IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 7, 2020, accepted July 22, 2020, date of publication August 4, 2020, date of current version August 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3014186

Applying Pose Estimation to Predict Amateur Golf Swing Performance Using Edge Processing

THEODORE T. KIM, MOHAMED A. ZOHDY, (Senior Member, IEEE), AND MICHAEL P. BARKER
Department of Electrical and Computer Engineering, Oakland University, Rochester, MI 48309, USA

Corresponding author: Theodore T. Kim (tkim2@oakland.edu)

ABSTRACT We present an analysis and approach for utilizing vision-based pose estimation to find key video frames in a full golf swing to assist in providing feedback for improvement. Using both still photos and videos, the proposed system discovers key moments in the golf swing to be evaluated and can identify metrics of the golfer such as posture, swing tempo, and swing consistency. These key frames can also predict the swing outcome by creating a path projection. The images and videos processed analyze the golf swing from a down the line perspective. For the computations, we utilize a low cost tensor processing unit (TPU) to run inference and data processing which set the performance baseline for the video capturing system. Hardware and pose estimation limitations and inaccuracies are identified and compensated for by using a Savitzky-Golay filter. This will allow for a markerless swing tracking analysis system in a low cost, small form factor.

INDEX TERMS Pose estimation, golf, video filtering, golf swing, driving range.

I. INTRODUCTION

Artificial intelligence (AI) and deep learning are being applied as advanced statistical tools in many disciplines including sports. As new golfers seek to progress as quickly as possible, AI is being adapted to help. Deep learning and AI once required data centers with multiple host computers' Central Processing Units (CPU)/Graphical Processing Units (GPUs) and neural network models connected through the cloud.

AI with sensors such as cameras can help identify movements and be used to correct techniques to improve consistency and performance. This work is to use computer vision in golf to identify swing techniques and predict the swing outcome.

With positive identification of key golf swing attributes, we intend to create a tool which would act as a golf assistant during practice. This tool would be used to accelerate feedback to the user, in a low cost, easy to use hardware system.

This approach would allow for any golfer to bring a small piece of equipment to the driving range to provide immediate feedback. It requires no potentially intrusive measuring devices to be attached to the user allowing for a more natural swing.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhiwei Gao.

II. RELATED WORK

The world of sports is abundant in such quantifiable elements, making it ideal for the use of artificial intelligence [1]. AI has multiple applications in sports: scouting/recruitment, training/performance analysis, and broadcasting and advertising. Many companies are focusing on embedded hardware such as connected golf clubs, apparel, and golf balls such as golf assistant hardware like Swingbyte, Arccos Golf, SensorIA, and Gen i1. Each of the companies mentioned utilize measurement devices that attach to the golf club or are built into the golf ball which utilize sensors to provide data from accelerometers or gyroscopes which are processed and provided to the customer as data. Sensors are added to the end of the grip, on the shaft, or inside golf balls which can be lost, damaged, or impact the aesthetics of the golf club.

The novelty of our system is that we do not attach to any sensors to the golf equipment to provide feedback on the golf swing. We capture videos from a specified position and have the flexibility to utilize the TPU development hardware or collect simply with the camera on a smartphone, to be

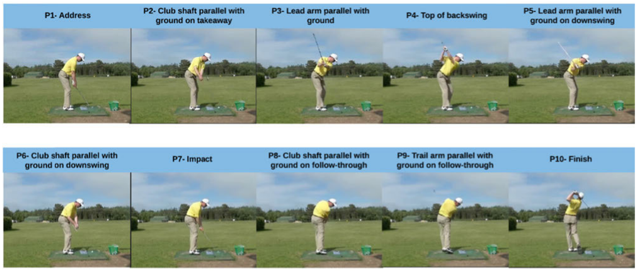

FIGURE 1. Golf swing P1-P10 positions [3].

analyzed and processed by our algorithm. Our focus was to provide swing analysis based on collection of multiple swings from the user and initial input to build the machine learning prediction inference customized to the customer.

Within the field of golf, we did not find many research papers on using AI to provide feedback. There were similar papers for other sports such as boxing, which helped provide some baseline, but the necessary data collection and analysis differs much like the sports themselves. Other projects performed similar pose studies but utilized different hardware than just a normal camera.

Due to the differences in data captured such as club attachments and golf ball measurements, there is no standardized data that can be used as training data for our model. Each company defines and utilizes artificial intelligence in varying ways. Arccos uses club selection and GPS information to develop an automatic shot tracking system.

