High Level Implementation in Olympic Weightlifting of Hybrid Features Defined by Static and Spatio-Temporal Features

Ryan Perkins
University of Georgia

Abstract

The application of deep learning can provide insight to computers from data collected from almost any imaginable source. I begin my approach to the application of deep learning by studying the basic concepts of computer vision, video classification, convolutional neural networks, and using obscure data sources to supplement the context of video. Using these concepts, I extracted hybrid features [5] from Olympic weightlifting events to develop a model capable of classifying various weightlifting movements. The model achieved an average accuracy of 97.01% classification accuracy on testing videos scaled down to as little as two frames per second.

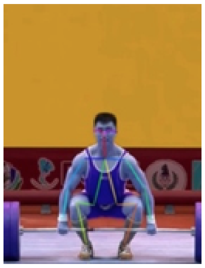
1. Introduction

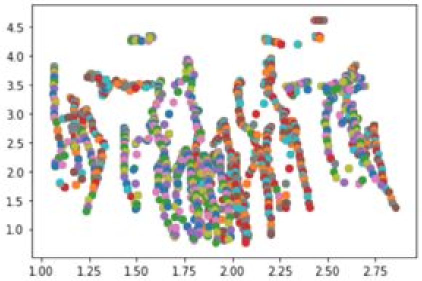
Olympic weightlifting is an extremely technical form of weightlifting that requires concurrent execution of strength, agility, and mobility to lift astonishing weight overhead. It requires not only consistent movement throughout the lift, but also ideal anatomical proportions. For this reason, I extracted data on the static and spatio-temporal features associated with a lifters joints to implement and show the power of hybrid features as well as correctly classify Olympic weightlifting movements—the snatch, power snatch, clean, and power clean.

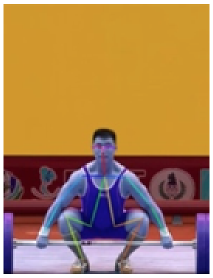
Convolution Neural Networks (CNN) have become one of the most common implementations of neural networks for image recognition in modern machine learning. The model uses a customizable kernel that iterates over a matrix of pixel data to extract key feature details. These details are then fed into a multi-layer perceptron for learning. However, the high dimensional nature of both the processing of video and the contents within video (i.e., the environment) introduces a challenge when attempting to achieve real-time performance. The features and environment constantly change over the course of a video and inconsistent viewing angles, distances, and image clarities within in each frame requires a model to be highly robust. A simple solution to add such robustness is to add motion as a feature—although the physical form of a figure may change across frames, the *movement* of the same elements across the same frames may have consistencies and patterns that can aid in classification.

Paired with modern machine learning techniques and computational power, collecting and processing video data in parallel to classify video in real-time has become increasingly possible. A possible high-level application of real-time video analysis is in the classification of unique movements performed by individuals dedicated to achieving perfection. Olympic weightlifters.

Sports at all levels require some degree of accountability for fair play and all sports have their respective accountability monitor. Some have judges and some have referees. In American Football, calls by the referees may be challenged and video replay confirmed, but in Olympic gymnastics, judges are only allowed to deduct







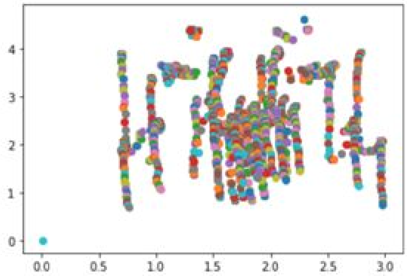


Fig 1. (top left) Choe Jon Wi in the starting clean position.
Fig 2. (top right) 2D representation of his joints across the video.

Fig 3. (bot. left) Choe Jon Wi in the starting snatch position.
Fig 4. (bot. right) 2D representation of his joints across the video.

points for what they can literally see during the performance. Both have some variation of check to ensure the rules are followed as closely as possible, but why not help the Olympic gymnastics judges and provide them with the technology that can spot *all* imperfections.

