Computer Vision: Actions speak louder
Initial Document

James P Norman
976690

Submitted to Swansea University in fulfilment
of the requirements for the Degree of Bachelor of Science

Department of Computer Science
Swansea University

01/11/2020

Abstract

Can a refined Fencing system be produced to mitigate argument, game stalling and point dispute using a computer vision model and be available to the public within budget. Having this machine learning referee to supplement pre-existing judges helps form a non-biased second opinion and can aid both players, judges and spectators.

Being a fast and high entry level sport there has been only a few specialised investigations into different categories of fencing my project will aim to tackle the épée sub category and perfect a system that can referee this class of sword and help coach those using it. This will be valuable data going forward as VAR and computer vision based systems become increasingly ubiquitous and affordable with easy to access community tools and hardware within the sports.

Using Object Detection and pose estimation this system will be able to differentiate hits from misses and give an accuracy score, a combined effort of both areas does not exist for épée and in a sport with constant debate over points it has become ever relevant to have another voice backed up with data.

Table of contents

1 Introduction

1.1 Motivation

Competitive sport no matter how organised and formatted has always had subjective aspects, in football; contributing to van Dijk's v Pitchford's in the Merseyside Derby[2], a last-minute decision on the touch line and in fencing a tie breaking hit, could we get this system more accurate and can computers augment video refereeing?.
Refereeing and coaching has always conventionally done by humans however there has been an increasing need in a more consistent and un-biased system VAR(Video assisted Refereeing) has become ever present in highly competitive environments, and is already a main stay of FIFA since 2018[3]. Could a computer vision enhanced VAR system mean the difference in a win or loss scenario to fencing? better yet guide the player to conform to game rules and improve quality of play.
I would like to develop a joint Referee/coaching system for Fencing épée. A similar system exists for the sabre class of sword called 'Saber-Box'(Sect. 1.3) which implements other game rules of that particular category, I would like to refine and produce a system similar but purposed for épée exclusively. providing a system like this would decrease the equipment cost of an already expensive sport and supplement coaching/refereeing by offering tangible data about a player's position, pose and movement. The system would act similarly to VAR systems but augmented using computer vision; specifically, 'Object detection' and 'Pose estimation'. This will supplement decision making, present visual feedback for coaching and provide more evidence for a referees decision, hopefully elevating some contention in the fencing community between the decision by the referee, the player and if not replaced the electronic contact system.
The difficulty in developing a system like this stems from the precision and speed of the sport(see sect.4 Appendix). A computer vision approach would have to be thoroughly consolidated to process the ruleset of such a fast acting sport, however épée was chosen as the most pronounced lending itself to be the test bed for my computer vision system, along with my personal experience of the sport and rules it is an appropriate facet to test and develop my system.

1.2 Aims

To consolidate the above into a series of aims would entail the following:

• To generate a Computer vision model using pose estimation
This is the main component of my system and will be the visual data behind the coaching and development aspect of my software it will involve mapping points to the body and generating bones between them to form a simplistic view of the kinematics of the human body during the activity.

• Further integrate object detection to better determine sword distance to contribute towards an accuracy and hit detection rating.
To generate a system which can better recognise hits the project will use precise object detection it is important that this feature functions appropriately as it will make up the majority of the refereeing module of the program and contribute to the accuracy score of the model

• Display a 'real-time' feed of both former aims.
A real-time system is vital to refereeing a match so it would be preferable if both mentioned points were recorded in real-time and have the capability to later be played back for further analysis. The real-time component must be capable of overlaying both object detection and pose estimation at an appropriate framerate which will be able to support most standard in-built laptop webcams and external higher framerate cameras

• Better develop and integrate a rating system and hit detection using both aforementioned data.
The object detection component of hit detection could be further refined to produce an accuracy rating that would be able to convey the probability that a hit landed during the current bout. The data produced from this would be valuable to further refine the system and also aid those playing and refereeing the game.