The related work is broad and the lack of accessible training data for an individualistic sport is an issue in developing deep learning for golf.

III. FEATURES

To establish a baseline for capturing the golf swing, the first task was to determine which features from a golfer were recognizable compared to non-golfing movements or positions. The process of identification could lead to training tools such as video analysis with statistical success, golf swing performance improvement metrics, and automated swing correction recommendations. The method of recognizing key attributes in a golf swing is nontrivial, as identification requires understanding multiple poses of a person as it relates to the swing motion process.

The golf swing for this project was separated into 10 positions:

- P1- Address
- P2- Club shaft parallel with ground on takeaway
- P3- Lead arm parallel with ground
- P4- Top of backswing
- P5- Lead arm parallel with ground on downswing
- P6- Club shaft parallel with ground on downswing
- P7- Impact
- P8- Club shaft parallel with ground on follow-through
- P9- Trail arm parallel with ground on follow-through
- P10- Finish

FIGURE 2. Depiction of golf swing path- On Plane, Outside In, and Inside Out [24].

FIGURE 3. Professional golfers Dustin Johnson, Hideki Matsuyama, and Soren Kjeldsen swing positions P1-P5 [6]–[8].

These positions are considered checkpoints in the swing to determine if your body and club are on plane. As a baseline, this camera angle is typically called "down the line" [25] or "down range" and the swing positions will be used with the pose estimation algorithm.

There are key positions during the golf swing that golf coaches focus on when providing guidance. The most important being P1- Address, P4- Top of backswing, and P7- Impact. Starting with the posture at address, the golfer's position at set-up and their ability to maintain proper posture throughout the swing is one of the key parameters. As part of the backswing phase, the body rotation and movements have checkpoints of being on-plane. These points are P2- Club shaft parallel with ground on takeaway and P3- Lead arm parallel with ground, where the club is aligned with the target then the shoulders respectively. Proceeding the top of the backswing, the transition phase of P5- Lead arm parallel with ground on downswing and P6- Club shaft parallel with ground on downswing, carry critical information on swing path. Figure 4 shows the path between P5 and P6 which is dependent on the transition from P4.

To predict the outcome of the golf swing, with pose estimation, key positions and body points will be used as parameters to determine On Plane (Straight), Outside In (Slice-Right), and Inside Out (Hook- Left).

Also, in the field of computer vision, human pose estimation has seen an increase in research in terms of methods and application. Pose estimation is classified into two categories: Single-person pose estimation (SPPE) and multi-person pose estimation (MPPE). For our application, we will be using the MPPE model. This model uses a bottom-up approach

FIGURE 4. Hideki Matsuyama posture at P1- Address, 78° degrees [7].

in which it associates pixel-level predictions to each object instance. The idea behind the bottom up approach is to detect body parts first and then groups them to form the body. The pose estimation model takes an input in the form of a Red-Green-Blue (RGB) image. That input then passes through the model which consists of a Convolutional Neural Network (CNN) that predicts: heatmaps, offsets for the X/Y in terms of short-range and mid range offsets. The parameters are then passed through a person pose decoding algorithm to detect human poses.

The pose estimation algorithm uses a body model with N-number of part identifiers which represent the skeleton model. A heatmap is produced with 17 channels, one for each body part, along with offsets which are 34 channels (part displacements in x and y). The 17 part identifiers define the main body joints: right shoulder, right ear, right knee, right wrist, left ear, right elbow, nose, left wrist, left knee, left shoulder, right eye, right ankle, left eye, left hip, left ankle, right hip, left elbow. The parts are associated with each other and passed into the models as parameters and this forms the base skeleton of the pose.

The short range and mid range offsets are used together to refine the location of key body parts, find connected pairs such as the Right Elbow to Right Shoulder, and then use the locations and the heatmaps to generate part detections. The keypoints are detected along with an instance level detection score.

To demonstrate pose estimation could be used in the field, we captured data of ourselves at a golf practice facility using a smartphone camera mounted to a tripod that was triggered manually from a smartwatch. The set up was to model how users would use a smartphone or similar hardware device that could be mounted to the golfer's bag.

The tripod was set to approximately 32 inches which is roughly the height of most golf bags. We set the distance of the camera to 10-12 feet behind the golfer to stay behind the safety line and not impede walkway traffic. This allowed us to have a full field of view of the golfer's appendages through the full motion of the golf swing. The camera was also set directly behind the golfer so that the down the line angle could be captured.