In this paper, I begin my exploration of the application of deep learning for computer vision in the Olympic sport of weightlifting. In section 2, I discuss the steps, concepts behind, performance of, and various approaches to classification of video in depth. In section 3, I walk through my data collection process, preprocessing stage, and machine learning strategies invoked. The remaining sections then discuss results and propose structure of future research.

2. Literature Review

2.1 Convolution

The kernel attributes, padding amount, stride length, and pooling specifications are all highly customizable attributes that account for the shape of the output from the convolution and pooling layers. The size of these outputs affects the amount of data that a model must process for classification and thereby affects the runtime. University of Montréal researchers V. Dumoulin and F. Visin dive deep into the concepts surrounding both convolution and pooling arithmetic [10].

The use of various kernel parameters can greatly affect the performance of a model as it determines how efficiently the images will be processed. This was a major factor when choosing a model for interest point data extraction as I would need to process thousands of frames to source sufficient training and later, testing data. The model that I utilized was a pose estimation model that located eighteen various joints on any number of individuals in an image.

2.2 Approaching Video

The standard approach to video classification has largely been broken into three stages [5, 6, 8].

First, the visual features that describe the region of a video are extracted either densely or at a sparse set of interest points. These features can further be broken into a few categories defined by researchers J. Liu, J. Luo, and M. Shah: Local motion features – features in view that emulate motion, well predicted by discriminative learning models, such as SVM. Local static features – just as humans can recognize motion, static figures in images provide context of an environment and can aid in classification of video. Hybrid features – many videos may have pose information that conflicts with its motion information [5]. For example, when looking at the pose of a weightlifter in their starting position, the position of their joints in a 2D space may look very similar for a snatch and clean & jerk, *Fig. 1 and 3*. The 2D movement of the joints may also look very similar since when viewing from a direct angle, all movement is relatively vertical , *Fig. 2 and 4*.

However, the proportion of movement, of and in between their joints differs. Unlike above where the lifter's physical positions are similar, his movements vary. The movement of Choe's right wrist during a clean in *Fig. 5.* has a much more consistent movement while his right wrist in a snatch in *Fig. 6.* has a much more explosive start, and the levels out.

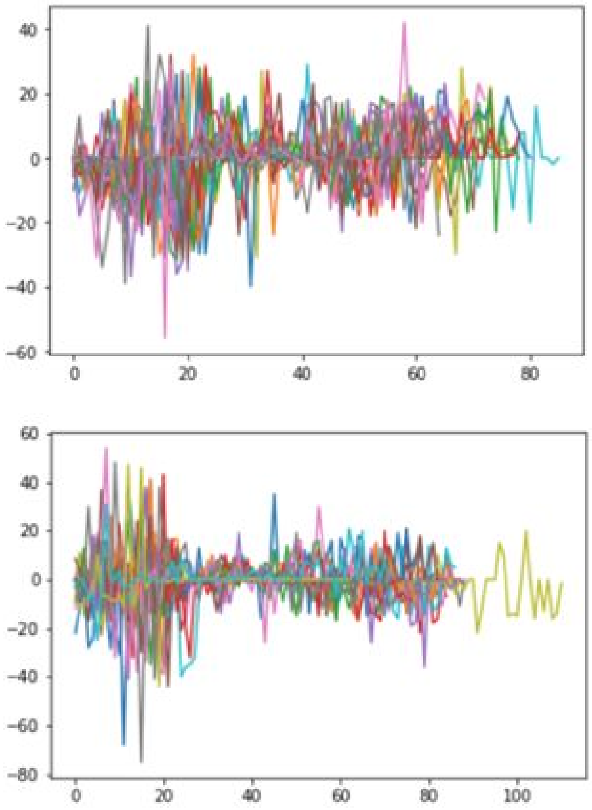


Fig 5. (top) movement of the right wrist in a clean.
Fig 6. (bot.) movement of the right wrist in a snatch.

This concept of patterns in movement can be capitalized on to improve the overall accuracy of a model within an ever-changing environment.

The use of the hybrid features allowed researchers to attain average accuracies ~5.8% higher than motion features alone [5]. These results influenced my decision to use the distance between the 2D coordinates of the joints as static features and the motion of said features as spatio-temporal features to create hybrid features that define various Olympic weightlifting movements as classifiable targets.