I believe these initial aims are achievable although problems such as "determine sword distance to contribute towards an accuracy and hit detection rating." Will have to be somewhat improvised with further development of my system as there is a lot of potential variance in the hardware, player and technique; these factors will have to be accounted for.

1.3 Related work

A similar system exists in a public repository which has been produced for the Sabre ('Saber_Box')[4] category of fencing, developing it's own 'ROW Right of Way'[5] & 'Sabre object detection' model[6] to act as a "virtual directing aid."[4] This system would function in a similar manner to my system, épée fencing does not utilize the ROW rule; points are awarded solely awarded from hits. My system will differ from 'Saber_Box' as it will function more like a training and refereeing tool giving live feedback and comparison to determine the

score with assistance from a referee.
A generalised system for pose estimation across multiple sports exists called ST37[7], it uses similar pose estimation techniques such as identifying pivots or joints on the body[8] using a ball model. I would like to further improve this by adapting bones across these points to better understand the movement and limb position of both players.
Detection of the point of the sword is an area that an object detection/pose estimation approach may find hard to determine a conclusive hit other proposed systems such as the recent 'real-time sword trajectory'[9] produce visual information through an "infrared (IR) image by detecting IR light reflected from retroreflective tape placed on the tip of the sword"[9]Over the All Japan Fencing Championships and evaluating 450 unique images "containing fast-moving swords and noise objects for this evaluation so that detection of sword tips from the dataset would be difficult." I carry this concern forward into my project as similar to their accuracy concern it may also interfere with the detection of the tip of the sword in my model.

2 Research for my Fencing System

2.1 A system for coaching

Developing my system for coaching applications it is important to recognise information and features that may be more useful to coaching and overall perfecting play. However fast paced games can be difficult to analyse and breakdown in the moment "involves fast and accurate motion that is not only challenging for competitors to master, but can be difficult for coaches and trainers to analyse"[11] it has been discussed that a computer vision approach to coaching and coaching aids may be effective, 'Microsoft's Xbox Kinect ™' has inspired sports based applications from its technology lending itself to movement based sports, its low cost has made the technology easily accessible[12] notably 'Nike+ Kinect training' is a software for the 'Xbox 360' that is an example of coaching applications utilizing computer vision. Similarly for my project I will be using a standard webcam to capture and generate 'Posenet'[13] based models and further adapting pose estimation to create coaching features; such as using Multi-Person Pose estimation to track and maintain a clear map of bones and joint throughout the duration of each bout. Using this approach could aid coaches in player communication as there is a break between each bout for fencers to prepare again, this valuable down time could be used to advise play and change tactics. Sporting Cameras in football and rugby use this technique during game down time and halftime BBC and other sports broadcasts use specialised cameras integrated with computer vision the "STATS SportVU"[14] is a coaching/training camera used by "all 30 NBA teams" it Is used to track the ball and provide player analytics. Camera integrated computer vision systems like this are used in professional applications however no valuable computer vision integrated camera specialized for professional fencing exists. Cameras like this are not very accessible due to the cost and setup, my system will act as a low budget coaching system exclusively for épée.

A coaching system as proposed for épée should have the following coaching features that would be valuable:

2.1.a -Pose tracking and referencing

By using pose estimation to track the positions of body parts vital to technique in fencing; left and right foot, Dominant hand, torso, shoulders and head, in real time this data can be used to compare to proper/expected technique that should be applied in that game scenario. A coach then would be able to advise the fencer between bouts on aspects which do not match an ideal reference model or discard the computer vision assistance and advise the fencer exclusively.