With the camera distance, height, and direction set, there would be limitations in predicting all 10 positions due to occlusions of key body parts however we are able to demonstrate with positions up to impact, P1-P7, that can be used to predict swing outcomes. All swing positions from P1-P10 can be detected even with the occlusions and visibility of only the sagittal plane due to key frame selection created as part of the data processing model.

IV. POSE ESTIMATION

A. METHODOLOGY OVERVIEW

The general process for collecting the data and analyzing the results are:

1. Video data of a swing from a down range camera position is collected
2. Video is converted to individual frames
3. PoseNet machine learning model is run on the frames on the TPU
4. Frame-by-frame pose positions are collected
5. Data is smoothed using a Savitzky-Golay filter to help minimize sudden changes caused by PoseNet inaccuracies
6. Critical points in the swing and their associated frames are found using the feature table below

Swing Position	Defining Feature
P1	Starting Frame
P2	Wrists Intersect Hips
P3	Wrists Intersect Right Shoulder
P4	Wrist Global Minimum
P5	Wrists Intersect Right Shoulder
P6	Wrists Intersect Hips
P7	Wrist Y Position Global Maximum
P8 - P10	Not Analyzed Currently

B. FRAME ANALYSIS

We sampled seven professional golfers of different swing types at random to find commonality. The goal was to identify if there were common attributes between the varying swings that could be used to label the golf swing. TensorFlow's PoseNet, was used as a baseline model to capture the pose estimation. For this stage we only used pictures and videos

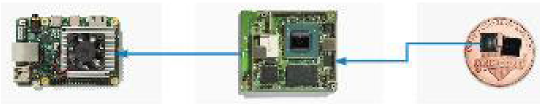

FIGURE 5. Coral Dev Board, SOM, Google TPU.

with camera angles behind the golfer or what is called "down the line".

For each pose, the picture was processed using the pose estimation model. The data output included an overall Pose Score, 17 Part Identifiers with Part Scores and X/Y coordinates within the frame. The scores were from 0 to 1 with 1 being the most confident. From the frames, we can see that each position P1-P10 from down the line, created some challenges for the estimation model. Due to the down the line camera angle, most part IDs that are associated with the left side of the body had scores significantly lower than the right because they are partially or completely obscured.

Given the data collected, the challenge that we found is the golf swing is unique to each golfer even at the professional level. There are key club sequences positions that can be used as boundaries however at this point, we found the data inconclusive to use as swing success identifiers. The data did give some insight into metrics that can be collected and used as analytics to a professional golf swing. As an example, at P1- Address, we used the right hip coordinates along with the right shoulder and right wrist to calculate the posture at P1.

The angle at P1 ranges from 78° degrees to 105° degrees depending on the golfer with an average of 90° degrees. As more samples are collected, the data analytics could help determine optimal or trends in golf swings that can be used as a feedback to the user. This approach helped us understand the data available and trends between golfers, however more information in the swing motion would be needed to help understand what forms and components of the swing need to be measured and what could be used as a baseline to help someone improve their swing.

C. SOFTWARE PROCESSING

The second task used as part of this research was to run the pose estimation model on hardware to determine the performance limitations of the hardware and to define constraints of the system. We started by selecting the Coral Dev Board by Google due to the hardware performance and I/O flexibility, software framework for TensorFlow Lite, and production integration capability of the System on Module (SOM).

Processing on the board is commanded through various Python libraries provided with the board. These libraries handle the loading of a TensorFlow Lite (TFlite) model into the Tensor Processing Unit (TPU) and running inference on the model to produce outputs. Other Python libraries can be loaded as needed assuming that they are compatible with the TPU board. Depending on the model used, it can be used to perform post-processing on data or to actively process live data like from a camera.

To begin, we used a publicly available repository called project-posenet that was created for the TPU by Google. Included in the repository was a version of PoseNet that was compiled into a TensorFlow lite model to run on the TPU. The project also featured a Python script that was able to read in static images and find multiple poses in it along with one to read in a direct camera feed and draw the pose skeletons live. These were used as a starting point to examine running the PoseNet model on the Google Coral board. The repository contains 3 different tflite compilations of the PoseNet model based on the mobilenet v1 network. The different tflite models accept different resolution image inputs, depending on how high or low the image resolution is. The output is directly the keypoints found by the model so there are no intermediate steps, like the heatmaps, revealed to the user's software. The model still contains the heatmaps and trained by the same data as other PoseNet systems but includes a post processing network that will handle the heatmap for the user.