Secondly, features are translated into a collection of video level descriptions—one implementation of this approach was at the University of California, San Diego. Researchers P. Dollár, V. Rabaud, G. Cotterll, and S. Belongie defined these descriptions in their model as the output of an object recognition cuboid cluster. Their method included extracting a large number of cuboids from training data that consisted of spatio-temporal windowed pixel values that were transformed to include normalized pixel values, the brightness gradient, and windowed optical flow. One of three methods were then tested to create a classifiable feature vector from the given transformed cuboid [6]:

(1) flattening the cuboid into a vector,
(2) histogramming the values in the cuboid,
(3) local histograms, used as part of Lowe's 2D SIFT descriptor—provided the best performance.

The practice of using cuboids was also used by A. Klaser, I. Laptev, C. Schmid, et al. They compared the detection accuracy of cuboids to the performance of Harris3D, Hessian, and dense detectors. The researchers even applied the same smoothing kernel and 1D Gabor filters to their model that were applied by the U.C. San Diego team. Results found that the dense sampling outperforms the other detection methods, but performs worse on the KTH dataset, a basic video dataset consisting of six human actions including walking, jogging, running, boxing, hang waving, hand clapping [8]. Although I will not be implementing one of the above listed motion detectors, I am planning to base my approach off their concept of movement data extraction. The detectors above are used to detect different forms of movement (i.e., rapid, background panning, etc.) in unsupervised environments. Patterns in the detected movement are then classified as various forms of movement. However, I plan to focus on key interest points thereby removing the need for detection. Classification will then commence to locate patterns within the movement of the located interest points.

Lastly, a classifier is trained on the resulting descriptions to distinguish among the visual classes of interest—the cuboids above were clustered by a k-means clustering algorithm during training, and then cuboids generated from new video were compared to said clusters and classified as either a known type or an outlier.

2.3 Traditional Approach

There are two traditional approaches for motion analytics in video [7, 9].

First is computing of the optic flow—Dr. Florain Raudies defines optic flow as the calculation of the change of structured light in the image on the retina or the camera's sensors, due to a relative motion between the eyeball or camera and the scene. *Fig 7.* illustrates just the beginning to a proposed solution to the estimation of optic flow and few of the many constraints that may be specified [9].

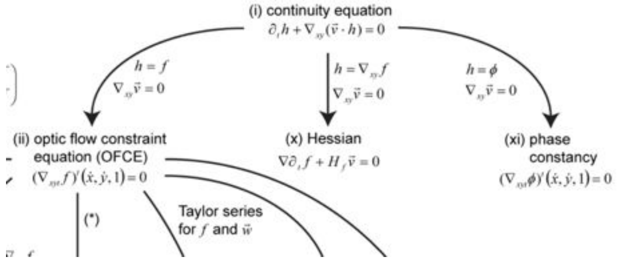

(i) continuity equation
$\partial_t h + \nabla_{xy}(\vec{v} \cdot h) = 0$
$h = f$
$\nabla_{xy}\vec{v} = 0$
$h = \nabla_{xy} f$
$\nabla_{xy}\vec{v} = 0$
$h = \phi$
$\nabla_{xy}\vec{v} = 0$
(ii) optic flow constraint equation (OFCE)
$(\nabla_{xyt} f)^t (\dot{x}, \dot{y}, 1) = 0$
(x) Hessian
$\nabla\partial_t f + H_f \vec{v} = 0$
(xi) phase constancy
$(\nabla_{xyt}\phi)^t (\dot{x}, \dot{y}, 1) = 0$
(*)
Taylor series for f and $\vec{w}$

Fig 7. the beginning of the flow of constraints during the computation of optic flow.

The second is feature tracking—where the detected feature clusters found within an image are tracked over the course of a video to detect the motion.

While both have their advantages, I. Laptev found that optic flow approaches mostly capture first-order motion and may fail with more instances of motion and feature trackers often assume constant appearances that may fail if two objects merge or split. However, he also found that "neighborhoods can be described in terms of spatio-temporal derivatives and then be used to distinguish different events in video." Laptev was successful in his research to generate neighborhoods of spatio-temporal data normalized with Gaussian derivatives capable of classifying walking [7].

