu, M., Ding, R., & Liu, H. (2024). A Survey on 3D Skeleton-Based Action Recognition Using Learning Method. *Cyborg and Bionic Systems*, *5*, 0100. https://doi.org/10.34133/cbsystems.0100

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, *39*(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Redmon, J., & Farhadi, A. (2018, April 8). YOLOv3: An Incremental Improvement. *arXiv*. https://doi.org/10.48550/arXiv.1804.02767

Saoudi, E. M., Jaafari, J., & Andaloussi, S. J. (2023). Advancing human action recognition: A hybrid approach using attention-based LSTM and 3D CNN. *Scientific African*, *21*, e01796. https://doi.org/10.1016/j.sciaf.2023.e01796

References (cont.)

Simonyan, K., & Zisserman, A. (2014). Two-Stream Convolutional Networks for Action Recognition in Videos. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems (Vol. 27)*. Curran Associates, Inc.

Ulareanu, M. V., Potop, V., Timnea, O. C., & Cheran, C. (2014). Biomechanical characteristics of movement phases of clean & jerk style in weightlifting performance. *Proceedings of the International Scientific Conference "Sports, Education, Culture - Interdisciplinary Approaches in Scientific Research(2013)"*, 137, 64–69. https://doi.org/10.1016/j.sbspro.2014.05.253

Viola, P., & Jones, M. (2001). Rapid Object Detection Using a Boosted Cascade of Simple Features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), (Vol. 1)*. https://doi.org/10.1109/CVPR.2001.990517

Wei, S., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016). Convolutional Pose Machines. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4724–4732. https://doi.org/10.1109/CVPR.2016.511

Wang, Z., Lu, H., Jin, J., & Hu, K. (2022). Human Action Recognition Based on Improved Two-Stream Convolution Network. *Applied Sciences*, *12*(12). https://doi.org/10.3390/app12125784

Xu, G.-L., Zhou, H., Yuan, L.-Y., & Huang, Y.-Y. (2021). Using Improved Dense Trajectory Feature to Realize Action Recognition. *Journal of Computers*, *32*(4), 94–108. https://doi.org/10.53106/199115992021083204008

Yoshikawa, F., Kobayashi, T., Watanabe, K., Shirai, K., & Otsu, N. (2010). Start and end point detection of weightlifting motion using CHLAC and MRA. *Proceedings of the 1st International Workshop on Bio-inspired Human-Machine Interfaces and Healthcare Applications* (pp. 44–50). SciTePress. https://doi.org/10.5220/0002813100440050

References (cont.)

Zhang, H.-B., Zhang, Y.-X., Zhong, B., Lei, Q., Yang, L., Du, J.-X., & Chen, D.-S. (2019). A Comprehensive Survey of Vision-Based Human Action Recognition Methods. *Sensors*, *19*(5),1005. https://doi.org/10.3390/s19051005

Biography

Name	Ming Qi
Date of birth	27 October 1981
Place of birth	Heilongjiang/China
Education background	Harbin Institute of Technology, Harbin, China Bachelor of Engineering in Computer Science and Technology, 2008
Address	No. 801, Building 9, Agile South Lake Peninsula Garden, 998 Tonghe Road, Baiyun District, Guangzhou, Guangdong Province, China
Email Address	ming.q65@rsu.ac.th