Experimentation with the board began by using the camera to look at how quickly the TPU would be able to process a live camera feed with the model in the repository. Two different models were used based that were included in the project-posenet repository, one that read in images at 641×481 and another at 1281×781. Running the lower resolution model indicated an approximate 30 frames per second (FPS) limitation on the Coral development board. The higher resolution model saw an expectedly lower performance of around 20 FPS. We used this as a baseline to determine if real-time estimations could be achieved with Pose estimate positions and accuracies for P1-P10 swing identification could be run real-time. To start, we chose to examine how the information would appear at 30 FPS, which is more ideal.

Examining the example swings and calculating approximate time a normal golf swing takes showed that 30 FPS would be acceptable in most cases. The backswing contains the most key points that indicate whether a swing will be successful or not. These positions would be a good starting point for someone to focus on when improving their golf swing. Enough data is captured on the downswing to analyze some features, but the model would need to run faster to fully analyze the rapid movement. Generally, a swing can be complete in 3 to 4 seconds which only translates to up to 240 sample points with the distribution heavily weighted towards the slow backswing, creating a higher data fidelity and more opportunities to capture key swing positions.

To avoid processing delays on the TPU and to allow us to use existing videos as samples, we chose to post-process the data by pulling frames from captured videos. Using more Python scripts and a computer vision module, every frame of the videos was separated out into individual jpg files, loaded onto the TPU, and then run through the PoseNet model. After the inference is performed, the data is recorded into a comma separated values type file for further analysis. Later, the data can be directly manipulated on the TPU board to produce direct conclusions for the swing that is being analyzed.

This would be handled by another model, for example, or using the data analysis scripts detailed below.

D. POSITION ANALYSIS

The data for frame by frame analysis was initially performed on a 30 FPS video found of a professional golfer. The video is approximately 3 seconds long which was processed into an 86 frame data set. With the data collected processed through the pose estimation model, we focused on key part identifiers to determine if the swing poses could be selected. Initially we plotted all attributes' X and Y values against the frame count to see what the standout features of the swing were. This graph proved to be a massive amount of data.

From this graph, we started with the Y-axis part data for the left hip, right hip, left wrist, right wrist, and left shoulder to compare frame by frame. The overall movement of these parts pointed to some key attributes that linked back to the different positions of P1-P10. Such as when the wrists cross the hips, the first intersection links to the P2 position. The chart below breaks the example video we used as a baseline into the positions of interest during the swing.

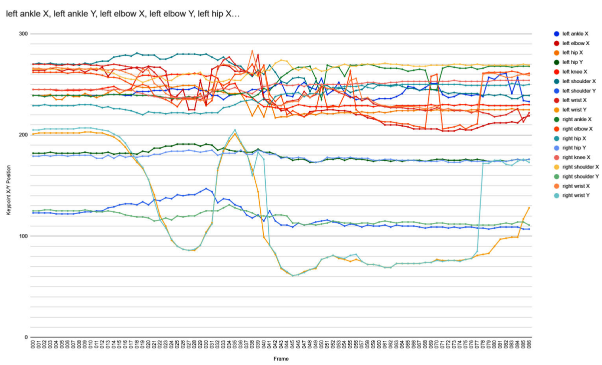

FIGURE 6. Shane Franklin pose estimation output-keypoints X/Y.

Using this information, we constructed a script in Python to locate important features of these body parts' motions and then filter them down into the key frames. To start, the data has a level of inconsistency and uncertainty that comes from the PoseNet model's ability to accurately find body parts. To combat some of the sudden shifts, we used a Savitzky-Golay filter. This assisted in filtering out some of the larger sudden jumps in the data that we saw (this can be easily seen in Figure 6) that was a product of the inaccuracies of the model.

After filtering, we find out the frames at which certain groups of values cross others. For example, because we're looking at the subject from downrange, the wrists will generally stay close to each other with slight differences in positions assuming that the pose estimation is confident for them. So, when both the wrists cross both sides of the hips, we can say with good confidence that the subject is at position P2.