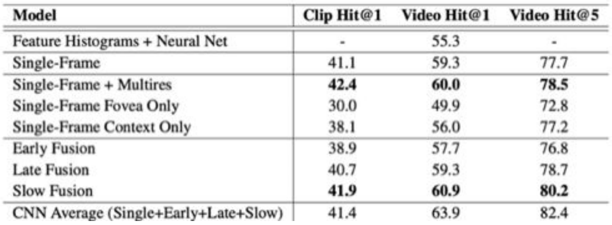
The success in classifying movement from spatio-temporal features attained by Laptev encourages my approach of using the movement of known interest points as spatio-temporal features. However, I plan to use a random forest classifier instead of k-means clustering due to the number of features and their individual weights.

2.4 Part Affinity Fields

The work and analysis from previous researchers not only outlines the approach I plan to take, but also justifies the models I use to extract required data. An example of one of these models is the pose estimation model I use to obtain usable interest point data from video of weightlifters. The model is from the work of Z. Cao, T. Simon, S.E. Wei, and Y. Sheikh and is powered by a CNN. The model also returns the 2D pixel coordinates of each joint [2, 3]. This data can be transformed into static and spatio-temporal features to define hybrid features that are Olympic weightlifting movements. The static features are the distances between various joints proportions in each frame of the video and are defined by the average distance between two joints, between two frames. The spatio-temporal features are the *change* between two joint distances, between two frames.

2.5 Large Scale CNN

Google and Stanford researchers A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei testing various video classification practices with CNN's on a dataset of 1 million YouTube videos



Model	Clip Hit@1	Video Hit@1	Video Hit@5
Feature Histograms + Neural Net	-	55.3	-
Single-Frame	41.1	59.3	77.7
Single-Frame + Multires	**42.4**	**60.0**	**78.5**
Single-Frame Fovea Only	30.0	49.9	72.8
Single-Frame Context Only	38.1	56.0	77.2
Early Fusion	38.9	57.7	76.8
Late Fusion	40.7	59.3	78.7
Slow Fusion	**41.9**	**60.9**	**80.2**
CNN Average (Single+Early+Late+Slow)	41.4	63.9	82.4

Fig 8. The results from the large-scale CNN study by Google and Stanford researchers [1].

They compared multiple video data extraction formats including single-frame – where a model simply classifies video by aggregating predictions across single frames, early fusion – the model takes a large contiguous clip from the video, late fusion – the model concatenated the first and last frames of the clip, and slow fusion – similar to early fusion with long contiguous clips, but it combines 4 partially overlapping clips in the convolutional layers. Results are displayed in *fig 8.* [1].

Since classification where ground truth is provided called for double the accuracy (column titled "Video Hit@5"), this reinforced my plan to approach classification with a single-frame, supervised learning strategy.

2.6 Application

An interesting paper I researched was from the work of I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, who worked to classify natural human action in movies by classifying human action from the movie's script and providing automatic annotations. The use of text to annotate motion in video shows how obscure data sources help provide large amounts of context for more complex classification [4]. This is not currently applied in my research but it opens the idea of using obscure data sources to provide context. Using obscure data sources in the classification of Olympic weightlifting movements from video is just one way to increase the accuracy and reliability of classification models. Many frames include fixed static items like barbells and scoreboards—even the scorecard data itself could be used with text-based classification.

Following the basic approach to classifying video [5, 6, 8], through the combination of using motion as a reliable movement classifier [4, 7, 9], and using CNN's to extract reliable and accurate data [1, 2, 3, 10], I developed a classifier capable of classifying Olympic weightlifting movements from video, regardless of the viewing parameters.

3. Methods

3.1 Data

Video data for training video classification models is extremely sparse, even now with the digitization and publication of most modern records. Narrowing the scope to the small sport of Olympic weightlifting does not help. However, as with any elite level sport, Olympic weightlifters record hours of material to critique form, teach to beginners, and publicize the practice. Global competitions are also held and documented extensively.