2.a -Pose recording and comparison

Another application of pose estimation for coaching would involve removing extracting the generated overlay and using the recorded skeleton of precise techniques, coaches can use this to focus on the kinematics of the body to better perfect technique in individual training or practice, actions such parrying(Prime, Seconde, Sixte, Septime, Octave, Nuiveme)[15], lunges, fleche(quickly pacing forward) and other footwork based movements

2.b -Sword Tracking

In fencing it is important to be able to track your own and the opponents sword, a vital part of the sport is the sword play and technique. Using object detection and pose estimation of the dominant hand could offer both coaches and players extra insight or focus into the opponent's technique to better pre-empt tactics in the next bout.

2.c -Frame by Frame recording/analysis

Making use of real-time computer vision and recording it would be beneficial to coaches as well as referees being able to see the moment a hit landed often reveals where a defence has failed, offering another learning opportunity where it would often be lost due to the speed of the sport.

2.1 Computer Vision: PoseNet and Object Detection

For the complete system to operate as intended the conclusion is that both Pose Estimation and Object Detection are needed in order to produce accurate and understandable data to both coaches and players, the usefulness of each half of the project are detailed above in Sect. 2.a – 2.c

2.1.a Posenet for fencing

This project is using 'Tensorflow'[16] library and community resources in order to generate and appropriate model for fencing, using 'Posenet'[17] to generate a basic model of the body is an effective way of tracking points on the body. NPM components of Posenet being able to track specific limbs on the body makes an ideal base for developing a model of a faster more intricate figure in motion.

ResNet50 is used for more accurate but slower high detail models, this would be better applied to individual frames as the real-time system would demand a response from the program frame by frame from the camera at least 1/20 frames


```
tf.keras.applications.ResNet50(
  include_top=True, weights='imagenet', input_tensor=None, put_shape=None,
  pooling=None, classes=1000, **kwargs
)
```

(*fig.1 ResNet50 in Tesnsorflow*[25])

Dependant on the frame count on the camera Posenet also offers scalable accuracy and therefore processing time, this works in tandem with Nvidia CUDA® [18] offering "parallel computing platform and programming model developed by NVIDIA for general computing on graphical processing units (GPUs). With CUDA, developers are able to dramatically speed up computing applications by harnessing the power of GPUs." making use of parallel acceleration and running hardware based off of CUDA cores and Tensorflow's options for scalability means that the system will be able to operate in real-time using a portable device such as a laptop or can be better analysed on a stationary workstation with a CUDA enabled GPU.
Using hardware currently available to me I will be able to run Posenet models using a stationary workstation with 1920 CUDA cores at 1506 MHz fully capable of appropriate ResNet50 analysis.
For lightweight portability using a laptop running a real-time Posenet model on 750 CUDA cores at 1354 MHz if this does not comply to a real-time standard it will be scaled back to use 'MobileNetv1' which I better suited to portable applications and hardware such as mobile and laptops.

(*fig.2 MobileNetV1 in Tensorflow*[26])

```
tf.keras.applications.MobileNet(
    input_shape=None, alpha=1.0, depth_multiplier=1, dropout=0.001,
    include_top=True, weights='imagenet', input_tensor=None, pooling=None,
    classes=1000, classifier_activation='softmax', **kwargs
)
```

In order to reduce costs as proposed in the motivation(see sect whatever here) it is important that this system is portable as budget fencing occurs in multiple non-standardised 'piste' this would make a laptop the ideal hardware target for my application as it is portable contains it's own reserve of power and usually has an integrated web camera. Using this information also to determine and average framerate of the feed meaning optimizations to the model to record, process and execute the Posenet model at an average of 50hz 30 fps at 720p however this should not impose limits on external cameras capable of greater framerate and resolution recording.

2.1.b Object Detection of precise and fast objects

The main aim of precise object detection in my project will be for hit detection, therefore it is important that I have chosen the correct model not only for computing at an acceptable speed but also producing real-time and accurate results.