The Python script performs a sort of "group intersection" function where it checks when one or more sets of data

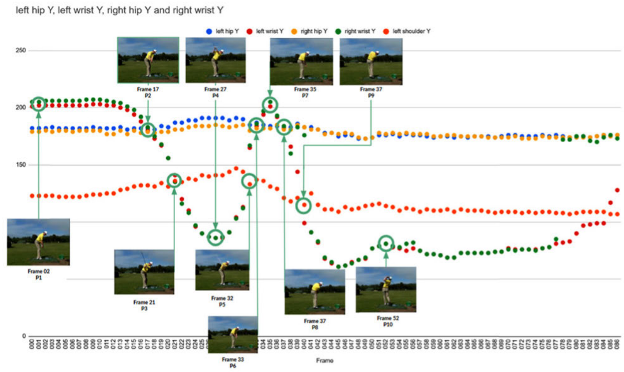

FIGURE 7. Five y-axis parameters used for key frame selection [3].

intersects one or more other sets such as when each of the wrists' Y positions intersects each of the hips' Y positions and returns all the unique frames that are crossed. The frame of the intersection is chosen by rounding the intersection value (which is rarely a whole number). Next, sequential frames are filtered down further by checking to see the frequency in which the frames occur across all possible intersections and picking the one that has occurred the most when performing the group intersection calculation. If the two frames have been found the same number of times, the earlier frame is chosen. This decision was based on the observations that the later frame of the intersection tends to be after the point of interest.

Next, we search for the minimums and maximums in the wrists to determine some of the key positions such as P4 at the top of the swing. The general shape of a standard swing would make finding the critical points simple, but the pose estimation confidence comes into play. Positions that seem constant visually may jump around if the pose estimation is low confidence. Our data showed that, in general, the estimation was close enough that with some simple manipulation, like applying a low pass filter to the data and looking at the overall features of the graph, can help remove and filter out local minima and maxima that are incorrect.

When the wrists intersect the hip, all the minima and maxima before that point can be assumed to be caused by oscillations in the positions calculated from the model. After that, the first large minimum which represents the top of swing (P4) can be found by checking the frames after the wrists and shoulders intersect. This point can be difficult to calculate because the time a person spends at and around P4 can vary depending on how slow or fast their back swing is, and the noise of the pose estimation may create additional minima. Some of these can be filtered out while others can be found with more confidence since both wrists' positions are being used. If one position is noisy (i.e. it's obscured), the other may be a cleaner signal that will give a more accurate frame.

Additionally, some of the inherent uncertainty of the PoseNet model required for the data to be filtered to help remove sudden changes in the data.

E. SWING PREDICTION

From the Position Analysis work, we generated what we are calling the Key Frame Selector (KFS), this identifies frames of interest based on the keypoints we selected. The focus for the swing prediction resides in the information that can be gathered from the keypoints during swing positions P1-P7. Our methodology started with gathering data from the driving range with me and a few other test subjects. We collected 72 video samples over two days. This included documenting swing outcome and club selection along with measured data from Trackman such as ball speed and launch angle.

From the collected samples, we created a subset of data in a simplified radar chart. The data shows the P1-P7 right wrist x axis data plotted as a smooth plot. We observed path indicators which align with hook versus fade swing outcomes. Figure 8 shows data with the center of the plot being towards the left of the frame with the outer bounds showing wrist movement towards the right of the picture frame. It was observed that, with a hook outcome swing, as the swing moved towards P4 (top of swing), the wrist showed a trajectory farther inside compared to the others. As the swing progresses through P5-P7, the inside nature of the swing can be seen with the outcome being a hook.

Similar to the hook swing outcome, the fade swing shows similar attributes from P2 - P4 where the wrist x-axis shows a path father outside. At the transition from P4 to P5, you can see the wrists have already moved farther to the right which can be observed as an outside-in swing path. As an example, the figure below shows the swings compared at P5. As you can see, it can be difficult to observe the difference visually however the plot can show distinct differences. The data shows that the swing path plotted, and outcome capture are proportional and can be used as part of the swing prediction model. With the proper amount of data, a good swing from one golfer could potentially be used as a baseline for other golfers.

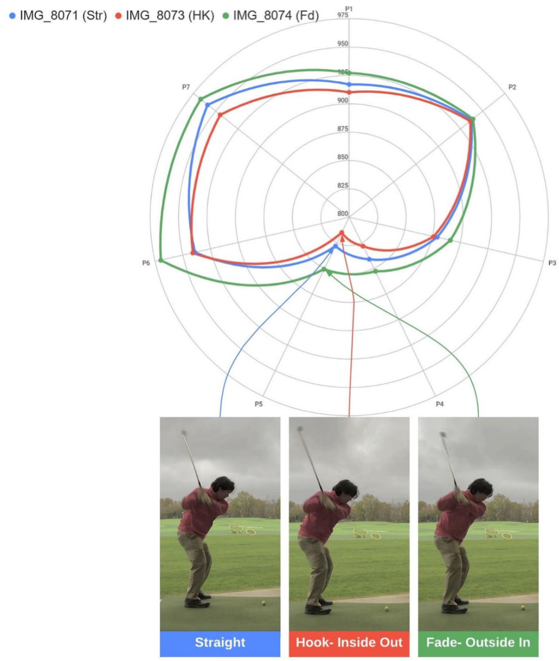

FIGURE 8. KFS Radar plot: three swing outcomes.