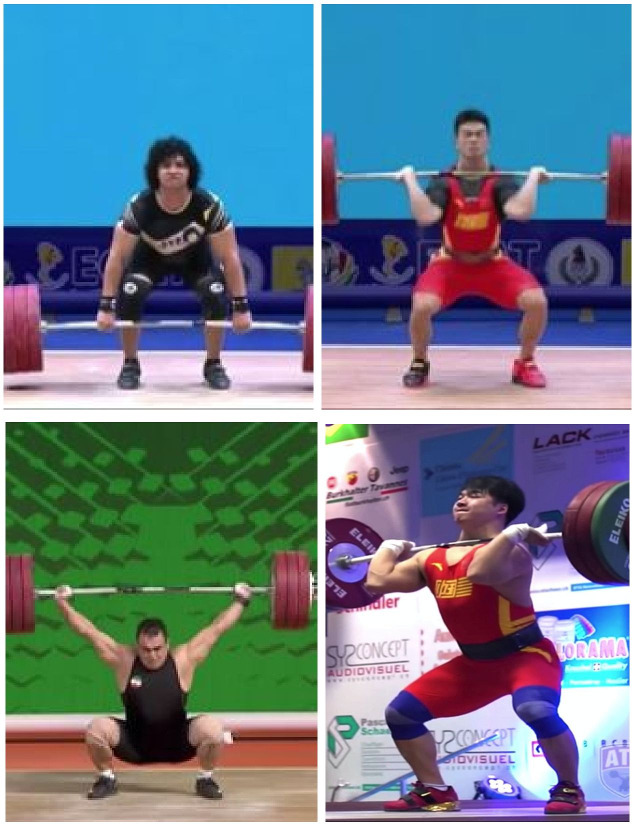


Fig 9. (top left) Meso Hassona - clean
Fig 10. (top right) Shi Zhiyong - power clean
Fig 11. (bot. left) Sohrab Moradi - snatch
Fig 12. (bot. right) Tian Tao - power clean

The training data was extracted from a number of Olympic athletes including China's Shi Zhiyong, Qatar's Meso Hassona, and Iran's Sohrab Moradi lifting at various International Weightlifting Federation (IWF) World Championship events, *Fig. 9, 10, 11.* The consistency of IWF media allowed for both optimal and consistent movement, angles, and camera distance. However, since these recording parameters cannot be controlled indefinitely, supplemental training data of Chinese lifters at Chinese Power Weekend 2019 was added to the training data to add feature data from about a 45-degree angle, *Fig 12*.

	candidate in frame		
joints	**0**	**1**	**2**
nose	(515, 491, 2)	…	…
neck	(517, 458, 5)	…	…
Rsho	(443, 452, 8)	…	…
…	…	…	…

Table 1. Peaks – defines, for each candidate in the frame, the coordinates (x, y) and the joint numbers.

	joint		
candidate	**nose**	**neck**	**Rsho**
0	(2)	(5)	(8)
1	…	…	…
…	…	…	…

Table 2. Subset – defines which joint number is associated with which body part and candidate.

x	y	weight	joint #
515	491	…	2
517	458	…	5
443	452	…	8
…	…	…	…

Table 3. Candidate – defines the coordinates (x, y), weight, and joint number for all joints.

All training data was recorded at 30 fps, in a 910x1184 pixel frame. The 2D joint coordinate, static feature, spatio-temporal feature, and hybrid feature data were all individually extracted from video of 18 cleans, 11 power cleans, 20 snatches, and 7 power snatches. The datasets were created from the joint coordinates extracted by the pose estimation model which returned peak, subset, and candidate data frames as described in *Tables 1, 2, and 3*. This allowed me to extract specific movement data for each of the candidates mapped in each frame. Although the ability to recognize and classify any number of candidates performing a movement in a frame without loss of efficiency is extremely useful, I ensured that all training videos only featured one candidate. The multi-pose estimation efficiency of this model would pair well with the application of motion classification, but not necessarily the training. Further transformation of the extracted data would also be required to feed the movement data in frame-by-frame, candidate-by-candidate.