Detection in computer vision is still a developing field there are many competing models and

software for different scenarios and many standards working towards unifying and creating definitive models that can be used universally, for this project the usefulness of these models will be compared with the benefits that they can offer fencing and as a result a wider comparison to sports that feature similar degrees of action/movement.
Object detection in the context of Computer vision is useful to detect many or just few precise objects, using standard models like R-CNN to extract, compute and classify regions. Standards like 'SSD'[19] aim to reduce computational complexity for real time applications "presents the first deep network based object detector that does not resample pixels or features for bounding box hypotheses and is as accurate as approaches that do. This results in a significant improvement in speed for high-accuracy detection" In sporting applications it is important to distinguish fast objects accurately in fencing this would constitute tracking both players and most importantly the small fast moving point of the épée will be key to identify and track as it is the point of contact that increments the score. This needs to happen simultaneously to the opponent's point being tracked. Simultaneous object detection and tracking is supported by most models available through community resources, to do this with precision and in real time is computationally intense. Using the previously mentions 'SSD' to perform these operations would allow a framerate of 59fps
"Our real time SSD300 model runs at 59 FPS, which is faster than the current real time YOLO alternative, while producing markedly superior detection accuracy." Suiting sporting applications since 60 FPS is a popular video framerate standard and will allow the stream to have fast updates to the bounding box of the object. A greater frame count can also make use of standards that use an estimation between each frame to better determine the location of the object in the next few frames helping to increase precision and effectiveness of the tracking and identification.
Noisy environments and large field of view feeds can cause significant difficulty for accurate objects and correct/relevant objects to be detected. Objects that look very similar to the trained target could be detected and set their own bounding box which would obscure the important information; mistaking spare swords in the background as the appropriate tracking target. Using a large field of view in addition to creating a lot of noise for the model to distinguish between also introduces a lot of other objects that can also obscure the focus of the model; tracking spectators instead of the players. All of this can also be encapsulated within computational delay once we compute bounding objects it is important to update in real-time to produce a feed, the process of playing a sport features many different game states the need for minimal computational delay is required "Detecting an object with minimum delay is desired, and detection after certain time is no longer important"[20] applying to all real-time applications of object detection. For this reason, a feasible approach to utilizing and developing a model for an épée fencing application would use SSD as a model for object detection due to its low computation time and expected maximum framerate allowing access to 60 FPS cameras.

3 Structure of my Fencing System

3 Planned structure and functionality

As previously mentioned my system will have two primary halves that each encapsulate features that are geared towards Coaching and Refereeing. It is important that each are defined, explained and documented to aid in development and give structure to the resulting system. Due to the system being made for sporting budget and training purposes it should be easily accessible and usable by users with a mid-level understanding of current technology and recording equipment.

Each of the components are ordered in a priority of implementation, not meaning that one is more important than the other but to aid ease of implementation when it comes to building each component of the project.

The following are the necessary components:

3.a -Component 1 | Recording and frame analysis

The system will be reliant on a live feed of data meaning that recording will be essential, the system will simply be able to record information given the permission through the webcam and or external camera of the users device, the feed used will be the direct input to each computer vision model, in order to analyse a frame the same can be done but the input format will be a compatible image format and resolution, this must be done with some hindsight as it will need to be a capture from the game or frame of the recording of the game which can be screenshotted at any moment by the user.

3.b -Component 2 | Object detection overlay

During play an overlay of object detected bounding boxes will be drawn over any input given to the system, these bounding boxes must be able to update quickly and occupy precise regions of the input this will be done using the SSD model under tensorflow's 'mobilenetv2'[21] allowing fast and precise updates on mostly mobile hardware. Due to the overlay sometimes being obtrusive to some viewing the system will allow the overlay to be toggled on and off so that the game can be observed and analysed with and without either pose estimation or object detection

3.c -Component 3 | Pose estimation overlay

Similarly this will contribute as the other half to the system overall, using tensorflow's 'Posenet' model and parallel gpu acceleration with Nvidia 'CUDA' allowing both the option for faster drawing and response on greater workstation hardware but also scalable performance using mobilenet with performance checking[22], this feature will map points to each player and be able to determine the pose that they are in, a crucial technique in fencing that will give valuable feedback to coaches about the form of a certain player. Like the object detection component the overlay for this should also be toggleable as point mapping can obscure the view of the game/player and or interfere with the visual data of object detection.