F. SWING DATASETS

Along with the videos our team captured, we analyzed over 200 samples from over 20 golfers. The videos analyzed came from YouTube (Figure 8) and golf websites such as Golf Digest. We selected videos that were professionally captured and average golfers with varying lighting conditions and camera positions. The varying methods of capture resulted in swing pose accuracies from 20% to 81.1%.

Estimating how many datasets would be required to build and train a machine model to analyze the swings and predict the outcome is currently unclear. The pose accuracies in the videos proved to be highly inconsistent in low lighting at times lending to some datasets being difficult to pull useful information from. The conditions in which the records were made greatly affected the pose accuracies. Videos collected from YouTube also had some bias problems where the swings shown were only the good swings. These in combination made it difficult to find a collection of swings that both had the proper recording conditions to get good pose estimation accuracies and contain good and bad swings to not over train a machine learning model (i.e. it only has bad swings to learn from).

Even though the collection of datasets resulted in poor results, there were some key takeaways to highlight. The datasets we analyzed which had high pose score accuracy resulted in KFS detections of fade and draw determinations which were of higher fidelity outcomes than what we anticipated could be results from our model. For this example, the average pose score was 71.2%. If we can improve the data collection method and hardware to achieve such a high accuracy, this tool could be used to improve even low handicap golfer swings by indicating swing tendencies. Figure 9 shows the outcome of the KFS for Video 1. All of the driver swing outcomes were straight but with varying path starts and trajectories. With the higher accuracy scores, we are able to dissect swing by swing, key positions from P1 to P7 that the golfer could utilize from our data.

The datasets which resulted in poor accuracies gave us additional sensitivities that need to be understood and factored into the machine learning model/user interface. Additional work in the environment assessment of the data capture and the posenet outcomes need to be researched further in our next steps.

Our model can reliably utilize and predict swing predictions with pose scores as low as 37.2%. Any scores higher than this simplify the prediction and reduce the need for filtering methods such as the Savitzky-Golay filter or more advanced techniques to handle the issues we encountered like hip positions flipping sometimes in the model.

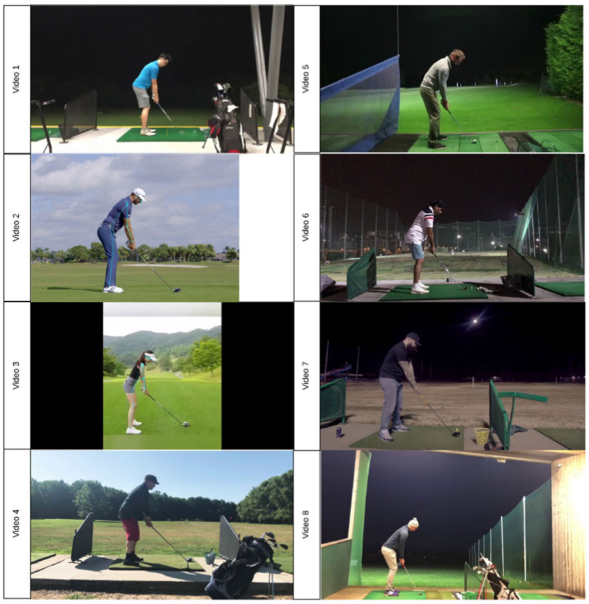

FIGURE 9. Random YouTube golfer selection.

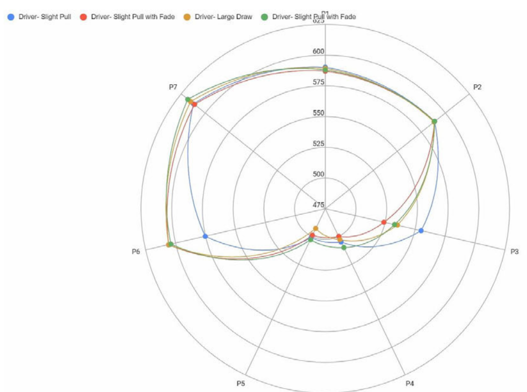

FIGURE 10. YouTube Video 1 Golfer- KFS Radar plot of straight driver shots analyzed by swing path and swing trajectory.