Below is a list of all of the joints that are located by the model

- Head: nose, neck, right eye, left eye, right ear, left ear
- Arms: right shoulder, right elbow, right wrist, left shoulder, left elbow, left wrist
- Legs: right hip, right knee, right ankle, left hip, left knee, left ankle

3.2 Preprocessing

The following joint proportions were then taken from the joint pixel coordinates:

Arm width		
R. wrist	→	L. wrist
R. elbow	→	L. elbow
Arm to arm		
R. wrist	→	R. elbow
R. wrist	→	R. shoulder
R. elbow	→	R. shoulder
L. wrist	→	L. elbow
L. wrist	→	L. shoulder
L. elbow	→	L. shoulder
Leg to leg		
R. ankle	→	R. ankle
R. ankle	→	R. ankle
R. knee	→	R. knee
L. ankle	→	L. ankle
L. ankle	→	L. ankle
L. knee	→	L. knee
Arm to leg		
R. wrist	→	R. hip
R. wrist	→	R. knee
R. wrist	→	R. ankle
L. wrist	→	L. hip
L. wrist	→	L. knee
L. wrist	→	L. ankle
R. elbow	→	R. hip
R. elbow	→	R. knee
R. elbow	→	R. ankle

L. elbow	→	L. hip
L. elbow	→	L. knee
L. elbow	→	L. ankle
Height measurements		
nose	→	R. ankle
nose	→	L. ankle
nose	→	R. knee
nose	→	L. knee
nose	→	R. hip
nose	→	L. hip
nose	→	R. wrist
nose	→	L. wrist
nose	→	R. elbow
nose	→	L. elbow

Table 4. joint proportions measured.

The Euclidean distance between each of the above listed relationships was then calculated. An average of these values between successive frames represent the static features of my hybrid features. Then for each distance, the change in distance between successive frames was calculated to equal the motion variable that represents the spatio-temporal features of my hybrid features.

With the combined accuracy of the part affinity field model and the handling of values, of the over 3,000 labeled instances created, there were almost no missing instances in the training data. Any missing variables in the training data were caused by a candidate missing coordinates for a specific joint and thereby causing no distance to be calculated. To account for missing joint data, the peak coordinates of the same joint from the previous frame were appended in its place. This caused the average distance, the static feature, to alter only slightly and the change in the change in distance, the spatio-temporal feature, to equal zero. This leads the model to assume that the specific feature is in a state of no motion. This is acceptable because the data is no longer seen as an outlier since the pixel change between frames of 30 fps video should be small enough to where no motion of some joints is a semi-frequent occurrence. However, in the case of missing values for the distance between joints, zero does not signify no motion, it signifies that there is no 2D space in between the joints. This does affect the model as an outlier, hence the appendage of preceding frame data.

3.3 Machine Learning Strategies

The classifier of choice for the Olympic weightlifting movement dataset was a Random Forest classifier. The large number of features and their individual low levels of importance makes a random forest desirable because it will test the different features in a customizable amount of decision trees. The reason for choosing decision trees is that when decision trees split for numeric attributes, they split into ranges. This means that during testing, if the frames per second (FPS) are reduced, the spatio-temporal features may numerically increase since the distance between joints between frames will be greater. The numerical splits will then hopefully favor the larger movement values when attempting to classify movement.

The number of estimators was the only noticeable parameter and the cost of time began to outweigh the benefit of accuracy as supplemental estimators were added. An estimator with fifty decision trees was chosen based off the results shown in *Fig. 13*.