3.d -Component 4 | Hit registration and accuracy

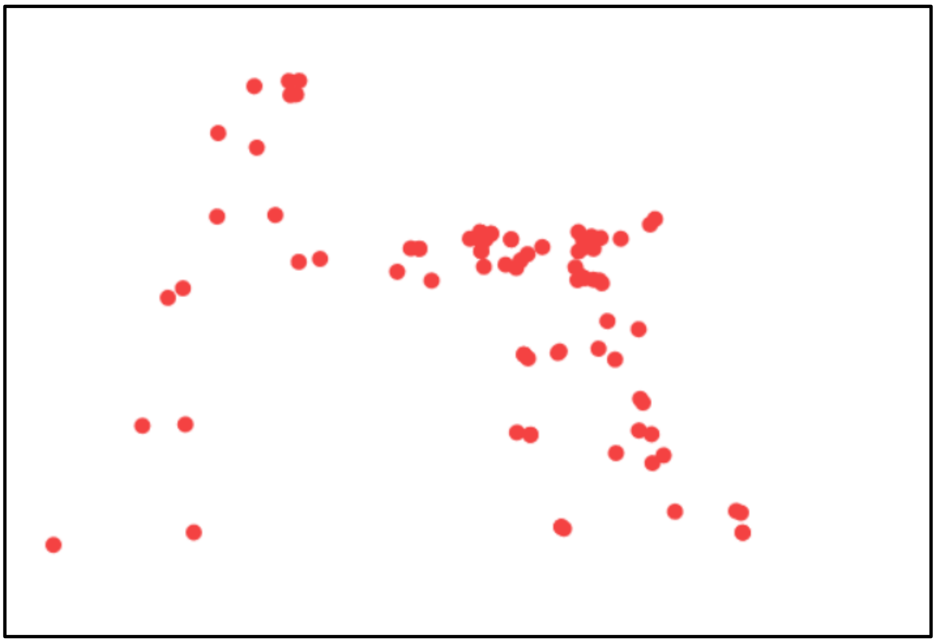
In order to detect a hit for refereeing purposes a blend of both techniques will be used to provide and accuracy score of if a hit has or would have connected. This component will use object detection to identify the curvature of a standard épée comparing it to the live-feed and then distinguishing how closely it matches a likely hit using the bend in the blade to define a characteristic of a hit. Combined with this accuracy statistic pose estimation will also utilise locational data of the points and bones to define the likelihood of the minimum distance for a hit to be possible. Together this will produce an accuracy score which will help contribute to a debate and ruling on referee decision when it comes to potential hits/no hits.

3.1 Example of functionality

This example will be displaying the applicability in fencing of using such a model, using posenet from tensorflow and this example project website[23] I have generate a new image from the source image:

This is a standard yet complex image of a fencing bout, utilising pose estimation the points can be mapped to each player to produce a fuzzy yet relevant pose mapping
(fig.3 Two épée *Fencers on Piste* [24]*)*

The pose estimation has created this point mapping showing the rough joints and outline of each fencer, there are small clusters of points close to where the model is less confident, this would not be uncommon in fencing due to complex movement however this can be reduced by setting a confidence threshold.
(fig.4 Point mapped image of the Fencers[23])



Here the model has drawn bones between the points to produce an expected skeleton and pose, this technique needs some refinement that can be done with tweaking of the tensorflow model to reduce clusters and produce more accurate skeletons (limit bones, change height) *(fig.5 Bones drawn between points with confidence reduction*[23])

This image was particularly tricky for pose estimation to map points to for several reasons, the perspective is at a slight angle to the right, the movement is complex, and the height of both figures greatly differs. This is a good test of the model's ability and many of these problems will be reduced where possible. Using this tensorflow model with adjustment and object detection we can produce accurately mapped poses and better confirm the players pose better creating mapped and accurate data to draw conclusions from, to be used as a supplement to a human referee and with the advice from a human coach also.