V. NEXT STEPS

Our next steps are to improve the swing predictor by using a deep learning network. Though we had some success with python scripts in processing the data, we were not able to predict styles without manual feedback. We did not have enough labeled data yet to build a trusted implementation for the predictor. Our plan is to standardize the collection of data that can be used to create a useful tool through continued testing.

Datasets collected and sampled had varying accuracies which are being analyzed as sensitivity to the model that is needed to improve the KFS model. Feedback from the sensitivity and further understanding of the impacts to keypoint detection in the pose estimation would reduce the manual processing of the data.

The transition of the swing from P4-P7 is critical to the outcome of the swing. This has been noted that P6 contains critical information that needs further analysis past the wrist position in relation to the video frame. The need for inclusion of more key points in relation to a square path is pertinent to improve prediction.

VI. CONCLUSION

This work initially focused on single frame classification of a golf swing which has evolved into a pattern recognition of movements or poses. The experiment resulted in various methods being attempted to solve the problem one step at a time. We first identified constraints of the system by running PoseNet directly on hardware. We then collected samples of photos from the down the line shot that could be used to classify activities. Then we created a Key Frame Selector model to highlight frames of interest. We created some data through the collection of videos at a driving range which highlighted the need for data filtering and a better understanding of camera hardware. Then finally a KFS plotter of keypoints to predict swing outcomes.

REFERENCES

[1] N. Joshi. (Mar. 15, 2019). *Here's How AI Will Change the World of Sports Forbes*. [Online]. Available: https://www.forbes.com/sites/cognitiveworld/2019/03/15/heres-how-ai-will-change-the-world-of-sports/#5da880b0556b

[2] B. Alberstadt, "Getting a grip on golf's $2.6 billion equipment industry," Nat. Golf Found., Jupiter, FL, USA, Tech. Rep., Jan. 2018. [Online]. Available: https://www.thengfq.com/2018/01/getting-a-grip-on-golfs-2-6-billion-equipment-industry/

[3] S. Franklin. (May 2015). Shane Franklin swing normal speed. YouTube. [Online]. Available: https://www.youtube.com/watch?v=Zxd0UwxbKV8

[4] R. Turner. Which one best depicts you. Highgate Golf Club. Accessed: 2019. [Online]. Available: http://www.robin-turner.co.uk/featuredetail.aspx?articleid=aff8287a-3317-42af-9fe1-15d2f2eac58d

[5] D. Oved. (May 2018). *Real-Time Human Pose Estimation in the Browser With TensorFlow.js Medium*. [Online]. Available: https://medium.com/tensorflow/real-time-human-pose-estimation-in-the-browser-with-tensorflow-js-7dd0bc881cd5

[6] M. Rudy. *Swing Sequence: Dustin Johnson Golf Digest*. [Online]. Available: https://www.golfdigest.com/gallery/swing-sequence-dustin-johnson

[7] R. Kaspriske. *Swing Sequence: Hideki Matsuyama Golf Digest*. Accessed: 2019. [Online]. Available: https://www.golfdigest.com/gallery/hideki-matsuyama-photos

[8] R. Schiffman. *Swing Sequence: Soren Kjeldsen Golf Digest*. [Online]. Available: https://www.golfdigest.com/gallery/swing-sequence-soren-kjeldsen

[9] L. Jang. (Aug. 2017). 20170817 golf practice range swing is back!!! YouTube. [Online]. Available: https://www.youtube.com/watch?v=dTPRHbfYwHU

[10] A. Mears, E. Wallace, J. Roberts, and P. W. Kong. Golf coaches' perceptions of key technical swing parameters compared to biomechanical literature. Research Gate. Accessed: 2019. [Online]. Available: https://www.researchgate.net/publication/283165996

[11] M. Tyka. (Sep. 8, 2019). *Google-Coral/Project-Posenet*. [Online]. Available: https://github.com/google-coral/project-posenet

[12] R. Wightman. (Sep. 8, 2019). *Rwightman/Posenet-Python*. [Online]. Available: https://github.com/rwightman/posenet-python

[13] M. Rudy. (Sep. 27, 2017). *Swing Sequence: Brooks Koepka Golf Digest*. [Online]. Available: https://www.golfdigest.com/gallery/swing-sequence-brooks-koepka

[14] R. Kaspriske. (Oct. 23, 2018). *Cameron Champ's Power Move Golf Digest*. [Online]. Available: https://www.golfdigest.com/story/cameron-champs-power-move

[15] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least square procedure," *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964, doi: 10.1021/ac60214a047.