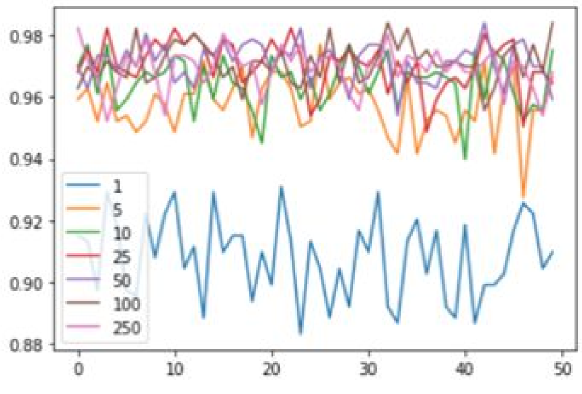


Fig 13. The accuracy of 50 random forests with varying amounts of estimators

4. Results + Analysis

4.1 Hybrid Features

All images used for testing were converted to the same resolution of training frames (910x1184px) before joint data was extracted. Different values for the FPS factor parameter were tested across the different movements. The FPS factor parameter is a customizable parameter I included to change the FPS of testing video. It will only process every n^{th} frame of a testing video, where n is the FPS factor. Frames are then processed with the pose estimation model where returned joint data is normalized to a 256x256 plane for lower floating-point arithmetic.

Once a testing video is broken into its frame-by-frame joint data, it is converted to its static and spatio-temporal features, which are then fit by a random forest model trained by the training data extracted above. Before moving to classifying movement overall, I tested the performance of the local static features, the local motion features, and then their combined hybrid feature performances as shown below:

Average Accuracy of 100 Random Forests			
angle	**Spatio**	**Static**	**Hybrid**
direct	68.11%	96.45%	96.92%
boundless	68.19%	96.50%	97.01%

Table 5. Average accuracies.

Although the performance of the spatio-temporal features on their own does not match that of the static features, the almost 70% accuracy on the spatio-temporal features hints that there may be patterns within motion itself. Everyday movement, while generally similar, will appear different for all partaking since everyone has a different physical form—my theory is that there is a pattern in and of the general movement. I chose to explore Olympic weightlifting because on the elite stage, lifters will achieve almost perfection in their form and movement. This ensures consistent movement across varying anatomical proportions to hopefully find the pattern in lifters consistent, but slightly varied, motion.

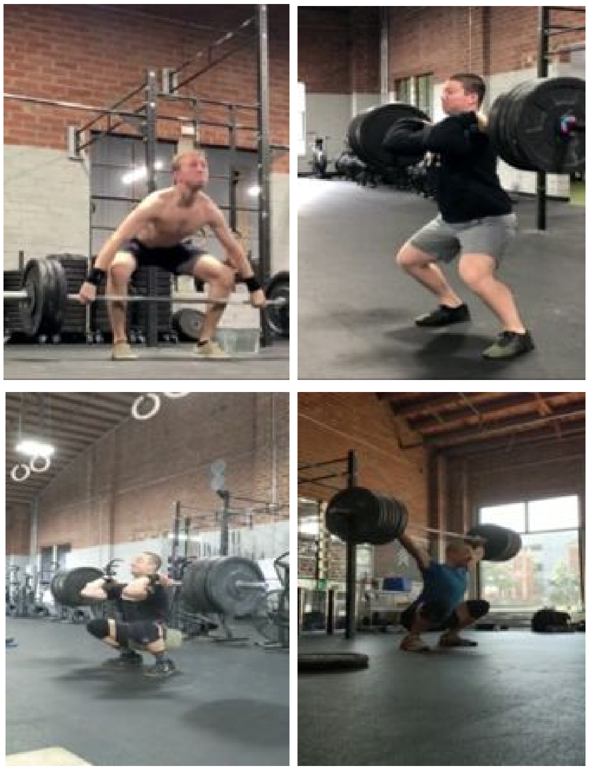


Fig 14. (top left) Ryan - power snatch
Fig 15. (top right) Kendall - power clean
Fig 16. (bot. left) Jan - clean
Fig 17. (bot. right) Jan - snatch

4.2 Classifying Movement

As for classifying movements as a whole, each frame of an unknown video is classified as a specific movement, the summation and proportion of the classifications then determines the overall movement classification. If three frames are classified as a deadlift, two as a clean, three as a front squat, one as a power clean, it can then be assumed that the movement was a clean. The model breaks down and classifies each frame of a lift into their most basic lift and uses the combination of these basic lifts to correctly classify a movement as a whole. If the same proportions as above were used, but there were zero front squats and four power cleans, the model would then assume a power clean was executed instead of a clean. This movement estimation follows the single frame estimation method as described by Google and Stanford researchers and