3.2 Risk analysis

Development of this project comes with some risk to completion from multiple areas. Computer vision is still a developing field and for that reason a potential risk may be limiting factors of what machine learning is capable of, there will always be inaccuracy to some degree with this technology however it will be my responsibility to deal with some of that appropriately by mitigation or acknowledgment for this reason there is a risk that the system may not be as refined as expected and therefore impact its effectiveness as a tool and aid to coaching and refereeing. External risks that I can not fully be accountable for are also present as of writing 10/2020 there is an on-going pandemic that has caused much of the UK and currently Wales to be locked down and Swansea to be deemed a high-risk area. If this pandemic Is to continue as projected there is no guarantee that I may become at risk of catching the virus while outdoors and as a result be less capable of finishing my project, necessary precautions will be taken as they have been put into place but this does not assure safety. Not only physical safety has been put at risk of the virus but also mental wellbeing affecting productivity, sleep and other natural habits.

4 Appendix & References

4 Appendix

background information on Fencing (épée)
épée is one of three sword classes

My proposed system will be designed for the épée category of Fencing the rules of which are outlined in the 'FIE Technical Rules'[1] [[t.1]t.91 – t.94] are particularly relevant to this project. The general rules and play of this game are as follows:

-Each player starts on their line(14 metres) of the piste(playing area).
-Each player coming to an 'En Garde' [[1]t.22] position before starting (sword facing towards the others guard just above hip height for épée).
-The referee will call to indicate the start of the bout.
-Any contact with the point to the opponents body is a valid hit.
-Invalid hits occur on Tip to Tip, Tip to Piste or Tip to guard or blade[[1]t.93-t.94].
-If both contacts go off at the same time and is indicated by both lights illuminating on the box this is a 'double hit' meaning both players are awarded a point[[1]t.92].
-If no hits are scored within 1 minute or in a team 3 minutes no points are awarded.

These make up some of the essential rules for épée fencing, this information will be referenced throughout the project as it is important to the technical content of development.

4.1 References

[1] 2019. *FIE GENERAL RULES AND RULES COMMON TO THE THREE WEAPONS*. [PDF-Rules] fie, pp.1-38. Available at: <https://static.fie.org/uploads/23/119683-Technical%20rules%20ang%20%281%29.pdf> [Accessed 20 October 2020].

[2] Stone, S., 2020. Jordan Pickford challenge on Virgil Van Dijk will not bring retrospective punishment. *BBC Sport*, [online] p.1. Available at: <https://www.bbc.co.uk/sport/football/54601721> [Accessed 20 October 2020].

[3] IFAB International Football Associations Board. 2020. *Historic Step For Greater Fairness In Football*. [online] Available at: <https://www.theifab.com/news/historic-step-for-greater-fairness-in-football> [Accessed 20 October 2020].

[4] Kohn, B., 2020. *Saber_Box*. [online] GitHub Saber_Box. Available at: <https://github.com/BenKohn2004/Saber_Box> [Accessed 18 October 2020].

[5] Kohn, B., 2020. *ROW Model*. https://drive.google.com/file/d/1SCLmuZvzmvtNv4HZFZcxSmPLbnsbn5C8/view: Saber_Box.

[6] Kohn, B., 2020. *Bell Guard*. https://drive.google.com/file/d/1SCLmuZvzmvtNv4HZFZcxSmPLbnsbn5C8/view: Saber_Box.

[7] ST37. 2019. *ST37 - Sportech 37*. [online] Available at: <https://st37.fr/> [Accessed 19 October 2020].