[16] J. Lue, K. Ying, and J. Bai, "Savitzky–Golay smoothing and differentiation filter for even number data," *Signal Process.*, vol. 85, no. 7, pp. 1429–1434, 2005, doi: 10.1016/j.sigpro.2005.02.002.

[17] H. Hassanpour, "A time–frequency approach for noise reduction," *Digit. Signal Process.*, vol. 18, no. 5, pp. 728–738, 2008, doi: 10.1016/j.dsp.2007.09.014.

[18] J. Seo, H. Ma, and T. Saha, "On Savitzky–Golay filtering for online condition monitoring of transformer on-load tap changer," *IEEE Trans. Power Del.*, vol. 33, no. 4, pp. 1689–1698, Sep. 2017.

[19] J. Parker. *The 10 Golf Swing Positions of Mac O Grady James Parker Golf*. Accessed: 2019. [Online]. Available: https://www.jamesparkergolf.com/10-golf-swing-positions-mac-o-grady/

[20] W. Shaw. (May 12, 2019). Too many swing thoughts—What is the answer? GolfInsiderUK. [Online]. Available: https://golfinsideruk.com/golf-swing-thoughts/

[21] K. Sato. (Aug. 30, 2018). What makes TPUs fine-tuned for deep learning? Google Cloud. [Online]. Available: https://cloud.google.com/blog/products/ai-machine-learning/what-makes-tpus-fine-tuned-for-deep-learning

[22] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "PersonLab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model," in *Proc. ECCV*, 2018, pp. 269–286.

[23] P. Dar, "'AI Guardman'—A machine learning application that uses pose estimation to detect shoplifters," *Anal. Vidhya*, Jun. 2018.

[24] R. Turner. Which one best depicts you. Highgate Golf Club. Accessed: 2019. [Online]. Available: http://www.robin-turner.co.uk/featuredetail.aspx?articleid=aff8287a-3317-42af-9fe1-15d2f2eac58d

[25] M. Wheeler. Use proper camera alignment when filming your swing. GolfWRX. Accessed: 2019. [Online]. Available: http://www.golfwrx.com/160889/use-the-proper-proper-camera-alignment-when-filming-your-swing/?

THEODORE T. KIM received the B.S. degree in electrical engineering from Kettering University, MI, USA, in 2006, and the M.S. degree in electrical engineering from the University of Michigan, MI, in 2008. He is currently pursuing the Ph.D. degree in electrical engineering with Oakland University, MI.

From 2002 to 2007, he was a Product Design Engineer with Visteon Corporation. From 2008 to 2018, he had multiple development engineer positions in chassis electronics and high-voltage battery electronics with General Motors LLC. Since 2018, he has been the Technical Program Engineering Manager for autonomous technologies. His research interests include the development of engine and powertrain electronic control modules, radio frequency communication systems, high-density/low-cost battery management systems, machine learning applications, and ASIC hardware development.

MOHAMED A. ZOHDY (Senior Member, IEEE) was born in Cairo, Egypt. He received the B.Sc. degree in electrical engineering from Cairo University, Egypt, in 1968, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Waterloo, Canada, in 1974 and 1977, respectively.

He is currently a Professor with Oakland University, MI, USA. His research interests include advanced control and estimation, intelligent pattern information processing, neural, fuzzy, evolutionary systems, chaos control, smart simulation, and hybrid systems. His research contracts with government and industry; recent seed funds on fuel cell modeling and control for transportation, hold considerable promise for improving vehicle energy supply, also FCA Powertrain controls, Lear Power Electronics, Kia motors, NSF, and USAID.

Dr. Zohdy has received multiple honors and awards, including but not limited to a Professional Engineer in charge of foreign Science Degree accreditation, Eligible for Fellow, IEEE, and SCS, Active in EKN Society, Sigma Si Society, a Co-Editor and the Conference Chair ACC, a Distinguished Counselor, and Awarded the Millennium Medal from IEEE.

MICHAEL P. BARKER received the B.S. degree in electrical engineering and applied mathematics from Kettering University, MI, USA, in 2009, and the M.S. degree in electrical engineering from Oakland University, MI, in 2015.

In 2009, he has worked in various positions within General Motors LLC working with high-voltage propulsion batteries, including testing and developing battery monitoring electronic systems and diagnostics. His research interests include machine learning, control systems, and electronics.