# estimators	1	3
Test 1 – Power Snatch, FPS factor = 4 (*fig 10.*)		
prediction	power snatch	power snatch
accuracy	97.52%	96.28%
# sn. deadlift	2	8
# snatch	3	10
# ohs	6	18
# power snatch	10	29
# deadlift	8	17
# clean	4	15
# front squat	0	0
# power clean	0	0
Test 2 – Power Clean, FPS factor = 4 (*fig 11.*)		
prediction	power clean	power clean
accuracy	96.28%	97.40%
# sn. deadlift	1	7
# snatch	1	1
# ohs	0	0
# power snatch	5	22
# deadlift	9	28
# clean	2	4
# front squat	0	1
# power clean	6	9
Test 3 – Clean, FPS factor = 16 (*fig 12.*)		
prediction	clean	clean
accuracy	96.99%	97.05%
# sn. deadlift	0	0
# snatch	0	0
# ohs	0	0
# power snatch	0	0
# deadlift	0	3
# clean	2	3
# front squat	3	9
# power clean	0	0
Test 4 – Snatch, FPS factor = 32 (*fig 13.*)		
prediction	clean	clean
accuracy	97.52%	97.46%
# sn. deadlift	0	0
# snatch	1	3
# ohs	1	2
# power snatch	0	1
# deadlift	0	0
# clean	0	2
# front squat	1	1
# power clean	0	0

Table 6. Testing output.

I found it to be quite accurate, even as I began to reduce the FPS before extracting joint data, *Table 6*.

I tested my model on four different videos, one of each of the classifiable movements, and each from a different recording angle and distance. I planned to run the test video data through multiple random forests trained on different splits of the training data, but through testing, the use of as little as one random forest was successful in all test cases—three random forests data are also shown. The FPS factor was set to 4, 4, 16, and 32, all still correctly classifying their respective movements. Increasing the factor not only decreased the time it took to process the test video through the pose estimation model, but it also lowered the number of misclassified features.

5. Discussion

For processing the natural world, data does not reach a much higher dimension than it does with video. It takes the already high dimensional nature of image processing and then adds thirty, sixty, even ten thousand images just to process and classify a single scene. Audio may be feature of the video that can be processed to provide context. I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld even found you could classify video with the aid of text-based classification from the script of movies [4]. Computer vision can combine a multitude of classification approaches to give computers real-time vision in the natural world. Part of this concept is the computer's ability to perceive motion. In this paper, I explored the basics of computer vision and video classification in the field of Olympic weightlifting by combining static and spatio-temporal features to classify movement in a high level implementation of a single frame video classifier.

6. Future

The next steps are to continue research in video classification and begin low level implementations of the concepts I learned and applied over the course of this term. I can begin applying many of the above concepts to the specific environment of Olympic weightlifting—rebuild a model where the static features are the equipment pieces in the frame, where the spatio-temporal features are key interest points detected in unsupervised environments, and most importantly, where all possible movements are classifiable.

Sources

[1] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale Video Classification with Convolutional Neural Networks. In *CVPR*, 2014.

[2] Z. Cao, T. Simon, S.E. Wei, and Y. Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *CVPR*, 2017.

[3] S.E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR* 2016.

[4] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[5] J. Liu, J. Luo, and M. Shah. Recognizing realistic action from videos “in the wild”. In *CVPR*, 2009.

[6] P. Dollár, V. Rabaud, G. Cotterll, and S. Belongie. Behavior Recognition via Sparse Spatio-Temporal Features. In *International Workshop on Visual Surveillance and Performaec Evaluation of Tracking and Surveillance*, 2005.

[7] I. Laptev. On space-time interest points. *IJVC, 64(2-3):107-123*, 2005.

[8] H. Wang, M.M Ullah, A. Klaser, I. Laptev, C. Schmid, et al. Evaluation of local spatio-temporal features for action recognition. In *BMVC*, 2009.

[9] F. Raudies. Optic flow. In *Scholarpedia, 8(7):30724*, 2013.

[10] V. Dumoulin and F. Visin. A guide to convolution arithmetic for deep learning. In *arXiv:1603.07285v2*, 2018.