[8] st37, 2020. *Fencing | Video Refereeing | AI Best Referee Assistant*. [video] Available at: <https://www.youtube.com/watch?v=_ZpqbS9v_l4> [Accessed 21 October 2020].

[9] Takahashi, M., Yokozawa, S., Mitsumine, H. *et al.* Real-time visualization of sword trajectories in fencing matches. *Multimed Tools Appl* **79,** 26411–26425 (2020). https://doi.org/10.1007/s11042-020-09249-y

[10] Anusha, G. and Golden Julie, E., 2014. Improving the Performance of Video Tracking Using SVM. *International Journal of Engineering Trends and Technology(IJETT)*, [online] 11(3). Available at: <http://ijettjournal.org/volume-11/number-3/IJETT-V11P226.pdf> [Accessed 20 October 2020].

[11] Thomas, G., Gade, R., Moeslund, T., Carr, P. and Hilton, A., 2017. Computer Vision and Image Understanding. *Computer Vision in Sports*, [online] 159, pp.3-18. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S1077314217300711> [Accessed 22 October 2020].

[12] J. Han, L. Shao, D. Xu and J. Shotton, "Enhanced Computer Vision With Microsoft Kinect Sensor: A Review," in *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1318-1334, Oct. 2013, doi: 10.1109/TCYB.2013.2265378.

[13] Xing, X., 2020. *Tensor Flow Javascript Models*. https://github.com/tensorflow/tfjs-models/tree/master/posenet: TesnsorFlow.

[14] Thomas, Graham & Gade, Rikke & Moeslund, Thomas & Carr, Peter & Hilton, Adrian. (2017). Computer vision for sports: Current applications and research topics. Computer Vision and Image Understanding. 159. 10.1016/j.cviu.2017.04.011.

[15] Wikiwand. 2020. [online] Available at: <https://www.wikiwand.com/en/Parry_(fencing)> [Accessed 26 November 2020].

[16] TensorFlow. 2015. *Tesnsorflow ML*. [online] Available at: <https://www.tensorflow.org/> [Accessed 1 November 2020].

[17] Xing, X., 2019. *Pose Net*. https://github.com/tensorflow/tfjs-models/tree/master/posenet: TensorFlow.

[18] CUDA Zone. 2015. *Nvidia CUDA*. [online] Available at: <https://developer.nvidia.com/cuda-zone> [Accessed 27 October 2020].

[19] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[20] H. Mao, X. Yang and B. Dally, "A Delay Metric for Video Object Detection: What Average Precision Fails to Tell," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019, pp. 573-582, doi: 10.1109/ICCV.2019.00066.

[21] GitHub. 2020. *Object Detection*. [online] Available at: <https://www.tensorflow.org/hub/tutorials/object_detection> [Accessed 28 November 2020].

[22] Tesnsorflow. 2018. Performance Measurement. [online] Available at: <https://www.tensorflow.org/lite/performance/measurement> [Accessed 29 October 2020].

[23] Observable. 2019. Estimating human pose in the browser with Posenet and Tensorflow.js. [online] Available at: <https://observablehq.com/@j0000r111/estimating-human-pose-in-the-browser-with-posenet-and-tenso> [Accessed 30 October 2020].

[24] Marie-Lan Nguyen, (2012), *Final of the Challenge Réseau Ferré de France–Trophée Monal 2012, épée world cup tournament in Paris.* [ONLINE]. Available at: <https://upload.wikimedia.org/wikipedia/commons/1/10/Final_Trophee_Monal_2012_n08.jpg> [Accessed 31 October 2020].

[25] 2018. *Tf.Keras.Applications.Resnet50*. https://www.tensorflow.org/api_docs/python/tf/keras/applications/ResNet50: TensorFlow.

[26] 2018. *Tf.Keras.Applications.MobileNet*. https://www.tensorflow.org/api_docs/python/tf/keras/applications/MobileNet:TensorFlow.